Computation of Adjoints for Surfaces
Proposal

Hoon Hong and Josef Schicho
RISC-Linz
Johannes Kepler University, A-4040 Linz
e-mail: {Hoon.Hong,Josef.Schicho}@risc.uni-linz.ac.at
http: http://www.risc.uni-linz.ac.at/people/{hhong,jschicho}

January 23, 1998

Contents

1 Summary

Goal of the project

The goal is to devise an efficient algorithm for the following problem.

Input: A surface in 3-space, given implicitly.
Two integers n, m.
Output: The 'm-adjoints of degree n' of the surface.

An 'm-adjoint of degree n' of a surface is a polynomial of degree n that vanishes with a certain order (depending on m) on the singularities of the surface. For example, look at the surface in figure 1. It has a singular line L. The polynomial given by G in figure 2 is a 1-adjoint of degree 1.

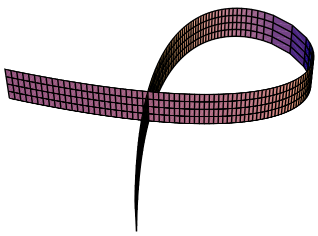


Figure 1: A surface with a singular line.

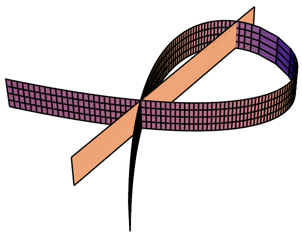


Figure 2: An 1-adjoint.

The set of all m-adjoints of degree n forms a vector-space. The output of the algorithm should be a basis.

Importance of the goal

Computation of adjoints is important because it is currently the only way to solve the parameterization problem, which is essential to compute NURBS-representations, which have numerous applications in CAD/CAM, such as

- reliable surface plotting and display,
- motion display (computing transformations),
- computing cutter offset surfaces,
- computing curvatures for shading and colouring,

- and many others.

Moreover, the computation of adjoints is important because it can be applied to solve classification problems for surfaces.

State of the art

In the curve case, adjoints are well understood theoretically, and there are several algorithms for the computation, some of them efficient.

In the surface case, adjoints are also well understood theoretically, but we still cannot compute them efficiently. Currently, there is only algorithm (by Schicho, one of the proposers), and this one is too slow.

A problem closely related to the computation of adjoints is the resolution of the singularities. There are several algorithms for these problems. But no complexity bounds are known for any of them.

How the goal could be reached

A natural approach for adjoints computation is to use a resolution of the singularities. Unfortunately, resolution is still a very complex problem. So, we propose to devise an algorithm that does not use a resolution. A promising alternative is using fractional power series.

If this approach fails, we propose to compare the existing algorithms for singularity resolution and analyze them from a computational point of view, in order to devise an efficient algorithm for resolution.

There is a risk that the goal might not be achieved. Even if a complexity bound could be established, it might be unacceptably high. However, any scientific venture bears such a risk, and we have the strong intuition that an efficient algorithm is possible (otherwise we would not have proposed this project).

Qualification of the researchers

Hoon Hong is qualified as a project leader because

- he already managed to finish several research projects successfully;
- he is the author of several published papers in the area of computational algebraic geometry.

Josef Schicho is qualified to do the main research because

- he is the leading expert in the algorithmic theory of adjoints;
- he gave the only existent algorithm for the computation of adjoints for surfaces, even though it is inefficient yet;
- he is the author of several published papers in the area of computational algebraic geometry.

2 Goal of the Project

2.1 The problem

We want to give an efficient algorithm for the following problem.

Input: A surface in 3-space, given implicitly.
Two integers n, m.
Output: The 'm-adjoints of degree n' of the surface.

The description of the problem was intentionally imprecise in the summary. Here, we give the precise description.

'3-space' means projective 3-space. The surface is given by an equation $F(x, y, z, w) = 0$, where F is a squarefree homogenous polynomial over some coefficient field k, which is supposed to be some computable field of characteristic zero in which we can factor univariate polynomials. For instance, k could be the field of real algebraic numbers.

The 'm-adjoints' are the elements of a certain homogenous ideal A_m of the graded coordinate ring $k[x, y, z, y]/(F)$. For the applications, it is not necessary to know the ideal completely, for instance by knowing a Gröbner basis [Buc85] of A_m. It suffices to know the vector-spaces of homogenous elements of some fixed degree n. This vector-space is finite-dimensional, and we want to compute a basis.

2.2 What is an m-adjoint?

An exact definition of adjoints and multi-adjoints can be found in [Enr49, Zar71, BL87, Sch95]. Here, we want to give a brief informal definition.

We first consider the affine case. Let X be a surface in affine 3-space, given by a polynomial equation $F(x, y, z) = 0$. Then $k[X] := k[x, y, z]/(F)$ is the function ring of X. A function $G \in k[X]$ is an m-adjoint iff it vanishes with order $\geq m(r-1)$ at each r-fold curve of X and with order $\geq m(r-2)$ at each r-fold point of X. Singularities in the infinitely near have to taken into account, too. These are introduced by blowing up the singularities. Moreover, the singularities have to be computed over the algebraic closure of k. Nevertheless, it is known that the vector-space of adjoints is generated by adjoints with coefficients in k itself.

Example 1 A classical case is the image of a generic projection of a smooth surface in $\mathbf{P}^5$ into $\mathbf{P}^3$. It is well-known (see [Kur82]) that the only possible singularities are ordinary double curves (two smooth branches intersecting tranversally), ordinary triple points (three smooth branches intersecting tranversally), or pinch points (branch points of the double curve).

In this case, one can easily show that the m-adjoints are the functions vanishing with order m along the double curve (see also [Enr49]).

In the homogenous case, a form G is an m-adjoint iff, for each of the 4 variables, dehomogenizing G with respect to that variable yields an m-adjoint for the dehomogenized surface.

Below is a list of sample outputs of the problem.

<table>
<tr><th>m</th><th>n</th><th>m-adjoints of degree n</th></tr>
<tr><td colspan="3">$x^2 + y^2 + z^2 - w^2 = 0$</td></tr>
<tr><td colspan="3">Singularities: none</td></tr>
<tr><td>1</td><td>1</td><td>$\langle x, y, z, w \rangle$</td></tr>
<tr><td>2</td><td>1</td><td>$\langle x, y, z, w \rangle$</td></tr>
<tr><td colspan="3">$x^2zw + y^2zw + x^2y^2$</td></tr>
<tr><td colspan="3">Singularities: line $x = y = 0$</td></tr>
<tr><td>1</td><td>1</td><td>$\langle x, y \rangle$</td></tr>
<tr><td>2</td><td>1</td><td>$\langle \rangle$ (the zero space)</td></tr>
<tr><td>2</td><td>2</td><td>$\langle x^2, xy, y^2 \rangle$</td></tr>
<tr><td colspan="3">$(y^2 - xz)(z^2 - yw) + (xw - yz)^2 = 0$</td></tr>
<tr><td colspan="3">Singularities: cubic curve $(x : y : z : w) = (1 : t : t^2 : t^3)$</td></tr>
<tr><td>1</td><td>1</td><td>$\langle \rangle$</td></tr>
<tr><td>1</td><td>2</td><td>$\langle y^2 - xz, z^2 - yw, xw - yz \rangle$</td></tr>
<tr><td>2</td><td>3</td><td>$\langle \rangle$</td></tr>
<tr><td>2</td><td>4</td><td>$\langle P^2, Q^2, PQ, PR, QR \rangle$, where
$P := y^2 - xz$, $Q := z^2 - yw$, $R := xw - yz$</td></tr>
<tr><td colspan="3">$x^2w^3 + y^3w^2 + z^5 = 0$</td></tr>
<tr><td colspan="3">Singularities: point $(0 : 0 : 0 : 1)$, line $z = w = 0$</td></tr>
<tr><td>1</td><td>1</td><td>$\langle \rangle$</td></tr>
<tr><td>1</td><td>2</td><td>$\langle yw, z^2, zw, w^2 \rangle$</td></tr>
<tr><td>2</td><td>2</td><td>$\langle \rangle$</td></tr>
<tr><td>2</td><td>3</td><td>$\langle z^2w, w^3 \rangle$</td></tr>
<tr><td>3</td><td>3</td><td>$\langle \rangle$</td></tr>
<tr><td>3</td><td>4</td><td>$\langle w^4 \rangle$</td></tr>
<tr><td colspan="3">$x^2w^4 + y^3w^6 + x^6 + y^6 + z^6 = 0$</td></tr>
<tr><td colspan="3">Singularities: point $(0 : 0 : 0 : 1)$</td></tr>
<tr><td>1</td><td>1</td><td>$\langle x, y, z \rangle$</td></tr>
<tr><td>1</td><td>2</td><td>$\langle x^2, xy, xz, xw, y^2, yz, yw, z^2, zw \rangle$</td></tr>
<tr><td>2</td><td>1</td><td>$\langle x, y \rangle$</td></tr>
<tr><td>2</td><td>2</td><td>$\langle x^2, xy, xz, xw, y^2, yz, yw, z^2 \rangle$</td></tr>
<tr><td>3</td><td>1</td><td>$\langle x \rangle$</td></tr>
<tr><td>3</td><td>2</td><td>$\langle x^2, xy, xz, xw, y^2, yz \rangle$</td></tr>
<tr><td>4</td><td>1</td><td>$\langle \rangle$</td></tr>
<tr><td>4</td><td>2</td><td>$\langle x^2, xy, xz, y^2 \rangle$</td></tr>
</table>

(If n is larger than or equal to the degree of the given surface, then the polynomials in the adjoints vectorspace have to be regarded as equivalence classes modulo multiples of the equation of the surface.)

2.3 Efficiency

An efficient method for the computation of adjoints is necessary for applications (see next section). Obviously, presenting an efficient algorithm includes providing a - reasonably low - bound for the computational complexity.

Whenever singularities are involved, there is a large difference between worst-case complexity and average-case complexity with respect to some statistical measure. The typical situation is that complicated singularities are statistically rare. This is not surprising, since singularities are something degenerate, non-generic, by nature.

However, there are many situations in which the degenerate cases are more interesting than the generic case. Not just because they are more complicated, which makes them interesting from a theoretical point of view. Sometimes, they are also more useful in applications. For instance, one is especially interested in rational surfaces in the context of geometric modeling. Except for degree 2 and 3, the surface must then be sufficiently singular, in order that the geometric genus vanishes.

Therefore, it is important to have an algorithm for the computation of adjoints with a worst-case complexity bound. A complexity analysis of 'some average case' is not sufficient.

3 Importance of the Goal

3.1 Rational Parameterization and Problems in CAD/CAM

Computation of adjoints is important because it is the only way to solve the parameterization problem. Currently, we have only one parameterization algorithm [Sch95], except for very limited classes of surfaces, like quadric and cubic surfaces [AB87, SS87] or canal surfaces [Pot96]. Computing the adjoints is the most difficult step in Schicho's general parameterization algorithm (see [Sch95, Sch97]).

The parameterization problem has the following specification.

Input: A surface in projective 3-space, given implicitly.
Output: A rational parameterization of the surface, if exists.
`NotExist` otherwise.

The input surface is given by an equation $F(x,y,z,w) = 0$, where F is an irreducible homogenuous polynomial over some computable field k (for instance, some subfield of the reals).

A rational parameterization is a parameterization in terms of rational functions in two variables s, t. Since we are in the projective case, we may clear denominators. Algebraically, a rational parameterization is a quadruple

$$\tau := (\overline{x} : \overline{y} : \overline{z} : \overline{w}) \in k[s,t]^4,$$

such that the following two conditions hold.

- $F(\overline{x}, \overline{y}, \overline{z}, \overline{w}) = 0$ for all s, t.
- The rank of the matrix

$$M(\tau) := \begin{pmatrix} \overline{x} & \overline{y} & \overline{z} & \overline{w} \\ \partial_s \overline{x} & \partial_s \overline{y} & \partial_s \overline{z} & \partial_s \overline{w} \\ \partial_t \overline{x} & \partial_t \overline{y} & \partial_t \overline{z} & \partial_t \overline{w} \end{pmatrix}$$

  over the field $k(s,t)$ is 3.

The second condition ensures that the image of the τ in $\mathbf{P}^3$ is not just a point or a curve on the surface.

Example 2 A rational parametrization of the unit sphere $x^2 + y^2 + z^2 - w^2 = 0$ is given by

$$(\overline{x} : \overline{y} : \overline{z} : \overline{w}) = (2s : 2t : s^2 + t^2 - 1 : s^2 + t^2 + 1).$$

Surfaces which have a rational parameterization are called unirational (see [Sha74]). In the algebraically closed case, unirationality coincides with rationality (see [Sha65]). There are irrational surfaces, for instance the K3-surface with equation $x^4 + y^4 + z^4 - w^4 = 0$. However, the rational surfaces form an interesting and important class of surfaces (see [Con39] for a decent theoretical introduction to this topic).

In more recent times, computing rational parameterizations has become especially important, because it is the most difficult step in computing NURBS- (non-uniform rational B-spline) -representions, which are used for many applications in Computer Aided Design and Manufacture, such as

- reliable surface plotting and display,
- motion display (computing transformations),
- computing cutter offset surfaces,
- computing curvatures for shading and colouring,
- and many others

(see also [BFK84, Bar85, QD87, Far88, Hof89]). NURBSs are already a de facto standard in CAD/CAM environments (see [Nie93, BCX94]), and their popularity is still growing.

3.2 Solving Geometric Problems on Irrational Surfaces

The scope of the method of adjoints reaches beyond parameterization. Like in the case of algebraic curves, adjoints can be used to solve various problems in the classification theory of surfaces.

They can be used to compute birational invariants, like the geometric and the arithmetic genus, the plurigeni, and the maybe also the Kodaira-dimension (see [Har77] for a definition of these numbers). It can be used to construct canonical models for surfaces of general type (see [Sha65] for definitions). Probably, it can also be used to construct the structure maps for irrational ruled and elliptic surfaces.

In the following, we will give a brief discussions of these possible applications.

Numerical invariants. The Kodaira-dimension, the geometric and arithemtic genus and the plurigeni of a surface are invariant under birational transformations. Therefore, they are important for classification. For instance, rational surfaces may be characterized by the vanishing of the arithmetical genus and by the second plurigenus (see [Cas96, Ser57, Zar58]).

The geometric and the arithmetic genus and the plurigeni can easily be computed with the method of adjoints (see [BL79, Sch95]). Computing the Kodaira-dimension is more difficult. This number describes the asymptotic growth of the plurigeni. Some theorems are required to compute this number by a computation of only finitely many plurigeni.

Canonical models. Canonical models are also important for classification. Two surfaces of general type are birationally equivalent iff their canonical models are projectively isomorphic (see [Sha65]).

The construction of canonical models is easy with the method of adjoints. See [Sch95] for a sketch of a construction.

Structure maps. Ruled surfaces are characterized by the property that they contain an algebraic family of rational curves. These curves appear as the fibers of a rational 'structure map'.

Constructing the structure map is an open problem. For the rational case, it is solved by Schicho's parameterization algorithm [Sch95, Sch97] with the method of adjoints. (In this case, we have several structure maps.) For the irrational case, one can use the Albanese map (see [Sha65]) to construct this structure maps theoretically. However, it is not known how to construct the Albanese map in an algebraic way. But maybe the idea in Schicho's parameterization algorithm can be generalized to find the structure map of an irrational ruled surface.

Similar things can be said about for elliptic surfaces, which contain an algebraic family of elliptic curves. Again, these curves appear as the fibers of a birational 'structure map'. Again, we can hope that one can construct this map by the method of adjoints, using ideas from Schicho's parameterization algorithm.

The classification problems are interesting for themselves, but they can also be used to solve diophantine problems like finding rational points on the surface (see [Lan74, Maz94]).

4 State of the Art

4.1 The curve case

For algebraic curves, the concept of adjoints was developed by A. Clebsch and M. Nöether in the 19th century. Soon, it became a standard technique in the theory of algebraic curves (see [?]). There are several algorithms for adjoints computation, and some of them are efficient ([SW91, vH94, AGR95]). Complexity bounds for some of these algorithms can be found in [Tei90, Koz94, MSW96]. An implemented version is part of the CASA package [TW97] (see also [Wal93]), which is available on the ftp site `ftp://ftp.risc.uni-linz.ac.at/pub/casa`.

The method of adjoints is applied for the parameterizing curves and for applications in CAD/CAM (see [AB87]). However, there are more and more difficult applications in CAD/CAM that deal with algebraic surfaces. This is the impact to make a method of adjoints for surfaces available.

4.2 The theory of adjoints

The theory of adjoints for surfaces has its origin in the work of A. Clebsch and M. Nöether [N$\ddot{7}$1]. Adjoints and multiadjoints have played a fundamental role in the surface theory of the classical Italian school of algebraic geometry (see [Enr49]), similar to the role of the canonical divisor in more modern treatments [Sha65, Kur82]. The surface monographies [Zar71] and [Enr49] both devote a whole chapter to adjoints (chapter 3 in both books). However, both references do not give algorithms for the computation of adjoints.

In the modern literature, references for adjoints on surfaces or varieties of higher dimension ([BL79, BL87, BBL95, Sch95]) are thin on the ground. Especially, no adjoints computation algorithm has been devised until '95, when Schicho gave one in [Sch95].

It should be noted that several contemporary authors have written papers on a topic called 'adjoints for surfaces' ([LP84, Som79, SvdV87, Oli90, Pal90].) However, in these papers, 'adjoints' are used in the context of abstract smooth surfaces or abstract surfaces with rather harmless singularities. Their main object is not computation but classification. They have nothing to do with the theory of adjoints for surfaces in 3-space in the sense of Clebsch and Nöether.

4.3 The only algorithm

There is only a sole adjoints computation algorithm, which is contained in [Sch95]. It is based on the concept of a resolution of the singularities by blowing ups (see [Har77]). To give an account on its performance, we have to describe its structure briefly.

For convenience of the recursive description of the algorithm, we change the specification slightly. It is obvious how to reduce the original adjoints computation problem to the task specified below.

Input: A surface X in 3-space, given implicitly.
A basis B for a vectorspace $V \subset k[X]$.
An integer $m > 0$.
Output: A basis for the subvectorspace of m-adjoints.

Here is a simplified (but working) description of the adjoint computation algorithm.

Algorithm *Adjoints*(m, X, V):
    if X is smooth then
        return B;
    else
(1)   choose a singularity S;
        $r :=$ the multiplicity of X at S;
        if S is a curve then
            $p := m(r-1)$;
        else (S is a point)
            $p := m(r-2)$;
(2) $(\pi : \tilde{X} \to X, E) :=$ (the blowup of X at S;
        $e :=$ the equation of the exceptional divisor;
(3) $V_1 :=$ the subvectorspace of the elements
                that vanish with order $\geq p$ at S;
(4) $f : V_1 \to k[\tilde{X}] :=$ division by $e^p \circ \pi^*$;
(5) $V_2 :=$ *adjoints*$(m, \tilde{X}, f(V_1))$;
(6)   return $f^{-1}(V_2)$;

There is some arbitrariness in step 1, but there are some rules for a valid choice.

Rule 1: S must be either a point or a smooth curve.

Rule 2: If S is a point, then it must not lie on a curve with the same multiplicity.

Rule 3: If S is a curve, then it must not contain a point with a higher multiplicity.

Rule 1 is necessary to perform the blowing up in step 2 in a way that $\tilde{X}$ is again a surface in 3-space. Here, the exact meaning of '3-space' is a bit tricky. See below.

Rule 2 and rule 3 ensure finiteness of the recursion.

Obviously, one can always find a singularity S according to the rules, provided that X is not smooth. Finding a singularity amounts to solving a system of algebraic equations in 3 unknowns over the algebraic closure of k. This can be done with Gröbner bases [Buc65, Buc85, BH91, BW93, Hon96a, Hon96b] (as suggested in [Sch95]) or with resultants [Col67, BT71, BCL82, Hon90, HON92b, HON92a, Mis93, HON95]. An equation solver specialized to 3 unknowns can be found in [Kal95]. The problem is not so easy, but it can be solved with a polynomial number (polynomial in the degree) of field operations.

If S is not defined over the coefficient field k (but in some algebraic extension), then the subvectorspace V_1 cannot be generated by elements with coefficient in k. However, singularities which are not defined over k come in conjugated sets. Therefore, it is always possible to eliminate the algebraic coeffients in basis elements at some later step.

4.3.1 A severe technical problem

A severe technical problem is how to perform blowing up in a way that the proper transform appears as a surface in 3-space. When blowing up a point (or a straight line), then it is possible to decompose the blown-up 3-fold into three (two) patches, each of which is isomorphic to $\mathbf{A}^3$ (see [GH78]). When the blowing up center is a smooth space curve, there is no such construction. If the curve is given as the intersection of two surfaces (with equations F and G) intersecting transversally (note that this always holds locally), the usual blowing-up construction (see [Kur82]) needs 4 affine coordinates for a patch of the blowing up, namely $x, y, z, F/G$. Hence we can embed the patch in $\mathbf{A}^4$, but not in $\mathbf{A}^3$.

Schicho solves this problem by eliminating one of the 4 coordinates with algebraic functions. In general, the algebraic functions then appear in the basis of the adjoint vectorspace. However, Schicho shows that it is possible to eliminate the algebraic functions in step 6.

This solution may be quite elegant and relatively simple, but one has to pay a prize for it. Namely, many sub-problems turn out to be more complex with algebraic functions present. One of these is the problem of computing the singularities and multiplicities. Indeed, the algebraic functions must be treated as new unknowns computationally, which increases the costs for solving equation systems drastically (see [Col67]). Another example is step 3, which becomes a nontrivial ideal membership problem, when algebraic functions are present.

In short: if we perform blowing-ups with Schicho's method, then we need to compute several Gröbner bases (or resultants) with an increasing number of variables during the computation. This is extremely costful. It really "blows up" the computation.

4.3.2 A severe theoretical problem

The termination of the algorithm above is not obvious. It depends on the following result [Hir84].

Theorem 1 *By repeating blowing up a singularity S, chosen according to the rules below, one arrives at a smooth surface after finitely many steps.*

Rule 1: *S must be either a point or a smooth curve.*

Rule 2: *If S is a point, then it must not lie on a curve with the same multiplicity.*

Rule 3: *If S is a curve, then it must not contain a point with a higher multiplicity.*

This result could be called 'Hironaka's strategy for the resolution of surface singularities'. It is not hard to show that the number of blowing-ups is bounded for fixed degree (Schicho, private communication). However, there is the following problem.

Remark 1 *We do not know any upper bound - depending on the degree of the surface - for the number of blowing-ups needed in Hironaka's strategy.*

As a consequence of the remark, there is **no bound for the computational complexity** of Schicho's adjoint computation algorithm.

There are two possible reasons for this problem.

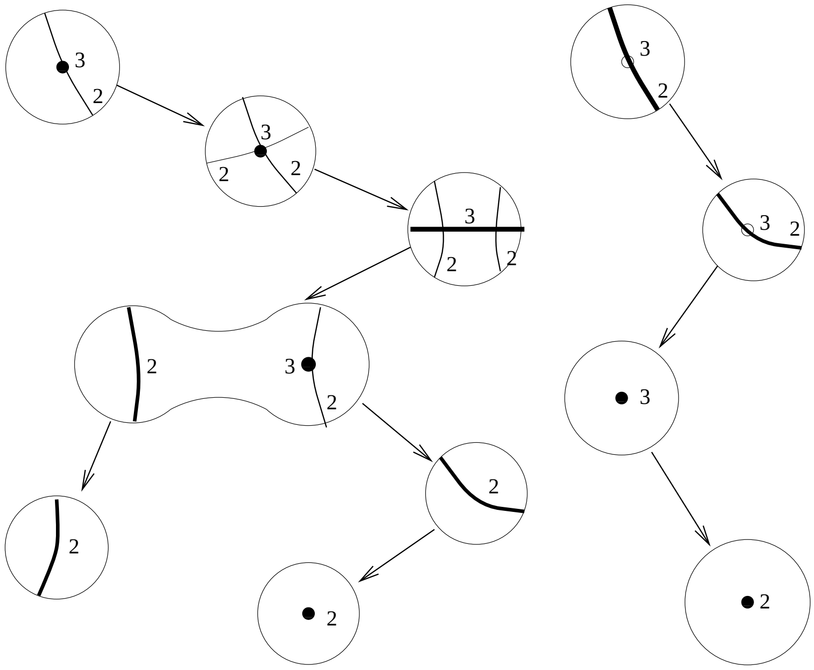
1. The problem itself (resolution by blowing ups) is complex.

2. Hironaka's strategy is complex.

An indication for the first possibility is the harmless looking, classical example $x^2 + y^3 + z^5 = 0$ (see [Har77], ex. 5.5.8). The double point at the origin cannot be resolved by less than 8 blowups.

As an indication for the second possibility, consider the surface $x^3 + x^2y^3 + z^5 = 0$. It has a triple point on the origin, lying on a double curve $x = z = 0$. Applying Hironaka's strategy, we have to blow up the triple point (because of rule 3). On the blown-up surface, we find that the exceptional divisor is a new double curve, intersecting the 'old' double curve transversally in a triple point. Now, we have to blow up the triple point.

Continuing with Hironaka's strategy, we need 8 blowing ups (see figure 2). The rules do not leave much choice in this case.



Hironaka's strategy A simpler resolution

Figure 2: Two resolutions of $x^3 + x^2y^3 + z^5 = 0$.

On the other hand, there is a simpler resolution by blowing ups, using only 4 steps (see figure 2). It starts with blowing up the double curve $x = z = 0$, violating rule 3.

This example shows that Hironaka's strategy is not always the optimal way to resolve by blowing-ups.

4.4 The related problem of the resolution of singularities

The problem of the resolution of singularities is related to the problem of adjoints computation because resolving the singularities is the most difficult step in Schicho's adjoints computation algorithm (see section 3.2.2). Obviously, the adjoints computation algorithm works with *any* strategy for the resolution of the singularities by blowing up.

We already mentioned (section 3.2.2) that we do not know a bound for the number of blowing ups needed in Hironaka's strategy. Consequently, there is no bound for the computational complexity of the resolution algorithm. But several author's [Jun08, Wal35, Zar39, Abh69, Hir84, Lip78, Vil89, BM91] gave procedures for the resolution of surface singularities.

Hironaka's strategy (section 3.2.2) is the simplest, but probably not the most efficient resolution procedure (see the example in section 3.2.2). The most promising alternatives are Abhyankar's strategy [Abh90] and Zariski's strategy (see [Lip75]). However, here is the sad truth about the existing resolution algorithms: for none of them, a complexity bound is known. We do not even know a bound for the number of blowing ups in any of the resolution algorithms.

Moreover, there is currently no implementation of any of the resolution algorithms. The resolution problem for surface singularities was considered an 'open problem in computational algebraic geometry' in [Eis93]. (In some sense, this problem was solved by Schicho's algorithmic realization of Hironaka's strategy [Sch95]. However, there is still no implementation.)

Nevertheless, we want to discuss two important alternatives for Hironaka's strategy, since they are relevant for the goal of this project.

4.4.1 Abhyankar's method

Abhyankar's work on the resolution of surface singularities is outstanding. Also, he represents his ideas with great clarity and many explicit calculations.

The concept of multiplicity is also used here: using Abhyankar's strategy, we can be sure that the multiplicities of the new singularities are not larger than the old multiplicities which have been "blown away". But we are sometimes violating Hironaka's rule 1, blowing up certain points with the same multiplicity as a smooth curve through it.

The main concept to control this is the notion of good and bad points. If a curve contains only good points, then (by definition of good) it is possible to reduce the multiplicity just by blowing up curves.

The advantage - with respect to Hironaka's strategy - is that it suffices to blow up the curves generically: when the multiplicity has dropped at the generic point, then the multiplicity has dropped everywhere.

Computationally, this is much easier than the usual blowing up of a smooth curve. We basically just have to blow up a zero-dimensional set, where the ground field has been enlarged by a transcendental extension.

Abhyankar's strategy proceeds along the following steps.

1. Blow up the bad points, until there are only good points.
2. Blow up curves, until the multiplicity drops.

Abhyankar shows that step 1 can be done with finitely many blowing ups of bad points. Step 2 works because of the definition of good points.

4.4.2 Zariski's method

Zariski's method (described in [Lip75]) uses not just blowing ups, but also normalization.

An essential observation is that a normal surface can only have isolated singularities. Zariski shows that a surface can be desingularized by alternating the following two steps.

1. Normalize the surface.

2. If there is an isolated singular point, blow it up.

There are several algorithms [Vas91, Sei75] for normalization of surfaces (or varieties of arbitrary dimension). Normalization is a very efficient way to get rid of curves in the singular locus. Zariski's method is considered the most efficient method for the resolution of surface singularities (see [Gre91]).

A technical problem might be that the normalization does not give an embedding into 3-space. Therefore, one has to look for another convenient way of representing the intermediate surfaces, in order to apply the algorithm recursively.

5 How the Goal could be Reached

5.1 Without Resolution of the Singularities

We propose to first think of an adjoint computation algorithm without resolution of the singularities.

We know that such algorithms are possible in the curve case. For instance, adjoints of curves may be computed using the Newton-Puiseaux algorithm for expanding algebraic functions into fractional power series (see [vH94],[vH97]). It was remarked [Vol94] that one step in the Newton-Puiseux algorithm 'corresponds' to several blowing up steps in the resolution, and this is why expansion into fractional power series is faster than resolving by blowing ups.

It was shown [ALR88] that the Newton-Puiseaux algorithm can be generalized to particular surface singularities, namely quasi-ordinary singularities. For a surface singularity which is not quasi-ordinary, there might also exist an expansion in fractional power series. We cannot apply the Newton-Puiseaux algorithm, but there are more general methods available [McD95]. Therefore, fractional power series provide a reasonable approach for the computation of adjoints for surfaces.

Another promising approach for adjoints computation is to approximate A_m from below, starting with the m-th power of the ideal B_m generated by the partial derivatives of the equation F. It is known that $B_m \subset A_m$ ([Zar71]). One has to find some algebraic way of adding more elements on the left hand side, until the generated ideal coincides with A_m in the prescribed degree. Such 'completion algorithms' have been quite successfull in computational algebra, for instance for the computation of Gröbner bases [Buc65], or for the computation of integral closures [Tra84].

5.2 Using some Resolution Algorithm

If the above approach fails, then we propose to analyze the most promising existing algorithms: Hironaka's strategy [Hir84], Abhyankar's strategy [Abh90], Zariski's strategy [Lip75], from a computational point of view. Careful analysis could yield a bound for the computational complexity. One can also expect that such an analysis would lead to a computational improvement of the existing method.

Several technical problems have to be addressed. One has already been discussed in section 3.2.1: how to perform blowing ups at smooth curves? Schicho's solution in [Sch95] is acceptable for small inputs, but has very bad asymptotic behavior. A different method must be found.

Another technical problem is how to perform normalizations (for Zariski's strategy only). The normalized surface may not be embedded in 3-space. Therefore, one has to give up the idea of representing surfaces by their equation in 3-space. In order to devise a recursive algorithm, one has to find a convenient representation of surfaces which also applies to normalized surfaces.

Still another technical problem is how to deal with algebraic numbers. They can always be eliminated in the final result, but they are essential in the intermediate computation. This problem, however, has been addressed by many authors [Loo75, Wan76, Lan85, DDD85, Sme89, Duv94] and there are several solutions for it.

5.3 Risks of the two Approachs

The risk of the first approach - computing adjoints without resolving the singularities - is obvious: since adjoints are defined via a desingularization, it is not clear that there be an alternative way of computing them without a desingularization. Even though there are alternative adjoints computation algorithms for curves, it might be the case that there are none for surfaces.

If the first approach turns out to be impossible, then we still can proceed following the second approach: computing adjoints using an existing resolution algorithm.

Both approaches bear the risk that the problem itself could be too hard. Even if some bound for the computational complexity could be established, it might be unacceptably high. It is not clear that the problem of adjoints computation must have an efficient (say, polynomial or almost polynomial) solution. If such a solution would not be found, then the goal of giving an efficient algorithm would have been missed.

However, a similar risk is inherently connected with any kind of research. If the answer were clear from the beginning, then no research would be necessary. But we have the strong intuition that an efficient algorithm for adjoints computation is possible (otherwise we would not have proposed this project).

6 Qualification of the Researchers

6.1 Hoon Hong

Hoon Hong is qualified as the project leader because he has already managed to finish several research projects successfully (see the attached curriculum). He also was the advisor of 2 PhD students and 4 diploma students. He also gave courses on algebraic constraint solving, computer programming and parallel symbolic computation at the university of Linz. He also is the author of several scientific papers (see the attached list of publications).

6.2 Josef Schicho

Josef Schicho is qualified to do the main research because he is the leading researcher in the algorithmic theory of adjoints. He already did a good groundwork for this project in his PhD thesis, under the advise of Prof. Jochen Pfalzgraf (a postscript file is available at `http://www.risc.uni-linz.ac.at/www/people/jschicho`). There, he gave an algorithm for adjoints computation of surfaces. The algorithm is still inefficient, but it proves that adjoints can be computed effectively.

The main topic of Schicho's thesis is an important application of adjoints, namely a parameterization algorithm for algebraic surfaces. Therefore, he also gave evidence for the usefulness of the method of adjoints.

Schicho has done research in the field of constructive algebraic geometry for several years. He is the author of several scientific papers (see the attached list of publications). He also has taught a course on algebraic surfaces on the University of Linz.

References

[AB87] S. S. Abhyankar and B. Bajaj. Automatic parametrization of curves and surfaces III. *Computer Aided Geometric Design*, 5-4:309–323, 1987.

[Abh69] S. S. Abhyankar. Resolution of singularities of algebraic surfaces. In *Algebraic Geometry*, pages 1–11. Oxford Univ. Press, 1969.

[Abh90] S. S. Abhyankar. *Algebraic Geometry for Scientists and Engineers*. Number 35 in Mathematical Surveys and Monographs. American Mathematical Society, Providence, Rhode Island, 1990.

[AGR95] C. Alonso, J. Gutierrez, and T. Recio. Reconsidering algorithms for real parametric curves. *Applicable Algebra in Engineering, Communication and Computing*, 6:345–352, 1995.

[ALR88] M. E. Alonso, I. Luengo, and M. Raimondo. An algorithm on quasi-ordinary polynomials. *Applicable Algebra in Engineering, Communication and Computing*, 6:59–73, 1988.

[Bar85] R. E. Barnhill. Surfaces in computer-aided geometric design: a survey with new results. *Comp. Aided Geom. Design*, 2:1–17, 1985.

[BBL95] P. Blass, J. Blass, and J. Lang. Zariski surfaces. *ULAM*, 2, 1995.

[BCL82] B. Buchberger, G. E. Collins, and R. Loos, editors. *Computer Algebra: Symbolic and Algebraic Computation*. Springer-Verlag, Vienna-New York, 2 edition, 1982.

[BCX94] C. Bajaj, J. Chen, and G. Xu. Free form surface design with a-patches. In *Proc. Graph. Interface 94 Banff*, 1994.

[BFK84] W. Böhm, G. Farin, and J. Kahmann. A survey of curve and surface methods in cagd. *Comp. Aided Geom. Design*, 1:1–60, 1984.

[BH91] B. Buchberger and H. Hong. Speeding-up Quantifier Elimination by Gröbner Bases. Technical Report 91-06, RISC-Linz, Johannes Kepler University, Linz, Austria, 1991.

[BL79] P. Blass and J. Lipman. Remarks on adjoints and arithmetic genera of algebraic varieties. *Amer. J. Math.*, 101:331–336, 1979.

[BL87] P. Blass and J. Lang. *Zariski surfaces*. Marcel Dekker, 1987.

[BM91] E. Bierstone and P. Milman. A simple constructive proof of canonical resolution of singularities. In A. C. Lopez and L. N. Macarro, editors, *Effective methods in algebraic geometry*, pages 11–30. Birkhaeuser, 1991.

[BT71] W. S. Brown and J. F. Traub. On Euclid's algorithm and the theory of subresultants. *Journal ACM*, 18(4):505–514, October 1971.

[Buc65] B. Buchberger. *An Algorithm for Finding a Basis for the Residue Class Ring of a Zero-Dimensional Polynomial Ideal*. PhD thesis, Universitat Innsbruck, Institut fur Mathematik, 1965. German.

[Buc85] B. Buchberger. Groebner bases: An algorithmic method in polynomial ideal theory. In N. K. Bose, editor, *Recent Trends in Multidimensional Systems Theory*, chapter 6. D. Riedel Publ. Comp., 1985.

[BW93] T. Becker and V. Weispfenning. *Gröbner bases - a computational approach to commutative algebra*. Graduate Texts in Mathematics. Springer Verlag, New York, Berlin, Heidelberg, 1993.

[Cas96] G. Castelnuovo. Sulle superficie di genere zero. In *Memoria scelte*, pages 307–334. 1896.

[Col67] G. E. Collins. Subresultants and reduced polynomial remainder sequences. *Journal ACM*, 14:128–142, 1967.

[Con39] F. Conforto. *La superfici razionali.* Zanichelli, 1939.

[DDD85] J. Della Dora, C. Dicrescenzo, and D. Duval. About a new method for computing in algebraic number fields. In *Proc. EUROCAL 85*, volume 204 of *Springer Lecture Notes in Computer Science*, pages 289–290, Berlin, Heidelberg, New York, Tokyo, 1985. Springer-Verlag.

[Duv94] Dominique Duval. Algebraic numbers: An example of dynamic evaluation. *J. Symbolic Computation*, 18:429–445, 1994.

[Eis93] D. Eisenbud. Open problems in computational algebraic geometry. In *Cortona 1991*, volume 34 of *Proc. Symp. Math.*, pages 49–70, 1993.

[Enr49] F. Enriques. *La superfici algebriche.* Zanichelli, 1949.

[Far88] G. F. Farin. *Curves and Surfaces for Computer Aided Geometric Design: A practical guide.* Academic Press, New York, 1988.

[GH78] P. Griffiths and J. Harris. *Principles of algebraic geometry.* John Wiley, 1978.

[Gre91] G.-M. Greuel. Deformation und Klassifikation von Singularitäten. Technical report, Univ. Kaiserslautern, 1991.

[Har77] R. Hartshorne. *Algebraic geometry.* Springer–Verlag, New York, Heidelberg, Berlin, 1977.

[Hir84] H. Hironaka. Desingularization of excellent surfaces. In Vincent Gossart, Jean Giraud, and Ulrich Orbanz, editors, *Resolution of surface singularities*, pages 99–132. Springer, 1984.

[Hof89] C. M. Hoffmann. *Geometric and Solid Modelling – an Introduction.* Morgan Kauffmann Publisher, San Mateo, California, 1989.

[Hon90] H. Hong. *Improvements in CAD–based Quantifier Elimination.* PhD thesis, The Ohio State University, 1990.

[HON92a] Hoon HONG. Half Resultant. Technical Report 92-51, RISC-Linz, Johannes Kepler University, Linz, Austria, 1992.

[HON92b] Hoon HONG. Slope Resultant. Technical Report 92-52, RISC-Linz, Johannes Kepler University, Linz, Austria, 1992.

[HON95] Hoon HONG. Multivariate Resultants Under Composition. Technical Report 95-56, RISC-Linz, Johannes Kepler University, Linz, Austria, 1995.

[Hon96a] H. Hong. Groebner basis under composition i. *Journal of Symbolic Computation*, 1996. To appear.

[Hon96b] H. Hong. Groebner basis under composition ii. In *Proceedings of ISSAC 96 (International Symposium on Symbolic and Algebraic Computation)*, pages 79–85. ACM, 1996.

[Jun08] H. Jung. Darstellung der Funktionen eines algebraischen Körpers zweier unabhängiger Veränderlicher. *J. Reine Angew. Math.*, 133:289–314, 1908.

[Kal95] M. Kalkbrener. An algorithm for solving systems of algebraic equations in three variables. In J. Pfalzgraf and D. Wang, editors, *Automated practical reasoning*, pages 81–90. Springer, 1995.

[Koz94] Dexter Kozen. Efficient resolution of singularities of plane curves. In *Foundations of software technology and theoretical computer sc ience, Madras*, LNCS 880, pages 1–11. Springer, 1994.

[Kur82] H. Kurke. *Vorlesungen über algebraische Flächen*. Teubner, 1982.

[Lan74] S. Lang. Higher dimensional diophantine problems. *Bull. AMS*, 80:779–787, 1974.

[Lan85] S. Landau. Factoring polynomials over algebraic number fields. *SIAM Journal on Computing*, 14(1):184–195, February 1985.

[Lip75] J. Lipman. Introduction to the resolution of singularities. In *Arcata 1974*, volume 29 of *Proc. Symp. Pure Math*, pages 187–229, 1975.

[Lip78] J. Lipman. Desingularization of two-dimensional schemes. *Ann. Math.*, 107:151–207, 1978.

[Loo75] R. G. K. Loos. A constructive approach to algebraic numbers. *SIAM Journal on Computing*, 1975.

[LP84] A. Lanteri and M. Palleschi. About the adjunction process for polarized algebraic surfaces. *J. rei. angew. Math*, 352:15–23, 1984.

[Maz94] B. Mazur. Questions of decidability and indecidability in number theory. *J. Symb. Logic*, 59:353–371, 1994.

[McD95] J. McDonald. Fiber polytopes and fractional power series. *J. Pure Appl. Algebra*, 104:213–233, 1995.

[Mis93] B. Mishra. *Algorithmic Algebra*. Springer-Verlag, New York, 1993.

[MSW96] M. Mňuk, J. R. Sendra, and F. Winkler. On the complexity of parametrizing curves. *Beitr. Algebra und Geometrie*, 37/2, 1996.

[N̈71] M. Nöether. Zur Theorie eineindeutigen Entsprechens höherer Raumgebilde. *Math.Ann.*, 4, 1871.

[Nie93] G. M. Nielson. Cagd's top ten: what to watch. *IEEE Comp. Graphics and Appl.*, 1993.

[Oli90] C. Oliva. On the pluriadjoint maps of polarized normal gorenstein surfaces. In *Proc. L'Aquila 1988 on algebraic geometry*, pages 180–183, 1990.

[Pal90] M. Palleschi. On the adjoint line bundle to an ample and spanned one. In *Proc. L'Aquila 1988 on algebraic geometry*, pages 184–190, 1990.

[Pot96] H. Pottmann. Applications of laguerre geometry in cagd. Technical Report 30,31, Techn. Univ. Wien, 1996.

[QD87] D. Qiulin and B. J. Davies. *Surface engineering geometry for computer-aided design and manufacture*. John Wiley, 1987.

[Sch95] J. Schicho. *Rational parametrization of surfaces*. PhD thesis, RISC Linz, 1995.

[Sch97] J. Schicho. Rational parameterization of surfaces. *Journal of Symbolic Computation*, 1997.

[Sei75] A. Seidenberg. Construction of the integral closure of a finite integral domain. *Proc. Amer. Soc. Math.*, 52:368–372, 1975.

[Ser57] J.-P. Serre. Critere de rationalite pour les surfaces algebrique, apres k. kodaira. *Sem. Bourbaki*, 146, 1957.

[Sha65] I. R. Shafarevich, editor. *Algebraic surfaces*. Proc. Steklov Inst. Math., 1965. transl. by AMS 1967.

[Sha74] I. R. Shafarevich. *Algebraic geometry*. Springer, 1974.

[Sme89] Trevor J. Smedley. *Fast Methods for Computation with Algebraic Numbers*. PhD thesis, University of Waterloo, 1989. Univ. of Waterloo Computer Science Tech. Report No. CS-90-12.

[Som79] A. Sommese. Hyperplane sections of projective surfaces i - the adjunction mapping. *Duke Math. J.*, 46:377–401, 1979.

[SS87] T. W. Sederberg and J. P. Snively. Parametrization of cubic algebraic surfaces. In *The mathematics of surfaces*, pages 299–341. Clarendon Press, 1987.

[SvdV87] A. Sommese and A. van de Ven. On the adjunction mapping. *Math. Ann.*, 278:593–603, 1987.

[SW91] J.R. Sendra and F. Winkler. Symbolic parametrization of curves. *Journal of Symbolic Computation*, 12(6):607–632, 1991.

[Tei90] J. Teitelbaum. The computational complexity of the resolution of plane curve singularities. *Math. Comp.*, 54, 1990.

[Tra84] B. Trager. *Integration of algebraic functions.* PhD thesis, MIT, 1984.

[TW97] Quoc-Nam Tran and F. Winkler. *CASA Reference Manual.* http://www.risc.uni-linz.ac.at/software/casa, 1997.

[Vas91] W. Vasconcelos. Computing the integral closure of an integral domain. *Proc. Amer. Math. Soc.*, 113:633–638, 1991.

[vH94] M. van Hoeij. Computing parametrizations of rational algebraic curves. In *ISSAC-94*, pages 187–190. ACM Press, 1994.

[vH97] M. van Hoeij. Rational parametrizations of rational algebraic curves. *Journal of Symbolic Computation*, 1997. to appear.

[Vil89] O. Villamayor. Constructiveness of Hironaka's resolution. *Ann. Scient. Ecole Norm. Sup. 4*, 22:1–32, 1989.

[Vol94] E. Volcheck. Computing in the jacobian of a plane algebraic curve. In *Proc. Alg. Num. Theory I*, pages 221–233. Springer, 1994.

[Wal35] R. J. Walker. Reduction of the singularities of an algebraic surface. *Ann. Math.*, 36:336–365, 1935.

[Wal93] B. Wall. *Symbolic computation with algebraic sets.* PhD thesis, RISC-Linz, 1993.

[Wan76] P. S. Wang. Factoring multivariate polynomials over algebraic number fields. *Math. Comp.*, 32(144):324–336, October 1976.

[Zar39] O. Zariski. The reduction of singularities of an algebraic surface. *Ann. Math.*, 40:639–689, 1939.

[Zar58] O. Zariski. Castelnuovos criterion of rationality $p_a = P_2 = 0$ of an algebraic surface. *Ill. J. Math.*, 2:303–315, 1958.

[Zar71] O. Zariski. *Algebraic surfaces.* Springer, 2 edition, 1971.