COMPUTATIONAL ASPECTS OF SINGULARITIES

ANNE FRÜHBIS–KRÜGER

Preface

This text is a set of notes of a mini-course entitled *Computational Aspects of Singularities* given at the *School on Singularities in Geometry and Topology* in August 2005. The aim of these talks was to introduce the participants to the use of computational methods for checking and studying properties of explicit examples of singularities; to this end, the first part of the course was devoted to an overview of computational tools for tasks from singularity theory. We then proceed step by step from the simple application of predefined computational tools to more complex applications, like studying families of singularities and constructing hypersurfaces with prescribed singularities. Using the algorithmic resolution of singularities as an example, we also show, how a rather complex computational task can be tackled by decomposing it into several smaller tasks.

As this set of talks was embedded into a school on singularities, some familiarity of the readers with the (singularity theory) background of the treated computational tasks is assumed, but references providing a starting point for reading on the theoretical background are also specified for each topic. Hence, this text only puts each task into context by recalling basic definitions and some properties of the discussed objects from the algebraic point of view before outlining the computational approach to it and discussing a practical example. For the convenience of the readers, all practical examples have been treated using the same Computer Algebra System SINGULAR (see [22]); but, of course, many of the discussed algorithms and applications are also available in other Computer Algebra Systems.

In the first section of this article, the calculation of the singular locus is used as an example on how to describe a given singularity in a computer algebra system and on how to determine basic properties of it using standard algorithms from computer algebra. The subsequent section is then devoted to the computational study of germs of singularities and related invariants like e.g. dimension, multiplicity and Milnor number, whereas the last two chapters lead to more complex applications including deformations, hypersurfaces with assigned singularities and algorithmic resolution of singularities in characteristic zero.

I would like to thank the organizers of this conference, the professors J.-P. Brasselet, J. Damon, M. Lejeune-Jalabert and Lê D.-T. for the opportunity to take part in and contribute to this interesting meeting. In addition to that I would like to thank H. Schönemann, G. Pfister and several students at the University of Kaiserslautern for many useful comments on earlier versions of this text, and F.-O. Schreyer and the referee for many helpful comments on the exposition of the material.

1. Studying the Singular Locus

For novice users of CA-systems, a first obstacle to using software for studying explicit examples is often the question how to encode the geometric object in the language of the computer algebra system. Using SINGULAR as an example of a CA-system, we shall illustrate these first steps with the calculation of the singular locus of a given variety. Along the way, we see applications of standard methods of computational commutative algebra like elimination of variables, primary decomposition and normalization. For a more detailed discussion of these techniques and for a description of the underlying algorithms see e.g. [21].

1.1. *The Jacobian criterion*

Computational Task

Given an affine variety $V(I) \subset K^n$ over a (perfect) field K, corresponding to an ideal $I = \langle f_1, \ldots, f_m \rangle \subset K[x_1, \ldots, x_n]$, our first task is to to determine its singular locus by means of the Jacobian criterion.

Background

For a detailed discussion of the following notions and related topics, see, for instance, [21],section 5.7.

Definition 1. (Singular Locus) Given a ring $A = K[x_1, \ldots, x_n]/I$ the set

$$Sing(A) := \{P \in Spec(A) \mid A_p \text{ is not regular}\}$$

is called the singular locus of A.

The singular locus can be determined by means of the Jacobian criterion:

Lemma 2. *(Jacobian criterion) Let K be a perfect field, $I = \langle f_1, \ldots, f_m \rangle$ and let $A = K[x_1, \ldots, x_n]/I$ be equidimensional. Let $J \subset A$ be the ideal generated by the $(n - dim(A))$-minors of the Jacobian matrix $(\frac{\partial f_i}{\partial x_j})$. Then*

$$Sing(A) = V(J).$$

Computational Solution (Irreducible Case)

Example 1 (irreducible curve specified by parametrization)
In this first example, we consider a curve in $\mathbb{A}^3_{\mathbb{C}}$ which is specified by the parametrization

$$\begin{aligned} \mathbb{A}^1_{\mathbb{C}} &\longrightarrow \mathbb{A}^3_{\mathbb{C}} \\ t &\longmapsto (t^3, t^4, t^5). \end{aligned}$$

Our computational subtasks here will be

- computation of the ideal of the curve
- application of the Jacobian criterion to compute the singular locus
- simplifying the description of the resulting set of points.

At this point, it is important to mention that, in general, all calculations in a computer algebra system are performed over the rationals or over suitable field extensions thereof, but not over the real or complex numbers. This does not change the calculations, but we need to be aware of this when discussing computational results.

We now determine the **ideal of the curve** by using an elimination technique for the parameter t. More precisely, we already know that the ideal of the graph of the parametrization map in $\mathbb{A}^1_{\mathbb{C}} \times \mathbb{A}^3_{\mathbb{C}}$ is

$$\langle x - t^3, y - t^4, z - t^5 \rangle \subseteq \mathbb{C}[x, y, z, t],$$

where the variable t corresponds to $\mathbb{A}^1_{\mathbb{C}}$. We can now determine the image of the graph under the projection map $\mathbb{A}^1_{\mathbb{C}} \times \mathbb{A}^3_{\mathbb{C}} \longrightarrow \mathbb{A}^3_{\mathbb{C}}$ via elimination of the variable t. Elimination is available as a standard command in many CA-systems; details on its implementation using Gröbner bases w.r.t. elimination orderings can, for example, be found in [21], sections 1.2 and 1.8.

In the language of SINGULAR the computations which we just described can be performed as follows[a]:

```
                               // ring of char 0 containing
> ring r=0,(t,x,y,z),dp;       // variables x,y,z and t

> ideal Ip=x-t^3,y-t^4,z-t^5;  // input of the parametrization

> ideal Ie=eliminate(I,t);     // compute ideal of the curve:
> Ie;                          // elimination provides equations
_[1]=y2-xz                     // of the projection to A^3
_[2]=x2y-z2
_[3]=x3-yz

> ring r2=0,(x,y,z),dp;        // we do not need t any more:
                               // move the ideal of the
> ideal I=imap(r,Ie);          // curve to this ring
```

As the variety was specified by means of its parametrization, we already know that we are dealing with an irreducible curve. Hence we do not need to worry about the equidimensionality condition in the Jacobian criterion. We can simply proceed by computing the **Jacobian matrix** and the **ideal of minors** of the appropriate size. For didactic reasons, however, we also include the computation of the dimension of $K[x,y,z]/I$ (of which we know that it is 1):

```
                               // compute dimension of
                               // K[x,y,z]/I;
                               // 'std' (Groebner basis) required
> int dimA=dim(std(I));        // for using command 'dim'
> dimA;                        // we already knew that it is 1
1

> matrix Jac=jacob(I);         // determine Jacobian matrix
> print(Jac);                  // show the matrix
```

[a]The sequence of characters '//' in SINGULAR–output marks a comment and should not be considered as input.
The sequence of characters 'dp' in the declaration of the ring specifies the ordering on the monomials. The effects of the choice of ordering on the computations are discussed in detail in [21], section 1.5

```
-z, 2y,-x,
2xy,x2,-2z,
3x2,-z,-y

> ideal J=minor(Jac,3-dimA);   // determine the minors
> ideal sL=J+I;                // ideal of singular locus
> sL;                          // show content of variable sL
sL[1]=-x2y-2z2
sL[2]=-2xy2+6x2z
sL[3]=-2y2-xz
sL[4]=3x3+yz
sL[5]=3x4+2xyz
sL[6]=6x2y-z2
sL[7]=x3-4yz
sL[8]=2x2y+2z2
sL[9]=4xy2+x2z
sL[10]=y2-xz
sL[11]=x2y-z2
sL[12]=x3-yz
```

12 generators for the ideal of the singular locus seem to be quite a lot. Indeed, ideals generated by minors of matrices tend to have a high number of redundant generators and we can try to find a smaller set of generators by applying appropriate commands such as the SINGULAR-command `mstd`. But in our particular case, we are only interested in the **set of singular points**, which is by the Hilbert Nullstellensatz[b] also the vanishing locus of the radical[c] of the ideal which we computed:

```
                               // the command radical is contained
                               // in a Singular library which
```

[b]Over an **algebraically closed** field K, Hilbert's Nullstellensatz states that, given an ideal $I \subset K[x_1, \ldots, x_n]$, the ideal of $V(I)$ is precisely the radical $\sqrt{I} = \{f \in K[x_1, \ldots, x_n] | f^m \in I \text{ for some } m\}$. This is one of the situations, where we need to remind ourselves that even though we are computing over $\mathbb{Q}$, our reasoning takes place over $\mathbb{C}$

[c]There are different algorithms for the computation of the radical of a given ideal, e.g. the one of Krick and Logar [27] or the one of Eisenbud, Huneke and Vasconcelos [13]. As different approaches have different advantages and drawbacks, a CA-system often provides several different commands for it. It is often worthwhile to try another algorithm, if the first one did not show good performance on a particular class of examples.

```
> LIB "primdec.lib";           // needs to be loaded before using
> radical(sL);                 // the command
_[1]=z
_[2]=y
_[3]=x
```

Thus we see that the only singular point of this curve is the origin.

Background (continued)

In the following examples, we will use primary decomposition. To fix notation we briefly sketch some definitions and statements of this field. For a detailed discussion of the theoretical aspects see e.g. [12], section 3.3, for a treatment from the algorithmic point of view see e.g. [21], section 4.3.

Definition 3. (Associated Primes) Let R be a noetherian ring and let $I \subset R$ be an ideal. The set of associated primes of I is defined as

$$Ass(I) := \{P \subset R | P \ \ prime, \ \ P = Ann_{R/I}(b) \ \text{ for some } \ b \in A\}.$$

Let $P, Q \in Ass(I)$ and $Q \subsetneq P$, then P is called an embedded prime ideal of I; the elements of $Ass(I)$ which are not embedded are referred to as minimal prime ideals of I.

Theorem 4. *(Primary Decomposition for Ideals) Let R be an noetherian ring. Every ideal $I \subset R$ is the intersection of finitely many primary ideals. Furthermore, if $I = \bigcap_{i=1}^{s} Q_i$ is an irredundant[d] primary decomposition of I (with Q_i being P_i-primary), then the P_i are precisely the associated prime ideals of R/I.*

Computational Solution (equidimensional case)

Example 2 (reducible, equidimensional variety)
In the second example, we consider a very simple reducible variety consisting of 3 smooth hyperplanes. Our computational subtasks will be:

- defining each of the hyperplanes separately and forming their union by intersecting the corresponding ideals

[d] A primary decomposition is called irredundant, if no Q_i can be omitted in the intersection.

- applying the predefined procedure `slocus` which performs precisely the same steps as we did in example 1
- studying the singular locus using primary decomposition.

```
                                   // polynomial ring:
> ring r=0,(x,y,z,w),dp;           //    char. 0, 4 var.
> ideal I1=x,w;                    // the y-z plane
> ideal I2=y,z;                    // the x-w plane
> ideal I3=z-x^2-y^2,w;            // another smooth surface
> ideal Itemp=intersect(I1,I2);  // union of y-z and x-w planes
> Itemp;
Itemp[1]=yw
Itemp[2]=zw
Itemp[3]=xy
Itemp[4]=xz
> ideal I=intersect(Itemp,I3);    // union of all three surfaces
> I;
I[1]=zw
I[2]=yw
I[3]=-x2zw-y2zw+z2w
I[4]=-x2yw-y3w+yzw
I[5]=-x3z-xy2z+xz2
I[6]=-x3y-xy3+xyz
I[7]=-y2w+zw
```

In this case, we expect the **singular locus** to be the locus where the surfaces meet, since each of the surfaces is smooth. Using the ideals of the respective surfaces, an easy calculation by hand shows that the surfaces V(I1) and V(I2) respectively V(I2) and V(I3) meet in the point $V(\langle x,y,z,w\rangle)$, whereas the intersection locus of V(I1) and V(I3) is the curve $V(\langle x,w,z-y^2\rangle)$. We now determine the singular locus using the procedure `slocus` and compare this result to the result of our computation by hand.

```
                                   // slocus is in the library
> LIB "sing.lib";                  //        'sing.lib'
> ideal sL=slocus(I);              // compute singular locus
> size(sL);                        // size of set of generators
91
```

To see both components of the singular locus, we cannot restrict our considerations to the radical or the minimal associated primes in this case, because one component, the point, is contained in the other component. We need to consider a full **primary decomposition**[e] of the ideal sL, to find both components:

```
> LIB "primdec.lib";              // library for primary decomp.

> minAssGTZ(sL);                  // minimal associated primes
[1]:                              // just one minimal prime
   _[1]=y2-z
   _[2]=w
   _[3]=x

> primdecGTZ(sL);                 // complete primary decomp.
[1]:                              // first primary component
   [1]:                           // * primary ideal
      _[1]=w
      _[2]=y2-z
      _[3]=x
   [2]:                           // * corresponding prime
      _[1]=w
      _[2]=y2-z
      _[3]=x
[2]:                              // 2nd primary component
   [1]:                           // * primary ideal
      _[1]=w2
      _[2]=zw
      _[3]=z2
      _[4]=yw
      _[5]=y3z
      _[6]=y4-y2z
```

[e]Some aspects of the theoretical background of primary decomposition are briefly sketched at the beginning of section 1.3; an introduction to the computation of a primary decomposition, of the radical and of several related tasks can be found in [21], chapter 4. As for the radical there are also different approaches to the calculation of the primary decomposition, e.g. the one of Gianni, Trager and Zacharias [18] and the one of Shimoyama and Yokoyama [36].
In the primary decomposition commands implemented in SINGULAR, the list containing the result is not ordered. Therefore permutations of the list entries occur quite often.

```
    _[7]=xyz
    _[8]=xy2
    _[9]=x3w
    _[10]=x3z
    _[11]=x3y
    _[12]=x6
[2]:                              // * corresponding prime
    _[1]=w
    _[2]=z
    _[3]=y
    _[4]=x
```

1.2. *The non-equidimensional case*

The two previous examples were constructed in a suitable way to make sure that they are equidimensional. But in general this is a priori unknown. Hence the variety needs to be decomposed first. Using primary decomposition at this point is expensive and often leads to a rather high number of components whose singular loci and intersections need to be computed. Moreover, it is unnecessary, because we only need to satisfy the condition that each of the parts is equidimensional; a task which is performed by the algorithm of **equidimensional decomposition**.

Background

A detailed discussion on equidimensional decomposition and its computation can be found e.g. in [21], section 4.4. Here we only recall the definitions:

Definition 5. (Equidimensional Part) Let R be a noetherian ring, $I \subset R$ an ideal and let $I = \bigcap_{i=1}^{s} Q_i$ be an irredundant primary decomposition. The equidimensional part of I is the intersection of all primary ideals Q_i for which $dim(R/Q_i) = dim(R/I)$; I is called equidimensional if it coincides with its equidimensional part.

Remark 6. (Equidimensional Decomposition) Iterating the process of determining the equidimensional part and removing it for ideals without embedded primes, we obtain a decomposition of I into equidimensional ideals. In the presence of embedded primes, however, we only obtain a decomposition into equidimensional ideals such that the intersection of their radicals is the radical of I.

Computational Solution (general case, no embedded primes)

Example 3 We now consider the union of the space curve of example 1 and the surface $V(x^3 - y^2)$ which possesses a non-isolated singularity along the z-axis. Here the computational subtasks which we consider for this example will be

- defining the ideals of the components separately and intersecting them to obtain the ideal describing our variety
- computing an equidimensional decomposition of the ideal
- computing the singular locus by determining the singular loci and intersections of the equidimensional parts
- computing a primary decomposition of the singular locus

```
                                // polynomial ring:
> ring r=0,(x,y,z),dp;          //    char 0, 3 var.
                                // the previously computed ideal
> ideal I1=y2-xz,x2y-z2,x3-yz;  //    of the curve in example 1
> ideal I2=x^3-y^2;             // the singular surface

> ideal I=intersect(I1,I2);     // the union of the two varieties
> I;
I[1]=x3y2-x4z-y4+xy2z
I[2]=x5y-x2y3-x3z2+y2z2
I[3]=x6-x3y2-x3yz+y3z
                                // equidim. decomp. is in
> LIB "primdec.lib";            //    library 'primdec.lib'

                                // compute list of
> list li=equidim(I);           //    equidim. parts of I
> li;
li[1]:                          // part of dim. 1
   _[1]=y2-xz
   _[2]=x2y-z2
   _[3]=x3-yz
li[2]:                          // part of dim. 2
   _[1]=x3-y2
```

Using this equidimensional decomposition, we can then compute the singular locus of each of the equidimensional parts by the Jacobian criterion. The union of these **singular loci** and of the **intersection locus** of the various parts is precisely the singular locus of the whole variety.

Example 4 (example 3 continued)

```
> LIB "sing.lib";                    // 'sing.lib' contains slocus
> ideal sL1=slocus(li[1]);           // sing. locus of 1-dim. part
                                     //    i.e. point from example 1
> ideal sL2=slocus(li[2]);           // sing. locus of 2-dim. part
> sL2;
sL2[1]=x3-y2
sL2[2]=-2y
sL2[3]=3x2

                                     // intersection of 1- and
> ideal inter12=li[1]+li[2];         // 2-dim. parts
> inter12;
inter12[1]=y2-xz
inter12[2]=x2y-z2
inter12[3]=x3-yz
inter12[4]=x3-y2

                                     // union of contributions
                                     // to sing. locus
> ideal sL=intersect(sL1,sL2,inter12);
```

Now let us check whether we can **identify the various contributions** in the primary decomposition of the singular locus:

```
> primdecGTZ(sL);
[1]:                                 // first primary component:
   [1]:                              // singular locus of surface
      _[1]=y
      _[2]=x2
   [2]:
      _[1]=y
      _[2]=x
[2]:                                 // second primary component:
   [1]:                              // singular locus of curve
      _[1]=z2                        // but also one of intersection
```

```
      _[2]=y2-yz                      // points of the two parts
      _[3]=xyz
      _[4]=x2z
      _[5]=x3-yz
   [2]:
      _[1]=z
      _[2]=2y-z
      _[3]=x
[3]:                                  // third primary component:
   [1]:                               // other intersection point
      _[1]=z-1                        // of the two parts
      _[2]=y-1
      _[3]=x-1
   [2]:
      _[1]=z-1
      _[2]=y-1
      _[3]=x-1
```

1.3. *Finding the Correct Number of Components*

In the previous examples, we have already used primary decomposition to determine the components of a given variety. This is, however, one of the situations where the fact that we are calculating over the rationals, but arguing over the complex numbers can easily lead to wrong conclusions.

Computational Task

Given a curve $V(I) \subset \mathbb{A}^k_{\mathbb{C}}$, determine the number of branches.

Background

There are several possible approaches to this task. We shall only consider two, the first one based on primary decomposition (see section 1.1), the other one on normalization (see e.g. [21], sections 3.2 and 3.6 and second half of section 5.7 for algorithmic aspects, or [7], section 4.4 for a treatment from the point of view of singularities). For the second one, we briefly recall notation and some important statements here:

Definition 7. (Normalization) Let R be a reduced ring. The normalization of R is the integral closure of R in its total ring of fractions.

Theorem 8. *(Serre's Normality Criterion) Let R be a reduced noetherian ring. Then R is normal if and only if the following two conditions are satisfied:*

(R1) R_P is a regular local ring for every prime ideal P of height one.
(S2) Let $f \in R$ be a non-zerodivisor, then $minAss(\langle f \rangle) = Ass(\langle f \rangle)$.

Remark 9. A regular local ring is normal.

Remark 10. A normal one-dimensional variety is non-singular.

Computational Pitfalls and Solutions

Example 5 Consider the variety $V(\langle x^4 - yz^2, xy - z^3, y^2 - x^3z \rangle) \subset \mathbb{A}^3_{\mathbb{C}}$. It is a curve which has only one singular point at the origin. The task is to compute the number of branches of this space curve.

```
> ring r=0,(x,y,z),dp;
> ideal I=x4-yz2,xy-z3,y2-x3z;     // the ideal of the curve
> primdecGTZ(I);                   // primary decomposition
[1]:
   [1]:
      _[1]=z8+yz6+y2z4+y3z2+y4
      _[2]=xz5+z6+yz4+y2z2+y3
      _[3]=-z3+xy
      _[4]=x2z2+xz3+xyz+yz2+y2
      _[5]=x3+x2z+xz2+xy+yz
   [2]:
      _[1]=z8+yz6+y2z4+y3z2+y4
      _[2]=xz5+z6+yz4+y2z2+y3
      _[3]=-z3+xy
      _[4]=x2z2+xz3+xyz+yz2+y2
      _[5]=x3+x2z+xz2+xy+yz
[2]:
   [1]:
      _[1]=-z2+y
      _[2]=x-z
   [2]:
      _[1]=-z2+y
      _[2]=x-z
```

This result seems to imply that the number of branches could be 2. To check the plausibility of this conclusion, we consider the Milnor number[f] of the singularity at the origin, which turns out to be 12. But by the formula $\mu = 2\delta - r + 1$ (see [4]), an even number of branches r would imply an odd Milnor number. Therefore, the conclusion that there are two branches is not plausible. The reason for this strange 'result' is that we are calculating over the rationals, but all arguments and considerations are performed over the complex numbers. In particular, the first primary component actually consists of 4 components, as we see by considering the normalization.[g]

```
                                 // normalization is in
> LIB "normal.lib";              //    the library 'normal.lib'
> list li=normal(I);             // compute normalization
// 'normal' created a list of 1 ring(s).
// To see the rings, type (if the name of your list is nor):
     show( nor);
// To access the 1-st ring and map (similar for the others), type:
     def R = nor[1]; setring R;  norid; normap;
// R/norid is the 1-st ring of the normalization and
// normap the map from the original basering to R/norid

> size(li);                      // how many branches (over Q)
1

> def norring=li[1];             // consider branch more closely
> setring norring;
> basering;
//   characteristic : 0
//   number of vars : 2
//        block   1 : ordering a
//                  : names    T(1) T(2)
//                  : weights     1    0
```

[f]Obviously, this singularity is neither a hypersurface nor an ICIS. Therefore its Milnor number cannot be computed directly by means of the tools discussed below in section 2.3. But the singularity is a quasihomogeneous curve singularity of Cohen-Macaulay type 2 and the tools of section 2.3 allow us to determine the Tjurina number which is 13 here. Hence we can compute the Milnor number by the formula $\mu = \tau - t + 1$ for quasihomogeneous curve singularities, where t is the Cohen-Macaulay type (cf. [20] for details on this formula).

[g]Recall that for curves the normalization coincides with a parametrization.

```
//        block   2 : ordering dp
//                  : names    T(1) T(2)
//        block   3 : ordering C
> norid;
norid[1]=T(2)^5-1
```

At first glance, it might seem strange that the ring describing the normalization has two variables, although it describes a 1-dimensional object. But looking at `norid`, we see that the second variable is only used to specify an appropriate field extension over the rationals such that all branches are separated. In particular, we see that we have 5 branches.

Other possibilities to determine the correct number of branches, are the use of a generic projection to the plane followed by a Puiseux expansion (see 2.4 below) or the use of absolute primary decomposition. A detailed example of the first of these alternative approaches, however, is beyond the scope of this article; an application of absolute primary decomposition (in a different context) can be found in section 4.2.5.

2. Computing Invariants of Isolated Singularities

After briefly discussing some aspects of algorithmic calculations in local rings, this section contains a few examples of invariants which can be computed in practice.[h] To each example, we also mention where further information on the algorithmic aspects can be found. This list of examples is by no means exhaustive, it is intended as a kind of appetizer for the reader to start discovering what is available as algorithmic tools for their field of research; for simplicity of the presentation, we only discuss examples whose implementation is also available in Singular and not even half of the functionality of Singular in this area is mentioned.

2.1. *Local and Global Considerations*

Up to this point, we have only studied varieties, but not germs. As a consequence all computations have been performed in polynomial rings, not in power series rings.[i] Actually, a full implementation of power series

[h] Readers who are not familiar with the theoretical background of some of the examples in subsections 2.3 to 2.6 may safely skip the respective subsection.
[i] In Singular, the functionality of primary decomposition, radical and normalization is only available in polynomial rings, not in localizations thereof. For a primary decom-

rings on a computer is not feasible, but nevertheless many practical tasks can be tackled by using the localization of the respective polynomial ring at the origin instead. (For obvious reasons, the input and output still need to be specified in terms of polynomial data.)

To understand the basic idea behind the implementation of this type of localizations of polynomial rings, we first consider the representation of polynomials on the computer. The need to represent polynomials on the computer in a *unique* way forces us to use a total ordering on the set of all monomials; this ordering has to be compatible with multiplication of monomials. If the monomial 1 is the smallest monomial, the monomial ordering is called global and the ring is a polynomial ring; if 1 is the largest monomial, all elements whose largest term is a constant are considered as units; the ring is therefore a localization of the polynomial ring at the origin and the monomial ordering is called local. Orderings, in which some, but not all monomials are smaller than 1, are also possible and are usually referred to as mixed orderings. A detailed discussion of the influence of the choice of ordering on the ring is beyond the scope of this set of two talks and we refer the participants to a suitable textbook, e.g. [21], section 1.5, or [8], sections 3.2 and 9.1.

Computational Task

Given an ideal $I \subset K[x_1, \dots, x_n]$, compare results of Gröbner basis computations w.r.t local and global orderings, i.e. in the rings $K[x_1, \dots, x_n]_{\langle x_1, \dots, x_n \rangle}$ and in $K[x_1, \dots, x_n]$.

Computational Example

Example 6 In this example, we show some very simple calculations to illustrate the contrast between the local and global monomial orderings.

The first of these small tasks is the calculation of a Gröbner basis resp. standard basis for the ideal of the variety consisting of the plane $V(z+1)$ and the two lines $V(x, y)$ and $V(x-1, y-1)$ in $\mathbb{A}^3_{\mathbb{C}}$:

---

position in a localization of a polynomial ring it is, however, possible to compute the primary decomposition in the polynomial ring and to subsequently drop those components which are irrelevant in the localization. But this approach might be misleading as the following example shows: Consider the plane curve defined by $y^2 - x^2 + x^3 = 0$; a primary decomposition in the polynomial ring provides just one component, although the germ of this curve at the origin consists of two branches.

```
                                    // polynomial ring in 3 var.:
> ring rg=0,(x,y,z),dp;             //    global ordering dp
                                    // localization at origin:
> ring rl=0,(x,y,z),ds;             //    local ordering  ds

> setring rg;                       // go back to polynomial ring
> ideal I1=z+1;                     // the plane V(z+1)
> ideal I2=x,y;                     // first line
> ideal I3=x-1,y-1;                 // second line

> ideal Itemp=intersect(I1,I2);     // union of the first two
> ideal I=intersect(Itemp,I3);      // union of all three
> I;
I[1]=-xz+yz-x+y
I[2]=xyz+xy-yz-y
                                    // remark: I is radical by
                                    //         construction

> ideal J=groebner(I);              // compute Groebner basis
> J;
_[1]=xz-yz+x-y
_[2]=y2z+y2-yz-y

> setring rl;                       // now go to localization
> def I=imap(rg,I);                 // map ideal via identity map
> I;
I[1]=-x+y-xz+yz                     // observe the different way
I[2]=-y+xy-yz+xyz                   // of writing I[1]

> ideal J=groebner(I);              // compute standard basis
> J;
_[1]=x                              // we only see the components
_[2]=y                              // meeting the origin
```

Continuing with the same example, we now compute the dimensions and check whether the variety/germ is contained in the plane $V(x)$:

```
> setring rg;                       // back to polynomial ring
                                    // and compute dimension
                                    //  (remark: 'dim' needs
```

```
> dim(J);                               //   Groebner/standard basis)
2                                       // dimension of the plane

> setring rl;                           // back to localization
> dim(J);                               // applying 'dim' at 0
1                                       // dimension of components
                                        //   meeting 0

> setring rg;
                                        // ideal membership test:
                                        //      x in J ?
                                        //  (remark: 'reduce' needs
                                        //   Groebner/standard basis
> reduce(x,J);                          //   of J)
x                                       // answer: no
>setring rl;
> reduce(x,J);                          // same question locally
0                                       // answer: yes
```

2.2. *Dimension and Multiplicity*

As we already used the dimension of a variety or a germ in previous examples, this seems to be a good moment to look at its calculation and at related data. The notion of dimension itself can be phrased in several ways (e.g. for a local ring $(R, \mathfrak{m})$: maximal length of chains of prime ideals, minimal number of generators of an $\mathfrak{m}$–primary ideal in a local ring $(R, \mathfrak{m})$,etc.), but most accessible to the use in practical calculations is the definition by means of the degree of the Hilbert-Samuel polynomial.

Computational Task

Given an ideal $I \subset K[x_1, \ldots, x_n]$, compute the dimension and multiplicity of $V(I)$ at the origin.

Background

A detailed discussion on the Hilbert-Samuel polynomial and its calculation can be found in [21], sections 5.4 and 5.5. Here we only recall its definition to fix notation.

Definition 11. (Hilbert-Samuel Polynomial) Let $(R, \mathfrak{m})$ be a noetherian local ring where $\mathfrak{m}$ is generated by r elements, and assume for simplicity that $K = R/\mathfrak{m}$. The Hilbert-Samuel function of R (w.r.t. the filtration $\{\mathfrak{m}^k\}_{k\in\mathbb{N}}$) is defined as

$$\lambda(k) := \dim_K(R/\mathfrak{m}^k).$$

There exists a polynomial $f(t) \in \mathbb{Q}[t]$, the Hilbert-Samuel polynomial of R, of degree at most r such that $\lambda(k) = f(k)$ for all sufficiently large $k \in \mathbb{N}$. Writing $f(t)$ as $\sum_{i=0}^{d} a_i t^i$, the degree d of f coincides with the dimension of $(R, \mathfrak{m})$ and the multiplicity of $(R, \mathfrak{m})$ is defined as $d! \cdot a_d$.

Remark 12. It is possible to explicitly compute a Hilbert-Samuel polynomial of a given ideal with polynomial generators in the localization of a polynomial ring at the origin. The general idea of this calculation is to find a suitable system of generators (i.e. a standard basis of the ideal w.r.t. a local degree ordering), then pass to the ideal generated by the largest terms of the generators (the so-called leading ideal) and compute the desired $\mathbb{C}$-vector space dimensions for this new (monomial) ideal in a combinatorial way.

Computational Example

In SINGULAR, the dimension and multiplicity are directly accessible as kernel commands `dim` and `mult` which require a standard basis w.r.t. a local degree ordering as an input.[j]

Example 7 To illustrate the use of these commands, we now consider a space curve singularity at the origin consisting of an E_6 singularity in the x-y plane and the z-axis.

```
> ring r=0,(x,y,z),ds;               // a local degree ordering
> ideal I=xz,yz,x3-y4;               // a space curve singularity
> I=groebner(I);                     // compute standard basis
> I;
I[1]=xz
```

[j]Note that the command **hilb** does **not** compute the Hilbert-Samuel polynomial. It computes the Hilbert-Poincaré series of the homogeneous ideal generated by the initial terms of the given generators of the ideal. From these it is, of course, possible to compute both the Hilbert polynomial and a Hilbert-Samuel polynomial of the given ideal, the generators formed a standard basis w.r.t. to an appropriate monomial ordering as is explained e.g. in [21], sections 5.2 and 5.5 respectively.

```
I[2]=yz
I[3]=x3-y4

                                 // ideal generated by largest
                                 // monomials of the generators
> lead(I);                       // of I
_[1]=xz
_[2]=yz
_[3]=x3
> dim(I);                        // compute dimension
1
> mult(I);                       // multiplicity
4
> dim(lead(I)),mult(lead(I));    // should give the same values
// ** _ is no standardbasis      // ** automatic warnings can be
// ** _ is no standardbasis      // ** ignored if ideal monomial
1 4
```

2.3. *Milnor and Tjurina Number*

Computational Task

Given a polynomial $f \in \mathbb{C}[x_1, \ldots, x_n]$ such that all singularities of $V(f)$ are isolated, determine the Milnor and Tjurina numbers at all critical (resp. singular) points. In particular, determine these invariants at the origin.

Background

For a discussion of the Milnor and Tjurina numbers of hypersurface singularities see any book on singularity theory, e.g. [7], section 3.4.

Definition 13. Let $f \in \mathbb{C}\{\underline{x}\}$ define a germ of an isolated hypersurface singularity at the origin. The Milnor number of the germ $(V(f), 0)$ is defined as the $\mathbb{C}$-vector space dimension

$$\mu = \dim_{\mathbb{C}} \left(\mathbb{C}\{\underline{x}\}/\langle \frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n} \rangle \right).$$

Its Tjurina number is defined as the $\mathbb{C}$-vector space dimension

$$\tau = \dim_{\mathbb{C}} \left(\mathbb{C}\{\underline{x}\}/\langle f, \frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n} \rangle \right).$$

Computational Solution

Example 8 In this example, we compute the Milnor and Tjurina number at the origin for the plane curve consisting of two cusps $V(x^2 - y^3)$ and $V(x^3 - y^2)$.

```
> ring r=0,(x,y),ds;                // local ring in 2 variables
> ideal I=(x^2-y^3)*(x^3-y^2);      // the curve
> ideal Jac=jacob(I);               // jacobian ideal of I
> groebner(Jac);                    // compute standard basis
_[1]=2x2y-5y4
_[2]=2xy2-5x4
_[3]=x5-y5
_[4]=y6

                                    // the command 'vdim' needs
                                    // a standard basis as input
> vdim(groebner(Jac));              // the Milnor number
11
> vdim(groebner(Jac+I));            // the Tjurina number
10
```

Alternatively, we can also use the corresponding predefined commands in the library 'sing.lib':

```
> LIB "sing.lib";
> milnor(I);                        // the Milnor number
11
> tjurina(I);                       // the Tjurina number
10
```

But what would have happened, if we had specified a global ordering instead of the local ordering?

```
> ring r2=0,(x,y),dp;               // global ordering
> ideal I=(x^2-y^3)*(x^3-y^2);      // the curve
> ideal Jac=jacob(I);               // jacobian ideal of I
> groebner(Jac);                    // compute Groebner basis
_[1]=3x2y3-5x4+2xy2
_[2]=3x3y2-5y4+2x2y
_[3]=x5-y5
_[4]=9y7-19x4y+10xy3
```

```
> vdim(groebner(Jac));
21
> vdim(groebner(Jac+I));
15
```

The numbers, which we computed here, are precisely the sums over the Milnor resp. Tjurina numbers of all singularities of the affine curve. Therefore we expect to find further critical points outside the origin whose multiplicities add up to 10 and further singular points whose Tjurina numbers add up to 5. To check this, we determine the singular locus, move to each of the other singular points and compute Milnor and Tjurina numbers there. Subsequently, we also study the critical locus, which, of course, contains the singular locus.

```
> LIB "primdec.lib";
> minAssGTZ(slocus(I));            // components of sing. locus
[1]:                               // the origin -- we already
   _[1]=y                          //    considered this one
   _[2]=x
[2]:                               // the point (1,1)
   _[1]=y-1
   _[2]=x-1
[3]:                               // a set of 4 points
   _[1]=y4+y3+y2+y+1               // <--- keep this in mind (*)
   _[2]=y3+y2+x+y+1

> setring r;                       // go back to local ring
                                   // translation of (1,1)
> map m1=r2,x+1,y+1;               //    to origin
> def I2=m1(I);                    // apply translation to curve
> I2;
I2[1]=6x2-13xy+6y2+9x3-11x2y-11xy2+9y3+5x4-3x3y-10x2y2-3xy3+5y4
      +x5-3x3y2-3x2y3+y5-x3y3
> milnor(I2);                      // Milnor number
1
> tjurina(I2);                     // Tjurina number
1

                                   // extend basefield to look
                                   // at the 4 points
```

```
> ring r2a=(0,a),(x,y),ds;          // adjoining parameter a
> minpoly=a4+a3+a2+a+1;             // minimal polynomial,
                                    //    see (*) above

                                    // translation of one of the
> map m2=r2,x-a3-a2-a-1,y+a;        //    4 points to origin
> def I3=m2(I);                     // apply translation to curve
> milnor(I2);                       // Milnor number
1
> tjurina(I2);                      // Tjurina number
1

> setring r2;                       // go back to r2 (global)
                                    // decompose set of
> minAssGTZ(jacob(I));              //    crit. points
[1]:                                // origin
   _[1]=y                           //    already considered
   _[2]=x
[2]:                                // (1,1)
   _[1]=y-1                         //    already considered
   _[2]=x-1
[3]:                                // 4 points
   _[1]=y4+y3+y2+y+1                //    already considered
   _[2]=y3+y2+x+y+1
[4]:                                // 4 critical points
   _[1]=81y4+54y3+36y2+24y+16
   _[2]=27y3+18y2+12x+12y+8
[5]:                                // 1 critical point
   _[1]=3y-2
   _[2]=3x-2
```

Actually, it would not have been necessary to move to each of the points and check the Milnor and Tjurina numbers explicitly, because we only had a difference of 10 for the Milnor and of 5 for the Tjurina number and this equals the number of additional points in the critical resp. singular locus.

The Milnor and Tjurina numbers for isolated complete intersection singularities are available by the same command (in the case of the Milnor number by use of the Lê-Greuel formula, see [30]). The Tjurina number for

Cohen-Macaulay codimension 2 singularities, which are not ICIS, is provided in the library 'spcurve.lib'; in the general case it can be obtained via the command T^1, see 3.1 below. For further details on these invariants see any textbook on singularities, e.g. [40] in the curve case, [7] for hypersurfaces or [31] in the case of complete intersections.

2.4. *Puiseux Expansion*

Computational Task

Given an element $0 \neq f \in \mathbb{C}\{x, y\}$ (which is assumed to be a polynomial for practical reasons) defining an isolated singularity at 0, determine the number of branches of the corresponding germ of the plane curve at the origin. Additionally compute the δ-invariants, the degree of the conductors for the branches and the intersection multiplicities of the branches.

Background

Here we do not state the rather technical definition of the Hamburger-Noether expansion of which a detailed discussion can be found in [5], chapter 1. Instead, we focus on recalling the definitions of the invariants which we want to determine. A compact and accessible discussion of these topics can be found in [7], chapter 5.

Definition 14. (Puiseux Expansion) Let $0 \neq f \in \mathbb{C}\{x, y\}$ be irreducible. Then there exist power series $x(t), y(t) \in \mathbb{C}\{t\}$ such that

- $f(x(t), y(t)) = 0$
- $\dim_{\mathbb{C}}(\mathbb{C}\{t\}/\mathbb{C}\{x(t), y(t)\}) < \infty$

Definition 15. (δ-Invariant, Conductor) Let $(R, \mathfrak{m})$ be the local ring of an irreducible plane curve singularity; R is a subring of its normalization $\tilde{R} = \mathbb{C}\{t\}$.

- The δ-invariant is defined as $\delta(R) := \dim_{\mathbb{C}}(\tilde{R}/R)$.
- The semigroup of values of R is defined as $\Gamma(R) := \{\text{ord}_t(a) |\ a \in R, a \neq 0\}$.
- The conductor is defined as $c(R) := \min\{b \in \Gamma(R) |\ b + \mathbb{N} \subset \Gamma(R)\}$.

Lemma 16. $Ann_R(\tilde{R}/R) = t^{c(R)}\tilde{R}$

Lemma 17. $\delta(R) = \#(\mathbb{N} \setminus \Gamma(R))$

Definition 18. (Intersection Multiplicity) Let $0 \neq f, g \in \mathbb{C}\{x, y\}$ define two germs of plane curve singularities (at 0) not having a common component. The intersection multiplicity of the two germs is defined as

$$\dim_{\mathbb{C}}(\mathbb{C}\{x, y\}/\langle f, g\rangle).$$

Computational Example

Example 9 (example 8 continued) Continuing where we stopped in our calculations in the previous example, we now apply Hamburger-Noether expansion and extract information about the given plane curve from it. Recall that this curve consisted of two branches $V(x^2 - y^3)$ and $V(y^2 - x^3)$.

```
> LIB "hnoether.lib";               // load corresponding library
                                    // hnexpansion needs argument
> poly f=I[1];                      //    of type poly
                                    // call Hamburger-Noether
> hnexpansion(f);                   //    expansion
[1]:                                // result lives in a new ring
   //   characteristic : 0
   //   number of vars : 2
   //        block   1 : ordering ls
   //                  : names    x y
   //        block   2 : ordering C
> def S=_[1];                       // give that ring the name S
> setring S;                        // and change to it

> hne;                              // result can be found in hne
[1]:                                // technical data, not
   [1]:                             //    really readable
      _[1,1]=0
      _[1,2]=x
      _[1,3]=0
      _[2,1]=0
      _[2,2]=1
      _[2,3]=x
   [2]:
      1,2
   [3]:
      0
```

```
   [4]:
      0
[2]:
   [1]:
      _[1,1]=0
      _[1,2]=x
      _[1,3]=0
      _[2,1]=0
      _[2,2]=1
      _[2,3]=x
   [2]:
      1,2
   [3]:
      1
   [4]:
      0
                                    // a more readable way to
> displayHNE(hne);                  //    look at it ;-)
// Hamburger-Noether development of branch nr.1:
HNE[1]=-y+z(0)*z(1)
HNE[2]=-x+z(1)^2

// Hamburger-Noether development of branch nr.2:
HNE[1]=-x+z(0)*z(1)
HNE[2]=-y+z(1)^2
                                    // Caution!
                                    // numbering of branches may
                                    // change when calling
                                    // hnexpansion a 2nd time on
                                    // the same input
```

Usually, we are not interested in the Hamburger-Noether or Puiseux expansion itself, but rather in invariants which can easily be extracted from it. Therefore these invariants are provided by a separate post-processing command:

```
> displayInvariants(hne);
 --- invariants of branch number 1 : ---
 characteristic exponents  : 2,3
 generators of semigroup   : 2,3
```

```
 Puiseux pairs            : (3,2)
 degree of the conductor   : 2
 delta invariant           : 1
 sequence of multiplicities: 2,1,1

 --- invariants of branch number 2 : ---
 characteristic exponents  : 2,3
 generators of semigroup   : 2,3
 Puiseux pairs            : (3,2)
 degree of the conductor   : 2
 delta invariant           : 1
 sequence of multiplicities: 2,1,1

 ------------- contact numbers : -------------

branch |    2
-------+-----
    1  |    1

 ------------- intersection multiplicities : -------------

branch |    2
-------+-----
    1  |    4

 ------------- delta invariant of the curve :  6
```

2.5. *Classification of Hypersurface Singularities*

Sometimes, we also want to check whether a given singularity is in Arnold's list of hypersurface singularities [1]. This test is implemented in SINGULAR as well:

Example 10 Still continuing with the singularity which we have been considering in the previous examples, we now use the Arnold-classifier to determine its type:

```
> LIB "classify.lib";                     // classifier library
> setring r;                              // needs local ring
```

```
                                          // input needs to be
> poly f=I[1];                            //    of type 'poly'
                                          // first guess via
> quickclass(f);                          //    invariants
Singularity R-equivalent to :  Z[k,12k+6r-1]=Z[1,11] or
                               Y[k,r,s]=Y[1,1,1]
Hilbert-Code of Jf^2
We have  2 cases to test
null form
[1]:
    Z[k,12k+6r-1]=Z[1,11] Y[k,r,s]=Y[1,1,1]
[2]:
   2
                                           // following Arnold's
> classify(f);                             //   algorithm
About the singularity :
          Milnor number(f)   = 11
          Corank(f)          = 2
          Determinacy       <= 8
Guessing type via Milnorcode:   Z[k,12k+6r-1]=Z[1,11] or
                                Y[k,r,s]=Y[1,1,1]

Computing normal form ...
   Arnold step number 16
The singularity
   -x2y2+x5+y5-x3y3
is R-equivalent to Y[1,p,q] = T[2,4+p,4+q].
   Milnor number = 11
   modality      = 1
```

2.6. *Monodromy and Spectral Numbers*

Computational Task

Given an element $0 \neq f \in \mathbb{C}\{x_1, \ldots, x_n\}$ (which is assumed to be a polynomial for practical reasons) defining an isolated singularity at the origin, determine the Jordan normal form of the monodromy of f and the spectral numbers of f.

Background

A very brief introduction to these topics an be found in [10], which also provides many references for further reading on various aspects of the topic[k], a more detailed discussion of the definitions and basic statements can be found in the textbook [9], chapter 5. A description of algorithmic and computational methods in this area can be found in [34].

Remark 19. (General Situation) Let $f : (\mathbb{C}^{n+1}, 0) \longrightarrow (\mathbb{C}, 0)$ be a germ of an analytic function defining an isolated hypersurface singularity at the origin. Passing to representatives, let $B_\varepsilon \subset \mathbb{C}^{n+1}$ be a ball of sufficiently small radius $\varepsilon > 0$ around the origin in $\mathbb{C}^{n+1}$ such that it intersects $f^{-1}(0)$ transversally. Let $D_\delta \subset \mathbb{C}$ be a closed disc of sufficiently small radius $\delta > 0$ around the origin in $\mathbb{C}$ such that $f^{-1}(s)$ intersects B_ε transversally for all $s \in D_\delta$. Now set

$$X_s := f^{-1}(s) \cap B_\varepsilon \text{ for all } s \in D_\delta.$$

Milnor (see [33]) showed that in this situation

$$f^{-1}(D_\delta \setminus \{0\}) \longrightarrow D_\delta \setminus \{0\}$$

is a locally trivial differentiable fiber bundle and that the fibers X_s have the homotopy type of a bouquet of $\mu(f)$ n-spheres. Hence for $s \in D_\delta \setminus \{0\}$, $H_0(X_s, \mathbb{Z}) = \mathbb{Z}$ and $H_n(X_s, \mathbb{Z}) = \mathbb{Z}^n$ are the only non-zero homology groups.
Parallel translation along the path

$$\gamma : [0, 1] \longrightarrow D_\delta \quad t \longmapsto \delta \cdot e^{2\pi it}$$

yields a diffeomorphism $h : X_\delta \longrightarrow X_\delta$

Definition 20. (Monodromy) The induced morphism $h_* : H_n(X_\delta, \mathbb{C}) \longrightarrow H_n(X_\delta, \mathbb{C})$ is called the complex monodromy of the singularity.

Theorem 21. *(Monodromy Theorem) The eigenvalues of h_* are roots of unity. The size of the blocks in the Jordan normal form of h_* is at most $(n + 1) \times (n + 1)$.*

Definition 22. (Hodge Filtration and Spectrum) Steenbrink proved in [37] that there is a mixed Hodge structure on the Milnor fiber consisting of an

[k] In this volume there are articles by Ebeling [11] and Steenbrink [39] discussing material from these areas.

increasing weight filtration

$$0 = W_{-1} \subset W_0 \subset \cdots \subset W_{2n} = H^n(X_\delta, \mathbb{Z}) \otimes \mathbb{Q}$$

on $H^n(X_\delta, \mathbb{Z}) \otimes \mathbb{Q}$ and a decreasing Hodge filtration

$$H^n(X_\delta, \mathbb{Z}) \otimes \mathbb{C} = F^0 \supset F^1 \supset \cdots \supset F^n \supset F^{n+1} = 0.$$

To define the spectrum of f, let $p \in \mathbb{Z}, 0 \leq p \leq n$. A rational number $\alpha \in \mathbb{Q}$ with $n - p - 1 < \alpha \leq n - p$ is in the spectrum of f if and only if $e^{2\pi i \alpha}$ is an eigenvalue of the semisimple part of h^* on $F^p H^n(X_\delta, \mathbb{Z})/F^{p+1} H^n(X_\delta, \mathbb{Z})$.

Example 11 Let us consider the example of the isolated hypersurface singularity defined by the polynomial $f = x^5 + y^5 + x^2y^2$. We first want to compute a matrix M such that $e^{-2\pi i M}$ is the monodromy matrix of the given f:

```
> LIB "gmssing.lib";            // monodromy, spectrum etc.
> ring r=0,(x,y),ds;            // as usual: first a 'ring',
> poly f=x5+y5+x2y2;            // then the polynomial
                                // compute data of
> monodromy(f);                 //    the monodromy:
[1]:                            // eigenvalues of M
   _[1]=1/2
   _[2]=7/10
   _[3]=9/10
   _[4]=1
   _[5]=11/10
   _[6]=13/10
[2]:                            // sizes of blocks
   2,1,1,1,1,1
[3]:
   1,2,2,1,2,2                  // multiplicities
```

Therefore, the Jordan normal form of M has the following structure:

$$\begin{pmatrix} \frac{1}{2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{7}{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{7}{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{9}{10} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{9}{10} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{11}{10} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{11}{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{13}{10} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{13}{10} \end{pmatrix}.$$

The same library also provides support for calculation of the spectral numbers of f using standard basis methods for the microlocal structure of the Brieskorn lattice. For details on this algorithmic approach and on more sophisticated data which can also be acquired along these lines see [34].

```
> spectrum(f);                    // compute the spectrum
[1]:                              // spectral numbers
   _[1]=-1/2
   _[2]=-3/10
   _[3]=-1/10
   _[4]=0
   _[5]=1/10
   _[6]=3/10
   _[7]=1/2
[2]:                              // their multiplicities
   1,2,2,1,2,2,1
                                  // pretty printing of
> spprint(_);                     //    previous output:
(-1/2,1),(-3/10,2),(-1/10,2),(0,1),(1/10,2),(3/10,2),(1/2,1)
```

3. Deformations of Singularities

After using computational methods for studying the singular locus of a variety and for determining invariants of isolated singularities, we now turn our interest to families of singularities. More precisely, we first consider the computation of T^1 and T^2 of an isolated singularity and the construction of versal families and then proceed to a more detailed study of certain special families of singularities.

3.1. T^1 *and* T^2

Computational Task

Let $(X,0)$ be a germ of an isolated singularity. Compute the vector space of first order deformations and the $T^2_{X,0}$. Then proceed to determining a versal deformation of the given singularity up to a given degree.

Background

A very accessible introduction to the definitions and properties of T^1, T^2 and versal deformations can e.g. be found in the textbook [7], sections 10.2–10.3.

Lemma 23. *(First Order Embedded Deformations) Let $(X,0)$ be a germ of a complex space, defined by an ideal $I = \langle f_1,\dots,f_r\rangle \subset \mathbb{C}\{x_1,\dots,x_n\}$, let $g_1,\dots,g_r$ be further elements of $\mathbb{C}\{x_1,\dots,x_n\}$ and denote by $(T,0)$ the germ of the fat point corresponding to the ring $\mathbb{C}\{\varepsilon\}/\langle\varepsilon^2\rangle$. Then $f_1+\varepsilon g_1,\dots,f_r+\varepsilon g_r$ defines a flat deformation $(\mathcal{X},0) \longrightarrow (T,0)$ if and only if*

$$f_1 \longmapsto g_1,\dots,f_r \longmapsto g_r$$

yields a well-defined element of $Hom_{\mathbb{C}\{x_1,\dots,x_n\}}(I,\mathbb{C}\{x_1,\dots,x_n\}/I)$.

Definition 24. (Normal Module) The normal module of the singularity $(X,0)$ is the module $Hom_{\mathbb{C}\{x_1,\dots,x_n\}}(I,\mathbb{C}\{x_1,\dots,x_n\}/I)$.

Definition 25. (T^1) Let θ denote the free module of $\mathbb{C}\{x_1,\dots,x_n\}$–derivations. Then the cokernel of the map

$$\begin{aligned} \alpha : \theta &\longrightarrow N_{X,0} \\ \vartheta &\longmapsto (f \mapsto \vartheta(f)) \end{aligned}$$

is denoted by $T^1_{X,0}$.

Lemma 26. *(First Order Deformations) The isomorphism classes of deformations of $(X,0)$ over $(T,0)$ are in one-to-one correspondence to the elements of $T^1_{X,0}$.*

Lemma 27. *(First Order Deformations of an Isolated Singularity) For a germ $(X, 0)$ of an isolated singularity, the $T^1_{X,0}$ is a finite dimensional $\mathbb{C}$ vector space.*

As shorthand notation, we now use $\mathcal{O}$ for $\mathbb{C}\{x_1, \ldots, x_n\}$ and we denote the module of syzygies of I by $\mathcal{R}$ and the submodule of Koszul relations by $\mathcal{R}_0$; that is we have an exact sequence

$$0 \longrightarrow \mathcal{R} \longrightarrow \mathcal{O}^k \longrightarrow \mathcal{O} \longrightarrow \mathcal{O}/I \longrightarrow 0$$

and the submodule $\mathcal{R}_0$ is generated by the relations $f_j \cdot e_i - f_i \cdot e_j$, $1 \leq i, j \leq r$.

Definition 28. (Obstructions) The module $T^2_{X,0}$ is defined as

$$T^2_{X,0} := Hom_{\mathcal{O}}(\mathcal{R}/\mathcal{R}_0, \mathcal{O}_{X,0})/Hom_{\mathcal{O}}(\mathcal{O}^k, \mathcal{O}_{X,0}).$$

Lemma 29. *(Obstructions of an Isolated Singularity) Let $(X, 0)$ be a germ of an isolated singularity, then $T^2_{X,0}$ is a finite dimensional $\mathbb{C}$ vector space.*

Theorem 30. *(Obstructions and Lifting of Deformations) Consider an exact sequence*

$$0 \longrightarrow J \longrightarrow B \longrightarrow A \longrightarrow 0$$

where $(A, \mathfrak{m}_A)$ and $(B, \mathfrak{m}_B)$ are local Artinian $\mathbb{C}$-algebras and $\mathfrak{m}_B J = 0$. Let $(T_A, 0)$ and $(T_B, 0)$ denote the complex space germs corresponding to A and B respectively and let $(\mathcal{X}_A, 0) \longrightarrow (T_A, 0)$ be a flat deformation of $(X, 0)$ (denoted by ξ for short).
There exists a well-defined obstruction element

$$ob(\xi)_{B \longrightarrow A} \in T^2_{X,0} \otimes J.$$

This obstruction element is zero if and only if there exists a flat deformation $(\mathcal{X}_B, 0) \longrightarrow (T_B, 0)$ extending ξ.

This general approach to computing T^1 and T^2 can be rather time consuming; therefore additional information on special cases which allows a more direct calculation should be used whenever available.

Lemma 31. *(T^1 and T^2 of an ICIS[1]) Let $(X, 0)$ be an ICIS. Then all relations are generated by the Koszul relations, i.e. $\mathcal{R} = \mathcal{R}_0$ and hence*

[1] In the case of Cohen-Macaulay codimension 2 singularities, there is a direct method for computing these data, too. Whenever there is such a direct approach, it tends to be much more efficient than the general one and hence should be preferred.

$T^2_{X,0} = 0$.
The normal module is $(\mathcal{O}/I)^r$ *and thus* $T^1_{X,0} = (\mathcal{O}/I)^r / Im(\alpha)$.

Computational Solutions:

Example 12 As a complete intersection example, let us consider the isolated space curve singularity defined by the ideal $I = \langle x^2 + y^2 + z^3, yz \rangle$.

```
> ring r=0,(x,y,z),ds;
> ideal I=x2+y2+z3,yz;               // the singularity
                                     // presentation of the
> def N=I*freemodule(2);             //    normal module
> def T=jacob(I)+N;                  // presentation of T^1

> vdim(std(T));                      // the Tjurina number
6
                                     // base of vector space
> kbase(std(T));                     //    of 1st order deform.
_[1]=z2*gen(1)
_[2]=z*gen(1)
_[3]=gen(1)
_[4]=z*gen(2)
_[5]=x*gen(2)
_[6]=gen(2)
```

This is, of course also available as a SINGULAR command:

```
> LIB "sing.lib";                     // command is in 'sing.lib'
> Tjurina(I);                         // compute T1, ICIS case
// Tjurina number = 6
_[1]=x*gen(1)
_[2]=y*gen(2)+3z2*gen(1)
_[3]=2y*gen(1)+z*gen(2)
_[4]=x2*gen(2)+y2*gen(2)+z3*gen(2)
_[5]=xz*gen(2)
_[6]=z2*gen(2)
_[7]=z3*gen(1)
                                     // base of 1st order
> kbase(std(_));                     //    miniversal deform.
_[1]=z2*gen(1)
_[2]=z*gen(1)
```

```
_[3]=gen(1)
_[4]=z*gen(2)
_[5]=x*gen(2)
_[6]=gen(2)
```

In the general case, however, we cannot avoid computing T^1 and T^2 as described at the the beginning of this section. To illustrate this, we consider the isolated singularity at the origin of the cone over the rational normal curve of degree 4:

Example 13 As an example in the general case, let us consider the isolated singularity defined by the 2-minors of the matrix

$$\begin{pmatrix} x & y & z & u \\ y & z & u & v \end{pmatrix}$$

and compute its T^1 and T^2 using the appropriate built-in commands of SINGULAR.

```
> LIB "sing.lib";                     // T1, T2 are in 'sing.lib'
> ring r1=0,(x,y,z,u,v),ds;           // local ring in 5 var.
> matrix M[2][4] = x,y,z,u,y,z,u,v; // the matrix, see above
> ideal I=minor(M,2);                 // the ideal
> I;
I[1]=-u2+zv
I[2]=-zu+yv
I[3]=-yu+xv
I[4]=z2-yu
I[5]=yz-xu
I[6]=-y2+xz

> T_12(I);                            // compute T1 and T2
// dim T_1 = 4
// dim T_2 = 3
[1]:                                  // standard basis for T^1
   _[1]=gen(8)+2*gen(4)
   _[2]=gen(7)
   _[3]=gen(6)+gen(2)
   _[4]=gen(5)+gen(1)
   _[5]=gen(3)
```

```
   _[6]=x*gen(9)
   _[7]=2x*gen(4)+z*gen(2)
   _[8]=x*gen(2)
   _[9]=x*gen(1)+y*gen(2)
   _[10]=y*gen(9)+z*gen(2)
   _[11]=2y*gen(4)-z*gen(1)+u*gen(2)
   _[12]=y*gen(2)
   _[13]=y*gen(1)+z*gen(2)
   _[14]=z*gen(9)+u*gen(2)
   _[15]=2z*gen(4)-u*gen(9)-u*gen(1)
   _[16]=z*gen(2)
   _[17]=3z*gen(1)-u*gen(2)
   _[18]=u*gen(9)+3u*gen(1)
   _[19]=2u*gen(4)-v*gen(9)-v*gen(1)
   _[20]=u*gen(2)
   _[21]=2u*gen(1)-v*gen(2)
   _[22]=v*gen(9)+v*gen(1)
   _[23]=v*gen(4)
   _[24]=v*gen(2)
   _[25]=v*gen(1)
[2]:                                // standard basis for T2
   _[1]=gen(9)
   _[2]=gen(7)+gen(5)
   _[3]=gen(6)
   _[4]=gen(3)
   _[5]=gen(2)
   _[6]=gen(1)
   _[7]=x*gen(8)
   _[8]=x*gen(5)
   _[9]=x*gen(4)
   _[10]=y*gen(8)-z*gen(5)-u*gen(4)
   _[11]=y*gen(5)+z*gen(4)
   _[12]=y*gen(4)
   _[13]=z*gen(8)
   _[14]=z*gen(5)+u*gen(4)
   _[15]=z*gen(4)
   _[16]=u*gen(8)
   _[17]=u*gen(5)+v*gen(4)
   _[18]=u*gen(4)
```

```
   _[19]=v*gen(8)
   _[20]=v*gen(5)
   _[21]=v*gen(4)
> list li=_;
                                  // basis of finite dim.
> kbase(li[1]);                   // vector space T^1
_[1]=gen(1)
_[2]=gen(2)
_[3]=gen(4)
_[4]=gen(9)
                                  // basis of finite dim.
> kbase(li[2]);                   // vector space T^2
_[1]=gen(4)
_[2]=gen(5)
_[3]=gen(8)
```

These results for the bases of the vector spaces seem rather difficult to interpret at first glance. But as soon as we know the system of generators of the normal module (respectively of $Hom_{\mathcal{O}}(\mathcal{R}/\mathcal{R}_0, \mathcal{O}_{X,0})$) with respect to which the results have been expressed, we have all the data we need. These additional pieces of information can be extracted from the same command by supplying an optional second parameter of arbitrary type. As this creates rather lengthy output (approx. 4 pages in our example), we only state the respective systems of generators: For the normal module the system of generators can be expressed in terms of the perturbations of I defined by

$$\begin{pmatrix}0\\0\\-u\\0\\z\\-y\end{pmatrix}, \begin{pmatrix}0\\-u\\0\\z\\0\\x\end{pmatrix}, \begin{pmatrix}0\\0\\v\\0\\u\\z\end{pmatrix}, \begin{pmatrix}-u\\0\\0\\-y\\-x\\0\end{pmatrix}, \begin{pmatrix}0\\v\\0\\u\\0\\-y\end{pmatrix}, \begin{pmatrix}v\\0\\0\\z\\y\\0\end{pmatrix}, \begin{pmatrix}-u\\-z\\-y\\0\\0\\0\end{pmatrix}, \begin{pmatrix}z\\y\\x\\0\\0\\0\end{pmatrix}, \begin{pmatrix}v\\0\\z\\z\\0\\x\end{pmatrix}$$

where the 1st, 2nd, 4th and 9th form a vector space basis of the T^1 according to the output of our previous computation, namely of `kbase`(li[1]).For the module $Hom_{\mathcal{O}}(\mathcal{R}/\mathcal{R}_0, \mathcal{O}_{X,0})$ the corresponding generators are

$$\begin{pmatrix}0\\0\\x\\0\\0\\y\\0\\z\end{pmatrix}, \begin{pmatrix}0\\x\\0\\0\\y\\0\\z\\-u\end{pmatrix}, \begin{pmatrix}x\\0\\0\\y\\0\\0\\u\\-v\end{pmatrix}, \begin{pmatrix}0\\0\\y\\0\\0\\z\\0\\u\end{pmatrix}, \begin{pmatrix}0\\y\\0\\0\\z\\0\\u\\-v\end{pmatrix}, \begin{pmatrix}y\\0\\u\\z\\0\\v\\v\\0\end{pmatrix}, \begin{pmatrix}0\\0\\z\\0\\0\\u\\0\\v\end{pmatrix}, \begin{pmatrix}0\\z\\u\\0\\u\\v\\v\\0\end{pmatrix}, \begin{pmatrix}z\\-u\\0\\u\\-v\\0\\0\\0\end{pmatrix}.$$

Background (continued)[m]

Definition 32. (Versal Deformation) Let $(\mathcal{X}_S, 0) \longrightarrow (S, 0)$ be a deformation of $(X, 0)$ over a base space $(S, 0)$. This deformation is called versal if for any deformation $(\mathcal{X}_{S'}, 0) \longrightarrow (S', 0)$ of $(X, 0)$ over a base space $(S', 0)$, there exists a map $(S', 0) \longrightarrow (S, 0)$ such that $(\mathcal{X}_{S'}, 0)$ is isomorphic to the pull-back of $(\mathcal{X}_S, 0)$ over this map.

Theorem 33. *(Grauert) Let $(X, 0)$ be a germ of a complex space and suppose that $\dim_{\mathbb{C}} T^1_{X,0} < \infty$. Then there exists an analytic semi-universal deformation of $(X, 0)$.*

Being able to compute T^1 and T^2 explicitly, the natural subsequent step is to ask whether we can also determine versal deformations up to a given degree in practice. The answer is affirmative and the corresponding algorithm is implemented in the library `deform.lib`; a detailed description of the algorithm can be found in [32].

Computational Solutions (continued)

Example 14 (example 13 continued)

```
> LIB "deform.lib";
                                   // compute versal deformation
> list L=versal(I,5);              //    up to degree 5
```

[m] As in the first part of this section, a good reference for a discussion of these topics is e.g. [7], sections 10.2–10.3

```
// ready: T_1 and T_2
// start computation in degree 2.

// 'versal' returned a list, say L, of four rings. In L[1] are stored:
//   as matrix Fs: Equations of total space of miniversal deform.,
//   as matrix Js: Equations of miniversal base space,
//   as matrix Rs: syzygies of Fs mod Js.
// To access these data, type
     def Px=L[1]; setring Px; print(Fs); print(Js); print(Rs);

> L;                              // result is list of rings
[1]:
   //   characteristic : 0
//   number of vars : 9
//        block   1 : ordering ds
//                  : names    A B C D
//        block   2 : ordering ds
//                  : names    x y z u v
//        block   3 : ordering C
[2]:
   //   characteristic : 0
//   number of vars : 9
//        block   1 : ordering ds
//                  : names    A B C D
//        block   2 : ordering ds
//                  : names    x y z u v
//        block   3 : ordering C
// quotient ring from ideal ...
[3]:
   //   characteristic : 0
//   number of vars : 4
//        block   1 : ordering ds
//                  : names    A B C D
//        block   2 : ordering C
[4]:
   //   characteristic : 0
//   number of vars : 9
//        block   1 : ordering ds
```

```
//                   : names    A B C D
//        block   2 : ordering ds
//                   : names    x y z u v
//        block   3 : ordering C
// quotient ring from ideal ...

> def R1=L[1];
> setring R1;                   // go to 1st of returned rings

                                // equations of miniversal
> Js;                           //    base space
Js[1,1]=BD
Js[1,2]=-AD+D2
Js[1,3]=-CD

                                // equations of miniveral
> Fs;                           //    total space
Fs[1,1]=-u2+zv+Bu+Dv
Fs[1,2]=-zu+yv-Au+Du
Fs[1,3]=-yu+xv+Cu+Dz
Fs[1,4]=z2-yu+Az+By
Fs[1,5]=yz-xu+Bx-Cz
Fs[1,6]=-y2+xz+Ax+Cy
```

3.2. *Studying Families of Singularities*

Having constructed versal families in the previous example, we now proceed to study the question of stratifying the base space of a certain classes of families of singularities with respect to the Tjurina number. This question can be dealt with algorithmically for versal families of simple hypersurface and Cohen-Macaulay codimension 2 singularities and for families of semi-quasihomogeneous singularities (again hypersurfaces or CM codimension 2) with fixed initial part. In the first case, it can be used as one ingredient to determining an adjacency to another singularity explicitly[n]; in the second case, it is one step in the construction of moduli spaces for semiquasihomo-

[n]Along these lines it was possible to complete the list of adjacencies for Giusti's list of simple ICIS, see [15].

geneous singularities with fixed initial part (for more details on this topic see e.g. [14]). We only consider the first situation here, as the latter one involves the use of a rather technical modification of the standard basis algorithm.

Computational Task

Given a simple hypersurface singularity $(X, 0)$, determine the stratification of the base space of its versal deformation by the Tjurina number.

Background

The computational tools we are using here are Fitting Ideals and Flattening Stratification. We only state the definitions, a detailed discussion can e.g. be found in [21], sections 7.2–7.3. From the point of view of singularity theory a treatment of the background of the topics discussed in this section can be found in [29].

Definition 34. (Fitting Ideal) Let R be a ring and M a R-module with presentation

$$R^m \xrightarrow{\phi} R^n \longrightarrow M \longrightarrow 0$$

and denote by A a matrix of the map ϕ w.r.t. some chosen bases of R^m and R^n. For $k \in \mathbb{Z}$, define $F_k(M)$ to be the ideal generated by the $(n-k)$–minors of the matrix A, the k-th Fitting ideal[o]

Lemma 35. *(Fitting Ideals) Fitting ideals are independent of the choice of the presentation and of the choice of bases for R^m and R^n. Furthermore, they are compatible with base change.*

Definition 36. (Flattening Stratification) For $r \geq 0$, the locally closed subset

$$Flat_r M := \{P \subset R \text{ prime ideal } | P \supset F_{r-1}(M) \text{ and } P \not\supset F_r(M)\}$$

of $Spec(R)$ is called the flattening stratum of rank r of M. The collection of these strata is called the flattening stratification.

[o]For the remaining values of k the following definitions are used: if $k \geq n$, then $F_k(M) := R$; if $n - k > min\{n, m\}$, then $F_k(M) := 0$.

Definition 37. (The Relative T^1 of the Versal Family) Let the power series $f \in \mathbb{C}\{x_1, \ldots, x_n\}$ define an isolated hypersurface singularity $(X, 0)$, let $g_1, \ldots, g_{\tau_0}$ be generators for $T^1_{X,0}$ and let Φ denote a versal deformation of $(X, 0)$ specified by

$$F = f + \sum_{i=1}^{\tau_0} s_i g_i \in \mathbb{C}\{x_1, \ldots, x_n, s_1, \ldots, s_{\tau_0}\}.$$

Passing to sufficiently small representatives $\mathcal{X}$ of the total space and S of the base space of the deformation, we can consider a family of deformations with section, i.e. a morphism of complex spaces $\Phi : \mathcal{X} \longrightarrow S$ and a section $\sigma : S \longrightarrow \mathcal{X}$ such that $\Phi_s : (\mathcal{X}, \sigma(s)) \longrightarrow (S, s)$ is flat and defines a deformation for all $s \in S$ and such that $(\Phi^{-1}(s), \sigma(s))$ is an isolated singularity. The relative T^1 of this family of deformations can be computed as

$$T^1_{\underline{s}} = \mathbb{C}\{x_1, \ldots, x_n, s_1, \ldots, s_{\tau_0}\}/\langle F, \frac{\partial F}{\partial x_1}, \ldots \frac{\partial F}{\partial x_n}\rangle.$$

Remark 38. Note that fixing a value for the parameters $s_1, \ldots, s_{\tau_0}$, the $\mathbb{C}$ vector space dimension of the stalk of $T^1_{\underline{s}}$ at this points of S provides the Tjurina number of the respective isolated hypersurface singularity.

Computational Solution

Example 15 To keep the calculations as simple as possible, we only consider a very small but well-known example, an A_3–singularity.[p] We first compute a versal family by means of calculation of a vector space basis for the T^1 (Tjurina algebra) and the relative T^1 of this family:

```
> ring r=0,(x,y),ds;
> poly f=x^4+y^2;                      // the singularity
> ideal kb=kbase(Tjurina(f));          // vector space basis for T1
> kb;
kb[1]=x2
kb[2]=x
kb[3]=1
                                       // move to suitable ring
```

[p] As these calculations involve ideals generated by minors of a matrix and as this matrix is, in general, not of a simple structure, the computations tend to become very lengthy and this approach should hence be considered as a brute force approach which should only be used in combination with the use of all additional information that could possibly lower the complexity.

```
> ring rt=0,(a,b,c,x,y),ds;         //    for total space
> poly F=x^4+y^2+a+b*x+c*x^2;       // versal family

> ideal jF=diff(F,x),diff(F,y),F;   // presentation of rel. T1
                                    //    but as module over rt,
                                    //    we need it as
                                    //    C[a,b,c]-module
> jF;
jF[1]=b+2cx+4x3
jF[2]=2y
jF[3]=a+bx+y2+cx2+x4
```

We know that jF is a finitely presentable $K[a,b,c]$ module. As f is a hypersurface singularity, we can determine the corresponding presentation matrix by looking at the Euler relation and suitable products of it with monomials in x and y. (In this example only the products with x and x^2 are relevant.)

```
                                    // suitable ring for
                                    //    finding Euler rel.
> ring rg=0,(x,y,a,b,c),(dp(2),dp); //    (Q[a,b,c])[x,y]

> def jF=imap(rt,jF);               // fetch jF from rt
> jF=mstd(jF)[2];                   // find minial system of
                                    //    generators for jF
> jF;                               // look at jF
jF[1]=y
jF[2]=4x3+2xc+b
jF[3]=2x2c+3xb+4a                   // <-- Euler relation

> matrix Tmat[3][3];
> def tempmat=coef(jF[3],xy);       // give temporary name,
                                    //    because lists can
                                    //    only be formed
> Tmat[1,1..3]=tempmat[2,1..3];     //    from named objects
> tempmat=coef(reduce(jF[3]*x,jF[2]),xy);
> Tmat[2,1..3]=tempmat[2,1..3];
> tempmat=coef(reduce(jF[3]*x^2,jF[2]),xy);
> Tmat[3,1..3]=tempmat[2,1..3];
> print(Tmat);                      // presentation matrix of T1
```

```
                                        // as Q[a,b,c]-module
2c,    3b,    4a,
3b,    -c2+4a,-1/2bc,
-c2+4a,-2bc,  -3/4b2
```

The strata of constant Tjurina number can now be obtained by means of the flattening stratification of the relative T^1. This implies that we need to determine the Fitting ideals of the module - that is we need to determine the minors of size one, two and three:

```
> ideal min1=mstd(minor(Tmat,1))[2];  // minimal set of gen.
> min1;                               // for 1-minors
min1[1]=c
min1[2]=b
min1[3]=a
> ideal min2=mstd(minor(Tmat,2))[2];  // dito for 2-minors
> min2;
min2[1]=2c3+9b2-8ac
min2[2]=bc2+12ab
min2[3]=3b2c-8ac2+32a2
> ideal min3=mstd(minor(Tmat,3))[2];  // and for 3-minors
> min3;
min3[1]=4b2c3-16ac4+27b4-144ab2c+128a2c2-256a3
```

From this computation, we can see that the maximal value of the Tjurina number is attained exactly for the fiber over the point $V(a,b,c)$ of the base. For fibers over points outside of $V(4b^2c^3 - 16ac^4 + 27b^4 - 144ab^2c + 128a^2c^2 - 256a^3)$, which is the swallowtail singularity (cf. figure 1), on the other hand there are no singularities. The Tjurina number is 2 for points in $V(\texttt{min2}) \setminus V(a,b,c)$ (cf. figure 2). It is 1 for points in $V(\texttt{min3}) \setminus V(\texttt{min2})$, i.e. points on the swallowtail which do not lie on the curve $V(\texttt{min2})$.

4. Varieties with Singularities

In this last section, we consider two areas of more complex applications of computational methods in singularity theory: the task of finding hypersurfaces with prescribed singularities and the task of resolution of singularities. In the first case, the goal is more of theoretical nature and explicit calculations are basically used to check whether certain conditions are satisfied or for finding good examples which show certain properties. In the second

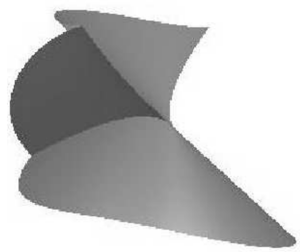

Figure 1. The swallowtail singularity: $V(\texttt{min3})$.

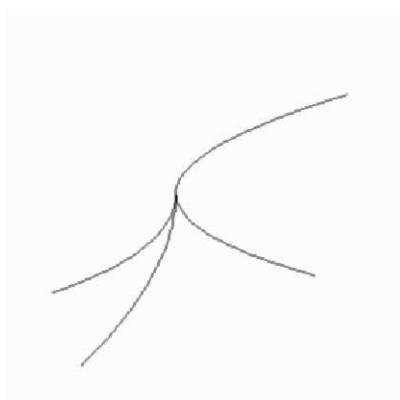

Figure 2. The singular locus of the swallowtail singularity: $V(\texttt{min2})$.

case, the set-up is different: The task itself is computational, but it consists of many different computational aspects each of which needs to be treated carefully in order to obtain a usable implementation.

4.1. *Hypersurfaces with Prescribed Singularities*

Here, we briefly sketch two applications of computer algebra tools in this area: first we treat the question of finding an upper bound for how many singularities of a given type can fit on a hypersurface of a given degree, then we outline how computational tools aided in the search for examples of surfaces of fixed degree with a high number of double points.

Computational Task

Given a hypersurface of fixed degree and a singularity type, give an upper bound for the number of singularities of this type which can appear on such a hypersurface.

Background

In this application we use another property of the singularity spectrum. Recall that the spectrum was already considered in section 2.6; the property which we are going to use now was proved by Steenbrink in [38].

Lemma 39. *(Properties of the Spectrum) The spectrum of an isolated hypersurface singularity is constant under μ-constant deformations.*
The number of spectral numbers in a half open interval $(a, a+1]$ is uppersemicontinuous under small deformations of isolated hypersurface singularities; for semi-quasihomogeneous isolated hypersurface singularities the same property also holds for intervals $(a, a+1)$.

Computational Solution

Example 16 The question, which we are treating in this example, is the following: What is the maximal number of singularities of type $T_{3,3,3}$ that can occur on a surface of degree 7 in $\mathbb{P}^3$?
Let us fist recall that the singularities of type $T_{3,3,3}$ form a μ-constant 1-parameter family given by equations of the kind

$$x^3 + y^3 + z^3 + t \cdot xyz = 0, \quad \text{where} \quad t^3 \neq -27.$$

To obtain the desired bound, we now use the semicontinuity property of the spectrum. More precisely, the number of spectral numbers of the singularities of a deformation of a given hypersurface in an interval $(a, a+1]$ cannot exceed the number of spectral numbers of the original singularity in this interval; for semiquasihomogeneous singularities the same statement also holds for the intervals $(a, a+1)$.

```
> LIB "gmssing.lib";              // spectrum related commands
> ring R=0,(x,y,z),ds;            // local ring in 3 variables
> poly f=x^3+y^3+z^3;             // a singularity of type T_333
> list s1=spectrum(f);            // compute its spectrum
```

```
> s1;
[1]:                                  // spectral numbers
   _[1]=0
   _[2]=1/3
   _[3]=2/3
   _[4]=1
[2]:                                  // multiplicities
   1,3,3,1
                                      // any surface of degree 7 is
> poly g = x^7+y^7+z^7;               // deformation of this surface
> list s2 = spectrum(g);              // compute its spectrum
                                      //   * takes some time!!
> s2;
[1]:                                  // spectral numbers
   _[1]=-4/7
   _[2]=-3/7
   _[3]=-2/7
   _[4]=-1/7
   _[5]=0
   _[6]=1/7
   _[7]=2/7
   _[8]=3/7
   _[9]=4/7
   _[10]=5/7
   _[11]=6/7
   _[12]=1
   _[13]=8/7
   _[14]=9/7
   _[15]=10/7
   _[16]=11/7
[2]:                                  // multiplicities
   1,3,6,10,15,21,25,27,27,25,21,15,10,6,3,1
> spsemicont(s2,list(s1));            // checking semicont.condition
[1]:
   18
> spsemicont(s2,list(s1),1);          // checking sqh.semicont.cond.
[1]:
   17
```

Thus a septic in $\mathbb{P}^3$ can at most contain 17 singularities of type $T_{3,3,3}$.

A Non-Computational Task Tackled by Computational Means

On the other hand, computer algebra methods have recently been successfully used by O. Labs and D. van Straten to construct a septic with 99 nodes (see figure 3 for a picture of the singularity, [28] for details on the approach).[q] The basic idea behind the approach of Labs and van Straten is the following: They start with a 7-parameter family of septics and develop conditions to easily determine the number of nodes on a given septic from a 5-parameter subfamily of this family. Then they pass to small prime fields (with primes $11 \leq p \leq 53$) and explicitly check the actual number of nodes on the septic for all possible parameter combinations to obtain those which provide exactly 99 nodes. Further geometric considerations in characteristic zero lead to a condition for the parameters which can be described as the zero locus of a single univariate polynomial of degree 150, which is, of course, still too large to be of any direct use. Therefore they factorize the polynomial and plug into each of the factors the solutions which were previously obtained over the small prime fields. This leads to only one factor of degree 3 whose vanishing locus contains one real solution; it can then be checked by explicit calculation that the surface corresponding to this parameter value has precisely 99 nodes and no other singularities.

4.2. *Resolution of Singularities*

The last computational aspect which we want to consider is how to tackle more complex algorithmic tasks, in this case the task of resolution of singularities. As the series of talks of H. Hauser at this school was devoted to the theoretical background of this topic, we only recall the most important definitions and statements in section 4.2.1 before considering the practical side of it.

[q] Up to degree 6 the maximal number of nodes on a surface is known, that is there are known examples possessing exactly the number of nodes specified by an upper bound. In degree 7, however, Varchenko's spectrum bound and Giventhal's bound both lead to an upper bound of 104 for the number of nodes on a septic, the septic with the highest number of nodes that had been known prior to the example of Labs and van Straten had 93 nodes.

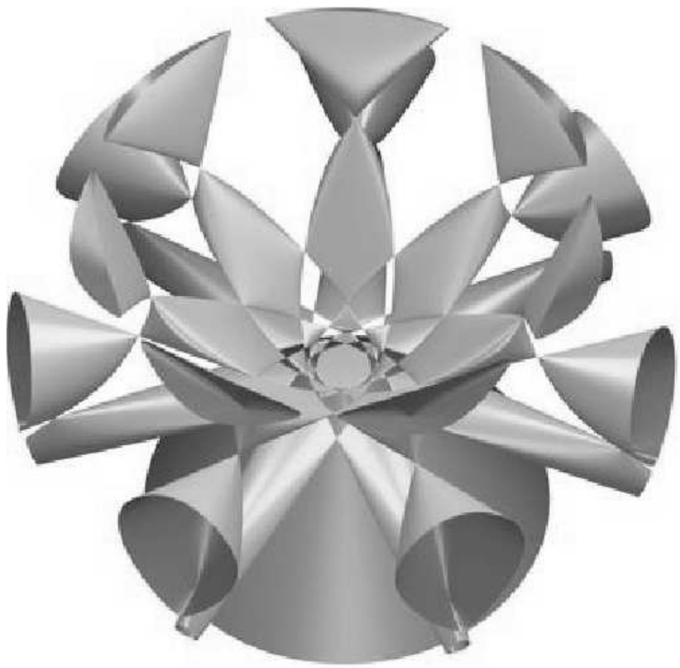

Figure 3. The septic surface with 99 real nodes found by O. Labs and D. van Straten.

4.2.1. *Theoretical Background*

The existence of an embedded resolution of singularities in characteristic zero has been proved by H. Hironaka in his article [25] in 1964. But it took another 25 years, until algorithmic approaches to this task were found independently by E. Bierstone and P. Milman and by the group of O. Villamayor (see e.g. [2] and [3] for recent articles on these approaches). For recent and accessible introductions to the topic, the article [24] of H. Hauser, which contains many illustrations, and the notes of the Seattle lecture of J. Kollar [26] are good references.

Theorem 40. *(Embedded Resolution of Singularities) Let W be a smooth algebraic scheme (over a field K of characteristic zero) and let X be a subscheme (with ideal sheaf $\mathcal{I}_X \subset \mathcal{O}_W$). There exists a sequence of*

$$W = W_0 \stackrel{\pi_1}{\longleftarrow} W_1 \stackrel{\pi_2}{\longleftarrow} \cdots \stackrel{\pi_r}{\longleftarrow} W_r$$

of blow-ups $\pi_i : W_i \longrightarrow W_{i-1}$ at smooth centers $C_{i-1} \subset W_{i-1}$ such that

(a) The exceptional divisor of the induced morphism $W_i \longrightarrow W$ *has only normal crossings and* C_i *has normal crossings with it.*
(b) Let $X_i \subset W_i$ *be the strict transform of* X*. All centers* C_i *are disjoint from* $Reg(X) \subset X_i$*, the set of points where* X *is smooth.*[r]
(c) X_r *is smooth and has normal crossings with the exceptional divisor of the morphism* $W_r \longrightarrow W$*.*
(d) The morphism $(W_r, X_r) \longrightarrow (W, X)$ *is equivariant under group actions.*

Considering the above theorem from a computational point of view, we immediately see two central tasks: the calculation of the blowing up which will be discussed in section 4.2.2 and the choice of the centers which is by far the more difficult task and will be dealt with in sections 4.2.3–4.2.4. Further computational tasks are added due to the fact that in this context we are not dealing with affine varieties, but with algebraic varieties or schemes. This implies, in particular, that we cannot assume that the whole situation can be encoded (for computational purposes) by means of an ideal/ideals in one polynomial ring; instead we may only assume that W is specified by means of affine charts in each of which we can again work with ideals in a polynomial ring[s]. This poses the problem of gluing and identification of objects that appear in more than one chart; section 4.2.5 discusses these practical aspects.

The central point of desingularization by a sequence of blow ups is the appropriate choice of the centers. Therefore algorithms for resolution of singularities are often just stated as algorithms for the choice of centers. This choice is controlled by assigning a rather elaborate invariant value (from a totally ordered set) to each point of X; the locus of maximal value is then used as the subsequent center. The rather complicated structure of such invariants is a consequence of the facts that the center has to satisfy the properties which were stated in the above theorem (e.g. it has to be a closed set, normal crossing with the exceptional divisors) and that the improvement of the situation has to be measured by the invariant, eventually ensuring termination of the algorithm. The definition of the invariant and

[r]This is not a typographical error, it is really $Reg(X)$, not $Reg(X_i)$. This condition simply ensures that the sequence of blow-ups is an isomorphism on $Reg(X)$.

[s]A priori, we can make sure that each of the charts of W simply looks like an $\mathbb{A}^k$. But it will turn out in section 4.2.2 that, after blowing up, it might be necessary to consider charts in which we do no longer have this special structure. Nevertheless, each of these charts can still be expressed as an affine variety in some larger $\mathbb{A}^N$, a fact which will be used in the subsequent sections.

the computation of its maximal locus is based on an inductive construction which first assigns a part of the invariant to the given point and then constructs an auxiliary object embedded in some ambient space of smaller dimension such that a value has already been assigned to the corresponding point there by induction hypothesis.
As already mentioned at the beginning of this section, there are currently two main kinds of algorithms, the ones based on the work of E. Bierstone and P. Milman, which are very close to the original proof of Hironaka, and the ones based on Villamayor's approach, which is more accessible to practical calculations. The latter point of view is the one we shall use here. Its basic idea is to achieve a reduction of the order of a given ideal by means of order reductions of auxiliary ideals. Before we discuss the underlying notions and the invariant in detail, we now give an example of the difficulties:

Example 17 Consider $\mathcal{I}_X = \langle z^2 - x^2y^2 \rangle \subset \mathbb{C}[x,y,z] = \mathcal{O}_{\mathbb{A}^3_{\mathbb{C}}}$. The singular locus of X can easily be determined to be $V(z,xy)$, i.e. the union of the x- and the y-axis, which is clearly singular itself. As there is nothing special about either of the lines, any choice between the two would be completely at random. Hence the only natural choice for a center here is the coordinate origin $V(x,y,z)$. After blowing up, however, we are facing one chart in which there are no singularities and two other charts (which look identical) in which we encounter the original situation with just one small change: the presence of an exceptional divisor which contains one of the lines (allowing a choice between the two lines).

As the previous example shows, the data we need to consider does not only consist of $\mathcal{I}_X$ and the ambient space W, we also need to take into account the set E of exceptional divisors, which we assume to be ordered chronologically according to the moment of birth of the respective divisors in the resolution process. The last piece of data which is added is an integer b, which by default states the maximal order[t] of the ideal $\mathcal{I}_X$. In the induction step, the construction of auxiliary objects, however, it is usually a different value is assigned.

[t]An exact definition of the order of an ideal will be given below. For now it is sufficient to see the order of an ideal as the generalization of the order of a power series.

Definition 41. (Basic Objects) A collection of data $(W, \mathcal{I}_X, b, E)$ as described above is called a basic object[u].

Definition 42. (Resolution Algorithm) Let $(\mathfrak{I}, \leq)$ be a totally ordered set. A family of functions[v]

$$f_{(W,\mathcal{I}_X,b,E)} : X \longrightarrow \mathfrak{I},$$

which is equivariant under isomorphism of basic objects, is said to be governing a blow up[w] $\pi : (W_1, \mathcal{I}_{X_1}, b, E_1) \longrightarrow (W_0, \mathcal{I}_{X_0}, b, E_0)$, if the following conditions hold:

(a) The set of points $\mathrm{Max} f_{(W_0,\mathcal{I}_{X_0},b,E_0)} \subset X_0$, where $f_{(W_0,\mathcal{I}_{X_0},b,E_0)}$ takes its maximal value $\max f_{(W_0,\mathcal{I}_{X_0},b,E_0)}$, is a closed subset of W_0.
(b) $\mathrm{Max} f_{(W_0,\mathcal{I}_{X_0},b,E_0)}$ is a permissible center, that is, it is regular, has normal crossings with E_0 and is disjoint from the set of points[x] $\{x \in X_0 | x \notin Sing_b(X_0), x \notin E_{0_i} \ \ \forall 1 \leq i \leq \#E_0\}$.
(c) $\mathrm{Max} f_{(W_0,\mathcal{I}_{X_0},b,E_0)}$ is the center of the blow-up π.
(d) $\max f_{(W_0,\mathcal{I}_{X_0},b,E_0)} > \max f_{(W_1,\mathcal{I}_{X_1},b,E_1)}$.
(e) (Compatibility with open restrictions)

$$f_{(W_0,\mathcal{I}_{X_0},b,E_0)}(x) = f_{(W_1,\mathcal{I}_{X_1},b,E_1)}(x)$$

for each point $x \in X_0 \backslash \mathrm{Max} f_{(W_0,\mathcal{I}_{X_0},b,E_0)}$, where $x \in X_1$ is identified with the corresponding point in X_0 by means of the fact that the blow-up is an isomorphism outside the center.

An algorithm of resolution of basic objects consists of such a family of functions dictating the subsequent blow-ups for any given basic object $(W, \mathcal{I}_X, b, E)$ subject to the additional conditions that

(a) (Termination of the algorithm)
There is an index N depending on the basic object such that the object is resolved after N steps.

[u] Other notions for these types of collections of data include Marked Ideal and Presentation.
[v] By the notation $f_{(W,\mathcal{I}_X,b,E)}$ we want to emphasize that the function itself depends on the whole basic object, not just on X, although it assigns values to the points of X.
[w] The precise definition of the transform of a basic object under a blow up can be found at the end of the next section.
[x] See the subsequent definition of the order of an ideal and of the b-singular locus for an explanation of the notion $Sing_b(X_0)$.

(b) If X_0 is a regular pure-dimensional subscheme of dimension r, $b = 1$ and $E_0 = \emptyset$, then there is a value $\mathfrak{s}(r) \in \mathfrak{I}$ such that $f_{(W_0,X_0,b,E_0)}(x) = \mathfrak{s}(r)$ for all $x \in X_0$.

Definition 43. (Order of an Ideal and b-Singular Locus) The order of an ideal $\mathcal{I} \subset \mathcal{O}_{W,x}$ at a point $x \in X$ is defined as

$$ord_x(\mathcal{I}) := \max\{m \in \mathbb{N} | \ \mathcal{I} \subset \mathfrak{m}_{W,x}^m\}.$$

$Sing_b(X)$, the b–singular locus of a basic object[y] $(W, \mathcal{I}_X, b, E)$ is defined as the closed set of all points at which the order of the ideal of X is at least b.

Lemma 44. *(Semicontinuity of the Order) The order of an ideal at a point is infinitesimally upper-semicontinuous and Zariski-upper-semicontinuous, that is it does not increase under blow ups and in a sufficiently small Zariski open neighborhood of a point there are no points at which the ideal has a higher order.*

Using this definition it is now possible to state the first two entries of Villamayor's invariant for a given basic object $(W, \mathcal{I}_X, b, E)$:

$$f_{\mathrm{trunc}}(x) = (ord_x(\mathcal{I}), N_E(x))$$

where $N_E(x)$ is an integer counting the exceptional divisors containing x which have been born before the order at x attained its current value. To continue the definition of the invariant, we need to pass to an auxiliary basic object in an ambient space of lower dimension to apply the induction hypothesis there. This new lower dimensional ambient space is a so-called hypersurface of maximal contact. A detailed discussion of its construction and properties may be found in [19], a slightly different point of view is outlined in [26].

Definition 45. (Hypersurface of Maximal Contact) Given a basic object $(W, \mathcal{I}_X, b, E)$, where b is the maximal order of $\mathcal{I}_X$, a smooth hypersurface $Z \subset W$ is called a hypersurface of maximal contact, if

(a) for every open set $U \subset W$, the locus of maximal value of the truncated governing function is contained in $Z|_U$,
(b) for every open set $U \subset W$ and every sequence of blow ups at centers of maximal value of the truncated governing function starting at the basic object $(U, \mathcal{I}_X|_U, b, E|_U)$ the center of every blow up is again contained in the respective strict transform of $Z|_U$,

[y]This is, of course, the concept of the idealistic exponent of Hironaka.

(c) Z has transversal intersections with each exceptional divisor which arose after the maximal order dropped to the current value,
(d) the set $\{E_i \cap Z | E_i \in E$ born after maximal order dropped to $b\}$ is normal crossing.

These conditions seem to be rather strange at first glance. But conditions (a) and (b) simply ensure that we do not loose any points of maximal value of the truncated governing function when passing to the hypersurface of maximal contact and that we do not need to choose a new hypersurface as long as our maximal value does not drop. Conditions (c), on the other hand, ensures that the centers determined by means of passing to the hypersurface Z are also permissible as centers for the given basic object, whereas condition (d) is ensuring the normal crossing condition after passing to Z. Note that although (c) and (d) are very similar, neither one implies the other.[z]

Remark 46. (Existence of Hypersurface of Maximal Contact) A hypersurface of maximal contact does not always exist globally; locally, however, it does. One of the central points in the proof of algorithmic desingularization is the independence of the construction of the (local) choice of the hypersurfaces of maximal contact.

By the construction of the coefficient ideal, which will be discussed in section 4.2.4, it is now possible to obtain an auxiliary basic object, using the hypersurface of maximal contact as the new ambient space, and determine the values of the governing function for this object. As the construction of the auxiliary object is rather technical and involves notions that will be considered in detail in the following sections, we do not state the definition of the coefficient ideal yet. Marking the descent in dimension of the ambient space by ';', we can now (at least) state the general structure of Villamayor's invariant[A] for a given basic object $\mathcal{B} = (W, \mathcal{I}_X, b, E)$ and its

[z]To see that conditions (c) and (d) are truly different, consider the following two examples: For the set of exceptional divisors $\{V(x), V(y)\}$ in $\mathbb{A}^3_{\mathbb{C}}$ and the hypersurface $Z = V(x + y)$ (c) is satisfied, but (d) obviously fails, since the two $E_i \cap Z$ coincide; for the set of exceptional divisors $\{V(x-1), V(y+1)\}$ and the hypersurface $Z = V(x^2+y^2-1)$ condition (d) is satisfied in a trivial way as the $E_i \cap Z$ do not meet, but of course each of the two exceptional divisors is tangent to Z contradicting to condition (c).

[A]In fact, we omit one special case here, the so-called monomial case which is solved by a separate algorithm of combinatorial nature and does not pose any special computational problems. It can, for instance, be found in [3].

auxiliary objects $\mathcal{B}_{dimW-1}, \ldots, \mathcal{B}_2$:

$$f_{\mathcal{B}}(x) = (f_{\mathcal{B},\text{trunc}}(x); f_{\mathcal{B}_{dimW-1},\text{trunc}}(x); \ldots; f_{\mathcal{B}_2}(x)).$$

From the computational point of view, these considerations show that the choice of the center can be split up into several subtasks:

- Computation of the locus of maximal order of a given basic object
- Computation of the locus of maximal N_E inside the locus of maximal order
- Descent in dimension and construction of the auxiliary basic object

While the second task is straight forward, the other two require several non-trivial considerations before an efficient implementation is possible. Therefore section 4.2.3 is devoted to the first subtask and the subsequent one 4.2.4 to the third.

Based on the implementation of the resolution process, it is also possible to determine resolution related invariants explicitly. As an example, we discuss how to determine the intersection matrix of the exceptional divisors in a (non-embedded) resolution of a surface. For a detailed discussion of practical aspects of other applications see [16].

4.2.2. *Blowing Up*

Computational Task

Given a basic object $(W, \mathcal{I}_X, b, E)$ and a permissible center C of a blow up, determine the total/weak/strict transform of it under this blow up.

Background

Since we need to blow up at general smooth centers, not just at points, we would like to recall the definitions and basic properties of blow ups (for a detailed discussion see e.g. [23] section II.7).

Definition 47. (Blowing up) Let W be a scheme, $Y \subset W$ a subscheme corresponding to the coherent ideal sheaf $\mathcal{J}$. The blowing up of W with center Y is

$$\pi : \tilde{W} := Proj(\bigoplus_{d \geq 0} \mathcal{J}^d) \longrightarrow W.$$

The notations W, $\tilde{W}$, Y and $\mathcal{J}$ stay the same as in the previous definition for the rest of this section on the mathematical background.

Lemma 48. *(Basic Properties of Blowing Up) Let* $\pi : \tilde{W} \longrightarrow W$ *be the blow up of* W *at a center* Y*. Then* $\pi^{-1}\mathcal{J}\mathcal{O}_{\tilde{W}}$ *is an invertible sheaf on* $\tilde{W}$*; the corresponding subscheme of* $\tilde{W}$ *is called the exceptional divisor of the blow up.*
$\pi : \pi^{-1}(W \setminus Y) \longrightarrow W \setminus Y$ *is an isomorphism.*

Lemma 49. *(Universal Property of Blowing Up) If* $f : Z \longrightarrow W$ *is any morphism such that* $f^{-1}\mathcal{J}\mathcal{O}_Z$ *is an invertible sheaf on* Z*, then there exists a unique morphism* $g : Z \longrightarrow \tilde{W}$ *factoring* f*.*

Lemma 50. *(Strict Transform) Let* $Z_1 \overset{i}{\hookrightarrow} W$ *be a closed subscheme. Let* $\pi_1 : \tilde{Z}_1 \longrightarrow Z_1$ *be the blow up of* Z_1 *along* $i^{-1}\mathcal{J}\mathcal{O}_{Z_1}$*. Then the following diagram commutes*

$$\begin{array}{ccc} \tilde{Z}_1 & \hookrightarrow & \tilde{W} \\ \pi_1 \downarrow & & \downarrow \pi \\ Z_1 & \hookrightarrow & W. \end{array}$$

$\tilde{Z}_1$ *is called the strict transform of* Z_1 *under the blow up* $\pi : \tilde{W} \longrightarrow W$*, whereas* $\pi^*(Z_1)$ *is called the total transform of* Z_1 *under this blow up.*

On the other hand, we shall need some standard techniques from computer algebra which are discussed in detail in [21], section 1.8.

Lemma 51. *(Kernel of a Ring Homomorphism, Simplest Case) Let* $\phi : K[x_1, \ldots, x_n] \longrightarrow K[y_1, \ldots, y_m]$ *be a morphism of rings over a field* K *specified by* $\phi(x_i) = f_i \in K[y_1, \ldots, y_m]$ *for* $1 \leq i \leq n$*. Then the ideal* $Ker(\phi) \subset K[x_1, \ldots, x_n]$ *can be determined by elimination of the variables* $y_1, \ldots, y_m$ *from the ideal* $\langle x_1 - f_1, \ldots, x_n - f_n \rangle$*.*

Definition 52. (Saturation) Let $I_1, I_2 \in K[x_1, \ldots, x_n]$ be ideals. The ideal quotient of I_1 by I_2 is defined as $(I_1 : I_2) := \{g \in K[x_1, \ldots, x_n] | gI_2 \subset I_1\}$. Iterating the operation of taking the ideal quotient, we obtain an ascending chain of ideals

$$I_1 \subset (I_1 : I_2) \subset (I_1 : I_2^2) \subset \ldots$$

which eventually stabilizes, since the polynomial ring is noetherian.

$$(I_1 : I_2^\infty) := \bigcup_i \geq 0 (I_1 : I_2^i)$$

is called the saturation of I_1 w.r.t. the ideal I_2. Obviously, it coincides with $(I_1 : I_2^s)$ for all sufficiently large s.

Lemma 53. *(Computation of Ideal Quotients) Let $I_1 \subset K[x_1, \ldots, x_n]$ and let $0 \neq h \in K[x_1, \ldots, x_n]$. Denoting a set of generators for $I_1 \cap \langle h \rangle$ by $g_1 \cdot h, \ldots, g_r \cdot h\rangle$, $(I_1 : \langle h \rangle) = \langle g_1, \ldots, g_r \rangle$.*
For a non-principal ideal I_2, this approach can be generalized by choosing a set of generators $h_1, \ldots, h_m$ for I_2:

$$(I_1 : I_2) = \bigcap_{i=1}^{m} (I_1 : \langle h_i \rangle).$$

Saturation can then be computed by iterating the operation of taking ideal quotients and checking for stabilization.

Computational Approach

When dealing with explicit examples, it is usually more convenient to pass to a covering by affine charts. In particular, the calculations of the blow ups can be formulated from this point of view allowing a direct implementation.

Let $U \subset W$ be an affine open subset and and denote $\Gamma(U, \mathcal{O}_W)$ by A and $\Gamma(U, \mathcal{J}) = \langle f_1, \ldots, f_m \rangle \subseteq A$ by J. Then the blowing up of U at the center $Y \cap U$ is

$$\pi^{-1}(U) = Proj(\bigoplus_{d \geq 0} J^d).$$

To compute this blow up explicitly, we consider the canonical graded A–algebra homomorphism

$$\Phi : A[y_1, \ldots, y_m] \longrightarrow \bigoplus_{n \geq 0} J^n t^n \subseteq A[t]$$

defined by $\Phi(y_i) = tf_i$. Then $\oplus_{n \geq 0} J^n$ is obviously isomorphic to the ring $A[y_1, \ldots, y_m]/Ker(\Phi)$ and we can hence describe the situation by means of the embedding $\pi^{-1}(U) \cong V(Ker(\Phi)) \subseteq Spec(A) \times \mathbb{P}^{m-1}$. In particular, $\pi^{-1}(U)$ can again be covered by affine charts $D(y_i)$ (each of which is, of course, precisely the complement of the vanishing locus of the corresponding variable y_i).

For simplicity of presentation, we assume from now on that we are only dealing with the situation $W \subset \mathbb{A}^N$ for some $N \in \mathbb{N}$, by passing to an affine covering as described above and considering each of these charts separately. Let $I \subset \mathcal{O}_W$ be an ideal and X the corresponding subscheme of W. Then the exceptional divisor and the different transforms of X under the blow up of W at center Y can be computed in the following way:

exceptional divisor	$I(H) = J\mathcal{O}_{\tilde{W}}$
total transform	$\pi^*(I) = I\mathcal{O}_{\tilde{W}}$
strict transform	$I_{\tilde{X}} = (I\mathcal{O}_{\tilde{W}} : J\mathcal{O}_{\tilde{W}}{}^{\infty})$
weak transform	$(I\mathcal{O}_{\tilde{W}} : J\mathcal{O}_{\tilde{W}}{}^{k})$
	where $k = max\{l \in \mathbb{N} \| (I\mathcal{O}_{\tilde{W}} : J\mathcal{O}_{\tilde{W}}{}^{l-1})$
	$= (I\mathcal{O}_{\tilde{W}} : J\mathcal{O}_{\tilde{W}}{}^{l}) \cdot J\mathcal{O}_{\tilde{W}}\}$
controlled transform	(w.r.t. a control c)
	$(I\mathcal{O}_{\tilde{W}} : J\mathcal{O}_{\tilde{W}}{}^{c})$

At this point, we are now ready to state how the transform $(W_1, \mathcal{I}_{X_1}, b, E_1)$ of a basic object $(W_0, \mathcal{I}_{X_0}, b, E_0)$ under a blow up $\pi : (W_1, \mathcal{I}_{X_1}, b, E_1) \longrightarrow (W_0, \mathcal{I}_{X_0}, b, E_0)$ is defined: $W_1 = \tilde{W}_0$, $\mathcal{I}_{X_1}$ is the ideal of the weak[B] transform of $\mathcal{I}_{X_0}$, b remains unchanged and E_1 is the union of the strict transforms of the exceptional divisors from E_0 and the new exceptional divisor.

Obviously, the difficulty of the computation of the blow up, which is a preimage computation, depends very much on the generators g_i of the center and on the total number of variables involved, because in the very heart of the computation there is an elimination, that is a Gröbner basis computation in $n + s + 1$ variables w.r.t. an elimination ordering for t. In particular, this causes successive blowing-ups in smooth irreducible centers to be by far less expensive than blowing-up at several smooth (disjoint) irreducible centers simultaneously. Therefore it is usually a good idea to apply primary decomposition of the center and then blow up at each of the components separately. Clearly, this is possible because, a blow-up is an isomorphism outside of the center and because the components of the non-singular center are obviously disjoint. The draw-back of this improvement is the fact that more charts are produced and hence more duplicate calculations can occur in future steps of the resolution process; but this trade-off still pays off in a very large number of practical applications.

Another enhancement to the resolution process follows from the fact that not all s charts arising from a single blow-up contain new information. It may very well happen that in one or more charts we do not see any new information that is not already provided by the other charts. In this case,

[B]This is the case in the algorithms following Villamayors approach, in the approach of Bierstone and Milman strict transforms are used at this point. Although we are not discussing the latter algorithm here, this principal difference needs to be mentioned.

such charts may be dropped. We have been careful not to state what the relevant information is in the previous phrase, because that can depend very much on the data that is to be computed from the resolution: If the goal is, e.g., to compute the intersection matrix of the exceptional divisors of a resolved surface, no relevant information can be provided by data at points outside of the strict transform of the surface. If on the other hand, the goal is the computation of a ζ-function, the information on the intersections of the exceptional divisors (even outside the weak transform of the variety) is vital and no charts may be dropped.

4.2.3. *Computing the Locus of Maximal Order*

Computational Task

Given a basic object $(W, \mathcal{I}_X, b, E)$, determine the locus of maximal order of $\mathcal{I}_X$.

Background

In the exposition of this part, we follow closely the definitions in [3].

Definition 54. $(\Delta(\mathcal{I}_X))$ For a basic object $(W, \mathcal{I}_X, b, E)$, we define $\Delta(\mathcal{I}_X) \subset \mathcal{O}_W$ as the sheaf of ideals locally generated by

$$\{g_i | 1 \leq i \leq s\} \cup \{\frac{\partial g_i}{\partial x_j} | 1 \leq i \leq s, 1 \leq j \leq d\},$$

where $x_1, \ldots, x_d$ is a regular system of parameters for $\mathcal{O}_{W,w}$ and $g_1, \ldots, g_s$ are a set of generators for I_w. $\Delta^i(\mathcal{I}_X)$ is then inductively defined as $\Delta(\Delta^{i-1}(\mathcal{I}_X))$.

Lemma 55. *(Locus of Order at least c) The locus of order at least c of $\mathcal{I}_X$ coincides with $V(\Delta^{c-1}(J))$.*

Computational Approach

The definition of $\Delta(J)$ and hence the calculation of the locus of maximal order heavily rely on using generators for the ideal $I_w \subset \mathcal{O}_{W,w}$ and a regular system of parameters for $\mathcal{O}_{W,w}$ at the given closed point $w \in W$. Theoretically this is fine, but in practice it is, of course, not feasible to compute at each point of W. Here, the use of a set of generators of $I_w \subset \mathcal{O}_{W,w}$ does not cause any problems, since we are working on affine charts and on each chart we are specifying $\mathcal{I}_X$ by a set of generators anyway. For simplicity, we

now assume again that our ambient space W is contained in some $\mathbb{A}^N$ as in the previous section. The difficulties arise from the computational need to have a global system $y_1, \ldots, y_d \subset \mathcal{O}_W$ inducing a local regular parameters for $\mathcal{O}_{W,w}$ on the whole W, which, in general, does not exist.
To avoid this problem, it is necessary to pass to a suitable open covering $\{U_j\}$ of $\mathbb{A}^N$ such that for each U_j we can find a global system giving rise to a regular system of parameters at each point of U_j. This, of course, increases the number of charts, a drawback which can, in turn, be eliminated by recombining the results on the U_j to one on $\mathbb{A}^N$ in a suitable way. More precisely, $\Delta(J)$ is determined by the following algorithm:

Algorithm Delta

Input $(g_1, \ldots, g_r)$ generating $\mathcal{I}_W \subset \mathbb{C}[x_1, \ldots, x_n] = \mathcal{O}_{U_i}$
$(f_1, \ldots, f_s)$ generating $\mathcal{I}_X \subset \mathbb{C}[x_1, \ldots, x_n]$
such that $V(\mathcal{I}_W)$ is equidimensional and regular and $\mathcal{I}_W \subset \mathcal{I}_X$.
Output $\Delta(\mathcal{I}_X) \subset \mathbb{C}[x_1, \ldots, x_n] = \mathcal{O}_{U_i}$

(1) if $\mathcal{I}_W = (0)$
then return$((f_1, \ldots, f_s, \frac{\partial f_1}{\partial x_1}, \ldots, \frac{\partial f_s}{\partial x_n}))$
(2) Initialization
$C = \{f_1, \ldots, f_s\}$
$D = (1)$
$L1 = \{$ $n-\dim(W)$ square submatrices of the Jacobian matrix of $\mathcal{I}_W$ whose determinant is non-zero$\}$ [C]
(3) **while** $(L1 != \emptyset)$
- choose $M \in L1$
  $L1 = L1 \setminus \{M\}$
- $q = \det(M)$
- determine an $n - \dim(W)$ square matrix A such that
  $A \cdot M = q \cdot E_{n-\dim(W)}$ [D]
- determine components of $\Delta(J)$ not lying inside $V(q)$:

$$C_M = C \cup \left\{ q \cdot \frac{\partial f_i}{\partial x_j} - \sum_{\substack{k \text{ column of } M \\ l \text{ row of } M}} \frac{\partial g_l}{\partial x_j} A_{lk} \frac{\partial f_i}{\partial x_k} \,\middle|\, {}_{j \text{ not column of } M}^{1 \le i \le s,} \right\}$$

[C] For simplicity, the row and column indices used inside the submatrices will be the ones of the corresponding rows resp. columns in the Jacobian matrix.
[D] E_j denotes the $j \times j$ unit matrix. As before, we use row and column indices corresponding to those of M for simplicity

$C_M = sat(C_M, q)$

- Add these components to the previously found ones:
  $D = D \cap C_M$

(4) return(D)

The basic idea behind this algorithm is that W is regular and hence at each point there is at least one $(n - \dim(W))$-minor of the Jacobian matrix of $\mathcal{I}_W$ which does not vanish. This allows us to pass to the open covering defined by the complements of the vanishing loci of these $(n-\dim(W))$-minors. Fixing one such $(n - \dim(W))$-submatrix M of the Jacobian matrix and working on the complement of the minor $detM$, the desired global system of local parameters can be determined based on the following observation: As $detM$ is now invertible, the generators of the ideal of W corresponding to the respective rows in M can be used as a local system of parameters at each point of our open set. Direct calculation then yields expressions for the derivatives of the generators of $\mathcal{I}_X$. To obtain elements of the polynomial ring, these expressions need to be multiplied by the unit $detM$ which provides precisely

$$q \cdot \frac{\partial f_i}{\partial x_j} - \sum_{\substack{k \text{ column of } M \\ l \text{ row of } M}} \frac{\partial g_l}{\partial x_j} A_{lk} \frac{\partial f_i}{\partial x_k}$$

used in the algorithm.
The set containing these expressions and the original generators of $\mathcal{I}_X$ generates an ideal $\mathtt{C_M}$ in the polynomial ring which contains $\mathcal{I}_W$ as we assumed at the beginning of the algorithm that $\mathcal{I}_W \subset \mathcal{I}_X$. The subscheme corresponding to this ideal coincides with the desired $\Delta(\mathcal{I}_X)$ on the complement of $detM$; on $V(detM)$, however, we are likely to see components which do not have any relation to $\mathcal{I}_X$. So we remove all components contained in $V(detM)$ by saturation. As the missing components of $\mathcal{I}_X$ inside $V(detM)$ appear automatically when considering the other U_j, we make sure that we have determined all components by forming the intersection of all ideals $\mathtt{C_M}$ in the last step of the algorithm.

From the practical point of view the above algorithm still needs to be improved to avoid redundant calculations. In particular, one should first check whether there is a minor of the appropriate size which is itself an element of $\mathbb{C}$. In this case, the complement of the minor is the whole $\mathbb{A}^N$ and the other minors obviously do not provide any new contributions.

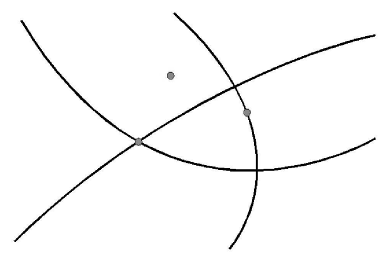

Figure 4. As an example for the problem of computing $\Delta(J)$, let us consider the situation illustrated in the above picture: There are three minors whose determinant does not vanish (each one illustrated by one of the curves in the above picture) and $V(\Delta(J))$ consists of the three points. Then computing on the complement of just one of the minors will not provide all points of $V(\Delta(J))$, because each of the curves meets at least one point.

4.2.4. *Descent in Dimension*

Computational Task

Given a basic object $(W, \mathcal{I}_X, b, E)$, where b is the maximal order of $\mathcal{I}_X$, find (at least locally) a hypersurface of maximal contact and determine an auxiliary basic object permitting the induction step of the resolution process.

Background

Among the notions and facts necessary for this task, the definition of a hypersurface of maximal contact has already been defined in section 4.2.1; for the other statements a good reference is [3] or [24]. A different point of view for the construction is taken in [26] discussing conditions on the possible constructions of an auxiliary object rather than just stating one construction.

Lemma 56. *(Choice of a Hypersurface of Maximal Contact) Given a basic object $(W, \mathcal{I}_X, b, E)$, where b is the maximal order of $\mathcal{I}_X$, and a point $w \in X \subset W$, any order 1 element of $\Delta^{b-1}(\mathcal{I}_X)_w \subset \mathcal{O}_{W,w}$ which satisfies conditions (c) and (d) can be chosen as a hypersurface of maximal contact in a sufficiently small neighborhood of the point w.*

Definition 57. (Auxilliary Basic Object (Villamayor's Construction)) Given a basic object $(W, \mathcal{I}_X, b, E)$, where b is the maximal order of $\mathcal{I}_X$, and an open set $U \subset W$ on which the hypersurface Z can be chosen as hypersurface of maximal contact for the given basic object, the auxiliary basic object $(Z, \mathcal{I}_{new}, c, E_{new})$ (on U) is defined as

$$\mathcal{I}_{new} := Coeff_Z(\mathcal{I}_X) = \sum_{i=0}^{b-1} (\Delta^i(\mathcal{I}_X))\mathcal{O}_Z)^{\frac{b!}{b-i}}$$

$$c := b!$$

$$E_new := \{E_i \cap Z | E_i \in E \text{ born after maximal order dropped to b}\}$$

Lemma 58. *(Coefficient Ideal and Blow Up 'commute')*
Let $\mathcal{B} = (W, \mathcal{I}_X, b, E)$ be a basic object and let $\mathcal{A} = (Z, \mathcal{I}_{new}, c, E_new)$ be an auxiliary basic object as defined above. Then the controlled transform of $\mathcal{A}$ w.r.t. the control c under a blow up at a center determined by the governing invariant coincides with the auxiliary basic object constructed from the weak transform of $\mathcal{B}$ under the same blow up using the strict transform of Z as the hypersurface of maximal contact.

Computational Approach

For the descent in dimension, that is the computation of the coefficient ideal, the crucial point is hence the choice of the smooth hypersurface Z which is subject to two normal crossing conditions regarding the exceptional divisors. As soon as such a hypersurface is found, the computation of the coefficient ideal only involves determining the Δ^i of the ideal which has previously been discussed and basic operations on ideals such as taking powers and sums.

As already mentioned, such a hypersurface Z usually does not exist globally. In an implementation, the choice of the hypersurface involves passing to a suitable open covering such that on each open set U_j there is a hypersurface which can be used as Z for each point $w \in U_j$. The basic idea for finding such a covering is to consider $\Delta^{c-1}(\mathcal{I}_X)$. As the intersection of the singular loci of the generators of $\Delta^{c-1}(\mathcal{I}_X)$ is empty (c is the maximal order), it is possible to express 1 as a combination of the generators of the ideals of these singular loci and use the complements of those generators appearing with non-zero coefficients as the open covering[E]

[E]Of course, it is necessary to check that the two normal crossing conditions hold and, if necessary, pass to a different way of expressing 1 in terms of the generators of the singular loci.

The need to pass to an open covering can enlarge the number of charts significantly which slows down the subsequent steps of the resolution process due to duplicate calculations for points/subvarieties/centers appearing in more than one chart, as we already mentioned before. The first idea to keep the number of open sets as low as possible is to recombine in the end in the same way as in the algorithm for determining Δ (that is passing to the closure, dropping components which do not meet the respective open set and taking the intersection of the resulting ideals). Unfortunately, the auxiliary objects really depend on the chosen hypersurface, although the resulting value of the governing function at each point is independent of this choice. Therefore, we cannot recombine directly as before; instead, we continue with the algorithm for finding the maximal locus of the governing function in each of the open sets and then (carefully) recombine those maximal loci in the following way: After passing to the closure and dropping components not meeting the open set U_j, each open set U_j provides an ideal $\mathcal{I}_{Y_j}$ describing a candidate for the next center. To this ideal $\mathcal{I}_{Y_j}$ we can associate the value v_j of the governing function corresponding to one (and hence any) point in Y_j. The next center then corresponds to the ideal

$$\bigcap_{\substack{j \text{ such that} \\ v_j \text{ maximal}}} \mathcal{I}_{Y_j} \quad .$$

4.2.5. *Identification of Exceptional Divisors*

The last subtask, which we want to discuss, is the identification of points resp. subvarieties which occur in more than one chart; in particular, we need to decide whether two given exceptional divisors living in two different charts actually belong to the same exceptional divisor (after gluing the charts). To this end, we move through the tree of charts arising during the resolution process, first blowing-down from the first chart to the one in which the history of the two charts in question branched, and then blowing-up again to the other chart with which we want to compare (cf. figure 5).

As blow-ups are isomorphisms away from the center, this process of successively blowing-down and then blowing-up again does not cause any problems for points which do not lie on an exceptional divisor at all or only lie on exceptional divisors, which already exist in the chart at which the history of the considered charts branched.

If, however, the point lies on an exceptional divisor which arises later, then blowing-down beyond the moment of birth of this divisor will inevitably lead to incorrect results. To see this let us consider an exceptional divisor

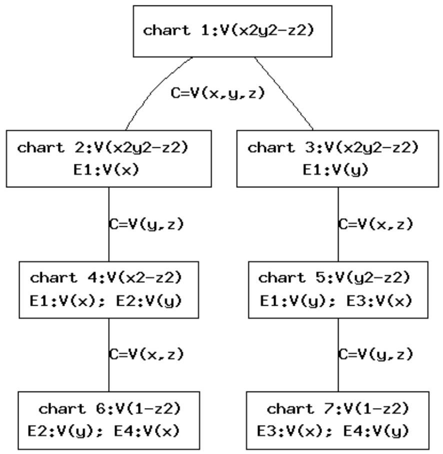

Figure 5. The tree of blow-ups in the resolution of the singularity $V(x^2y^2 - z^2) \subset \mathbb{A}^3$. For simplicity of notation the variable names in all charts have been chosen to be x, y, z. To determine whether two exceptional divisors in two different charts actually belong to the same exceptional divisor, we need to move through the tree by first blowing down and then blowing up again; for instance, the question, whether the divisors $V(x)$ in chart 6 and $V(y)$ in chart 7 belong to the same divisor, can only be answered by comparing the centers in charts 4 and 5. To this end, we have to move from chart 4 to chart one by blowing down twice and then proceed to chart 5 by blowing up twice.

$Y \times \mathbb{P}^k$ originating from a blow up at Y and a point $y \in Y$: two distinct points p and q in $y \times \mathbb{P}^k$ are always blown down to the same point y. To avoid this problem, we need to represent the points on the exceptional divisor as the locus of intersection of the exceptional divisor with an auxiliary variety which is not contained in the exceptional divisor. More formally speaking, we use the following simple observation from commutative algebra:

Remark 59. Let $I \subset K[x_1, \dots, x_n]$ be a prime ideal, $J \subset K[x_1, \dots, x_n]$ another ideal such that $I + J$ is equidimensional and $ht(I) = ht(I + J) - r$ for some integer $0 < r < n$. Then there exist polynomials $p_1, \dots, p_r \in I + J$ and a polynomial $f \in K[x_1, \dots, x_n]$ such that

$$\sqrt{I + J} = \sqrt{(I + (p_1, \dots, p_r)) : f}.$$

In our situation, the ideal I is, of course, the ideal of the intersection of the exceptional divisors in which the point or subvariety $V(J)$ is contained.

Any sufficiently general set of polynomials $p_1, \ldots, p_r \in J \setminus (I \cap J)$ leading to the correct height of $I + (p_1, \ldots, p_r)$ may be chosen and the only truly restricting condition on f is that it has to exclude all extra components of $I + (p_1, \ldots, p_r)$. Thus we also have enough freedom of choice of the $p_1, \ldots, p_r, f$ to achieve that none of these is contained in any further exceptional divisor whose moment of birth has to be crossed when blowing-down. Having solved the problem of identification of points existing in more than one chart, we are now prepared to determine which exceptional divisor in one chart coincides with which one in another chart by simply comparing the centers giving rise to these exceptional divisors. To this end, we start at the root of the tree of charts of the resolution and work our way up to the final charts. The criteria for identification of the centers are quite simple:

- centers do not coincide, if the corresponding values of the governing function do not coincide,
- centers do not coincide, if the exceptional divisors in which they are contained do not coincide
- explicit comparison of centers (which are not excluded by the previous criteria) is possible by means of successive blowing down and blowing up as described above.

At this point, we need to recall that computations in a computer algebra system are performed over $\mathbb{Q}$ not over the complex numbers although the reasoning often takes place over $\mathbb{C}$. This is particularly important during interpretation of the results here, because using this computational approach we have not yet been able to determine, for instance, the correct number (over $\mathbb{C}$) of exceptional divisors arising during the resolution process. This will be a crucial issue in the subsequent section.

4.2.6. *Intersection Matrix of Exceptional Curves*

Computational Task

Given an embedded resolution of a isolated surface singularity (where the singular locus consists of precisely one point), pass to a non-embedded resolution and compute the intersection matrix of the exceptional divisors. This task may be split into three subtasks:

(1) determine a non-embedded resolution from a given embedded one
(2) determine the intersections $E_i.E_j$ for exceptional divisors $E_i \neq E_j$
(3) determine the self-intersection numbers

Background (1)

As the theorem of Embedded Resolution of Singularities has already been stated in theorem 40, we only state the more general task of non-embedded resolution of singularities here and illustrate the difference in an example. For a detailed comparison of various resolution type tasks and theorems including embedded and non-embedded resolution, see e.g. [26], sections 2–3, and [6], sections 13 and 17.8.

Theorem 60. *(Resolution of Singularities) Let X be an algebraic variety over a field of characteristic zero. Then there is a resolution of singularities, i.e. a proper birational morphism $\pi : \tilde{X} \longrightarrow X$ and a non-singular variety $\tilde{X}$ such that π is an isomorphism away from the singular locus of X.*

Example 18 (Comparison of Embedded and Non-Embedded Task) Given an affine variety $X = V(x^2 - y^3) \subset \mathbb{A}^2_{\mathbb{C}}$, consider the blow up at the center $V(x, y)$. The total transform of X under the blow up is given by the variety $V(x^2 - y^3, ay - bx) \subset \mathbb{A}^2_{\mathbb{C}} \times \mathbb{P}^1_{\mathbb{C}}$ where $(a : b)$ are the variables corresponding to the $\mathbb{P}^1_{\mathbb{C}}$. For the corresponding non-embedded blow up, we may use the same calculation (due to lemma 50) keeping in mind that the exceptional divisor needs to be considered as a subscheme of $\tilde{X}$ in this case. In charts, we thus have:

- $a \neq 0$ $\tilde{X} \cap U_1 = V(x^2 - y_{new}^3 x^3)$ is the total transform where $y_{new} := \frac{b}{a}$, the exceptional divisor is $E = V(x)$ and the strict transform is $V(1 - y_{new}^3 x)$ which is non-singular and does not meet E. In this chart, we have hence achieved both a resolution and an embedded resolution of singularities (the sense that all further blow ups in both resolution processes will be isomorphisms on this chart).
- $b \neq 0$ $\tilde{X} \cap U_2 = V(x_{new}^2 y^2 - y^3)$ is the total transform where $x_{new} := \frac{a}{b}$, the exceptional divisor is $E = V(y)$ and the strict transform is $V(x_{new}^2 - y)$ which is non-singular, but tangent to E. Therefore, we have achieved non-embedded resolution of singularities, but not embedded resolution of singularities for this chart.

These considerations show that the given blow up is a resolution of singularities, but not an embedded resolution of singularities.

Note that a comparison of the two tasks as in the above example was possible because X was embedded in W. This is also the condition under which all further computational remarks need to be understood, because we are actually using Villamayor's algorithm for embedded resolution of singularities to obtain a non-embedded one.[F]

Remark 61. (Functoriality/Canonicity) An algorithm for resolution of singularities is called functorial (or canonical) if it commutes with smooth morphisms and closed embeddings and is compatible with change of fields (see e.g. [26], section 3, or [41], section 2.4). Villamayor's algorithm, however, is not functorial because it depends on the embedding as the following example shows:
Let $(W, \mathcal{I}_X, b, \emptyset) = (\mathbb{A}^4_{\mathbb{C}}, \langle x^2 - y^3, y^4 z^3 - w^8\rangle, 2, \emptyset)$ be a basic object where and consider a closed embedding $i : W \hookrightarrow \mathbb{A}^5_{\mathbb{C}} = W_1$ inducing a basic object $(W_1, \mathcal{I}_1 := \mathcal{I}_X \mathcal{O}_{W_1}, b, \emptyset) = (\mathbb{A}^5_{\mathbb{C}}, \langle x^2 - y^3, y^4 z^3 - w^8\rangle, 2, \emptyset)$. Choosing the center $V(x, y, w) \subset \mathbb{A}^4_{\mathbb{C}}$ as required by Villamayor's algorithm, we obtain order reduction in all three charts of the corresponding blow up. In $\mathbb{A}^5_{\mathbb{C}}$ the corresponding choice of center $V(x, y, w, v)$ also yields order reduction in three of the four charts; in the chart with exceptional divisor $V(v)$, however, the maximal order stays two which forces us to blow up once again in contradiction to the situation before embedding into $\mathbb{A}^5_{\mathbb{C}}$.
This is an important fact to keep in mind when using Villamayor's algorithm in practical examples, but it is not as much of a problem as it might seem at first, because we choose the embedding at the very beginning of our calculations and do not change it afterwards.

Computational Approach (1)

Given an embedded resolution of an isolated surface singularity, stored as a tree of charts, we would like to pass to a non-embedded resolution by dropping unnecessary blow-ups at the end of the branches of the tree of charts. To this end, we compute the list of exceptional divisors by identifying them in the different charts as described in the previous section. Starting at the final charts, we then move backwards through the resolution tree and cancel those blowing ups which are not necessary for the non-embedded resolution (see illustration 6 for an example).

Then we consider the intersection of the remaining exceptional divisors of the embedded resolution with the strict transform to obtain the exceptional locus of the non-embedded resolution. We can easily decompose

[F] It is in general not true that any given scheme X can be embedded into a smooth scheme. An example can e.g. be found in [26], remark 33 in section 3.

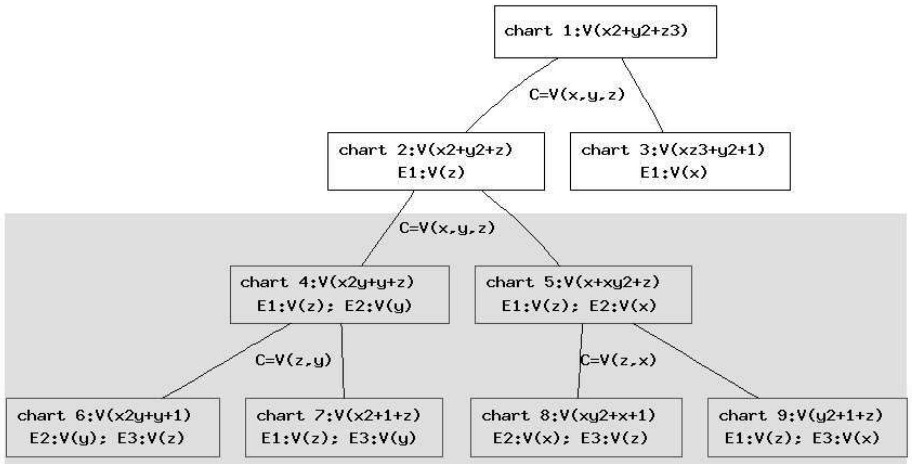

Figure 6. Tree of the embedded resolution process of an A_2 surface singularity. All charts which are marked by gray background arise from blow-ups which are only necessary in the embedded case, but not for a non-embedded resolution.

these intersections into irreducible components over $\mathbb{Q}$, but these components may still be reducible over $\mathbb{C}$.

Background (2)

For computing a decomposition over $\mathbb{C}$ of the exceptional divisors which is necessary for determining their intersection matrix, the following theorem (cf. [17]) is used:

Theorem 62. *(Gao/Ruppert) Let $f \in \mathbb{Q}[x,y]$ be irreducible of bidegree (m,n). Let $G = \{g \in \mathbb{Q}[x,y] | (m-1,n) \geq deg(g), \exists h \in \mathbb{Q}[x,y], \frac{\partial(g/f)}{\partial y} = \frac{\partial(h/f)}{\partial x}\}$. The vector space G has the following properties*

- *(i) f is irreducible in $\mathbb{C}[x,y]$ if and only if $dim_{\mathbb{Q}}(G) = 1$.*
- *(ii) $gG \subset \frac{\partial f}{\partial x} G$ mod f for all $g \in G$.*
- *(iii) Let $g_1, \ldots, g_a \in G$ be a basis and $g \in G \setminus \mathbb{Q}\frac{\partial f}{\partial x}, gg_i = \sum a_{ij} g_j \frac{\partial f}{\partial x}$ mod f. Let $\chi(t) = det(tE - (a_{ij}))$ be the characteristic polynomial. Then χ is irreducible in $\mathbb{Q}[t]$.*
- *(iv) $f = \prod_{c \in \mathbb{C}, \chi(c)=0} gcd(f, g - c\frac{\partial f}{\partial x})$ is the decomposition of f into irreducible factors in $\mathbb{C}[x,y]$.*

We use this theorem for the decomposition of the exceptional curves of our surface, which are irreducible over $\mathbb{Q}$, over $\mathbb{C}$ by means of the following corollary:

Corollary 63. [G] *Let $I \subset \mathbb{Q}[x_1, \ldots, x_n], ht(I) = 1$, be a prime ideal. Then there exists an irreducible polynomial $\chi(t) \in \mathbb{Q}[t]$ such that the complex zeros of $\chi(t) = 0$ correspond to the associated prime ideals of $I\mathbb{C}[x_1, \ldots, x_n]$.*

Background (2 and 3)

Definitions and properties on intersection numbers of divisors on surfaces can be found in many books on algebraic geometry, among others in [23], section V.1, and in [35], section IV.1.

Definition 64. (Intersection Numbers) If D_1, D_2 are divisors on a non-singular surface X in general position[H], then

$$D_1.D_2 := \sum_{x \in D_1 \cap D_2} (D_1.D_2)_x$$

is the intersection number of D_1 and D_2, where $(D_1.D_2)_x$ denotes the intersection multiplicity at x.

Lemma 65. *(Invariance under Linear Equivalence) For any divisors D_1 and D_2 on the non-singular surface X, there exist divisors D_1', D_2' such that $D_i \sim D_i'$ and D_1', D_2' are in general position.*
If D_1, D_2 and D_1', D_2' are two tuples of divisors in general position, then $D_1.D_2 = D_1'.D_2'$

Note that this lemma allows the definition of intersection numbers for any two divisors on X by means of passing to linearly equivalent divisors in general position.

[G] For $n = 2$ this statement is obvious; in the case $n > 2$, a (suitable) generic linear coordinate change leads to $I \cap \mathbb{Q}[x_{n-1}, x_n] = (f)$ where the associated primes of I correspond to the associated primes of (f). Geometrically this is a generic projection of the curve defined by I to the plane.

[H] D_1 and D_2 are called in general position, if the intersection $Supp(D_1) \cap Supp(D_2)$ is either empty or a finite set of points.

Computational Approach (2)

Example 19 As a simple example of the situation, let us consider the plane curve $V(x3 - 2y3)$ which is $\mathbb{Q}$-irreducible, but consists of three components over $\mathbb{C}$:

```
ring R=0,(x,y),dp;                // the ring
poly p=x3-2y3;                    // the polynomial
getMinpoly(p);

[1]:                              // the polynomial \chi(t)
   poly p=t^3-2;
[2]:                              // its 3 complex zeros
   [1]:
      (-0.62996052494743658238336+i*1.09112363597172140356O1)
   [2]:
      (-0.62996052494743658238336-i*1.09112363597172140356O1)
   [3]:
      1.25992104989487316476721061
[3]:
   3
```

Using the field extension proposed by the output of `GetMinpoly`, we now pass to the field extension $\mathbb{Q}[t]/t^3 - 2$ and factorize:

```
 ring T=(0,t),(x,y),dp;           // new ring, with parameter t
 minpoly=t3-2;                    // minimal polynomial
 factorize(x3-2y3);               // factorization

[1]:
   _[1]=1
   _[2]=x2+(t)*x*y+(t2)*y2
   _[3]=x+(-t)*y
[2]:
   1,1,1
```

This is obviously not a complete factorization. But a complete factorization can only be achieved using a Galois extension which is of higher degree (in this case degree 6). Therefore the factorization is more expensive from a computational point of view, i.e. it takes more time and memory, and so are all further calculations over this field. (Just look at the number of summands in the coefficients of y in our example below!).

```
 ring T=(0,t),(x,y),dp;
 minpoly=t6+3t5+6t4+11t3+12t2-3t+1;
 factorize(x3-2y3);

[1]:
   _[1]=1
   _[2]=x+(2/9t5+7/9t4+14/9t3+26/9t2+37/9t+2/9)*y
   _[3]=x+(1/9t5+2/9t4+4/9t3+4/9t2-1/9t-11/9)*y
   _[4]=x+(-1/3t5-t4-2t3-10/3t2-4t+1)*y
[2]:
   1,1,1,1
```

To identify the $\mathbb{C}$-components of an exceptional divisor E (irreducible over $\mathbb{Q}$) in a chart, we, therefore, store E, $\chi(t)$ and the respective numerical root of $\chi(t)$. Given these data, we can then proceed in the same way as for the identification of the $\mathbb{Q}$-components in the previous section. As soon as the exceptional divisors in the different charts are identified, we can directly compute the intersection numbers $E_i.E_j$ for all $i \neq j$.

Background (3)

To compute the self intersection numbers, we need some more properties of intersection numbers under birational morphisms. For a more detailed discussion of the topic of intersection numbers of divisors and their behavior under blowing up see e.g. [35], sections IV,3–4[I]

Theorem 66. *(Intersection Numbers and Blowing Up) Let $\pi : \tilde{X} \longrightarrow X$ be a resolution of singularities of the surface X. Let D_1 be a divisor on $\tilde{X}$ all of whose components are exceptional curves of π and let D_2 be any divisor on X, then*

$$\pi^*(D_2).D_1 = 0$$

[I]The statements there are all for non-singular surfaces. For passing from the case of a non-singular surface X to a singular one see the remark in [35] right after section III,1.3, theorem 1.

Computational Approach (3)

Let $\pi : \tilde{X} \longrightarrow X$ be a resolution of the surface X whose singular locus consists of just one point as specified in the statement of the task. Let $E_1, \ldots, E_s$ the exceptional divisors arising during the respective blow ups and let D be a non-trivial divisor on X (defined by means of a linear form $h : X \longrightarrow \mathbb{C}$ passing through the only singular point of X). Then we know

$$\pi^*(D).E_i = 0 \quad \text{and} \quad \pi^*(D) = \sum_{i=1}^{s} c_i E_i + H,$$

where the c_i are integers and H denotes the strict transform of D. The self intersection numbers can then be computed using the formula

$$0 = \pi^*(h)E_i = \sum_{i=1}^{s} c_j E_j.E_i + H.E_i \quad \forall 1 \leq i \leq s,$$

as all other intersection numbers can be/have already been computed directly.

References

1. Arnold,V., Gusein-Zade,S., Varchenko,A.: *Singularities of Differentiable Maps I*, Birkhäuser (1985)
2. Bierstone,E., Milman,P.:*Desingularization Algorithms I: The Role of Exceptional Divisors*, Mosc. Math. J. 3 (2003), 751-805
3. Bravo,A., Encinas,S., Villamayor,O.:*A Simplified Proof of Desingularisation and Applications*, Rev. Math. Iberoamericana 21 (2005), 349–458
4. Buchweitz,R.-O., Greuel,G.-M.: *The Milnor number and deformations of complex curve singularities*, Invent. Math. 58 (1980), 241-281
5. Campillo,A.: *Algebroid Curves in Positive Characteristic*, Springer (1980)
6. Cutkosky,S.D.: *Resolution of singularities*, Graduate Studies in Mathematics, 63, AMS (2004)
7. de Jong,T., Pfister,G.: *Local Analytic Geometry*, Vieweg, (2000)
8. Decker,W., Lossen,C.: *Computing in Algebraic Geometry - A quick start using* SINGULAR, Algorithms and Computation in Mathematics 16, Springer Verlag (2006).
9. Ebeling,W.: *Funktionentheorie, Differentialtopologie und Singularitäten*, Vieweg (2001)
10. Ebeling,W.: *Monodromy*, to appear in Singularities and Computer Algebra, Cambridge University Press (2006)
11. Ebeling,W.: Notes to the series of Talks entitled *Monodromy of Isolated Singularities* at this summer school
12. Eisenbud,D.: *Commutative Algebra with a View toward Algebraic Geometry*, Springer (1995)

13. Eisenbud,D., Huneke,C., Vasconcelos,W.: *Direct Methods for Primary Decomposition*, Invent. Math.110 (1992), 207–235
14. Frühbis-Krüger,A.: *Construction of Moduli Spaces for Space Curve Singularities* JPAA 164 (2001), 165-178
15. Frühbis-Krüger,A.,Neumer,A.: *Simple Cohen-Macaulay Codimension 2 Singularities*, preprint (2004)
16. Frühbis-Krüger,A., Pfister,G.: *Some Applications of Resolution of Singularities from a Practical Point of View*, in Proceedings of Computational Commutative and Non-commutative Algebraic Geometry, Chisinau 2004 (2005), 104–117
17. Gao,S.:*Factoring Multivariate Polynomials via Partial Differential Equations*, Math.Comp **72** (2003), 801–822
18. Gianni,P., Trager,B., Zacharias,G.: *Gröbner Bases and Primary Decomposition of Polynomial Ideals*, JSC 6 (1988), 149–167
19. Giraud,J.: *Sur la theorie de contact maximal*, Math.Z. 137 (1974), 285–310
20. Greuel,G.-M.: *On deformation of curves and a formula of Deligne*, in: Algebraic Geometry (Proceedings, La Rbida 1981) Springer (1983), 141-168.
21. Greuel,G.-M., Pfister,G.: *A* SINGULAR *Introduction to Commutative Algebra*, Springer (2002)
22. Greuel,G.-M., Pfister,G., Schönemann,H.: SINGULAR 3.0, `http://www.singular.uni-kl.de/`
23. Hartshorne,R.: *Algebraic Geometry*, Springer (1977)
24. Hauser,H.: *The Hironaka Theorem on resolution of singularities*, Bull. Amer. Math. Soc. 40 (2003), 323-403
25. Hironaka,H.: *Resolution of Singularities of an Algebraic Variety over a Field of Characteristic Zero*, Annals of Math. **79** (1964), 109–326
26. Kollar,J.: *Resolution of Singularities - Seattle Lecture*, math.AG/0508332
27. Krick,T., Logar,A. *An Algorithm for the Computation of the Radical of an Ideal in the Ring of Polynomials*, AAECC9, Springer LNCS 539 (1991), 195–205
28. Labs,O.: *A Septic with 99 Real Nodes*, math.AG/0409348
29. Laudal,O.A., Pfister,G.:*Local Moduli and Singularities*, Springer-Verlag, (1988)
30. Lê D.-T.: *Calculation of the Milnor number of an isolated singularity of a complete intersection*, Funkt. Anal. Ego Prilozheniya 8(2) (1974), 45-49
31. Looijenga,E.: *Isolated Singular Points of Complete Intersections*, Cambridge University Press, LNS 77 (1984)
32. Martin,B.: *Computing versal deformations with* ***Singular***, in: Algorithmic Algebra ans Number Theory, Springer (1998), 283-294
33. Milnor,J.: *Singular Points of Complex Hypersurfaces*, Princeton University Press (1968)
34. Schulze,M.: *A Normal Form Algorithm for the Brieskorn Lattice*, J.Symb.Comp.**38** (2004), 1207–1225
35. Shafarevich,I.: *Basic Algebraic Geometry*, Springer (1977)
36. Shimoyama,T., Yokoyama,K.: *Localization and Primary Decomposition of Polynomial Ideals*, JSC 22 (1996), 247–277

37. Steenbrink,J.: *Mixed Hodge Structure on the Vanishing Cohomology*, in Real and Complex Singularities (Proceedings Oslo 1976) SijthoffNoordhoff, Alphen a/d Rijn (1977), 525-563
38. Steenbrink,J.: *Semicontinuity of the singularity spectrum*, Invent. Math. 79(3) (1985), 557-565
39. Steenbrink,J.: *Mixed Hodge Theory: The search for purity*, in this volume
40. Wall,C.T.C.: *Singular Points of Plane Curves*, London Mathematical Society Student Texts 63 (2004)
41. Wlodarczyk,J.: *Simple Hironaka Resolution in Characteristic Zero*, J. Amer. Math. Soc. **18** (2005), 779-822