Computing Sums of Radicals in Polynomial Time*

Johannes Blömer
Institut für Informatik, Fachbereich Mathematik
Freie Universität Berlin
Arnimallee 2-6, W-1000 Berlin 33, Germany

Abstract

For sums of radicals $\sum_{i=1}^{k} \gamma_i \sqrt[r_i]{\beta_i}$, where γ_i, β_i are elements of some real algebraic number field, $\sqrt[r_i]{\beta_i} \in \mathbf{R}$ we present a Monte Carlo algorithm that runs in polynomial time to decide whether the sum is contained in some number field $\mathbf{Q}(\alpha)$ and, if so, we compute its coefficient representation in $\mathbf{Q}(\alpha)$. The time is polynomial in the number of bits required to represent α, the γ_i's, β_i's and r_i's even if the error probability is chosen exponentially small in the input-size. As a special case the algorithm decides whether the sum is zero. The main algorithm is based on a subalgorithm which is of interest in its own right. This algorithm uses probabilistic methods to check for an element β of an arbitrary (not necessarily real) algebraic number field $\mathbf{Q}(\alpha)$ and some positive, rational integer r whether there exists an r-th root of β in $\mathbf{Q}(\alpha)$.

1 Introduction

A standard problem in Computer Algebra as well as in other areas of Computer Science is to decide whether some complicated expression, which may be given as a result of symbolic computations, is zero or, more general, contained in some field, for example the field of rational numbers. In this paper using various theorems from algebraic number theory we show how to solve this problem for a large class of expressions in polynomial time with small error probability.

Consider for example a sum S of the form $S = \sum_{i=1}^{k} c_i \sqrt[r_i]{q_i}$ with $c_i, q_i \in \mathbf{Q}, r_i \in \mathbf{N}$ such that $\sqrt[r_i]{q_i} \in \mathbf{R}$. We prove that the question whether this sum is zero or, more general, rational can be decided in time polynomial in the number of bits necessary to represent S. Observe that sums of the form described above play an important role in various geometric problems (e.g. Euclidean shortest paths, Euclidean traveling salesman tours). It is not known how to decide efficiently whether such a sum is positive. Although our result concerning these sums obviously relates to this question it has only little effect on the complexity of determining the sign of a sum of radicals. In fact, it only shows that if the latter problem is in NP then it is already in NP∩co-NP.

In the main part of the paper we describe an algorithm that applies to sums of a much more general form than the one mentioned above. Let α be real and algebraic. Therefore α is a root of some integer polynomial $m(X) \in \mathbf{Z}[X]$. Consider the smallest field containing α and the rational numbers. This field is denoted by $\mathbf{Q}(\alpha)$. Any field of this form is called an *algebraic number field*. If n is the degree of the smallest-degree integer polynomial p with $p(\alpha) = 0$, then n is called the degree of α and of the field extension $\mathbf{Q}(\alpha) : \mathbf{Q}$. Furthermore any number $\beta \in \mathbf{Q}(\alpha)$ may then uniquely be written as $\beta = \sum_{i=0}^{n-1} q_i \alpha^i$, $q_i \in \mathbf{Q}$. We show that given a sum S of the form $S = \sum_{i=1}^{k} \gamma_i \sqrt[r_i]{\beta_i}$, where $\gamma_i, \beta_i \in \mathbf{Q}(\alpha), r_i \in \mathbf{N}$, $\sqrt[r_i]{\beta_i} \in \mathbf{R}$, one can decide in polynomial time and with small error probability (for example exponentially small in the input-size) whether this sum is zero or, more general, is itself contained in $\mathbf{Q}(\alpha)$. Here as throughout the rest of the paper for even r_i $\sqrt[r_i]{\beta_i}$ is supposed to be the positive r_i-th real root of β_i. Furthermore we assume that the input-size is specified by the number of bits necessary to represent α, i.e., to represent the minimal polynomial m of α and the number of bits necessary to represent an isolating interval of α. The elements γ_i, β_i of $\mathbf{Q}(\alpha)$ are then identified by n-tuples of rational numbers, where n is the degree of α. These n-tuples represent γ_i and β_i as linear combinations of the elements of the basis $B = \{1, \alpha, \ldots, \alpha^{n-1}\}$ (called the standard basis) of the extension $\mathbf{Q}(\alpha)$ over $\mathbf{Q}$. Also $\log r_i$ bits are needed for r_i.

The correctness of our algorithm for sums of radicals over algebraic number fields is based on a corollary to a theorem due to C. L. Siegel [11]:
Let F be any real field, i.e., $F \subseteq \mathbf{R}$. If for positive integers r_i and elements $\beta_i \in F$, $i = 1, \ldots, k$, with $\sqrt[r_i]{\beta_i} \in \mathbf{R}$,

$$\frac{\sqrt[r_i]{\beta_i}}{\sqrt[r_j]{\beta_j}} \notin F \text{ for all } i \neq j, \tag{1}$$

*This research was supported by the Deutsche Forschungsgemeinschaft under Grant Al 253/1-3, Schwerpunktprogramm "Datenstrukturen und effiziente Algorithmen".

CH3062-7/91/0000/0670$01.00 © 1991 IEEE

then the elements of the set $\{\sqrt[r_1]{\beta_1}, \ldots, \sqrt[r_k]{\beta_k}\}$ are linearly independent over F.

This theorem reduces the question whether a sum of radicals is zero to several tests whether for a pair of radicals $\sqrt[r_i]{\beta_i}$, $\sqrt[r_j]{\beta_j}$ their ratio $\sqrt[r_i]{\beta_i}/\sqrt[r_j]{\beta_j}$ is in F, where $\beta_i, \beta_j \in F = \mathbf{Q}(\alpha)$. We describe an algorithm solving this problem that runs in polynomial time. If the algorithm returns that a ratio $\sqrt[r_i]{\beta_i}/\sqrt[r_j]{\beta_j}$ is contained in F then its coefficient representation with respect to the standard basis B is computed as it is required by the main algorithm.

It is shown that wlog. one can assume $r_i = r_j$. Furthermore if $\mathbf{Q}(\alpha)$ and $\sqrt[r]{\beta}$ are real then the question whether $\sqrt[r]{\beta} \in \mathbf{Q}(\alpha)$ is equivalent to the question whether $X^r - \beta$ is solvable in $\mathbf{Q}(\alpha)$. Therefore we only have to design an algorithm that decides whether $\beta \in \mathbf{Q}(\alpha)$ is an r-th power of some element $\gamma \in \mathbf{Q}(\alpha)$, and, if so, the algorithm has to compute the representation of γ. The algorithm for this problem has two phases. In the first phase we compute the coefficients c_i of an element $\gamma = \sum_{i=0}^{n-1} c_i \alpha^i$ of $\mathbf{Q}(\alpha)$ such that if there exists any r-th root of β in $\mathbf{Q}(\alpha)$ then γ must be one of them. At this point we use the fact that $\mathbf{Q}(\alpha)$ is isomorphic to the fields generated by the other roots of the minimal polynomial of α to reduce the problem to an interpolation problem. In the second phase of the algorithm we check whether the computed number γ actually satisfies $\gamma^r = \beta$. It is here that probabilistic methods are used.

One way to check whether $\gamma^r = \beta$ is to compute both numbers with accuracy good enough to distinguish them if they do not agree. But the accuracy required for this approach is polynomial in r which will not lead to an algorithm with time complexity polynomial in the *bit-size* of r. On the other hand, if we simply compute the coefficients of γ^r by successive squaring and compare them with the coefficients of β there is no guarantee that the coefficients will not get too large during our computations (in the intermediate steps they may be exponentially large even if the coefficients of the final result are of moderate size). We circumvent this problem by transforming $\gamma^r = \beta$ into an equation between algebraic integers and computing the coefficients in this equation modulo some randomly chosen rational integers $z_1, \ldots, z_L$ whose bit-size is polynomial in the input-size of $\sqrt[r]{\beta}$. It is shown that if the coefficients are zero modulo all of the randomly chosen integers $z_1, \ldots, z_L$ then $\gamma^r = \beta$ with high probability. In the rational case as well as in some other cases (quadratic euclidean number fields) the algorithm can be made deterministic.

We note that our methods can be used to decide in polynomial time and with small error probability whether there is some integer relationship between radicals of the form described above, i.e., we can decide in polynomial time whether rational integers c_i exist such that $\sum_{i=1}^{k} c_i \sqrt[r_i]{\beta_i} = 0$ (more general, whether there are algebraic integers contained in some number field with this property). This problem has been examined for arbitrary real numbers by Hastad et al. [6]. But while we are interested in the bit-complexity Hastad et al. considered the arithmetic complexity of the problem. Moreover the main algorithm of this paper can be used to detect degeneracies in arrangements of geometric objects.

The paper is organized as follows:

In Section 2 we present Siegel's Theorem and deduce the corollary on which the correctness of our main algorithm will be based. In Section 3 we give a short description of the deterministic algorithm for the rational case. Section 4 contains the results from algebraic number theory which we will use in Section 5 where we describe the algorithm that decides whether a ratio of two radicals is contained in a given algebraic number field. In Section 6 the main algorithm for the general case is described.

In this extended abstract we cannot give a detailed analysis of the bit complexity of the algorithms instead we will only mention all of the basic facts needed for the analysis and prove some of them. The detailed analysis will appear in a subsequent paper.

2 A theorem by Siegel

Throughout this section we assume that F is an arbitrary real field, $r_i \in \mathbf{N}$, $\beta_i \in F$, $i = 1, \ldots, k$, and $\sqrt[r_i]{\beta_i}$, $i = 1, \ldots, k$, is real. First we introduce some notations. Let us denote the field $F\left(\sqrt[r_1]{\beta_1}, \ldots, \sqrt[r_k]{\beta_k}\right)$ by $F^{(k)}$. The multiplicative group which is generated by the non-zero elements of F and the radicals $\sqrt[r_i]{\beta_i}, i = 1, \ldots, k$, is denoted by $\Gamma^{(k)}$. Using these notations Siegel's theorem can be stated as follows

Theorem 1 *The degree of the field extension $F^{(k)} : F^{(k-1)}$ is equal to the group index f of $\Gamma^{(k-1)}$ in $\Gamma^{(k)}$. Moreover the numbers $\sqrt[r_k]{\beta_k^e}$, $0 \le e \le f-1$, form a basis of this field extension.*

The theorem states that the minimal polynomial p of $\sqrt[r_k]{\beta_k}$ over the field $F^{(k-1)}$ is not only of the form $p(X) = X^f - \gamma$ for some $\gamma \in F^{(k-1)}$ (a proof of this fact can be found for example in [9]) but γ is already contained in the group $\Gamma^{(k-1)}$. Therefore the degree of the field extension $F^{(k)} : F^{(k-1)}$ equals the smallest integer d such that $(\sqrt[r_k]{\beta_k})^d \in \Gamma^{(k-1)}$.
Using induction on k this theorem leads to the following theorem that describes the field extension $F^{(k)} : F$

Theorem 2 *Let f_i be the index of $\Gamma^{(i-1)}$ in $\Gamma^{(i)}$, $i = 1, \ldots, k$, where $\Gamma^{(0)} = F \backslash \{0\}$. Then the degree $[F^{(k)} : F]$ of the field extension $F^{(k)} : F$ is given by $[F^{(k)} : F] =$*

$\Pi_{i=1}^{k} f_i$. *Furthermore the elements*

$$\Pi_{i=1}^{k} \sqrt[r_i]{\beta_i^{e_i}},\ 0 \le e_i \le f_i - 1,\ i = 1, \ldots, k,$$

form a basis of this field extension.

As Siegel already noted this theorem enables us to describe all radicals contained in $F^{(k)}$.

Corollary 3 *Let $\Gamma^{(i)}$, f_i be as in the previous theorem. If $\sqrt[r]{\beta} \in F^{(k)}$, where $r \in \mathbf{N}$, $\beta \in F$, then $\sqrt[r]{\beta}$ is of the form $\sqrt[r]{\beta} = \gamma \Pi_{i=1}^{k} \sqrt[r_i]{\beta_i^{e_i}}$ for some $\gamma \in F$ and $0 \le e_i \le f_i - 1$, $i = 1, \ldots, k$.*

Proof: By assumption the field extension $F^{(k)}(\sqrt[r]{\beta}) : F^{(k)}$ is of degree 1. Therefore the index of $\Gamma^{(k)}$ in the multiplicative group generated by $\Gamma^{(k)}$ and $\sqrt[r]{\beta}$ is also 1, which implies $\sqrt[r]{\beta} \in \Gamma^{(k)}$. □

Now we are able to prove the theorem on which the correctness of the main algorithm will be based.

Theorem 4 *Let $S = \sum_{i=1}^{k} \gamma_i \sqrt[r_i]{\beta_i}$, where $r_i \in \mathbf{N}$, $\gamma_i, \beta_i \in F$ such that $\sqrt[r_i]{\beta_i}$ is real and assume that there exists no pair of indices (i, j), $i \neq j$, with $\sqrt[r_i]{\beta_i} / \sqrt[r_j]{\beta_j} \in F$. Then $S \in F$ if and only if there exists at most one $\gamma_i \neq 0$ and, in this case, $\sqrt[r_i]{\beta_i} \in F$. That is, arbitrary real radicals $\sqrt[r_i]{\beta_i}$ are linearly dependent over F if and only if there are already two radicals $\sqrt[r_i]{\beta_i}$, $\sqrt[r_j]{\beta_j}$ that are linearly dependent over F.*

Proof: We claim that $\{ \sqrt[r_1]{\beta_1}, \ldots, \sqrt[r_k]{\beta_k} \}$ can be extended to a basis of the field extension $F^{(k)} : F$. To prove the claim observe that although not all elements in $\{ \sqrt[r_1]{\beta_1}, \ldots, \sqrt[r_k]{\beta_k} \}$ need to be an element of the basis given by Theorem 2 (some of the f_i may be 1) it follows from Corollary 3 that any $\sqrt[r_i]{\beta_i}$ is a non-zero multiple of a basis element. Since by assumption $\sqrt[r_i]{\beta_i} / \sqrt[r_j]{\beta_j} \notin F$ for all pairs (i, j) any radical $\sqrt[r_i]{\beta_i}$ must be a multiple of a different basis element. This proves the claim and hence the theorem. □

3 The rational case

In this section we briefly describe how to check condition (1) for the rational numbers, i.e., given $q_1, q_2 \in \mathbf{Q}$ and $r_1, r_2 \in \mathbf{N}$ such that $\sqrt[r_i]{q_i} \in \mathbf{R}$, $i = 1, 2$, decide whether $(\sqrt[r_1]{q_1} / \sqrt[r_2]{q_2})$ is rational. The first lemma shows that we may restrict ourselves to the case $r_1 = r_2$.

Lemma 5 *Let q_1, q_2, r_1, r_2 be as above and define r to be the greatest common divisor (r_1, r_2) of r_1, r_2. Let $r_1' := r_1/r$, $r_2' = r_2/r$. Then $(\sqrt[r_1]{q_1} / \sqrt[r_2]{q_2}) \in \mathbf{Q}$ if and only if*

1. $\sqrt[r_1']{q_1} = q_1' \in \mathbf{Q}$ *and* $\sqrt[r_2']{q_2} = q_2' \in \mathbf{Q}$
2. $\sqrt[r]{\frac{q_1'}{q_2'}} \in \mathbf{Q}$.

Proof: $(\sqrt[r_1]{q_1} / \sqrt[r_2]{q_2}) = (\sqrt[r_1']{q_1} / \sqrt[r_2']{q_2})^{\frac{1}{r}}$. Therefore $(\sqrt[r_1]{q_1} / \sqrt[r_2]{q_2}) \in \mathbf{Q}$ implies $(\sqrt[r_1']{q_1} / \sqrt[r_2']{q_2}) \in \mathbf{Q}$, but since $(r_1', r_2') = 1$ this is possible if and only if $\sqrt[r_1']{q_1} = q_1' \in \mathbf{Q}$ and $\sqrt[r_2']{q_2} = q_2' \in \mathbf{Q}$ (consider the prime factorization of the numerators and denominators of q_1, q_2). This proves that the conditions of the lemma are necessary and sufficient. □

The next lemma shows how to check $\sqrt[r]{q} \in \mathbf{Q}$ for $r \in \mathbf{N}, q \in \mathbf{Q}, \sqrt[r]{q} \in \mathbf{R}$.

Lemma 6 *Let r, q be as above. Write q as $\frac{s}{t}$, $s \in \mathbf{Z}, t \in \mathbf{N}, (s, t) = 1$. Then $\sqrt[r]{q} \in \mathbf{Q}$ if and only if $\sqrt[r]{s} \in \mathbf{Q}$, $\sqrt[r]{t} \in \mathbf{Q}$.*

Proof: Consider the prime factorization of s and t. □

So far we reduced the original problem to the test whether $\sqrt[r]{z} \in \mathbf{Q}$ for $r \in \mathbf{N}, z \in \mathbf{Z}$ and $\sqrt[r]{z} \in \mathbf{R}$. The following well-known and simple lemma is useful in the design of an algorithm for this test.

Lemma 7 *Let r, z be as above. $\sqrt[r]{z} \in \mathbf{Q}$ if and only if $\sqrt[r]{z} \in \mathbf{Z}$.*

We now describe the algorithm that checks whether $\sqrt[r]{z} \in \mathbf{Z}$. First expressing $\sqrt[r]{z}$ as $\exp \frac{1}{r} \ln z$ and using the algorithms of Brent [4] we compute an approximation to $\sqrt[r]{z}$ with absolute error less than $\frac{1}{2}$. The computed interval therefore contains exactly one integer z', which is the only candidate (up to sign) for the r-th root of z in $\mathbf{Z}$. By successive squaring we compute z'^r and test whether it is equal to z. Using the results of R. Brent [4] a careful but straightforward analysis shows

Lemma 8 *Let r, z be m-bit integers. It can be checked in time $O(M(m) \log m)$ whether $\sqrt[r]{z} \in \mathbf{Q}$, and if so, the time needed to compute this number is also bounded by $O(M(m) \log m)$. Here $M(m)$ denotes the time needed to multiply two m-bit integers.*

It follows from the previous lemmata that the test $(\sqrt[r_1]{q_1} / \sqrt[r_2]{q_2}) \in \mathbf{Q}$ can be reduced to a constant number of applications of the algorithm for $\sqrt[r]{z} \in \mathbf{Z}$, $r \in \mathbf{N}$, $z \in \mathbf{Z}$, plus a constant number of gcd computations in $\mathbf{Z}$. Therefore we proved

Theorem 9 *Let r_1, r_2, q_1, q_2 be m-bit numbers. One can check in time $O(M(m) \log m)$ whether the ratio $(\sqrt[r_1]{q_1} / \sqrt[r_2]{q_2})$ is rational. If $(\sqrt[r_1]{q_1} / \sqrt[r_2]{q_2}) \in \mathbf{Q}$, this number can be computed in time $O(M(m) \log m)$.*

It is clear that our arguments in this section rely heavily on the fact that $\mathbf{Z}$ is a unique factorization domain (UFD). It is well-known that for an arbitrary algebraic number field the ring of integers of this field is not a UFD. This is the main reason why we need different techniques than the ones used so far for the general case. In the next section we will state the facts from algebraic number theory that we need to handle the algebraic number field case.

4 Some basics from number theory

Throughout this section we consider an arbitrary algebraic number field $F = \mathbf{Q}(\alpha)$. That is, $\alpha \in \mathbf{C}$ is a root of a polynomial $m(X) = \sum_{i=0}^{n} m_i X^i, m_i \in \mathbf{Z}$, and $\mathbf{Q}(\alpha)$ is defined to be the smallest field containing α and $\mathbf{Q}$. Any element in this field may be uniquely written as $\sum_{i=0}^{n-1} q_i \alpha^i, q_i \in \mathbf{Q}$, if n is the smallest possible degree of an integer polynomial with root α. This polynomial is called the *minimal polynomial* of α. The *ring of algebraic integers* (or simply *integers*) in $F = \mathbf{Q}(\alpha)$, denoted by R_F, is the set of all $\beta \in \mathbf{Q}(\alpha)$ such that there exists an integer polynomial $p(X) = \sum_{i=0}^{n} p_i X^i$ with root β and $p_n = 1$. It is known that this set with the usual addition and multiplication in $\mathbf{C}$ forms a ring (cf. [10]). We may assume that α itself is an algebraic integer. If α is not and $m(X) = \sum_{i=0}^{n} m_i X^i$ is the minimal polynomial of α then $m_n \alpha$ is an integer and generates the same field. Under these conditions R_F contains the $\mathbf{Z}-$module $\mathbf{Z}[\alpha] = \{\sum_{i=0}^{n-1} c_i \alpha^i | c_i \in \mathbf{Z}\}$. Since we cannot afford to compute a basis for R_F itself we will need a superset of R_F as well. First we give a definition.

Definition 10 *Let α be an algebraic integer and $F = \mathbf{Q}(\alpha)$. Furthermore denote by $\alpha = \alpha_0, \alpha_1, \ldots, \alpha_{n-1}$ the roots of the minimal polynomial of α. Then the number*

$$d = d(\alpha) = (\prod_{0=i<j=n-1} (\alpha_i - \alpha_j))^2$$

is called the discriminant *of α.*

$d = d(\alpha)$ is always a rational integer and is bounded in absolute value by $2^{2n^2 s}$ if $|m_i| \leq 2^s, i = 0, \ldots, n$. This follows from the usual bounds on the absolute value of the roots of an integer polynomial (cf. [3]).

A proof for the following lemma can be found in [10].

Lemma 11 *Let $F = \mathbf{Q}(\alpha)$, α an algebraic integer. Then the ring of integers R_F of $\mathbf{Q}(\alpha)$ is contained in the $\mathbf{Z}-$module $G = (\mathbf{Z}/d) \oplus (\mathbf{Z}/d)\ \alpha \oplus \ldots \oplus (\mathbf{Z}/d)\ \alpha^{n-1}$, i.e., any number in R_F can be uniquely written as $\frac{1}{d} \sum_{i=0}^{n-1} c_i \alpha^i$, $c_i \in \mathbf{Z}$.*

Note that the inclusion is always strict.

We need one more definition from algebraic number theory. Let $\alpha = \alpha_0, \alpha_1, \ldots, \alpha_{n-1}$ be as above. Then the mappings $\sigma_j, j = 0, \ldots, n-1$, which are defined as follows

$$\begin{aligned} \sigma_j : \mathbf{Q}(\alpha) &\rightarrow \mathbf{Q}(\alpha_j) \\ \sum_{i=0}^{n-1} q_i \alpha^i &\rightarrow \sum_{i=0}^{n-1} q_i \alpha_j^i, \end{aligned}$$

are called the (n distinct) *embeddings of $\mathbf{Q}(\alpha)$ into the complex numbers.* The σ_j's are field isomorphisms as it is easily checked. These embeddings will play an important role in our algorithm, which we are now going to describe.

5 Ratios of radicals

Let $r_1, r_2 \in \mathbf{N}$ and $\beta_1, \beta_2 \in \mathbf{Q}(\alpha)$, $\alpha \in \mathbf{R}$ such that $\sqrt[r_i]{\beta_i} \in \mathbf{R}$. We want to check whether $\sqrt[r_1]{\beta_1} / \sqrt[r_2]{\beta_2}$ is contained in $\mathbf{Q}(\alpha)$. As in the rational case we will show later that if $\alpha \in \mathbf{R}$ we can restrict ourselves to the test whether a radical $\sqrt[r]{\beta}, r \in \mathbf{N}, \beta \in \mathbf{Q}(\alpha)$, is contained in $\mathbf{Q}(\alpha)$. For $\alpha, \sqrt[r]{\beta} \in \mathbf{R}$ this is in turn equivalent to the question whether there exists any r-th root of β in $\mathbf{Q}(\alpha)$. We describe an algorithm that solves the last problem not only for real but for arbitrary algebraic number fields.

As we mentioned in the introduction our algorithm has two phases. In fact, the basic strategy is the same as in the rational case. We first compute the coefficient representation of a number $\gamma = \sum_{i=0}^{n-1} c_i \alpha^i \in \mathbf{Q}(\alpha)$ such that $\gamma^r = \beta$ provided there exists an $r-$th root of β in $\mathbf{Q}(\alpha)$. In the second phase we test whether $\gamma^r = \beta$ holds.

Computing a candidate. We want to compute $c_i \in \mathbf{Q}$ such that if $\sqrt[r]{\beta} \in F = \mathbf{Q}(\alpha)$ then $\gamma^r = (\sum_{i=0}^{n-1} c_i \alpha^i)^r = \beta$. In the algorithm we will need a bound on the size of the denominators of the c_i's. To deduce such a bound we use the following lemma

Lemma 12 *If $\beta \in R_F$ then $\sqrt[r]{\beta} \in F = \mathbf{Q}(\alpha)$ if and only if $\sqrt[r]{\beta} \in R_F$. Here $\sqrt[r]{\beta}$ denotes any r-th root of β.*

Proof: $\sqrt[r]{\beta}$ is an algebraic integer in the complex numbers. Therefore if $\sqrt[r]{\beta}$ is contained in F then it is already contained in the ring of integers of this field. □

We get

Lemma 13 *Let $\beta = \sum_{i=0}^{n-1} q_i \alpha^i$, and let q be the l.c.m. (least common multiple) of the denominators of the q_i's. If there exists a $\gamma \in \mathbf{Q}(\alpha)$ such that $\gamma^r = \beta$, then the denominators of c_i in the representation of γ as $\sum_{i=0}^{n-1} c_i \alpha^i, c_i \in \mathbf{Q}$, are bounded by qd, where d is the discriminant of α.*

Proof: Write β as $\frac{1}{q}\beta'$, $\beta' \in R_F$, $\beta' = \sum_{i=0}^{n-1} b_i\alpha^i, b_i \in \mathbf{Z}, q \in \mathbf{Z}$. Denote by $\sqrt[r]{\beta}$ any r-th root of β. Then $\sqrt[r]{\beta} = \sqrt[r]{\frac{\beta'}{q}} = \frac{1}{q}\sqrt[r]{q^{r-1}\beta'}$. Now $q^{r-1}\beta \in R_F$ and if $\sqrt[r]{\beta} \in F$ then $\sqrt[r]{q^{r-1}\beta'} \in F$. Therefore $\sqrt[r]{q^{r-1}\beta'} \in R_F$ and (due to Lemma 11) it has a representation of the form $\sqrt[r]{q^{r-1}\beta'} = \frac{1}{d}\sum_{i=0}^{n-1} z_i\alpha^i$, $z_i \in \mathbf{Z}$. The lemma follows. □

Suppose that there exists in fact a $\gamma = \sum_{i=0}^{n-1} c_i\,\alpha^i \in F = \mathbb{Q}(\alpha)$ such that $\gamma^r = \beta$. We apply the distinct field embeddings σ_j to this equation and get

$$\beta_j = \sigma_j(\beta) = \left(\sum_{i=0}^{n-1} c_i\alpha_j^i\right)^r, \text{ or } \sqrt[r]{\beta_j} = \sum_{i=0}^{n-1} c_i\alpha_j^i.$$

But here we really have to worry a bit more about the actual meaning we give to $\sqrt[r]{\beta_j}$ since an r-th root is not uniquely defined. On the other hand the complex logarithm (if we fix the arc to the interval $[0, 2\pi)$) and the exponential function (restricted in the imaginary part of the argument to the interval $[0, 2\pi)$) are uniquely defined and bijective. Therefore in the equations above we interpret $\sqrt[r]{\beta_j}$ as $\exp\frac{1}{r}\ln\beta_j$. Using this convention our previous argument can be made precise: $\beta_j = \left(\sum_{i=0}^{n-1} c_i\alpha_j^i\right)^r$ implies $\frac{1}{r}\ln\beta_j = \ln\sum_{i=0}^{n-1} c_i\alpha_j^i$. Hence $\exp\frac{1}{r}\ln\beta_j = \sum_{i=0}^{n-1} c_i\alpha_j^i$, for all $j = 0, \ldots, n-1$. So the coefficients c_i, if they exist, are a solution to the following equation

$$\begin{bmatrix} 1 & \alpha_0 & \ldots & \alpha_0^{n-1} \\ 1 & \alpha_1 & \ldots & \alpha_1^{n-1} \\ \vdots & & & \\ 1 & \alpha_{n-1} & \ldots & \alpha_{n-1}^{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} \exp\frac{1}{r}\ln\beta_0 \\ \exp\frac{1}{r}\ln\beta_1 \\ \vdots \\ \exp\frac{1}{r}\ln\beta_{n-1} \end{bmatrix}$$

We may also view this problem as an interpolation problem, where we are asking for a rational polynomial $p(x) = \sum_{i=0}^{n-1} c_i x^i$, such that $p(\alpha_j) = \exp\frac{1}{r}\ln\beta_j, j = 0, \ldots, n-1$. A solution is given by

$$p(x) = \sum_{j=0}^{n-1} \exp\frac{1}{r}\ln\beta_j \frac{\Pi_{i\neq j}(x-\alpha_i)}{\Pi_{i\neq j}(\alpha_j-\alpha_i)}.$$

So in order to compute the c_i's we approximate the coefficients of p with absolute error $\leq \frac{1}{2}(\frac{1}{qd})^2$, where q and d are as in Lemma 13.

This in turn can be done by approximating the coefficients of $m^{(j)}(x) = \Pi_{i\neq j}(x-\alpha_i) = m(x)/(x-\alpha_j)$, approximating $\Pi_{i\neq j}(\alpha_j-\alpha_i)$ and approximating $\exp\frac{1}{r}\ln\beta_j$. In all three cases we need approximations to the zeros of $m(x)$, which can be computed using Schönhages algorithm [13]. Approximations to the coefficients of $m^{(j)}(x)$ are quite easy since looking at the elementary symmetric functions we see that the coefficients are given by the recursive formulae $m_{n-1}^{(j)} = 1, m_{n-(\ell+1)}^{(j)} = m_{n-\ell} + \alpha_j m_{n-\ell}^{(j)}$ or by the formulae $m_\ell^{(j)} = \sum_{i=0}^{n-\ell-1} m_{n-i}\,\alpha_j^{n-\ell-i-1}$, $\ell = 0, \ldots, n-2$. So good approximations to α_j give rise to good approximations of $m_\ell^{(j)}$. Approximations to $\Pi_{j\neq i}(\alpha_j-\alpha_i)$ are even simpler since $\Pi_{j\neq i}(\alpha_j-\alpha_i) = m'(\alpha_j)$. Approximations to the β_j's are obtained by expressing β_j in polar coordinates and by using the algorithms of R. Brent [4] for efficiently evaluating exp, ln, sin, cos, and arccot with high precision.

A more detailed analysis which we have to omit in this extended abstract shows that in order to compute the coefficients of p with absolute error less than $\frac{1}{2}(\frac{1}{dq})^2$, d the discriminant of α and q the l.c.m of the denominators of the q_i's in the representation of β, we have to work with approximations of absolute error 2^{-D}, where D is polynomial in the bit-size of the representation of β, i.e., the bit-size of $m(x)$, the minimal polynomial of α, and the bit-size of the q_i's, where $\beta = \sum_{i=0}^{n-1} q_i\alpha^i$.

A probabilistic test for equality. So far we computed the coefficients of a number $\gamma \in \mathbb{Q}(\alpha)$ such that if $X^r - \beta = 0$ is solvable in $\mathbb{Q}(\alpha)$ then γ is a solution. We will use probabilistic methods to check whether $\gamma^r = \beta$ holds.

Let us begin by arguing why the usual method to distinguish two algebraic numbers won't lead to an algorithm that is polynomial in $\log r$. First of all in infinitely many cases (to be more precise whenever the field norm $N(\beta)$ is one) we cannot bound the size of r in terms of the input-size of β. That is, we cannot give an upper bound such that if r exceeds this bound then $X^r - \beta = 0$ is not solvable in $\mathbb{Q}(\alpha)$. This problem arises especially if the ring of integers of $\mathbb{Q}(\alpha)$ is not a unique factorization domain.

Because there is no upper bound on r the usual root separation bounds to distinguish two algebraic integers will result in an algorithm that is polynomial only in r rather than in $\log r$. This is true because using the notations of the previous paragraph we have to distinguish the algebraic integers γ'^r and $q^{r-1}d^r\beta'$.

Finally we cannot compute γ^r by successive squaring since the coefficients in the intermediate steps may get exponentially large. This can happen even if the coefficients of the final result are of polynomial size.

We now describe an algorithm that will give the correct answer to the question whether $\gamma^r = \beta$ or equivalently $\gamma'^r - q^{r-1}d^r\beta' = 0$ with probability at least $1 - 2^{-t}$ for $t > 0$. The running-time of the algorithm is polynomial in $t, \log r$ and the number of bits necessary to represent β.

Let $I = [0, 2^T]$, where $T = c(\log N + \log r + t)$, N is an upper bound on the bit-size of the representation of β and $c > 0$ is some constant to be specified later. Randomly choose $6(t+1)T$ rational integers z_j from I. Compute

the coefficients of $\gamma'^r - q^{r-1}d^r\beta'$ modulo all z_j's. If for all $j = 1, \ldots, T$ every coefficient of $\gamma'^r - q^{r-1}d^r\beta'$ is zero output $\gamma^r = \beta$ otherwise output that $\gamma^r \neq \beta$ and that $X^r - \beta$ is not solvable in $\mathbf{Q}(\alpha)$.

We claim that this algorithm is polynomial in $t, \log r$ and N and that it fails to give the correct answer with probability less than 2^{-t}.

We start by analyzing the running-time. The proof of the following lemma is straightforward.

Lemma 14 *Let $\beta_1, \beta_2 \in \mathbf{Z}[\alpha]$ and denote by $(\beta_i)_z$ the number that is obtained by reducing all coefficients modulo z. Then $(\beta_1)_z(\beta_2)_z = (\beta_1\beta_2)_z$ and $(\beta_1)_z + (\beta_2)_z = (\beta_1 + \beta_2)_z$ using arithmetic in $\mathbf{Z}[\alpha]$.*

We therefore may compute the coefficients of $\gamma'^r - q^{r-1}d^r\beta'$ modulo z_j by computing γ'^r, q^{r-1}, d^r and β' modulo z_j. γ'^r, q^{r-1} and d^r in turn are computed modulo z_j by successive squaring and reducing the coefficients modulo z_j after each step. Since arithmetic in $\mathbf{Q}(\alpha)$ can be done in polynomial time (cf. [7]) the first part of the claim follows.

Next we analyze the error probability of the algorithm.

Definition 15 *If $\gamma'^r - q^{r-1}d^r\beta' \neq 0$ then we call a number $z \in \mathbf{Z}$ unlucky if it divides all coefficients of $\gamma'^r - q^{r-1}d^r\beta'$ or equivalently the gcd of these coefficients. Otherwise we call z lucky.*

If a number $z \in \mathbf{Z}$ is unlucky then we have $\gamma^r \neq \beta$ although the coefficients of $\gamma'^r - q^{r-1}d^r\beta'$ are zero modulo z. On the other hand if we find an integer z such that not all coefficients of $\gamma'^r - q^{r-1}d^r\beta'$ are divisible by z then $\gamma^r \neq \beta$. This shows that if the algorithm outputs $\gamma^r \neq \beta$ then the decision will always be correct. If $\gamma^r \neq \beta$ then we have to show that with probability at least $1 - 2^{-t}$ there is a lucky number among the $6(t+1)T$ randomly chosen integers. Since in this case the algorithm fails to give the correct answer if and only if all integers z_j are unlucky.

Now the gcd of the coefficients of $\gamma'^r - q^{r-1}d^r\beta'$ is bounded in size by $2^{c'Nr}$ for some constant $c' > 0$ (cf. [7]). Unfortunately an integer $z \in \mathbf{Z}$ may have $z^{\mathcal{O}(\frac{1}{\log\log z})}$ different divisors (cf. [5]). Therefore we cannot prove that most numbers in I are lucky. On the other hand I contains at least $\frac{1}{6}2^T/T > 2^{(c-1)(\log N + \log r + t)}$ distinct primes (cf. [2]). If $\gamma'^r - q^{r-1}d^r\beta' \neq 0$ then the gcd of the coefficients of $\gamma'^r - q^{r-1}d^r\beta'$ has at most $c'Nr = 2^{\log c' + \log N + \log r}$ distinct prime divisors. Hence a random prime in I is lucky with probability $> 1 - 2^{-(t+1)}$ provided the constant c is chosen appropriately.

Next we show that among the randomly chosen numbers z_j's there is a prime with probability at least $1 - 2^{-(t+1)}$. We prove

Lemma 16 *Let $J = [0, x]$. If $6(t+1)\log x$ numbers are chosen randomly from J then with probability at least $1 - 2^{-(t+1)}$ one of the numbers is prime.*

Proof: A random number in J is composite with probability at most $1 - 1/(6\log x)$ (cf. [2]). Therefore the probability that none of the chosen numbers is prime is bounded by $(1 - 1/(6\log x))^{6(t+1)\log x} \cdot (1 - 1/(6\log x)) \leq e^{-\frac{1}{6\log x}}$ hence $(1 - 1/(6\log x))^{6(t+1)\log x} \leq e^{-(t+1)} < 2^{-(t+1)}$. The lemma follows. □

Our arguments imply

Lemma 17 *If $6c(t+1)(\log N + \log r + t)$ integers are randomly chosen from the interval $I = [0, 2^{c(\log N + \log r + t)}]$ then with probability at least $1 - 2^{-t}$ one of the numbers is lucky.*

This lemma proves the claim on the error probability of our algorithm.
We summarize the results of the two previous paragraphs in

Theorem 18 *Let $F = \mathbf{Q}(\alpha)$ be an arbitrary algebraic number field, $\beta \in F$, $r \in \mathbf{N}$. There exists a probabilistic algorithm of Monte Carlo type with error probability less than 2^{-t} that decides whether there exists a number $\gamma \in F$ such that $\gamma^r = \beta$. The running-time of the algorithm is polynomial in $t, \log r$ and the input-size of β. If the algorithm returns that there exists such a number the coefficients of an element of F with this property are computed.*

A detailed analysis shows

Theorem 19 *If the coefficients of the minimal polynomial of α and the coefficients q_i in $\beta = \sum_{i=0}^{n-1} q_i\alpha^i$ are s-bit numbers then the running-time of the algorithm is*

$$\mathcal{O}(n^2M(n^4s) + t(\log ns + \log r + t)(M(n^3s) +$$

$$n\log rM(n(\log ns + \log r + t)))).$$

Here $M(m)$ denotes the time needed to multiply two m-bit integers.

We remark that there is a different probabilistic algorithm that checks whether $\gamma^r = \beta$. First one computes in expected polynomial time an integer in I that is prime and lucky with high probability and then tests the equality only modulo this number. The dependence in the (expected) running-time of this algorithm on t is worse than in the algorithm we presented here in more detail. On the other hand the dependence on n and s is slightly better.

An algorithm for ratios of radicals. We return to the original problem of deciding whether $\sqrt[r]{\beta_1}/\sqrt[r]{\beta_2} \in \mathbb{Q}(\alpha)$, where $\alpha \in \mathbf{R}$, $r_i \in \mathbf{N}$, $\beta_i \in \mathbb{Q}(\alpha)$ such that $\sqrt[r]{\beta_i} \in \mathbf{R}$.

Theorem 20 *Let $\alpha, r_1, r_2, \beta_1, \beta_2$ be as above. It can be checked with error probability less than 2^{-t} and in time polynomial in $t, \log r_1, \log r_2$ and in the bit-size of the representation of β_1, β_2 whether $\sqrt[r]{\beta_1}/\sqrt[r]{\beta_2} \in \mathbb{Q}(\alpha)$. Furthermore if the algorithm decides that this ratio of radicals is in $\mathbb{Q}(\alpha)$ its coefficient representation is computed.*

Proof: Let r be the gcd of r_1 and r_2, $r_1' = r_1/r, r_2' = r_2/r$. First we show that $\sqrt[r]{\beta_1}/\sqrt[r]{\beta_2} \in \mathbb{Q}(\alpha)$ is equivalent to

(i) $\sqrt[r_1']{\beta_1} = \beta_1' \in \mathbb{Q}(\alpha)$, $\sqrt[r_2']{\beta_2} = \beta_2' \in \mathbb{Q}(\alpha)$

(ii) $(\beta_1'/\beta_2') \in \mathbb{Q}(\alpha)$.

Suppose $\sqrt[r]{\beta_1}/\sqrt[r]{\beta_2} = (\sqrt[r_1']{\beta_1}/\sqrt[r_2']{\beta_2})^{\frac{1}{r}} \in \mathbb{Q}(\alpha)$. This implies $\sqrt[r_1']{\beta_1}/\sqrt[r_2']{\beta_2} \in \mathbb{Q}(\alpha)$. So assume $\sqrt[r_1']{\beta_1} = \beta\sqrt[r_2']{\beta_2}$, with $\beta \neq 0 \in \mathbb{Q}(\alpha)$.
We claim that the algebraic degree over $\mathbb{Q}(\alpha)$ of $\sqrt[r_1']{\beta_1}$ and $\beta\sqrt[r_2']{\beta_2}$ is a divisor of r_1', r_2', respectively. Since the degree of $\beta\sqrt[r_2']{\beta_2}$ is the same as the degree of $\sqrt[r_2']{\beta_2}$ we will only show that the degree of $\sqrt[r_1']{\beta_1}$ is a divisor of r_1'.
By Theorem 2 (applied to $\mathbb{Q}(\alpha)$ and $k = 1$) the degree of $\sqrt[r_1']{\beta_1}$ is the smallest integer d such that $(\sqrt[r_1']{\beta_1})^d \in \mathbb{Q}(\alpha)$. Suppose r_1' is not divisible by d. Then $r_1' = qd + e$, for some $q, e \in \mathbf{Z}$, $0 < e < d$. This implies $(\sqrt[r_1']{\beta_1})^e = \beta_1/(\sqrt[r_1']{\beta_1})^{qd} \in \mathbb{Q}(\alpha)$, contradicting the minimality of d. Hence $\sqrt[r_1']{\beta_1} = \beta\sqrt[r_2']{\beta_2}$ is possible if and only if $\sqrt[r_1']{\beta_1} = \beta_1' \in \mathbb{Q}(\alpha)$ and $\sqrt[r_2']{\beta_2} = \beta_2' \in \mathbb{Q}(\alpha)$. But then the equivalence we want to establish is immediate.

So in order to check whether $\sqrt[r]{\beta_1}/\sqrt[r]{\beta_2} \in \mathbb{Q}(\alpha)$ we compute r, r_1', r_2' and apply the algorithms of the previous paragraphs to $\sqrt[r_1']{\beta_1}$ and $\sqrt[r_2']{\beta_2}$. If (i) is true we apply the algorithms once more to $(\beta_1'/\beta_2')^{\frac{1}{r}} = \sqrt{\beta}$ to decide whether (ii) is also true.

Now by our results on computing a candidate the bit-size of β_1', β_2' is polynomially related to the bit-size of β_1, β_2, respectively. Hence the size of $\beta = \beta_1'/\beta_2'$ is polynomially related to the bit-size of β_1 and β_2. Furthermore β can be computed in polynomial time (for both facts see [7]). Since we apply the algorithms of the previous paragraphs at most three times the error probability of the algorithm for ratios of radicals is at most three times larger than the error probability of the algorithms for radicals $\sqrt{\beta}$. The theorem follows. □

6 The main algorithm

In this section F is a real algebraic number field $F = \mathbb{Q}(\alpha), \beta_i \in F, r_i \in \mathbf{N}$, $i = 1, \ldots, k$, and the radicals $\sqrt[r]{\beta_i}$ are real. Furthermore we assume that for even r_i $\sqrt[r]{\beta_i}$ is the positive real root.

We describe briefly the main algorithm that transforms any sum of radicals in a sum satisfying $\sqrt[r]{\beta_i}/\sqrt[r]{\beta_j} \notin F$. Let $S = \sum_{i=1}^{k} \gamma_i \sqrt[r]{\beta_i}$, $R = \{\sqrt[r]{\beta_1}, \ldots, \sqrt[r]{\beta_k}\}$. First we partition R using the algorithm of the previous section into subsets $R_1, \ldots, R_h$ such that two radicals are in the same subset if and only if their ratio is contained in F. Wlog. we assume $\sqrt[r]{\beta_\ell} \in R_\ell$ for $\ell = 1, \ldots, h$. According to this partition we split S into sums S_ℓ, where in S_ℓ we sum only over those terms in S such that the corresponding radical is in R_ℓ. Then S_ℓ is of the form $\theta_\ell \sqrt[r]{\beta_\ell}$, $\theta_\ell \in F$. As is easily seen the representation of θ_ℓ as $\sum_{i=j}^{n-1} c_i\alpha^i$ can be computed efficiently using again the algorithm of Section 5 (or in the rational case the algorithm of Section 3).
Now according to Theorem 4 $S \in F$ if and only if there exists only one $\theta_\ell \neq 0$ and $\sqrt[r]{\beta_\ell} \in F$. Therefore in the last step of the algorithm we have to check this property. Clearly the error probability of the above algorithm is bounded by $\binom{k}{2}$ times the error probability of the algorithm that checks ratios of radicals. We showed

Theorem 21 *Let $S = \sum_{i=1}^{k} \gamma_i \sqrt[r]{\beta_i}$, where α is a real algebraic number of degree n, $r_i \in \mathbf{N}$, $\beta_i, \gamma_i \in \mathbb{Q}(\alpha)$ and $\sqrt[r]{\beta_i}$ is real. Assume that S is represented by the minimal polynomial of α, the coefficients of γ_i, β_i, $i = 1, \ldots, k$, in their representation as linear combinations of $\{1, \alpha, \ldots, \alpha^{n-1}\}$ and the positive integers r_i. Then a Monte Carlo algorithms with error probability less than 2^{-t} exists that runs in time polynomial in t and the input-size of S and that decides whether the sum S is contained in $\mathbb{Q}(\alpha)$. If the algorithm decides $S \in \mathbb{Q}(\alpha)$ the representation of S as a linear combination of $\{1, \alpha, \ldots, \alpha^{n-1}\}$ is also computed.*

A more detailed analysis shows

Theorem 22 *If the coefficients of the minimal polynomial of α and the coefficients in the representation of $\beta_i, i = 1, \ldots, k$, are s_1-bit number and $r_i \leq 2^{s_2}, i = 1, \ldots, k$, then the running-time of the algorithm is*

$$\mathcal{O}(k^2(n^2M(n^9s_1)+ (t + \log k)(\log ns_1 + s_2 + t + \log k) (M(n^8s_1) + s_2nM(n(\log ns_1 + s_2 + t + \log k)))) + nkM(n^8ks_1)).$$

Here $M(m)$ denotes the time needed to multiply two m-bit integers.

7 Conclusions

The algorithm described in Section 6 is correct only if all radicals are real. If some of the radicals $\sqrt[r]{\beta_i}$ are not real

Siegel's theorem cannot be applied and the algorithm will not give the correct answer. However, we may interpret $\sqrt[r_i]{\beta_i}$ in case $\beta_i < 0$, $r_i \equiv 0 \pmod 2$ either as $i\sqrt[r_i]{-\beta}$ if $r \equiv 2 \pmod 4$ or as $\zeta_8\sqrt[r_i]{-\beta}$ if $r \equiv 0 \pmod 4$, here ζ_8 denotes a primitive 8th root of unity and then apply the algorithm to the real and imaginary part of the sum. Besides, we don't work with ground field $\mathbb{Q}$ but with ground field $\mathbb{Q}\left(\sqrt{2}\right)$ since ζ_8 may be chosen as $\frac{1}{\sqrt{2}}(1+i)$. The time bounds given in Theorem 22 don't change.

A generalization of the main algorithm can be used to solve the following problem: Given a sum S of radicals and a conjecture that S is contained in an extension $\mathbb{Q}(\alpha, \theta)$ of $\mathbb{Q}(\alpha)$. Decide whether the conjecture is true. This problem can also be solved in polynomial time, polynomial in the number of bits required to represent S and to specify θ.

Unfortunately the methods presented in this paper cannot be used to determine efficiently the sign of a sum of square roots.

Acknowledgement I would like to thank Helmut Alt, Emo Welzl and Chee Yap for several helpful discussions and for reading previous versions of this work.

References

[1] H. Alt, "Comparing the Combinatorial Complexities of Arithmetic Functions", *Journal of the ACM*, Vol 35, pp. 447-460, 1988.

[2] T. M. Apostol, *Introduction to Analytic Number Theory*, Springer-Verlag, 1976.

[3] M. Ben-Or, E. Feig, D. Kozen, P. Tiwari, "A Fast Parallel Algorithm for Determining All Roots of a Polynomial with Real Roots", *SIAM Journal on Computing*, Vol. 17, pp. 1081-1092, 1988.

[4] R. P. Brent, "Fast Multiple-Precision Evaluation of Elementary Functions", *Journal of the ACM*, Vol. 23, pp. 242-251, 1976.

[5] K. Chandrasekharan, *Arithmetical Functions*, Springer-Verlag, 1970.

[6] J. Hastad, B. Just, J. C. Lagarias, C. P. Schnorr "Polynomial Time Algorithms for Finding Integer Relations among Real Numbers", *SIAM Journal on Computing* Vol. 18, pp. 859-881, 1989.

[7] R. Loos, "Computing in Algebraic Extensions", *Computing*, Suppl. 4, pp. 173-187, 1982.

[8] R. Loos, "Generalized Polynomial Remainder Sequences", *Computing*, Suppl. 4, pp. 115-137, 1982.

[9] L. J. Mordell, "On the Linear Independence of Algebraic Numbers", *Pacific Journal of Mathematics*, Vol. 3, pp. 625-630, 1953.

[10] D. A. Marcus, *Number Fields*, Springer-Verlag, 1977.

[11] C. L. Siegel, "Algebraische Abhängigkeit von Wurzeln", *Acta Arithmetica*, Vol. 21, pp. 59-64, 1971.

[12] A. Schönhage, "Schnelle Berechnung von Kettenbruchentwicklungen", *Acta Informatica*, Vol. 1, pp. 139-144, 1971.

[13] A. Schönhage, "The Fundamental Theorem of Algebra in Terms of Computational Complexity", *Preliminary Report*, Universität Tübingen, 1982.

[14] A. Schönhage, V. Strassen, "Schnelle Multiplikation großer Zahlen", *Computing*, Vol. 7, pp. 281-292, 1971.

[15] R. Solovay, V. Strassen, "A Fast Monte Carlo Test for Primality", *SIAM Journal on Computing*, Vol. 6, pp. 84-85, 1977.