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

© Parisa Alvandi 2017

Follow this and additional works at: https://ir.lib.uwo.ca/etd

Part of the Theory and Algorithms Commons

Recommended Citation
https://ir.lib.uwo.ca/etd/4565

This Dissertation/Thesis is brought to you for free and open access by Scholarship@Western. It has been accepted for inclusion in Electronic Thesis and Dissertation Repository by an authorized administrator of Scholarship@Western. For more information, please contact tadam@uwo.ca, wlswadmin@uwo.ca.
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.
Acknowledgements

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

**Abstract**

**Acknowledgments**

**List of Algorithms**

**List of Figures**

**List of Tables**

## 1 Overview

1.1 Goals .......................................................... 8

1.2 Thesis accomplishments ........................................... 9

1.2.1 Computing limit points of quasi-components of regular chains of dimension one. ........................................... 10

1.2.2 Improving the extended Hensel construction. ................. 10

1.2.3 Computing the real limit points of the quasi-component of a regular chain of dimension one. ........................................... 11

1.2.4 Studying regular chains under changes of coordinates. .......... 11

1.2.5 Introducing new tools for computing tangent cones of space curves. ................. 11

1.2.6 Computing limit of real multivariate rational functions. .......... 11

1.2.7 Separating the real and complex branches of space curves. .......... 12

1.2.8 Thesis contribution in RegularChains and PowerSeries libraries. .......... 13

1.3 Contribution statement ........................................... 14

1.4 Thesis outline .................................................. 14

## 2 Background and Related Work

2.1 Solving polynomial systems ....................................... 16

2.1.1 Limit points .................................................. 16

2.2 Power series and Puiseux expansions ............................. 21

2.3 The problem and related work .................................... 23

## 3 Extended Hensel Construction

3.1 Introduction .................................................. 26

3.2 Extended Hensel construction ................................... 28

3.2.1 Extended Hensel construction of multivariate polynomials .......... 32

3.2.2 Complete factorization in $\mathbb{C}((Y))[X]$ ........................ 34
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3</td>
<td>On the Yun-Moses polynomials</td>
<td>34</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Computing the $W_\lambda$</td>
<td>37</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Complexity analysis</td>
<td>38</td>
</tr>
<tr>
<td>3.4</td>
<td>Lifting the factors</td>
<td>39</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Complexity analysis</td>
<td>41</td>
</tr>
<tr>
<td>3.5</td>
<td>Experimentation</td>
<td>41</td>
</tr>
<tr>
<td>4</td>
<td>Computing Limit Points via Puiseux Series Expansions</td>
<td>44</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>44</td>
</tr>
<tr>
<td>4.2</td>
<td>Preliminaries</td>
<td>47</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Basic techniques</td>
<td>49</td>
</tr>
<tr>
<td>4.3</td>
<td>Puiseux expansions of a regular chain</td>
<td>52</td>
</tr>
<tr>
<td>4.4</td>
<td>Puiseux parametrization in finite accuracy</td>
<td>54</td>
</tr>
<tr>
<td>4.5</td>
<td>Computing in finite accuracy</td>
<td>56</td>
</tr>
<tr>
<td>4.6</td>
<td>Accuracy estimates</td>
<td>60</td>
</tr>
<tr>
<td>4.7</td>
<td>Algorithm</td>
<td>62</td>
</tr>
<tr>
<td>4.8</td>
<td>Experimentation</td>
<td>64</td>
</tr>
<tr>
<td>4.9</td>
<td>Concluding remarks</td>
<td>65</td>
</tr>
<tr>
<td>5</td>
<td>Real Limit Points of Space Curves</td>
<td>66</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>66</td>
</tr>
<tr>
<td>5.2</td>
<td>Real limit points</td>
<td>68</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Real branches of bivariate polynomials</td>
<td>70</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Real branches of space curves</td>
<td>73</td>
</tr>
<tr>
<td>5.3</td>
<td>Experimentation</td>
<td>74</td>
</tr>
<tr>
<td>6</td>
<td>Computing Limit Points via Changes of Coordinates</td>
<td>76</td>
</tr>
<tr>
<td>6.1</td>
<td>Introduction</td>
<td>76</td>
</tr>
<tr>
<td>6.2</td>
<td>Preliminaries</td>
<td>77</td>
</tr>
<tr>
<td>6.3</td>
<td>Algorithm for linear change of coordinates</td>
<td>78</td>
</tr>
<tr>
<td>6.3.1</td>
<td>The PALGIE algorithm for the prime case</td>
<td>80</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Regularity test in IsRegular($p, C, R$)</td>
<td>89</td>
</tr>
<tr>
<td>6.3.3</td>
<td>The PALGIE algorithm for linear change of coordinates</td>
<td>89</td>
</tr>
<tr>
<td>6.4</td>
<td>Noether normalization and regular chains</td>
<td>90</td>
</tr>
<tr>
<td>6.5</td>
<td>Applications of random linear changes of coordinates</td>
<td>91</td>
</tr>
<tr>
<td>6.6</td>
<td>On the computation of $\text{lim}(W(T))$ and $\text{sat}(T)$</td>
<td>93</td>
</tr>
<tr>
<td>6.7</td>
<td>Conclusion</td>
<td>99</td>
</tr>
<tr>
<td>7</td>
<td>Tangent Cones of Space Curves</td>
<td>100</td>
</tr>
<tr>
<td>7.1</td>
<td>Introduction</td>
<td>100</td>
</tr>
<tr>
<td>7.2</td>
<td>Preliminaries</td>
<td>101</td>
</tr>
<tr>
<td>7.2.1</td>
<td>Tangent cone of a space curve</td>
<td>102</td>
</tr>
<tr>
<td>7.2.2</td>
<td>Regular chains</td>
<td>103</td>
</tr>
<tr>
<td>7.3</td>
<td>Computing intersection multiplicities in higher dimension</td>
<td>104</td>
</tr>
<tr>
<td>7.4</td>
<td>Computing tangent lines as limits of secants</td>
<td>105</td>
</tr>
</tbody>
</table>
8 Computing Limits of Multivariate Rational Functions

8.1 Introduction ......................................................... 115
8.2 Preliminaries ......................................................... 118
  8.2.1 Lagrange multipliers ........................................... 118
  8.2.2 Regular chain theory ......................................... 119
  8.2.3 Parametric polynomial systems .............................. 120
  8.2.4 Triangular decomposition of semi-algebraic sets .......... 121
  8.2.5 Puiseux series .................................................. 123
8.3 Basic lemmas ......................................................... 123
8.4 Main Algorithm ...................................................... 126
8.5 Optimizations ......................................................... 130
8.6 Limits of multivariate rational functions: general case .... 131
8.7 Experimentation ....................................................... 134
8.8 Conclusion ........................................................... 135

9 Conclusion ............................................................. 137
  9.1 Computing limit points of quasi-components of regular chains 137
  9.2 Computing Puiseux expansions of bivariate polynomials .......... 138
  9.3 Computing tangent cones of space curves at their singular points 138
  9.4 Computing limits of real multivariate rational functions ....... 139
## List of Algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>EHC_Lift</td>
<td>32</td>
</tr>
<tr>
<td>NonzeroTerm</td>
<td>56</td>
</tr>
<tr>
<td>NewtonPuiseux</td>
<td>57</td>
</tr>
<tr>
<td>LimitPointsAtZero</td>
<td>63</td>
</tr>
<tr>
<td>LimitPoints</td>
<td>63</td>
</tr>
<tr>
<td>RealPuiseuxExpansions</td>
<td>72</td>
</tr>
<tr>
<td>RealRegularChainBranches</td>
<td>74</td>
</tr>
<tr>
<td>IsRegular($p, C, \mathcal{R}$)</td>
<td>81</td>
</tr>
<tr>
<td>Saturate($C, H, \mathcal{R}$)</td>
<td>82</td>
</tr>
<tr>
<td>Extend($C, D, \mathcal{R}$)</td>
<td>82</td>
</tr>
<tr>
<td>EnsureRank($p, \mathcal{R}, C, \mathcal{R}$)</td>
<td>83</td>
</tr>
<tr>
<td>EnsureLeadingCoefficient($p, v, \mathcal{R}, C, \mathcal{R}$)</td>
<td>83</td>
</tr>
<tr>
<td>Gcd($q, p, v, \overline{C}_v, \mathcal{R}, C, \mathcal{R}$)</td>
<td>84</td>
</tr>
<tr>
<td>Palgie($C, \mathcal{R}, \mathcal{R}$)</td>
<td>85</td>
</tr>
<tr>
<td>Closure($T$)</td>
<td>96</td>
</tr>
<tr>
<td>TangentCone</td>
<td>109</td>
</tr>
<tr>
<td>LimitAlongCurve</td>
<td>127</td>
</tr>
<tr>
<td>RandomEllipse</td>
<td>127</td>
</tr>
<tr>
<td>LimitInner</td>
<td>129</td>
</tr>
<tr>
<td>Limit</td>
<td>130</td>
</tr>
</tbody>
</table>
List of Figures

1.1 The commands `solve` and `RealRoot` ............................................. 2
1.2 The Steiner surface $S$ ................................................................. 3
1.3 $q(s,t) := s^3 + t^3 + s - t + 1 = 0$ does not have any real solutions. .......... 3
1.4 The image of the map $r$ is contained in the surface $S$ ............................. 4
1.5 The intersection of the image of the parametrization $r$ with plane $y = 1$. ........ 4
1.6 The intersection of Steiner surface $S$ with plane $y = 1$. ........................ 5
1.7 The intersection of Steiner surface $S$ with plane $y = 1$. ......................... 6
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 $r$ given by (1.2) and the plane $y = 1$. ................................................................. 7
1.9 Closure of $\text{Image}(r)$ ............................................................... 7
1.10 The tangent cone of the fish given by $f := y^2 - x^2(x + 4) = 0$ at the origin. .... 8
1.11 Intersection multiplicity ............................................................... 9
1.12 The surface defined by $q = z$ around the origin. ................................ 10
1.13 A surface and the paths represented by its discriminant variety .................. 12
1.14 Separating the branches of the curve given by $f := y(x - 2)^2 + x = 0$ .......... 13
1.15 `RegularChainBranches` command for regular chain $rc$ ....................... 13

2.1 The eve surface ............................................................................. 20
2.2 The real solutions of $V(F)$ computed by `RealTriangularize` .................... 20
2.3 The complex solutions of $V(F)$ computed by `Triangularize` .................... 21
2.4 Weierstrass Preparation Factorization and Extended Hensel construction ...... 22
2.5 Extended Hensel construction applied to a bivariate polynomial .................. 22

3.1 EHC applied to a trivariate polynomial ............................................. 26
3.2 Computational of limit points: complex and real cases ......................... 28
3.3 Comparison of three different implementations of EHC algorithm ............... 42

5.1 Computing the limit points of a regular chain: complex case vs real case ........ 67

7.1 The planner curve called "fish" and its tangent cone at the origin .................. 102
7.2 Limiting secants along $V(x^2 + y^2 + z^2 - 1, x^2 - y^2 - z)$. .................... 111
7.3 Secants along $V(x^2 + y^2 + z^2 - 1) \cap V(x^2 - y^2 - z(z - 1))$ limiting to $(0, 0, 1)$. 113

8.1 The graph corresponding to $f(x,y) = z$ ............................................. 133
8.2 Limit of real bivariate rational functions .......................................... 134
List of Tables

3.1 The total number of multiplications and additions of $M_1 M_2$ . . . . . . . . . 39
3.2 Comparing EHC versus Kung-Traub’s method (timings are in seconds) . . . . 43
4.1 Removing redundant components . . . . . . . . . . . . . . . . . . . . . . . . . . 64
5.1 Complex limit points vs real limit points . . . . . . . . . . . . . . . . . . . . 75
8.1 Comparisons of the commands limit, TestLimit, and RationalFunctionLimit . . 136
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.

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

---

> solve({x + y + z = 2, 2*x + y = 3, z = 1});
  {x = 2, y = -1, z = 1}

> solve({x + y + z^2 - 1, x + y^2 + z - 1, x^2 + y + z - 1});
  {x = 0, y = 0, z = 1}, {x = 0, y = 1, z = 0}, {x = 1, y = 0, z = 0}, {x = RootOf(_Z^2 + 2*_Z - 1), y = RootOf(_Z^2 + 2*_Z - 1)}

> realroot(x^8 + 5*x^7 - 4*x^6 + 20*x^3);
  \[
  \begin{bmatrix}
  -\frac{183}{32}, & -\frac{731}{128} \\
  \end{bmatrix},
  [0, 0]
  \]

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:

\[
r : \mathbb{R}^2 \rightarrow \mathbb{R}^3 \\
(s, t) \mapsto r(s, t)
\]

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

Consider the Roman surface or Steiner surface\(^2\) with implicit formula \( f = 0 \), where \( f \) is the following polynomials in the variables \( x, y, z \):

\[
f := 4x^4 - 8yx^3 + 9x^2y^2 - 8yzx^2 - 5y^3x + 8y^2zx + y^4 \\
-2y^3z + 3y^2z^2 - 2yz^3 + z^4 - 8yx^2 + 8zx^2 + 8y^2x \\
-8xyz - 2y^3 - 2yz^2 - 2y^2z + 4x^2 - 4yx + y^2. \tag{1.1}
\]

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

\[
r : \mathbb{R}^2 \rightarrow \mathbb{R}^3 \\
(s, t) \mapsto \left( \frac{s^2}{q(s, t)}, \frac{s^2 + t^2}{q(s, t)}, \frac{s^2 + s + t + t}{q(s, t)} \right), \tag{1.2}
\]

\(^2\)see [https://en.wikipedia.org/wiki/Roman_surface](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($r$) = $S$ 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($r$) ⊆ $S$ 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($r$) = $S$ does not hold. To do so, we shall compare the points of Image($r$) satisfying $y = 1$ with the points of $S$ satisfying $y = 1$.

First, we compute the intersection of the image of the parametrization $r$ 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:

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

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

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

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

Thus the parametrization $r$ 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 $r$ 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 $r$ with the polynomial
Chapter 1. Overview

Figure 1.4: The command Difference computes the points in the image of \( r \) that do not belong to surface \( S \), which is empty.

\[
\begin{align*}
&f := 4x^4 - 8y^3 + 9x^2 + 6y^2 - 8z^2 + 5y^3 + 8z + y = 0 \\
y^2 - z + x = 2y^2 + 3y = 2z - y_0^2 + 8y = 8z^2 + 3y = 8x = 8y = z = 0 \\
R := \text{PolynomialRing}([x, y, z]) ; \\
dec1 := \text{Triangulate}([f], R) ; S := \text{GeneralConstruct}([\text{map}\{\text{Initial}\}, \text{Equations}\{\text{dec1}\[1\], R, R}] ; \\
\text{dec1} := [\text{regular} \_ \text{cham}] \\
S := \text{constructible} \_ \text{set}
\end{align*}
\]

\( > \quad \text{lm1} := \text{Difference}(\text{imageR}, S, R) ; \text{IsEmpty}(\text{lm1}, R) ; \text{lm1} := \text{constructible} \_ \text{set} \)

True

Figure 1.5: The intersection of the image of the parametrization \( r \) with plane \( y = 1 \).

\[
\begin{align*}
R := \text{PolynomialRing}([x, y, z]) ; \\
q := s^2 + t^2 + s - t + 1 ; \\
F := [q \times - s^2, q \times - (s^2 + t^2), q \times - (s^2 + s + s + t)] ; \\
dec2 := \text{Triangulate(F, R)} ; \text{imageR} := \text{GeneralConstruct}([\text{map}\{\text{Initial}\}, F, R, R] ; \\
\text{dec2} := [\text{regular} \_ \text{cham}] \\
\text{imageR} := \text{constructible} \_ \text{set}
\end{align*}
\]

\( > \quad \text{lm1} := \text{Difference}(\text{imageR}, S, R) ; \text{IsEmpty}(\text{lm1}, R) ; \\
\text{lm1} := \text{constructible} \_ \text{set} \)

True

System \( R \) shown in Equation (1.3)

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

(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(\( r \)) 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 \( \text{Image}(r) \subseteq S \) holds or not, can be answered by

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

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

(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

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

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

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

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 $y = 1$.

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

Therefore, $(0, 0, 0)$ is a missing point, or in other words, a \textit{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 $z^2$:

\[
\begin{align*}
\left\{ \begin{array}{l}
z x - y^2 = 0 \\
y^5 - z^2 = 0 \\
z \neq 0
\end{array} \right. 
\end{align*}
\]

(1.5)

The following is a parametrization of the above system:

\[
\begin{align*}
x &= t^{4/5} \\
y &= t^{2/5} \\
z &= t
\end{align*}
\]

Therefore,

\[
\begin{align*}
\lim_{t \to 0} z &= \lim_{t \to 0} t = 0, \\
\lim_{t \to 0} y &= \lim_{t \to 0} t^{2/5} = 0, \\
\lim_{t \to 0} x &= \lim_{t \to 0} \frac{t^2}{t} = \lim_{t \to 0} \frac{(t^{2/5})^2}{t} = \lim_{t \to 0} t^{1/5} = \pm \infty.
\end{align*}
\]

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


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

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:

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

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.

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 \( R \subseteq \mathbb{Q}[x_1, \ldots, x_n] \), 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 Eu-
clidean topology and Zariski topology, respectively, we want to determine the set \( \text{lim}(Z_\mathbb{R}(R)) := \overline{Z_\mathbb{R}(R)} \setminus Z_\mathbb{R}(R) \) and \( \text{lim}(W(R)) := \overline{W(R)} \setminus W(R) \).

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 of a space curve at one of its singular
points.
1.2 Thesis accomplishments

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

---

6This package can be downloaded from [http://www.regularchains.org/downloads.html](http://www.regularchains.org/downloads.html)
1.2. Thesis accomplishments

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 $q := \frac{x^4+3x^2y-x^2-y^2}{x^2+y^2}$, 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 $\lim_{(x_1,...,x_n) \to (0,...,0)} q$ 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 $q := \frac{x^4+3x^2y-x^2-y^2}{x^2+y^2} = z$ 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 Figure 1.13. 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
1.2. Thesis accomplishments

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).

![Image](image.png)

```plaintext
> R := PolynomialRing([x, y]);
> rc := Chain([[x-y^2+2*y^2+2], y^2+2*y+1], Empty(R), R); Display(rc, R), Display(rc, R), 
> RegularChainBranches(rc, R, [y], coefficient = complex);
```

```
\[
\begin{align*}
& x^4 \cdot y^3 - y^2 = 0 \\
& -y^2 + y^2 + 2^2 = 0 \\
& \text{Coef} = 0
\end{align*}
\]

\[
\begin{align*}
& \left[ z = \sqrt{2}, y = \frac{1}{2} r^6 \left( -r^5 + 2 \cdot \text{RootOf}(z^2 + 1) \right), x = \frac{1}{8} r^2 \left( -r^3 + 6 \cdot \text{RootOf}(z^2 + 1) + 10 \right) + 8 \right],
\end{align*}
\]

\[
\begin{align*}
& \left[ z = \sqrt{2}, y = -\frac{1}{2} r^2 \left( r^2 + 2 \cdot \text{RootOf}(z^2 + 1) \right), x = \frac{1}{8} r^2 \left( r^2 + 6 \cdot \text{RootOf}(z^2 + 1) - 10 \right) - 8 \right],
\end{align*}
\]

\[
\begin{align*}
& z = \sqrt{2}, y = \sqrt{2} + 1, x = -\sqrt{2} \left( r^3 + 2 \right) + 1
\end{align*}
\]

```

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. 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

\[
\begin{align*}
f_1 &= x^2 + y + z - 10, \\
f_2 &= y^2 + x + z - 1, \\
f_3 &= z^2 + x + y - 1
\end{align*}
\]

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:

\[
\begin{align*}
R_1 &:= \begin{cases} 
    x - z = 0 \\
y - z = 0 \\
z^2 + 2z - 1 = 0
\end{cases}, & R_2 &:= \begin{cases} 
    x = 0 \\
y = 0 \\
z - 1 = 0
\end{cases}, & R_3 &:= \begin{cases} 
    x = 0 \\
y - 1 = 0 \\
z = 0
\end{cases}, & R_4 &:= \begin{cases} 
    x - 1 = 0 \\
y = 0 \\
z = 0
\end{cases}.
\end{align*}
\]
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 \( X^\alpha \). 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 \( k \) is denoted as \( k[X_1, \ldots, X_n] \), or \( k[X] \), and is called a polynomial ring over \( k \). In this thesis, we are mainly interested in polynomial rings over the real and complex numbers, that is, when \( k \) 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 \( \text{mvar}(f) \), and the leading coefficient of \( f \) w.r.t. \( \text{mvar}(f) \) is called the initial of \( f \), denoted by \( \text{init}(f) \). These notations support algorithms where the polynomial \( f \) is considered as a univariate polynomial in its main variable.

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

Let \( k \) be a field and \( \overline{k} \) be its algebraic closure. Since in this thesis, the field \( k \) will often be the field \( \mathbb{R} \) of the real numbers or the field \( \mathbb{C} \) of the complex numbers, we have \( \overline{k} = \mathbb{C} \). For a polynomial set \( F \subseteq k[X] \), 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:

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

**Example 2.** Let \( F = \{ X_1 + X_2^2, X_1^2 + X_2 \} \subseteq \mathbb{C}[X_1, X_2] \). 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 \( k[X] \) is said to be a triangular set, if for all \( f, g \in R \), with \( f \neq g \), we have \( \text{mvar}(f) \neq \text{mvar}(g) \). A variable \( X_i \) is said free w.r.t. \( R \) if there is no \( f \in R \) such that \( \text{mvar}(f) = X_i \), otherwise, it is said algebraic.

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

\[
R = \left\{ \begin{array}{l}
    r_2 := X_1X_3 - X_2^3 \\
    r_1 := X_2^4 - X_1^5
\end{array} \right\} \subseteq \mathbb{Q}[X_1, X_2, X_3].
\]

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

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

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

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

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

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 $\text{sat}(R)$, to be the ideal

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

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 $\langle 0 \rangle$.

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

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

2 modulo $I$.

**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 $\text{sat}(\{r_1, \ldots, r_{e-1}\})$. 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

$$ R := \{ X_1 X_3 - X_2^2, X_2^5 - X_1^2 \}. $$

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

---

2.1. Solving polynomial systems

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 \( S \subset \mathbb{k}^d \). The Zariski closure of the set \( S \), denoted as \( \overline{S} \), is defined to be \( V(I(S)) \) where
\[
I(S) = \{ f \in \mathbb{k}[X_1, \ldots, X_n] \mid f(a) = 0 \text{ for all } a \in S \}.
\]
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:
\[
W(R) = V(\text{sat}(R)).
\]

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

Example 5. Let \( F := \{5y^6 - 15y^5 + 15y^4 + 2xz^2 - 5y^3 + 5x^2 + 5z^2\} \subset \mathbb{Q}[z < y < x] \). 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 2.4. 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 \( F = \{ax + b, bx + y\} \) and the regular chain \( R_1 = \{bx + y, ay - b^2\} \). Using calculations in Maple, one can verify that we have
\[
V(F) = \overline{W(R)}.
\]
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:

\[
\left\{ \begin{array}{l}
x = \frac{-y}{b} \\
y = \frac{b^2}{a}
\end{array} \right.,
\]

for any complex value of \( a \) and \( b \) where \( ab \neq 0 \). 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

\[
\bar{W}(R_1) \setminus W(R_1) = V(R_2) \cup V(R_3),
\]

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.
2.2. Power series and Puiseux expansions

2.1.1 Limit points

Let \((X, \tau)\) 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 \(\text{lim}(S)\) the set of non-trivial limit points of \(S\).

Definition 9. For the regular chain \(R \subset \mathbb{k}\{X_1, \ldots, X_n\}\) a point \(p \in \bar{\mathbb{k}}^n\) 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 \(\text{lim}(W(R))\). 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

\[
\sum_{i=k}^{\infty} c_i x^{d_i},
\]

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

theorem, has a geometrical interpretation. These linear factors correspond to the branches of the plane curve $F(X_1, X_2) = 0$ 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 $F := X_1X_2^3 + X_2^2 + X_1 + 1$, over $\mathbb{C}(X_1)$, we obtain the following three linear factors:

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

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.

---

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.

---

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

---

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 $\text{sat}(R)$. The set $\lim(W(R))$ of the non-trivial limit points of $W(R)$ satisfies $V(\text{sat}(R)) = W(R) = W(R) \cup \lim(W(R))$. Hence, $\lim(W(R))$ is the set-theoretic difference $V(\text{sat}(R)) \setminus W(R)$. Deducing $\lim(W(R))$ or $V(\text{sat}(R))$ 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 $\text{sat}(R)$ 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...
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. Munford [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 Nother 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

\[^{10}\text{This package can be downloaded from http://regularchains.org/downloads.html}\]
2.3. THE PROBLEM AND RELATED WORK

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.

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

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, 46, 83]. See also the works of S. Abhyankar [11] 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
3.1. Introduction

\( k \) and total degree \( d \), we show that the Yun-Moses polynomials (needed when applying the EHC to \( F(X, Y) \)) can be computed within \( O(d^3 M(d)) \) operations in \( k \), where \( n \mapsto M(n) \) 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 \( O(k d M(d)^2) \) operations in \( k \) (instead of \( O(k^2 d M(d)^2) \) in a direct approach) or within \( O(k d M(d)) \) 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 \( O(d k M(k)) \) operations in \( \mathbb{C} \) using a linear lifting scheme (Theorem 5.2 in \([51]\)) and \( O(d M(k)) \) 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 \( O(d^2 k M(k)) \) and \( O(d^2 M(k)) \) 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 \( O(k^2 d M(d)) \) 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 \( \mathbb{C}(Y^\ast) \), computing \( k \) terms in each branch, is \( O(d^3 M(d) + k^2 d M(d)) \) for the EHC and \( O(d^2 k M(k)) \) (resp. \( O(d^2 M(k)) \)) 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 \( O(d^2 k M(k)) \) operations over \( \mathbb{C} \), using a linear lifting scheme, and the complexity of our EHC algorithm, to compute all branches, is \( O(d^3 M(d) + k^2 d M(d)) \) 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 \([73],[74]\) presented an algorithm in which the cost of computing the splitting field of \( f(X) \) is a polynomial to \( d \), \( c(G_f) \) and \( M(\log(B_B)) \) where \( G_f \) is Galois group of \( f \) and \( c(G_f) \) is \( c \)-size of it \([74]\), also \( B_B \) is such that \( \prod_{i=1} q_i = O(B_B) \) 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 \((f_1, \ldots, f_m)\) 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

\[
G^{(0)}_i(X, Y) := (X - \zeta_i Y^{\delta/d})^{m_i},
\]

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

**Puiseux series.** Let \( k \) be an algebraic number field and \( \overline{k} \) its algebraic closure. We denote by \( k[[Y]] \) and \( k(Y) \) the respective rings of formal power series and convergent power series in \( Y \) with coefficients in \( k \). We denote by \( k[[Y^*]] = \bigcup_{n \geq 1} k[[Y^n]] \) the ring of formal Puiseux series. Hence, given \( \varphi \in k[[Y^*]] \), there exists \( \ell \in \mathbb{N}_{>0} \) such that \( \varphi \in k[[Y^\ell]] \) holds and we can write \( \varphi = \sum_{m=0}^\infty a_m Y^\ell^m \), for some \( a_0, \ldots, a_m, \ldots \in k \). We denote by \( k((Y^*)) \) the quotient field of \( k[[Y^*]] \). Let \( \varphi \in k[[Y^*]] \) and \( \ell \in \mathbb{N} \) such that \( \varphi = f(Y^\ell) \) holds for some \( f \in k[[Y]] \). We say that the Puiseux series \( \varphi \) is convergent if we have \( f \in k(Y) \). The ring of convergent Puiseux series is denoted by \( k(Y^*) \) and its quotient field by \( k((Y^*)) \). We recall Puiseux’s theorem: if \( k \) is an algebraically closed field of characteristic zero, the field \( k((Y^*)) \) of formal Puiseux series over \( k \) is the algebraic closure of the field of formal Laurent series over \( k \); moreover, if \( k = \mathbb{C} \), then the field \( \mathbb{C}((Y^*)) \) 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 \( G_i(X, Y) \in \mathbb{C}((Y^*))[[X]] \) and \( \deg_X(G_i) = m_i \), for \( 1 \leq i \leq r \). Thus, the EHC factorizes \( F(X, Y) \) over \( \mathbb{C}((Y^*)) \). However, \( \deg_X(G_i) = 1 \) may not hold for some \( i \). Nevertheless, as shown after factorizing \( F(X, Y) \) into linear factors is achieved by repeated applications of the EHC. Lemma 2 and Theorem 2 are the fundamental results of the EHC.

**Lemma 2** (Yun-Moses polynomials). Let \( \hat{G}_i(X, \hat{Y}) \in \mathbb{C}(\hat{Y})[X] \), 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 \( d = \deg_X(\hat{G}_1 \cdots \hat{G}_r) \) 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 \( \{W^{(\ell)}_i(X, \hat{Y}) \in \mathbb{C}(\hat{Y})[X] \mid i = 1, \ldots, r\} \) satisfying

\[
W^{(\ell)}_1((\hat{G}_1 \cdots \hat{G}_r)/\hat{G}_1) + \cdots + W^{(\ell)}_r((\hat{G}_1 \cdots \hat{G}_r)/\hat{G}_r) = X^{\ell}\hat{Y}^{d-\ell},
\]

where \( \deg_X(W^{(\ell)}_i(X, \hat{Y})) < \deg_X(\hat{G}_i(X, \hat{Y})) \), \( i = 1, \ldots, r \). The polynomials \( W^{(0)}_1, \ldots, W^{(d-1)}_r \) for \( 1 \leq i \leq r \) are homogeneous in \((X, \hat{Y})\) of degree \( m_i \). We call Yun-Moses polynomials the elements of \( \{W^{(\ell)}_i \mid (\ell, i) \in \{0, \ldots, d-1\} \times \{1, \ldots, r\}\} \).
Proof. We shall first prove that there exists only one set of polynomials \([W_i^{(r)}(X, 1) \mid i = 1, \ldots, r]\) 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 \(A_1 \frac{\hat{G}_1 - \hat{G}_2}{a_1} + \cdots + A_r \frac{\hat{G}_1 - \hat{G}_r}{a_r} = 1\).

If we multiply both sides of the above equality by \(X^\ell\), then we have

\[
A_1X^\ell \frac{\hat{G}_1 \cdots \hat{G}_r}{\hat{G}_1} + \cdots + A_rX^\ell \frac{\hat{G}_1 \cdots \hat{G}_r}{\hat{G}_r} = X^\ell. 
\]  

(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 \(\text{deg}_X(R_i) < \text{deg}_X(\hat{G}_i)\).

Thus the last equality can be re-written as:

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

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

\[
\text{deg}_X(A_iX^\ell + \sum_{i=1}^{r-1} Q_i \hat{G}_i) < m_r = \text{deg}_X(\hat{G}_r).
\]

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

\[
\text{deg}_X(W_i^{(r)}(X, 1) (\hat{G}_1 \cdots \hat{G}_r) / \hat{G}_i) < d,
\]

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

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

\[
S_k = \langle X^d Y^{(k+1)/d}, X^d Y^{(k+2)/d}, \ldots, X^d Y^{(k+d)/d} \rangle, 
\]

(3.2)

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

\[
F(X, Y) = G_1^{(k)}(X, Y) \cdots G_r^{(k)}(X, Y) \mod S_{k+1},
\]

(3.3)

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

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

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

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

$$\Delta F^{(k)}(X, Y) = f^{(k)}_d x^{d_1} y^{d_1^1} + \cdots + f^{(k)}_0 x^{d_0} y^{d_0^1}$$

where $f^{(k)}_d = c^{(k)}_d y^{d_1^1}$ and $c^{(k)}_d \in \mathbb{C}$ for $\ell = 0, \ldots, d - 1$. We construct $G_i^{(k)}(X, Y)$, and thus $\Delta G_i^{(k)}$, $\Delta G_r^{(k)}$, 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:

$$F(X, Y) \equiv \prod_{i=1}^r \left( G_i^{(k-1)} + \Delta G_i^{(k)} \right) \mod S_{k+1}$$

$$\equiv G_i^{(k-1)} \cdots G_r^{(k-1)} + \Delta G_i^{(k)} \frac{F^{(0)}}{G_i^{(0)}} + \cdots + \Delta G_r^{(k)} \frac{F^{(0)}}{G_r^{(0)}} \mod S_{k+1}$$

$$\equiv G_i^{(k-1)} \cdots G_r^{(k-1)} + \Delta G_i^{(k)} \frac{F^{(0)}}{G_i^{(0)}} + \cdots + \Delta G_r^{(k)} \frac{F^{(0)}}{G_r^{(0)}} \mod S_{k+1}.$$ 

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

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

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

$$\sum_{i=1}^r \Delta G_i^{(k)} \frac{F^{(0)}}{G_i^{(0)}} = \sum_{\ell=0}^{d_1-1} f^{(k)}_\ell x^{\ell_1} y^{\ell_2} = \sum_{\ell=0}^{d_1-1} f^{(k)}_\ell \left( \sum_{i=1}^r W_i^{(f)} \frac{F^{(0)}}{G_i^{(0)}} \right)$$

$$= \sum_{i=1}^r \left( \sum_{\ell=0}^{d_1-1} f^{(k)}_\ell W_i^{(f)} \right) \frac{F^{(0)}}{G_i^{(0)}}.$$ 

Since $\deg_x(f^{(k)}_\ell W_i^{(f)}) < \deg_x(G_i^{(0)})$ and $\deg_x(\Delta G_i^{(k)}(X, Y)) < \deg_x(G_i^{(0)})$ both hold for $i = 1, \ldots, r$, we deduce $\Delta G_i^{(k)}(X, Y) = \sum_{\ell=0}^{d_1-1} W_i^{(f)}(X, Y) f^{(k)}_\ell(Y)$, for $i = 1, \ldots, r$. □

Remark 1. Theorem [1] still holds if $G_i^{(0)}(X, Y), \ldots, G_r^{(0)}(X, Y)$ just satisfy the same properties as $\tilde{G}_1(X, Y), \ldots, \tilde{G}_r(X, Y)$ of Lemma 2.
Remark 2. Write \( F(X, 0) = X^d + a_1X^{e_1} + \cdots + a_nX^{e_n} + a_{n+1} \). 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 \( a \in k \). 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

\[
\begin{align*}
\text{Input: } & \text{ a given } F \text{ as in Notation } [1] \text{ and a positive integer } k \\
\text{Output: } & \text{ Extended Hensel Construction on } F. \\
\text{begin} & \\
\text{Compute the Newton polynomial } F^{(0)} \text{ and } \hat{d}, \hat{\delta}; \\
\text{Compute } F^{(0)} = G_1^{(0)} \cdots G_r^{(0)}, \text{ see Remark } [1] \\
\text{if } r = 1 \text{ then} & \\
\quad \text{Apply the change of variable in Remark } [3] \\
\text{Compute the Yun-Moses polynomials } W_i^{(\ell)} \text{ for } i = 1, \ldots, r \text{ and } \ell = 0, \ldots, d - 1; \text{ (see Section } [3.3]; \\
\text{for } j = 1, \ldots, k \text{ do} & \\
\quad \text{Compute } \Delta F^{(j)}(X, Y) := F(X, Y) - \prod_{i=1}^{r} G_i^{(j-1)} \mod \hat{\delta}_{j+1} \text{ (see Section } [3.4] \text{ as well as Page 13 of [82]);} \\
\quad \text{Compute } \Delta G_i^{(j)} = \sum_{\ell=0}^{d-1} W_i^{(\ell)} f_\ell^{(j)}, \text{ for } i = 1, \ldots, r; \\
\quad \text{Let } G_i^{(j)} = G_i^{(j-1)} + \Delta G_i^{(j)} \text{ for } i = 1, \ldots, r; \\
\quad \text{Reverse the change of variable, if any; } \\
\text{return } G_1^{(k)}, \ldots, G_r^{(k)}; \\
\text{end} & \\
\end{align*}
\]

3.2.1 Extended Hensel construction of multivariate polynomials

Let \( F(X, Y_1, \ldots, Y_n) \in k[X, Y_1, \ldots, Y_n] \) 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 \( k(Y_1, \ldots, Y_n)[X, U] \).

Newton Line for multivariate polynomials. We plot each non-zero term \( c X^{e_1} Y_i^{e_{i1}} \cdots Y_n^{e_{in}} \) 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 \( \text{Newton Line} \) 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 \( e_s/d + e_U/\delta = 1 \) for some \( \delta \in \mathbb{Q} \). We define \( \delta, \delta \in \mathbb{Z}^* \) such that \( \delta/d = \delta/d \) and \( \gcd(\delta, \delta) = 1 \) 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 \textit{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

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

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

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

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:

\[
F^{(0)}(X, Y_1, \ldots, Y_n) = H_1^{(0)}(X, Y_1, \ldots, Y_n)^{m_1} \cdots H_s^{(0)}(X, Y_1, \ldots, Y_n)^{m_s},
\]

\[
\gcd(H_i^{(0)}, H_j^{(0)}) = 1, \quad \text{for any } i \neq j, \quad \text{and } \deg_X(H_i^{(0)}) = r_i
\]

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

\[
F^{(0)}_i = H_i^{(0)}(X, Y_1, \ldots, Y_n)^{m_i} \quad \text{for } i = 1 \ldots s.
\]

**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,

- \( W_1^{(\ell)} \left( F_1^{(0)} \cdots F_s^{(0)}/F_1^{(0)} \right) + \cdots + W_s^{(\ell)} \left( F_1^{(0)} \cdots F_s^{(0)}/F_s^{(0)} \right) = X^{\ell} \)

- \( \deg_X(W_i^{(\ell)}) < \deg_X(F_i^{(0)}), \quad i = 1, \ldots, s \)

For each \( i = 1, \ldots, s \), \( W_i^{(\ell)} \hat{U}^{d-\ell} \) are homogenous in \( X \) and \( \hat{U} \) where \( \hat{U} = U^{\delta/d} \) 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 multivariate 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 \( \text{(3.6)} \). 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 \( X^{e_U}/U^{e_U} \) in \( F_i^{(k)}(X, Y_1, \ldots, Y_n) \) is of the form \( N/D \) 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 $\mathbb{C}(\langle Y^\circ \rangle)[X]$

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 $\text{EHC}\_\text{Lift}(F, k)$ 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 $G := G^{(k)}_j(X, Y)$, 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 $G^{(k)}_j(X, Y)$, with the notations of Theorem [1] we apply the change of coordinates $X = X - \zeta Y$. 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 $\mathbb{C}(\langle Y^\circ \rangle)[X]$ 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 $\tilde{Y} = Y^{1/\hat{\delta}}$. 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 $\Delta F^{(k)} \equiv F - G^{(k-1)}_1 \cdots G^{(k-1)}_r \mod S_{k+1}$, where $G^{(k-1)}_i = G^{(0)}_i + \Delta G^{(1)}_i + \cdots + \Delta G^{(k-1)}_i$. And we have

$$\Delta F^{(k)}(X, \tilde{Y}) = f^{(k)}_d X^{d-1} \tilde{Y}^{\hat{\delta}} + \cdots + f^{(k)}_0 X^0 \tilde{Y}^{\hat{\delta}}$$

where $f^{(k)}_\ell = c^{(k)}_\ell \tilde{Y}^{\hat{\delta}}$ with $c^{(k)}_\ell \in \mathbb{C}$ for $\ell = 0, \ldots, d - 1$. The goal is to prove $c^{(k)}_\ell \in \mathbb{K}$ and we prove it by induction. For $k = 1$, $\Delta F^{(1)} \equiv F - F^{(0)} \mod S_2$. 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 $G^{(k-2)}_1 \cdots G^{(k-2)}_r = F - \Delta F^{(k-1)} \in \mathbb{K}[X, \tilde{Y}]$. We want to prove $\Delta F^{(k)} \in \mathbb{K}[X, \tilde{Y}]$. In modulo $S_{k+1}$, we have

$$\Delta F^{(k)} \equiv \Delta G^{(k-1)}_1 \cdots \Delta G^{(k-1)}_r \equiv \Delta G^{(k-1)}_1 \cdots \Delta G^{(k-1)}_r \equiv \Delta G^{(k-1)}_1 \cdots \Delta G^{(k-1)}_r + \sum_{i=1}^r \Delta G^{(k-1)}_i \frac{F^{(0)}}{G^{(0)}_i}.$$

Last equivalence is valid, due to $\Delta G^{(k-1)}_i \Delta G^{(k-1)}_j \equiv 0 \mod S_{k+1}$ and $(G^{(k-1)}_1 \cdots G^{(k-1)}_r) / G^{(k-1)}_i \equiv (G^{(0)}_1 \cdots G^{(0)}_r) / G^{(0)}_i \mod S_{k+1}$. On the other hand, $\Delta G^{(k-1)}_i = \sum_{\ell=0}^{d-1} W^{(\ell)} \tilde{F}_r^{(k-1)}$. So, we have

$$\sum_{i=1}^r \Delta G^{(k-1)}_i \frac{F^{(0)}}{G^{(0)}_i} = \sum_{i=1}^r \sum_{\ell=0}^{d-1} W^{(\ell)}_i f^{(k-1)}_\ell \frac{F^{(0)}}{G^{(0)}_i} = \sum_{\ell=0}^{d-1} f^{(k-1)}_\ell \sum_{i=1}^r W^{(\ell)}_i \frac{F^{(0)}}{G^{(0)}_i} = \sum_{\ell=0}^{d-1} f^{(k-1)}_\ell \tilde{Y}^{\hat{\delta}(d-\ell)},$$
therefore, modulo $S_{k+1}$, we have

$$
\Delta F^{(k)} \equiv F - (G_1^{(k-1)} \cdots G_r^{(k-1)} + \sum_{\ell=0}^{d-1} f_x^{(k-1)} X^d Y^{d-\ell}) 
\equiv F - (G_1^{(k-1)} \cdots G_r^{(k-1)} + \sum_{\ell=0}^{d-1} c_\ell^{(k-1)} X^d Y^{(k-1)d-\ell}).
$$

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

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

$$
\sum_{i=1}^r W^{(0)}_i F^{(0)}_i = X^\ell \hat{Y}^{d-\ell} \quad \text{for } \ell = 0, \ldots, d - 1,
$$

(3.7)

where $\hat{Y} = Y^{m_1}$ with $G_i^{(0)} = (X - \xi_i \hat{Y})^m$ where $\xi_i$ is a root of $F^{(0)}(X, 1)$ and $m_i$ is its multiplicity. Also, we have $\deg_X W^{(0)}_i < m_i$, thus, we write $W^{(0)}_i = \sum_{j=0}^{m_i-1} w_{i,j}(\hat{Y})X^j$ for any $\ell$. Let us fix $\lambda$ in $\{1, \ldots, r\}$. Define the column vector $X^{\ell}_{X} = [w^{(0)}_{i,j}]$. The goal is to find $X^{\ell}_{\hat{Y}}$, what we shall do by solving a system of linear equations. Now for $\mu = 0, 1, \ldots, m_1 - 1$, we take the $\mu$-th derivative of each side in Equation (3.7) and let $X = \xi_i \hat{Y}$ in those derivatives. In other words, we have

$$
\frac{\partial^\mu}{\partial X^\mu} \left( \sum_{i=1}^r W^{(0)}_i F^{(0)}_i \right)_{x = \xi_i \hat{Y}} = \frac{\partial^\mu}{\partial X^\mu} (X^\ell \hat{Y}^{d-\ell})_{x = \xi_i \hat{Y}}.
$$

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

$$
\frac{\partial^\mu}{\partial X^\mu} \left( W^{(0)}_i G^{(0)}_i \right)_{x = \xi_i \hat{Y}} = \frac{\partial^\mu}{\partial X^\mu} (X^\ell \hat{Y}^{d-\ell})_{x = \xi_i \hat{Y}}.
$$

Also we have $W^{(0)}_i = \sum_{j=0}^{m_i-1} w_{i,j} \hat{Y}X^j$, thus, we have

$$
\sum_{j=0}^{m_i-1} \frac{\partial^\mu}{\partial X^\mu} \left( X^j F^{(0)} G^{(0)} \right)_{x = \xi_i \hat{Y}} w^{(0)}_{i,j} \hat{Y} = \frac{\partial^\mu}{\partial X^\mu} (X^\ell \hat{Y}^{d-\ell})_{x = \xi_i \hat{Y}}.
$$

(3.8)

On the other hand, we know that

$$
\frac{\partial^\mu}{\partial X^\mu} \left( F^{(0)} G^{(0)} \right)_{x = \xi_i \hat{Y}} = \frac{1}{m_1!} \frac{\partial^{\mu+m_1}}{\partial X^{\mu+m_1}} (F^{(0)})_{x = \xi_i \hat{Y}}.
$$

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

$$
\mathcal{W}_\lambda = [\alpha_{j,\mu}] \text{ with } \alpha_{j,\mu} = \frac{\partial^\mu}{\partial X^\mu} \left( X^j F^{(0)} G^{(0)} \right)_{x = \xi_i \hat{Y}}.
$$

(3.9)
unknown vector $X'_{\alpha} = [w_{\alpha,j}^{(\ell)}]$ and constant vector

$$B_{\alpha}^{(\ell)} = [\beta_\mu] \text{ with } \beta_\mu = \left. \frac{\partial^\mu}{\partial X^\mu} (X'^d-\ell) \right|_{X = \zeta, \hat{\gamma}}$$

(3.10)

for $j, \mu = 0, 1, \ldots, m_\alpha - 1$. The matrix $\mathcal{W}_\alpha$ 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 $\left( X_j^d(\zeta) \right)_{|z = \zeta, \hat{\gamma}}$, for $j = 0, 1, \ldots, m_\alpha - 1$, are, indeed, linearly independent polynomials in $k(\zeta_i)[\hat{\gamma}]$, therefore, the Wronskian matrix $\mathcal{W}_\alpha$ is invertible.

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

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

$$M_{1j} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \frac{j_j!}{f' \cdots 1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \left( \frac{m_\alpha - 1}{j} \right) f^{(m_\alpha - 1)} & \cdots & 0 \end{bmatrix}$$

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 $\gamma_{j,k} = (-1)^{j+k} \binom{k}{j} \zeta_j^{k-j} \hat{\gamma}^{k-j}$ if $j \leq k$ and $\gamma_{j,k} = 0$ if $j > k$, for $j, k \in \{0, 1, \ldots, m_\alpha - 1\}$.

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

**Claim:** $M_2^{-1}$ is upper triangular with $\left( \frac{k}{k-j} \right) \zeta_j^{k-j} \hat{\gamma}^{k-j}$ as $(j, k)$-entry.

**Proof of the claim:** Let $A$ be the upper triangular matrix with $\left( \begin{array}{c} k \\ k-j \end{array} \right) T^{k-j}$ as $(j, k)$-entry where $T$ is a new variable. We show that $A|_{T=\zeta, \hat{\gamma}} \cdot M_2 = I$ where $I$ is the identity matrix of order $m_\alpha$. 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 $k \geq j$. This dot product is:

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

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

$$\left( \binom{k}{j} \right) \sum_{l=0}^{k-j} (-1)^{k+j+l} \binom{k-j}{l} T^l \zeta_j^{k-j-l} \hat{\gamma}^{k-j-l} = \binom{k}{j} (T - \zeta A \hat{\gamma})^{k-j}. \quad (3.11)$$
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 \( T = \zeta \hat{Y} \), it is 0. Hence, we have \( A|_{T=\zeta \hat{Y}} \cdot M_2 = I \) and \( M_2^{-1} = A|_{T=\zeta \hat{Y}} \), proving the claim.

Now, it is enough to show that \( M_2^{-1} = M_1 \cdot W_\lambda \), 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 \( N_{j-1} := M_{1(j-1)} \cdots M_{10} W_\lambda \). By multiplying \( M_{1j} \) by \( N_{j-1} \), we are factoring out \( f \) from the \((j + 1)\)-th row and adding \( (-1)^j f^{(k)} \) multiple of the \((j + k)\)-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, \( M_{1(m_{j-1})} \cdots M_{10} W_\lambda \) 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 \( W_\lambda \) is \( \frac{\partial^j}{\partial X^j} \left( X^k \frac{F^{(0)}}{G^{(0)}} \right) \) at \( X = \zeta \hat{Y} \), the \((j + 1, k + 1)\)-entry of \( M_{1(m_{j-1})} \cdots M_{10} W_\lambda \) is

\[
\frac{1}{j!} \frac{k!}{(k-j)!} \zeta^{k-j} \hat{Y}^{k-j} f = \binom{k}{k-j} \zeta^{k-j} \hat{Y}^{k-j},
\]

which is exactly \( M_2^{-1} \). This completes the proof. \( \square \)

Lemma 2 yields the following for Yun-Moses polynomials.

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

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

\[
\frac{\partial^\mu}{\partial X^\mu} \left( \frac{F^{(0)}}{G^{(0)}} \right) \bigg|_{X=\zeta \hat{Y}} \in \mathbb{K}(\zeta)(\hat{Y}).
\]

Hence, the coefficients of all entries of \( W_\lambda^{-1} \), defined in Proposition 1, live in \( \mathbb{K}(\zeta) \). Also, observe that the coefficients of all entries in matrix \( B_\lambda^{(f)} \) defined in (3.10) live in the same field \( \mathbb{K}(\zeta) \), therefore, \( w_{\lambda,j} \in \mathbb{K}(\zeta)(Y) \), for all \( j = 0, 1, \ldots, m_\lambda - 1 \). Hence \( W_\lambda^{(f)} \in \mathbb{K}(\zeta)(Y)[X] \). \( \square \)

### 3.3.1 Computing the \( W_\lambda \)

In this section, we discuss how we compute the Yun-Moses polynomials \( W_\lambda \). We regard each \( W_\lambda \) 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 \( \hat{Y} = Y^{\delta/d} \). Therefore, we need to compute the inverse of the Wronskian matrix \( \mathcal{W}_\lambda \) by computing \( M_{10}, M_{11}, \ldots, M_{1(m_{j-1})} \) and \( M_2 \) at \( X = \zeta \hat{Y} \). Since \( \frac{F^{(0)}}{G^{(0)}}(X, \hat{Y}) \) 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 \( \mathcal{W}_\lambda^{-1} = M_2 M_{10} \cdots M_{1(m_{j-1})} \), 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 \( B_\lambda^{(f)} \)
is \( d - j \) (see Section 3.3.2). Hence, computing the product \( W_n^{-1} = M_2 M_1 \cdots M_{1(m_1 - 1)} \) can be done as if those matrices had coefficients in \( \mathbb{K}(\zeta, \hat{\lambda}) \) rather than \( \mathbb{K}(\zeta, \hat{\lambda})[\hat{\lambda}] \).

### 3.3.2 Complexity analysis

Let \( f := F^{(0)}(X, \hat{\lambda})/(X - \zeta \hat{\lambda})^{m_1} \), where \( \zeta \) is a root of \( F^{(0)}(X, 1) \), and let \( d_f \) be the degree of \( f \) w.r.t. \( X \). So \( d_f = d - m_1 \). We use the notations of Proposition 1. After evaluation at \( X = \zeta \hat{\lambda} \), 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 \( j, k = 1, \ldots, m_1 \), with \( j \geq k \). Thus, in the \((j, k)\)-entry, the degree of \( \hat{\lambda} \) is \(-(d - m_1 + j - k)\). In \( M_2 \), the degree of \( \hat{\lambda} \) on the \((j, k)\)-entry is \( k - j \) for \( j, k = 1, \ldots, m_1 \) with \( k \geq j \). Hence, the \( \hat{\lambda} \)-degree in the \((j, k)\)-entry of \( W_n^{-1} \) is \( 2m_1 - d + k - j \).

In particular, \( M(n) \) is an upper bound for the number of operations in \( \mathbb{K}(\zeta, \hat{\lambda}) \) required for multiplying two univariate polynomials in \( \mathbb{K}(\zeta, \hat{\lambda}) \) with degree less than \( n \). Let \( A(n) \) 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: \( A(n) \in O(M(n)) \). Observe that the cost of evaluating \( f \) and its derivatives up to \( f^{(m_1 - 1)} \) is negligible. Let \( C_1 \) be the cost of constructing the matrices \( M_{10}, M_{11}, \ldots, M_{1(m_1 - 1)} \) and \( M_2 \). Assuming that \( 1/\zeta \) and all involved binomial coefficients are precomputed, we have:

\[
C_1 = \left( \frac{(m_1 - 1)m_1}{2} + \sum_{j=0}^{m_1 - 1} m_1 - j \right) A(d).
\]

The cost \( C_2 \) of multiplying \( M_{10}, M_{11}, \ldots, M_{1(m_1 - 1)} \) and \( M_2 \) is:

\[
C_2 = \left( \sum_{j=1}^{m_1 - 1} (m_1 - j)(2j - 1) + m_1 \sum_{j=1}^{m_1} 2(j - 1) - 2 \sum_{k=1}^{m_1} \sum_{j=1}^{k} j \right) A(d).
\]

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 \( \mathbb{k} \), it is sufficient to work with its irreducible factors over \( \mathbb{k} \), see Remark 1. Therefore, the cost \( C_{YM} \) of computing the Yun-Moses polynomials \( W^{(i)} \), for \( \ell \in \{0, \ldots, d - 1\} \), is given by \( C_{YM} = C_1 + C_2 = O(m^3 \text{M}(d)) \). This leads us to:

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

**Proof.** For constructing the matrices \( M_{10}, M_{11}, \ldots, M_{1(m_1 - 1)} \), we need, respectively, \( m_1, m_1 - 1, \ldots, 1 \) arithmetic calculations and therefore, we have \( \left( \frac{(m_1 - 1)m_1}{2} \right) A(d) \) as the total cost. Also for \( M_2 \), since it is an upper triangular matrix, it needs \( \sum_{j=0}^{m_1 - 1} m_1 - j \) arithmetic computations. Thus the total cost for constructing the matrices \( M_{10}, M_{11}, \ldots, M_{1(m_1 - 1)} \), and \( M_2 \) is

\[
C_1 = \left( \frac{(m_1 - 1)m_1}{2} + \sum_{j=0}^{m_1 - 1} m_1 - j \right) A(d).
\]

For multiplying matrices \( M_{10}, M_{11}, \ldots, M_{1(m_1 - 1)} \), due to sparsity of these matrices, there are \( \sum_{j=1}^{m_1 - 1} (m_1 - j)(2j - 1) \) multiplications and additions required to compute \( M_1 = M_{10}M_{11} \cdots M_{1(m_1 - 1)} \).
3.4 Lifting the factors

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

<table>
<thead>
<tr>
<th>Row $j$ of $M_1$</th>
<th>Column $k$ of $M_2$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>$\cdots$</th>
<th>$k$</th>
<th>$\cdots$</th>
<th>$m_A$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$2m_A$</td>
<td>$2(m_A-1)$</td>
<td>$2(m_A-2)$</td>
<td>$\cdots$</td>
<td>$2(m_A-k-1)$</td>
<td>$\cdots$</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$2(m_A-1)$</td>
<td>$2(m_A-1)$</td>
<td>$2(m_A-2)$</td>
<td>$\cdots$</td>
<td>$2(m_A-k-2)$</td>
<td>$\cdots$</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td></td>
</tr>
<tr>
<td>$j$</td>
<td>$2(m_A-j+1)$</td>
<td>$2(m_A-j+1)$</td>
<td>$2(m_A-j+1)$</td>
<td>$\cdots$</td>
<td>$2(m_A-k-j)$</td>
<td>$\cdots$</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td></td>
</tr>
<tr>
<td>$m_A$</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>$\cdots$</td>
<td>2</td>
<td>$\cdots$</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

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

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

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

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

Thus the cost of computing Yun-Moses polynomials $W^{(\ell)}_A$, for $\ell = 0, \ldots, d-1$, is $O(m_A^3M(d))$. Thus the total cost for computing all the Yun-Moses polynomials $W^{(\ell)}_A$, for $\ell = 0, \ldots, d-1$, and for $i = 1, \ldots, r$, is $O(d^3 M(d)).$ $\square$

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 $Θ(i)$ instead of $Θ(i^2)$.

Let $\tilde{Y} = Y^{1/d}$. Let $\Delta_i^k$ be such that $\Delta G_i^{(k)} = \Delta_i^k \tilde{Y}^k$ and define $\Delta_i^0 = G_i^{(0)}$. Therefore, $\Delta_i^k$, for $k > 0$, is homogeneous with respect to $(X, \tilde{Y})$ 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 \( P_j^k \equiv G_j^{(k-1)} \cdots G_j^{(0)} \mod S_{k+1} \). So, initially, we have \( P_j^1 \equiv G_j^{(0)} \cdots G_j^{(0)} \mod S_2 \), for \( j = 2, \ldots, r \).

For \( j = 2 \) and \( k > 1 \) we have

\[
P_2^k \equiv G_2^{(k-1)} \mod S_{k+1}, \text{ so}
\]

\[
P_2^k = \Delta_1^{(k)} \Delta_2^{(k)} + \left( \Delta_1^{(k)} \Delta_1^{(k-1)} + \Delta_2^{(k)} \Delta_1^{(k-1)} \right) \tilde{Y} + \cdots
\]
\[
+ \left( \Delta_1^{(k-2)} + \cdots + \Delta_2^{(k-1)} \Delta_1^{(k-1)} \right) \tilde{Y}^{k-1}
\]
\[
+ \left( \Delta_1^{(k-1)} \Delta_1^{(k-1)} + \cdots + \Delta_2^{(k) \Delta_1^{(k-1)}} \right) \tilde{Y}^k
\]
\[
+ \left( \Delta_1^{(k)} \Delta_1^{(k)} + \cdots + \Delta_2^{(k) \Delta_1^{(k)}} \right) \tilde{Y}^{k+1}.
\]

For the next iteration, that is from \( k \) to \( k + 1 \), we have:

\[
P_{2}^{k+1} \equiv G_2^{(k)} \mod S_{k+2}, \text{ so}
\]

\[
P_{2}^{k+1} = \Delta_1^{(k)} \Delta_2^{(k)} + \left( \Delta_1^{(k)} \Delta_1^{(k-1)} + \Delta_2^{(k)} \Delta_1^{(k-1)} \right) \tilde{Y} + \cdots
\]
\[
+ \left( \Delta_1^{(k-2)} + \cdots + \Delta_2^{(k-1)} \Delta_1^{(k-1)} \right) \tilde{Y}^{k-1}
\]
\[
+ \left( \Delta_1^{(k-1)} \Delta_1^{(k-1)} + \cdots + \Delta_2^{(k) \Delta_1^{(k-1)}} \right) \tilde{Y}^k
\]
\[
+ \left( \Delta_1^{(k)} \Delta_1^{(k)} + \cdots + \Delta_2^{(k) \Delta_1^{(k)}} \right) \tilde{Y}^{k+1}.
\]

If we assume that \( P_2^k \) has been computed and stored at the previous iteration, then it is enough to compute \( \Delta_1 \Delta_2 \Delta_3 \) and \( \Delta_1 \Delta_3 \) in the current iteration in order to deduce \( P_{2}^{k+1} \), with the following recursive formula:

\[
P_{2}^{k+1} = P_2^k + \left( \Delta_1 \Delta_2 + \Delta_1 \Delta_3 \right) \tilde{Y} + \left( \Delta_1 \Delta_3 + \cdots + \Delta_1 \Delta_2 \right) \tilde{Y}^k.
\]

Now for \( j = 3, \ldots, r \), define

\[
P_j^k = P_j^k \equiv P_j^{k-1} \mod S_{k+1}, \text{ so}
\]

\[
P_j^k = P_j^{k-1} \Delta_j^{(k-1)} + \left( P_j^{k-1} \Delta_j^{(k-1)} + P_j^{k-1} \Delta_j^{(k-1)} \right) \tilde{Y} + \cdots
\]
\[
+ \left( P_j^{k-1} \Delta_j^{(k-1)} + \cdots + P_j^{k-1} \Delta_j^{(k-1)} \right) \tilde{Y}^{k-1}
\]
\[
+ \left( P_j^{k-1} \Delta_j^{(k-1)} + \cdots + P_j^{k-1} \Delta_j^{(k-1)} \right) \tilde{Y}^k
\]
\[
+ \left( P_j^{k-1} \Delta_j^{(k-1)} + \cdots + P_j^{k-1} \Delta_j^{(k-1)} \right) \tilde{Y}^{k+1}.
\]

where \( P_j^{k-1} = P_j^{k-1} + P_j^{k-1} \tilde{Y} + \cdots + P_j^{k-1} \tilde{Y}^k \). Hence, we deduce:

\[
P_j^{k+1} = P_j^{k+1} \equiv P_j^{k+1} \Delta_j^{(k)} \mod S_{k+2}, \text{ so}
\]

\[
P_j^{k+1} = P_j^{k+1} \Delta_j^{(k)} + \left( P_j^{k+1} \Delta_j^{(k)} + P_j^{k+1} \Delta_j^{(k)} \right) \tilde{Y} + \cdots
\]
\[
+ \left( P_j^{k+1} \Delta_j^{(k)} + \cdots + P_j^{k+1} \Delta_j^{(k)} \right) \tilde{Y}^{k-1}
\]
\[
+ \left( P_j^{k+1} \Delta_j^{(k)} + \cdots + P_j^{k+1} \Delta_j^{(k)} \right) \tilde{Y}^k
\]
\[
+ \left( P_j^{k+1} \Delta_j^{(k)} + \cdots + P_j^{k+1} \Delta_j^{(k)} \right) \tilde{Y}^{k+1}.
\]

If we assume that \( P_j^k \) and \( P_j^{k-1} \) have been computed and stored at the previous iteration, then we can recycle some of the terms of \( P_j^k \) and \( P_j^{k-1} \) 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 \( P_j^{k+1} \Delta_j^{(k)} \) and \( P_j^{k+1} \Delta_j^{(k)} + \cdots + P_j^{k+1} \Delta_j^{(k)} \).
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

$$q_j^{k+1} = p_{j-1}^{k+1,0} \Delta_j^k + q_{j-1}^{k+1,0} \Delta_j^0$$

with $q_2^{k+1} = \Delta_2^k \Delta_1^0 + \Delta_2^0 \Delta_1^k$.

(3.12)

Now observe that we have

$$p_{j-1}^{k+1,k} = p_{j-1}^{k+1,0} \Delta_j^k + \cdots + p_{j-1}^{k,k,0} \Delta_j^0$$

$$= p_{j-1}^{k+1,0} \Delta_j^k + p_{j-1}^{k,1} \Delta_j^k + \cdots + p_{j-1}^{k,1} \Delta_j^1 + (p_{j-1}^{k,k} + q_j^{k+1}) \Delta_j^0$$

$$= p_{j-1}^{k+1,0} \Delta_j^k + p_{j-1}^{k,k} + q_j^{k+1} \Delta_j^0 = p_{j-1}^{k,k} + d_j^{k+1}.$$ 

Therefore, we can write

$$P_{j+1}^k = P_j^k + q_{j+1}^{k+1} Y_k + (p_{j-1}^{k+1,1} \Delta_j^k + \cdots + p_{j-1}^{k+1,k} \Delta_j^0) Y_k^{k+1}.$$ 

(3.13)

Note: the term $d_j^{k+1}$ 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 $P_j^k$, for $0 \leq \ell \leq k + 1$, is derived from $P_{j-1}^{\ell-1}$ and $P_{j-1}^{\ell}$ in a Pascal Triangle fashion. More precisely, letting $\ell = k + 1$, if $P_j^k$ and $P_{j-1}^{k+1}$ are known, computing $P_{j+1}^k$ 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 $P_j^\ell$ for $j = 1, \ldots, r$ and $\ell = 1, \ldots, k$ need to be computed before computing $P_r^{k+1}$. Let $C_{\text{lift}}$ be the cost of computing $P_r^{k+1}$. We have: $C_{\text{lift}} = \sum_{\ell=2}^r (k + 2)M(\max(m_1 + \cdots + m_{\ell-1}, m_\ell))A(d)$. This leads us to the following result.

**Theorem 4.** The $k$-th iteration of Step 9 in the Algorithm 1 runs in $O(k d M(d)^2)$ 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 [www.regularchains.org/papers/Benchmark-ISSAC-2017.zip](http://www.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 $\infty$ 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.
### Table 3.2: Comparing EHC versus Kung-Traub’s method (timings are in seconds)

<table>
<thead>
<tr>
<th>Ex</th>
<th>MD</th>
<th>KT Lin KT10</th>
<th>KT Lin KT20</th>
<th>KT Quad KT10</th>
<th>KT Quad KT20</th>
<th>EHCWM EHC10</th>
<th>EHCWM YM1</th>
<th>EHCWM EHC20</th>
<th>EHCEEA EHC10</th>
<th>EHCEEA YM2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>2.22 18.6</td>
<td>4.93 4.91</td>
<td>0.48 0.22</td>
<td>0.73 0.90</td>
<td>0.90 0.21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>5.60 65.8</td>
<td>0.56 0.58</td>
<td>0.22 0.14</td>
<td>0.23 0.36</td>
<td>0.34 0.13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>14.9 230</td>
<td>1.25 1.25</td>
<td>0.23 0.13</td>
<td>0.36 0.36</td>
<td>0.46 0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>5.53 114</td>
<td>1.51 1.56</td>
<td>0.30 0.11</td>
<td>0.39 0.29</td>
<td>0.88 0.10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>2.71 42.0</td>
<td>0.28 0.63</td>
<td>0.16 0.08</td>
<td>0.20 0.20</td>
<td>0.32 0.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td>0.46 2.34</td>
<td>0.21 0.21</td>
<td>0.16 0.08</td>
<td>0.17 0.17</td>
<td>0.26 0.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>0.50 6.86</td>
<td>0.28 0.32</td>
<td>0.16 0.08</td>
<td>0.18 0.18</td>
<td>0.30 0.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>4</td>
<td>0.86 10.9</td>
<td>0.50 0.48</td>
<td>0.26 0.15</td>
<td>0.28 0.28</td>
<td>0.46 0.24</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>4</td>
<td>3.21 34.8</td>
<td>0.69 0.71</td>
<td>0.26 0.15</td>
<td>0.34 0.34</td>
<td>0.52 0.24</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>6</td>
<td>27.6 535</td>
<td>4.85 4.85</td>
<td>0.64 0.42</td>
<td>0.82 0.82</td>
<td>2.05 1.08</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>7</td>
<td>45.6 836</td>
<td>8.45 9.91</td>
<td>0.64 0.43</td>
<td>0.92 1.17</td>
<td>2.33 1.74</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>7</td>
<td>145 ∞</td>
<td>23.4 23.2</td>
<td>0.78 0.43</td>
<td>3.37 3.82</td>
<td>4.12 1.77</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>4</td>
<td>0.14 0.16</td>
<td>0.16 0.14</td>
<td>0.39 0.26</td>
<td>0.45 0.45</td>
<td>0.51 0.15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>4</td>
<td>2.79 7.98</td>
<td>0.77 0.82</td>
<td>0.26 0.15</td>
<td>0.29 0.29</td>
<td>0.50 0.24</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>4</td>
<td>8.58 143</td>
<td>1.96 1.93</td>
<td>0.23 0.12</td>
<td>0.31 0.31</td>
<td>0.47 0.16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>5</td>
<td>2.90 24.8</td>
<td>1.11 1.11</td>
<td>0.26 0.15</td>
<td>0.35 0.35</td>
<td>0.49 0.17</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>7</td>
<td>1.83 9.45</td>
<td>0.90 1.00</td>
<td>0.46 0.31</td>
<td>0.50 0.50</td>
<td>0.73 0.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>8</td>
<td>2.35 12.3</td>
<td>3.09 3.29</td>
<td>0.66 0.53</td>
<td>0.74 0.74</td>
<td>2.18 1.80</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>8</td>
<td>60.8 2876</td>
<td>23.1 27.1</td>
<td>0.77 0.53</td>
<td>1.20 1.20</td>
<td>2.31 1.28</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>9</td>
<td>215 ∞</td>
<td>73.8 123</td>
<td>1.88 1.03</td>
<td>2.11 2.11</td>
<td>7.03 4.92</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>17</td>
<td>∞ ∞ ∞ ∞</td>
<td>∞ ∞ ∞ ∞</td>
<td>39.8 6.70</td>
<td>41.3 41.3</td>
<td>53.8 16.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>32</td>
<td>∞ ∞ ∞ ∞</td>
<td>∞ ∞ ∞ ∞</td>
<td>599 24.9</td>
<td>∞ ∞ ∞ ∞</td>
<td>∞ ∞ ∞ ∞</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>33</td>
<td>∞ ∞ ∞ ∞</td>
<td>∞ ∞ ∞ ∞</td>
<td>224 25.0</td>
<td>∞ ∞ ∞ ∞</td>
<td>∞ ∞ ∞ ∞</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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 \( k[X_1, \ldots, X_n] \) over a field \( k \), 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 \( \text{sat}(R) \), the saturated ideal of \( R \). In fact, the affine variety defined by \( \text{sat}(R) \) 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 \( R_1, \ldots, R_e \).
4.1. Introduction

\[ V(I) = \overline{W(R_1)} \cup \cdots \cup \overline{W(R_r)} \] while a decomposition of the form \( V(I) = W(R_1) \cup \cdots \cup W(R_f) \) 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 \( \text{sat}(R) \subseteq \text{sat}(S) \) holds or not. In the algebraic case, this inclusion can be tested by computing a set of generators of \( \text{sat}(R) \), 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 \( \text{sat}(R) \), 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 \( \text{sat}(R) \) has dimension one. The case \( \text{sat}(R) \) 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 \( k = \mathbb{C} \).
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 \( r_1 = X_1 X_2^2 + X_2 + 1, r_2 = (X_1 + 2)X_1 X_3^2 + (X_2 + 1)(X_3 + 1) \). We have \( W(R) = V(R) \setminus V(h_R) \) with \( h_R = X_1^2 (X_1 + 2) \). To determine \( \text{lim}(W(R)) \), we compute Puiseux series expansions of \( r_1 \) at \( X_1 = 0 \) and \( X_1 = -2 \). For such calculation, we use MAPLE’s command \texttt{algcurves[puiseux]} [95]. We start with \( X_1 = 0 \). The Puiseux expansions of \( r_1 \) at \( X_1 = 0 \) are:

\[
\begin{align*}
[X_1 = T, X_2 &= -\frac{T^2 - T}{T} + O(T^2)], \\
[X_1 = T, X_2 &= -\frac{1 + T^2 + T}{T} + O(T^2)].
\end{align*}
\]

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

\[
\begin{align*}
r_2' &= r_2(X_1 = T, X_2 = -\frac{T^2 - T}{T} + O(T^2), X_3) \\
&= (T + 2)TX_3^2 + (-T + O(T^2))(X_3 + 1)
\end{align*}
\]

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

\[
\begin{align*}
[T = T, X_3 &= 1 - 1/3T + O(T^2)], \\
[T = T, X_3 &= -1/2 + 1/12 T + O(T^2)].
\end{align*}
\]

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:

\[
\begin{align*}
[X_1 = T - 2, X_2 &= 1 + 1/3 T + O(T^2)], \\
[X_1 = T - 2, X_2 &= -1/2 - 1/12 T + O(T^2)].
\end{align*}
\]

After substitution into \( r_2 \), we obtain:

\[
\begin{align*}
r_{12}' &= r_2(X_1 = T - 2, X_2 = 1 + 1/3 T + O(T^2), X_3) \\
&= (T - 2)TX_3^2 + (2 + 1/3 T + O(T^2))(X_3 + 1) \\
r_{22}' &= r_2((X_1 = T - 2, X_2 = -1/2 - 1/12 T + O(T^2)) \\
&= (T - 2)TX_3^2 + (1/2 - 1/12 T + O(T^2))(X_3 + 1) .
\end{align*}
\]

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

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

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

**Relation between Zariski topology and the Euclidean topology.** When $k = \mathbb{C}$, the affine space $A^s$ 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: every nonempty Zariski open set is dense in the Euclidean topology on $A^s$. 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 $V \subseteq A^s$ 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 $(X, \tau)$ 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 $\text{lim}(S)$ the set of non-trivial limit points of $S$.

**Regular chain.** A set $R$ of non-constant polynomials in $k[X_1, \ldots, X_s]$ is called a triangular set, if for all $p, q \in R$ with $p \neq q$ we have $\text{mvar}(p) \neq \text{mvar}(q)$. A variable $X_i$ is said free w.r.t. $R$ if there exists no $p \in R$ such that $\text{mvar}(p) = X_i$. For a nonempty triangular set $R$, we define the saturated ideal $\text{sat}(R)$ 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 \( \text{sat}(R) \) has several properties, in particular it is unmixed \([19]\).

We denote its height, that is the number of polynomials in \( R \), by \( e \), thus \( \text{sat}(R) \) has dimension \( s - e \). Let \( X_{i_1} < \cdots < X_{i_r} \) 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 \( \text{sat}(\{ r_1, \ldots, r_{e-1} \}) \). The regular chain \( R \) is said strongly normalized whenever each of the main variables of the polynomials of \( R \) (that is, \( X_{i_1} < \cdots < X_{i_r} \)) 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 \( V(\text{sat}(R)) \), that is, the affine variety of \( \text{sat}(R) \). 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 \( k \) is an algebraically closed field. From now on, we further assume that \( k \) is topologically complete. Hence \( k \) 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 \( k[[X_1, \ldots, X_s]] \) and \( k\langle X_1, \ldots, X_s \rangle \) the rings of formal and convergent power series in \( X_1, \ldots, X_s \) with coefficients in \( k \). Note that the ring \( k\langle X_1, \ldots, X_s \rangle \) is a subring of \( k[[X_1, \ldots, X_s]] \). When \( s = 1 \), we write \( T \) instead of \( X_1 \). Thus \( k[[T]] \) and \( k(T) \) are the rings of formal and convergent univariate power series in \( T \) and coefficients in \( k \). For \( f \in k[[X_1, \ldots, X_s]] \), its order is defined by

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

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

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

Let \( f \in k[[X_1, \ldots, X_s]] \) 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 \( \text{ord}(f) \mod \mathcal{M}_{s-1} = k \).

Formal Puiseux series. We denote by \( k[[T^*]] = \bigcup_{n=1}^{\infty} k[[T^{1/n}]] \) the ring of formal Puiseux series.

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

Convergent Puiseux series. Let \( \varphi \in \mathbb{C}[[T^*]] \) and \( n \in \mathbb{N} \) such that \( \varphi = f(T^{1/n}) \) holds for some \( f \in \mathbb{C}[[T]] \). We say that the Puiseux series \( \varphi \) is convergent if we have \( f \in \mathbb{C}(T) \). Convergent Puiseux series form an integral domain denoted by \( \mathbb{C}(T^*) \); its quotient field is denoted by \( \mathbb{C}((T^*)) \). 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

\[
\varphi = \sum_{m=n}^{\infty} a_m T^{m/n} \quad \text{and} \quad a_n \neq 0.
\]
Then, we define \( \ord (\varphi) = \frac{q}{r} \).

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

**Puiseux expansion.** Let \( B = \mathbb{C}(\!(X)\!) \) or \( \mathbb{C}(\!(X)\!) \). Let \( f \in B[Y] \), where \( d := \deg(f, Y) > 0 \). Let \( h := \text{lc}(f, Y) \). According to Puiseux Theorem, there exists \( \varphi_i \in B \), \( i = 1, \ldots, d \), 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 \( f \in \mathbb{C}(X)[Y] \). A Puiseux parametrization of \( f \) is a pair \((\psi(T), \varphi(T))\) of elements of \( \mathbb{C}(T) \) 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 \( \mathbb{C}(T) \), and (3) there is no integer \( k > 1 \) such that both \( \psi(T) \) and \( \varphi(T) \) are in \( \mathbb{C}(T^k) \). The index \( \varsigma \) 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 \( \varsigma \) in \( \mathbb{C} \). Then \( \varphi(z_i X^{1/\varsigma}) \), for \( i = 1, \ldots, \varsigma \), are \( \varsigma \) Puiseux expansions of \( f \). For a Puiseux expansion \( \varphi \) of \( f \), let \( c \) minimum such that both \( \varphi = g(T^{1/c}) \) and \( g \in \mathbb{C}(T) \) 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 \( V(h_N) = V(h_R) \) 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 \( \varphi \in \mathbb{C}((T^*)) \) and let \( p/q \in \mathbb{Q} \) be the order of \( \varphi \). 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, \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 $\alpha$. 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 $\varphi_1, \ldots, \varphi_d \in \mathbb{C}(\langle X_1^\ast \rangle)$ such that the following holds

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

in $\mathbb{C}(\langle X_1^\ast \rangle)[X_2]$. 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 have

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

**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 $(\alpha_n, \varphi(\alpha_n)) \notin V(h_1)$ 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. \qed
Lemmas \[\text{I} \] and \[\text{II} \] immediately imply the following.

**Proposition 2.** With \( h_1(0) = 0 \), for all \( \beta \in \mathbb{C} \), we have
\[
(0, \beta) \in \text{lim}(W(r_1)) \iff \beta \in \{\varphi_1(0), \ldots, \varphi_c(0)\}.
\]

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 \((0, \beta, \gamma) \in \text{lim}(W(r_1, r_2))\).

Lemma \[\text{I} \] tells us that \((0, \beta, \gamma) \in \text{lim}(W(r_1, r_2))\) implies \((0, \beta) \in \text{lim}(W(r_1))\). This observation together with Proposition \[\text{II} \] 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
\[
(0, \beta, \gamma) \in \text{lim}(W(r_1, r_2)) \iff (0, \beta, \gamma) \in V(r_1, r_2).
\]

We turn now our attention to the case \( h_1(0) = h_2(0) = 0 \). Since \((0, \beta) \in \text{lim}(W(r_1))\) is a necessary condition for \((0, \beta, \gamma) \in \text{lim}(W(r_1, r_2))\) to hold we apply Proposition \[\text{II} \] 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 \( f_j^2 \in \mathbb{C}[(X_1^*)][X_3] \) by
\[
f_j^2(X_1, X_3) = r_2(X_1, \varphi_j(X_1), X_3).
\]

Let \( b \) be the degree of \( f_2^j \). Applying again Puiseux theorem, we consider \( \psi_1, \ldots, \psi_b \in \mathbb{C}((X_1^*)) \) such that the following holds
\[
\frac{f_j^2}{h_2} = (X_3 - \psi_1) \cdots (X_3 - \psi_b)
\]
in \( \mathbb{C}((X_1^*))[X_3] \). We assume that the series \( \psi_1, \ldots, \psi_a \) 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) \((0, 0, \gamma) \in \text{lim}(W(r_1, r_2))\) 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) \( \lim_{n \to \infty} (\alpha_n, \beta_n, \psi_k(\beta_n)) = (0, 0, \gamma) \).
Chapter 4. Computing Limit Points via Puiseux Series Expansions

Proof. Proving the implication (ii) ⇒ (i) is easy. We now prove the implication (i) ⇒ (ii). By Lemma 9, there exists a sequence (((α_n, β_n, γ_n), n ∈ ℕ) in A^3 s.t. for all n ∈ ℕ we have: (1) h_1(α_n) ≠ 0, (2) h_2(α_n) ≠ 0, (3) r_1(α_n, β_n) = 0, (4) r_2(α_n, β_n, γ_n) = 0, (5) \lim_{n→∞} (α_n, β_n, γ_n) = (0, 0, γ). Following the proof of Lemma 12, we know that for n large enough the product (β_n - ϕ_1(α_n)) · · · (β_n - ϕ_c(α_n)) is zero. Therefore, from one of the sequences (β_n - ϕ_1(α_n)), . . . , (β_n - ϕ_c(α_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 β_n = ϕ_j(α_n) holds, for all n ∈ ℕ. Hence, for all n ∈ ℕ, we have f^j_2(α_n, γ_n) = r_2(α_n, β_n, γ_n) = 0. Together with Equation (4.3) and following the proof of Lemma 12 we deduce the desired result.

\[ \square \]

Lemmas 11 and 13 immediately imply the following.

Proposition 4. For all γ ∈ C, the following two conditions are equivalent.

(i) (0, 0, γ) ∈ lim(W(r_1, r_2)) holds,

(ii) there exist integers j, k with 1 ≤ j ≤ c and 1 ≤ k ≤ a, such that ϕ_j(0) = 0 and ψ_k(0) = γ.

Therefore, applying Puiseux theorem to r_1 and f^j_2, then checking the constant terms of the series ψ_1, . . . , ψ_b provides a way to compute all γ ∈ C such that (0, 0, γ) is a limit point of W(r_1, r_2). Theorem 7 in Section 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) ≠ 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, . . . , r_{s-1}} ⊂ C[X_1 < · · · < 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 ρ > 0 be small enough such that the set U_ρ := \{x = (x_1, . . . , x_s) ∈ C^s | 0 < |x_1| < ρ\} does not contain any zeros of h_R. Define V_ρ(R) := V(R) ∩ U_ρ. Then, we have W(R) ∩ U_ρ = V_ρ(R). Let R' := (primpart(r_1), . . . , primpart(r_{s-1})). Then \( V_ρ(R) = V_ρ(R') \).

Proof. Let x ∈ W(R) ∩ U_ρ, then x ∈ V(R) and x ∈ U_ρ hold, which implies that W(R) ∩ U_ρ ⊆ V(R) ∩ U_ρ. Let x ∈ V(R) ∩ U_ρ. Since U_ρ ∩ V(h_R) = ∅, we have x ∈ W(R). Thus V(R) ∩ U_ρ ⊆ W(R) ∩ U_ρ. So W(R) ∩ U_ρ = V_ρ(R). Similarly, we have V_ρ(R) = V_ρ(R').

\[ \square \]

Notation 2. Let W ⊆ C^s. Denote \( \operatorname{lim}_0(W) := \{x = (x_1, . . . , x_s) ∈ C^s | x ∈ \lim(W) \text{ and } x_1 = 0\} \).

Lemma 15. We have \( \operatorname{lim}_0(W(R)) = \operatorname{lim}_0(V_ρ(R)) \).

Proof. By Lemma 14 we have W(R) ∩ U_ρ = V_ρ(R). Meanwhile, \( \operatorname{lim}_0(W(R)) = \operatorname{lim}_0(W(R) ∩ U_ρ) \) holds. Thus \( \operatorname{lim}_0(W(R)) = \operatorname{lim}_0(V_ρ(R)) \) holds.

\[ \square \]
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 \(\mathbb{C}((X_i^*))|X_2, \ldots, X_s\). Let \(V^*(R)\) be the zero set of \(R\) in \(\mathbb{C}((X_i^*))^{s-1}\). 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 \(\mathbb{C}((X_i^*))\) is an algebraically closed field. \(\square\)

**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 \(V^*_\rho(R) := \cup_{i=1}^m \{x \in \mathbb{C}^s \mid 0 < |x| < \rho, x_{j+1} = \Phi_i^j(x_1), j = 1, \ldots, s-1\}\).

**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 \(\Phi_i'\) 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

\[
r_{s-1}(X_1, X') = \Phi_i'(X_s) = h_{s-1}(X_1) \prod_{k=1}^{d_{s-1}} (X_s - \Phi_i^{s-1}(X_1)).
\]

(4.4)

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

\[
V^*_\rho(R) = \cup_{i=1}^m \cup_{k=1}^{d_{i-1}} \{x \mid x \in V_\rho(R'), x' = \Phi_i'(x_1), x_s = \Phi_i^{s-1}(x_1)\}
\]

\[
= \cup_{k=1}^{d_{i-1}} \{x \mid (x_1, x') \in V_\rho(R'), x_s = \Phi_i^{s-1}(x_1)\}
\]

\[
= \{x \mid (x_1, x') \in V_\rho(R'), x_{s-1}(x_1, x', x_s) = 0\}
\]

\[
= \{x \mid (x_1, x') \in V_\rho(R'), x_{s-1}(x_1, x', x_s) = 0\}
\]

\[
= V_\rho(R).
\]

\(\square\)

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

\[
\lim_0(W(R)) = \cup_{\Phi \in V^*_\rho(R)} \{(X_1 = 0, \Phi(X_1 = 0))\}.
\]
Proof. By definition of $V^*_0(R)$, we immediately have
\[
\lim_0(V^*_0(R)) = \bigcup_{\Phi \in V^*_0(R)} \{ (X_1 = 0, \Phi(X_1) = 0) \}.
\]
Next, by Theorem 5, we have $V^*_0(R) = V^*_0(R)$. Thus, we have $\lim_0(V^*_0(R)) = \lim_0(V^*_0(R))$. Besides, with Lemma 13, we have $\lim_0(W(R)) = \lim_0(V^*_0(R))$. Thus the theorem holds.

Definition 11. Let $V^*_0(R)$ be as defined in Theorem 6. Let $M = |V^*_0(R)|$. For each $\Phi_i = (\Phi_i^1, \ldots, \Phi_i^{s-1}) \in V^*_0(R)$, $1 \leq i \leq M$, we know that $\Phi_i^j \in \mathbb{C}((X_i^*))$. Moreover, by Equation (4.4), we know that for $j = 1, \ldots, s - 1$, $\Phi_i^j$ is a Puiseux expansion of $r(X_i, X_j) = \Phi_i^j, \ldots, X_j = \Phi_i^{j-1}, X_{j+1}$). Let $\varsigma_{ij}$ be the ramification index of $\Phi_i^j$ and $(T^{\varsigma_{ij}}, X_{j+1} = \varphi_i^j(T))$, where $\varphi_i^j \in \mathbb{C}(T)$, be the corresponding Puiseux parametrization of $\Phi_i^j$. Let $\varsigma_i$ be the least common multiple of $\{\varsigma_{ij}, \ldots, \varsigma_{ij-1}\}$. Let $g_i^j = \varphi_i^j(T = T^{\varsigma_i^{j-1}})$. We call the set $\theta_{R} := \{(X_1 = T^\varsigma, X_2 = g_1^1(T), \ldots, X_s = g_s^{s-1}(T)), i = 1, \ldots, M\}$ a system of Puiseux parametrizations of $R$.

Theorem 7. We have
\[
\lim_0(W(R)) = \theta_R(T = 0).
\]
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 $\lim_0(W(R))$ can be computed in the following manner. Compute a regular chain $N$ which is strongly normalized and such that $\text{sat}(R) = \text{sat}(N)$ and $V(h_N) = V(h_R)$ both hold, where $h_R$ is the iterated resultant of $h_R$ w.r.t $R$. See [25]. Let $X_1 := \text{mvar}(h_R)$. 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 3 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 $f \in \mathbb{C}(X)[Y]$ in finite accuracy, see Definition 12. For $f \in \mathbb{C}(X)[Y]$, we also define the approximation of $f$ for a given accuracy, see Definition 13. 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 $\text{disc}(f, Y) \neq 0$ 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 $f = \sum_{i=0}^{d} a_{i}(X)Y^{i} \in \mathbb{C}(X)[Y]$. For any $\tau \in \mathbb{N}$, we call $\tilde{f}^{(\tau)} := \sum_{i=0}^{d} a_{i}^{(\tau)}Y^{i}$ the approximation of $f$ of accuracy $\tau + 1$. 
Definition 13. Let \( f \in \mathbb{C}(X)[Y] \), 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 \( \sigma \) if there exists a Puiseux parametrization \((T^\tau, \varphi(T))\) of \( f \) such that:

(i) \( \sigma \) divides \( \varsigma \).
(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 \( \tau \).

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

Definition 14 ([33]). A \( \mathbb{C} \)-term 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 \( \pi \) is finite if there are only finitely many elements in \( \pi \).

Definition 15. Let \( \pi = (t_1, \ldots, t_N) \) be a finite \( \mathbb{C} \)-expansion. We define a pair \((T^\pi, g(T))\) 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 \( a := \prod_{i=1}^{N} q_i \), \( c_i := \sum_{j=1}^{N} \left( p_i \prod_{k=j+1}^{N} q_k \right) (1 \leq i \leq N) \), and \( \delta_i := c_i / \gcd(a, c_1, \ldots, c_N) \).

\((1 \leq i \leq N) \). 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 \((T^\pi, g(T))\) the Puiseux parametrization of \( \pi \) of accuracy \( \delta_N + 1 \). Denote by ConstructParametrization an algorithm to compute \((T^\pi, g(T))\) from \( \pi \).

Definition 16. Let \( f \in \mathbb{C}(X)[Y] \) 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 \( f \in \mathbb{C}(X)[Y] \). Next we present an algorithm, called NewtonPoly, to compute the segments in the Newton Polygon of \( f \). This algorithm is from R.J. Walker’s book [101].

NewtonPoly\((f, I)\)

**Input:** A polynomial \( f \in \mathbb{C}(X)[Y] \); 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 \( 0 \leq i \leq d \), define \( \delta_i := \text{ord}(b_i) \).
- For \( 0 \leq i \leq d \), we plot the points \( P_i \) with coordinates \((i, \delta_i)\); 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\([3]\).

Let \( f \in \mathbb{C}[X, Y] \), \( t = (q, p, \beta) \) be a \( \mathbb{C} \)-term and \( \ell \in \mathbb{N} \) s.t. 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 \( \Delta \) be a segment of the Newton

\[^3\text{It is a simplified version of Duval’s definition.}\]
Polygon of $f$. Denote $\text{SegmentPoly}(f, \Delta) := (q, p, \ell, \phi)$ 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 $C$-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 $(t, \ell)$, where $t$ is a $C$-term, and $\ell \in \mathbb{N}$.

```plaintext
begin
  1 S := \emptyset;
  2 for each $\Delta \in \text{NewtonPolygon}(f, I)$ do
  3      $(q, p, \ell, \phi) := \text{SegmentPoly}(f, \Delta)$;
  4      for each root $\xi$ of $\phi$ in $\mathbb{C}$ do
  5          for each root $\beta$ of $U^q - \xi$ in $\mathbb{C}$ do
  6              $t := (q, p, \beta)$;
  7              $S := S \cup \{(t, \ell)\}$
  8      return $S$
end
```

4.5 Computing in finite accuracy

Let $f \in \mathbb{C}(X)[Y]$. 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 $\tau$?

(b) If yes, how to deduce $m$ from $f$ and $\tau$?

(c) Provide a bound on $m$.

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

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

$$
\Phi_g : \mathbb{C}[[X]] \rightarrow \mathbb{C}[[Y]]
$$

$$
f \mapsto f(g_1(Y), \ldots, g_s(Y)).
$$

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

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

\[
\begin{align*}
&\text{begin} \\
&\quad \pi := \varnothing; S := \{(\pi, f)\}; P := \emptyset; \\
&\quad \text{while } S \neq \emptyset \text{ do} \\
&\quad \quad \text{choose } (\pi^*, f^*) \in S; S := S \setminus \{(\pi^*, f^*)\}; \\
&\quad \quad \text{if } \pi^* = (\ ) \text{ then } I := 1 \text{ else } I := 2; \\
&\quad \quad (T^\sigma, g(T)) := \text{ConstructParametrization}(\pi^*); \\
&\quad \quad \text{if } \deg(g(T), T) + 1 < \tau \text{ then} \\
&\quad \quad \quad C := \text{NonzeroTerm}(f^*, I); \\
&\quad \quad \quad \text{if } C = \emptyset \text{ then} \\
&\quad \quad \quad \quad P := P \cup \{(T^\sigma, g(T))\}/\text{ a finite Puiseux parametrization is found} \\
&\quad \quad \quad \text{else} \\
&\quad \quad \quad \quad \text{for each } (t = (p, q, \beta), \ell) \in C \text{ do} \\
&\quad \quad \quad \quad \quad \pi^{**} := \pi^* \cup (t); \\
&\quad \quad \quad \quad \quad f^{**} := \text{NewPoly}(f^*, t, \ell); \\
&\quad \quad \quad \quad \quad S := S \cup \{(\pi^{**}, f^{**})\} \\
&\quad \quad \quad \quad \text{else} \\
&\quad \quad \quad \quad \quad P := P \cup \{(T^\sigma, g(T))\} \\
&\quad \text{end} \\
&\text{return } P
\end{align*}
\]

**Lemma 18.** Let \( f \in \mathbb{C}(X)[Y] \). 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 \( q, p, \ell \) define a line \( L : qj + pi = \ell \) in the \((i, j)\)-plane such that:

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

(b) For any \((j, i) \in \text{carr}(f)\), 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 \( f_1 \in \mathbb{C}(X_1)[Y_1] \).

(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 \( \text{ord}(X_1^q) = q > 0 \) holds. We also have

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

Let

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

We prove (ii). We have

$$f_1(X_1, Y_1) \mod \langle X_1^{m_1} \rangle = \sum_{i=0}^{d} \left( \sum_{j=0}^{\infty} a_{i,j} X_1^{q_j+i-\ell} \right) (\beta + Y_1)^j.$$ 

Since $q \in \mathbb{N}_{>0}$ and $m_1$, $\ell$ 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 $q_j + pi - \ell < m_1$, 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 $m' \leq \left\lfloor \frac{(m_1+\ell)}{q} \right\rfloor$. Next we show that $m_1 \geq 1$ 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. \(\square\)

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

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

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

**Remark 7.** Let $f(X, Y) := \sum_{i=0}^{d} \left( \sum_{j=0}^{\infty} a_{i,j} X^i \right) Y^j$. 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 $f \in \mathbb{C}(X)[Y]$. Let $\tau \in \mathbb{N}_{>0}$. Let $\sigma \in \mathbb{N}_{>0}$ and $g(T) = \sum_{k=0}^{\sigma-1} b_k T^k$. Assume that $(T^\sigma, g(T))$ is a Puiseux parametrization of $f$ of accuracy $\tau$. Then one can compute a number $m \in \mathbb{N}$ such that $(T^\sigma, g(T))$ is a Puiseux parametrization of accuracy $\tau$ of $\tilde{f}^{m-1}$, where $\tilde{f}^{m-1}$ is the approximation of $f$ of accuracy $m$. We denote by AccuracyEstimate an algorithm to compute $m$ from $f$ and $\tau$.

**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

$$f_i := X_1^{-\ell_i} f_{i-1}(X_1^{q_i}, X_1^{p_i} (\beta_i + Y_i))$$

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 $\sigma$ 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 $\tau$ 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 $f_0 := \sum_{i=0}^{d} \left( \sum_{j=0}^{r_i} a_{i,j} X^j \right) Y^i$. That is, $f_0$ is the approximation of $f$ of accuracy $m' + 1$. We run the Newton-Puiseux algorithm to check whether the terms $a_{k,m} X^{m'} Y^k$, $0 \leq k \leq d$, 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 $a_{k,m} X^{m'} Y^k$, $0 \leq k \leq d$, makes any contributions in constructing the Newton Polygons of all $f_i$. By Remark 16 we can set $m := m'$.

Lemma 19. Let $d, \tau \in \mathbb{N}_{>0}$. Let $a_{i,j}$, $0 \leq i \leq d$, $0 \leq j < \tau$, and $b_k$, $0 \leq k < \tau$ be symbols. Write $a = (a_{0,0}, \ldots, a_{0,\tau-1}, \ldots, a_{d,0}, \ldots, a_{d,\tau-1})$ and $b = (b_0, \ldots, b_{\tau-1})$. Let $f(a, X, Y) = \sum_{i=0}^{d} \left( \sum_{j=0}^{r_i} a_{i,j} X^j \right) Y^i \in \mathbb{C}[a][X, Y]$ and let $g(b, X) = \sum_{k=0}^{\tau-1} b_k X^k \in \mathbb{C}[b][X]$. Let $p := f(a, X, Y) = g(b, X)$. Let $F_k := \text{coeff}(p, X^k)$, $0 \leq k < \tau - 1$, and $F := \{F_0, \ldots, F_{\tau-1}\}$. Then under the order $a < b$ and $b_0 < b_1 < \cdots < b_{\tau-1}$, $F$ forms a zero-dimensional regular chain in $\mathbb{C}(a)[b]$ 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) $\text{init}(F_1) = \cdots = \text{init}(F_{\tau-1}) = \text{der}(F_0, b_0) = \sum_{i=1}^{d} i \cdot a_{i,0} b_0^{-i}$.

Proof. Write $p = \sum_{i=0}^{d} \left( \sum_{j=0}^{r_i} a_{i,j} X^j \right) \left( \sum_{k=0}^{\tau-1} b_k X^k \right)^i$ 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 $\mathbb{C}(a)[b]$. Moreover, we have $\text{mvar}(F_0) = b_0$ and $\text{mdeg}(F_0) = d$.

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 $0 \leq k < \tau$, $b_k$ appears in $F_k$. Moreover, for any $k \geq 1$ 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 $1 \leq k < \tau$. 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 $\text{init}(F_1) = \cdots = \text{init}(F_{\tau-1}) = \sum_{i=1}^{d} i \cdot a_{i,0} b_0^{-i-1}$, 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 $0 \leq i < \tau$, can be completely determined by $a_{i,j}$, $0 \leq i \leq d, 0 \leq j < \tau$ if and only if $b_0$ is a simple zero of $f(0, Y)$. Therefore, “generically”, all coefficients $b_i$, for $0 \leq i < \tau$, can be completely determined by the approximation of $f$ of accuracy $\tau$.

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

$$\sum_{i=0}^{d} \left( \sum_{j<\tau} a_{i,j} X^j \right) \left( \sum_{k<\tau} b_k X^k \right)^i = 0 \mod \langle X^\tau \rangle.$$
Let \( p = \sum_{i=0}^{d} \left( \sum_{j < r} a_{i,j} X^{j} \right) \left( \sum_{k < r} b_{k} X^{k} \right)^{i} \). Let \( F_{i} := \{ \text{coeff}(p, X^{i}) \}, 0 \leq i < \tau \), 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 \( a_{i,0}, 0 \leq i \leq d \). 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 \( b_{s} \), \( 0 \leq k < \tau \) can be completely determined from almost all the values of \( a_{i,j}, 0 \leq i \leq d, 0 \leq j < \tau \).

\[ \square \]

### 4.6 Accuracy estimates

Let \( R := \{ r_{1}(X_{1},X_{2}), \ldots, r_{\tau}(X_{1}, \ldots, X_{s}) \} \subset \mathbb{C}[X_{1} \cdots X_{s}] \) 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 \( f = a_{d}(X)Y^{d} + \cdots + a_{0}(X) \in \mathbb{C}(X)[Y] \), where \( d = \deg(f, Y) \geq 0 \). For \( 0 \leq i \leq d \), let \( \delta_{i} := \text{ord}(a_{i}) \). Let \( k := \min(\delta_{0}, \ldots, \delta_{d}) \). Let \( \tilde{f} := X^{k}f \). Then we have \( \tilde{f} \in \mathbb{C}(X)[Y] \) and \( \tilde{f} \) is general in \( Y \). This operation of producing \( f \) from \( \tilde{f} \) is called “making \( f \) general” and we denote it by MakeGeneral.

**Proof.** Since \( k = \min(\delta_{0}, \ldots, \delta_{d}) \), there exists \( i, 1 \leq i \leq d \), such that \( k = \delta_{i} \). Moreover, for all \( 1 \leq j \leq d \), we have \( \delta_{j} \geq k \). Thus for every such \( i \), we have \( \text{ord}(a_{i}(X)/X^{k}) = 0 \) and \( a_{j}(X)/X^{k} \in \mathbb{C}(X), 0 \leq j \leq d \). This shows that \( \tilde{f} \in \mathbb{C}(X)[Y] \) and \( \tilde{f} \) is general in \( Y \).

\[ \square \]

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

**Lemma 22.** Let \( f \in \mathbb{C}(X)[Y] \), 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 \( 1 \leq i \leq c \), for some \( c \) such that \( 1 \leq c \leq \deg(f, Y) \), let \( (X = T^{\sigma_{i}}, Y = \varphi_{i}(T)) \) be the distinct Puiseux parametrizations of \( f \). By Lemma 14 and Theorem 7, we have \( \lim_{0} V_{\rho}(f) = \bigcup_{i=1}^{c} \{(0, y) \in \mathbb{C}^{2} \mid y = \varphi_{i}(0)\} \). 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 \( \tilde{f} \) of \( f \) of some finite accuracy such that \( (X = T^{\sigma_{i}}, g_{i}(T)), i = 1, \ldots, c \), are also Puiseux parametrizations of \( f \) of accuracy 1. Thus, we have \( \varphi_{i}(0) = g_{i}(0) \), \( i = 1, \ldots, c \). Since \( \tilde{f} \) is general in \( Y \), by Theorem 2.3 in [10], we have \( \bigcup_{i=1}^{c} \{(0, y) \in \mathbb{C}^{2} \mid y = g_{i}(0)\} = \{(0, y) \in \mathbb{C}^{2} \mid \tilde{f}(0, y) = 0\} \). Since \( \tilde{f}(0, y) = f(0, y) \), the lemma holds.

\[ \square \]

**Lemma 23.** Let \( a(X_{1}, \ldots, X_{s}) \in \mathbb{C}[X_{1}, \ldots, X_{s}] \). Let \( g_{i} = \sum_{j=0}^{\infty} c_{i,j} T^{j} \in \mathbb{C}(T) \), 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 \( 1 \leq i \leq s, 0 \leq j \leq k \).

**Proof.** We observe that any \( c_{i,j} \), where \( j > k \), does not make any contribution to \( b_{k} \).

\[ \square \]
4.6. Accuracy estimates

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

Proof. Let $\tilde{f} := \text{MakeGeneral}(f)$. Observe that $f$ and $\tilde{f}$ 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 $1 \leq i \leq s - 1$, let $h_i := \text{init}(r_i)$, $d_i := \deg(r_i, X_{i+1})$ and $\delta_i := \text{ord}(h_i)$. We define $f_i, \varsigma_i, T_i, \varphi_i(T_i), 1 \leq i \leq s - 1$, as follows. Let $f_1 := r_1$. Let $(X_1 = T_1^0, X_2 = \varphi_1(T_1))$ be a Puiseux parametrization of $f_1$. For $i = 2, \ldots, s - 1$ do

(i) Let $f_i := r_i(X_1 = T_1^0, X_2 = \varphi_1(T_1), \ldots, X_i = \varphi_{i-1}(T_{i-1}), X_{i+1})$.

(ii) Let $(T_{i-1} = T_i^0, X_{i+1} = \varphi_i(T_i))$ be a Puiseux parametrization of $f_i$.

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

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

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

$$T_{i-1} = T_i^0 = \left(T_{s-2}^{\prod_{k=1}^{i+1} s_k}\right)^{\varsigma_i} = \left(T_{s-2}^{\prod_{k=1}^{i+1} s_k}\right)_{\varsigma_i}.$$

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

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 $\varphi_i$ of accuracy $\tau_i, 1 \leq i \leq s - 2$. Moreover, if we write the algorithm AccuracyEstimate for short as $\theta$, the accuracies $\tau_i$ can be computed in the following manner:

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

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

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

Let $\tau_{s-2} := (\prod_{k=1}^{s-2} s_k)\delta_{s-1} + 1$. 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 $\tau_{s-2}$.

By Lemma 22 we need to compute the polynomial parts of $\varphi_i(g_i(T_{s-2})), 1 \leq i \leq s - 2$, of accuracy $\tau_{s-2}$. Since $\text{ord}(g_i(T_{s-2})) = (\prod_{k=1}^{s-2} s_k)$, to achieve this accuracy, it is enough to compute the polynomial parts of $\varphi_i$ of accuracy $(\prod_{k=1}^{i+1} s_k)\delta_{s-1} + 1$, for $1 \leq i \leq s - 2$.

Since we have $f_i = r_i(X_1 = T_1^0, X_2 = \varphi_1(T_1), \ldots, X_i = \varphi_{i-1}(T_{i-1}), X_{i+1})$ and $(T_{i-1} = T_i^0, X_{i+1} = \varphi_i(T_i))$ is a Puiseux parametrization of $f_i$, by Theorem 9 and Lemma 23 to compute the polynomial part of $\varphi_i$ of accuracy $\tau_i$, we need the polynomial part of $\varphi_{i-1}$ of accuracy $\theta(f_i, \tau_i)$. Thus, $\tau_{s-2} := (\prod_{k=1}^{s-2} s_k)\delta_{s-1} + 1$ and $\tau_{i-1} := \max(\theta(f_i, \tau_i), (\prod_{k=1}^{i-1} s_k)\delta_{s-1} + 1)$ for $2 \leq i \leq s - 2$ 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 $\tau_i$, for $i = 1, \ldots, s - 1$. Moreover, generically, one can choose $\tau_i, i = 1, \ldots, s - 1$, as follows

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

and each index $\varsigma_k$ 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 $\text{ord}(h_i(X_1)) = \delta_i$, we have

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

By Lemma 24, generically a Puiseux parametrization of $f_i$ of accuracy $\tau_i$ 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 $1 \leq k \leq s - 2$. \hfill $\square$

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 $\tau_i$. 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. \hfill $\square$

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

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 $\text{lim}(W(R))$ 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 $x = (x_1, \ldots, x_n) \in \text{lim}(W(R))$, 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.

begin
1. let \( S := \{(T_0)\} \);
2. compute the accuracy estimates \( \tau_1, \ldots, \tau_{s-2} \) by Theorem 10; let \( \tau_{s-1} = 1 \);
3. for \( i \) from 1 to \( s - 1 \) do
4. \hspace{1em} let \( S' := \emptyset \);
5. \hspace{1em} for \( \Phi \in S \) do
6. \hspace{2em} \( f_i := r_i(X_1 = \Phi_1, \ldots, X_i = \Phi_i, X_{i+1}) \);
7. \hspace{2em} if \( i > 1 \) then
8. \hspace{3em} let \( \delta := \text{ord}(f_i, T_{i-1}) \); let \( f_i := f_i / T^\delta_{i-1} \);
9. \hspace{3em} let \( E := \text{NewtonPuiseux}(f_i, \tau_i) \);
10. \hspace{3em} for \( (T_{i-1} = \phi(T_i), X_{i+1} = \phi(T_i)) \in E \) do
11. \hspace{4em} \( S' := S' \cup \{\Phi(T_{i-1} = \phi(T_i)) \cup (\varphi(T_i))\} \)
12. \hspace{2em} \( S := S' \)
13. if \( S = \emptyset \) then
14. \hspace{1em} return \( \emptyset \)
15. else
16. \hspace{1em} return eval\( (S, T_{s-1} = 0) \)
17. end

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 \( W(R) \).

begin
1. let \( h_R := \text{init}(R) \); let \( L \) be the set of zeros of \( h_R \) in \( \mathbb{C} \);
2. \( S := \emptyset \);
3. for \( \alpha \in L \) do
4. \hspace{1em} \( R_\alpha := R(X_1 = X_1 + \alpha) \);
5. \hspace{1em} \( S_\alpha := \text{LimitPointsAtZero}(R_\alpha) \);
6. \hspace{1em} update \( S_\alpha \) by replacing the first coordinate of every point in \( S_\alpha \) by \( \alpha \);
7. \hspace{1em} \( S := S \cup S_\alpha \)
8. return \( S \)
9. end
Table 4.1: Removing redundant components.

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

new symbols. Let \( G := \{g_1(Y_1), g_2(Y_1, Y_2), \ldots, g_n(Y_1, Y_2, \ldots, Y_m) \} \) 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 \( S_x := \{X_1 = f_1(Y_1, \ldots, Y_m), \ldots, X_n = f_n(Y_1, \ldots, Y_m), G(Y_1, \ldots, Y_m) = 0\} \). The projection of the zero set of \( S_x \) 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 \( \lim(W(R)) = \bigcup_{x \in \lim(W(R))} W(R_x) \). □

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 \( W(R) \subseteq \overline{W(R')} \) 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 \( \text{sat}(R) \) 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 \( \mathbb{Q} \):

\[ f_1 = -y^3 + y^2 + z^5 \quad \text{and} \quad f_2 = z^4 x + y^3 - y^2. \]  

(5.1)

For each real value of \( z \) and each real value of \( y \) satisfying both \( f_1 = 0 \) and \( z \), 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 \( \overline{Z_\mathbb{R}(T)} \) 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 \( \lim(\overline{Z_\mathbb{R}(T)}) : = \overline{Z_\mathbb{R}(T)} \setminus Z_\mathbb{R}(T) \). On the above example, our first question asks to compute \( \lim(\overline{Z_\mathbb{R}(\{f_1, f_2\})}) \). 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 \( W(T) \) the closure of \( W(T) \) in the Zariski topology, the algorithms of \([6]\) determine the set \( \lim(W(T)) := W(T) \setminus W(T) \). The work of \([6]\) requires that \( T \) is a one-dimensional regular chain, that is, \( W(T) \) 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 \( \lim(W(T)) \) consists of two points, as shown in Figure 5.1 with our implementation in the RegularChains library at www.regularchains.org. Notably, both points have real coordinates. But only one of them belongs to \( \lim(Z_\mathbb{R}(T)) \). 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 \( \text{RootOf}(\mathbb{C}^2 + 1) \) in their coefficients. In Maple, the notation \( \text{RootOf}(\mathbb{C}^2 + 1) \) 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 \( \lim(Z_\mathbb{R}(T)) \).
5.1. Introduction

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

![Figure 5.1: Computing the limit points of a regular chain: complex case vs real case.](image)

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 \( \lim(Z_{\mathbb{R}}(T)) \), 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 a 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 \( W(T) \setminus W(T) \) 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 $Z_R(S)$ 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 $\lim(Z_R(S))$ 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 $W(T) \setminus \bar{W}(T)$ (where $\bar{W}(T)$ 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 $W_R(T) := Z_R(T) \setminus Z_R(h_T)$, that is, the set $W_R(T) \setminus \bar{W}_R(T)$, where $W_R(T)$ is the closure of $W_R(T)$ in $\mathbb{R}^n$ endowed with the Euclidean topology. Unfortunately, it is not true that the non-trivial limit points of $W_R(T)$ 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 $W_R(T)$. This adaptation is explained hereafter. The LimitPoints command of the RegularChains library in Maple handles both cases, $\bar{W}(T) \setminus W(T)$ and $W_R(T) \setminus \bar{W}_R(T)$. The Puiseux parametrizations of the regular chain $T$ in Definition 18 (see also [6], Definition 2) encode all the branches of $V(\text{sat}(T))$ 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 $T := \{t_1, \ldots, t_{n-1}\} \subseteq \mathbb{Q}[X_1 < \cdots < X_n]$ 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 $\varsigma_j$ 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 $\varsigma := \text{lcm}(\varsigma_2, \ldots, \varsigma_n)$ and $\phi_j = \chi_j(U^\varsigma_j)$. Then $(U^\varsigma, \phi_2, \ldots, \phi_n)$ is called a Puiseux parametrization of $T$ around $U = 0$.

**Remark 9.** If $\alpha$ 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 $\mathbb{C}(U)^\varsigma$.

**Example 8.** Let $T := \{t_1, t_2\} \subseteq \mathbb{Q}[X_1 < X_2 < X_3]$ be a regular chain where $t_1 := X_1^4 - 2X_2^3 + X_2^2 + X_1^3$ and $t_2 := X_1^4X_3 + X_3^2 - X_2^2$. 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:

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

(5.2)

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

Now by substituting $\Phi_1$ 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 $\Phi_2$, one obtains a Puiseux parametrization in which negative exponents of $U$ appear as well. However, the scenario is different when substituting $\Phi_3$ into

---
[1]This library is freely available from www.regularchains.org
\[ t_2. \text{ Indeed, this substitution yields } t_{23} := U^8 X_3 + (\sqrt{-1} U^5 - U^{10})^3 - (\sqrt{-1} U^5 - U^{10})^2, \]

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 \( T \):

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

Proceeding similarly with \( \Phi_{2,4} \), one obtains the second Puiseux parametrization of \( T \):

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

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

**Definition 19.** Using the notations of Definition 8, the Puiseux parametrization \((U^i, \phi_2, \ldots, \phi_n)\)

is called a real Puiseux parametrization of \( T \) if \( \phi_i \in \mathbb{R}(U) \), 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 \cite{Ref1}, 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 \( k \) be an algebraic number field and \( f(U, Y) \in k(U)[Y] \) 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 \( \sigma_\ell \) as well as algebraic numbers \( \Theta_\ell^1, \ldots, \Theta_\ell^s \) over \( k \) such that

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

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 \( \Theta_1, \ldots, \Theta_{\ell'} \) over \( k \) such that \( \chi_\ell(U) \in k(\Theta_1, \ldots, \Theta_{\ell'})((U')) \), for \( \ell = 1, \ldots, s \).

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

Based on Lemma 5 and Corollary 1, there is at most one algebraic number \( \Theta_1 \) with minimal polynomial \( h_1(Y) \in k[Y] \) (which is, indeed, computed by substituting \( U = 1 \) in Newton polynomial of \( f_0 \) such that \( k_1 := k(\Theta_1) \) and \( f_1 \in k_1((U')) \)). Since EHC is applied recursively on polynomials \( f_i \), thus there are algebraic numbers \( \Theta_i \) with minimal polynomials \( h_i(Y) \in k_{i-1}[Y] \), where \( k_i := k_{i-1}(\Theta_i) \) and \( f_i \in k_i((U')) \), for \( i = 2, \ldots, \ell' \). Therefore, \( f_{\ell'} \in k_{\ell'}((U')) \) and consequently, \( \chi_s(U) \in k_{\ell'}((U')) = k(\Theta_1, \ldots, \Theta_{\ell'})((U')) \).

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. \( \square \)

Proposition 6 follows from the extended Hensel construction. This proposition shows that there is a finite extension of \( k \) 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 \( k = \mathbb{Q} \), then determining whether or not \( \chi_i(U) \) is a real Puiseux expansion is equivalent to the fact that each \( \Theta^{-1}_i \) is a real algebraic number over \( \mathbb{Q}(\Theta_1, \ldots, \Theta_{\ell'}^{-1}) \), for \( i = 1, \ldots, \ell' \).

Furthermore, based on the construction of \( h_1(Y) \), all of the roots of each polynomial \( h_1(Y) \) 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 \( h_1(Y) \) that allows us to separate the real ones from the others. For simplicity of presentation, we do that by considering the splitting fields of \( h_1(Y) \), see Remark 11. For computational efficiency, one should prefer techniques based on real algebraic closures as in [76].

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

Remark 11. To construct the splitting field \( L \) of \( h(Y) \) and compute the factorization of \( h(Y) \) into linear factors over \( L \), one can proceed as follows.

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

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

\[
k[Y] \subset \frac{k[X_i]}{(R_i)}[Y] \subset \cdots \subset \frac{k[X_1, \ldots, X_{s'}]}{(R_{s'})}[Y].
\]

**Algorithm 6: RealPuiseuxExpansions**

**Input:** \( f(U, Y) \in k[U, Y] \).

**Output:** Real Puiseux expansions of \( f \) when \( U \to 0 \).

1. \begin{algorithmic}
   2. \State \( \mathcal{B} := \) Puiseux expansions of \( f(U, Y) \) at \( U = 0 \);
   3. \State \( \mathcal{R} := \{\} \);
   4. \For{\( \chi(U) \in \mathcal{B} \)}
   5. \State let \( \chi(U) \in k(\Theta^1, \ldots, \Theta^\sigma)((U^*)) \);
   6. \State let \( R^i_{j_i} \subset k[X_{i,1}, \ldots, X_{i,j_i}] \) be the zero-dimensional regular chain encoding the algebraic number \( \Theta^i \), for \( i = 1, \ldots, \sigma \);
   7. \State let \( C \) be a regular chain encoding the field \( k \);
   8. \State \( F := C \cup R^1_{j_1} \cup \cdots \cup R^\sigma_{j_\sigma} \);
   9. \If{\( \text{RealTriangularize}(F) \neq \emptyset \)}
   10. \State \( \mathcal{R} := \mathcal{R} \cup \{\chi(U)\} \);
   11. \EndIf
   12. \EndFor
   13. \State return \( \mathcal{R} \);
\end{algorithmic}

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 \( \frac{Q[X_1, X_2]}{(R_2)}[Y] \) 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 \( \Theta \) be a root of \( h(Y) \). Let also \( j \) be the smallest integer for which \( \Theta \in \frac{k[X_1, \ldots, X_j]}{(R_j)} \), then \( R_j \) is called a regular chain encoding of \( \Theta \).

Following up on Definition \([20]\) determining whether or not \( \Theta \) is a real algebraic number over \( k \) is equivalent to check whether or not \( Z_\mathbb{R}(R_j) \) has a real solution or not over \( k \). Furthermore, \( k \) must be a real extension of \( Q \). 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 \( F \subset Q[X_1, \ldots, X_n] \). Thus, for checking whether or not \( \Theta \) is a real algebraic
number over \( \mathbf{k} \), using \texttt{RealTriangulize}, 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 \( \mathbb{Q}[Y_{1}, \ldots, Y_{m}] \) is isomorphic to \( \mathbf{k} \). This means that one can apply \texttt{RealTriangulize} 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 \( \Theta \) 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 \( \Theta_j^1, \ldots, \Theta_j^n \) over \( \mathbf{k}_{j-1} \) such that \( \phi_j(U) \in \mathbf{k}_j[U] \), where \( \mathbf{k}_1 := \mathbb{Q} \) and \( \mathbf{k}_j := \mathbf{k}_{j-1}(\Theta_j^1, \ldots, \Theta_j^n) \) for some non-negative integer \( j \).

**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 \( \Theta_2^1, \ldots, \Theta_2^n \) over \( \mathbb{Q} \) such that \( \chi_2(U) \in \mathbb{Q}(\Theta_2^1, \ldots, \Theta_2^n) \). Suppose \( \chi_{j-1}(U) \in \mathbf{k}_{j-1}[U] \). Thus, \( \chi_j(U) \) is a Puiseux expansion of bivariate polynomial

\[
\begin{align*}
&= t_{j-1}(U^{S-1}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j)
\end{align*}
\]

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

**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 \( \mathbb{Q} \). \( \square \)

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 \((U^S, \chi_j(U))\) of the bivariate polynomial

\[
\begin{align*}
&= t_{j-1}(U^{S-1}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j),
\end{align*}
\]

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 \( t_{j-1}(U^{S-1}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j) \). To
Algorithm 7: RealRegularChainBranches

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

begin
\[ \mathcal{R} := \{\}; \]
\[ \varsigma_1 = 1; \]
for $j$ from 2 to $n$ do
\[ \mathcal{R}_j := \{\}; \]
for $\chi_2(U), \ldots, \chi_{j-1}(U) \in \mathcal{R}$ do
\[ \mathcal{B} := \text{RealPuiseuxExpansions}(t_{j-1}(U^{j-1}), \chi_2(U), \ldots, \chi_{j-1}(U), X_j, 0); \]
\[ \varsigma_j = \text{RamificationIndex}(t_{j-1}(U^{j-1}), \chi_2(U), \ldots, \chi_{j-1}(U)); \]
if $\mathcal{B} \neq \emptyset$ then
\[ \text{let } \mathcal{B} := \{\chi_j(U), \ldots, \chi_j^\ell(U)\}; \]
\[ \mathcal{R}_j := \mathcal{R}_j \cup \{\chi_2(U), \ldots, \chi_{j-1}(U), \chi_j^\ell(U)\}; \]
\[ \mathcal{R} := \mathcal{R}_j; \]
return $\mathcal{R};$
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 $(U^{j-1}, \chi_j(U))$ of the bivariate polynomial $t_{j-1}(U^{j-1}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j)$, 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 $t_{j-1}(U^{j-1}, \chi_2(U), \ldots, \chi_{j-1}(U), X_j)$. 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 respectively, the time spent for computing and the number of real limit points. The fourth and fifth columns are respectively, 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($\mathcal{R}$, coefficient = real) and LimitPoints($\mathcal{R}$, coefficient = complex) for computing, respectively, the real and complex limit points of the input regular chain $\mathcal{R}$. 
Table 5.1: Complex limit points vs real limit points

<table>
<thead>
<tr>
<th>Sys</th>
<th>RealLimit (sec)</th>
<th>#LM</th>
<th>ComplexLimit (sec)</th>
<th>#LM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liu-Lorenz</td>
<td>777.300</td>
<td>4</td>
<td>1708.829</td>
<td>9</td>
</tr>
<tr>
<td>MontesS3</td>
<td>0.015</td>
<td>0</td>
<td>0.015</td>
<td>0</td>
</tr>
<tr>
<td>Neural</td>
<td>1.538</td>
<td>3</td>
<td>2.368</td>
<td>3</td>
</tr>
<tr>
<td>cox-issac07</td>
<td>0.438</td>
<td>0</td>
<td>0.575</td>
<td>1</td>
</tr>
</tbody>
</table>
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 $\text{sat}(T)$. The set $\text{lim}(W(T))$ of the non-trivial limit points of
$W(T)$ satisfies $V(\text{sat}(T)) = \overline{W(T)} = W(T) \cup \lim(W(T))$. Hence, $\lim(W(T))$ is the set-theoretic difference $V(\text{sat}(T)) \setminus W(T)$. Deducing $\lim(W(T))$ or $V(\text{sat}(T))$ from $T$ is a central question which has theoretical applications (like the so-called \textit{Ritt Problem}) and practical ones (like removing redundant components in triangular decomposition).

Of course $V(\text{sat}(T))$ can be computed from $T$ via \textit{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 \texttt{PALGIE}. Both algorithms are part of the \texttt{RegularChains} library in \texttt{Maple} and experimentally, the second one performs better, while its algebraic complexity is unknown.

Returning to Noether normalization\footnote{Section 6.4 contains a brief review of Noether normalization which makes this chapter self-contained.}, we ask in Section 6.4 how “simple” can $T$ be if we assume that $\text{sat}(T)$ 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 $V(\text{sat}(T))$. 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 $V(\text{sat}(T))$ without \textit{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 “\textit{Gröbner-basis-free}” general algorithm for $\lim(W(T))$ or $V(\text{sat}(T))$ remains unsolved.

\section{Preliminaries}

Throughout this chapter, polynomials have coefficients in a field $k$ and variables in a set $x$ of $n$ ordered variables $x_1 < \cdots < x_n$. The corresponding polynomial ring is denoted by $k[x]$. Let $F$ be a subset of $k[x]$. We denote by $\langle F \rangle$ the ideal generated by $F$ in $k[x]$. Recall that a polynomial $f \in k[x]$ is \textit{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, $\overline{k}$ stands for the algebraic closure of $k$ and $V(F) \subset \overline{k}^n$ for the algebraic set consisting of all common zeros of all $f \in F$. For a set $W \subset \overline{k}$, we denote by $\overline{W}$ the \textit{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 $f \in k[x]$, we denote by $\text{mvar}(f), \text{mdeg}(f)$ and
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 k[x] is called triangular if no two polynomials from T have the same main variable. Let T ⊂ k[x] be a triangular set. Observe that T is necessarily finite and that every subset of T is itself triangular. For a variable v ∈ x, if there exists f ∈ T such that mvar(f) = v, we denote this polynomial by T_v and say that v is free w.r.t. T, otherwise we say that v is free w.r.t. T. In all cases, we define T_{<v} := \{g ∈ T \mid mvar(g) < v\} and denote by free(T) the set of the variables from x which are free w.r.t. T. We denote by h_f the product of the polynomials init(f), for f ∈ T. We say that T is strongly normalized if all variables occurring in h_f are in free(T); when this holds, it is easy to check that T is a Gröbner basis of the ideal that T generates in k[u]|x \setminus u where u := free(T) and k(u) is the field of rational functions over k and with variables in u. Moreover, we say that T is monic whenever h_f ∈ k holds. The saturated ideal of T, written sat(T), is defined as the colon ideal sat(T) = ⟨T⟩ : h_f^n. The quasi-component of T is the basic constructible set given by W(T) := V(T) \setminus V(h_f). The following two properties are easy to prove:

\[
\overline{W(T)} = V(\text{sat}(T)) \quad \text{and} \quad \overline{W(T)} = W(T) \cup \lim(W(T)),
\]

where lim(W(T)) := \overline{W(T)} \cap V(h_f) 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 sat(T_{<w}), where w is the largest main variable of a polynomial in T. If T consists of n − d polynomials, for 0 ≤ d < n, then sat(T) has dimension d and either lim(W(T)) is empty or has dimension d − 1; moreover, we have k[u] \cap sat(T) = ⟨0⟩, where u := free(T).

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

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

\[
A : \overline{k}^n \rightarrow \overline{k}^n \quad \text{x} \longmapsto (A_1(x), \ldots, A_n(x))
\]

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

\[
V^A(F) := V(\{f^A \mid f ∈ F\}),
\]

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

\[
W^A(T) = V^A(T) \setminus V^A(h_f).
\]
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 $T \subseteq \mathbb{k}[x]$ and given a linear change of coordinates $A$ in $\mathbb{k}^n$, compute finitely many regular chains $C_1, \ldots, C_e$ such that

$$W^A(T) = W(C_1) \cup \cdots \cup W(C_e).$$

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

**Problem 2.** Given two total orderings $\mathcal{R}$ and $\mathcal{R}'$ on $\{x_1, \ldots, x_n\}$, given $T \subseteq \mathbb{k}[x_1, \ldots, x_n]$, assuming that
1. $T$ is a regular chain for the ordering $\mathcal{R}$ on $\{x_1, \ldots, x_n\}$ and,
2. the saturated ideal $\text{sat}(T, \mathcal{R})$ (which is an alias of $\text{sat}(T)$ with a second argument recalling the ordering) of $T$ of $\mathbb{k}[x_1, \ldots, x_n]$ is prime,
compute $C \subseteq \mathbb{k}[x_1, \ldots, x_n]$ such that
3. $C$ is a regular chain for the ordering $\mathcal{R}'$ on $\{x_1, \ldots, x_n\}$ and,
4. the saturated ideal $\text{sat}(C, \mathcal{R}')$ of $C$ in $\mathbb{k}[x_1, \ldots, x_n]$ is equal to $\text{sat}(T, \mathcal{R})$.

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 $\mathcal{R}'$ on $\{x_1, \ldots, x_n\}$, given $T \subseteq \mathbb{k}[x_1, \ldots, x_n]$, 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 $\text{sat}(T, \mathcal{R})$ of $T$ in $\mathbb{k}[x_1, \ldots, x_n]$ is equal to the intersection of the radicals of the saturated ideals $\text{sat}(C_i, \mathcal{R}')$ of $C_i$ in $\mathbb{k}[x_1, \ldots, x_n]$, 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 $\mathcal{R}'$ on $\{x_1, \ldots, x_n\}$ as well as a regular chain $T \subseteq \mathbb{k}[x_1, \ldots, x_n]$ for the order $\mathcal{R}$ such that its saturated ideal $\text{sat}(T, \mathcal{R})$ 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 $\mathcal{R}'$ can be interpreted as a permutation $\sigma$ of the sequence $(x_1, \ldots, x_n)$. Let $A$ be the linear change of coordinates replacing the column vector $(x_1, \ldots, x_n)'$ with $M_\sigma(y_1, \ldots, y_n)'$ where $(y_1, \ldots, y_n)$ stand for the new coordinates and $M_\sigma$ is the matrix of $\sigma$ w.r.t. the canonical basis of $\mathbb{k}^n$ as a vector space over $\mathbb{k}$. 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 $k(\text{free}(T))$) make them unique which completes the proof.

### 6.3.1 The PALGIE algorithm for the prime case

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

The goal of this section is to describe an algorithm, namely Algorithm 14, which given $C$, $R$ and $\overline{R}$, computes a polynomial set $\overline{C} \subset k[x]$ which is a regular chain for the total ordering $\overline{R}$ 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 $p \in k[x]$ and for a regular chain $\overline{C} \subset k[x]$ w.r.t. $\overline{R}$, the function call $\text{red}(p, \overline{C}, \overline{R})$ returns a polynomial $r \in k[x]$ such that $p - r \in \text{sat}(\overline{C}, \overline{R})$.

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. sat($C, R$) 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. \(\square\)

**Proposition 10.** Algorithm 9 satisfies its specifications.

**Proof.** This follows easily from the specifications of Algorithm 8. \(\square\)

**Proposition 11.** Algorithm 10 satisfies its specifications.

**Proof.** This follows easily from the specifications of Algorithm 8. \(\square\)

**Proposition 12.** Algorithm 11 satisfies its specifications.
6.3. Algorithm for linear change of coordinates

Algorithm 8: IsRegular($p, C, \mathcal{R}$)

Input: $C \subset k[x]$ a regular chain for the variable order $\mathcal{R}$ and a polynomial $p \in k[x]$. 
Output: pairs $(b_1, C_1), \ldots, (b_e, C_e)$ where each $b_i$ is a Boolean and each $C_i \subset k[x]$ 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. sat($C_i, \mathcal{R}$) if and only if $b_i$ is true, (3) $p$ is zero modulo sat($C_i, \mathcal{R}$) if and only if $b_i$ is false, and (4) at most one $b_i$ is true.

begin
1 \textbf{if } p = 0 \textbf{ then} \\
2 \hspace{1em} \textbf{return} (false, $C$) ; \\
3 \textbf{if } p \in k \textbf{ then} \\
4 \hspace{1em} \textbf{return} (true, $C$) ; \\
5 \textbf{if } C = \emptyset \textbf{ then} \\
6 \hspace{1em} \textbf{return} (true, $C$) ; \\
7 \textbf{if } (\forall q \in C \text{ deg}(p, \text{mvar}(q)) = 0 \textbf{ then} \\
8 \hspace{1em} \textbf{return} (true, $C$) ; \\
9 \hspace{1em} \text{Let } v \text{ be the largest variable in } p \text{ which is algebraic in } C ; \\
10 \hspace{1em} r := \text{res}(p, C_v, v) ; \\
11 \hspace{1em} \textbf{if } r = 0 \textbf{ then} \\
12 \hspace{2em} g := \gcd(p, C_v) ; \\
13 \hspace{2em} C_1 := C_v^- \cup \{g\} \cup \overline{C}_v^+ ; \\
14 \hspace{2em} q := C_v/g ; \\
15 \hspace{2em} C_2 := C_v^- \cup \{q\} \cup \overline{C}_v^+ ; \\
16 \hspace{2em} \textbf{output} (false, $C_1$) ; \\
17 \hspace{2em} \textbf{return} \text{IsRegular}(p, C_2, \mathcal{R}) ; \\
18 \textbf{if } r \in k \textbf{ then} \\
19 \hspace{1em} \textbf{return} (true, $C$) ; \\
20 \hspace{1em} \textbf{return} \text{IsRegular}(r, C, \mathcal{R}) ; \\
21 \textbf{end}
Algorithm 9: Saturate\((C, H, \mathcal{R})\)

**Input:** \(C \subseteq k[x]\) a regular chain for the variable order \(\mathcal{R}\) and a polynomial set \(H \subseteq k[x]\).

**Output:** \(D \subseteq k[x]\) 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.  
3.  for \(h \in H\) do
4.    
5.      for (bool, \(D\)) \in IsRegular\((h, C, \mathcal{R})\) do
6.        if bool then
7.          \(C := D;\)
8.          break;
9.    return \(C;\)
10. end

Algorithm 10: Extend\((C, D, \mathcal{R})\)

**Input:** \(C, D \subseteq k[x]\) 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:** \(E \subseteq k[x]\) 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.  
3.  \(E := C;\)
4.  repeat
5.    
6.      Let \(p \in D\) with minimum rank in \(\mathcal{R}\);  
7.      \(D := D \setminus \{p\}\);  
8.      for (bool, \(F\)) \in IsRegular\((\text{init}(p, \mathcal{R}), E, \mathcal{R})\) do
9.        if bool then
10.          \(p := \text{PrimitivePart}(\text{red}(p, F, \mathcal{R}, \text{mvar}(p, \mathcal{R})));\)
11.          \(E := F \cup \{p\};\)
12.          break;
13. until \(D = \emptyset;\)
14. return \(E;\)
15. end
Algorithm 11: EnsureRank \((p, \overline{R}, C, \mathcal{R})\)

**Input:** \(\mathcal{R}\) and \(\overline{R}\) total orders on \(x\); \(C \subseteq k[x]\) a regular chain for \(\mathcal{R}\) and a polynomial \(p \in k[x]\) such that \(p \in I\).

**Output:** a polynomial \(r \in k[x]\) and two polynomial sets \(P', H' \subseteq k[x]\) such that \(p - r \in \langle P' \rangle\), \(P' \subseteq I\) and \(H' = \{\text{init}(r, \overline{R})\}\), if \(r \notin k\), otherwise \(H' = \emptyset\).

1. \(P' := \emptyset\);
2. while \(p \neq 0\) and \(\text{init}(p, \overline{R}) \in \text{sat}(C, \mathcal{R})\) do
3. \(P' := P' \cup \{\text{init}(p, \overline{R})\}\);
4. \(p := \text{tail}(p, \overline{R})\);
5. if \(p \neq 0\) then
6. \(H' := \{\text{init}(p, \overline{R})\}\);
7. return \((p, P', H')\);
8. end

Algorithm 12: EnsureLeadingCoefficient \((p, v, \overline{R}, C, \mathcal{R})\)

**Input:** \(\mathcal{R}\) and \(\overline{R}\) total orders on \(x\); \(C \subseteq k[x]\) a regular chain for \(\mathcal{R}\) and a polynomial \(p \in k[x]\) such that \(p \in I\) and \(v \in x\) a variable.

**Output:** a polynomial \(r \in k[x]\) and two polynomial sets \(P', H' \subseteq k[x]\) such that \(p - r \in \langle P' \rangle\), \(P' \subseteq I\) and \(H' = \{\text{lc}(r, v)\}\), if \(r \notin k\), otherwise \(H' = \emptyset\).

1. \(P' := \emptyset\);
2. while \(p \neq 0\) and \(\text{lc}(p, v) \in \text{sat}(C, \mathcal{R})\) do
3. \(P' := P' \cup \{\text{lc}(p, v)\}\);
4. \(p := \text{reductum}(p, v)\);
5. if \(p \neq 0\) then
6. \(H' := \{\text{lc}(p, v)\}\);
7. return \((p, P', H')\);
8. end
Algorithm 13: Gcd\(_n(q, p, v, \overline{C}_v, \overline{R}, C, R)\)

**Input:** \(R\) and \(\overline{R}\) total orders on \(x\); \(v \in x\) a variable; \(q, p \in k[x]\) two non-constant polynomials with common main variable \(v\) w.r.t. \(\overline{R}\) s.t. \(\deg(q, v) > \deg(p, v)\) holds; \(C \subset k[x]\) a regular chain for \(R\) s.t. \(I := \text{sat}(C, R)\) is prime; \(\overline{C}_v \subset k[x]\) a regular chain for \(\overline{R}\) where all variables are less than \(v\) w.r.t. \(\overline{R}\) and both init\((p, \overline{R})\), init\((q, \overline{R})\) are regular w.r.t. sat\((\overline{C}_v, \overline{R})\).

**Output:** a polynomial \(g \in k[x]\) s.t. \(g\) is a GCD of \(p, q\) in \(L[v]\) where \(L\) is the total ring of fractions of \(k[x_i \in x \mid x_i < \overline{R} v]/\text{sat}(\overline{C}_v, \overline{R})\) s.t. if both \(q, p\) belong to \(I\) then \(g \in I\) and \(\deg(g, v) > 0\) both hold; two polynomial sets \(P, H \subset k[x]\) s.t. (1) \(g\) belongs to the ideal generated by \(q, p\) and \(P\); (2) we have \(P \subset I\) and (3) \(H\) is the set of the initials of the intermediate pseudo-remainders from \(q, p\) to \(g\).

\begin{verbatim}
1 begin
2     \((p_1, p_2) := (q, p)\);  
3     \((P, H) := (\emptyset, \emptyset)\);  
4     while true do  
5         \(\delta := \deg(p_1, v) - \deg(p_2, v)\);  
6         \(\psi := -1\);  
7         \(\beta := (-1)^{\delta+1}\);  
8         \(\alpha := \text{lc}(p_2, v)\);  
9             while \(\deg(p_2, v) > 0\) do  
10                \(p_3 := \text{exquo}(\text{prem}(p_1, p_2, v), \beta)\);  
11                \(d_3 := \deg(p_3, v)\);  
12                \((p_3, P', H') := \text{EnsureLeadingCoefficient}(p_3, v, \overline{C}_v, \overline{R}, \overline{R})\);  
13                \((P, H) := (P \cup P', H \cup H')\);  
14                \(p_3 := \text{red}(p_3, \overline{C}_v, \overline{R})\);  
15                if \(p_3 = 0\) then  
16                    return\((p_2, P, H)\);  
17                if \(\deg(p_3, v) < d_3\) then  
18                    \((p_1, p_2) := (p_2, p_3)\);  
19                    break;  
20                if \(\delta > 0\) then  
21                    \(\psi := \text{exquo}((-\alpha)^{\delta}, \psi^{\delta+1})\);  
22                    \((p_1, p_2) := (p_2, p_3)\);  
23                if \(\deg(p_2, v) > 0\) then  
24                    \(\delta := \deg(p_1, v) - \deg(p_2, v)\);  
25                    \(\beta := -\alpha \psi^{\delta}\);  
26                    \(\alpha := \text{lc}(p_2, v)\);  
27                if \(p_2 = 0\) then  
28                    return\((p_1, P, H)\);  
29                else  
30                    return\((p_2, P, H)\);  
31 end
\end{verbatim}
Algorithm 14: Palgie($C, \mathcal{R}, \overline{\mathcal{R}}$)

**Input:** $\mathcal{R}$ and $\overline{\mathcal{R}}$ total orders on $x$; $C \subset k[x]$ a regular chain for $\mathcal{R}$ s.t. $I := \text{sat}(C, \mathcal{R})$ is prime.

**Output:** $\overline{C} \subset k[x]$ a regular chain for $\overline{\mathcal{R}}$ s.t. $\text{sat}(\overline{C}, \mathcal{R}) = \text{sat}(\overline{C}, \overline{\mathcal{R}})$.

```plaintext
begin

1. $P := C$;
2. $H := \{\text{init}(p, \mathcal{R}) \mid p \in C\}$;
3. $\overline{C} := \emptyset$;
4. repeat
5. Let $p \in P$ with minimum rank in $\overline{\mathcal{R}}$;
6. $P := P \setminus \{p\}$;
7. $p := \text{red}(p, \overline{C}, \overline{\mathcal{R}})$;
8. $(p, P', H') := \text{EnsureRank}(p, \overline{\mathcal{R}}, C, \mathcal{R})$;
9. $(P, H) := (P \cup P', H \cup H')$;
10. if $p \neq 0$ then
11. $v := \text{mvar}(p)$;
12. $\overline{C} := \text{Saturate}(\overline{C}, \{\text{init}(p, \overline{\mathcal{R}})\}, \overline{\mathcal{R}})$;
13. if $(\forall q \in \overline{C}) \text{mvar}(q) \neq v$ then
14. $\overline{C} := \text{Extend}(\overline{C}_v \cup \{p\}, \overline{C}_v^+, \overline{\mathcal{R}})$;
15. else
16. $(g, P', H') := \text{Gcd}_n(\overline{C}_v, p, \overline{C}_v^+, \overline{\mathcal{R}}, C, \mathcal{R})$;
17. $(P, H) := (P \cup P', H \cup H')$;
18. $\overline{C} := \text{Saturate}(\overline{C}, \{\text{init}(g, \overline{\mathcal{R}})\}, \overline{\mathcal{R}})$;
19. $\overline{C} := \overline{C} \setminus \{\overline{C}_v\} \cup \{g\}$;
20. $\overline{C} := \text{Saturate}(\overline{C}, H, \overline{\mathcal{R}})$;
21. until $P = \emptyset$
22. return(\overline{C});
end
```
Proposition 13. Algorithm $\text{Algorithm}_{13}$ 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 $\text{Algorithm}_{13}$ computes a GCD of $q, p$ in the ring $L[v]$. Note that $L$ 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 $\text{Algorithm}_{13}$ 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 $k[x_j \in x \mid x_j \not<_{\overline{R}} v][v]$ are indeed exact when $p_3$ is replaced by $\text{red}(p_3, \overline{C}_v, \overline{R})$. 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 $\lambda_3$ is any regular element in the ring $L$.

The last (non-trivial) point that remains to be explained is the following claim: if both $q, p$ belong to $I$ then $\deg(g, v) > 0$ holds. Hence we assume that $q, p \in I$ holds. Observe that, by construction, the polynomial $g$ belongs to $I$ as well. Assume, by contradiction, that $\deg(g, v) = 0$ holds. At Line (12), the input polynomial $p_3$ of EnsureRank belongs to $I$; 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 $\text{init}(p'_3, \overline{R}) \in \text{sat}(\overline{C}_v, \overline{R})$. However, by construction, the polynomials of $\overline{C}$ belong to $I$ while each of their initials does not (see Proposition [16]). Hence, we have $\text{sat}(\overline{C}_v, \overline{R}) \not\subseteq I$. Therefore, we would have $\text{init}(p'_3, \overline{R}) \in I$, 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 $\text{Algorithm}_{14}$

(i) $P \subseteq I$,

(ii) $\overline{C}$ is a regular chain for $\overline{R}$,

(iii) every polynomial $h \in H$ is regular w.r.t. $\text{sat}(\overline{C}, \overline{R})$.

Proof. We first prove (i). At Line (2), the set $P$ is initialized with $C$ which satisfies $C \subseteq C : H = I$. At Lines (10) and (18), the set $P$ is incremented directly or indirectly by the procedure EnsureRank which adds to $P$ polynomials belonging to $\text{sat}(C, \overline{R}) = I$. 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 $\overline{C} = 0$; 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 $\overline{R}$. 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 $p \in I$ from (i) and $I$ is not the entire
polynomial ring $k[x]$. Hence, the main variable $v$ of $p$ is well-defined. After executing Line (13), the initial of $p$ w.r.t. $R$ is regular w.r.t. $\text{sat}(C, R)$, thus $C_v \cup \{p\}$ is a regular chain for $R$. If $v$ is not algebraic in $C$, then the specifications of $\text{Extend}$ imply that $C$ is still a regular chain after executing (15). If $v$ is algebraic in $C$, then the specifications of $\text{RegularGcd}$ imply that $C$ 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 $R$ may not be regular w.r.t. $C_v$. 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 $\text{Saturate}$. 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. $\text{sat}(C, R)$ 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 $C$ 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 $C_i$ the value of the regular chain $C$ at the end of that iteration. Let $\ell$ 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 $\text{sat}(C_i, \overline{R}) \subseteq \text{sat}(C_{i+1}, \overline{R})$,

(ii) for all $1 \leq i \leq \ell$, we have $p_i \in \text{sat}(C, \overline{R})$.

**Proof.** We first prove (i). It clearly follows from the specifications of $\text{Saturate}$ that at each of the Lines (13), (19), (21) the input and output values of $C$, say $C_{\text{in}}$ and $C_{\text{out}}$, satisfy $\text{sat}(C_{\text{in}}, \overline{R}) \subseteq \text{sat}(C_{\text{out}}, \overline{R})$. Similarly, if we denote by $C_{\text{in}}$ and $C_{\text{out}}$, the regular chains computed at Line (13) and (15) respectively, we have again $\text{sat}(C_{\text{in}}, \overline{R}) \subseteq \text{sat}(C_{\text{out}}, \overline{R})$. 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 $p_\ell$ is proved to be zero modulo $\text{sat}(C, \overline{R})$ at Line (8),
- or $p_\ell$ is added to $C$ at Line (15),
- or $C_v$ is replaced by $g = \text{Gcd}_v(C_v, p_\ell, C_v, \overline{R}, C, \overline{R})$ at Line (20).

In each of these three cases, it is clear that we have $p_\ell \in \text{sat}(C, \overline{R})$. 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 $\text{sat}(C, \overline{R})$. We shall prove that $p_i$, with $i = \ell - j$, also satisfies $p_i \in \text{sat}(C, \overline{R})$. Observe that after Line(9):

- either $p_i$ was proved to be zero modulo $\text{sat}(C, \overline{R})$ 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 \( \text{sat}(\overline{C}, \overline{R}) \). Hence, it is enough to prove that \((p') \in \text{sat}(\overline{C}, \overline{R})\). 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. \( \square \)

**Proposition 15.** At the end of last iteration of the **while**-loop of the execution of Algorithm 14 the properties hold:

(i) \( C \subseteq \text{sat}(\overline{C}, \overline{R}) \),

(ii) \( \text{sat}(C, \mathcal{R}) \subseteq \text{sat}(\overline{C}, \overline{R}) \).

**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. \( \square \)

**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. \( \text{sat}(C, \mathcal{R}) \),

(ii) \( \overline{C} \subseteq \text{sat}(C, \mathcal{R}) \),

(iii) \( \text{sat}(\overline{C}, \overline{R}) \subseteq \text{sat}(C, \mathcal{R}) \).

**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 \( I \); 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 \( I \) 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 \( I \), 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).

\[ \square \]

**Theorem 12.** Algorithm 14 satisfies its specifications.

**Proof.** From Proposition 15 we have \( \text{sat}(C, R) \subseteq \text{sat}(\overline{C}, \overline{R}) \), while from Proposition 16 we have \( \text{sat}(\overline{C}, \overline{R}) \subseteq \text{sat}(C, R) \). Moreover, from Proposition 14, the triangular set \( \overline{C} \) is a regular chain, which completes the proof.

\[ \square \]

### 6.3.2 Regularity test in \( \text{IsRegular}(p, \overline{C}, \overline{R}) \)

The motivation is to prevent from checking that \( p \) is regular w.r.t. \( \text{sat}(\overline{C}, \overline{R}) \) whenever this could be deduced from previous computations. To this end, let us assume that \( \text{IsRegular}(p, \overline{C}, \overline{R}) \) is called by a wrapper-function \( \text{IsRegular}(p, \overline{C}, \overline{R}, T) \) where the fourth argument \( T \) is a table whose keys are polynomials and values are bit vectors. The wrapper-function \( \text{IsRegular}(p, \overline{C}, \overline{R}, T) \) works as follows:

1. if \( p \) is not a key of \( T \), then

   (a) \( \text{IsRegular}(p, \overline{C}, \overline{R}) \) is run; let (true, \( D \)) 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 \( x \) w.r.t. \( \overline{R} \) is algebraic in \( D \).

   (c) we add \((p, V(p))\) to \( T \) and return what \( \text{IsRegular}(p, \overline{C}, \overline{R}) \) 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 \( \overline{R} \) and \( \ell \) be its rank in \( \overline{R} \);

   (c) if the first \( \ell \) bits of \( V(p) \) and \( V_0(p) \) agree, then no computation is needed and (true, \( \overline{C} \)) is returned;

   (d) otherwise, \( \text{IsRegular}(p, \overline{C}, \overline{R}) \) is run; let (true, \( D \)) be the returned pair item whose first item is true; we update \( V(p) \) into \( T \) using \( D \) and we return what \( \text{IsRegular}(p, \overline{C}, \overline{R}) \) computed.
6.3.3 The PALGIE algorithm for linear change of coordinates

We turn our attention back to Problem 1. Let $T \subset \mathbb{k}[x]$ be a regular chain and let $A$ be a linear change of coordinates in $\overline{\mathbb{k}}$. We denote by $\text{sol}_A$ solving of the polynomial system $S$ to a solution of Problem 1. Let $T_0 = \{t_{d+1}, \ldots, t_n\}$ 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

$$
\begin{align*}
  t_n^A(x) &= 0 \\
  \vdots & \vdots \\
  t_{d+1}^A(x) &= 0 \\
  h_{d+1}(x) \cdots h_n^A(x) &\neq 0
\end{align*}
$$

We denote by $Z(S) \subset \overline{\mathbb{k}}$ the zero set of $S$. Observe that for all polynomials $f \in \mathbb{k}[x]$, we have

$$
f \in \langle Z(S) \rangle \iff f^{A^{-1}} \in \sqrt{\text{sat}(T)}.
$$

where $\langle Z(S) \rangle$ is the ideal of $\mathbb{k}[x]$ 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 $\mathcal{P} \subset \mathbb{k}[x]$ 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 $\mathcal{P} \subset \mathbb{k}[x]$ be a (proper) prime ideal and $G$ the reduced lexicographical Gröbner basis of $\mathcal{P}$. Recall that $x$ counts $n$ variables ordered as $x_1 < \cdots < x_n$. We assume that $\mathbb{k}$ is an infinite field. We denote by $T_\mathcal{P}$ the set defined by

$$
T_\mathcal{P} = \{v \in x \mid (\forall g \in G) \text{ mvar}(g) \neq v\}.
$$

This set satisfies two important properties:

- $T_\mathcal{P}$ is algebraically independent modulo $\mathcal{P}$ that is, $\mathcal{P} \cap \mathbb{k}[T_\mathcal{P}] = \langle 0 \rangle$,
- the number of elements in $T_\mathcal{P}$ gives the dimension of $\mathcal{P}$, that is, $\text{dim}(\mathcal{P}) = \text{card}(T_\mathcal{P})$.

A variable $x_s \in x$ is said integral over $\mathbb{k}[x_1, \ldots, x_{s-1}]$ modulo $\mathcal{P}$ if there exists $f \in \mathcal{P} \cap \mathbb{k}[x_1, \ldots, x_{s-1}, x_s]$ such that $\text{mvar}(f) = x_s$ and $\text{init}(f) \in \mathbb{k}$. Integral variables satisfy two important properties:

- A variable $x_s \in x$ is integral over $\mathbb{k}[x_1, \ldots, x_{s-1}]$ modulo $\mathcal{P}$ if and only if there exists $g \in G$ such that $\text{lm}(g) = x_s^{d_s}$ for some positive integer $d_s$,
- if a variable $x_s \in x$ is integral over $\mathbb{k}[x_1, \ldots, x_{s-1}, u]$ modulo $\mathcal{P}$, with $u \subseteq T_\mathcal{P}$ disjoint from $\{x_1, \ldots, x_s\}$, then $x_s$ is also integral over $\mathbb{k}[x_1, \ldots, x_{s-1}]$ modulo $\mathcal{P}$. 


6.4. Noether normalization and regular chains

Thanks to the above properties, we may assume w.l.o.g. that if \( d = \dim(P) \) then we have \( T_P = \{x_1, \ldots, x_d\} \). Consider a linear change of coordinates \( A \) in \( k^n \) defined by a matrix \( M \) of the following form:

\[
M = \begin{pmatrix}
I_{d \times d} & a_{1,d+1} & \cdots & a_{1,n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{d,d+1} & \cdots & a_{d,n} \\
0 & & & I_{(n-d) \times (n-d)}
\end{pmatrix}
\]  

(6.8)

where \( a_{i,j} \in k \). We denote by \( P^A \) the ideal generated by \( f^A \) for all \( f \in P \). 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 \( P^A \),
2. \( x_{d+i} \) is integral over \( k[x_1, \ldots, x_d] \) modulo \( P^A \) for all \( i = 1, \ldots, n - d \).

In this case, we say that \( P^A \) is in Noether position.

We turn our attention to the regular chain representation of the prime ideal \( P \). 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 \( P \). Let \( H \) be the reduced lexicographical \( \text{Gröbner} \) basis of \( P^A \) 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 init\((f) \in k \).

**Proof.** Assume by contradiction that there exists \( f \in C \) such that init\((f) \notin k \) and let us choose such an \( f \) with minimum main variable. Since \( x_1, \ldots, x_d \) are algebraically independent modulo \( P^A \) and since \( C \) is a regular chain, one can compute a polynomial \( f' \) such that init\((f') \in k[x_1, \ldots, x_d] \) and sat\((C') = \text{sat}(C) \) holds with \( C' = C \setminus \{f\} \cup \{f'\} \).

Let \( \text{mvar}(f) = x_r \). Since \( P^A \) 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 \( x_r^d \). Since init\((H_{x_r}) \in k \), we have

\[
\deg(f', x_r) = \deg(f, x_r) < d_r = \deg(H_{x_r}, x_r).
\]  

(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 \( C' = k[x_1, \ldots, x_r] \) is a basis of \( P^A \cap k[x_1, \ldots, x_r] \). Therefore, the polynomial \( H_{x_r} \) reduces to zero through multivariate division by \( C' \cap k[x_1, \ldots, x_r] \) and thus by \( C \cap k[x_1, \ldots, x_r] \). This contradicts the fact that \( H \) is a reduced \( \text{Gröbner} \) basis.

**Remark 12.** Theorem [13] states that if \( T \) generates sat\((T) \) and \( P^A \) is in Noether position, then \( C \) is monic. Unfortunately, if \( T \) does not generate sat\((T) \), then the previous conclusion may not hold as shown by the following example.

**Example 11.** Consider the regular chain \( T := \{x_5^5 - x_1^4, x_1x_3 - x_2^3\} \subset \mathbb{Q}[x_1 < x_2 < x_3] \) 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 $(T)^A$ 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 $c_1 = x_2^3 - 2x_2^4 + x_2^3 + 4x_1^2x_2^2 - x_1^4$ and $c_2 = (-x_1^3 + 2x_2^2x_1)x_3 + x_1^2x_2^2 - x_1^2 + x_2^2$. As you can see $\text{init}(c_2) \not\in \mathbb{Q}$.

### 6.5 Applications of random linear changes of coordinates

Let $T \subset k[x]$ be a regular chain whose saturated ideal has dimension $d$. Let $u$ 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 $\mathbb{K}$. 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 $C \subset k[x]$, thus satisfying

$$W^A(T) = \overline{W(C)}.$$  \hfill (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 $W(T)$ by one for which $\overline{W(T)} \cap V(h_T)$ can be computed by set-theoretic operations on constructible sets (represented by regular chains 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 $k[u]$ 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]. $\square$

**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 $k[x]$, 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 $W^A(T) = \overline{W(C)}$ 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 $k[x]$, 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
sat(C), and, from the specifications of the PALGIE algorithm, a basis of sat(T)^4 as well. Thus 
\( C_{4,1} := \{ f_{4,1}^{-1} \mid f \in C \} \) is a basis of 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 \( k(m_{ij}) \). Moreover, the base field \( k \) is either \( \mathbb{R} \) or \( \mathbb{C} \) 
so that the affine space \( \mathbb{K}^d \) 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 \( V(r_{T}^4, r_{C}) \) is empty, we have

\[
\lim(W(T)) = \{ A^{-1}(y) \mid y \in V(h_{T}^4) \cap W(C) \}. 
\]

**Proof.** Observe first that \( V(r_{T}^4, r_{C}) \) is empty if and only if \( V(r_{T}, r_{T}^{-1}) \) is empty. Observe next 
that any zero \( \zeta \in \mathbb{K}^d \) of \( h_{T} \) extends a zero \( \zeta' \in \mathbb{K}^d \) of \( r_{T} \), see [24]. Therefore, for any choice of 
the parameters \( (m_{ij}) \) such that \( V(r_{T}^4, r_{C}) \) 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 \( V(h_{T}^4) \cap W(C) \) are the images 
by \( A \) of the desired limit points of \( W(T) \).

**Example 12.** Consider the regular chain \( T := \{ x_4, x_2 x_3 + x_1^2 \} \subset \mathbb{Q}[x_1 < x_2 < x_3 < x_4] \) 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 \( C := \{ x_4, x_2^3 + x_2 x_3 + x_1^2 \} \) and consequently, 
\( r_{T}^4 = x_1^2 \) and \( r_{C} = 1 \). Then \( \langle r_{T}^4, r_{C} \rangle = \langle 1 \rangle \) holds. Using Triangularize command of Maple, 
one can get

\[
\langle C, h_{T}^4 \rangle^{A^{-1}} = \langle x_4, x_2, x_1 \rangle = \lim(W(T)). 
\]

**Corollary 2.** Assume that \( T \) generates \( \text{sat}(T) \). Then we have

\[
\lim(W(T)) = V(T) \setminus W(T) 
\]

**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 $\text{sat}(T)$

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 following equalities:

$$V(\text{sat}(T)) = W(T) = \left( W(T) \cap V(h_T) \right) \cup W(T) = \lim(W(T)) \cup W(T).$$

Therefore, computing $\lim(W(T))$ and computing $V(\text{sat}(T))$ 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 $V(\text{sat}(T))$ that would avoid Gröbner basis calculation.

**Lemma 28.** Let $I$ be a radical ideal of $k[x]$. Let $h \in k[x]$. Assume that the dimension of any associated prime $p$ of $I$ is at least $d$. Then $\dim(V(I, h)) < d$ implies that $h$ is regular modulo $I$. If the dimension of any associated prime $p$ of $I$ is $d$, that is, if $I$ is an unmixed ideal of dimension $d$, then $\dim(V(I, h)) < d$ holds if and only if $h$ is regular modulo $I$.

**Proof.** Let $I = \cap_{i=1}^s p_i$, where $p_i$ are the associated prime of $I$. Assume that $\dim(V(I, h)) < d$, it is enough to show that $h$ does not belong to any $p_i$. On the other hand, we have $V(I, h) = \bigcup_{i=1}^s V(p_i, h)$. If $h$ belongs to some $p_i$, then $V(p_i, h) = V(p_i)$. Since $\dim(p_i) \geq d$, we know that $\dim(V(I, h)) \geq d$, which is a contradiction to the assumption that $\dim(I, h) < d$.

If $I$ is an unmixed ideal of dimension $d$, by the above argument, $\dim(V(I, h)) < d$ implies that $h$ is regular modulo $I$. On the other hand, if $h$ is regular modulo $I$, then $h$ does not belong to any $p_i$. Thus $\dim(V(I, h)) = \max(\dim(V(p_i, h))) < \max(\dim(p_i)) = d$. \hfill $\Box$

**Theorem 15.** Let $T \subset k[x]$ 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 $\sqrt{T} = \sqrt{\text{sat}(T)}$ if and only if $\dim(V(T, h_T)) < d$ holds.

**Proof.** First we claim that for any associated prime $p$ of $\sqrt{T}$, we have $\dim(p) \geq n - d$. To prove this, we first notice that the associated primes $p$ of $\sqrt{T}$ are exactly the minimal associated primes $p$ of $\langle T \rangle$. On the other hand, since $\langle T \rangle \subseteq \text{sat}(T)$ and $V(\text{sat}(T)) \neq \emptyset$, we know that $\langle T \rangle$ generates a proper ideal. By Krull’s principle ideal theorem, for any minimal associated prime $p$ of $\langle T \rangle$, the height of $p$ is less than or equal to $|T|$. Since $|T| = n - d$, we have $\dim(p) \geq n - d$. The claim is proved.

Now we prove that we have $\sqrt{T} = \sqrt{\text{sat}(T)}$ 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 $\sqrt{T}$. Thus, we have $\sqrt{\text{sat}(T)} = \sqrt{T} : h_T^n = \sqrt{T} : h_T^n = \sqrt{T}$. Next, we show that the condition is necessary. If $\sqrt{T} = \sqrt{\text{sat}(T)}$, then $\sqrt{T}$ is an unmixed ideal and $h_T$ is regular modulo $\sqrt{T}$. Thus, $\dim(V(T, h_T)) < d$ holds by Lemma 28. \hfill $\Box$

**Remark 13.** As an immediate corollary, we have $V(T) = W(T)$ 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 Kalkbrenner triangular decomposition. We denote by IsClosure a procedure to test $V(T) = W(T)$, by applying Theorem 15.
Example 13. Consider the regular chain \( T := \{x_1 x_2 + x_1, x_1 x_3 + 1\} \) 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 \( \sqrt{\langle T \rangle} = \sqrt{\text{sat}(T)} \). Actually \( \langle T \rangle = \text{sat}(T) \) also holds.

Theorem 16. Let \( T \) be a regular chain of \( k[x] \) with free variables \( x_1, \ldots, x_d \). Let \( C_1, \ldots, C_s \subset k[x] \). Assume that \( \langle C_i \rangle \subseteq \sqrt{\text{sat}(T)} \) holds, for all \( i = 1, \ldots, s \). Let \( \mathcal{I} = \langle T, C_1, \ldots, C_s \rangle \). Then \( \sqrt{\text{sat}(T)} = \sqrt{\mathcal{I}} \) if and only if there exist regular chains \( T_i, i = 1, \ldots, t, \) such that each of the following properties hold:

(i) \( \sqrt{\mathcal{I}} = \cap_{i=1}^t \sqrt{\text{sat}(T_i)} \),

(ii) \( |T_1| = \cdots = |T_t| = n - d \),

(iii) \( h_T \) is regular modulo all \( \sqrt{\text{sat}(T)} \).

Proof. The direction “\( \Rightarrow \)” obviously holds. Next we prove the direction “\( \Leftarrow \)”. By (i) and (ii), we know that \( \sqrt{\mathcal{I}} \) is an unmixed ideal of dimension \( d \). Since \( h_T \) is regular modulo all \( \sqrt{\text{sat}(T)} \), by Lemma 28 we have \( \dim(V(h_T, \text{sat}(T))) < d \). 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 \( \langle \mathcal{I} \rangle \subseteq \mathcal{I} \), thus we deduce that \( \sqrt{\text{sat}(T)} \subseteq \sqrt{\mathcal{I}} \). Since \( I = \langle T, C_1, \ldots, C_s \rangle \) and \( \langle C_i \rangle \subseteq \sqrt{\text{sat}(T)} \), we also have \( \mathcal{I} \subseteq \sqrt{\text{sat}(T)} \). 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 \( \sqrt{\text{sat}(C_i)} = \sqrt{\text{sat}(T)} \) 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 \( \text{W}(\mathcal{I}) = \text{V}(\mathcal{I}) \) 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 \( \text{W}(\mathcal{I}) \) in some cases, see Algorithm 15.

Example 14. We illustrate Algorithm 15 on one example. Consider the regular chain \( T := \{x_2^5 - x_1^2, x_1 x_3 - x_2^2(x_2 + 1)\} \) 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 \( V(T) \neq V(\text{sat}(T)) \). Let \( C := \{x_2 x_3^2 - x_1^3 - 2x_2 - 1, x_3 x_1 - x_3^2 - x_2^3\} \) be another regular chain of \( \mathbb{Q}[x_1 < x_2 < x_3] \). One can verify that \( \text{sat}(C) = \text{sat}(T) \) 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 \( \text{V}(\text{sat}(T)) = \text{V}(\mathcal{I}) \).

Remark 15. We selected 22 one-dimensional non-primitive regular chains to test Algorithm 15. For 10 of them, the algorithm could successfully compute \( \text{W}(\mathcal{I}) \). 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 \( \text{W}(\mathcal{I}) \) for 10 of those 16 examples.
Algorithm 15: Closure(T)

Input: A non-empty regular chain $T$ of $k[x_1 < \cdots < x_n]$.
Output: Return $\emptyset$ or a polynomial set $G$ such that $W(T) = V(G)$. If $\emptyset$ is returned, this means that the algorithm fails to compute $W(T)$.

begin
  $G := \emptyset$;
  for $i$ from 1 to $n$ do
    if $i = 1$ then
      $C := T$;
    else
      let $R$ be the ordering $x_i < x_{i+1} < \cdots < x_n < x_1 \cdots < x_{i-1}$;
      $D := \text{PALGIE}(T, R)$;
      if $|D| \neq 1$ then
        return $\emptyset$
      else
        let $C$ be the only regular chain in $D$;
        if IsClosure($C$) then
          return $C$;
        else
          $G := G \cup C$;
          $D := \text{Triangularize}(G, \text{mode} = K)$ // compute a Kalkbrener triangular decomposition of $V(G)$
          if all regular chains in $D$ have dimension $d$ and $h_T$ is regular w.r.t. each of them then
            return $G$
          end
        end
      end
    end
  return $\emptyset$;
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 \( k[x_1 < \cdots < x_s] \). 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 \( \text{sat}(T_i) = \langle T_i \rangle \), 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 \( \text{tail}(t_i) \) is invertible modulo \( \langle \text{init}(t_i) \cup T_{i-1} \rangle \), then \( \langle T_i \rangle = \text{sat}(T_i) \) holds (see [27]). Suppose that \( \text{tail}(t_i) \) is not invertible modulo \( \langle \text{init}(t_i) \cup T_{i-1} \rangle \), then \( \langle \text{init}(t_i) \cup T_{i-1} \rangle \) generates a proper zero-dimensional ideal, since \( \text{init}(t_i) \) is regular modulo \( \langle T_{i-1} \rangle \). Let \( p \) be an associated prime of this ideal. If \( \text{tail}(t_i) \) is not regular modulo \( p \), then all the coefficients of \( t_i \) belong to \( p \). On the other hand, since \( t_i(x_1, x_i) \) is primitive, the ideal formed by the coefficients of \( t_i \) is the field \( k \), a contradiction. \( \square \)

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 \( \langle T' \rangle = \text{sat}(T') = \text{sat}(T) \).

Example 15. Let \( T := \{ x_1^3 - 2x_1, 3x_2^3 + 4x_1^4 \} \subset \mathbb{Q}[x_1 < x_2 < x_3] \) 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 \( T := \{ x_i x_3 + x_2, x_i x_4 + x_2 \} \), 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 \( x_4 - x_3 \in \text{sat}(T) \) because \( x_1 x_4 + x_2 = x_1(x_4 - x_3) \) modulo \( \langle x_1 x_3 + x_2 \rangle \).

Lemma 30. Let \( T \subset k[x] \) 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 \( x_1 < x_i < x \setminus \{ x_1, x_i \} \) such that \( \sqrt{\text{sat}(C_i)} = \sqrt{\text{sat}(T)} \). 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 \( W(T) = W(C_1) \), we know that \( W(T) \subseteq V(C_2, \ldots, C_n) \), 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 \( \text{init}(c_i) \subset k[x_1] \) 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. \( \square \)

Example 17. Let \( T := \{ x_2^2 - x_1^4, x_1 x_3 - x_2^2 \} \) be a regular chain of \( \mathbb{Q}[x_1 < x_2 < x_3] \). Let also \( C := \{ x_3^3 - x_1^3, x_2^3 - x_1^2 \} \) be a regular chain of \( \mathbb{Q}[x_1 < x_3 < x_2] \) for which we have \( \text{sat}(C) = \text{sat}(T) \). 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 \( x_1 < x_3 < x_2 \) 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 \( \text{sat}(T) \subseteq \text{sat}(D) \) holds. By Theorem 16 we conclude that \( \sqrt{\langle T, C \rangle} = \sqrt{\text{sat}(T)} \). However, for this example, Algorithm 15 fails to compute the set \( G \) such that \( W(T) = V(G) \), since \( T \) and \( D \) do not have the same height.
Lemma\cite{30} Example\cite{17} and Theorem\cite{16} show that it is possible to compute 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 $k[x]$ 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 $\sqrt{\text{sat}(C_1)} = \cdots = \sqrt{\text{sat}(C_n)}$. Does $\sqrt{\text{sat}(C_1)} = \sqrt{\bigcup_{i=1}^{n} C_i}$ always hold?

**Question 2.** Let $C_1, \ldots, C_n$ be polynomial sets of $k[x]$ 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 $\sqrt{\text{sat}(C_i)} = \sqrt{\text{sat}(C_j)}$ 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 $T := \{t_1, t_2\} \subseteq \mathbb{Q}[x_1 < x_2 < x_3 < x_4]$ is a regular chain of dimension two, where $t_1 = -93x_1x_2^2 + (53x_1 - 35)x_2 + 93x_1^3 - 26x_2^2 - 57x_1$ and $t_2 = 93x_1x_4 + (3233x_1 - 2135)x_2 + 5673x_1^3 + 213x_2^2 - 3477x_1)x_3 + (-530x_1^2 - 3091x_1)x_2 - 930x_1^4 + 6119x_3^2 + 570x_4^2 - 1767x_1$. One can verify that $T$ does not generate its saturated ideal.

Following the notations of Question\cite{7} 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\cite{7}. To see whether the statement of Question\cite{7} is true or not, on one hand, we can find the Kalkbrener triangular decomposition \{C_1, R_1, R_2\} for $V(\bigcup_{i=1}^{4} C_i)$ where $C_1 = T$, $R_1 := \{x_4 - 19, x_2, x_1\}$, and $R_2 := \{961x_4^3 + 42428x_4 + 279756, x_3, x_2, x_1\}$.

On the other hand, using methods based on Gröbner bases computations to find a generator for sat($C_1$), one can find the Kalkbrener triangular decomposition \{C_1, R_1\} for $V(\text{sat}(C_1))$.

Therefore, we have

$$V(\text{sat}(C_1)) = W(C_1) \cup W(R_1) \neq V(\bigcup_{i=1}^{4} C_i) = W(C_1) \cup W(R_1) \cup W(R_2).$$

This shows that the statement of Question\cite{7} 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 $\lim(W(C_1)) \neq V(C_1 \cup \{P_1, \ldots, P_4, H_1\})$. Consequently, for this example, the answer to both Questions\cite{7} and\cite{2} is negative.

In Example\cite{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 \( \text{sat}(T) \) has dimension \( d \) and let \( F \subset \text{sat}(T) \) 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. \( \square \)

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 \textit{PALGIE} to \( T \) w.r.t. the variable order \( x_3 < x_4 < x_1 < x_2 \) and consequently, “fish” the polynomial \( x_3 x_4 - 1 \in \text{sat}(T) \). Then
\[
V(T \cup F \cup \{h_T\}) = V(x_1 x_3 + x_2, x_2 x_4 + x_1, x_3 x_4 - 1, x_1 x_2) = V(x_1, x_2, x_3 x_4 - 1) = \lim(W(T)).
\]

6.7 Conclusion

Among all the methods we have considered for computing \( \lim(W(T)) \) and \( \text{sat}(T) \), those based on linear changes of coordinates seem very promising. They are a good trick for finding a subset \( F \subset \text{sat}(T) \) such that \( F \cup T \) is a basis of \( \text{sat}(T) \), 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 Co-
CoA, 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 algo-

rithm 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 \( h \)). Therefore, being able to compute the tangent cones of \( C \) at all its
points defined by a given regular chain \( h \), becomes a desirable operation. Similarly, and as dis-

cussed 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 $h$. 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 $k$ a field with algebraic closure $\overline{k}$, and by $A^{n+1}(\overline{k})$ the $(n+1)$-dimensional affine space over $\overline{k}$, for some positive integer $n$. Let $x := x_0, \ldots, x_n$ be $n+1$ variables ordered as $x_0 > \cdots > x_n$. We denote by $k[x]$ the corresponding polynomial ring. Let $h \subset k[x]$ be a subset and $h \in k[x]$ be a polynomial. We say that $h$ is regular modulo the ideal $\langle h \rangle$ of $k[x]$ whenever $h$ does not belong to any prime ideals associated with $\langle h \rangle$, thus, whenever $h$ is neither null nor a zero-divisor modulo $\langle h \rangle$. The algebraic set of $A^{n+1}(\overline{k})$ consisting of the common zeros of the polynomials in $h$ is written as $V(h)$. For $W \subset A^{n+1}(\overline{k})$, we denote by $I(W)$ the ideal of $k[x]$ generated by the polynomials vanishing at every point of $W$. The ideal $I(W)$ is radical and when $\overline{k} = k$ holds, Hilbert’s Nullstellensatz states that $\sqrt{\langle h \rangle} = I(V(h))$.

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 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 \( h \in k[x] \). Define \( V := V(h) \) and let \( p := (p_0, \ldots, p_n) \in V \) be a point. We denote by \( \dim_p(V) \) the maximum dimension of an irreducible component \( C \) of \( V \) such that we have \( p \in C \). Recall that the tangent space of \( V(h) \) at \( p \) is the algebraic set given by

\[
T_p(h) := V(\partial f/\partial x_0(p)) = V(f \partial f/\partial x_0(p))
\]

where \( \partial f/\partial x_0(p) \) is the linear part of \( f \) at \( p \), that is, the affine form \( \partial f/\partial x_0(p)(x_0 - p_0) + \cdots + \partial f/\partial x_n(p)(x_n - p_n) \). Note that \( T_p(h) \) is a linear space. We say that \( V(h) \) is smooth at \( p \) whenever the dimension of \( T_p(h) \) is \( \dim_p(V) \) and singular otherwise. The singular locus of \( V(h) \), denoted by \( \text{sing}(h) \), is the set of the points \( p \in V(h) \) at which \( V(h) \) is singular.

Let \( f \in k[x] \) be a polynomial of total degree \( d \) and \( p := (p_0, \ldots, p_n) \in \mathbb{A}^{n+1}k \) be a point such that \( f(p) = 0 \) holds. Let \( \alpha = (\alpha_0, \ldots, \alpha_n) \in \mathbb{N}^{n+1} \) be a \((n + 1)\)-tuple of non-negative integers. Denote: \( (x - p)^{\alpha} := (x_0 - p_0)^{\alpha_0} \cdots (x_n - p_n)^{\alpha_n} \), where \(|\alpha| = \alpha_0 + \cdots + \alpha_n \) is the total degree of \( x - p \). Since the polynomial \( f \in k[x] \) has total degree \( d \), it writes as a \( k \)-linear combination of the form:

\[
f = \sum_{|\alpha| = 0} c_\alpha (x - p)^\alpha + \cdots + \sum_{|\alpha| = d} c_\alpha (x - p)^\alpha
\]

with all coefficients \( c_\alpha \) belonging to \( k \). Each summand \( h_{c_\alpha, \alpha} = \sum_{|\alpha| = \alpha} c_\alpha (x - p)^\alpha \) is called the homogeneous component in \( x - p \) of \( f \) in degree \( \alpha \). Moreover, the homogeneous component of least degree of \( f \) in \( x - p \) is given by \( h_{c_\alpha, \alpha} = \sum_{|\alpha| = \alpha} c_\alpha (x - p)^\alpha \) where \( j_{\min} = \min(j \in \mathbb{N}^2 : h_{c_\alpha, \alpha} \neq 0) \).

**Definition 21** (Tangent Cone of a Curve). Let \( C \subset \mathbb{A}^{n+1}k \) 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 \( TC_p(C) = V(h_{c_\alpha, \alpha} : \alpha \in \mathbb{N}^2 : h_{c_\alpha, \alpha} \neq 0) \).

One can show that \( TC_p(C) \) consists of finitely many lines, all intersecting at \( p \).

\[\text{Figure 7.1: A figure displays the typical "fish" curve, which is a planar curve given by } h = y^2 - x^2(x + 1) \in \mathbb{Q}[x,y]. \text{ Clearly, two tangent lines are needed to form a "linear approximation" of the curve at the origin. Elementary calculations show these two lines actually form the tangent cone of the fish curve at the origin.}\]
If \( I(C) \) is generated by a single polynomial then computing \( TC_p(C) \) is easy. Otherwise, this is a much harder computation. Let \( h \subset k[x] \) be such that \( V(h) = C \). As pointed out by Mora et al. in [66], one can compute \( \langle \text{nc}_p(f; \min) : f \in I(C) \rangle \) via a graded Gröbner basis, say \( G \), of the homogenization of \( h \) (a process where an additional variable \( x_{n+1} \) is used to make every \( h \in h \) a homogeneous polynomial in \( k[x][x_{n+1}] \)). Dehomogenizing \( G \) by letting \( x_{n+1} = 1 \) produces the tangent cone of \( h \) (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 \( h \subset k[x] \). The intersection multiplicity of \( p \) in \( V(h) \) is defined by \( \text{im}(p; h) := \dim_{\text{vec}}(O/\langle h \rangle) \) where \( O := \{ f/g : f, g \in k[x], g(p) \neq 0 \} \) is the localization ring of \( k[x] \) at \( p \) and \( \dim_{\text{vec}}(O/\langle h \rangle) \) is the dimension of \( O/\langle h \rangle \) as a vector space over \( k \). Note by [30] Chapter 4.2, Proposition 11 we may substitute the power series ring \( k[[x - p]] \) for \( O \).

**Example 20.** Let \( x = [x, y, z] \) and \( h = \{ x, y - z^2, z^2(z^4 + 1) \} \). We have:

\[
k[[x]]/\langle h \rangle = k[[x]]/\langle x, y - z^3, z^2 \rangle = k[[x]]/\langle x, y, z^2 \rangle = \{ a + bz : a, b \in k \}
\]

implying \( \text{im}(0; h) = 2 \).

### 7.2.2 Regular chains

Broadly speaking, a *regular chain* of \( k[x] \) is a system of equations and inequations defined by polynomials in \( k[x] \) 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 \( h \subset k[x] \) be a non-constant polynomial. The *main variable* of \( h \) is the largest variable \( x \in x \) (for the ordering \( x_0 > \cdots > x_n \) 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 \( z^x + t \) is \( x \) in \( Q[x > y > z > t] \) and 1 in \( Q[t > z > y > x] \).

Let \( T \subset k[x] \) 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 \( \text{sat}(T) = \langle T \rangle : I_T^\infty \) and we call *quasi-component* of \( T \) the basic constructible set \( W(T) := V(T) \setminus V(I_T) \).

**Definition 23 (Regular Chain).** The triangular set \( T \subset k[x] \) 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 \( T_0, \ldots, T_r \subset k[x] \) form a Kalkbrener decomposition of \( \sqrt{\langle h \rangle} \) (resp. a Wu-Lazard decomposition of \( V(h) \)) whenever we have \( \sqrt{\langle h \rangle} = \sqrt{\text{sat}(T_0)} \cap \cdots \cap \sqrt{\text{sat}(T_r)} \)
\[ \sqrt{\text{sat}(T)} \] (resp. \( V(h) = W(T_0) \cup \cdots \cup W(T_e) \)). 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 \( W(T) \) (that is, the intersection of all algebraic sets containing \( W(T) \)) 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 \( W(T) \) 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 \( f \in k[x] \) is regular or not modulo the saturated ideal of a regular chain \( T \subset k[x] \). This allows us to specify an operation, called \texttt{Regularize}, as follows. The function call \texttt{Regularize} \((f, T)\) computes regular chains \( T_0, \ldots, T_e \subset k[x] \) 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 \texttt{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 \( h = h_0, \ldots, h_{n-1}, h_n \in k[x] \) 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 \( \pi \) 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 \( h \) to a lower dimensional subspace while leaving the intersection multiplicity of \( V(h) \) 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 \( \mathbb{A}^{n+1}(\overline{k}) \) 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 \( k \) 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 1.10 of Mumford’s book \([70]\)):

For an irreducible algebraic set \( V \) and a subset \( U \subseteq V \) open in the Zariski topology induced on \( V \), the closure of \( U \) in Zariski topology and the closure of \( U \) in the Euclidean topology are both equal to \( V \). It follows that, for a regular chain \( T \subset \mathbb{C}[x] \) the closure of \( W(T) \) in Zariski topology and the closure of \( W(T) \) in the Euclidean topology are equal, thus both equal to \( V(\text{sat}(T)) \). 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 \( T \subset \mathbb{C}[x] \), compute the (non-trivial) limit points of the quasi-component of \( T \), that is, the set \( \lim(W(T)) := \overline{W(T)} \setminus W(T) \).

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 \( \lim(W(T)) \) 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 \( h = \{h_0, \ldots, h_{n-1}\} \subset \mathbb{C}[x] \) be \( n \) polynomials such that \( C = V(h) \) 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 \( \{q_k : k \in \mathbb{N}_2\} \) 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 \( h \in h \), 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 \( D \) through \( p \) of \( C = V(h) \) 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 \( q \in (B \cap D) \setminus \{p\} \). Observe also that the singular locus \( \text{sing}(D) \) contains a finite number of points. It follows that we can take \( B \) small enough so that \( B \cap \text{sing}(D) \) is either empty or \( \{p\} \). Define

\[
\nu(q) := T_q(h_0) \cap \cdots \cap T_q(h_{n-1}),
\]

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 \( \nu(q) \) as \( q \) approaches \( p \). Since \( TC_p(C) \) is the union of all the \( TC_p(D) \), this yields a method for computing \( TC_p(C) \).
Lemma 32. The collection of limits of lines $v(q)$ as $q$ approaches $p$ in $(B \cap D) \setminus \{p\}$ gives the tangent cone of $D$ at $q$. That is to say

$$TC_p(D) = \lim_{q \to p} v(q) = \lim_{q \to p} T_q(h_0) \cap \cdots \cap T_q(h_{n-1}).$$

Proof. There are two cases, either

1. $D$ is smooth at $p$ and $B \cap \text{sing}(D) = \emptyset$, or

2. $D$ is singular at $p$ and $B \cap \text{sing}(D) = \{p\}$.

Case 1. Assume $q \in B \cap D$ is arbitrary and observe $D$ is smooth within $B$ and thereby the tangent cone of $D$ is simply the tangent space (i.e. $TC_q(D) = T_q(D)$).

Notice $T_q(D)$ is a sub-vector space of $v(q)$. Indeed, let $w \in T_q(D)$ be any tangent vector to $D$ at $q$. As $D$ is a curve in each $V(h)$ for $h \in h$ it follows $w$ is a vector tangent to each $V(h)$ as well. Correspondingly $w \in T_q(h)$ for any $h \in h$ 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 $h \in h$, the vector $w$ must span this subspace. Thus, for each $q \in B \cap D$, we have

$$T_q(D) = T_q(h_0) \cap \cdots \cap T_q(h_{n-1}).$$

Taking the limit of each side of the above equality, when $q$ approaches $p$ and using again the fact that $D$ is smooth at $q = p$, we obtain the desired result, that is, $TC_p(D) = \lim_{q \to p} v(q)$.

Case 2. Assume $D \cap B \setminus \{p\}$ is a finite union of smooth curves $D_0, \ldots, D_j$. These are the smooth branches of $D \cap B$ meeting at the singular point $p$. Each $j$ corresponds to a unique line

$$L_j = \lim_{q \to p} v(q) \subset T_p(D)$$

as $q$ approaches $p$ along $D_j$.

By Lemma [31] the tangent cone $TC_p(D)$ is the collection of limits to $p$ of secant lines through $p$ in $D$. Such lines given by secants along $D_j$ must coincide with $L_j$. More precisely

$$L_0 \cup \cdots \cup L_j \subset TC_p(D).$$

Because each $D_j$ is smooth there is only one secant line for each $j$ and thereby

$$L_0 \cup \cdots \cup L_j = TC_p(D)$$

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 \(C = V(h)\) with coordinates \(x\). Further let \(\overrightarrow{pq}\) be a unit vector in the direction of \(\overrightarrow{pq}\) (i.e. the line through \(p\) and \(q\)). To exploit Lemma \(^{31}\) we must calculate the set

\[
\left\{ \lim_{q \to p} \frac{\overrightarrow{pq}}{q \neq p} \right\},
\]

which is indeed a set because \(C\) may have several branches through \(p\) yielding several lines in the tangent cone \(TC_p(C)\).

Let \(T \subset \mathbb{C}[y][x]\) be a zero-dimensional regular chain encoding the point \(p\), that is, such that we have \(V(T) = \{p\}\). Note that the introduction of \(y\) for the coordinates of \(p\) is necessary because the “moving point” \(q\) is already using \(x\) for its own coordinates. Consider the polynomial set

\[s = T \cup h,\]

and observe that the ideal \(\langle s \rangle\) is one-dimensional in the polynomial ring \(\mathbb{C}[x_{n-1} > \cdots > x_0 > y_{n-1} > \cdots > y_0]\). Let \(T_0, \ldots, T_e \subset \mathbb{C}[y][x]\) be one-dimensional regular chains forming a Kalkbrener decomposition of \(\sqrt{s}\). Thus we have

\[V(s) = W(T_0) \cup \cdots \cup W(T_e).\]

Computing with the normal vector \(\overrightarrow{pq}\) is unnecessary and instead we divide the vector \(\overrightarrow{pq}\) by \(x_n - y_n\). Since the \(n\)-th coordinate of \(\overrightarrow{pq}_{x_n - y_n}\) 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 \(V(s)\). 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 \(x\), 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 these 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 \(T \subset \mathbb{C}[y][x]\) 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 \( V(s) \) 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 \texttt{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 \texttt{Regularize}, we may assume w.l.o.g. that either \( x_n - y_n \) is regular modulo \( \text{sat}(T_j) \) or that \( x_n - y_n \equiv 0 \mod \text{sat}(T_j) \) holds.

Consider the latter case first. If \( x_n - y_n \equiv 0 \mod \text{sat}(T_j) \) then \( W(T_j) \subseteq V(x_n - y_n) \) 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 \( x_d - y_d \not\equiv 0 \mod \text{sat}(T_j) \) necessarily holds. Indeed, if \( x_i - y_i \equiv 0 \mod \text{sat}(T_j) \) would hold for all \( i \in [0, n] \) then \( W(T_j) \subset V(x_0 - y_0) \cap \cdots \cap V(x_n - y_n) \) would hold as well. Since the \( y \) coordinates are fixed by \( T \), the algebraic set \( W(T_j) \) would be zero-dimensional—a contradiction.

Hence, up to a variable renaming, we can assume that \( x_n - y_n \) is regular modulo \( \text{sat}(T_j) \). Therefore, the algebraic set \( V(x_n - y_n) \cap W(T_j) \) 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 \( m = m_0, \ldots, m_n \) as new variables, that we call \textit{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” \( m \) when \( x_n \) approaches \( y_n \) with \( (y_0, \ldots, y_n, x_0, \ldots, x_n) \in W(T_j) \).

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 \( M_j \subset \mathbb{C}[m][y][x] \) given by
\[
M_j = T_j \cup \left\{ m_0(x_n - y_n) - (x_0 - y_0) \right\} \cdots \left\{ m_n(x_n - y_n) - (x_n - y_n) \right\}.
\]
Note that \( M_j \) is one-dimensional in this extended space and computing \( \lim(W(M_j)) \), using the algorithm of [6], solves for \( m \) when \( x_n \to y_n \) with \( (x, y) \in W(T_0) \). Therefore and finally, the desired set \( \{\lim_{q \to p, q \neq p} \overline{pq}\} \) 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 \texttt{LimitPoints}, is the function for computing \( \lim(W(M_j)) \) which is described in Section 7.4.1 Algorithm [5].

\textbf{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 \( m_n > \cdots > m_0 > x_n > \cdots > x_0 > y_n \cdots > y_0 \). Performing a change of variable ordering to \( x > m > y \) 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 \( T \subset k[x_0, \ldots, x_n] \) representing the point \( p \) on the curve \( C \); the set \( h \) containing \( n \) polynomials and representing the curve \( C \).

Output: The tangent cone of the curve \( C \) at the point \( p \).

begin
1. Consider new independent variables \( y_0, \ldots, y_n \);
2. Substitute the variables \( x_0, \ldots, x_n \) in \( T \) with \( y_0, \ldots, y_n \), respectively;
3. \( \{T_0, \ldots, T_e\} := \text{Triangularize}(h \cup T, R) \), where \( R \) is a variable ordering defined as \( R := x_n > \cdots > x_0 > y_n > \cdots > y_0 \);
4. \( lm := \{\} \);
5. for each \( T_i \) do
   6. for \( \ell \) from 0 to \( n \) do
   7. if \( x_\ell - y_\ell \) is regular modulo \( T_i \), then
   8. Let \( m_j := \frac{x_j - y_j}{x_\ell - y_\ell} \), for \( j = 0, \ldots, n \);
   9. Extend \( T_i \) with \( (x_\ell - y_\ell) m_j - (x_j - y_j) \) to a regular chain \( M_{i,\ell} \);
   10. \( lm := lm \cup \text{LimitPoints}(M_{i,\ell}, R') \), where \( R' \) is a variable ordering defined as \( R' := m_n > \cdots > m_0 > x_n > \cdots > x_0 > y_n > \cdots > y_0 \);
11. Remove redundant vectors such that all direction vectors in \( lm \) are pairwise colinear;
12. return \((lm)\)
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 \( W(T_j) \)) if and only if the vectors

\[
\begin{pmatrix}
\bar{p}q \\
x_n - y_n
\end{pmatrix}
= \begin{pmatrix}
1 \\
m_{n-1} \\
\vdots \\
m_0
\end{pmatrix} \quad \text{and} \quad \bar{p}S = \begin{pmatrix}
X_n - y_n \\
X_{n-1} - y_{n-1} \\
\vdots \\
X_0 - y_0
\end{pmatrix}
\]

are collinear. That is, if and only if we have the following relations

\[
\begin{align*}
X_n &= m_n(x_n - y_n) + y_n \\
& \vdots \\
X_0 &= m_0(x_n - y_n) + y_0.
\end{align*}
\] (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 \( X_n > \cdots > X_0 > m_{n-1} > \cdots > m_0 > y_n \cdots > y_0 \). 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 $\frac{x_i - y_i}{x_0 - y_0}$ for $m_i$ into the equations defining $m_0, \ldots, m_n$.

### 7.4.4 Examples

The following examples illustrate 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 set $h := \{x_1^2 - x_0^2(x_0 + 1)\}$ 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 $T_1 := \{x_1^2 - x_0^2(x_0 + 1), y_1, y_0\}$ which is a Kalkbrener decomposition of the set $h \cup T'$ w.r.t variable ordering $x_1 > x_0 > y_1 > y_0$ (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 $M_{1,0} := \{m_0 - 1, x_1^2 - x_0^2(x_0 + 1), y_1, y_0\}$ w.r.t variable ordering $m_1 > m_0 > x_1 > x_0 > y_1 > y_0$ (Line 10 of Algorithm 16).
5. Then by applying LimitPoints command on $M_{1,0}$ we obtain the following regular chains:

   $$lm_1 := \begin{cases} m_1 + 1 \\ m_0 - 1 \\ x_1 \\ x_0 \\ y_1 \\ y_0 \end{cases} \quad lm_2 := \begin{cases} m_1 - 1 \\ m_0 - 1 \\ x_1 \\ x_0 \\ y_1 \\ y_0 \end{cases}$$

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 $h$ 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 $h := \{x_2^2 + x_1^2 + x_0^2 + 1, x_2^2 - x_1^2 - x_0\} \subset k[x_2, x_1, x_0]$. 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 $h$ at the point $p$. Based on Algorithm 16 we have:
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 \(h \cup T'\) w.r.t variable ordering \(x_2 > x_1 > x_0 > y_2 > y_1 > y_0\) (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 \(m_2 > m_1 > m_0 > x_2 > x_1 > x_0 > y_2 > y_1 > y_0\) (Line 10 of Algorithm 16).

- Then by applying LimitPoints command on \(M_{1,0}\) we obtain the following regular chains:
After removing the redundancies, we obtain the following slopes:

\[
\begin{align*}
lm_1 &:= \begin{cases} 
2m_2 + y_1 \\
2m_1 + y_1 \\
m_0 - 1 \\
x_2 + y_1 \\
x_1 + y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_2 := \begin{cases} 
m_2 + 1 \\
m_1 - 1 \\
m_0 \\
x_2 - y_1 \\
x_1 + y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_3 := \begin{cases} 
m_2 - 1 \\
m_1 - 1 \\
m_0 + 4y_1 \\
x_2 + y_1 \\
x_1 - y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_4 := \begin{cases} 
m_2 - 1 \\
m_1 - 1 \\
m_0 \\
x_2 + y_1 \\
x_1 + y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_5 := \begin{cases} 
m_2 + 1 \\
m_1 - 1 \\
m_0 \\
x_2 + y_1 \\
x_1 + y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_6 := \begin{cases} 
m_2 - 1 \\
m_1 - 1 \\
m_0 + 4y_1 \\
x_2 + y_1 \\
x_1 - y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_7 := \begin{cases} 
m_2 - 1 \\
m_1 - 1 \\
m_0 \\
x_2 - y_1 \\
x_1 - y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_8 := \begin{cases} 
m_2 - 1 \\
m_1 - 1 \\
m_0 \\
x_2 + y_1 \\
x_1 + y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases} \\
lm_9 := \begin{cases} 
m_2 - 1 \\
m_1 - 1 \\
m_0 \\
x_2 - y_1 \\
x_1 + y_1 \\
x_0 \\
y_2 + y_1 \\
2y_1^2 - 1 \\
y_0
\end{cases}
\end{align*}
\]

- By applying the same process for \(x_1 - y_1\), we obtain:

In fact, in the computations above, there are some redundancies for computing the slopes. After removing the redundancies, we obtain the following slopes:

\[
\begin{align*}
s_1 &:= \begin{cases} 
m_2 + 1 = 0 \\
m_1 - 1 = 0 \\
m_0 = 0
\end{cases} \\
s_2 &:= \begin{cases} 
m_2 - 1 = 0 \\
m_1 + 1 = 0 \\
m_0 = 0
\end{cases} \\
s_3 &:= \begin{cases} 
2m_2x_1 = 0 \\
2m_1 + x_1 = 0 \\
m_0 - 1 = 0
\end{cases} \\
s_4 &:= \begin{cases} 
m_2 - 1 = 0 \\
m_1 - 1 = 0 \\
m_0 - 4x_1 = 0
\end{cases}
\end{align*}
\]

where \(x_1\) corresponds to the \(x_1\) coordinate of the solutions of the regular chain \(T\) which is \(\left\{ \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}} \right\}\).
Furthermore, based on Remark \[17\] one can obtain the equations of the lines contained in the tangent cone of \( h \) as following:

\[
\begin{align*}
X_0 &= 0 \\
X_2 + X_1 &= 0 \\
4X_1X_2 + X_0 - 2 &= 0 \\
X_2 - X_1 + 2X_1 &= 0
\end{align*}
\]

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 \( h \) 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 \( h \).

![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 \( h = \{x_2^2 + x_1^2 + x_0^2 - 1, x_2^2 - x_1^2 - x_0(x_0 - 1)\} \subset k[x_2, x_1, x_0] \). Consider the point \( (0, 0, 1) \) on this curve. Then using \texttt{TangentCone} command of \texttt{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 \texttt{slopes} to the command \texttt{TangentCone}. These slopes are given by the systems below:

\[
\begin{align*}
3 m_1^2 - 1 &= 0 \\
m_1 - 1 &= 0 \\
m_0 &= 0 \\
m_2 - 1 &= 0 \\
m_1^2 - 1 &= 0 \\
m_0 &= 0
\end{align*}
\]

The command \texttt{TangentCone} also returns the equations of the lines contained in the tangent cone of the curve \( C \), by default, as following:

\[
\begin{align*}
X_0 - 1 &= 0 \\
X_1^2 - 3X_2^2 &= 0
\end{align*}
\]

### 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 first assume 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 in 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 \(q(x, y, z, w) = \frac{z^2 w + x^2 y^2}{x^2 y^2 + z^2 + w^2}\). 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 \(\lim_{(x, y, z, w) \to (0, 0, 0, 0)} \chi(q)\). The set \(\chi(q)\), defined in Notation 4, is obtained by the method of Lagrange multipliers, see Section 8.2.7 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):

\[
\begin{cases}
  x = 0 \\
  y = 0 \\
  z = 0 \\
  w = 0
\end{cases}
\quad \text{and} \quad
\begin{cases}
  x = 0 \\
  y = 0 \\
  z + w = 0 \\
  z - w = 0
\end{cases}
\]

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.
the following Jacobian matrix

\[
\begin{bmatrix}
\frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} & \frac{\partial z}{\partial t} & \frac{\partial w}{\partial t} \\
\frac{\partial x}{\partial y} & \frac{\partial y}{\partial y} & \frac{\partial z}{\partial y} & \frac{\partial w}{\partial y} \\
\frac{\partial x}{\partial z} & \frac{\partial y}{\partial z} & \frac{\partial z}{\partial z} & \frac{\partial w}{\partial z} \\
\frac{\partial x}{\partial w} & \frac{\partial y}{\partial w} & \frac{\partial z}{\partial w} & \frac{\partial w}{\partial w}
\end{bmatrix}
= \begin{bmatrix} x & y & z & w \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

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 \texttt{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

\[ A_1 x \quad A_2 y \quad A_3 z \quad A_4 w \]

is full rank. We are ready to apply the method of Lagrange multipliers, considering the following matrix

\[
\begin{bmatrix}
A_1 x & A_2 y & A_3 z & A_4 w \\
x & y & z & w \\
\frac{\partial A_1}{\partial x} & \frac{\partial A_2}{\partial y} & \frac{\partial A_3}{\partial z} & \frac{\partial A_4}{\partial w}
\end{bmatrix}
= \begin{bmatrix} 3x & 1y & 2z & 3w \\ x & y & z & w \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

which has a single non-zero minor, namely \( m = 4xy \). Using again the incremental version of \texttt{RealTriangularize}, we compute \( Z_\mathbb{R}(R_1) \cap Z_\mathbb{R}(m = 0) \) 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 \( Z_\mathbb{R}(R_1), Z_\mathbb{R}(R_4) \) 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 \( q(x, y, z) = \frac{x^2y^2}{x^4+y^4} \). Here again, we aim at computing \( \lim_{(x,y,z) \to (0,0,0)} q \). \texttt{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:

\[
\begin{cases}
\begin{aligned}
x &= 0, \\
y &= 0, \\
z &= 0
\end{aligned}
\end{cases}
\quad \text{and} \quad
\begin{cases}
\begin{aligned}
x &= 0, \\
y &= 0, \\
z &= 0
\end{aligned}
\end{cases}
\]

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 $x(q)$. Hence, we first determine when the following matrix
\[
\begin{bmatrix}
\frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\
1 & 0 & 0
\end{bmatrix}
\]
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
\[
\begin{bmatrix}
A_1 x & A_2 y & A_3 z \\
x & y & z
\end{bmatrix}
\begin{bmatrix}
\frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\
1 & 0 & 0
\end{bmatrix}
\]
Its determinant is $8yz$ and we compute $Z_\mathbb{R}(R_5) \cap \{yz = 0\}$ yielding $Z_\mathbb{R}(R_6)$ with $R_6 := \{x = 0, z = 0, y \neq 0\}$. The regular semi-algebraic system $R_5$ 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 computing 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 $\Omega$ 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 $\Omega$. Define $\Sigma_h := \{ y \in \mathbb{R}^n : g_1(y) = b_1, \ldots, g_m(y) = b_m \}$. The point $x^*$ 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 $x^*$ such that $f(x)$ takes an extremal value (maximum or minimum) at $x = x^*$ on $\Sigma_h \cap U$. When this holds, the gradients $\nabla f(x^*), \nabla g_1(x^*), \ldots, \nabla g_m(x^*)$ are linearly dependent. The above theorem is, in fact, usually stated when $m < n$ holds and $\nabla g_1(x^*), \ldots, \nabla g_m(x^*)$ are linearly independent, i.e. when $g(x^*)$ is a regular value of the function $g = (g_1, \ldots, g_m)$ (at least if we restrict $g$ to some neighbourhood $V$ of $x^*$). Then, the necessary condition of the theorem can be translated into the identity
\begin{equation}
\nabla f(x^*) = \lambda_1^* \nabla g_1(x^*) + \ldots + \lambda_m^* \nabla g_m(x^*),
\end{equation}
for some real numbers $\lambda_1^*, \ldots, \lambda_m^*$ called Lagrange multipliers. Observe that, if we define the Lagrange function
\begin{equation}
F(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i (b_i - g_i(x)),
\end{equation}
then the conditions (8.1) and $x^* \in \Sigma_h$, for some neighbourhood $U$ of $x^*$ are equivalent to the fact that $(x^*, \lambda^*)$ 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 $\text{mvar}(p)$, $\text{init}(p)$, and $\text{mdeg}(p)$ respectively the greatest variable appearing in $p$ (called the main variable of $p$), the leading coefficient of $p$ w.r.t. $\text{mvar}(p)$ (called the initial of $p$), and the degree of $p$ w.r.t. $\text{mvar}(p)$ (called the main degree of $p$); denote by $\text{discrim}(p)$ the discriminant of $p$ w.r.t. $\text{mvar}(p)$. 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 $\text{mvar}(T)$ the set of main variables of the polynomials in $T$. A variable $v \in X$ is called algebraic w.r.t. $T$ if $v \in \text{mvar}(T)$, 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 $v \in \text{mvar}(T)$, we denote by $T_v$ and $T_{\bar{v}}$ the polynomial $f \in T$ with $\text{mvar}(f) = v$ and the polynomials $f \in T$ with $\text{mvar}(f) \neq v$, respectively. Let $h_T$ be the product of the initials of the polynomials in $T$. We denote by $\text{sat}(T)$ the saturated ideal of $T$: if $T$ is the empty triangular set, then $\text{sat}(T)$ 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 $\overline{W(T)} = V(\text{sat}(T))$. For $f \in \mathbf{k}[X]$, we denote by $\text{res}(f, T)$ the iterated resultant of $f$ w.r.t. $T$, that is, $f$ itself, if $f$ is constant, or $\text{res}(\text{res}(f, T_v), T_{\bar{v}})$ if $v \in \text{mvar}(T)$ and $v = \text{mvar}(f)$ hold, or $\text{res}(f, T_{\bar{v}})$ otherwise.
Regular chain. A triangular set $T \subset k[X]$ 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 $\text{sat}(T_v)$. Let $H \subset k[X]$. The pair $[T, H]$ is a regular system if each polynomial in $H$ is regular modulo $\text{sat}(T)$. 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 $\partial t$ is regular w.r.t. $\text{sat}(T)$, where $\partial t = \frac{\partial}{\partial v}$ and $v = \text{mvar}(t)$. By $[T, H_v]$, 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 $F \subset k[X]$. Regular chains $T_1, \ldots, T_e$ of $k[X]$ form a triangular decomposition of $V(F)$ in the sense of Kalkbrener (resp. Wu and Lazard) whenever we have $V(F) = \cup_{i=1}^e W(T_i)$ (resp. $V(F) = \cup_{i=1}^e W(T_i)$). We denote by Triangulize an algorithm, such as the one of [25], computing a Kalkbrener triangular decomposition.

Regularization. Let $p \in k[X]$ and $T \subset k[X]$ 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 $p \in \text{sat}(T_i)$ holds or $p$ is regular w.r.t. $\text{sat}(T_i)$; (2) we have $W(T) = W(T_1) \cup \cdots \cup W(T_e)$, and $\text{mvar}(T) = \text{mvar}(T_i)$ holds for each $i = 1, \ldots, e$.

Good specialization. Let $[T, H]$ be a square-free regular system of $k[X]$. Recall that $Y$ and $U = U_1, \ldots, U_d$ stand respectively for $\text{mvar}(T)$ and $X \setminus Y$. Let $a = (a_1, \ldots, a_d)$ be a point of $\overline{k}^d$. We say that $[T, H]$ specializes well at $a$ if: (i) for each $t \in T$ the polynomial $\text{init}(t)$ 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 $k[X]$. Let $b p$ be the primitive and square free part of the product of all $\text{res}(\partial t, T)$ $\text{res}(h, T)$ for $h \in H$ and $t \in T$. We call $b p$ 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 $a \in \overline{k}^d$ if and only if the border polynomial $b p(a) \neq 0$.

8.2.3 Parametric polynomial systems

The following is based on [55] and [68]. In the sequel of this section, the field $k$ is either $\mathbb{R}$ or $\mathbb{C}$. Let $f_1, \ldots, f_r, p_1, \ldots, p_r \in \mathbb{Q}[X]$, with, as before $X = X_1 < \cdots < X_n$. Consider the constructible set $C = \{x \in \mathbb{C}^d : f_i(x) = \cdots = f_r(x) = 0, p_1(x) \neq 0, \ldots, p_r(x) \neq 0\}$ and the semi-algebraic set $S = \{x \in \mathbb{R}^n : f_i(x) = \cdots = f_r(x) = 0, p_1(x) > 0, \ldots, p_r(x) > 0\}$. Let $1 \leq d < n$. We view the variables $X_1, \ldots, X_d$ as parameters and we rename them as $\overline{U} = U_1, \ldots, U_d$. We denote by $\Pi_{\overline{U}}$ the canonical projection on the parameter space.

Discriminant variety in the complex case. Let $\delta$ be the dimension of $\Pi_{\overline{U}}(C) = \overline{\Pi_{\overline{U}}(C)}$. An algebraic set $W \subset \mathbb{C}^d$ is a discriminant variety of $C$ w.r.t. $\Pi_{\overline{U}}$ if the following four conditions hold:

1. $W \subseteq \Pi_{\overline{U}}(C)$ holds,
2. $W = \overline{\Pi_{\overline{U}}(C)}$ holds if and only if $\Pi_{\overline{U}}^{-1}(u) \cap C$ is infinite for almost all $u \in \overline{\Pi_{\overline{U}}(C)}$.
3. the connected components $\mathcal{U}_1, \ldots, \mathcal{U}_k$ of $\Pi_U(C) \setminus W$ are analytic sub-manifolds of dimension $\delta$, and
4. for all $1 \leq i \leq k$, the pair $(\Pi_U^{-1}(\mathcal{U}_i), \Pi_U)$ is an analytic covering of $\mathcal{U}_i$.

This latter condition implies that there exists finitely many disjoint connected subsets $C_1, \ldots, C_k$ of $\mathbb{C}^n$ such that their union equals $\Pi_U^{-1}(\mathcal{U}_i) \cap C$ and $\Pi_U$ is a local diffeomorphism from $C_j$ onto $\mathcal{U}_i$, for $1 \leq j \leq k_i$ and $1 \leq i \leq k$. Moreover, $W$ contains the union of the critical values of the restriction of $\Pi_U$ to the regular locus of $C$, as well as the projection of the singular locus of $C$.

**Proposition 21** (Theorem 4, [63]). Let $[T, H]$ be a square-free regular system of $k[X]$ and $bp$ its border polynomial. Then, the zero set of $bp$ in $k$ is the $\in$-minimal discriminant variety of $[T, H]$ 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 $S$ 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 $W \neq \Pi_U(C)$ holds. Then, $(\Pi_U(C) \setminus W) \cap \mathbb{R}^d$ 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 $S$ over $\mathcal{U}_i$ is constant, and if $\Pi_U^{-1}(\mathcal{U}_i) \cap S$ is not empty, then $(\Pi_U^{-1}(\mathcal{U}_i) \cap S, \Pi_U)$ 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 $k$ of our polynomial coefficients is $\mathbb{Q}$ instead of $\mathbb{R}$. See [22] for details. Nevertheless, one can easily reduce the case where $k$ is a real algebraic extension of $\mathbb{Q}$ to the case $k = \mathbb{Q}$ 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 $H = \{h_1, \ldots, h_t\}$ 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 $P_\geq$ denote the set of the inequalities $\{p_1 > 0, \ldots, p_r > 0\}$. Let $H_\geq$ denote the set of inequalities $\{h_1 \neq 0, \ldots, h_t \neq 0\}$. We will denote by $[F, P_\geq, H_\geq]$ the basic semi-algebraic system $\{f_1 = 0, \ldots, f_s = 0, p_1 > 0, \ldots, p_r > 0\}$ and by $S := [F, N_\geq, P_\geq, H_\geq]$ 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 $h_1 \neq 0, \ldots, h_t \neq 0$. 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 $W \subset \mathbb{C}^n$, 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 $U = U_1, \ldots, U_d$ and $Y = Y_1, \ldots, Y_{n-d}$ 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 \( U \) variables. Let \( O \) be the semi-algebraic subset of \( \mathbb{R}^d \) defined by \( Q \). We say that \( R := \{ Q, T, P, H \} \) is a regular semi-algebraic system if either \( d = 0 \) and the semi-algebraic system \( \{ T, P, H \} \) 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 \( \{ T(a), P(a), H \} \) 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 \( R = \{ Q, T, P, H \} \) 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 \( O \), \( (\Pi_U^{-1}(C) \cap Z_{\mathbb{R}}(R), \Pi_U) \) 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 \( S := \{ F, N, P, H \} \) 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 \textsf{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 \textsf{RealTriangularize} algorithm is the fact that triangular decompositions of semi-algebraic sets can be computed incrementally. Indeed, this algorithm relies on a procedure, called \textsf{Intersect}, such that, for a given semi-algebraic constraint (that is, either a polynomial equation, or a polynomial inequality) \( C \), the function call \textsf{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 \( o \) 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 \( o \) 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 $k$ be an algebraic number field and $\overline{k}$ its algebraic closure. We denote by $k[[X_1,\ldots,X_n]]$ and $k\langle X_1,\ldots,X_n \rangle$ the respective rings of formal power series and convergent power series in $X_1,\ldots,X_n$ with coefficients in $k$. When $n = 1$, we write $U$ instead of $X_1$. Thus $k[[U]]$ and $k\langle U \rangle$ are the rings of formal and convergent univariate power series in $U$ and coefficients in $k$.

Puiseux series. We denote by $k[[U^\ast]] = \bigcup_{n=1}^{\infty} k[[U^{1/n}]]$ the ring of formal Puiseux series. Hence, given $\varphi \in k[[U^\ast]]$, there exists $\ell \in \mathbb{N}_{>0}$ such that $\varphi \in k[[U^{1/\ell}]]$ holds. Hence, we can write $\varphi = \sum_{m=0}^{\infty} a_m U^{\ell m}$, for some $a_0,\ldots,a_m,\ldots \in k$. We denote by $k((U^\ast))$ the quotient field of $k[[U^\ast]]$. Let $\varphi \in k[[U^\ast]]$ and $\ell \in \mathbb{N}$ such that $\varphi = f(U^{1/\ell})$ holds for some $f \in k[[U]]$. We say that the Puiseux series $\varphi$ is convergent if we have $f \in k(U)$. Convergent Puiseux series form an integral domain denoted by $k(U^\ast)$; its quotient field is denoted by $k((U^\ast))$.

Puiseux parameterization. Let $f \in k(X_1)[X_2]$ be of positive degree $d$ in $X_2$. A Puiseux parametrization of $f$ is a pair $(\psi(U), \varphi(U))$ of elements of $\overline{k}(U)$ for some new variable $U$, such that

1. $\psi(U) = U^\zeta$, for some $\zeta \in \mathbb{N}_{>0}$;
2. $f(\psi(U), \varphi(U)) = 0$ holds in $\overline{k}(U)$, and
3. there is no integer $\ell > 1$ such that both $\psi(U)$ and $\varphi(U)$ are in $\overline{k}(U^\ell)$.

The index $\zeta$ is called the ramification index of the parametrization $(U^\zeta, \varphi(U))$. 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 $\varphi_1,\ldots,\varphi_d \in k\langle U \rangle$ 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 $D^*_\rho$ 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 $q : \mathbb{R}^n \rightarrow \mathbb{R}$ be a rational function defined on $D^*_\rho$.

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

$$\chi(q) = \{(x, y) \in \mathbb{R}^2 \mid y \frac{\partial q}{\partial x} - x \frac{\partial q}{\partial y} = 0\}.$$ 

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 $\overline{Z_{q}(S)}$ of $Z_{q}(S)$ 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 < \( \delta < \rho \) 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

\[
\lim_{(x_1, \ldots, x_n) \to (0, \ldots, 0)} q(x_1, \ldots, x_n) = L
\]

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) \( \lim_{(x_1, \ldots, x_n) \to (0, \ldots, 0)} q(x_1, \ldots, x_n) \) 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 \( C_r = \{(x_1, \ldots, x_n) \in \mathbb{R}^n | \sqrt{x_1^2 + \cdots + x_n^2} = r\} \). 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. \( \square \)

**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) \( \lim_{(x_1, \ldots, x_n) \to (0, \ldots, 0)} q(x_1, \ldots, x_n) \) 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 \( \lim_{(x_1, \ldots, x_n) \to (0, \ldots, 0)} q(x_1, \ldots, x_n) \) 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 \( \text{Sing}(S) \) 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 \( \lambda \) 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 \( \Delta \) of \( h \) in \( X_n \) vanish nowhere on \( U_0 \). Then, for every smooth function \( u : U_0 \to \mathbb{R} \) 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.
Thus, the set $W$ 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

$$n(t) = \frac{(-\partial u/\partial X_1, \ldots, -\partial u/\partial X_{n-1}, 1)}{\sqrt{\left(\partial u/\partial X_1\right)^2 + \cdots + \left(\partial u/\partial X_{n-1}\right)^2 + 1}}.$$ 

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:

$$\left\{ \begin{array}{c}
\frac{X_i}{\sqrt{X_1^2 + \cdots + X_{n-1}^2 + u^2(X_1, \ldots, X_{n-1})}} = -\frac{\partial u/\partial X_i}{\sqrt{\left(\partial u/\partial X_1\right)^2 + \cdots + \left(\partial u/\partial X_{n-1}\right)^2 + 1}}, \\
\frac{\partial u/\partial X_i}{\sqrt{\left(\partial u/\partial X_1\right)^2 + \cdots + \left(\partial u/\partial X_{n-1}\right)^2 + 1}} = \frac{X_1^2 + \cdots + X_{n-1}^2 + u^2(X_1, \ldots, X_{n-1})}{\sqrt{\left(\partial u/\partial X_1\right)^2 + \cdots + \left(\partial u/\partial X_{n-1}\right)^2 + 1}} \end{array} \right\}, \text{ for } i = 1, \ldots, n-1$$

The last equality in (8.3) implies that we have:

$$u(X_1, \ldots, X_{n-1}) = \frac{\sqrt{X_1^2 + \cdots + X_{n-1}^2 + u^2(X_1, \ldots, X_{n-1})}}{\sqrt{\left(\partial u/\partial X_1\right)^2 + \cdots + \left(\partial u/\partial X_{n-1}\right)^2 + 1}}$$

Consequently, we obtain the following system of PDEs:

$$\left\{ \begin{array}{c}
u(X_1, \ldots, X_{n-1}) \frac{\partial u}{\partial X_i} = -X_i, \text{ for } i = 1, \ldots, n-1. \end{array} \right.$$ 

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

$$\frac{u^2(X_1, \ldots, X_{n-1})}{2} = -\frac{X_1^2}{2} + F_2(X_2, \ldots, X_{n-1}).$$

Now by taking derivative of both sides of Equation (8.4) with respect to $X_2$, we have $u \frac{\partial u}{\partial X_2} = \frac{\partial F_2}{\partial X_2}$. After substitution of the latter equality in the equation $u \frac{\partial u}{\partial X_2} = -X_2$, 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

$$u^2(X_1, \ldots, X_{n-1}) = -\frac{X_1^2}{2} - \cdots - \frac{X_{n-1}^2}{2} + C.$$ 

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 $p \in D^*_p$ which is part of a sphere centred at the origin.

**Lemma 36.** Assume $n \geq 3$. Let $S = \{Q, \{t_u\}, P, \}$ be a regular semi-algebraic system of $Q[X_1, \ldots, X_n]$ 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 $\text{mvar}(t_u) = X_n$ holds. Let $M$ be the $2 \times n$ matrix with the vector $(X_1, \ldots, X_n)$ as first row and the gradient vector $\nabla t_u = \left( \frac{\partial t_u}{\partial X_1}, \ldots, \frac{\partial t_u}{\partial X_n} \right)$ as second row. Then, there exists a non-empty set $O \subset D^*_p \cap Z_{\mathbb{R}}(S)$, which is open relatively to $Z_{\mathbb{R}}(S)$, such that $M$ 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 $t_u$. Assume that there exists $0 < r < \rho$ such that $M$ is not full rank at any point of $D^*_p \cap Z_{\mathbb{R}}(S)$. Then, the system of partial differential equations (PDEs) $X_j \frac{\partial u}{\partial X_i} - X_i \frac{\partial u}{\partial X_j} = 0$, for $1 \leq i < j \leq n$ holds at any point of $D^*_p \cap Z_{\mathbb{R}}(S)$. Lemma 35 implies that $t_u = u(X_1^2 + \cdots + X_n^2)$ 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 $D^*_p \cap Z_{\mathbb{R}}(S)$ 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 $M$ 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 17, 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
1 Let \( f, g \) be the numerator and denominator of \( q \);
2 Let \( R := \{gX_{n+1} - f\} \cup T \) with \( X_{n+1} \) a new variable;
// \( R \) is a regular chain for \( X_1 < \cdots < X_{n+1} \);
3 Compute the limit points of \( Z_R(R) \setminus Z_R(h_R) \) in \( \mathbb{R}^n \) for the Euclidean topology;
4 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 \( q \);
5 Return no_limit since \( q \) has no limit along \( C \) at the origin;
7 end

Algorithm 18: RandomEllipse

Input: \( n \in \mathbb{N} \)

Output: Ellipse in \( \mathbb{R}^n \) randomly generated

begin
1 repeat
2 choose \( A_1, \ldots, A_n, r \) randomly with \( r > 0 \);
3 let \( E_r := \sum_{i=1}^{n} A_i X_i^2 - r^2 \);
4 \( S := \begin{bmatrix} \frac{\partial E_r}{\partial X_1} & \cdots & \frac{\partial E_r}{\partial X_n} \\ X_1 & \cdots & X_n \end{bmatrix} \);
5 until \( S \) has at least one non-zero minor ;
7 return \((A_1, \ldots, A_n, r)\);
8 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 \( Z_R(S_1), \ldots, Z_R(S_e) \) 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 \( Z_R(S_i) \) 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 \( D \) of \( \chi(q) \) as defined in Proposition 23. Following Lemma 34, each regular semi-algebraic system \( S \in D \) which is zero-dimensional, or such that the origin is not contained in the closure of \( Z_R(S) \), 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 $Z_E(S)$ has dimension one or not. If $\dim(Z_E(S)) = 1$ holds, one runs 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 $m \in \text{Minors}(\mathcal{M})$ (where $\mathcal{M}$ is defined at Line (6) of Algorithm [19]) such that we have $Z_E(S) \not\subseteq Z_E(m)$. Note that we know that the Jacobian of $T$ (that is, the matrix formed with the $\nabla t$, 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 $m \in \text{Minors}(\mathcal{M})$ is found, we compute $Z_E(S) \cap Z_E(m \neq 0)$ using the 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 $Z_E(S')$ 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 $Z_E(S) \cap Z_E(m' = 0)$ is less than that of $Z_E(S)$. 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 $(x_1, \ldots, x_n) \in Z_E(S) \cap \{E_r = 0\}$ 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 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 $m \in \text{Minors}(\mathcal{M})$ and the first minor $m' \in \text{Minors}(\mathcal{M}')$ 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 $I$ 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 (x1, ..., xn) ∈ Zq(S) ∩ {Er = 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 (L1, S1), ..., (Ln, Sn) with the desired properties. □

Algorithm 19: LimitInner

Input: q ∈ Q(X1, ..., Xn); 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(Zq(S)) = 1 then
3: return (LimitAlongCurve(q, S), S);
4: end if
5: let [Q, T, Pz] := S;
6: M := \begin{bmatrix} X_1 & \cdots & X_n \\ \nabla t, t \in T \end{bmatrix};
7: state := search_first_minor;
8: for m ∈ Minors(M) do
9: if Zq(S) ⊆ Zq(m) then next;
10: end if
11: J := ∅;
12: for S' ∈ Intersect(S, {m ≠ 0}) do
13: if q ⊈ Zq(S') or dim(Zq(S')) = 0 then
14: next;
15: end if
16: (A1, ..., An, r) := RandomEllipse(n);
17: let E_r := \sum_{i=1}^{n} A_i X_i^2 - r^2;
18: M' := \begin{bmatrix} X_1 & \cdots & X_n \\ \nabla t, t \in T \end{bmatrix};
19: state := search_second_minor;
20: for m' ∈ Minors(M') do
21: if res(m', T') = 0 then
22: next;
23: end if
24: J := ∅;
25: for C ∈ Intersect(S, m' = 0) do
26: if q ⊈ Zq(C) or dim(Zq(C)) = 0 then
27: next;
28: end if
29: end if
30: end for
31: if L = backtrack then
32: state := backtrack;
33: break;
34: else
35: I := I \cup \{L\};
36: end if
37: end for
38: end if
39: if I ≠ ∅ and state ≠ backtrack then
40: J := J \cup I;
41: state := found_second_minor;
42: break;
43: end if
44: if state ≠ found_second_minor then
45: state := backtrack;
46: break;
47: end if
48: next;
49: end for
50: if state = found_first_minor then
51: return J;
52: end if
53: end if
54: end procedure
Algorithm 20: Limit

\[ q \in \mathbb{Q}(X_1, \ldots, X_n). \]

\[ \text{Output: Limit of rational function } q \text{ at origin.} \]

begin
\[ A := \text{Minors} \left( \begin{array}{ccc} X_1 & \cdots & X_n \\ \frac{\partial q}{\partial X_1} & \cdots & \frac{\partial q}{\partial X_n} \end{array} \right); \]
\[ D := \text{RealTriangularize}(A); \]
\[ L := \emptyset; \]
\[ \text{for } S \in D \text{ do} \]
\[ \text{if } \hat{S} \in \mathbb{Z}_R(S) \text{ and dim}(\mathbb{Z}_R(S)) > 0 \text{ then} \]
\[ L := L \cup \{ \text{LimitInner}(q, S) \}; \]
\[ \text{return } L; \]
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 \( \lim_{x \to o} f/g \) 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 \( \lim_{x \to o} f/g \) 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

\[
\lim_{(x_1, \ldots, x_n) \to (0, \ldots, 0)} \frac{x_1^{a_1} \cdots x_n^{a_n}}{c_1 x_1^{2m_1} + \cdots + c_n x_n^{2m_n}}
\]

exists and equals zero if and only if \( \sum_{i=1}^{N} \frac{a_i}{2m_i} > 1 \).

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 $\lim_{(x_1, \ldots, x_n)\rightarrow(0, \ldots, 0)} q = L$.

Theorem 18 gives a sufficient condition for $\lim_{(x_1, \ldots, x_n)\rightarrow(0, \ldots, 0)} \frac{f(x_1, \ldots, x_n)}{g(x_1, \ldots, x_n)}$ 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 $\lim_{(x_1, \ldots, x_n)\rightarrow(0, \ldots, 0)} \frac{f(x_1, \ldots, x_n)}{g(x_1, \ldots, x_n)}$ 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 $q := \frac{f}{g}$ 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 $q := (0, \ldots, 0) \in Z_\mathbb{R}(f) \cap Z_\mathbb{R}(g)$ holds. Let $C_{f, \mathbb{R}}$ and $C_{g, \mathbb{R}}$ be the connected components of $Z_\mathbb{R}(f)$ and $Z_\mathbb{R}(g)$ to which $q$ belongs. Assume $\dim(C_{g, \mathbb{R}}) > 0$.

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 equality $C_{g, \mathbb{R}} = C_{f, \mathbb{R}}$ holds for them.

**Example 27.** Consider $f := (X - Y)^2 + (Z - Y)^2$ and $g := (X - Y)^2 + (Z - X)^2$. Then $C_{g, \mathbb{R}} \cap C_{f, \mathbb{R}}$ consists of a single point, which is the origin.

Computing rational function limits often reduces to path tracking within 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 $0$ is in the closure of the semi-algebraic set $S$ defined by:

$$ f_1 = \cdots = f_m = 0, \quad g_1 > 0, \ldots, g_p > 0. $$

Then, there exists real analytic curve $\gamma : [0, \varepsilon) \rightarrow \mathbb{R}^n$, with $\gamma(0) = 0$, and $\gamma(t) \in S$ for $t > 0$.

**Remark 20.** For a semi-algebraic set $S$, testing $0 \in \overline{S}$ can be phrased as a quantifier elimination problem and thus solved by Cylindrical Algebraic Decomposition (CAD):

$$ 0 \in \overline{S} \iff (\forall \varepsilon > 0) (\exists \text{x} \in \mathbb{R}^n) \|\text{x}\| < \varepsilon \Rightarrow \text{x} \in S. $$

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 \( C_{g,\omega} \subseteq C_{f,\omega} \) does not hold.

**Proposition 25.** Assume that \( \omega \in \{g = 0, f > 0\} \), then \( \lim_{(x_1, \ldots, x_n) \to \omega} \frac{f(x_1, \ldots, x_n)}{g(x_1, \ldots, x_n)} \) does not exist.

**Proof.** Assume by contradiction that \( \lim_{(x_1, \ldots, x_n) \to \omega} \frac{f(x_1, \ldots, x_n)}{g(x_1, \ldots, x_n)} \) exists and equals \( \ell \in \mathbb{R} \). Let us fix \( \varepsilon > 0 \). Then, there exists \( r > 0 \) such that for all \( x \in B(\omega, r) \), we have \( \ell - \varepsilon \leq q(x) \leq \ell + \varepsilon \). Thus, \( q(x) \) is bounded on \( B(\omega, r) \). From the hypothesis, for all \( q \) such that \( \lim_{(x, y) \to \omega} f(x, y) = 0 \) as well as \( (x, y) \in B(y, r') \). This contradicts the fact that \( q(x) \) is bounded on \( B(\omega, r) \). \( \square \)

**Proposition 26.** Assume that both \( \omega \in \{f = 0, g > 0\} \) and \( \omega \in \{g = 0, f > 0\} \) 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 \( \omega \in \{g = 0, f > 0\} \) and Proposition 25 imply that \( \lim_{(x_1, \ldots, x_n) \to (0, \ldots, 0)} q(x_1, \ldots, x_n) \) cannot be finite. \( \square \)

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 \( \omega \in \{f \neq 0, g = 0\} \). Assume from now on that \( C_{g,\omega} \subseteq C_{f,\omega} \) holds. Let \( E \) be a connected component of \( \mathbb{R}^n \setminus C_{f,\omega} \). Theorem 2.1 in [54], suggests that one can apply a multivariate version of L'Hospital’s Rule to compute

\[
\lim_{(x_1, \ldots, x_n) \to (0, \ldots, 0)} \frac{f(x_1, \ldots, x_n)}{g(x_1, \ldots, x_n)},
\]

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, [105]). Let \( U \subseteq \{(x, y) \in \mathbb{R}^2 \mid 0 \leq x \text{ and } 0 \leq y\} \) be a neighbourhood of \( \omega \) 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 \( \lim_{x \to 0} f_{xy}, \lim_{y \to 0} g_{xy} \notin \{(0, 0), (\pm \infty, \pm \infty)\} \), that is, no indeterminate forms. Then we have

\[
\lim_{(x, y) \to (0, 0)} \frac{f(x, y)}{g(x, y)} = \lim_{(x, y) \to (0, 0)} \frac{f_{xy}(x, y)}{g_{xy}(x, y)}.
\]

**Example 28.** Consider \( f(x, y) = xy^2 - y \) 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:
8.6. Limits of multivariate rational functions: general case

Theorem 20 (Bivariate L’hôpital rule). Let $N$ be a neighbourhood in $\mathbb{R}^2$ containing a point $p$ at which two differentiable functions $f : N \to \mathbb{R}$ and $g : N \to \mathbb{R}$ are zero. Set $C := \{(x, y) \in N : f(x, y) = g(x, y) = 0\}$, and suppose that $C$ is a smooth curve through $p$. Suppose there exists a vector $v$ not tangent to $C$ at $p$ such that the directional derivative $D_v(g)$ of $g$ in the direction of $v$ is never zero within $N$. More generally, if $C$ consists of a union of two or more smooth curves through $p$, suppose that for each component $E_i$ of $N \setminus C$ we can find a vector $v_i$, not tangent at $p$ to any of the curves comprising $C$ such that $D_{v_i}(g) \neq 0$ on $E_i$. Then $\lim_{(x,y)\to p} \frac{f(x,y)}{g(x,y)}$ exists if the limits $\lim_{(x,y)\to p,(x,y)\in E_i} \frac{D_{v_i}(f)}{D_{v_i}(g)}$ 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 $\lim_{(x,y)\to p \in E_1} \frac{f(x,y)}{g(x,y)}$ 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.
Chapter 8. Computing Limits of Multivariate Rational Functions

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 $v$ such that $[0, v)$ intersects $E_i$ on a segment, we assume that there exists $r > 0$ such that $D_v g(x) \neq 0$ holds in $E_i \cap B(\underline{0}, r)$. Then we have:

$$\lim_{(x_1, \ldots, x_n) \to p} \frac{f(x_1, \ldots, x_n)}{g(x_1, \ldots, x_n)} = \lim_{(x_1, \ldots, x_n) \to p, (x_1, \ldots, x_n) \in E_i} \frac{D_v f(x_1, \ldots, x_n)}{D_v g(x_1, \ldots, x_n)}.$$

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 $v = (-1, 1, 0)$, thus $D_v g = \nabla g \cdot v = -4x + 4y$. Observe that $D_v f$ does not vanish within $E$. Then, we have:

$$\lim_{x \to \underline{0}} \frac{f(x)}{g(x)} = \lim_{x \to \underline{0}, x \in E} \frac{D_v f(x)}{D_v g(x)} = \lim_{x \to \underline{0}, x \in E} \frac{-2x - 2y}{-4x + 4y},$$

which implies that the latter limit, and consequently, $\lim_{x \to \underline{0}} \frac{f(x)}{g(x)}$ 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 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.

1The list of the examples and the timings corresponding to Table 8.1 can be found at [www.regularchains.org/RationalLimit/RationalFunctionLimit.zip](http://www.regularchains.org/RationalLimit/RationalFunctionLimit.zip) and [www.regularchains.org/RationalLimit/Report](http://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](http://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 Chapter 5, 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.
Table 8.1: Comparisons between three different commands for computing the limit of real multivariate 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 implemented 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 \( k \) and total degree \( d \), we show that the Yun-Moses polynomials (needed when applying the EHC to \( F(X, Y) \)) can be computed within \( O(d^3 M(d)) \) operations in \( k \), where \( n \mapsto M(n) \) 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 \( O(k d M(d)^2) \) operations in \( k \) (instead of \( O(k^2 d M(d)^2) \) in a direct approach) or within \( O(k d M(d)) \) operations in the algebraic closure of \( k \).

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 \( O(d^3 M(d) + k^2 d M(d)) \) operations in \( \mathbb{C} \). In \([51]\), H.T. Kung and J.F. Traub propose a method for computing 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 \( O(d^2 k M(k)) \) (resp. \( O(d^2 M(k)) \)) operations over \( \mathbb{C} \), using a linear (resp. quadratic) lifting scheme. Furthermore, by the improvements 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 \( O(d^2 M(k)) \) 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 \texttt{TangentCone} implements this algorithm in the \texttt{AlgebraicGeometryTools} subpackage \cite{5} of the RegularChains library.

\section{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 \texttt{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 \texttt{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 \texttt{AlgebraicGeometryTools} subpackage of RegularChains library.
Bibliography


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. 

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. 

Parisa Alvandi, Amir Hashemi. 
