CONTRIBUTIONS TO GENERAL ALGEBRA 12
Proceedings of the Vienna Conference, June 3–6, 1999
Verlag Johannes Heyn, Klagenfurt 2000

ADVANCES AND PROBLEMS IN ALGEBRAIC COMPUTATION

FRANZ WINKLER

ABSTRACT. In the last years there has been dramatic progress in all areas of algebraic computation. Many working mathematicians have access to and actually use algebraic software systems for their research. No end of this development is in sight. We report on some active topics in algebraic computation, namely the theory of integration and symbolic solution of systems of differential equations, the solution of systems of algebraic equations, the factorization of polynomials, and the design and analysis of algebraic curves and surfaces.

1. INTRODUCTION

The enormous development of scientific computation has placed mathematical capabilities at the fingertips of scientists and engineers. It has also provided mathematicians with a wealth of interesting new problems and research topics. There are, of course, different approaches to scientific computation. If the goal is to create approximate solutions to inexactly stated problems, we will probably apply the methods of numerical analysis and computation. Visualization in scientific computation is gaining ever more importance, and the mathematics behind some of the visualization algorithms is rather involved.

But there are also many important problems, in which we need to do exact computation, compute and reason about symbolic objects such as mathematical equations, algebraic curves, logical formulae, programs in a programming language, and similar objects. In the last years there has been dramatic progress in all areas of algebraic computation, both in algorithm development and in the design of program systems such as Maple [Char92], Mathematica [Wolf91], and Reduce [MacC91]. Many working mathematicians have access

Received by the editors October 10, 1999.

to and actually use algebraic software systems for their research. Even hand-held computers are capable of performing symbolic algebraic computations nowadays. No end of this development is in sight.

In Section 2 we report on some achievements in symbolic computation in the last few decades, such as the algebraic theory of integration and symbolic solution of systems of differential equations, the solution of systems of algebraic equations, the factorization of polynomials, and the design and analysis of algebraic curves and surfaces. In Section 3 we outline some currently active research topics in these and related areas.

2. Advances in algebraic computation

The research area of computer algebra was created about 40 years ago. Much of the mathematical development in making ever more parts of algebra treatable by computers was centered around the creation of software systems for algebraic computation. Some of the most important of these systems, started in the pioneering time of computer algebra, are Macsyma, SAC, Reduce, and Derive. Later starters are Maple, Mathematica, Axiom, and Magma. All important algorithmic ideas of computer algebra sooner or later have been implemented in these systems.

Of course, we cannot attempt to present a unified picture of computer algebra or algebraic computation in this limited space. We can only point to some of the more interesting achievements. For a general introduction to the mathematical background, we refer the reader to books such as [Bron97], [GeCL92], [Mign92], [Wink96].

2.1. **Integration theory.** Algorithmic methods for indefinite integration go back a long way, certainly to the work of Abel and Liouville in the 19th century. In his book [Ritt48] J.F. Ritt began to apply new algebraic techniques to the problem of integration in finite terms. But it took until the 1960's and 1970's for a decisive breakthrough to be achieved and decision procedures for indefinite integration to be implemented in software systems for computer algebra.

Let us start by considering rational functions. I.e. for given $p(x), q(x) \in \mathbb{Q}[x]$ we want to compute

$$\int \frac{p(x)}{q(x)} dx \ .$$

By partial integration and the Hermite reduction process we can determine a rational function $g(x)/h(x)$ and a polynomial $p^*(x)$ such that the integration problem can be rewritten as

$$\int \frac{p(x)}{q(x)} dx \ = \ \frac{g(x)}{h(x)} \ + \ \int \frac{p^*(x)}{q^*(x)} dx,$$

where $q^*(x)$ is the square-free part of $q(x)$, i.e. the multiplicity of every factor is reduced to 1, and $\deg(p^*) < \deg(q^*)$.

The integral $\int p^*/q^*$ can be computed in the following well–known classical way: Let $q^*(x) = (x-\alpha_1)\cdots(x-\alpha_n)$, where $\alpha_1, \ldots, \alpha_n$ are the distinct roots of q^*. Then

$$\int \frac{p^*(x)}{q^*(x)}dx \;=\; \sum_{i=1}^{n}\int \frac{c_i}{x-\alpha_i}dx \;=\; \sum_{i=1}^{n} c_i \log(x-\alpha_i)\ ,$$

$$\text{with} \qquad c_i = \frac{p^*(\alpha_i)}{q^{*\prime}(\alpha_i)}, \quad 1 \le i \le n.$$

No part of this sum of logarithms can be a rational function. According to this approach we have to compute the splitting field of the square-free denominator $q^*(x)$, which might be of high algebraic extension degree over the original coefficient field.

Example 2.1.1: Let us integrate $x/(x^2-2)$ according to this process. The denominator is square-free, and the numerator has lower degree than the denominator. So our integrand already has the form $p^*(x)/q^*(x)$ as considered above. Thus, we get

$$\begin{aligned}\int \tfrac{x}{x^2-2}dx &= \int \tfrac{1/2}{x-\sqrt{2}}dx \;+\; \int \tfrac{1/2}{x+\sqrt{2}}dx \\ &= \tfrac{1}{2}(\log(x-\sqrt{2}) + \log(x+\sqrt{2})) \;=\; \tfrac{1}{2}\log(x^2-2).\end{aligned}$$

So obviously we do not always need the full splitting field of q^* in order to express the integral of h/q^*. □

Of course there is no theoretical problem with working in algebraic extensions of the ground field for expressing the integral. But high degree algebraic extensions (or multiple extensions, for that matter) are costly in terms of computational complexity. So the problem arises of representing the result, and also the intermediate computations, over the algebraic extension field of lowest possible degree. The following theorem, which has been independently discovered by M. Rothstein and B. Trager in 1976, answers the question of what is the smallest field in which we can express the integral of a rational function.

Theorem 2.1.1: *Let $p, q \in \mathbb{Q}[x]$ be relatively prime, q monic and squarefree, and $\deg(p) < \deg(q)$. Let*

$$\int \frac{p}{q}dx \;=\; \sum_{i=1}^{n} c_i \log v_i,$$

where the c_i are distinct non–zero constants and the v_i are monic squarefree pairwise relatively prime elements of $\overline{\mathbb{Q}}[x]$ ($\overline{\mathbb{Q}}$ is the algebraic closure of $\mathbb{Q}$, i.e. the field of algebraic numbers). Let c be a new variable. Then the c_i are the *distinct roots of the resultant of* $p(x) - c \cdot q'(x)$ *and* $q(x)$ *w.r.t.* x*, and the* v_i *corresponding to* c_i *are* $v_i = \gcd(p(x) - c_i \cdot q'(x), q(x))$.

Example 2.1.1. (continued) We apply Theorem 2.1.1. $r(c) = \mathrm{res}_x(p - cq', q) = \mathrm{res}_x(x - c(2x), x^2 - 2) = -2(2c-1)^2$. There is only one root of $r(c)$, namely $c_1 = 1/2$. We get the argument of the corresponding logarithm as $v_1 = \gcd(x - \frac{1}{2}(2x), x^2 - 2) = x^2 - 2$. So

$$\int \frac{x}{x^2-2} dx \;=\; \frac{1}{2}\log(x^2-2).$$

So we arrive at the final answer over $\mathbb{Q}$ without having to take the detour via $\mathbb{Q}(\sqrt{2})$, the splitting field of $x^2 - 2$ over $\mathbb{Q}$. □

Example 2.1.2: Let us determine

$$\int \frac{x^4 - 3x^2 + 6}{x^6 - 5x^4 + 5x^2 + 4} dx.$$

The denominator $q(x) = x^6 - 5x^4 + 5x^2 + 4$ is squarefree, and actually irreducible over $\mathbb{Q}$. The resultant in the Rothstein-Trager theorem is

$$\mathrm{res}_x(x^4 - 3x^2 + 6 - c(6x^5 - 20x^3 + 10x),\ x^6 - 5x^4 + 5x^2 + 4) \;=\; 45796(4c^2+1)^3.$$

So we can express the integral in $\mathbb{Q}(i)$, namely as

$$\int \frac{x^4 - 3x^2 + 6}{x^6 - 5x^4 + 5x^2 + 4} dx \;=\; \frac{i}{2}\log(x^3 + ix^2 - 3x - 2i) - \frac{i}{2}\log(x^3 - ix^2 - 3x + 2i).$$

□

In fact, the factorization of the resultant in the Rothstein-Trager theorem can be avoided, if one uses the subresultant algorithm for computing the resultant.

Much of the theory of integration of rational functions can be transferred to integration of functions which can be expressed in elementary Liouville extensions of a field of constants. We start with a differential field C (of constants) with differentiation operation $'$, i.e. $c' = 0$ for all $c \in C$. Next we adjoin a variable of differentiation, i.e. a transcendental element x with $x' = 1$. Now we are allowed to extend the field by so-called *simple elementary extensions* i.e. by adjoining algebraic elements, logarithms (i.e. $\theta = \log \eta$ iff $\theta' = \eta'/\eta$ for some non-zero η), or exponentials (i.e. $\theta = \exp\eta$ iff $\theta'/\theta = \eta'$ for some η). A differential field L is an *elementary Liouville extension* of $C(x)$, iff there exists a tower of differential field extensions

$$C(x) = F_0 \subset F_1 \subset \cdots \subset F_n = L,$$

such that F_i is a simple elementary extension of F_{i-1} for all $1 \le i \le n$. The algorithmic theory of integration in elementary Liouville extensions is based on the following theorem.

Theorem 2.1.2 (Strong Liouville Theorem): *Let L be an elementary Liouville extension of the differential field K and let C be the field of constants of K. Let $f \in K$. If there exists an elementary Liouville extension L of K and $g \in L$ such that $g' = f$ (i.e. f has an integral in L), then there are $v \in K$, $c_1, \ldots, c_n \in \overline{C}$, and $v_1, \ldots, v_n \in K(c_1, \ldots, c_n)^*$ such that*

$$f = v' + \sum_{i=1}^{n} c_i \frac{v_i'}{v_i}$$

(i.e. the integral can be expressed in a purely logarithmic extension).

The first purely algorithmic proof of Liouville's Theorem was published by Rosenlicht [Rose68], and the first complete integration algorithm for purely transcendental elementary functions by Risch [Risc69]. The case of algebraic functions was treated by Davenport [Dave81] and Trager [Trag84]. Most computer algebra systems nowadays contain an implementation of the Risch integration algorithm. It is important to note that the Risch algorithm is NOT a collection of heuristics for possibly finding an integral of a given function, but it is actually a decision algorithm for integrability (in elementary Liouville extensions).

Example 2.1.3: By application of the Risch algorithm we see for instance that

$$\int \frac{3x^3 + 9x^2 + x - 6}{x} e^{1/x} dx \;=\; (x^3 + 5x^2 + 6x)e^{1/x} + \text{const},$$

and we get that

$$\int \frac{1}{(\log x)^2 - x^2} dx$$

is not elementary, i.e. there is no elementary Liouville extension of the field $\mathbb{C}(x, \log x)$, in which this integral can be expressed. □

2.2. **Solution of systems of algebraic equations.** We consider a system of algebraic equations

$$\begin{aligned} f_1(x_1, \ldots, x_n) &= 0, \\ &\vdots \\ f_m(x_1, \ldots, x_n) &= 0, \end{aligned} \tag{2.2.1}$$

over some field K, i.e. $f_i \in K[x_1, \ldots, x_n]$. Let $\overline{K}$ be the algebraic closure of K, and $\mathbb{A}^n(\overline{K}) = \mathbb{A}^n$ the n-dimensional affine space over $\overline{K}$. A root $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{A}^n$ of (2.2.1) is also a root of any linear combination of the f_i's, i.e. of any element of the ideal $I = \langle f_1, \ldots, f_m \rangle$ generated by the f_i's. So when we are studying solutions of systems of algebraic equations, we are actually studying common solutions for all elements of a polynomial ideal. On the other hand, because of Hilbert's basis theorem, every ideal in $K[x_1, \ldots, x_n]$ is generated by a finite basis. So the common solutions of any polynomial ideal I are actually the solutions of a finite system of algebraic equations.

The collection of points in $\mathbb{A}^n$ satifying (2.2.1) is a so-called algebraic set (or variety), and we denote it by $V(I)$. If $V(I)$ consists of only finitely many points, i.e. its dimension is 0, then we also say that I is a 0-dimensional ideal. In this section we assume that the ideal I generated by the polynomials in (2.2.1) is 0-dimensional (although much of this theory applies also to higher dimensional ideals). The problem we are considering in this section is the following.

Problem "Solution of 0-dimensional system of algebraic equations":

given: $I = \langle f_1, \ldots, f_m \rangle \subseteq K[x_1, \ldots, x_n]$, 0-dimensional,
find: all solutions of I in $\mathbb{A}^n$.

What we actually want are the *elimination ideals* of I, i.e.

$$I_k = I \cap K[x_1, \ldots, x_k], \qquad \text{for } 1 \leq k \leq n,$$

the ideals consisting of all those polynomials in I just depending on the first k variables. Having determined these elimination ideals, we can successively solve for the variables. So the determination of the elimination ideals plays the same rôle for nonlinear algebraic equations as the Gaussian algorithm plays for linear equations.

The method of resultants:

Let R be a commutative ring, $f(x), g(x) \in R[x]$ two univariate polynomials over R. The resultant of f and g, $h = \mathrm{res}(f, g)$, is an element of $\langle f, g \rangle$, and $\mathrm{res}(f, g) = 0$ if and only if f and g have a common factor (ref. [vdWa91],[Wink96], [CoLO98]).

Example 2.2.1: Consider the following system of equations:

$$\begin{aligned} f_1(x,y) &= 2x^4 - 3x^2y + y^4 - 2y^3 + y^2 &&= 0, \\ f_2(x,y) = \tfrac{\partial}{\partial x} f_1(x,y) &= 8x^3 - 6xy &&= 0. \end{aligned}$$

The solutions of this system are those points on the tacnode curve (see Fig. 2.4.3), which are either singular or have a vertical tangent. We are looking for the solutions in the plane over an algebraically closed field containing the field of definition $\mathbb{Q}$, i.e. over $\mathbb{C}$ or actually over $\overline{\mathbb{Q}}$, the field of algebraic numbers. The resultant w.r.t. x is

$$r(y) = \mathrm{res}_x(f_1, f_2) = (y^4 - 2y^3 + y^2)(64y^4 - 128y^3 - 8y^2)^2.$$

$r(y)$ has the roots

$$y = 0,\ 1,\ 1 + \frac{3}{4}\sqrt{2},\ 1 - \frac{3}{4}\sqrt{2}.$$

If, for instance, we substitute $1 + \frac{3}{4}\sqrt{2}$ for y in f_1 and f_2, we get

$$x = \frac{1}{4}\sqrt{12 + 9\sqrt{2}}.$$

So

$$(1 + \frac{3}{4}\sqrt{2},\ \frac{1}{4}\sqrt{12 + 9\sqrt{2}})$$

is one of the roots of this system of algebraic equations. □

This works perfectly for equations in 2 variables. For more variables, there can be "extraneous factors" of the resultant, i.e. solutions of the resultant, which cannot be continued to solutions of the given system.

Example 2.2.2: Consider the system

$$\begin{aligned}
f_1(x, y, z) &= 2xy + yz - 3z^2 &&= 0,\\
f_2(x, y, z) &= x^2 - xy + y^2 - 1 &&= 0,\\
f_3(x, y, z) &= yz + x^2 - 2z^2 &&= 0.
\end{aligned}$$

We compute

$$\begin{aligned}
a(x) &= \mathrm{res}_z(\mathrm{res}_y(f_1, f_3), \mathrm{res}_y(f_2, f_3))\\
&= x^6(x-1)(x+1)(127x^4 - 167x^2 + 4),\\
b(y) &= \mathrm{res}_z(\mathrm{res}_x(f_1, f_3), \mathrm{res}_x(f_2, f_3))\\
&= (y-1)^3(y+1)^3(3y^2-1)(127y^4 - 216y^2 + 81)(457y^4 - 486y^2 + 81),\\
c(z) &= \mathrm{res}_y(\mathrm{res}_x(f_1, f_3), \mathrm{res}_x(f_2, f_3))\\
&= z^4(z-1)(z+1)(3z^2-1)(127z^4 - 91z^2 + 16)(457z^4 - 175z^2 + 16).
\end{aligned}$$

All the solutions of the system, e.g. $(1, 1, 1)$, have coordinates which are roots of a, b, c. But there is no solution of the system having y-coordinate $1/\sqrt{3}$, although $b(1/\sqrt{3}) = 0$. So not every root of these resultants can be extended to a solution of the whole system. □

<u>The method of Gröbner bases:</u>

This method in elimination theory was invented by Buchberger in 1965. For an overview of applications and current research topics we refer to [BuWi98]. We don't want to go into details of definition and properties of Gröbner bases, this is done for instance in [Wink96]. Let us just make a few crucial remarks:

- a Gröbner basis is a particular basis for a polynomial ideal (over a field or certain other domains), depending on an "admissible" ordering of the terms or monomials,
- every polynomial ideal has a Gröbner basis,
- for every given finite basis for a polynomial ideal I, we can effectively determine, by Buchberger's algorithm or variants thereof, a finite Gröbner basis generating I, i.e. change from an arbitrary basis of I to a Gröbner basis of I,

- Buchberger's algorithm is implemented in the major computer algebra systems such as Maple, Mathematica, and Reduce.

Because of the elimination property of Gröbner bases, we can exactly determine the elimination ideals of a given ideal I by computing a Gröbner basis for I.

Theorem 2.2.1: (Elimination property) *Let $I = \langle f_1, \ldots, f_m \rangle$ be an ideal in $K[x_1, \ldots, x_n]$. Let G be a Gröbner basis for the ideal I w.r.t. the lexicographic term ordering with $x_1 < \cdots < x_n$. Then*

$$I \cap K[x_1, \ldots, x_k] \ = \ \langle G \cap K[x_1, \ldots, x_k] \rangle,$$

where the ideal on the right-hand side is generated over $K[x_1, \ldots, x_k]$.

Example 2.2.2 (continued) We are considering the system of equations

$$\begin{aligned} f_1(x,y,z) &= 2xy + yz - 3z^2 &&= 0, \\ f_2(x,y,z) &= x^2 - xy + y^2 - 1 &&= 0, \\ f_3(x,y,z) &= yz + x^2 - 2z^2 &&= 0. \end{aligned}$$

The set of polynomials $F = \{f_1, f_2, f_3\}$ generates an ideal $I = \langle f_1, f_2, f_3 \rangle$ in $\mathbb{Q}[x_1, x_2, x_3]$. The Gröbner basis for I w.r.t. the lexicographic term ordering with $x > y > z$ (i.e., we consider x as the highest variable) is

$$G = \{g_1, g_2, g_3, g_4\},$$

with

$$\begin{aligned} g_1 &= 78x - 2921z^5 + 3744z^3 - 901z, \\ g_2 &= 104y^2 - 2667z^6 + 3562z^4 - 895z^2 - 104, \\ g_3 &= 52yz - 2667z^6 + 3562z^4 - 947z^2, \\ g_4 &= 127z^7 - 218z^5 + 107z^3 - 16z. \end{aligned}$$

From this Gröbner basis G we can see immediately:

- every solution of $g_4(z) = z(z-1)(z+1)(127z^4 - 91z^2 + 16) = 0$, e.g. -1, can be extended to a solution of the system g_2, g_3, g_4, e.g. $(-1, -1)$, and every such solution can be extended to a solution of the whole system, e.g. $(-1, -1, -1)$,
- the system has 8 solutions (counted with multiplicity). This number corresponds to the 8 terms $1, y, z, z^2, \ldots, z^6$, which are not a multiple of any leading term in G,
- the 2-nd elimination ideal (eliminating x), for instance, is $\langle g_2, g_3, g_4 \rangle$. □

Although the basis G in the previous example might not look simpler than F, it has obvious advantages over F. In particular, G is triangularized, i.e. it contains one polynomial, g_4, which depends only on the least variable, z. In fact, because of the elimination property of Gröbner bases, every polynomial $g(z) \in I \cap \mathbb{Q}[z]$ is a multiple of g_4. Similarly, all the polynomials in I depending only on z and y are linear combinations of g_2, g_3, g_4 (over $\mathbb{Q}[y, z]$).

In order to decide, whether a polynomial $f(x, y, z)$ is in I, we can employ the *division algorithm*, i.e. in f we successively replace any occurrence of x by

$$\frac{1}{78}(2921z^5 - 3744z^3 + 901z),$$

any occurrence of y^2 by

$$\frac{1}{104}(2667z^6 - 3562z^4 + 895z^2 + 104),$$

any occurrence of yz by

$$\frac{1}{52}(2667z^6 - 3562z^4 + 947z^2),$$

and any occurrence of z^7 by

$$\frac{1}{127}(218z^5 - 107z^3 + 16z).$$

Obviously, if we reach 0 by this division process, we have represented f as a linear combination of the basis polynomials, i.e. $f \in I$. Conversely, w.r.t. a Gröbner basis, f must be reducible to 0 by the division algorithm (this fails to be so for an arbitrary basis).

Besides determination of elimination ideals, there are many other algebraic and geometric problems that can be successfully treated by Gröbner bases. Let us list just a few of them:

- ideal membership problem, i.e. "$f \in I$?",
- radical membership problem, i.e. "$f \in \sqrt{I}$?",
- equality of ideals, i.e. "$I = J$?",
- arithmetic of ideals, i.e. computation of $I \cap J, I : J$ ($I + J, I \cdot J$ are easy),
- computation of dimension of ideals, $\dim(I)$,
- computation of syzygies of sequences of polynomials.

Applications of the Gröbner basis method in mathematics, sciences, and engineering are collected in [TrWi00].

The method of Gröbner bases has enormous power, but despite, or rather because of, this power, there are also serious problems with Gröbner bases. Some of them need further research, and can, perhaps, be overcome. Others stem from intrinsic limitations. We just list a few of these problems and limitations:

- Given any basis for an ideal I, Buchberger's algorithm successively adds certain new elements of I to the basis, such that ultimately the basis becomes a Gröbner basis, i.e. the ideal membership problem can be solved by the division algorithm. This process is completely insensitive to the particular problem at hand. For instance, it destroys any sparsity present in the original system F. Specially taylored Gröbner basis

algorithms for various problems in algebra and geometry are still to be developed.
- The ideal membership problem (which is solved by Gröbner bases) has complexity which is double exponential in the number of variables n [MaMe82]. So any algorithm for computing a Gröbner basis must necessarily have complexity at least double exponential in n. Buchberger's algorithm has this complexity. For special problems, for instance 0-dimensional ideals, the complexity is only single exponential in n.
- For solving algebraic equations, we want a Gröbner basis w.r.t. a lexicographic term ordering, but this is much harder to compute than a Gröbner basis w.r.t. a graduated ordering. For 0-dimensional ideals, Faugère et al. [FGLM93] developed linear algebra techniques for transforming a Gröbner basis w.r.t. some term ordering into a Gröbner basis w.r.t. a desired ordering. For ideals of arbitrary dimension Collart et al. [CoKM97] have introduced the idea of the Gröbner walk for achieving such a transformation. Although this idea works very nicely in practical applications, there are still unresolved complexity questions concerning the Gröbner walk.

2.3. **Factorization of polynomials.** The factorization of (multivariate) polynomials is one of the most frequent subproblems in algebraic computation. We have seen it occur in integration of rational functions, it allows us to simplify algebraic equation problems, we need irreducible polynomials for generating finite fields and algebraic field extensions, and for most algorithms in algebraic geometry we need the irreducibility of the algebraic variety as a prerequisite.

Because of this fundamental importance, the factorization problem has received the attention of algebraists for a long time. In 1882 L. Kronecker described a reduction algorithm for the factorization of multivariate polynomials to the factorization of univariate polynomials. His idea was to map $f(x_1, \dots, x_n)$, an n-variate polynomial over the unique factorization domain R, to the univariate polynomial $f(y, y^d, \dots, y^{d^{n-1}})$, where d is an upper bound for the degree of f w.r.t. any one of the variables. In this way, one gets a 1-1, easily computable mapping

$$\phi : R[x_1, \dots, x_n]_{/\langle x_1^d, \dots, x_n^d \rangle} \longrightarrow R[y]_{/\langle y^{d^n} \rangle}.$$

Now for every irreducible factor g of f there are irreducible factors $g_1, \dots, g_s$ of $\phi(f)$, such that $g = \phi^{-1}(\prod_{j=1}^{s} g_j)$.

Although this algorithm works perfectly fine, it has the drawback of exponentially increasing the degree of the polynomial. For this reason, whenever possible, other approaches are used nowadays, based on Berlekamp's factorization algorithm over finite fields and Hensel's lifting lemma.

In 1968 E.R. Berlekamp [Berl68] published his algorithm for factoring univariate polynomials over finite fields $\mathrm{GF}(p) = \mathbb{Z}_p$, p a prime. Let $f(x) \in \mathrm{GF}(p)[x]$ be the squarefree polynomial to be factored. Let n be the degree of f. The key step in Berlekamp's algorithm is the computation of the null-space of a certain $(n \times n)$-matrix with coefficients in $\mathrm{GF}(p)$, followed by a few gcd-computations in $\mathrm{GF}(p)[x]$. The complexity of the Berlekamp factorization algorithm is proportional to pn^3.

In 1969 H. Zassenhaus [Zass69] showed how to use Hensel's p–adic lifting lemma for factoring univariate polynomials, for instance over the integers $\mathbb{Z}$.

Theorem 2.3.1: (Hensel's Lemma) *Let p be a prime number and $f(x), f_1(x), \ldots, f_r(x) \in \mathbb{Z}[x]$. Let $(f_1 \mod p), \ldots, (f_r \mod p)$ be pairwise relatively prime in $\mathbb{Z}_p[x]$ and $f(x) \equiv \prod_{i=1}^{r} f_i(x) \mod p$. Then for every natural number k there are polynomials $f_1^{(k)}(x), \ldots, f_r^{(k)}(x) \in \mathbb{Z}[x]$ such that*

$$f(x) \equiv \prod_{i=1}^{r} f_i^{(k)}(x) \mod p^k$$

and

$$f_i^{(k)}(x) \equiv f_i(x) \mod p \text{ for } 1 \leq i \leq r.$$

This Hensel lifting process can actually be carried out algorithmically, by application of the extended Euclidean algorithm.

The basic idea of the so-called *Berlekamp–Hensel algorithm* is the following:

- given a primitive squarefree polynomial $f(x) \in \mathbb{Z}[x]$, choose a prime p not dividing the leading coefficient of f, and such that f remains squarefree modulo p;
- apply Berlekamp's algorthm for factoring f modulo p;
- apply the Hensel Lemma to this factorization, to get a factorization of f modulo p^k, for high enough p (so that within this range of coefficients the coefficients of any factor f can be represented uniquely);
- now, every irreducible factor of f in $\mathbb{Z}[x]$ must correspond to a combination of factors modulo p^k.

Example 2.3.1: Let us use the Berlekamp–Hensel algorithm for factoring the polynomial

$$f(x) = 6x^7 + 7x^6 + 4x^5 + x^4 + 6x^3 + 7x^2 + 4x + 1$$

in $\mathbb{Z}[x]$. f is squarefree and it stays squarefree modulo $p = 5$. By an application of the Berlekamp algorithm, $f(x)$ is factored modulo 5 into

$$f(x) \equiv (x - 2)(x^2 - 2)(x^2 + 2)(x^2 - x + 2) \mod 5.$$

By an application of Hensel's Lemma we lift this factorization to a factorization modulo 25, getting

$$f(x) \equiv 6 \cdot \underbrace{(x-12)}_{v_1} \cdot \underbrace{(x^2-7)}_{v_2} \cdot \underbrace{(x^2+7)}_{v_3} \cdot \underbrace{(x^2+9x-8)}_{v_4} \mod 25.$$

Suppose we know that all the integer coefficients of the factorization of f are contained in the range $[-12, 12]$. So then, we check whether any one of the factor candidates, together with a factor of 6, gives us a factor over the integers. In fact, $2 \cdot v_1 \equiv 2x+1 \mod 25$, and this is indeed a factor. v_2 and v_3 do not lead to such a factor, but $3 \cdot v_4 \equiv 3x^2+2x+1 \mod 25$ does. Now we try to combine v_2 and v_3, and indeed $v_2 \cdot v_3 \equiv x^4+1 \mod 25$ leads to a factor. Thus, we have found the factorization of $f(x)$ in $\mathbb{Z}[x]$, namely

$$f(x) = (2x+1) \cdot (x^4+1) \cdot (3x^2+2x+1). \qquad \square$$

The idea of the Hensel lifting can be generalized to multivariate polynomials. Because of the final factor combination step, the Berlekamp-Hensel algorithm still has exponential worst case complexity. In 1982 Lenstra, Lenstra, and Lovász [LeLL82] described an algorithm for factoring polynomials in $\mathbb{Z}[x]$ in polynomial time. The LLL algorithm is based on a process for finding shortest vectors in lattices.

Once we can factor polynomials over a field K, we can also factor over any algebraic extension of K. An algorithm based on norm computation is described in [vdWa91]. But this is still not the end of the story. Given a bivariate (or multivariate) polynomial $f(x,y) \in K[x,y]$, we might want to determine, whether it can be factored over any algebraic extension of K, i.e. whether it can be factored in $\overline{K}[x,y]$. This problem of factoring over the algebraic closure of the given ground field is called *absolute factorization.* In recent years several people have suggested algorithms for absolute factorization. For an overview see [Wink96]. Many computer algebra software systems already have implementations of absolute factorization.

2.4. **Geometry of algebraic curves and surfaces.** Algebraic curves and surfaces have been studied intensively in algebraic geometry for decades and even centuries. Thus, there exists a huge amount of theoretical knowledge about these geometric objects. Recently, algebraic curves and surfaces play an important and ever increasing rôle in computer aided geometric design, computer vision, and computer aided manufacturing. Consequently, theoretical results need to be adapted to practical needs. We need efficient algorithms for generating, representing, manipulating, analyzing, rendering algebraic curves and surfaces.

One interesting subproblem is the rational parametrization of curves and surfaces. Consider an affine plane algebraic curve $\mathcal{C}$ in $\mathbb{A}^2(\overline{K})$ defined by the bivariate polynomial $f(x,y) \in K[x,y]$, i.e.

$$\mathcal{C} = \{(a,b) \mid (a,b) \in \mathbb{A}^2(\overline{K}) \text{ and } f(a,b) = 0\}.$$

Of course, we could also view this curve in the projective plane $\mathbb{P}^2(\overline{K})$, defined by $F(x,y,z)$, the homogenization of $f(x,y)$.

A pair of rational functions $(x(t), y(t)) \in \overline{K}(t)$ is a *rational parametrization* of the curve $\mathcal{C}$, if and only if $f(x(t), y(t)) = 0$ and for almost every point $(x_0, y_0) \in \mathcal{C}$ (i.e. up to finitely many exceptions) there is a parameter value $t_0 \in \overline{K}$ such that $(x_0, y_0) = (x(t_0), y(t_0))$. Only irreducible curves, i.e. curves whose defining polynomial is absolutely irreducible, can have a rational parametrization. Almost any rational transformation of a rational parametrization is again a rational parametrization, so such parametrizations are not unique.

Implicit representations (by defining polynomial) and parametric representations (by rational parametrization) both have their particular advantages and disadvantages. Given an implicit representation of a curve and a point in the plane, it is easy to check whether the point is on the curve. But it is hard to generate "good" points on the curve, i.e. for instance points with rational coordinates if the defining field is $\mathbb{Q}$. On the other hand, generating good points is easy for a curve given parametrically, but deciding whether a point is on the curve requires the solution of a system of algebraic equations. So it is highly desirable to have efficient algorithms for changing from implicit to parametric representation, and vice versa.

Example 2.4.1: Let us consider curves in the plane (affine or projective) over $\mathbb{C}$. The curve defined by $f(x,y) = y^2 - x^3 - x^2$ (see Fig. 2.4.1) is rationally parametrizable, and actually a parametrization is $(t^2 - 1, t(t^2 - 1))$.

On the other hand, the elliptic curve defined by $f(x,y) = y^2 - x^3 + x$ (see Fig 2.4.2) does not have a rational parametrization.

The tacnode curve (see Fig. 2.4.3) defined by $f(x,y) = 2x^4 - 3x^2y + y^4 - 2y^3 + y^2$ has the parametrization

$$x(t) = \frac{t^3 - 6t^2 + 9t - 2}{2t^4 - 26t^3 + 40t^2 - 32t + 9}, \quad y(t) = \frac{t^2 - 4t + 4}{2t^4 - 26t^3 + 40t^2 - 32t + 9}.$$

The criterion for parametrizability of a curve is its genus. Only curves of genus 0, i.e. curves having as many singularities as their degree permits, have a rational parametrization. $\square$

Computing such a parametrization essentially requires the full analysis of singularities (either by successive blow-ups, or by Puiseux expansion) and the determination of a regular point on the curve. We can control the quality of the resulting parametrization by controlling the field over which we choose this regular point. Thus, finding a regular curve point over a minimal field extension on a curve of genus 0 is one of the central problems in rational parametrization, compare [SeWi97], [SeWi99]. The determination of rational points on algebraic curves can be an extremely complicated problem. But for curves of genus 0 the situation can actually be controlled very well. For a curve over a field K of characteristic 0, we can determine whether the curve has a regular point over K, or otherwise find a quadratic field extension which admits such a regular point.

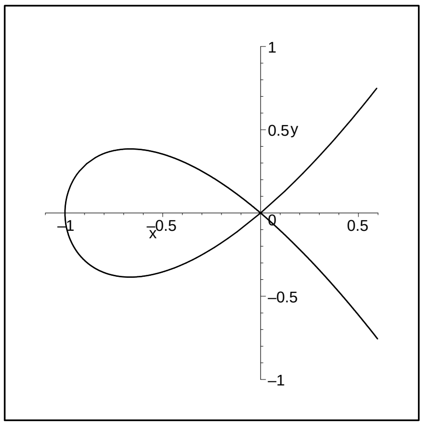

Fig. 2.4.1

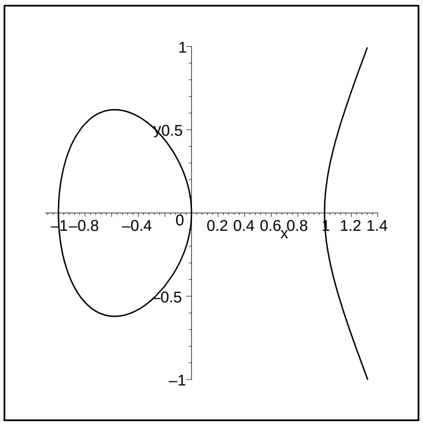

Fig. 2.4.2

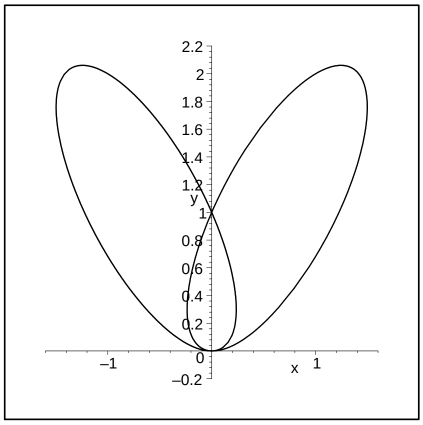

Fig. 2.4.3

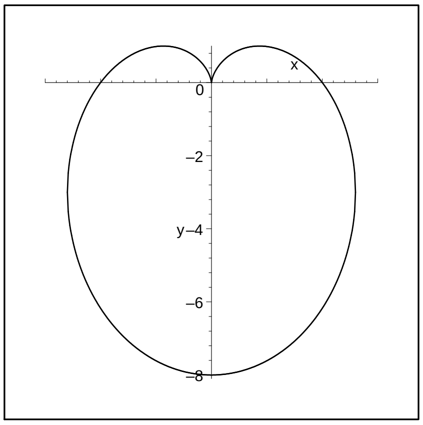

Fig. 2.4.4

Example 2.4.2: Let $\mathcal{C}$ be the curve in the complex plane defined by

$$f(x,y) = (x^2 + 4y + y^2)^2 - 16(x^2 + y^2) = 0.$$

For a picture of this curve in the real affine plane see Fig. 2.4.4.

The curve $\mathcal{C}$ has the following rational parametrization:

$$\begin{aligned} x(t) &= -32 \cdot \frac{-1024i + 128t - 144it^2 - 22t^3 + it^4}{2304 - 3072it - 736t^2 - 192it^3 + 9t^4}, \\ y(t) &= -40 \cdot \frac{1024 - 256it - 80t^2 + 16it^3 + t^4}{2304 - 3072it - 736t^2 - 192it^3 + 9t^4}. \end{aligned}$$

So, as we see in Fig.2.4.4, $\mathcal{C}$ has infinitely many real points. But generating any one of these real points from the above parametrization is not obvious. Does this real curve $\mathcal{C}$ also have a parametrization over $\mathbb{R}$? Indeed it does, let's see how we can get one.

In the projective plane over $\mathbb{C}$, $\mathcal{C}$ has three double points, namely $(0:0:1)$ and $(1:\pm i:0)$. Let $\tilde{\mathcal{H}}$ be the linear system of conics passing through all these double points. The system $\tilde{\mathcal{H}}$ has dimension 2 and is defined by

$$h(x,y,z,s,t) = x^2 + sxz + y^2 + tyz = 0,$$

i.e., for any particular values of s and t we get a conic in $\tilde{\mathcal{H}}$. Three elements of this linear system define a birational transformation

$$\begin{aligned} \mathcal{T} &= (h(x,y,z,0,1) : h(x,y,z,1,0) : h(x,y,z,1,1)) \\ &= (x^2 + y^2 + yz : x^2 + xz + y^2 : x^2 + xz + y^2 + yz) \end{aligned}$$

which transforms $\mathcal{C}$ to the conic $\mathcal{D}$ defined by

$$15x^2 + 7y^2 + 6xy - 38x - 14y + 23 = 0.$$

For a conic defined over $\mathbb{Q}$ we can decide whether it has a point over $\mathbb{Q}$ or $\mathbb{R}$. In particular, we determine the point $(1, 8/7)$ on $\mathcal{D}$, which, by $\mathcal{T}^{-1}$, corresponds to the regular point $P = (0, -8)$ on $\mathcal{C}$. Now, by restricting $\tilde{\mathcal{H}}$ to conics through P and intersecting $\tilde{\mathcal{H}}$ with $\mathcal{C}$ (for details see [SeWi97]), we get the parametrization

$$x(t) = \frac{-1024t^3}{256t^4 + 32t^2 + 1}, \quad y(t) = \frac{-2048t^4 + 128t^2}{256t^4 + 32t^2 + 1}.$$

over the reals. □

Many of these ideas which work for curves can actually be generalized to higher dimensional geometric objects. For instance, one subproblem in computer aided geometric design is the manipulation of offset curves, offset surfaces, pipe and canal surfaces. These are geometric objects keeping certain distances from a generating object. Let us just consider the case of a pipe surface in an example.

Example 2.4.3: We consider the space curve $\mathcal{C}$ in $\mathbb{A}^2(\mathbb{R})$ given parametrically by $(x(t), y(t), z(t)) = (t, t^2, t^3)$. We want to construct a parametric representation of the pipe surface $\mathcal{S}$ (at distance 1) along $\mathcal{C}$, i.e. the locus of points having normal distance 1 from $\mathcal{C}$. This pipe surface $\mathcal{S}$ is the envelope of spheres of radius 1 moving along $\mathcal{C}$, i.e. every point on $\mathcal{S}$ lies on a circle in a hypersurface perpendicular to the curve $\mathcal{C}$. If we can find a parametric representation of a curve $\tilde{\mathcal{C}}$ on $\mathcal{S}$, which meets every one of these circles, then by a pencil of lines in the corresponding hypersurface we can generate a rational representation for all the points on this circle, and thus finally a rational parametrization of the pipe surface.

Such a curve can by determined by algebraic computation, giving for instance the parametrization $(\tilde{c}_1(t), \tilde{c}_2(t), \tilde{c}_3(t))$ with

$$\begin{pmatrix} \tilde{c}_1(t) \\ \tilde{c}_2(t) \\ \tilde{c}_3(t) \end{pmatrix} = \begin{pmatrix} t + \frac{3(36t^4-13t^2-4\sqrt{5}t-5)t^2}{(1+4t^2)(21t^2+2\sqrt{5}t+5+27t^4)} \\ t^2 - \frac{3(60t^3+14t-\sqrt{5}+4\sqrt{5}t^2)t^2}{(1+4t^2)(21t^2+2\sqrt{5}t+5+27t^4)} \\ t^3 + \frac{21t^2+2\sqrt{5}t+5}{21t^2+2\sqrt{5}t+5+27t^4} \end{pmatrix}.$$

From this parametric representation of $\tilde{\mathcal{C}}$ we can compute a parametric representation of the pipe surface. □

For a geometric approach to parametrization of pipe and canal surfaces see [PePo97], an algebraic approach can be found in [HLSW99].

Now that we have seen some examples of parametrization treated by symbolic algebraic computation, let us just briefly discuss the inverse problem, namely the problem of implicitization. If we are given, for instance, a rational parametrization in $K(t)$ of a plane curve, i.e.

$$x(t) = p(t)/r(t), \quad y(t) = q(t)/r(t),$$

we essentially want to eliminate the parameter t from these relations, and get a relation just between x and y. We also want to make sure that we do not consider components for which the denominator $r(t)$ vanishes. This leads to the system of algebraic equations

$$\begin{aligned} x \cdot r(t) - p(t) &= 0, \\ y \cdot r(t) - q(t) &= 0, \\ r(t) \cdot z - 1 &= 0. \end{aligned}$$

The implicit equation of the curve must be the generator of the ideal

$$I = \langle x \cdot r(t) - p(t), y \cdot r(t) - q(t), r(t) \cdot z - 1 \rangle \ \cap \ K[x, y].$$

Using the elimination property of Gröbner bases, we can compute this generator by a Gröbner basis computation w.r.t. the lexicographic ordering based on $x < y < z < t$.

Example 2.4.4: Let us do this for the curve of Example 2.4.2. We start from the parametrization

$$x(t) = \frac{-1024t^3}{256t^4 + 32t^2 + 1}, \quad y(t) = \frac{-2048t^4 + 128t^2}{256t^4 + 32t^2 + 1}.$$

So we have to solve the equations

$$\begin{aligned} x \cdot (256t^4 + 32t^2 + 1) + 1024t^3 &= 0, \\ y \cdot (256t^4 + 32t^2 + 1) + 2048t^4 - 128t^2 &= 0, \\ (256t^4 + 32t^2 + 1) \cdot z - 1 &= 0. \end{aligned}$$

The Gröbner basis of this system w.r.t. the lexicographic ordering based on $x < y < z < t$ is

$$G = \{........, x^4 + y^4 + 8x^2y + 2x^2y^2 + 8y^3 - 16x^2\}.$$

So we have found the implicit equation of the curve. □

3. Problems in algebraic computation

After having described some of the achievements in different areas of algebraic computation, let us now point out some topics which are currently being investigated.

3.1. **Differential equations.** The integration problem discussed in Section 2.1, i.e. the problem of finding an expression y such that $\int f = y$, or $f = y'$, is a particular differential equation problem. As we have seen above, symbolic algorithmic approaches are already quite powerful in treating this integration problem. For more general differential equation systems, algorithmic methods are far more scarce.

In 1978 Kovacic developed an algorithm for computing "closed-form" solutions of 2-nd order linear homogeneous differential equations of the form

$$y''(x) + a(x) \cdot y'(x) + b(x) \cdot y(x) = 0,$$

where $a, b \in \mathbb{C}(x)$, and the solution is sought in a Liouvillian extension of $\mathbb{C}(x)$. Kovacic's algorithm decides, whether such a solution exists, and if so, constructs the solutions. So, for example, for the differential equation

$$y''(x) = \frac{4x^6 - 8x^5 + 12x^4 + 4x^3 + 7x^2 - 20x + 4}{4x^4} \cdot y(x),$$

Kovacic's algorithm determines the solution

$$\eta = \frac{x^2 - 1}{x^{3/2}} \cdot e^{-1/x + x^2/2 - x}.$$

Starting in 1991 Singer [Sing91], [SiUl93] extended Kovacic's algorithm and presented an algorithm for finding a basis for the space of Liouvillian solutions of a linear differential equation of arbitrary order

$$y^{(n)} + a_{n-1}y^{(n-1)} + \cdots + a_0 y = b,$$

where the coefficients $a_{n-1}, \ldots, a_0$ and the right hand side b are from a differential field K. Singer's algorithm works by determining differential Galois groups and is of high complexity.

The field of symbolic solution of differential equations is still wide open, in particular when partial differential equations are considered.

3.2. **Fast computation of Gröbner bases.** As we have noted in Section 2.2, the computing time for determining a Gröbner basis may vary considerably depending on the term ordering. Although there are infinitely many term orderings, cf. [Robb85], there are only finitely many different Gröbner bases for a fixed ideal I, cf. [MoRo88]. By "walking around" in the Gröbner fan of the ideal I, one may transform a Gröbner basis w.r.t. to a term ordering $<_1$ to a Gröbner basis w.r.t. a term ordering $<_2$. This idea was first developed in [CoKM97]. Practical experiments have shown considerable savings in computing time, cf. [AmGK96], [Tran98]. Recently Kalkbrener [Kalk99] has started to investigate the theoretical reasons for this practical speed-up. But we certainly need more complexity investigations for understanding these phenomena.

3.3. **Computations on algebraic curves and surfaces.** For algebraic curves over a computable field of characteristic 0 the problem of symbolic algebraic parametrization is completely solved. More precisely, we can determine whether a curve has a rational parametrization over the algebraic closure of the ground field, and in the affirmative case we can compute a proper parametrization with coefficients in an algebraic extension of the ground field of lowest extension degree. If the given irreducible curve has coefficients in $\mathbb{Q}$, is rational, and has infinitely many real points, then it actually has a parametrization over $\mathbb{R}$ which can be determined algorithmically, see [SeWi99].

But even so there remains the following open problem: a curve defined over $\mathbb{Q}$ might have a parametrization with coefficients in $\mathbb{Z}$. If so, it has infinitely many such parametrizations. How can we find the one with the smallest coefficients?

Also for algebraic surfaces the algorithmic parametrization problem is solved (with the exception of del Pezzo surfaces) in principal, see [Schi98a], [Schi98b]. But both for curves and for surfaces one of the most critical subproblems is the analysis of the singularities and the determination of adjoints. There are several theoretical approaches to these problems, e.g. blow-ups, or Puiseux series expansion. Much work is still needed for developing algorithms with good theoretically and practical complexity.

3.4. **Integration of symbolic and numerical computation.** Many problems in science and engineering are actually inexactly stated, but still we want a symbolic solution. For instance, we might have inexactly defined polynomials for two surfaces, know that they should have an intersection of dimension 2, and want to determine the defining polynomial of the intersection. The question then is actually: how can we vary the coefficients of the given polynomials slightly so that we get non-trivial intersection?

On the other hand, we might have an exact symbolic definition of, say, an algebraic curve or surface, and we want to create pixel information for rendering this object on a screen. How much algebraic computation do we really have to do for making sure that we get the topology of the object right? The remaining job should be handed over to a fast numerical approximation algorithm for filling in the regular parts of the object.

Recently Watt and Stetter [WaSt98] have collected approaches to the integration of symbolic and numerical computation. At the University of Linz, research groups in symbolic computation, numerical computation, and engineering have recently started a big cooperation project for trying to bridge the gap between these different paradigms of scientific computation. But obviously there is still a long way to go.

Acknowledgement: The author wants to acknowledge support from the Austrian *Fonds zur Förderung der wissenschaftlichen Forschung* under projects P11160-TEC (HySaX), and SFB F013/1304.

References

[AmGK96] B. Amrhein, O. Gloor, W. Küchlin, "Walking Faster", *Design and Implementation of Symbolic Computation Systems (Proc. DISCO'96)*, J. Calmet and C. Limongelli (eds.), Springer-Verlag LNCS 1128, 150–161 (1996).

[Berl68] E.R. Berlekamp, *Algebraic Coding Theory*, McGraw-Hill, New York (1968).

[Bron97] M. Bronstein, *Symbolic Integration I*, Springer–Verlag, Berlin Heidelberg (1997).

[BuWi98] B. Buchberger, F. Winkler, *Gröbner Bases and Applications*, London Math. Soc. Lecture Note Series 251, Cambridge Univ. Press (1998).

[Char92] B.W. Char et al., *First Leaves: A Tutorial Introduction to Maple V*, Springer-Verlag, Berlin Heidelberg New York (1992).

[CoKM97] S. Collart, M. Kalkbrener, D. Mall, "Converting Bases with the Gröbner Walk", *J. Symbolic Computation* 24/3-4, 465–469 (1997).

[CoLO98] D. Cox, J. Little, D. O'Shea, *Using Algebraic Geometry*, Springer-Verlag, New York (1998).

[Dave81] J.H. Davenport, *On the Integration of Algebraic Functions*, Lecture Notes in Computer Science 102, Springer-Verlag, Heidelberg (1981).

[FGLM93] J.C. Faugère, P. Gianni, D. Lazard, T. Mora, "Efficient Computation of Zero-Dimensional Gröbner Bases by Change of Ordering", *J. Symbolic Computation* 16/4, 377–399 (1993).

[GeCL92] K.O. Geddes, S.R. Czapor, G. Labahn, *Algorithms for Computer Algebra*, Kluwer Acad. Publ., Boston (1992).

[HLSW99] E. Hillgarter, G. Landsmann, J. Schicho, F. Winkler, "Generalized Offsets as Envelopes of a One-parameter Set of Spheres", Techn. Rep. RISC 99-27, RISC-Linz, J. Kepler Univ. Linz (1999).

[Kalk99] M. Kalkbrener, "On the Complexity of Gröbner Bases Conversion", *J. Symbolic Computation* 28/1&2, 265–273 (1999).

[LeLL82] A.K. Lenstra, H.W. Lenstra Jr., L. Lovász, "Factoring Polynomials with Rational Coefficients", *Math. Ann.* 261, 515–534 (1982).

[MacC91] M. MacCallum, F. Wright, *Algebraic Computing with Reduce*, Clarendon Press, Oxford (1991).

[MaMe82] E.W. Mayr, A.R. Meyer, "The Complexity of the Word Problem for Commutative Semigroups and Polynomial Ideals", *Adv. in Math.* 46, 305–329 (1982).

[Mign92] M. Mignotte, *Mathematics for Computer Algebra*, Springer-Verlag, New York (1992).

[MoRo88] T. Mora, L. Robbiano, "The Gröbner Fan on an Ideal", *J. Symbolic Computation* 6/2&3, 183–208 (1988).

[PePo97] M. Peternell, H. Pottmann, "Computing Rational Parametrizations of Canal Surfaces", *J. Symbolic Computation* 23/2&3, 255–266 (1997).

[Risc69] R. Risch, "The Problem of Integration in Finite Terms", *Transactions of the American Mathematical Society* 139, 167–189 (1969).

[Ritt48] J.F. Ritt, *Integration in Finite Terms*, Columbia Univ. Press, New York (1948).

[Robb85] L. Robbiano, "Term Orderings on the Polynomial Ring", *Proceedings EUROCAL'85*, B.F. Caviness (ed.), Springer-Verlag LNCS 204, 513–517, Berlin (1985).

[Rose68] M. Rosenlicht, "Liouville's Theorem on Functions with Elementary Integrals", *Pacific Journal of Mathematics* 24, 153–161 (1968).

[Schi98a] J. Schicho, "Rational Parametrization of Surfaces", *J. Symbolic Computation* 26/1, 1–29 (1998).

[Schi98b] J. Schicho, "Rational Parametrization of Real Algebraic Surfaces", *Proceedings of ISSAC'98*, O. Gloor (ed.), ACM-Press, New York (1998).

[SeWi97] J.R. Sendra, F. Winkler, "Parametrization of Algebraic Curves over Optimal Field Extensions", *J. Symbolic Computation* 23/2&3, 191–207 (1997).

[SeWi99] J.R. Sendra, F. Winkler, "Algorithms for Rational Real Algebraic Curves", *Fundamenta Informaticae* 39/1-2, 211–228 (1999).

[Sing91] M.F. Singer, "Liouvillian Solutions of Linear Differential Equations with Liouvillian Coefficients", *J. Symbolic Computation* 11/3, 251–273 (1991).

[SiUl93] M.F. Singer, F. Ulmer, "Galois Groups of Second and Third Order Linear Differential Equations", *J. Symbolic Computation* 16/1, 9–36 (1993).

[Trag84] B.M. Trager, *On the Integration of Algebraic Functions*, Ph.D. Thesis, Massachusetts Institute of Technology, Computer Science (1984).

[Tran98] Q.-N. Tran, "Parallel Computation and Gröbner Bases: An Application for Converting Bases with the Gröbner Walk", in [BuWi98], 519–531 (1998).

[TrWi00] Q.-N. Tran, F. Winkler (eds.), *Applications of Gröbner Bases*, special issue of the *J. Symbolic Computation*, to appear (2000).

[vdWa91] B.L. van der Waerden, *Algebra*, Vol.I, Springer-Verlag, New York (1991).

[WaSt98] S.M. Watt, H.J. Stetter (eds.), *Symbolic Numeric Algebra for Polynomials*, special issue of the *J. Symbolic Computation* 26/6 (1998).

[Wink96] F. Winkler, *Polynomial Algorithms in Computer Algebra*, Springer–Verlag, Wien New York (1996).

[Wolf91] S. Wolfram, *Mathematica — A System for Doing Mathematics by Computer*, 2nd ed., Addison-Wesley, Reading, MA (1991).

[Zass69] H. Zassenhaus, "On Hensel Factorization, I", *J. Number Theory* 1, 291–311 (1969).

Franz Winkler
RISC-Linz
Johannes Kepler Universität Linz
A-4040 Linz, Austria
email: `Franz.Winkler@risc.uni-linz.ac.at`