

On the complexity of some problems in algorithmic algebraic number theory

Dissertation
zur Erlangung des Grades
des Doktors der Naturwissenschaften
der Technischen Fakultät
der Universität des Saarlandes
von

Christoph Thiel

Saarbrücken
1995

I would like to thank my thesis advisor, Prof. Dr. Johannes Buchmann, for his thoughtful guidance and support during the course of my research. Numerous discussions with him were very helpful. I wish to thank Prof. Dr. Michael Pohst for evaluating this thesis. I am grateful to Stefan Neis who read a preliminary version of this thesis, and to Hellai Abdullah and Volker Müller who provided valuable comments and ideas.

This research was supported by a graduate fellowship of the Deutsche Forschungsgemeinschaft which was awarded within the Graduiertenkolleg Informatik at the Universität des Saarlandes. I would like to thank the Deutsche Forschungsgemeinschaft and all members of the Graduiertenkolleg.

Also, I would like to thank my friends Hellai Abdullah, Ingrid Biehl, Bernhard Kipper, Bernd Meyer, Bärbel Müller, Volker Müller, Petra Naumann-Kipper, Ralf Roth, and Diethelm Schlegel, who always had time for discussions and motivating conversations.

Finally, I especially thank my parents and my brothers who have supported me during my entire studies.

Zusammenfassung

Zu den wichtigen Problemen der algorithmischen algebraischen Zahlentheorie gehört die Bestimmung der Einheitengruppe bzw. eines Systems von Fundamenteinheiten einer Ordnung eines algebraischen Zahlkörpers. Dabei ist man einerseits an der Entwicklung von effizienten Berechnungsverfahren interessiert, die in der Praxis ein möglichst günstiges Laufzeitverhalten aufweisen. Andererseits möchte man auch ihr asymptotisches Laufzeitverhalten untersuchen.

In der vorliegenden Arbeit verbessern wir einen in [6] vorgestellten Algorithmus zur Berechnung der Fundamenteinheiten einer Ordnung und analysieren dessen Laufzeit. Dabei berücksichtigen wir erstmals die Abhängigkeit der Laufzeit vom Grad des zugrundeliegenden Zahlkörpers. Sowohl die Beschreibung als auch die Analyse unseres Algorithmus benutzt Fehlerabschätzungen, die sich im Zusammenhang mit der Approximation von algebraischen Zahlen ergeben.

In unserer Arbeit untersuchen wir ferner die Probleme, diskrete Logarithmen in der Klassengruppe $\text{Cl}_{\mathcal{O}}$ einer Ordnung \mathcal{O} eines algebraischen Zahlkörpers und Erzeuger von \mathcal{O} -Hauptidealen zu berechnen. Diese beiden Probleme stehen in engem Zusammenhang miteinander, aber auch mit dem Problem der Berechnung eines Systems von Fundamenteinheiten und des Regulators von \mathcal{O} .

Zentrales Hilfsmittel unserer Arbeit ist die von uns konstruierte kurze Darstellung algebraischer Zahlen in algebraischen Zahlkörpern. Mit ihrer Hilfe können wir beweisen, daß sowohl die genannten Probleme als auch das Problem, die Klassenzahl einer Ordnung zu bestimmen, in der Komplexitätsklasse $\mathcal{NP} \cap \text{co-}\mathcal{NP}$ liegen, falls die gegebene Ordnung die Hauptordnung und zudem eine verallgemeinerte Riemannsche Vermutung richtig ist. Zudem benutzen wir den Algorithmus zur Berechnung dieser kurzen Darstellung als Unteroutine in den Algorithmen zur Berechnung diskreter Logarithmen, der Fundamenteinheiten und Approximationen des Regulators.

Alle Algorithmen sind deterministisch. Ihre Laufzeiten sind abhängig vom Grad n des Zahlkörpers, von der Diskriminante $\Delta_{\mathcal{O}}$ der Ordnung \mathcal{O} und vom Regulator $R_{\mathcal{O}}$. Die von uns bewiesenen Laufzeiten lassen sich dann wie folgt beschreiben:

Problem	Laufzeit
Fundamenteinheiten	$R_{\mathcal{O}}^{1/2}(\log \Delta)^{O(n)}$
Approximation von $R_{\mathcal{O}}$ (Genauigkeit p Bits)	$R_{\mathcal{O}}^{1/2}(p + \log \Delta)^{O(n)}$
Hauptidealtest und -erzeuger reduzierter Ideale	$R_{\mathcal{O}}^{1/2}(\log \Delta)^{O(n)}$
Diskrete Logarithmen in $\text{Cl}_{\mathcal{O}}$	$ \Delta_{\mathcal{O}} ^{1/4}(\log \Delta)^{O(n)}$

Contents

1	Introduction	3
2	Preliminaries	6
2.1	Notation	6
2.2	Representation, Size, and Algorithms	8
3	Review of Algorithmic Algebraic Number Theory	9
3.1	Algebraic Number Fields	9
3.2	Orders of Algebraic Number Fields	11
3.3	Unit Groups	13
3.4	Ideals and Class Groups	14
3.5	Lattices	17
4	Approximations	27
4.1	Basic Definitions	27
4.2	Quality of Approximations	28
4.3	Approximating Conjugates	36
5	Minima and Reduced Ideals	41
5.1	Definitions and Properties	41
5.2	The Reduction Algorithm	45
5.3	Computing Neighbors	51
6	Binary Multiplicative Representations of Algebraic Numbers	68
6.1	Definitions and Preliminaries	68
6.2	Finding Minima Close to a Given Point	70
6.3	Compact Representations of Algebraic Integer	80
6.4	Applications in Complexity Theory	85

7	Computing Units and Discrete Logarithms in Class Groups	91
7.1	The Main Idea	91
7.2	Computing a System of Fundamental Units	93
7.3	The Containment Problem	117

Chapter 1

Introduction

The computation of the unit group (and of approximations to the regulator) of orders in algebraic number fields is one of the most important and most difficult tasks in computational algebraic number theory. This problem can be studied in a variety of ways. From a practical point of view one is interested in algorithms that can be implemented on a computer and that actually solve the problem “in not to much time for a reasonable input size”. Our point of view is different. Following the ideas of H. W. Lenstra (described in [32]) we are mainly interested in the computational complexity of problems and the asymptotic running time of the algorithms solving these problems. In this sense the best known algorithm for computing the unit group is described in [6], where the author also proves complexity results for his algorithm assuming that the degree n of the number field is fixed. In our work we shall partly improve the algorithms of [6]. We shall refine their analysis and show in which way the running time of the algorithms depends on the degree n . To do so we especially have to study the various situations where in the algorithms rational approximations to real numbers have to be computed. The precision of these approximations influences the correctness and the running time of the algorithms. Obviously, the same is true for computing approximations to the regulator of an order.

Algorithmic number theory has in recent times turned out to be a source of computational problems that can be used as a backbone of cryptographic systems. That means that the security of many cryptographic protocols is based on the assumption that certain problems in algorithmic number theory are computationally hard to solve. For example, the Diffie-Hellman key-exchange system that is described in [21], or systems presented in [22] or [44], are based on the difficulty of solving the *discrete logarithm problem* in the multiplicative group $\text{GF}(p)^*$ of prime fields $\text{GF}(p)$ of characteristic $p > 0$. But since it is by no means clear that the discrete logarithm problem in $\text{GF}(p)^*$ remains difficult in the future (cf. [28]) one must search for other problems that can serve as basis for cryptographic applications. A first step in this direction is to extend the techniques and ideas of the above mentioned protocols to other groups or sets with an appropriate exponentiation function, where the corresponding “discrete logarithm problem” may be harder. This was done for example in [14], [48], where the authors sketch cryptographic systems based on the discrete logarithm problem in the class group $\text{Cl}_{\mathcal{O}}$ of an order \mathcal{O} and on the discrete logarithm problem of \mathcal{O} , i.e. on the problem of computing generators

of principal \mathcal{O} -ideals. In [8] a protocol is proposed that uses the corresponding discrete logarithm problems in arbitrary algebraic number fields.

In our work we describe and analyze deterministic algorithms for solving these discrete logarithm problems. We shall also study algorithms for related problems, namely for principal ideal testing and computing relative generators of ideals. Hence, parts of our work can also be seen as a generalization of [3] and [6]. We give a precise analysis of the algorithms, and again, we show in which way the running time of the algorithms depends on the degree of the number fields.

It is conjectured that for infinitely many algebraic number fields \mathbb{F} the number of bits needed to write down the standard representation of a system of fundamental units of an order \mathcal{O} of \mathbb{F} is exponentially large in $\log |\Delta_{\mathcal{O}}|$, where $\Delta_{\mathcal{O}}$ is the discriminant of \mathcal{O} . In order to prove our complexity results we have to introduce a new representation of algebraic numbers, the so called *compact representation*, which allows us to represent these units using only $(\log |\Delta_{\mathcal{O}}|)^{O(1)}$ bits. Given such a representation of a system of fundamental units we can determine in polynomial time an approximation to the regulator $R_{\mathcal{O}}$ of \mathcal{O} . That kind of representation answers a question suggested by H. W. Lenstra in [32, Problem 5.2]. Using compact representations we can also show that principal ideal testing and computing discrete logarithms in the class group of orders belongs to the complexity class \mathcal{NP} . If \mathcal{O} is the ring of integers of a number field and if a certain generalized Riemann hypothesis is true then we can also show that principal ideal testing and the computation of the class number and compact representations of a system of fundamental units belong to the complexity class $\mathcal{NP} \cap \text{co-}\mathcal{NP}$.

Using the notation of [32] we can summarize our main results concerning running times as follows: Let \mathcal{O} be an order of a number field of degree n , let $\Delta_{\mathcal{O}}$ be the discriminant of \mathcal{O} and let $R_{\mathcal{O}}$ be the regulator of \mathcal{O} . Assume that \mathcal{O} is given by a multiplication table that can be described with $O(n^4(2 + \log |\Delta_{\mathcal{O}}|))$ bits (by [12] or [43] we can transform every multiplication table of \mathcal{O} into another one that satisfies this condition in polynomial time). Then we will prove the following running times:

problem	running time
fundamental units	$R_{\mathcal{O}}^{1/2}(\log \Delta)^{O(n)}$
approximation to $R_{\mathcal{O}}$ (precision p bits)	$R_{\mathcal{O}}^{1/2}(p + \log \Delta)^{O(n)}$
generators of reduced principal \mathcal{O} -ideals	$R_{\mathcal{O}}^{1/2}(\log \Delta)^{O(n)}$
discrete logarithms in $\text{Cl}_{\mathcal{O}}$	$ \Delta_{\mathcal{O}} ^{1/4}(\log \Delta)^{O(n)}$

The structure of this thesis is as follows. In chapter 2 we shall give notations and symbols which will be used in the following text without specific explanation. In chapter 3 we cover the basic terminology and the basic auxiliary results of algorithmic algebraic number theory to be used in later chapters. Chapter 4 is devoted to the problem of approximations to real numbers. This chapter contains many estimations of error bounds

that occur in the context of approximating vectors and matrices with real entries and approximating algebraic numbers and their conjugates. This results shall be used in the following chapters. In chapter 5 we give a short overview of the theory of *minima* and *reduced ideals* as introduced in [4], [5], and [6]. We describe and analyze algorithms for computing a minimum of a given ideal and *neighbors* of that minimum. Chapter 6 considers the problem of representing algebraic numbers as a power product of other “smaller” algebraic numbers. We introduce the above mentioned compact representation, describe and analyze algorithms for determining compact representations and for computing with compact representations, and show the complexity results for the classes \mathcal{NP} and $\text{co-}\mathcal{NP}$. Finally, chapter 7 contains the description and analysis of the algorithms for solving the discrete logarithm problems, computing compact representations of a system of fundamental units, computing approximations to the regulator of an order and computing compact representations of relative generators of ideals.

Chapter 2

Preliminaries

2.1 Notation

We shall denote by \mathbb{N} the set of natural numbers, where we make the convention that 0 is not an element of \mathbb{N} . We denote the set $\mathbb{N} \cup \{0\}$ by \mathbb{N}_0 . The letter \mathbb{Z} denotes the ring of rational integers and the letter \mathbb{P} the set of prime numbers in \mathbb{Z} . The letter \mathbb{Q} denotes the field of rational numbers, the letter \mathbb{R} the field of real numbers, and the letter \mathbb{C} the field of complex numbers. For a subset $A \subseteq \mathbb{R}$ we denote by $A_{>0}$ the set of all positive elements of A . Analogously, we use $A_{\geq 0}$, $A_{<0}$, and $A_{\leq 0}$. For an arbitrary finite set A we denote the cardinality of A by $|A|$.

For convenience, we introduce the following notation. Let $B = (b_k, b_{k+1}, \dots, b_\ell)$ with $k, \ell \in \mathbb{N}$, $k \leq \ell$, be a sequence of elements of a set X . Then we say that b is an element of B and write $b \in B$ if and only if there exists $i \in \mathbb{N}$, $k \leq i \leq \ell$, such that $b = b_i$.

Let X and Y be arbitrary sets, and let $f: X \rightarrow Y$ be a function. For a subset $B \subseteq X$ we denote by $f(B)$ the set $\{y: y = f(x), x \in B\}$. If $B = (b_k, b_{k+1}, \dots, b_\ell)$ with $k, \ell \in \mathbb{N}$, $k \leq \ell$, is a sequence then we denote by $f(B)$ the sequence $(f(b_k), f(b_{k+1}), \dots, f(b_\ell))$.

For a real number z we denote by $|z|$ the absolute value of z . By the symbol $\lceil z \rceil$ we denote the minimum of the set $\{x: x \in \mathbb{Z}, x \geq z\}$ and by $\lfloor z \rfloor$ the maximum of the set $\{x: x \in \mathbb{Z}, x \leq z\}$. The symbol $\lceil z \rceil$ denotes the closest integer to z .

If z is a positive real number then we denote by $\ln(z)$ the natural logarithm, and by $\log(z)$ the logarithm of z to the base 2.

For a complex number $z = x + iy$, where $x, y \in \mathbb{R}$ and $i^2 = -1$, $i \neq 1$, we set $\Re(z) = x$ and $\Im(z) = y$.

Vectors in the m -dimensional linear space \mathbb{R}^m ($m \in \mathbb{N}$) are always assumed to be *column vectors*, i.e. $m \times 1$ -matrices. They are denoted by bold small italic letters. Especially, the origin of \mathbb{R}^m is always denoted by $\mathbf{0}$. The i -th coordinate of a vector \mathbf{a} in \mathbb{R}^m ($1 \leq i \leq m$) is denoted by \mathbf{a}_i . If the bold letter denoting the vector already has a suffix, then that is put after the coordinate suffix. If $A = \{\mathbf{a}_1, \dots, \mathbf{a}_k\}$ ($k \in \mathbb{N}$) is a set of vectors in \mathbb{R}^m , then we denote by $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)$ or by $\text{span}(A)$ the subspace of \mathbb{R}^m generated by $\mathbf{a}_1, \dots, \mathbf{a}_k$. If $k = 0$ resp. $A = \emptyset$, then we consider $\text{span}(A)$ as being $\{\mathbf{0}\}$. The orthogonal complement of a subspace $V \subseteq \mathbb{R}^m$ is denoted by V^\perp .

For $\mathbf{a}, \mathbf{b} \in \mathbb{R}^m$ we denote by

$$\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{i=1}^m a_i b_i$$

the *inner product* of \mathbf{a} and \mathbf{b} . The *euclidean norm* or *length* of $\mathbf{a} \in \mathbb{R}^m$, induced by the inner product, is defined to be

$$\|\mathbf{a}\|_2 = \sqrt{\langle \mathbf{a}, \mathbf{a} \rangle}.$$

We shall also frequently use the *maximum norm* of the vector \mathbf{a} , given by

$$\|\mathbf{a}\|_\infty = \max \{ |a_i| : 1 \leq i \leq m \}.$$

Matrices are denoted by sans serif capitals. If A is a matrix, then the transpose of the matrix A is denoted by A^T . If the inverse of A exists then we denote it by A^{-1} . We simply write $A = (a_{i,j}) \in S^{m \times k}$ ($m, k \in \mathbb{N}$), when we want to state that A is a $m \times k$ -matrix with coefficients $a_{i,j}$ belonging to a set $S \subseteq \mathbb{C}$ ($1 \leq i \leq m$, $1 \leq j \leq k$). We also write $A = [\mathbf{a}_1, \dots, \mathbf{a}_k]$, if $\mathbf{a}_j \in S^m$ denotes the j -th column vector of A . Finally, we denote the length of the shortest column vector of A by $\lambda(A)$.

In this work we will use several matrix norms. The first is the *Frobenius norm*, defined by

$$\|A\|_f = \left(\sum_{i=1}^m \sum_{j=1}^k a_{i,j}^2 \right)^{\frac{1}{2}}.$$

We also mention the *maximum entry norm* of the matrix A , given by

$$\|A\|_\infty = \max \{ |a_{i,j}| : 1 \leq i \leq m, 1 \leq j \leq k \},$$

and the *spectral norm* $\|A\|_2$, which is the largest singular value of A . For properties of those norms we refer to [27].

Lemma 2.1.1 *For each matrix $A \in \mathbb{Z}^{n \times k}$ ($n, k \in \mathbb{N}$) of rank n there exists a unique matrix $H \in \mathbb{Z}^{n \times k}$ and an invertible matrix $U \in \mathbb{Z}^{k \times k}$ with $H = AU$ such that H satisfies the following conditions:*

- (a) *The first n columns of H form an invertible square matrix that is in upper triangular form, i.e., $h_{ij} = 0$ for $1 \leq j < i \leq n$.*
- (b) *The off diagonal entries of that triangular matrix are reduced modulo the diagonal entries, i.e., $0 \leq h_{ij} < h_{ii}$ for $1 \leq i < j \leq n$.*
- (c) *The last $k - n$ columns are zero.*

We call the matrix H in Lemma 2.1.1 the *Hermite normal form* of the matrix A and write $H = \text{HNF}(A)$. More generally, we say that every matrix $H = (h_{i,j}) \in \mathbb{Z}^{n \times k}$ that

satisfies the conditions (a)-(c) of Lemma 2.1.1 is in *Hermite normal form*. We refer to [52, pp. 45-51] or [40] for more details.

Often, we are not interested in the precise values of a function but only in its order of magnitude. In that cases we use the usual o- and O-notation. For definitions and properties of them we refer to [2, Sect. 2.3.]. We shall only use the precise functions, if we fell that they are of special interest on their own, as for example in section 4.2.

2.2 Representation, Size, and Algorithms

To investigate the computational complexity of a problem we have to agree on a way of *encoding* instances of a problem and to *measure the size* of that encoding. In our work we use the notions of encoding and representing data in a similar way as in [32]. Thus, mathematical objects are encoded by finite sequences of rational integers. Each rational integer z is given in the binary notation and has the *binary size*

$$\text{size}(z) = \lfloor \log |z| \rfloor + 2,$$

where the extra bit encodes the sign of z . We also set $\text{size}(0) = 2$. If O is an object encoded by the rational integers n_1, n_2, \dots, n_t ($t \in \mathbb{N}$), then we define its *binary size* $\text{size}(O)$ to be $\sum_{\ell=1}^t \text{size}(n_\ell)$. Clearly, this can only be thought of as proportional to the number of bits needed to code the n_ℓ in the real world. For example, each rational number $x \in \mathbb{Q}$ is represented by a pair of coprime integers (p, q) , where $q > 0$, and $\text{size}(x) = \text{size}(p) + \text{size}(q) = \lfloor \log |p| \rfloor + \lfloor \log |q| \rfloor + 4$.

It is assumed that the reader has an intuitive understanding of the notion of an *algorithm* as being a recipe that given one finite sequence of nonnegative integers, called the *input* data, produces another, called the *output*. Here, we will not give the precise meaning of the notions such as algorithm or *running time* and *complexity*, etc. Let us just mention that we say that an algorithm has *polynomial* running time, if its running time is $\ell^{O(1)}$, where $l \in \mathbb{N}$ is the binary size of the input. In that case we also call the algorithm a *polynomial time algorithm*. For conventions concerning these notions we refer to [32] or [38]. We choose as our “machine model” an idealized computer as described in [61], and the literature given there. In this model, the traditional algorithms for the normal *arithmetical operations*, namely addition and subtraction of rational integers, multiplication and division with remainder, as well as the Euclidean algorithm for the computation of greatest common divisors, have running time $O(l^2)$, where $l \in \mathbb{N}$ is the binary size of the input. If we describe algorithms explicitly, we use a pseudo programming language of which the constructs are similar to the constructs of PASCAL. Often, we combine these constructs with normal text, as proposed in [36] or [20]. Also, in many cases the description of an algorithm is implicitly contained in the constructive proof of the existence of an algorithm or of an estimation of a running time.

When we say that an object is the input for an algorithm then this means that the appropriate encoding is its input. If there is no way mentioned in [32] to represent an object we will describe the encoding as the object arises in the text.

Chapter 3

Review of Algorithmic Algebraic Number Theory

In this chapter we establish the basic definitions and results of algebraic and algorithmic algebraic number theory that we shall use. More details and the proofs of the results stated in this section can be found in [20], [32], [42], and [46]. For proofs of results from the theory of lattices (section 3.5) we also refer to [19] or [41].

3.1 Algebraic Number Fields

Definition 3.1.1 A subfield \mathbb{F} of the complex numbers is called an *algebraic number field* or simply *number field* if \mathbb{F} contains \mathbb{Q} and the dimension of \mathbb{F} as a vector space over \mathbb{Q} is finite. In that case, we call the dimension the *degree* of \mathbb{F} and a basis of \mathbb{F} a *\mathbb{Q} -basis* of \mathbb{F} .

Throughout this chapter, let \mathbb{F} be an algebraic number field of degree n ($n \in \mathbb{N}$).

There are various ways to represent \mathbb{F} in a constructive way (see for example in [20], [32], [45]), and most of them can be transformed one into the other in polynomial time. Here, we shall encode \mathbb{F} by describing the multiplication in \mathbb{F} on a \mathbb{Q} -basis of \mathbb{F} . This comes down to specifying a system of n^3 rational integers $a_{i,j,k} \in \mathbb{Z}$ ($1 \leq i, j, k \leq n$) and a positive rational integer $d \in \mathbb{N}$ such that

$$\omega_i \omega_j = \frac{1}{d} \sum_{k=1}^n a_{i,j,k} \omega_k$$

for some \mathbb{Q} -basis $\Omega = \{\omega_1, \dots, \omega_n\}$ of \mathbb{F} . The pair $(d, (a_{i,j,k}) \in \mathbb{Z}^{n \times n \times n})$ is called the *multiplication table* of the basis Ω and is denoted by $\text{MT}(\Omega)$. By $\|\text{MT}(\Omega)\|_\infty$ we denote the maximum of the absolute values of the $a_{i,j,k}$. Clearly, every number $\alpha \in \mathbb{F}$ can be uniquely written in the form

$$\alpha = \frac{1}{a_{n+1}} \sum_{i=1}^n a_i \omega_i$$

with $a_1, a_2, \dots, a_{n+1} \in \mathbb{Z}$, $a_{n+1} > 0$ and $\gcd(a_1, a_2, \dots, a_{n+1}) = 1$. We encode α by the sequence $(a_1, a_2, \dots, a_{n+1})$ which we shall call the *standard representation* of α with respect to the given basis Ω . Whenever we talk about operations with numbers in \mathbb{F} we

will assume that \mathbb{F} is given by a multiplication table of a basis and the operands are given in terms of their standard representation with respect to the same basis, and that the result has the same form.

By [32, Sect. 2.8] one sees that there is a polynomial time algorithm that given a positive rational integer and a system of n^3 rational integers decides if those numbers define a number field. Also by [32, Sect. 2.8], we have

Lemma 3.1.2 *There are polynomial time algorithms that given an algebraic number field \mathbb{F} and two elements $\alpha, \beta \in \mathbb{F}$, $\beta \neq 0$, determine the sum $\alpha + \beta$, the product $\alpha\beta$, and the quotient α/β .*

Definition 3.1.3 An *algebraic number* is a complex number α that is a root of a polynomial $a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \cdots + a_n = 0$ ($n \in \mathbb{N}$), where $a_0, a_1, \dots, a_n \in \mathbb{Q}$ and $a_0 \neq 0$. An *algebraic integer* ω is a complex number that is a root of a polynomial $x^n + b_1x^{n-1} + b_2x^{n-2} + \cdots + b_n = 0$, where $b_1, \dots, b_n \in \mathbb{Z}$.

If α is an algebraic number then α is a root of a unique monic irreducible polynomial $f(x)$ in $\mathbb{Q}[x]$, called the *minimal polynomial* of α . If the degree of the minimal polynomial is n , then α is called an *algebraic number of degree n* .

Theorem 3.1.4 *Let \mathbb{F} be an algebraic number field of degree $n \in \mathbb{N}$. Then there exists an algebraic number $\rho \in \mathbb{F}$ of degree n such that $F = \mathbb{Q}(\rho)$.*

We call ρ in Theorem 3.1.4 a *primitive element* of \mathbb{F} . The minimal polynomial of a primitive element of \mathbb{F} is called a *generating polynomial* of \mathbb{F} . The *signature* of \mathbb{F} is the pair $(s, t) \in \mathbb{N}_0 \times \mathbb{N}_0$, where s is the number of real zeros of a generating polynomial and t is the number of pairs of non real zeros; clearly, we have $s + 2t = n$. We note that the signature is independent of the choice of the generating polynomial and thus is an invariant of the number field.

In the following, we shall always assume that \mathbb{F} is a number field of signature (s, t) . If ρ is a primitive element of \mathbb{F} , then the numbers $1, \rho, \rho^2, \dots, \rho^{n-1}$ form a \mathbb{Q} -basis of \mathbb{F} . Hence, each $\alpha \in \mathbb{F}$ can be uniquely represented in the form

$$\alpha = \frac{1}{a_{n+1}} \sum_{i=1}^n a_i \rho^{i-1},$$

where $a_1, a_2, \dots, a_{n+1} \in \mathbb{Z}$, $a_{n+1} > 0$ and $\gcd(a_1, a_2, \dots, a_{n+1}) = 1$. Let $g(x)$ be the minimal polynomial of ρ , and let $\rho^{(1)}, \dots, \rho^{(n)}$ be the n complex roots of $g(x)$. Then we call $\alpha^{(1)}, \alpha^{(2)}, \dots, \alpha^{(n)}$ with

$$\alpha^{(j)} = \frac{1}{a_{n+1}} \sum_{i=1}^n a_i \left(\rho^{(j)} \right)^{i-1}$$

for $1 \leq j \leq n$ the *algebraic conjugates* of α . The n monomorphisms

$$\sigma_j: \mathbb{F} \rightarrow \mathbb{C}, \alpha \mapsto \alpha^{(j)}$$

are called the *embeddings* of \mathbb{F} into the field \mathbb{C} . Throughout this work we shall always assume that the zeros of a generating polynomial $g(x)$ are ordered such that for $1 \leq j \leq s$ we have $\varrho^{(j)} \in \mathbb{R}$ and $\varrho^{(j+t)} = \overline{\varrho^{(j)}} \in \mathbb{C} - \mathbb{R}$ for j with $s < j \leq m$, where $m = s + t$, and where the bar indicates complex conjugation. The order of the real zeros as well as the order of the nonreal zeros is assumed to be fixed. Thus, for $1 \leq j \leq s$ we have $\alpha^{(j)} \in \mathbb{R}$ and $\alpha^{(j+t)} = \overline{\alpha^{(j)}} \in \mathbb{C} - \mathbb{R}$ for j with $s < j \leq m$.

Definition 3.1.5 We define the *norm* of an element α in \mathbb{F} to be the number

$$N_{\mathbb{F}/\mathbb{Q}}(\alpha) = \prod_{i=1}^n \alpha^{(i)}.$$

Proposition 3.1.6 *The norm is a multiplicative map from \mathbb{F} onto \mathbb{Q} , i.e. for $\alpha \in \mathbb{F}$ we have $N_{\mathbb{F}/\mathbb{Q}}(\alpha) \in \mathbb{Q}$, and for any $\alpha, \beta \in \mathbb{F}$ we have $N_{\mathbb{F}/\mathbb{Q}}(\alpha\beta) = N_{\mathbb{F}/\mathbb{Q}}(\alpha)N_{\mathbb{F}/\mathbb{Q}}(\beta)$.*

We let $|\cdot|_1, \dots, |\cdot|_m$ be the *normalized archimedean valuations* on \mathbb{F} , i.e. for $\alpha \in \mathbb{F}$ we define

$$|\alpha|_j = \begin{cases} |\alpha^{(j)}| & \text{if } 1 \leq j \leq s, \\ |\alpha^{(j)}|^2 & \text{if } s+1 \leq j \leq m. \end{cases}$$

The normalized archimedean valuations have the property, that for $\alpha, \beta \in \mathbb{F}$ and for $1 \leq j \leq m$ we have $|\alpha\beta|_j = |\alpha|_j |\beta|_j$. We also know that $|\alpha|_j = 0$ if and only if $\alpha = 0$.

Proposition 3.1.7 *Let \mathbb{F} be a number field of signature (s, t) , and let $m = s + t$. Let $\alpha \in \mathbb{F}$. Then we have*

$$|N_{\mathbb{F}/\mathbb{Q}}(\alpha)| = \prod_{i=1}^m |\alpha|_i. \quad (3.1)$$

Definition 3.1.8 For $\alpha \in \mathbb{F}$ we define the *height* of α to be the number

$$H(\alpha) = \max \{r : r = |\alpha|_i, 1 \leq i \leq m\}.$$

3.2 Orders of Algebraic Number Fields

Definition 3.2.1 A subset M of a number field \mathbb{F} of degree n is called a *module* of \mathbb{F} if there exist elements $\gamma_1, \gamma_2, \dots, \gamma_n \in M$ such that for each $\gamma \in M$ there is a unique sequence of n elements $b_1, b_2, \dots, b_n \in \mathbb{Z}$ such that $\gamma = \sum_{i=1}^n b_i \gamma_i$. The sequence $(\gamma_1, \gamma_2, \dots, \gamma_n)$ is called a \mathbb{Z} -*basis* of M , and

$$\Delta_M = \left(\det \left(\gamma_i^{(j)} \right) \right)^2$$

the *discriminant* of M .

The discriminant of a module M of \mathbb{F} is independent of the choice of the \mathbb{Z} -basis. If $N \subseteq M$ is a module of \mathbb{F} then we have

$$\Delta_N = [M : N]^2 \Delta_M, \quad (3.2)$$

where $[M : N]$ is the order of the finite factor module M/N . Moreover, we have

$$[M : N] = |\det(\mathbf{A})|, \quad (3.3)$$

where $\mathbf{A} = (a_{i,j}) \in \mathbb{Z}^{n \times n}$ is a matrix such that there exist a basis $(\gamma_1, \gamma_2, \dots, \gamma_n)$ of M and a basis $(\beta_1, \beta_2, \dots, \beta_n)$ of N with

$$\beta_k = \sum_{i=1}^n a_{k,i} \gamma_i$$

for $1 \leq k \leq n$.

Definition 3.2.2 An *order* \mathcal{O} of a number field \mathbb{F} is a subring of \mathbb{F} containing 1 that also is a module of \mathbb{F} .

We will encode an order \mathcal{O} of a number field \mathbb{F} in a similar way as \mathbb{F} , that is to say, by a multiplication table $(1, (a_{i,j,k})) \in \mathbb{Z}^{n \times n \times n}$ with the property that there is a \mathbb{Z} -basis $\Omega = \{\omega_1, \dots, \omega_n\}$ of \mathcal{O} with

$$\omega_i \omega_j = \sum_{k=1}^n a_{i,j,k} \omega_k$$

for $1 \leq i, j \leq n$.

Lemma 3.2.3 *Let \mathcal{O} be an order of a number field \mathbb{F} . Then each \mathbb{Z} -basis of \mathcal{O} also is a \mathbb{Q} -basis of \mathbb{F} .*

From the above lemma it follows that $\text{MT}(\Omega) = (1, (a_{i,j,k}))$ also encodes \mathbb{F} . Since, on the other hand, given a number field \mathbb{F} one can construct an order of \mathbb{F} in polynomial time, we shall assume that \mathbb{F} is always given in this way, i.e., by a multiplication table of a \mathbb{Z} -basis of an order of \mathbb{F} .

Proposition 3.2.4 *There exist a polynomial time algorithm that given an order \mathcal{O} determines the discriminant $\Delta_{\mathcal{O}}$.*

Proposition 3.2.5 *There exist a polynomial time algorithm that given an order \mathcal{O} computes a multiplication table of \mathcal{O} of binary size $O(n^4(2 + \log |\Delta_{\mathcal{O}}|))$.*

A very detailed proof of this result can be found in [12] or [43]. For convenience, we shall call a multiplication table of binary size $O(n^4(2 + \log |\Delta_{\mathcal{O}}|))$ *short*.

In the literature the time bounds for algorithms that have an order \mathcal{O} as a part of their input never depend on the specific multiplication table by which \mathcal{O} is given

but only on the discriminant $\Delta_{\mathcal{O}}$ (see for example [32]). This is justified by the above propositions. We follow the tradition and always assume that orders are represented by short multiplication tables.

We also use the following result that follows from [42, p. 70].

Proposition 3.2.6 *Let \mathcal{O} be an order of a number field of degree $n \geq 2$. Then we have*

$$n \leq 2(\log |\Delta_{\mathcal{O}}|) / \log(3).$$

Among all orders of \mathbb{F} there is a unique maximal one (with respect to inclusion) which we shall denote by $\mathcal{O}_{\mathbb{F}}$. This maximal order is equal to the set $\mathbb{Z}_{\mathbb{F}}$ of all algebraic integers of \mathbb{F} . Thus, every order is a subset of $\mathbb{Z}_{\mathbb{F}}$. The discriminant of $\mathcal{O}_{\mathbb{F}}$ is called the *discriminant of \mathbb{F}* and denoted by $\Delta_{\mathbb{F}}$. Finally, we want to remind the reader of the following helpful results:

Lemma 3.2.7 *Let \mathcal{O} be an order of a number field \mathbb{F} . Then we have $N_{\mathbb{F}/\mathbb{Q}}(\alpha) \in \mathbb{Z}$ for all $\alpha \in \mathcal{O}$. For $\alpha \in \mathbb{Q}$ we have $N_{\mathbb{F}/\mathbb{Q}}(\alpha) = \alpha^n$.*

Lemma 3.2.8 *Let \mathcal{O} be an order of a number field \mathbb{F} , and let $\alpha \in \mathcal{O}$. Then we have $N_{\mathbb{F}/\mathbb{Q}}(\alpha)/\alpha \in \mathcal{O}$.*

3.3 Unit Groups

Let \mathcal{O} be an order of a number field \mathbb{F} . A number $\xi \in \mathcal{O}$ such that $1/\xi$ also belongs to \mathcal{O} is called a *unit* of \mathcal{O} . The set of all units of \mathcal{O} is a multiplicative abelian group that is called the *unit group* of \mathcal{O} and is denoted by \mathcal{O}^* . Units can also be described in the following way:

Proposition 3.3.1 *Let \mathcal{O} be an order of a number field \mathbb{F} , and let $\xi \in \mathcal{O}$. Then ξ is a unit of \mathcal{O} if and only if $N_{\mathbb{F}/\mathbb{Q}}(\xi) = \pm 1$.*

By Dirichlet's Unit Theorem (cf. for example [42]), we know

Theorem 3.3.2 *Let \mathcal{O} be an order of an algebraic number field of signature (s, t) . Then there exist units $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_r \in \mathcal{O}^*$ with $r = s + t - 1$, such that every unit $\varepsilon \in \mathcal{O}^*$ can be uniquely represented in the form*

$$\varepsilon = \zeta \varepsilon_1^{a_1} \varepsilon_2^{a_2} \cdots \varepsilon_r^{a_r},$$

where $\zeta \in \mathcal{O}^*$ is a root of unity and $a_1, a_2, \dots, a_r \in \mathbb{Z}$.

Definition 3.3.3 Let the notations be as in Theorem 3.3.2. Then we call the set $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_r\}$ a *system of fundamental units of \mathcal{O}* . A system of fundamental units of the maximal order of a number field \mathbb{F} is called a *system of fundamental units of \mathbb{F}* .

Definition 3.3.4 Let \mathcal{O} be an order of a number field of signature (s, t) , and let $r = s + t - 1$. Then we define the *regulator* $R_{\mathcal{O}}$ of \mathcal{O} to be the absolute value of the determinant of the matrix $(\ln |\varepsilon_i|_j) \in \mathbb{R}^{r \times r}$, where $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_r\}$ is a system of fundamental units of the order \mathcal{O} .

Clearly, the regulator is an invariant of the order and independent of the chosen system of fundamental units.

3.4 Ideals and Class Groups

Definition 3.4.1 Let \mathfrak{A} and \mathfrak{B} be two nonempty subsets of \mathbb{F} . Then we define their *product* to be the set

$$\mathfrak{A}\mathfrak{B} = \left\{ \gamma : \gamma = \sum_{i=1}^{\ell} \alpha_i \beta_i, \ell \in \mathbb{N}, \alpha_i \in \mathfrak{A}, \beta_i \in \mathfrak{B} \text{ for } 1 \leq i \leq \ell \right\}.$$

If \mathfrak{A} only contains one element α then we simply write $\alpha\mathfrak{B}$ instead of $\{\alpha\}\mathfrak{B}$.

Definition 3.4.2 Let \mathcal{O} be an order of a number field \mathbb{F} . An *ideal* of \mathcal{O} is a module $\mathfrak{A} \subseteq \mathcal{O}$ of \mathbb{F} such that $\mathcal{O}\mathfrak{A} \subseteq \mathfrak{A}$. A *fractional ideal* of \mathcal{O} is a module \mathfrak{A} of \mathbb{F} such that $d\mathfrak{A}$ is an ideal of \mathcal{O} for some positive rational integer d . The minimal such d is called the *denominator* of \mathfrak{A} (with respect to \mathcal{O}) and is denoted by $d(\mathfrak{A})$.

In this work, we will always assume that ideals are nonzero. Clearly, every ideal of an order \mathcal{O} is a fractional ideal of \mathcal{O} with denominator 1. Thus, abusing our notation we shall no longer distinguish between ideals and fractional ideals. Also note that \mathcal{O} itself is an ideal (of \mathcal{O}).

Let \mathfrak{A} be an ideal of an order \mathcal{O} with denominator $d(\mathfrak{A})$, and let $\Omega = \{\omega_1, \dots, \omega_n\}$ be a \mathbb{Z} -basis of \mathcal{O} . If $\{\beta_1, \beta_2, \dots, \beta_n\}$ is a \mathbb{Z} -basis of $d(\mathfrak{A})\mathfrak{A}$ then there exists a matrix $A = (a_{i,j}) \in \mathbb{Z}^{n \times n}$ with

$$\beta_j = \sum_{i=1}^n a_{j,i} \omega_i$$

for $1 \leq j \leq n$. We call the pair $(d(\mathfrak{A}), A)$ a *matrix representation* of \mathfrak{A} with respect to the \mathbb{Z} -basis Ω . Unfortunately, that representation is not unique. In fact, if $U \in \mathbb{Z}^{n \times n}$ is an invertible matrix and $H = (h_{i,j}) = AU$ then $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ with

$$\alpha_k = \sum_{i=1}^n h_{k,i} \omega_i$$

for $1 \leq k \leq n$ is another basis of $d(\mathfrak{A})\mathfrak{A}$ and thus $(d(\mathfrak{A}), AU)$ is another matrix representation of \mathfrak{A} . On the other hand, every basis and matrix representation can be obtained in this way. Applying Lemma 2.1.1 we conclude that we can encode an ideal \mathfrak{A} by a matrix

representation $(d(\mathfrak{A}), \mathbf{H})$ of \mathfrak{A} with respect to the \mathbb{Z} -basis Ω , where $\mathbf{H} = \text{HNF}(\mathbf{A})$ is a matrix in Hermite normal form. We call this representation the *standard representation* of the ideal \mathfrak{A} with respect to Ω . Whenever we talk about operations with ideals of an order \mathcal{O} we assume that they are given in the standard representation with respect to the same \mathbb{Z} -basis which is implicitly described by the multiplication table by which \mathcal{O} is given. Then that representation is unique, i.e., two ideals \mathfrak{A} and \mathfrak{B} of \mathcal{O} are equal, if and only if their standard representations are equal.

If \mathfrak{A} and \mathfrak{B} are ideals of an order \mathcal{O} and $\alpha \in \mathbb{F} - \{0\}$ then the products $\mathfrak{A}\mathfrak{B}$ and $\alpha\mathfrak{A}$ are ideals of \mathcal{O} , too.

Definition 3.4.3 We say that two ideals \mathfrak{A} and \mathfrak{B} of an order \mathcal{O} are *equivalent* and write $\mathfrak{A} \sim \mathfrak{B}$, if there exists $\alpha \in \mathbb{F}$ such that $\mathfrak{A} = \alpha\mathfrak{B}$. Any such α is called a *generator of \mathfrak{A} relative to \mathfrak{B}* . If $\mathfrak{B} = \mathcal{O}$ then \mathfrak{A} is called a *principal ideal* of \mathcal{O} and α is simply called a *generator* of \mathfrak{A} .

Using ideals of an order \mathcal{O} we find a third characterization of units of \mathcal{O} .

Lemma 3.4.4 *Let \mathcal{O} be an order and let \mathfrak{A} be an ideal of \mathcal{O} . A number $\varepsilon \in \mathcal{O}$ is a unit of \mathcal{O} if and only if $\varepsilon\mathfrak{A} = \mathfrak{A}$.*

Definition 3.4.5 An ideal \mathfrak{A} of an order \mathcal{O} is called *invertible in \mathcal{O}* if there exists an ideal \mathfrak{B} such that $\mathfrak{A}\mathfrak{B} = \mathcal{O}$. In that case we call \mathfrak{B} the *inverse of \mathfrak{A} in \mathcal{O}* and write $\mathfrak{B} = \mathfrak{A}^{-1}$.

For convenience, we say that an ideal \mathfrak{A} of \mathcal{O} is invertible when we mean that \mathfrak{A} is invertible in \mathcal{O} . This inaccuracy shall not lead to confusions since it shall always be clear from the context in which order the ideal is invertible.

Proposition 3.4.6 *Any principal ideal of an order is invertible.*

By [11, Sect. 5] and [25, Sect. 6.1] we have

Lemma 3.4.7 *There are polynomial time algorithms that given an order \mathcal{O} , ideals \mathfrak{A} and \mathfrak{B} of \mathcal{O} and $\alpha \in \mathbb{F} - \{0\}$ determine $\mathfrak{A}\mathfrak{B}$ and $\alpha\mathfrak{A}$. There is also a polynomial time algorithm that given an order \mathcal{O} and an ideal \mathfrak{A} of \mathcal{O} decides whether \mathfrak{A} is invertible, and in that case determines \mathfrak{A}^{-1} .*

Clearly, the set $\mathcal{I}_{\mathcal{O}}$ of all invertible ideals of an order \mathcal{O} is a multiplicative abelian group with the neutral element \mathcal{O} , in which the set $\mathcal{P}_{\mathcal{O}}$ of all principal ideals of \mathcal{O} is a subgroup.

Definition 3.4.8 The factor group $\text{Cl}_{\mathcal{O}} = \mathcal{I}_{\mathcal{O}}/\mathcal{P}_{\mathcal{O}}$ is called the *class group* of \mathcal{O} , its elements are called the *ideal classes* of \mathcal{O} , and its cardinality is called the *class number* of \mathcal{O} and is denoted by $h_{\mathcal{O}}$. The ideal class of an ideal \mathfrak{A} of \mathcal{O} is denoted by $[\mathfrak{A}]$.

Since the class group of an order is always finite, the class number is well defined. By [47] we can bound the product of the class number and the regulator of an order. We have

Theorem 3.4.9 *Let \mathbb{F} be a number field of degree n and signature (s, t) , and let \mathcal{O} be an order of \mathbb{F} with discriminant $\Delta_{\mathcal{O}}$. Then we have*

$$2^s h_{\mathcal{O}} R_{\mathcal{O}} < 2n(n+1) \left(\frac{4}{n-1} \right)^{n-1} \sqrt{|\Delta_{\mathcal{O}}|} (\log |\Delta_{\mathcal{O}}|)^{n-1} (\log \log |\Delta_{\mathcal{O}}|)^{n/2}.$$

If $[\mathcal{O}_{\mathbb{F}} : \mathcal{O}] > 1$ then

$$[\mathcal{O}_{\mathbb{F}}^* : \mathcal{O}^*] < 2[\mathcal{O}_{\mathbb{F}} : \mathcal{O}] \log \log (3[\mathcal{O}_{\mathbb{F}} : \mathcal{O}]^2).$$

Now, we can explain the *discrete logarithm problem in the class group* of an order \mathcal{O} . The problem is to decide for two reduced invertible ideals \mathfrak{A} and \mathfrak{D} of \mathcal{O} whether there exists a positive rational integer y such that $\mathfrak{A} \sim \mathfrak{D}^y$ resp. $[\mathfrak{A}] = [\mathfrak{D}]^y$. In that case we let $\log_{[\mathfrak{D}]}([\mathfrak{A}])$ be the minimal such y . Otherwise we set $\log_{[\mathfrak{D}]}([\mathfrak{A}]) = 0$.

Definition 3.4.10 The *norm* of an ideal \mathfrak{A} of an order \mathcal{O} is defined to be

$$N_{\mathcal{O}}(\mathfrak{A}) = \frac{[\mathcal{O} : d(\mathfrak{A})\mathfrak{A}]}{d(\mathfrak{A})^n}.$$

By (3.2) it follows

Proposition 3.4.11 *Let \mathfrak{A} be an ideal of \mathcal{O} . Then we have $\Delta_{\mathfrak{A}} = \Delta_{\mathcal{O}}(N_{\mathcal{O}}(\mathfrak{A}))^2$.*

By (3.3) and since \mathbf{A} is in Hermite normal form we also obtain

Proposition 3.4.12 *Let \mathfrak{A} be an ideal of \mathcal{O} and let $(d(\mathfrak{A}), \mathbf{A})$ with $\mathbf{A} = (a_{i,j}) \in \mathbb{Z}^{n \times n}$ be the standard representation of \mathfrak{A} . Then we have*

$$N_{\mathcal{O}}(\mathfrak{A}) = \frac{\det(\mathbf{A})}{d(\mathfrak{A})^n} = \frac{\prod_{i=1}^n a_{i,i}}{d(\mathfrak{A})^n}.$$

Proof. The assertions immediately follows from (3.3) and the properties of matrices in Hermite normal form. \square

Corollary 3.4.13 *Let \mathfrak{A} be an ideal of \mathcal{O} . Then we have*

$$\text{size}(\mathfrak{A}) \leq (n^2 + 1)(\log(d(\mathfrak{A})) + 2) + n^2 \log(N_{\mathcal{O}}(\mathfrak{A})).$$

and

$$N_{\mathcal{O}}(\mathfrak{A}) \leq 2^{\text{size}(\mathfrak{A})}.$$

Proof. To decode the standard representation $(d(\mathfrak{A}), \mathbf{A})$ with $\mathbf{A} = (a_{i,j}) \in \mathbb{Z}^{n \times n}$ of \mathfrak{A} we have to write down $d(\mathfrak{A})$ and the n^2 integers $a_{i,j}$ ($1 \leq i, j \leq n$) in binary notation. Since \mathbf{A} is in Hermite normal form the assertion follows. Clearly, the second assertion is a consequence of Proposition 3.4.12. \square

In general, it is not true that $N_{\mathcal{O}}(\mathfrak{A}\mathfrak{B}) = N_{\mathcal{O}}(\mathfrak{A})N_{\mathcal{O}}(\mathfrak{B})$ for two ideals $\mathfrak{A}, \mathfrak{B}$ of an order \mathcal{O} . Instead of this we have

Proposition 3.4.14 *Let $\mathfrak{A}, \mathfrak{B}$ be ideals of an order \mathcal{O} . If either \mathfrak{A} or \mathfrak{B} is invertible then we have $N_{\mathcal{O}}(\mathfrak{A}\mathfrak{B}) = N_{\mathcal{O}}(\mathfrak{A})N_{\mathcal{O}}(\mathfrak{B})$.*

We shall often use the connection between the norm of ideals of an order and the norm of elements of the number field. That connection is described in the following lemma.

Lemma 3.4.15 *Let \mathcal{O} be an order of a number field \mathbb{F} .*

- (a) *For any ideal \mathfrak{A} of \mathcal{O} and any $\alpha \in \mathfrak{A}$, $\alpha \neq 0$, we have $|N_{\mathbb{F}/\mathbb{Q}}(\alpha)| \geq N_{\mathcal{O}}(\mathfrak{A})$.*
- (b) *For $\mathfrak{B} = \beta\mathcal{O}$ with $\beta \in \mathbb{F} - \{0\}$ we have $|N_{\mathbb{F}/\mathbb{Q}}(\beta)| = N_{\mathcal{O}}(\mathfrak{B})$.*

3.5 Lattices

Within this section let n, k be natural numbers with $k \leq n$.

Definition 3.5.1 A *lattice* Λ in the real euclidean space \mathbb{R}^n is an additive subgroup of \mathbb{R}^n of the form

$$\{\mathbf{c}: \mathbf{c} = \sum_{i=1}^k x_i \mathbf{a}_i, x_i \in \mathbb{Z} \text{ for } 1 \leq i \leq k\},$$

where $\mathbf{a}_1, \dots, \mathbf{a}_k$ ($1 \leq k \leq n$) are linearly independent vectors in \mathbb{R}^n . The sequence $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ is called a *basis*, and the number k is called the *dimension* of Λ .

For convenience, we shall often consider a basis as a matrix consisting of the vectors of the basis. Thus, when we say that the matrix $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_k] \in \mathbb{R}^{n \times k}$ is a basis of a lattice Λ in \mathbb{R}^n , we mean that $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ is a basis of Λ .

A basis of a lattice is uniquely determined by that lattice *up to unimodular transformations*. That means, that if $\mathbf{A} \in \mathbb{R}^{n \times k}$ is a basis of the lattice $\Lambda \subseteq \mathbb{R}^n$ then for any matrix $\mathbf{V} \in \mathbb{Z}^{k \times k}$ satisfying $|\det(\mathbf{V})| = 1$ the matrix $\mathbf{A}\mathbf{V}$ is a basis of the same lattice. On the other hand each basis of Λ can be obtained in this way. Thus, we can define the following value which is independent of the choice of a basis.

Definition 3.5.2 The *determinant* $\det(\Lambda)$ of the lattice Λ which has a basis $\mathbf{A} \in \mathbb{R}^{n \times k}$ is defined by

$$\det(\Lambda) = |\det(\mathbf{A}^T \mathbf{A})|^{\frac{1}{2}}.$$

Geometrically, the determinant can be interpreted as the volume of any *fundamental parallelepiped* of a lattice Λ , i.e. a set

$$\left\{ \mathbf{b}: \mathbf{b} \in \mathbb{R}^n, \mathbf{b} = \sum_{i=1}^k x_i \mathbf{a}_i, x_i \in \mathbb{R}, 0 \leq x_i < 1, 1 \leq i \leq k \right\},$$

where $\{\mathbf{a}_1, \dots, \mathbf{a}_k\}$ is a basis of Λ . Note that the determinant is an invariant of Λ while a fundamental parallelepiped depends on the chosen basis.

If Λ_1 and $\Lambda_2 \subseteq \mathbb{R}^n$ are two n -dimensional lattices with $\Lambda_1 \subseteq \Lambda_2$, then Λ_1 is called a *sublattice* of Λ_2 .

Proposition 3.5.3 *Let Λ_1 and $\Lambda_2 \subseteq \mathbb{R}^n$ be two n -dimensional lattices with $\Lambda_1 \subseteq \Lambda_2$. Then $\det(\Lambda_1)$ is an integral multiple of $\det(\Lambda_2)$.*

A question we shall often be concerned with is whether there exists a nonzero lattice point in a suitable subset of \mathbb{R}^n . A positive answer is given by a theorem of Minkowski. To formulate it we need some more definitions.

A non-empty subset S of \mathbb{R}^n is called *convex* if for any two points $\mathbf{v}, \mathbf{w} \in S$ and any real number t with $0 < t < 1$ the point $t\mathbf{v} + (1-t)\mathbf{w}$ also belongs to S . The set S is called *symmetric with regard to the origin* if for any $\mathbf{v} \in S$ we also have $-\mathbf{v} \in S$.

Theorem 3.5.4 *If S is a convex subset of \mathbb{R}^n , symmetric with regard to the origin and of volume $\text{vol}(S)$, and Λ is a n -dimensional lattice in the same space such that the inequality*

$$\text{vol}(S) > 2^n \det(\Lambda)$$

holds, then the set S contains a nonzero lattice point $\mathbf{x} \in \Lambda$.

An additive subgroup of \mathbb{R}^n is a lattice if and only if it is *discrete*, that is to say, there exists a constant $\delta \in \mathbb{R}$, $\delta > 0$, such that for any two distinct elements \mathbf{x}, \mathbf{y} in the subgroup we have $\|\mathbf{x} - \mathbf{y}\|_2 \geq \delta$. This implies the existence of other important invariants of lattices, the so called *successive minima*.

Definition 3.5.5 Let Λ be a lattice in \mathbb{R}^n of dimension k and let $i \in \mathbb{N}$, $1 \leq i \leq k$. Then the minimal positive real number r with the property that there exist linearly independent vectors $\mathbf{v}_1, \dots, \mathbf{v}_i$ in Λ satisfying $\|\mathbf{v}_j\|_2 \leq r$ for all j with $1 \leq j \leq i$ is called the *i -th successive minimum* of Λ and is denoted by $\lambda_i(\Lambda)$.

The following chain of inequalities is a trivial consequence of the definition:

$$0 < \lambda_1(\Lambda) \leq \lambda_2(\Lambda) \leq \dots \leq \lambda_k(\Lambda). \quad (3.4)$$

We call any nonzero vector \mathbf{v} of a lattice Λ with $\|\mathbf{v}\|_2 = \lambda_1(\Lambda)$ a *shortest* vector of Λ . In order to bound the length of shortest vectors or more general, the size of successive minima of lattices, we introduce

Definition 3.5.6 For $k \in \mathbb{N}$ the k -th Hermite constant γ_k is defined by

$$\gamma_k = \sup \left\{ r : r = \lambda_1(\Lambda)^2 \det(\Lambda)^{-\frac{2}{k}}, \Lambda \subseteq \mathbb{R}^k \text{ is a lattice of dimension } k \right\}.$$

The Hermite constants are explicitly known only for $k \leq 8$. But applying Theorem 3.5.4 it can easily be shown that for $k \in \mathbb{N}$ we have

$$\gamma_k \leq k. \quad (3.5)$$

Theorem 3.5.7 Let $n \in \mathbb{N}$, and let Λ be a lattice of dimension n in \mathbb{R}^n . Then

$$\prod_{i=1}^n \lambda_i(\Lambda) \leq \gamma_n^{\frac{n}{2}} \det(\Lambda).$$

Especially, we have

$$\lambda_1(\Lambda) \leq \gamma_n^{\frac{1}{2}} \det(\Lambda)^{\frac{1}{n}}. \quad (3.6)$$

Before we describe special lattices arising in the context of algebraic number fields, we need a very basic construction from linear algebra and some helpful estimates.

Definition 3.5.8 Let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ be a sequence of linearly independent vectors in \mathbb{R}^m . Then the sequence $(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$ of their Gram-Schmidt vectors is defined by

$$\mathbf{a}_1^* = \mathbf{a}_1 \quad \text{and} \quad \mathbf{a}_i^* = \mathbf{a}_i - \sum_{j=1}^{i-1} \frac{\langle \mathbf{a}_i, \mathbf{a}_j^* \rangle}{\langle \mathbf{a}_j^*, \mathbf{a}_j^* \rangle} \mathbf{a}_j^* \text{ for } 2 \leq i \leq k.$$

In the following lemma we summarize some properties of Gram-Schmidt vectors we shall often refer to. Their proofs can be found for example in [29].

Lemma 3.5.9 Let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ be a sequence of linearly independent vectors in \mathbb{R}^m . Then the vectors $\mathbf{a}_1^*, \dots, \mathbf{a}_k^*$ are mutually orthogonal. For $1 \leq i \leq k$ the vector \mathbf{a}_i^* is the orthogonal projection of \mathbf{a}_i onto $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_{i-1})^\perp$, and we have

$$\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle = \langle \mathbf{a}_i^*, \mathbf{a}_i \rangle, \quad \text{and} \quad \|\mathbf{a}_i^*\|_2 \leq \|\mathbf{a}_i\|_2. \quad (3.7)$$

Moreover, let $\mathbf{c} = \sum_{i=1}^k x_i \mathbf{a}_i$, where $x_i \in \mathbb{R}$ for $1 \leq i \leq k$. Then we have

$$\mathbf{c} = \sum_{i=1}^k \frac{\langle \mathbf{c}, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \mathbf{a}_i^*. \quad (3.8)$$

Finally, for $A = (\mathbf{a}_1, \dots, \mathbf{a}_k) \in \mathbb{R}^{m \times k}$ and $A^* = [\mathbf{a}_1^*, \dots, \mathbf{a}_k^*]$, we have

$$\left(\det(A^T A) \right)^{\frac{1}{2}} = \left(\det(A^{*T} A^*) \right)^{\frac{1}{2}} = \prod_{i=1}^k \|\mathbf{a}_i^*\|_2 \leq \prod_{i=1}^k \|\mathbf{a}_i\|_2. \quad (3.9)$$

Using the Gram-Schmidt vectors we can also find a lower bound on the length of the shortest nonzero vector in a lattice.

Corollary 3.5.10 *Let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ be a basis of a lattice $\Lambda \in \mathbb{R}^n$, and let \mathbf{v} be a nonzero vector in Λ . Then we have*

$$\|\mathbf{v}\|_2 \geq \min \{\|\mathbf{a}_1^*\|_2, \dots, \|\mathbf{a}_k^*\|_2\}.$$

Proof. Let \mathbf{v} be a nonzero vector in Λ . Then we can write $\mathbf{v} = \sum_{j=1}^{\ell} x_j \mathbf{a}_j$, where $1 \leq \ell \leq k$, $x_i \in \mathbb{Z}$ for $1 \leq i \leq \ell$, and $x_\ell \neq 0$. By (3.8) we also have

$$\mathbf{v} = \sum_{i=1}^{\ell} \frac{\langle \mathbf{v}, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \mathbf{a}_i^* = \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} x_j \frac{\langle \mathbf{a}_j, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \mathbf{a}_i^*.$$

Since for $j < i$ the vectors \mathbf{a}_j and \mathbf{a}_i^* are orthogonal we obtain

$$\mathbf{v} = \sum_{i=1}^{\ell-1} \sum_{j=i}^{\ell} x_j \frac{\langle \mathbf{a}_j, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \mathbf{a}_i^* + x_\ell \mathbf{a}_\ell^*,$$

and therefore

$$\|\mathbf{v}\|_2^2 \geq x_\ell^2 \|\mathbf{a}_\ell^*\|_2^2 \geq \|\mathbf{a}_\ell^*\|_2^2, \quad (3.10)$$

where the right inequality of (3.10) follows from $x_\ell \in \mathbb{Z}$. This proves the assertion. \square

Next, we give some helpful estimates and introduce another notation. We start with the well known *Schwarz inequality*.

Lemma 3.5.11 *Let $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$. Then*

$$|\langle \mathbf{a}, \mathbf{b} \rangle| \leq \|\mathbf{a}\|_2 \|\mathbf{b}\|_2 \leq n \|\mathbf{a}\|_\infty \|\mathbf{b}\|_\infty. \quad (3.11)$$

We immediately obtain

Lemma 3.5.12 *For $\mathbf{a} \in \mathbb{R}^n$ we have $\|\mathbf{a}\|_2 \leq \sqrt{n} \|\mathbf{a}\|_\infty$.*

Corollary 3.5.13 *Let $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n] \in \mathbb{R}^{n \times n}$. Then we have*

$$\prod_{i=1}^n \|\mathbf{a}_i\|_2 \leq (\sqrt{n} \|\mathbf{A}\|_\infty)^n.$$

Lemma 3.5.14 *For $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$ we have $\|\mathbf{AB}\|_\infty \leq n \|\mathbf{A}\|_\infty \|\mathbf{B}\|_\infty$.*

We also need the well known *rule of Cramer*.

Lemma 3.5.15 Let $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n] \in \mathbb{R}^{n \times n}$ be a matrix of rank n , and let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ be vectors with $\mathbf{B}\mathbf{x} = \mathbf{y}$. Then for all i with $1 \leq i \leq n$ we have

$$\mathbf{x}_i = \frac{1}{\det(\mathbf{B})} \det([\mathbf{b}_1, \dots, \mathbf{b}_{i-1}, \mathbf{y}, \mathbf{b}_{i+1}, \dots, \mathbf{b}_n]).$$

Lemma 3.5.16 Let $\Lambda \subseteq \mathbb{Z}^n$ be a lattice with basis $(\mathbf{b}_1, \dots, \mathbf{b}_n)$. For $1 \leq i \leq n$ let $d_i = \prod_{j=1}^i \|\mathbf{b}_j^*\|_2$. Then we have

$$d_{i-1} \mathbf{b}_i^* \in \Lambda \subseteq \mathbb{Z}^n, \quad (3.12)$$

and for $1 \leq j < i \leq n$

$$d_j \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \in \mathbb{Z}. \quad (3.13)$$

Proof. Since $\mathbf{b}_j^* \in \text{span}(\mathbf{b}_1, \dots, \mathbf{b}_j)$ for $1 \leq j \leq n$, by the definition of the Gram-Schmidt vectors it follows that there exist $\delta_{i,j} \in \mathbb{R}$ with $\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \delta_{i,j} \mathbf{b}_j$ ($1 \leq i \leq n$). Thus, we have for $1 \leq \ell \leq i-1$

$$\langle \mathbf{b}_i, \mathbf{b}_\ell \rangle = \sum_{j=1}^{i-1} \delta_{i,j} \langle \mathbf{b}_j, \mathbf{b}_\ell \rangle.$$

We note that $\langle \mathbf{b}_i^*, \mathbf{b}_\ell \rangle = 0$ for $1 \leq \ell < i \leq n$. From Cramer's rule (see Lemma 3.5.15) and from (3.9) we thus obtain $d_{i-1} \delta_{i,j} \in \mathbb{Z}$. This proves (3.12) since $d_{i-1} \in \mathbb{Z}$ and

$$d_{i-1} \mathbf{b}_i^* = d_{i-1} \mathbf{b}_i - \sum_{j=1}^{i-1} d_{i-1} \delta_{i,j} \mathbf{b}_j.$$

Hence applying (3.12) and since $\|\mathbf{b}_i^*\|_2^2 = d_i/d_{i-1}$ we conclude that

$$d_j \mu_{i,j} = d_j \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} = d_{j-1} \langle \mathbf{b}_i, \mathbf{b}_j^* \rangle = \langle \mathbf{b}_i, d_{j-1} \mathbf{b}_j^* \rangle \in \mathbb{Z},$$

which proves (3.13). □

Definition 3.5.17 For a matrix $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n] \in \mathbb{R}^{n \times n}$ of rank n we define

$$\text{dft}(\mathbf{A}) = \frac{1}{|\det(\mathbf{A})|} \prod_{i=1}^n \|\mathbf{a}_i\|_2.$$

As an immediate consequence of Cramer's rule and Hadamard's inequality (see (3.9)) we obtain an upper bound for the solution of a linear system.

Corollary 3.5.18 *Let $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n] \in \mathbb{R}^{n \times n}$ be a matrix of rank n , and let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ be vectors with $\mathbf{B}\mathbf{x} = \mathbf{y}$. Then for $1 \leq i \leq n$ we have*

$$|\mathbf{x}_i| \leq \frac{1}{\lambda(\mathbf{B})|\det(\mathbf{B})|} \|\mathbf{y}\|_2 \prod_{j=1}^n \|\mathbf{b}_j\|_2 = \|\mathbf{y}\|_2 \frac{\text{dft}(\mathbf{B})}{\lambda(\mathbf{B})}.$$

Using Corollary 3.5.18 and Lemma 3.5.12 we can also estimate several norms of the inverse of a matrix.

Corollary 3.5.19 *Let $\mathbf{B} \in \mathbb{R}^{n \times n}$ be a matrix of rank n . Then we have*

$$\|\mathbf{B}^{-1}\|_\infty \leq \frac{\text{dft}(\mathbf{B})}{\lambda(\mathbf{B})} \quad \text{and} \quad \|\mathbf{B}^{-1}\|_f \leq n \frac{\text{dft}(\mathbf{B})}{\lambda(\mathbf{B})}.$$

The purpose of the rest of this section is to describe methods to embed a number field in a real vector space, in such a way that certain subsets of the number field map to lattices in this vector space. This opens the way to applications of Minkowski's theorems.

In the following let \mathbb{F} be an algebraic number field of degree n and signature (s, t) , and let $m = s + t$ and $r = s + t - 1$. We start by describing a map that transforms algebraic numbers into vectors.

Definition 3.5.20 We define the *Minkowski map* of the number field \mathbb{F} to be the map

$$\cdot: \mathbb{F} \longrightarrow \mathbb{R}^n, \quad \alpha \longmapsto \underline{\alpha} = (\alpha^{(1)}, \dots, \alpha^{(s)}, \Re(\alpha^{(s+1)}), \dots, \Re(\alpha^{(m)}), \Im(\alpha^{(s+1)}), \dots, \Im(\alpha^{(m)})).$$

For $S \subseteq \mathbb{F}$ we denote by \underline{S} the set $\{\mathbf{v}: \mathbf{v} = \underline{\alpha}, \alpha \in S\}$.

Proposition 3.5.21 *Let M be a module of \mathbb{F} with basis $(\gamma_1, \gamma_2, \dots, \gamma_n)$. Then \underline{M} is a n -dimensional lattice in \mathbb{R}^n with basis $(\underline{\gamma}_1, \underline{\gamma}_2, \dots, \underline{\gamma}_n)$ and determinant*

$$\det(\underline{M}) = 2^{-t} |\Delta_M|^{\frac{1}{2}}.$$

For $\alpha \in \mathbb{F}$ we call $\underline{\alpha}$ the *conjugate vector* of α . If M is a module of \mathbb{F} then we call \underline{M} the *Minkowski lattice* of M . In the special case of the module being an ideal of an order we can give a lower bound for the shortest vector of the corresponding Minkowski lattice.

Proposition 3.5.22 *Let \mathfrak{A} be an ideal of an order \mathcal{O} of \mathbb{F} . Then we have*

$$\lambda_1(\underline{\mathfrak{A}}) \geq \sqrt{\frac{n}{2}} (\mathbf{N}_{\mathcal{O}}(\mathfrak{A}))^{\frac{1}{n}}.$$

Proof. Let α be a nonzero element of the ideal \mathfrak{A} . Then from Lemma 3.4.15 we know

$$|\mathbf{N}_{\mathbb{F}/\mathbb{Q}}(\alpha)| = \prod_{i=1}^n |\alpha^{(i)}| \geq \mathbf{N}_{\mathcal{O}}(\mathfrak{A}).$$

Applying the arithmetic-mean–geometric-mean inequality we also have

$$|\mathbb{N}_{\mathbb{F}/\mathbb{Q}}(\alpha)|^{\frac{2}{n}} \leq \frac{1}{n} \sum_{i=1}^n |\alpha^{(i)}|^2 \leq \frac{2}{n} \sum_{i=1}^m |\alpha^{(i)}|^2 = \frac{2}{n} \|\underline{\alpha}\|_2^2.$$

This implies the assertion. \square

Next, we describe the dependence between the length of the conjugate vector of an algebraic number and the size of their standard representation.

Lemma 3.5.23 *Let \mathcal{O} be an order of \mathbb{F} given by a multiplication table $\text{MT}(\Omega)$ of a \mathbb{Z} -basis $\Omega = (\omega_1, \dots, \omega_n)$ of \mathcal{O} . Let $\alpha \in \mathbb{F}$ (given in standard representation). Then we have for all j with $1 \leq j \leq m$*

$$|\alpha^{(j)}| \leq n^2 \|\text{MT}(\Omega)\|_{\infty} 2^{\text{size}(\alpha)}.$$

Proof. Let $\alpha = (1/a_{n+1}) \sum_{i=1}^n a_i \omega_i$, where $a_i \in \mathbb{Z}$ for $1 \leq i \leq n+1$, and suppose that $\text{MT}(\Omega) = (1, (a_{i,j,k})) \in \mathbb{Z}^{n \times n \times n}$. Furthermore, let $\ell, h \in \mathbb{N}$, $1 \leq \ell \leq n$, $1 \leq h \leq m$, such that $|\omega_{\ell}^{(h)}| \geq |\omega_i^{(j)}|$ for all $i, j \in \mathbb{N}$ with $1 \leq i \leq n$ and $1 \leq j \leq m$. Then we have

$$\left| \omega_{\ell}^{(h)} \right|^2 \leq \sum_{k=1}^n |a_{\ell, \ell, k}| \left| \omega_{\ell}^{(h)} \right|,$$

and therefore

$$\left| \omega_{\ell}^{(h)} \right| \leq n \|\text{MT}(\Omega)\|_{\infty}.$$

Since $\alpha^{(j)} = (1/a_{n+1}) \sum_{i=1}^n a_i \omega_i^{(j)}$, for $1 \leq j \leq n$, this implies

$$|\alpha^{(j)}| \leq n^2 \max\{|a_1|, \dots, |a_n|\} \|\text{MT}(\Omega)\|_{\infty} \leq n^2 2^{\text{size}(\alpha)} \|\text{MT}(\Omega)\|_{\infty}. \quad \square$$

We immediately obtain

Corollary 3.5.24 *Let \mathcal{O} be an order of \mathbb{F} given by a multiplication table $\text{MT}(\Omega)$ of a \mathbb{Z} -basis $\Omega = (\omega_1, \dots, \omega_n)$ of \mathcal{O} . Let $\alpha \in \mathbb{F}$ (given in standard representation). Then we have*

$$\|\underline{\alpha}\|_2 \leq n^3 \|\text{MT}(\Omega)\|_{\infty} 2^{\text{size}(\alpha)}.$$

Lemma 3.5.25 *Let \mathcal{O} be an order of \mathbb{F} given by a multiplication table $\text{MT}(\Omega)$ of a \mathbb{Z} -basis $\Omega = (\omega_1, \dots, \omega_n)$ of \mathcal{O} . Let $\alpha \in \mathbb{F}$ (given in standard representation) be an element of an ideal \mathfrak{A} of \mathcal{O} . Then we have*

$$\text{size}(\alpha) \leq n \log \left(d(\mathfrak{A}) \sqrt{n} \max\{\mathbf{H}(\alpha), (\mathbf{H}(\alpha))^2\} \frac{2^n (n^3 \|\text{MT}(\Omega)\|_{\infty} 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \log(d(\mathfrak{A})).$$

Proof. We use the notation of the proof of Lemma 3.5.23. Let $\mathcal{O} = (\underline{\omega}_1, \dots, \underline{\omega}_n)$. Then we have $\mathcal{O}\mathbf{a} = a_{n+1}\underline{\alpha}$, where $\mathbf{a} = (a_1, \dots, a_n)$. By Corollary 3.5.18, Corollary 3.5.24 and Proposition 3.5.21 and Proposition 3.5.22 this implies that for $1 \leq i \leq n$

$$a_i \leq a_{n+1} \|\underline{\alpha}\|_2 \frac{\text{dft } \mathcal{O}}{\lambda(\mathcal{O})} \leq d(\mathfrak{A}) \|\underline{\alpha}\|_2 \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}}.$$

Here, we have used that $a_{n+1} \leq d(\mathfrak{A})$ and $\text{size}(\omega_i) \leq 2n$ for $1 \leq i \leq n$. Now, the assertion follows from Lemma 3.5.12 and the observation that $\|\underline{\alpha}\|_\infty \leq \max\{\mathbf{H}(\alpha), (\mathbf{H}(\alpha))^2\}$. \square

A second way to map subsets of the number field \mathbb{F} to a lattice is to use a logarithmic embedding of \mathbb{F} .

Definition 3.5.26 We define the *Dirichlet map* of \mathbb{F} to be the map

$$\text{Log} : \mathbb{F} - \{0\} \rightarrow \mathbb{R}^r, \alpha \mapsto \text{Log } \alpha = (\ln |\alpha|_1, \dots, \ln |\alpha|_r)^T.$$

By Dirichlet's unit theorem we have

Proposition 3.5.27 *Let \mathcal{O} be an order of \mathbb{F} and let $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_r\}$ a system of fundamental units of \mathcal{O} . Then the image $\text{Log } \mathcal{O}^*$ of \mathcal{O}^* is a r -dimensional lattice in \mathbb{R}^r with basis $(\text{Log } \varepsilon_1, \text{Log } \varepsilon_2, \dots, \text{Log } \varepsilon_r)$ and determinant*

$$\det(\text{Log } \mathcal{O}^*) = R_{\mathcal{O}}.$$

Proposition 3.5.28 *Let \mathcal{O} be an order of \mathbb{F} and let $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_r$ be units of \mathcal{O} such that $(\text{Log } \varepsilon_1, \text{Log } \varepsilon_2, \dots, \text{Log } \varepsilon_r)$ is a basis of $\text{Log } \mathcal{O}^*$. Then $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_r\}$ is a system of fundamental units of \mathcal{O} .*

For $\alpha \in \mathbb{F} - \{0\}$ we call $\text{Log } \alpha$ the *logarithm vector* of α . If \mathcal{O} is an order then we call $\text{Log } \mathcal{O}^*$ the *unit lattice* of \mathcal{O} .

As in the case of Minkowski lattices we can give a lower bound of the length of a shortest vector. By [34] we have

Proposition 3.5.29 *Let \mathcal{O} be an order of \mathbb{F} . Then we have*

$$\lambda_1(\text{Log } \mathcal{O}^*) \geq \max \left\{ \left(\frac{2}{n} \right)^{\frac{1}{2}} \left(\frac{1}{2000} \left(\frac{\log \log n}{\log n} \right)^3 - \frac{1}{2880000} \left(\frac{\log \log n}{\log n} \right)^6 \right), \frac{16}{17r} \frac{1}{4^{2+t}} \right\}, \quad (3.14)$$

and

$$\frac{1}{\lambda_1(\text{Log } \mathcal{O}^*)} \leq 2n. \quad (3.15)$$

Proof. Since $\mathcal{O}^* \subseteq \mathcal{O}_{\mathbb{F}}^*$, it is sufficient to prove the lower bound for $\lambda_1(\text{Log } \mathcal{O}_{\mathbb{F}}^*)$. But then by [34] we have

$$\lambda_1(\text{Log } \mathcal{O}_{\mathbb{F}}^*) \geq \left(\frac{2}{r+1}\right)^{\frac{1}{2}} \left(\frac{1}{2000} \left(\frac{\log \log n}{\log n}\right)^3 - \frac{1}{2880000} \left(\frac{\log \log n}{\log n}\right)^6\right).$$

By [49] for every unit $\varepsilon \in \mathcal{O}_{\mathbb{F}}^*$ that is not a root of unity there exists $i \in \mathbb{N}$, $1 \leq i \leq m$, such that

$$|\varepsilon|_i > 1 + c, \quad (3.16)$$

where $c = 1/4^{2+t}$.

Now, we examine the function $f(x) = \ln(1+x) - (16/17)x$. Its derivative is $f'(x) = 1/(x+1) - 16/17$. If $x \leq 1/16$ then $f'(x) \geq 0$. Since $f(0) = 0$ this implies that $f(x) \geq 0$ for $0 \leq x \leq 1/16$ and therefore

$$\ln(1+x) \geq (16/17)x \quad \text{for } 0 \leq x \leq 1/16. \quad (3.17)$$

Since $c \leq 1/16$ we obtain from (3.16) and (3.17)

$$\ln |\varepsilon|_i \geq \frac{16}{17}c.$$

On the other hand, by Proposition 3.3.1 we have $\sum_{j=1}^m \ln |\varepsilon|_j = 0$. Thus there must exist $j \in \mathbb{N}$, $1 \leq j \leq r$, such that

$$\left| \ln |\varepsilon|_j \right| \geq \frac{16}{17r}c.$$

This proves the first assertion. The second one can be easily seen by using that the left term in the maximum part of (3.14) is bigger than the right for $n \geq 100$. \square

We can also find an upper bound of the successive minima of a unit lattice.

Proposition 3.5.30 *Any order \mathcal{O} contains a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that*

$$\|\text{Log } \varepsilon_i\|_2 \leq \left(\frac{r(r+3)}{4}\right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16}\right)^{r-1} \quad \text{for } 1 \leq i \leq r. \quad (3.18)$$

Proof. Let $\{\varepsilon_1, \varepsilon_2, \dots, \varepsilon_r\}$ be a system of fundamental units of \mathcal{O} . By Proposition 3.5.27 $\text{Log } \mathcal{O}^*$ is a r -dimensional lattice in \mathbb{R}^r with basis $(\text{Log } \varepsilon_1, \text{Log } \varepsilon_2, \dots, \text{Log } \varepsilon_r)$ and determinant

$$\det(\text{Log } \mathcal{O}^*) = R_{\mathcal{O}}.$$

Theorem 5.2.6 implies, that there exists a system $\{\varepsilon_1, \dots, \varepsilon_r\}$ of fundamental units of \mathcal{O} such that $(\text{Log } \varepsilon_1, \text{Log } \varepsilon_2, \dots, \text{Log } \varepsilon_r)$ is a Korkine-Zolotaref reduced basis of the lattice $\text{Log } \mathcal{O}^*$. Hence, by Proposition 5.2.5 and (3.5) we have

$$\prod_{i=1}^r \|\text{Log } \varepsilon_i\|_2 \leq \left(\frac{r(r+3)}{4}\right)^{\frac{r}{2}} R_{\mathcal{O}},$$

and by Proposition 3.5.29

$$\|\mathrm{Log} \varepsilon_i\|_2 \leq \left(\frac{r(r+3)}{4}\right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16}\right)^{r-1} \quad \text{for } 1 \leq i \leq r. \quad (3.19)$$

This proves the assertion. \square

We shall also use the following very crude bound:

Corollary 3.5.31 *Any order \mathcal{O} contains a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that for $1 \leq i \leq r$ we have*

$$\|\mathrm{Log} \varepsilon_i\|_2 \leq (2r)^r 2^{(n+1)r} 4n^2 \sqrt{|\Delta_{\mathcal{O}}|} (\log |\Delta_{\mathcal{O}}|)^{n-1} (\log \log |\Delta_{\mathcal{O}}|)^{n/2}.$$

Proof. By Proposition 3.5.30 there exists a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that for all $1 \leq i \leq r$ we have

$$\|\mathrm{Log} \varepsilon_i\|_2 \leq \left(\frac{r(r+3)}{4}\right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16}\right)^{r-1}.$$

Now, the assertion follows from Theorem 3.4.9 and some simple estimations. \square

Chapter 4

Approximations

4.1 Basic Definitions

The elements of algebraic number fields are real or complex numbers. If we use their standard representations then we can perform the ordinary arithmetical operations as addition or multiplication using only rational integers. But in the following parts of this work we will be more and more involved in computations where we have to work with reals. Since we can not really compute with real numbers but only with rationals, we shall describe how to represent these numbers by appropriate approximations and to estimate the occurring rounding errors.

Thorough this chapter let m, n and q always be natural numbers.

Definition 4.1.1 A rational number z' is called a q -approximation to a real number z if $|z - z'| < 2^{-q-1}$.

Clearly, if we have a q -approximation z' to $z \in \mathbb{R}$ then we can compute in time linear in q and the size of z' a number $a \in \mathbb{Z}$ such that $z'' = a2^{-q}$ is a q -approximation to z too.

Definition 4.1.2 A complex number z' is called an q -approximation to the complex number z if $\Re(z')$ and $\Im(z')$ are q -approximations to $\Re(z)$ and $\Im(z)$.

Definition 4.1.3 A vector $\mathbf{v}' \in \mathbb{Q}^n$ is called a q -approximation to a vector $\mathbf{v} \in \mathbb{R}^n$ if \mathbf{v}'_i is a q -approximation to \mathbf{v}_i for $1 \leq i \leq n$.

Definition 4.1.4 A matrix $A' = (a'_{i,j}) \in \mathbb{R}^{n \times n}$ is called a q -approximation to a matrix $A = (a_{i,j}) \in \mathbb{R}^{n \times n}$ if $a'_{i,j}$ is an q -approximation to $a_{i,j}$ for $1 \leq i \leq n, 1 \leq j \leq n$.

If we talk about q -approximations then we call q the *precision* of the approximation. Instead of “is a q -approximation to” we sometimes also use the phrase “is an approximation of precision q to”.

Definition 4.1.5 A function $f: \mathbb{R}^m \rightarrow \mathbb{Q}^m$ is called an *approximating function of precision q* , if for all $\mathbf{v} \in \mathbb{R}^m$ the image $f(\mathbf{v})$ is an q -approximation to \mathbf{v} .

Definition 4.1.6 We say that a set $\Lambda' \subseteq \mathbb{Q}^m$ is a q -approximation to a set $\Lambda \subseteq \mathbb{R}^m$, if there exists an approximating function f of precision q , such that $f(\Lambda) = \Lambda'$.

Definition 4.1.7 We say that a sequence $B' = (\mathbf{v}'_1, \dots, \mathbf{v}'_\ell)$ of vectors in \mathbb{Q}^n is a q -approximation to a sequence $B = (\mathbf{v}_1, \dots, \mathbf{v}_\ell)$ of vectors in \mathbb{R}^n if there exists an approximating function f of precision q such that $B' = f(B)$.

By a simple observation we have

Proposition 4.1.8

- (a) Let $z' \in \mathbb{Q}$ be a q -approximation to $z \in \mathbb{R}$, and let $a \in \mathbb{Q}$. Then az' is an approximation of precision $(q - \lceil \log |a| \rceil)$ to az .
- (b) If $z' \in \mathbb{Q}$ is a q -approximation to $z \in \mathbb{R}$ and $y' \in \mathbb{Q}$ is a q -approximation to $y \in \mathbb{R}$, then $z' + y'$ is an approximation of precision $q - 1$ to $z + y$.
- (c) Let $z'_1, \dots, z'_\ell \in \mathbb{Q}$ ($\ell \in \mathbb{N}$) be q -approximations to the numbers $z_1, \dots, z_\ell \in \mathbb{R}$. Then their sum $\sum_{i=1}^{\ell} z'_i$ is an approximation of precision $q - \lceil \log(\ell) \rceil$ to $\sum_{i=1}^{\ell} z_i$.

4.2 Quality of Approximations

In this section we examine and describe the quality of approximations. After some preliminary observations, we start by examining the influence of approximations in the context of QR factorizations (see [58],[27]). For the proofs of the following results we use “brute force” calculus and techniques from numerical analysis and matrix theory, similar to those of the proof of [56, Theorem3.1]. We shall use assumptions that are weaker than those in [56, Theorem 3.1]. Furthermore, our bounds are sharper and more convenient to use.

In the later part of this chapter we describe the behaviour of linear equations and determinants in the presence of approximations. For the proofs of those error estimations we use straightforward algebraic transformations, similar to [9]. But by a refinement of the analysis we obtain bounds that are more precise than those presented there.

We start with the following simple observations.

Lemma 4.2.1 Let $z \in \mathbb{C}$, and let z' be a q -approximation to z . Then we have

$$\left| |z| - |z'| \right| \leq |z - z'| < 2^{-q}. \quad (4.1)$$

Let $\mathbf{v} \in \mathbb{R}^n$ and let $\mathbf{v}' \in \mathbb{Q}^n$ be a q -approximation to \mathbf{v} . Then we have

$$\|\mathbf{v} - \mathbf{v}'\|_2 \leq \sqrt{n} 2^{-q-1}. \quad (4.2)$$

Let $\mathbf{A} \in \mathbb{R}^{n \times n}$ and let \mathbf{A}' be a q -approximation to \mathbf{A} . Then we have

$$\|\mathbf{A} - \mathbf{A}'\|_f \leq n 2^{-q-1}. \quad (4.3)$$

Proof. Equation (4.1) is an immediate consequence of Definition 4.1.2, since by triangular inequality we have

$$\left| |z| - |z'| \right| \leq |z - z'| = \left(|\Re(z - z')|^2 + |\Im(z - z')|^2 \right)^{\frac{1}{2}} \leq \sqrt{2} 2^{-q-1} < 2^q.$$

Also, (4.2) follows from Definition 4.1.3 and Lemma 3.5.12. Finally, the inequality

$$\|A - A'\|_f \leq (n^2(2^{-q-1})^2)^{\frac{1}{2}} = n2^{-q-1}$$

proves (4.3). \square

Lemma 4.2.2 *Let $p \in \mathbb{N}$, and let $z' \in \mathbb{Q}$ be a p -approximation to $z \in \mathbb{R}$, $z, z' > 0$. Let $q \in \mathbb{Z}_{>0}$ and let $\kappa \in \mathbb{R}_{>0}$ with $\kappa > \ln(2)$. If $p > -\ln(z) + \kappa q$ then we have $|\ln(z') - \ln(z)| < 2^{-q}$.*

Proof. Since $-\ln(1+z) > -1/z - \ln(z)$ for every $z \in \mathbb{R}_{>0}$ we have

$$-\ln(1 + e^{\kappa q}) > -e^{-\kappa q} - \kappa p,$$

and therefore

$$\kappa q + 2^{-q} - \ln(1 + e^{\kappa q}) > 2^{-q} - e^{-\kappa q} > 0,$$

where the last inequality follows from $\kappa > \ln(2)$. Simple algebraic transformations imply that $\kappa q > -\ln(e^{2^{-q}} - 1)$. Thus, if $p > -\ln(z) + \kappa q$, then $p > -\ln(z) - \ln(e^{2^{-q}} - 1)$. We set $\varepsilon = z' - z$. Then it follows that

$$|\ln(z') - \ln(z)| = \ln \left| \frac{z + \varepsilon}{z} \right| = \ln \left| 1 + \frac{\varepsilon}{z} \right| \leq \ln \left| 1 + \frac{2^{-p-1}}{z} \right| \leq 2^{-q}. \quad \square$$

A *QR factorization* of a matrix $A \in \mathbb{R}^{n \times n}$ of rank n is a factorization of A of the form $A = QR$, where $R = (r_{i,j}) \in \mathbb{R}^{n \times n}$ is an upper triangular matrix of rank n with positive diagonal elements and $Q \in \mathbb{R}^{n \times n}$ is orthonormal, that is to say $QQ^T = I$, where I is the identity matrix. It is well known that the QR factorization is unique. The columns \mathbf{q}_i ($1 \leq i \leq m$) of Q are just the vectors that would be obtained by applying the Gram-Schmidt orthogonalization to the columns of A and then normalizing the obtained Gram-Schmidt vectors; thus we have $\mathbf{q}_i = \mathbf{a}_i^* / \|\mathbf{a}_i^*\|_2$ for $1 \leq i \leq m$, where \mathbf{a}_i resp. \mathbf{q}_i is the i -th column vector of A resp. Q . Furthermore, we know that the entry $r_{i,i}$ of R satisfies $r_{i,i} = \|\mathbf{a}_i^*\|_2$. For $1 \leq \ell \leq m-1$ the last $m-\ell$ columns of Q form an orthonormal basis of the orthogonal complement of the space generated by the first ℓ columns of A . We start by the following observation. The frobenius norm is *invariant under orthonormal transformations*, that is to say that for any orthogonal matrix $P \in \mathbb{R}^{n \times n}$ and any matrix $B \in \mathbb{R}^{n \times n}$ we have $\|PB\|_f = \|B\|_f$ (see [27]). Thus, we obtain

Proposition 4.2.3 *Let $A \in \mathbb{R}^{n \times n}$ be a matrix of rank n and let $A = QR$ be its QR factorization. Then*

$$\|A^{-1}\|_f = \|R^{-1}\|_f \quad \text{and} \quad \|A\|_f = \|R\|_f.$$

The effect of approximations on QR factorizations is explained in the following theorem.

Theorem 4.2.4 *Let $A \in \mathbb{R}^{n \times n}$ be a matrix of rank n and let $A = QR$ be its QR factorization. Let $A' \in \mathbb{R}^{n \times n}$ be a q -approximation to A , and let $c \in \mathbb{R}$, $0 < c < 1$. If*

$$q \geq \log(\|A^{-1}\|_f) + 3 \log\left(\frac{3}{\sqrt{2}}(n+6)\right) - \min\left\{0, \log\left(\frac{n+6}{\|A\|_f}\right)\right\} - \log(c) \quad (4.4)$$

then A' has rank n , and there exists a QR factorization of A' of the form $A' = (Q + W)(R + F)$, where $W \in \mathbb{R}^{n \times n}$ and $F \in \mathbb{R}^{n \times n}$ such that

$$\|W\|_f \leq c \quad \text{and} \quad \|F\|_f \leq c. \quad (4.5)$$

Proof. Let \mathfrak{L} be a linear mapping that maps the space $U(\mathbb{R}, m)$ of upper triangular $n \times n$ -matrices with real entries to the space $S(\mathbb{R}, n)$ of symmetric $n \times n$ -matrices, given by $\mathfrak{L}(X) = X + X^T$. Since for any symmetric matrix H there is a unique upper triangular matrix X such that $H = \mathfrak{L}(X)$, the map \mathfrak{L} is injective and surjective. Furthermore, since by [55] we have $\sqrt{2}\|X\|_f \leq \|X + X^T\|_f = \|H\|_f$, it follows that

$$\|\mathfrak{L}^{-1}(H)\|_f \leq (1/\sqrt{2})\|H\|_f. \quad (4.6)$$

Let $E = A' - A$. Then by (4.3) the matrix $E \in \mathbb{R}^{n \times n}$ satisfies

$$\|E\|_f < n2^{-q}, \quad (4.7)$$

and we have $A' = A + E$. Let $M(\mathbb{R}, n)$ be the set of all $n \times n$ -matrices.

Now, we consider at the function $\Phi: M(\mathbb{R}, n) \rightarrow U(\mathbb{R}, n)$, $X \mapsto \mathfrak{L}^{-1}(Y - X^T X)$, where

$$Y = Q^T E R^{-1} + (Q^T E R^{-1})^T + (E R^{-1})^T E R^{-1} \quad (4.8)$$

is a symmetric matrix. Applying (4.6) we obtain for all $X_1, X_2 \in U(\mathbb{R}, n)$

$$\begin{aligned} \|\Phi(X_1) - \Phi(X_2)\|_f &\leq \|\mathfrak{L}^{-1}(Y - X_1 X_1^T) - \mathfrak{L}^{-1}(Y - X_2 X_2^T)\|_f \\ &= \|\mathfrak{L}^{-1}(X_1 X_1^T - X_2 X_2^T)\|_f \\ &\leq \frac{1}{\sqrt{2}} \|X_1 X_1^T - X_2 X_2^T\|_f \\ &\leq \sqrt{2} \max\{\|X_1\|_f, \|X_2\|_f\} \|X_1 - X_2\|_f, \end{aligned}$$

and

$$\begin{aligned} \|\Phi(X_2) - X_2\|_f &= \|\mathfrak{L}^{-1}(Y - \mathfrak{L}^{-1}(X_2 X_2^T)) - X_2\|_f \\ &\leq \frac{1}{\sqrt{2}} \left(\|Y\|_f + \|X_2\|_f^2 + \|X_2\|_f \right). \end{aligned}$$

By Banach's fixpoint theorem (see for example [58]) there exists a fixpoint \mathbf{X}_0 of Φ such that $\|\mathbf{X}_0\|_f \leq (1/\sqrt{2})\|\mathbf{Y}\|_f$. From (4.8) and since the frobenius norm is invariant under orthogonal transformations, it follows that

$$\begin{aligned} \|\mathbf{X}_0\|_f &\leq \frac{1}{\sqrt{2}} \left(\|\mathbf{Q}^T \mathbf{E} \mathbf{R}^{-1}\|_f + \|(\mathbf{Q}^T \mathbf{E} \mathbf{R}^{-1})^T\|_f + \|(\mathbf{E} \mathbf{R}^{-1})^T\|_f \|\mathbf{E} \mathbf{R}^{-1}\|_f \right) \\ &\leq \sqrt{2} \|\mathbf{E} \mathbf{R}^{-1}\|_f + \frac{1}{\sqrt{2}} \|\mathbf{E} \mathbf{R}^{-1}\|_f^2. \end{aligned}$$

Now, using Proposition 4.2.3 we have

$$\|\mathbf{X}_0\|_f \leq \sqrt{2} \|\mathbf{A}^{-1}\|_f \|\mathbf{E}\|_f + \frac{1}{\sqrt{2}} (\|\mathbf{A}^{-1}\|_f \|\mathbf{E}\|_f)^2. \quad (4.9)$$

From (4.4) and (4.7) we derive

$$\begin{aligned} \|\mathbf{A}^{-1}\|_f \|\mathbf{E}\|_f &\leq \|\mathbf{A}^{-1}\|_f \left(\left(\frac{c}{n+6} \right) \frac{\sqrt{2}}{3} \|\mathbf{A}^{-1}\|_f^{-1} \right) \min \left\{ 1, \frac{n+6}{\|\mathbf{A}\|_f} \right\} \\ &\leq \left(\frac{c}{n+6} \right) \frac{\sqrt{2}}{3} \min \left\{ 1, \frac{n+6}{\|\mathbf{A}\|_f} \right\}, \end{aligned} \quad (4.10)$$

and thus $\|\mathbf{X}_0\|_f \leq 1$. Applying [58] we see that $(\mathbf{I} + \mathbf{X}_0)^{-1}$ exists, where

$$\|(\mathbf{I} + \mathbf{X}_0)^{-1}\|_f \leq \frac{1}{1 - \|\mathbf{X}_0\|_f}. \quad (4.11)$$

As a solution of the equation $\mathbf{X} = \mathfrak{L}^{-1}(\mathbf{Y} - \mathbf{X}^T \mathbf{X})$, the matrix \mathbf{X}_0 is an upper triangular matrix. By (4.8) it satisfies

$$\mathbf{X}_0 + \mathbf{X}_0^T + \mathbf{X}_0^T \mathbf{X}_0 = \mathbf{Q}^T \mathbf{E} \mathbf{R}^{-1} + (\mathbf{Q}^T \mathbf{E} \mathbf{R}^{-1})^T + (\mathbf{E} \mathbf{R}^{-1})^T \mathbf{E} \mathbf{R}^{-1}.$$

Multiplying with \mathbf{R}^T from the left and \mathbf{R} from the right we see that

$$\mathbf{R}^T \mathbf{X}_0 \mathbf{R} + \mathbf{R}^T \mathbf{X}_0^T \mathbf{R} + \mathbf{R}^T \mathbf{X}_0^T \mathbf{X}_0 \mathbf{R} = \mathbf{A}^T \mathbf{E} + \mathbf{E}^T \mathbf{A} + \mathbf{E}^T \mathbf{E},$$

which, since $\mathbf{R}^T \mathbf{R} = \mathbf{A}^T \mathbf{A}$, is equivalent to

$$(\mathbf{R} + \mathbf{X}_0 \mathbf{R})^T (\mathbf{R} + \mathbf{X}_0 \mathbf{R}) = (\mathbf{A} + \mathbf{E})^T (\mathbf{A} + \mathbf{E}).$$

Clearly, the product $(\mathbf{I} + \mathbf{X}_0) \mathbf{R}$ is invertible. Hence, there exists a matrix $\mathbf{W} \in \mathbb{R}^{n \times n}$ such that $(\mathbf{A} + \mathbf{E})(\mathbf{R} + \mathbf{X}_0 \mathbf{R})^{-1} = (\mathbf{Q} + \mathbf{W})$. Moreover, we have

$$(\mathbf{Q} + \mathbf{W})^T (\mathbf{Q} + \mathbf{W}) = ((\mathbf{R} + \mathbf{X}_0 \mathbf{R})^{-1})^T (\mathbf{A} + \mathbf{E})^T (\mathbf{A} + \mathbf{E}) (\mathbf{R} + \mathbf{X}_0 \mathbf{R})^{-1} = \mathbf{I}.$$

Let $\mathbf{F} = \mathbf{X}_0 \mathbf{R}$. Then, as a product of two upper triangular matrices, \mathbf{F} is an upper triangular matrix too, and thus $\mathbf{A}' = \mathbf{A} + \mathbf{E} = (\mathbf{Q} + \mathbf{W})(\mathbf{R} + \mathbf{F})$ is a QR factorization of $\mathbf{A} + \mathbf{E} = \mathbf{A}'$. Especially, \mathbf{A}' is invertible and has rank n .

Next, we have to find upper bounds on $\|W\|_f$ and $\|F\|_f$. Since $R + F$ is nonsingular we have $W = (QR + E - Q(R + F))(R + F)^{-1}$, and therefore,

$$W = E(R + F)^{-1} - QX_0(I + X_0)^{-1}. \quad (4.12)$$

The frobenius norm is invariant under orthogonal transformations, thus from (4.12) and (4.11) we obtain

$$\|W\|_f \leq \|E\|_f \|(R + F)^{-1}\|_f + \frac{\|X_0\|_f}{1 - \|X_0\|_f}. \quad (4.13)$$

It remains to find a bound of $\|(R + F)^{-1}\|_f$. Since $Q + W$ is orthogonal, $A + E$ and $R + F$ have the same singular values (see [26]). Let $\sigma(A)$ denote the smallest singular value of A . Then by [26, equation (5.3.14)] we have $\sigma(A)^{-1} = \|A^{-1}\|_2 \leq \|A^{-1}\|_f$ and $\sigma(A + E) \geq \sigma(A) - \|E\|_2$. Therefore, we obtain

$$\begin{aligned} \|(R + F)^{-1}\|_f &\leq n \|(R + F)^{-1}\|_2 = \frac{n}{\sigma(R + F)} \\ &\leq \frac{n}{\sigma(A) - \|E\|_2} = \frac{n \sigma(A)^{-1}}{1 - \sigma(A)^{-1} \|E\|_2} \leq \frac{n \|A^{-1}\|_f}{1 - \|A^{-1}\|_f \|E\|_f}. \end{aligned} \quad (4.14)$$

Finally, applying (4.9), (4.13), and (4.14), we have

$$\|W\|_f \leq \frac{n \|A^{-1}\|_f \|E\|_f}{1 - \|A^{-1}\|_f \|E\|_f} + \frac{\sqrt{2} \|A^{-1}\|_f \|E\|_f + \frac{1}{\sqrt{2}} (\|A^{-1}\|_f \|E\|_f)^2}{1 - \left(\sqrt{2} \|A^{-1}\|_f \|E\|_f + \frac{1}{\sqrt{2}} (\|A^{-1}\|_f \|E\|_f)^2\right)}. \quad (4.15)$$

To estimate $\|F\|_f$ we note that $F = X_0 R$. Therefore, from (4.9) and Proposition 4.2.3 it follows that

$$\|F\|_f \leq \left(\sqrt{2} \|A^{-1}\|_f \|E\|_f + \frac{1}{\sqrt{2}} (\|A^{-1}\|_f \|E\|_f)^2\right) \|A\|_f. \quad (4.16)$$

Combining (4.10) and (4.15) we obtain

$$\|W\|_f \leq \frac{n \left(\frac{c}{n+6}\right) \frac{\sqrt{2}}{3}}{1 - \left(\frac{c}{n+6}\right) \frac{\sqrt{2}}{3}} + \frac{\sqrt{2} \left(\frac{c}{n+6}\right) \frac{\sqrt{2}}{3} + \frac{1}{\sqrt{2}} \left(\left(\frac{c}{n+6}\right) \frac{\sqrt{2}}{3}\right)^2}{1 - \left(\sqrt{2} \left(\frac{c}{n+6}\right) \frac{\sqrt{2}}{3} + \frac{1}{\sqrt{2}} \left(\left(\frac{c}{n+6}\right) \frac{\sqrt{2}}{3}\right)^2\right)} \leq \frac{nc}{n+6} + \frac{6c}{n+6} = c,$$

and analogously by (4.16)

$$\|F\|_f \leq c \min \left\{ 1, \frac{n+6}{\|A\|_f} \right\} \frac{\|A\|_f}{n+6} = c. \quad (4.17)$$

This proves the last assertion of the theorem. \square

Finally, we just want to remark that by a more careful analysis the bound on $\|F\|_f$ given in (4.17) could be improved. But in our applications of Theorem 4.2.4 this effect would be lost because of the use of the O -notation.

As announced above, we next are interested in the quality of approximations of solutions to linear equations and related problems. We start with

Lemma 4.2.5 *Let $A \subseteq \mathbb{R}^{n \times n}$ be a matrix of rank n and A' be a q -approximation to A . Let $\mathbf{t} \in \mathbb{R}^n$, and let $c \in \mathbb{R}$, $0 < c < 1$. If*

$$q \geq \lceil \log \|\mathbf{t}\|_2 - \log(c) - \log(\|\mathbf{A}\mathbf{t}\|_2) + \log(n) \rceil - 1$$

then

$$\|\mathbf{A}\mathbf{t} - \mathbf{A}'\mathbf{t}\|_2 < c \|\mathbf{A}\mathbf{t}\|_2 .$$

Proof. For any matrix $B \in \mathbb{R}^{n \times n}$ we have $\|B\mathbf{t}\|_2 \leq \|B\|_f \|\mathbf{t}\|_2$. Thus, we obtain

$$\|\mathbf{A}\mathbf{t} - \mathbf{A}'\mathbf{t}\|_2 = \|(A - A')\mathbf{t}\|_2 \leq \|A - A'\|_f \|\mathbf{t}\|_2 \leq n2^{-q-1}\mathbf{t} \leq c \|\mathbf{A}\mathbf{t}\|_2 ,$$

where the second inequality follows from (4.3). \square

Corollary 4.2.6 *Let $A = [\mathbf{a}_1, \dots, \mathbf{a}_n] \subseteq \mathbb{R}^{n \times n}$ be a matrix of rank n and $A' = [\mathbf{a}'_1, \dots, \mathbf{a}'_n]$ be a q -approximation to A . Let $c \in \mathbb{R}$, $0 < c < 1$. If*

$$q \geq \lceil -\log(c) - \log(\lambda(A)) + \log(n) \rceil - 1$$

then we have for $1 \leq i \leq n$

$$\|\mathbf{a}_i - \mathbf{a}'_i\|_2 < c \|\mathbf{a}'_i\|_2 .$$

Proof. We set $i \in \mathbb{N}$, $1 \leq i \leq n$. Then we apply Lemma 4.2.5 where we choose \mathbf{t} to be the i -th unit vector in \mathbb{R}^n . Hence, the assertion follows. \square

In order to describe the influence of using approximations instead of the original matrices we quote the following result that is proven in [43, Sect. 2.3].

Lemma 4.2.7 *Let $A \in \mathbb{R}^{n \times n}$ and let A' be a q -approximation to A . Then we have*

$$|\det(A') - \det(A)| \leq \text{dft}(A) |\det(A)| \left(\left(1 + 2^{-q} \frac{\sqrt{n}}{\lambda_1(A)} \right)^n - 1 \right) .$$

We also use an other formulation of this result proven in [25].

Lemma 4.2.8 *Let $A \in \mathbb{R}^{n \times n}$ and let A' be a q -approximation to A . Then we have*

$$|\det(A) - \det(A')| \leq 2^{-q} \sqrt{n} 2^{n-1} n^{\frac{n+1}{2}} \|A\|_\infty^{n-1} .$$

If we restrict the parameter q we can give relative estimations, which, in a second step, make it possible to guarantee that the approximation of sufficiently large precision to a matrix of full rank has full rank too. We start with

Corollary 4.2.9 *Let $c \in \mathbb{R}$, $0 < c < 1$. Let $A \in \mathbb{R}^{n \times n}$ and let A' be a q -approximation to A . If*

$$q > \frac{3 \log(n)}{2} + \log \left(\frac{\text{dft}(A)}{\lambda(A)} \right) - \log(c) + 1$$

then we have

$$|\det(A') - \det(A)| \leq c |\det(A)| .$$

Proof. We set $c' = c/\text{dft}(\mathbf{A})$. Clearly, if

$$q > \frac{\log(n)}{2} - \log(\lambda(\mathbf{A})) - \log\left((c' + 1)^{\frac{1}{n}} - 1\right), \quad (4.18)$$

then we have

$$\left(1 + 2^{-q} \frac{\sqrt{n}}{\lambda(\mathbf{A})}\right)^n - 1 \leq c',$$

and by Lemma 4.2.7

$$|\det(\mathbf{A}') - \det(\mathbf{A})| \leq c |\det(\mathbf{A})|.$$

Thus, to prove the corollary, we have only to estimate $-\log((c' + 1)^{1/n} - 1)$.

We first observe that by (3.9) we have $0 < c' \leq c < 1$, and therefore

$$\left(\frac{n-1}{n}\right) \log(c' + 1) \leq 1.$$

Since we also know that

$$(c' + 1)^{\frac{1}{n}} - 1 = \frac{c'}{\sum_{i=0}^{n-1} (c' + 1)^{\frac{i}{n}}} \geq \frac{c'}{n(c' + 1)^{\frac{n-1}{n}}},$$

this implies

$$\begin{aligned} -\log\left((c' + 1)^{\frac{1}{n}} - 1\right) &\leq -\log(c') + \log(n) + \left(\frac{n-1}{n}\right) \log(c' + 1) \\ &\leq -\log(c') + \log(n) + 1. \end{aligned} \quad (4.19)$$

From (4.18) and (4.19) the assertion follows. \square

Corollary 4.2.10 *Let $\mathbf{A} \subseteq \mathbb{R}^{n \times n}$ be a matrix of rank n and \mathbf{A}' be a q -approximation to \mathbf{A} . If*

$$q > \frac{3 \log(n)}{2} + \log\left(\frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})}\right) + 2$$

then \mathbf{A}' has rank n .

Proof. From Corollary 4.2.9 it follows that $|\det(\mathbf{A}') - \det(\mathbf{A})| \leq (1/2)|\det(\mathbf{A})|$. Thus, we have $|\det(\mathbf{A})| - |\det(\mathbf{A}')| \leq |\det(\mathbf{A}') - \det(\mathbf{A})| \leq (1/2)|\det(\mathbf{A})|$. Therefore $|\det(\mathbf{A}')| \neq 0$, which implies the assertion. \square

Next, we consider matrices as bases of certain lattices, and examine properties of those lattices whose bases are approximations of bases of other lattices.

Lemma 4.2.11 *Let $\Lambda \subseteq \mathbb{R}^n$ be a n -dimensional lattice in \mathbb{R}^n with basis $\mathbf{A} \subseteq \mathbb{R}^{n \times n}$, and let $\Lambda' \subseteq \mathbb{R}^n$ be a n -dimensional lattice in \mathbb{R}^n with basis $\mathbf{A}' \subseteq \mathbb{Q}^{n \times n}$, where \mathbf{A}' is a q -approximation to \mathbf{A} . Let $c \in \mathbb{R}$, $0 < c < 1$. If*

$$q > \frac{3 \log(n)}{2} + \log\left(\frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})}\right) - \log(c) \quad (4.20)$$

then for $1 \leq i \leq n$ we have

$$\lambda_i(\Lambda') \leq (1+c)\lambda_i(\Lambda).$$

Proof. Let $i \in \mathbb{N}$, $1 \leq i \leq n$, and choose $\mathbf{t} \in \mathbb{Z}^n$ such that $\|\mathbf{A}\mathbf{t}\|_2 = \lambda_i(\Lambda)$. From Lemma 4.2.5 it follows that for

$$q > \log \|\mathbf{t}\|_2 - \log(c) - \log(\lambda_i(\Lambda)) + \log(n) \quad (4.21)$$

we have

$$\|\mathbf{A}'\mathbf{t}\|_2 - \|\mathbf{A}\mathbf{t}\|_2 \leq \|\mathbf{A}\mathbf{t} - \mathbf{A}'\mathbf{t}\|_2 \leq c \|\mathbf{A}\mathbf{t}\|_2,$$

and therefore

$$\lambda_i(\Lambda') \leq \|\mathbf{A}'\mathbf{t}\|_2 \leq (1+c)\|\mathbf{A}\mathbf{t}\|_2 = (1+c)\lambda_i(\Lambda).$$

To conclude the proof we only have to estimate $\|\mathbf{t}\|_2$. Applying Corollary 3.5.18 and Lemma 3.5.12 we obtain that

$$\|\mathbf{t}\|_2 \leq \sqrt{n} \lambda_i(\Lambda) \frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})}. \quad (4.22)$$

Thus the assertion follows from (4.21) and (4.22). \square

Especially, we are interested in the dependence between “short” vectors in lattices such that the basis of one of them is an approximation of the basis of the other. We have

Lemma 4.2.12 *Let $\Lambda \subseteq \mathbb{R}^n$ be a n -dimensional lattice in \mathbb{R}^n with basis $\mathbf{A} \subseteq \mathbb{R}^{n \times n}$, and let $\Lambda' \subseteq \mathbb{R}^n$ be a n -dimensional lattice in \mathbb{R}^n with basis $\mathbf{A}' \subseteq \mathbb{Q}^{n \times n}$, where \mathbf{A}' is a q -approximation to \mathbf{A} . Let $c \in \mathbb{R}$, $0 < c < 1$. Finally, let $i \in \mathbb{N}$, $1 \leq i \leq n$, let $s \in \mathbb{R}$, $s \geq 1$, and let $\mathbf{t} \in \mathbb{Z}^n$ such that $\|\mathbf{A}'\mathbf{t}\|_2 \leq s\lambda_i(\Lambda')$. If*

$$q > n \log \left(\frac{3}{2} \right) + \frac{3 \log(n)}{2} + \log \left(\frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})} \right) + \log(s) - \log(c) + 3 \quad (4.23)$$

then

$$\|\mathbf{A}\mathbf{t}\|_2 \leq s(1+c)\lambda_i(\Lambda).$$

Proof. Lemma 4.2.5 and Lemma 4.2.11 imply that for

$$q > \max \left\{ \log \|\mathbf{t}\|_2 - \log \left(\frac{c}{3} \right) - \log(\lambda_i(\Lambda)) + \log(n), \frac{3 \log(n)}{2} + \log \left(\frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})} \right) - \log(c) \right\} \quad (4.24)$$

we have

$$\|\mathbf{A}\mathbf{t}\|_2 \leq \left(\frac{1}{1-\frac{c}{3}} \right) \|\mathbf{A}'\mathbf{t}\|_2 \leq \left(\frac{s}{1-\frac{c}{3}} \right) s \lambda_i(\Lambda')$$

and

$$\lambda_i(\Lambda') \leq (1+c)\lambda_i(\Lambda), \quad (4.25)$$

and therefore

$$\|\mathbf{A}\mathbf{t}\|_2 \leq s(1+c)\lambda_i(\Lambda).$$

To prove the lemma we have to estimate $\|\mathbf{t}\|_2$. By Lemma 3.5.12 and Corollary 3.5.18 and by (4.25) we have

$$\|\mathbf{t}\|_2 \leq \sqrt{n} s \lambda_i(\Lambda') \frac{\text{dft}(\mathbf{A}')}{\lambda(\mathbf{A}')} \leq \sqrt{n} s (c+1) \lambda_i(\Lambda) \frac{\text{dft}(\mathbf{A}')}{\lambda(\mathbf{A}')}. \quad (4.26)$$

Thus it suffices to prove an upper bound of $\text{dft}(\mathbf{A}')/\lambda(\mathbf{A}')$. Let $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$, and let $\mathbf{A}' = [\mathbf{a}'_1, \dots, \mathbf{a}'_n]$. By Corollary 4.2.9 we know that for q satisfying (4.24) we have

$$\frac{1}{|\det(\mathbf{A}')|} \leq \frac{3}{2|\det(\mathbf{A})|}, \quad (4.27)$$

and by Corollary 4.2.6 we also have for $1 \leq i \leq n$

$$\|\mathbf{a}'_i\|_2 \leq \frac{4}{3} \|\mathbf{a}_i\|_2. \quad (4.28)$$

Thus from (4.27) and (4.28) it follows that

$$\frac{\text{dft}(\mathbf{A}')}{\lambda(\mathbf{A}')} \leq \left(\frac{3}{2}\right)^n \frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})}. \quad (4.29)$$

Combining (4.24), (4.26), and (4.29) concludes the proof. \square

4.3 Approximating Conjugates

In our work, the main motivation for studying approximations, is the necessity to compute with vectors of the Minkowski lattice of an ideal or the unit lattice of an order. In both cases, we have to compute the conjugates of algebraic numbers, or to be more precise, to compute approximations of sufficiently large precision to them.

In what follows let \mathbb{F} be an algebraic number field of degree n and signature (s, t) . We let $m = s + t$ and $r = s + t - 1$. Furthermore, let \mathcal{O} be an order of \mathbb{F} with \mathbb{Z} -basis $\Omega = (\omega_1, \omega_2, \dots, \omega_n)$ and assume that \mathcal{O} (and \mathbb{F}) is given by a reduced multiplication table $\text{MT}(\Omega)$ of \mathcal{O} (see section 3.2).

Here, we shall not explain in full detail how to approximate conjugates of an algebraic number $\alpha \in \mathbb{F}$. We will only give a crude sketch of an appropriate algorithm. For more details, we refer for example to [43, Sect. 4.1] or [25, Sect. 5]. First, the algorithm computes the standard representation (with respect to the given basis $\omega_1, \dots, \omega_n$ of \mathcal{O}) of a primitive element ρ of \mathbb{F} . Then it determines the standard representations of the powers ρ^i for $1 \leq i \leq n - 1$. This means, it computes a matrix $A = (a_{i,j}) \in \mathbb{Z}^{n \times n}$ such that $\rho^i = \sum_{j=1}^n a_{i,j} \omega_j$ for $0 \leq i \leq n - 1$. It computes the minimal polynomial f of ρ and by means of the methods of Schönhage (cf. [50]) approximations to the n roots $\rho^{(1)}, \dots, \rho^{(n)}$ of f . From the powers $\rho^{(j)i-1}$ and from A^{-1} the algorithm can find approximations to $\omega_i^{(j)}$ and then to $\alpha^{(j)}$ for $1 \leq i, j \leq n$.

Theorem 4.3.1 *There is an algorithm that on input of an order \mathcal{O} of \mathbb{F} given by a multiplication table $\text{MT}(\Omega)$ of a \mathbb{Z} -basis Ω of \mathcal{O} , a nonzero element $\alpha \in \mathbb{F}$, and a natural number q determines a q -approximation to the conjugate vector $\underline{\alpha}$ in time*

$$O\left(n^8 (q + \text{size}(\alpha) + (\log(n \|\text{MT}(\Omega)\|_\infty))^2) (\log(q + \text{size}(\alpha) + \log \|\text{MT}(\Omega)\|_\infty))^2\right).$$

Proof. Let $\Omega = (\omega_1, \omega_2, \dots, \omega_n)$. Then from [43, Theorem 4.1.10] it follows that there is an algorithm that for $p \in \mathbb{N}$ computes p -approximations $\underline{\omega}_1', \dots, \underline{\omega}_n'$ to the conjugate vectors $\underline{\omega}_1', \dots, \underline{\omega}_n'$ in time

$$O\left((n^3 p + n^7 (\log(n \|\text{MT}(\Omega)\|_\infty))^2) (\log(n(p + \log \|\text{MT}(\Omega)\|_\infty)))^2\right).$$

Now, let $\alpha = (1/a_{n+1}) \sum_{i=1}^n a_i \omega_i$, where $a_i \in \mathbb{Z}$ for $1 \leq i \leq n+1$. We set $p = q + \text{size}(\alpha) + \lceil \log(n) \rceil$. Then applying (a) and (c) of Proposition 4.1.8 we see that $\underline{\alpha}' = (1/a_{n+1}) \sum_{i=1}^n a_i \underline{\omega}_i'$ is an approximation of precision $p - \lceil \log(\max\{|a_1|, \dots, |a_{n+1}|\}) \rceil - \lceil \log(n) \rceil \geq p - \text{size}(\alpha) - \lceil \log(n) \rceil = q$ to $\underline{\alpha}$. From Lemma 3.5.23 it follows that given the p -approximations $\underline{\omega}_1', \dots, \underline{\omega}_n'$ we can compute $\underline{\alpha}'$ in time

$$O(n(p + \log(n^2 \|\text{MT}(\Omega)\|_\infty 2^{2n})) \text{size}(\alpha)).$$

Since $\text{size}(\alpha) > \log(n)$ the assertion follows from the properties of the O-notation and some very crude estimations. \square

Using the above theorem, we can therefore approximate the conjugates of an arbitrary element of \mathbb{F} . We also have to describe how to approximate archimedean evaluations and logarithms of them.

Theorem 4.3.2 *There is an algorithm that given an order \mathcal{O} of \mathbb{F} by a multiplication table $\text{MT}(\Omega)$ of a \mathbb{Z} -basis Ω , a nonzero element $\alpha \in \mathbb{F}$, and a natural number q determines a q -approximation to the logarithm vector $\text{Log } \alpha$ in time*

$$O\left(n^8 (q + \text{size}(\alpha) + (\log(n \|\text{MT}(\Omega)\|_\infty))^3) (\log(q + \text{size}(\alpha) + \log \|\text{MT}(\Omega)\|_\infty))^2\right).$$

Proof. Using the algorithm of Theorem 4.3.1 we compute an approximation \mathbf{a} of precision

$$p \geq \lceil -\log(\max\{|\alpha^{(1)}|, \dots, |\alpha^{(m)}|\}) \rceil + q + 3 \quad (4.30)$$

to $\underline{\alpha}$. Then from \mathbf{a} we immediately obtain p -approximations \mathbf{a}_i of the conjugates $\alpha^{(i)}$ ($1 \leq i \leq m$). By (4.1) we know that $|\mathbf{a}_i|$ is an approximation to $|\alpha^{(i)}|$ of precision $p-1$, thus Lemma 4.2.2 implies that $\ln |\mathbf{a}_i|$ is an approximation to $\ln |\alpha^{(i)}|$ of precision $q+2$.

From [51] it follows that on input of a positive rational number $z' = a2^{-p}$, where $a, p \in \mathbb{N}$, we can compute an approximation of precision p to $\ln(z')$ in time $O(\log(a) + p^2 \log(p))$. Thus, given $|\mathbf{a}_i|$ we can compute an approximation to $\ln |\alpha_i|$ of precision $q+2$, which by part (a) and (b) of Proposition 4.1.8 is a q -approximation to $\ln |\alpha_i|$, in time

$$O(p \log(|\alpha_i|) + p^2 \log(p)) = O(p \log(1 + |\alpha^{(i)}|) + p^2 \log(p)),$$

or applying Lemma 3.5.23,

$$O(p(2\log(n) + \log \|\text{MT}(\Omega)\|_\infty + \text{size}(\alpha)) + p^2 \log(p)) . \quad (4.31)$$

The question we have to answer in order to prove the theorem is how p depends on α . Let $\alpha = (1/b_{n+1}) \sum_{i=1}^n b_i \omega_i$, where $b_i \in \mathbb{Z}$ for $1 \leq i \leq n+1$. Then $b_{n+1}\alpha \in \mathcal{O}$, and by Proposition 3.5.22 we have

$$\|\alpha\|_2 \geq \frac{1}{b_{n+1}} \sqrt{\frac{n}{2}} .$$

Hence, there exists $i \in \mathbb{N}$, $1 \leq i \leq m$, such that

$$|\alpha^{(i)}| \geq \frac{1}{b_{n+1}} \sqrt{\frac{1}{2n}} \geq \frac{1}{2^{\text{size}(\alpha)}} \sqrt{\frac{1}{2n}} . \quad (4.32)$$

Combining (4.30) and (4.32) we obtain that it is sufficient to choose p such that

$$p > \text{size}(\alpha) + \frac{1}{2} \log(n) + q + 4 .$$

Thus, from Theorem 4.3.1, Lemma 3.5.23 and (4.31) the assertion follows. \square

Theorem 4.3.3 *There is an algorithm that given an order \mathcal{O} of \mathbb{F} by a multiplication table $\text{MT}(\Omega)$ of a \mathbb{Z} -basis Ω , a nonzero element $\alpha \in \mathbb{F}$, and a natural number q determines a q -approximation to the valuations $|\alpha|_i$ ($1 \leq i \leq m$) in time*

$$O\left(n^8 (q + \text{size}(\alpha) + (\log(n \|\text{MT}(\Omega)\|_\infty))^3) (\log(q + \text{size}(\alpha) + \log \|\text{MT}(\Omega)\|_\infty))^2\right) .$$

Proof. As in the proof of Theorem 4.3.2 we start by computing p -approximation \mathbf{a}_i to the conjugates $\alpha^{(i)}$ ($1 \leq i \leq m$) by means of the algorithm of Theorem 4.3.1, where we set

$$p > \log(|\alpha^{(i)}| + 1) + q + 2 . \quad (4.33)$$

For $1 \leq i \leq s$ the number $|\mathbf{a}_i|$ is a p -approximation to $|\alpha^{(i)}|$. For $s+1 \leq i \leq m$ we have $|\alpha|_i = |\alpha^{(i)}|^2$, and

$$\begin{aligned} \left| |\mathbf{a}_i|^2 - |\alpha^{(i)}|^2 \right| &= \left| |\mathbf{a}_i| - |\alpha^{(i)}| \right| \left| |\mathbf{a}_i| + |\alpha^{(i)}| \right| \\ &\leq 2^{-p} \left(2|\alpha^{(i)}| + 2^{-p} \right) \leq 2^{-p+1} \left(|\alpha^{(i)}| + 1 \right) \leq 2^{-q-1} . \end{aligned}$$

Thus, the assertion follows from Theorem 4.3.1 and Lemma 3.5.23. \square

Finally, we use approximations of algebraic integers to compare their archimedean valuations.

Theorem 4.3.4 *Let \mathfrak{A} be an ideal of \mathcal{O} and let α and β be elements of \mathfrak{A} . Let \mathbf{a} and \mathbf{b} be q -approximations to $\underline{\alpha}$ and $\underline{\beta}$. If*

$$q > \log(d(\mathfrak{A})) + \log\left(\max\left\{\mathbf{H}(\alpha), \sqrt{\mathbf{H}(\alpha)}, \mathbf{H}(\beta), \sqrt{\mathbf{H}(\beta)}\right\} + 1\right) + p + 3,$$

where $p \in \mathbb{N}$ with

$$p > n^2 \log\left(4(d(\mathfrak{A}))^2 \max\left\{\mathbf{H}(\alpha), \sqrt{\mathbf{H}(\alpha)}, \mathbf{H}(\beta), \sqrt{\mathbf{H}(\alpha)}, 1\right\}\right) + 2$$

then we have $|\alpha|_i - |\beta|_i = 0$ if $|\mathbf{a}|_i - |\mathbf{b}|_i \leq 2^{-p}$, $|\alpha|_i - |\beta|_i > 0$ if $|\mathbf{a}|_i - |\mathbf{b}|_i > 2^{-p}$, and $|\alpha|_i - |\beta|_i < 0$ if $|\mathbf{a}|_i - |\mathbf{b}|_i < -2^{-p}$.

Proof. For convenience, we set $\alpha' = d(\mathfrak{A})\alpha$ and $\beta' = d(\mathfrak{A})\beta$. Let ρ be a primitive element of \mathbb{F} , i.e. $\mathbb{F} = \mathbb{Q}(\rho)$. Let $i \in \mathbb{N}$, $1 \leq i \leq m$, and set $\gamma = |\alpha'|_i - |\beta'|_i$. Then γ is an algebraic integer that is an element of an algebraic number field K . Let k be the degree of K . We have to distinguish two cases: If $1 \leq i \leq s$ then we can take $K = \mathbb{Q}(\rho^{(i)})$ and $k = n$. For $s + 1 \leq i \leq m$ we have $\gamma = \alpha'^{(i)}\overline{\alpha'^{(i)}} - \beta'^{(i)}\overline{\beta'^{(i)}} \in K$, where $K = \mathbb{Q}(\rho^{(i)})\mathbb{Q}(\overline{\rho^{(i)}})$ is the composite of $\mathbb{Q}(\rho^{(i)})$ and $\mathbb{Q}(\overline{\rho^{(i)}})$. Then we have $k = n\ell$ with

$$\ell = [K : \mathbb{Q}(\rho^{(i)})] = [K : \mathbb{Q}(\overline{\rho^{(i)}})] \leq n - 1$$

(see for example [24, Page 124]).

Clearly, since α' and β' are algebraic integers the number γ is an algebraic integer too. Hence, by Lemma 3.2.7 we have

$$|\mathbf{N}_{K/\mathbb{Q}}(\gamma)| \geq 1, \quad (4.34)$$

provided that $\gamma \neq 0$. We shall use (4.34) to find a lower bound on γ . Let κ be a positive rational number with

$$\kappa \geq 2 \max\{\alpha'^{(1)}, \dots, \alpha'^{(m)}, \beta'^{(1)}, \dots, \beta'^{(m)}, 1\}. \quad (4.35)$$

Then for $1 \leq j \leq m$ we have

$$\left| |\alpha'|_j - |\beta'|_j \right| \leq \kappa. \quad (4.36)$$

If $1 \leq i \leq s$ then (4.34) and (4.36) imply that

$$1 < \gamma \kappa^{n-1},$$

and therefore $\gamma > \kappa^{1-n}$.

If $s + 1 \leq i \leq m$ the situation is more complicated. Each monomorphism of $\mathbb{Q}(\rho^{(i)})$ or $\mathbb{Q}(\overline{\rho^{(i)}})$ into \mathbb{C} has ℓ distinct extensions to a \mathbb{Q} -monomorphism of K into \mathbb{C} . Let $\tau_1, \tau_2, \dots, \tau_{\ell n}$ be the monomorphism of K into \mathbb{C} and assume that τ_1 is the identity and τ_2 is the complex conjugation. Hence, from (4.34) it follows that

$$1 \leq |\mathbf{N}_{K/\mathbb{Q}}(\gamma)| \leq \gamma^2 \prod_{j=3}^{\ell n} |\tau_j(\gamma)| \leq \gamma^2 \kappa^{2n\ell}.$$

Thus we have $\gamma > \kappa^{-nl} > \kappa^{-n^2}$.

Therefore, in both cases κ^{-n^2} is a lower bound of γ . Clearly, analogously to the proof of Theorem 4.3.3 we see that if \mathbf{a} and \mathbf{b} are approximations to $\underline{\alpha}$ and $\underline{\beta}$ of precision $q \in \mathbb{N}$ with

$$q > \log(d(\mathfrak{A})) + \log(\max\{\alpha^{(1)}, \dots, \alpha^{(m)}, \beta^{(1)}, \dots, \beta^{(m)}\} + 1) + p + 3, \quad (4.37)$$

where $p \in \mathbb{N}$, then $\gamma' = d(\mathfrak{A})|\mathbf{a}|_i - d(\mathfrak{A})|\mathbf{b}|_i$ is an approximation to γ of precision p i.e.

$$|\gamma - \gamma'| < 2^{-p}.$$

Hence, for $p > \log(2\kappa^{n^2})$ we have $\gamma = 0$ if $|\gamma'| \leq 2^{-p}$, $\gamma > 0$ if $\gamma' > 2^{-p}$, and $\gamma < 0$ if $\gamma' < -2^{-p}$. Finally, we note that

$$\max\{\alpha^{(1)}, \dots, \alpha^{(m)}, \beta^{(1)}, \dots, \beta^{(m)}\} \leq (d(\mathfrak{A}))^2 \max\{\mathbf{H}(\alpha), \sqrt{\mathbf{H}(\alpha)}, \mathbf{H}(\beta), \sqrt{\mathbf{H}(\beta)}\}. \quad (4.38)$$

Thus, from (4.37) and (4.35) the assertion follows. \square

Applying Lemma 3.5.23 we immediately obtain

Corollary 4.3.5 *Let \mathfrak{A} be an ideal of \mathcal{O} and let α and β be elements of \mathfrak{A} . Let \mathbf{a} and \mathbf{b} be q -approximations to $\underline{\alpha}$ and $\underline{\beta}$. If*

$$q > \log(d(\mathfrak{A})) + \log\left(n^2 \|\text{MT}(\Omega)\|_\infty 2^{\max\{\text{size}(\alpha), \text{size}(\beta)\}}\right) + p + 3,$$

where $p \in \mathbb{N}$ with

$$p > n^2 \log\left(\left(4n^2 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\alpha) + \text{size}(\beta)} + 1\right)\right) + 2$$

then we have $|\alpha|_i - |\beta|_i = 0$ if $|\mathbf{a}|_i - |\mathbf{b}|_i \leq 2^{-p}$, $|\alpha|_i - |\beta|_i > 0$ if $|\mathbf{a}|_i - |\mathbf{b}|_i > 2^{-p}$, and $|\alpha|_i - |\beta|_i < 0$ if $|\mathbf{a}|_i - |\mathbf{b}|_i < -2^{-p}$.

Chapter 5

Minima and Reduced Ideals

5.1 Definitions and Properties

The main tool used in our work is the theory of minima and reduced ideals in algebraic number fields as presented in [4], [5] and [6]. In this section, we give a short overview of this theory.

In the following sections, let \mathbb{F} be a number field of degree n and of signature (s, t) , and set $m = s + t$ and $r = s + t - 1$. Let \mathcal{O} be an order of \mathbb{F} , and for convenience, let $\Delta = |\Delta_{\mathcal{O}}|$. Furthermore, we assume that \mathcal{O} is given by the multiplication table $\text{MT}(\Omega)$, where $\Omega = (\omega_1, \dots, \omega_n)$ is a \mathbb{Z} -basis of \mathcal{O} .

Definition 5.1.1 A number μ of an ideal \mathfrak{A} of \mathcal{O} is called a *minimum* of \mathfrak{A} if there exists no $\alpha \in \mathfrak{A}$, $\alpha \neq 0$, such that $|\alpha|_i < |\mu|_i$ for all $1 \leq i \leq m$. A minimum μ' of \mathfrak{A} is called a *neighbor* of the minimum μ if there exists no $\alpha \in \mathfrak{A}$, $\alpha \neq 0$, such that $|\alpha|_i < \max\{|\mu|_i, |\mu'|_i\}$ for all $1 \leq i \leq m$.

Definition 5.1.2 An ideal \mathfrak{A} is called *reduced* if 1 is a minimum of \mathfrak{A} .

By simple observations we obtain now

Proposition 5.1.3 *Let \mathfrak{A} be an ideal. Then the reduced ideals equivalent to \mathfrak{A} are exactly the ideals $(1/\mu)\mathfrak{A}$, where μ is a minimum of \mathfrak{A} . Furthermore, let $\xi \in \mathbb{F} - \{0\}$. Then μ is a minimum of \mathfrak{A} if and only if $\xi\mu$ is a minimum of the ideal $\xi\mathfrak{A}$, and the minimum μ' is a neighbor of μ , if and only if $\xi\mu'$ is a neighbor of $\xi\mu$.*

Thus, every ideal class of the class group contains reduced ideals.

Now, we can explain the second discrete logarithm problem, namely the *discrete logarithm problem of an order*, i.e. to determine for a given reduced principal ideal of the order and $p \in \mathbb{N}$ a p -approximation to the logarithm vector of a generator of the ideal.

In [4] and [5] the author describes many properties of minima and reduced ideals in great detail. In the following we shall quote those results that we need in our thesis. We shall only sketch the proof of the results, if we can improve or simplify them. We use

the following notation: We define for any $\mathbf{v} = (\mathbf{v}_1, \dots, \mathbf{v}_n)^T \in \mathbb{R}^n$ and for $i \in \{1, \dots, m\}$

$$|\mathbf{v}|_i = \begin{cases} |\mathbf{v}_i| & \text{if } 1 \leq i \leq s, \\ \mathbf{v}_i^2 + \mathbf{v}_{i+t}^2 & \text{if } s+1 \leq i \leq m. \end{cases}$$

(Clearly, the notation could be confused with the archimedean valuations. But this abuse will always be clarified by the context.)

Definition 5.1.4 The *norm body* of a finite set $A = \{\alpha_1, \dots, \alpha_\ell\} \subset \mathbb{F}$ ($\ell \in \mathbb{N}$) is defined to be

$$\mathcal{N}(A) = \{\mathbf{x}: \mathbf{x} \in \mathbb{R}^n, |\mathbf{x}|_i < \max\{|\alpha_1|_i, \dots, |\alpha_\ell|_i\} \text{ for } 1 \leq i \leq m\}.$$

As a consequence of this definition we have

Proposition 5.1.5 *An element μ of an ideal \mathfrak{A} is a minimum of \mathfrak{A} if and only if $\mathcal{N}(\{\mu\})$ contains no nonzero vector of the Minkowski lattice \mathfrak{A} . A minimum μ' of \mathfrak{A} is a neighbor of the minimum μ if and only if $\mathcal{N}(\{\mu, \mu'\})$ contains no nonzero vector of the Minkowski lattice \mathfrak{A} .*

Lemma 5.1.6 *Let $\Lambda \subseteq \mathbb{R}^n$ be a n -dimensional lattice in \mathbb{R}^n , and let c_1, \dots, c_m be positive real numbers. Then the volume of the set $X = \{\mathbf{x}: \mathbf{x} \in \mathbb{R}^n, |\mathbf{x}|_i < c_i\}$ is*

$$\text{vol}(X) = 2^s \pi^t \prod_{i=1}^m c_i.$$

Proof. See for example [57, Lemma 9.2]. □

Corollary 5.1.7 *Let $\alpha \in \mathbb{F}$. Then the volume of the norm body $\mathcal{N}(\alpha)$ is*

$$\text{vol}(\mathcal{N}(\{\alpha\})) = 2^s \pi^t |N_{\mathbb{F}/\mathbb{Q}}(\alpha)|.$$

From Theorem 3.5.4, Proposition 3.5.21, Proposition 5.1.5 and Corollary 5.1.7 we obtain

Proposition 5.1.8 *Let \mathfrak{A} be an ideal of \mathcal{O} and let μ be a minimum of \mathfrak{A} . Then we have*

$$|N_{\mathbb{F}/\mathbb{Q}}(\mu)| \leq \left(\frac{2}{\pi}\right)^t N_{\mathcal{O}}(\mathfrak{A}) \sqrt{\Delta}.$$

Corollary 5.1.9 *Let \mathfrak{A} be a reduced ideal of an order \mathcal{O} , and let $\mu \in \mathfrak{A}$ be a neighbor of the minimum 1. Then we have*

$$H(\mu) \leq \sqrt{\Delta}.$$

Proof. As a consequence of Theorem 3.5.7, Proposition 5.1.5, and Lemma 5.1.6 we obtain that

$$\prod_{i=1}^m \max\{|\mu|_i, 1\} \leq 2^t \det(\mathfrak{A}).$$

Now, Proposition 3.5.21, Proposition 3.4.11, and Corollary 5.1.10 imply that for $1 \leq i \leq m$ we have

$$|\mu|_i \leq \max\{|\mu|_i, 1\} \leq N_{\mathcal{O}}(\mathfrak{A})\sqrt{\Delta} \leq \sqrt{\Delta}.$$

This concludes the proof. \square

Corollary 5.1.10 *If \mathfrak{A} is a reduced ideal of \mathcal{O} , then $1/\sqrt{\Delta} \leq N(\mathfrak{A}) \leq 1$.*

Proof. Since 1 is a minimum in \mathfrak{A} we know by Lemma 3.4.15 that $1 = |N_{\mathbb{F}/\mathbb{Q}}(1)| \geq N_{\mathcal{O}}(\mathfrak{A})$. On the other hand we get by Theorem 5.1.8 that $1 = |N_{\mathbb{F}/\mathbb{Q}}(1)| \leq N_{\mathcal{O}}(\mathfrak{A})\sqrt{\Delta}$, which proves the assertion. \square

One feature of reduced ideals is that their standard representations are of “small” binary size. To estimate this binary size we need the following result proved in [6]:

Lemma 5.1.11 *Let \mathfrak{A} be a reduced ideal of \mathcal{O} and let $(d(\mathfrak{A}), \mathbf{A})$ be the standard representation of \mathfrak{A} . Suppose that $\mathbf{A} = (a_{i,j}) \in \mathbb{R}^{n \times n}$. Then we have*

$$0 \leq |a_{i,j}| \leq d(\mathfrak{A}) \leq \sqrt{\Delta} \quad \text{for } 1 \leq i \leq n, 1 \leq j \leq n.$$

The above lemma immediately implies

Corollary 5.1.12 *Let \mathfrak{A} be a reduced ideal of \mathcal{O} . Then $\text{size}(\mathfrak{A}) \leq (n^2 + 1) \log(\sqrt{\Delta})$.*

Definition 5.1.13 For $\mathbf{z} \in \mathbb{R}^r$ we define

$$\mathcal{W}(\mathbf{z}) = \left\{ \mathbf{x} : \mathbf{x} \in \mathbb{R}^r, |\mathbf{z}_i - \mathbf{x}_i| \leq \frac{1}{4} \log(\Delta) \text{ for } 1 \leq i \leq r \right\}.$$

By combinatorial arguments (see [5, Sect. 3]) we have

Lemma 5.1.14 *Let \mathfrak{A} be a reduced ideal of \mathcal{O} and let $\mathbf{z} \in \mathbb{R}^r$. Then the number N of minima $\mu \in \mathfrak{A}$ with $\text{Log } \mu \in \mathcal{W}(\mathbf{z})$ satisfies*

$$1 \leq N \leq 4^n (\log(\Delta))^r.$$

Corollary 5.1.15 *Let \mathfrak{A} be a reduced ideal of \mathcal{O} , and let $B_1, \dots, B_m \in \mathbb{R}_{>0}$. Then the number of minima μ of \mathfrak{A} with $|\mu|_i \leq B_i$ for $1 \leq i \leq m$ is bounded by*

$$4^n \left(\log(\Delta) + \sum_{j=1}^m |\ln(B_j)| \right)^r.$$

The binary size of each such minimum is bounded by

$$n \log \left(\sqrt{n\Delta} \max\{B_1, \dots, B_n, B_1^2, \dots, B_n^2\} \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \log(\sqrt{\Delta}).$$

Proof. Suppose that μ is a minimum satisfying the assertion of the corollary. Then we have

$$\ln |\mu|_i \leq \ln(B_i)$$

for $1 \leq i \leq m$. By Proposition 3.1.7 and Corollary 5.1.10 we also know that

$$\prod_{i=1}^m |\mu|_i \geq N_{\mathcal{O}}(\mathfrak{A}) \geq \frac{1}{\sqrt{\Delta}}.$$

This implies that for $1 \leq i \leq m$

$$\ln |\mu|_i \geq -\ln(\sqrt{\Delta}) - \sum_{\substack{1 \leq j \leq m \\ j \neq i}} |\ln(B_j)|.$$

Hence, applying Lemma 5.1.14 we see that the number of the minima μ that satisfy our condition is at most

$$\left(\frac{\ln(\sqrt{\Delta}) + \sum_{j=1}^m |\ln(B_j)|}{\log(\sqrt{\Delta})} \right)^r 4^n (\log(\Delta))^r.$$

This implies the first assertion.

By Lemma 5.1.11 we have $d(\mathfrak{A}) \leq \sqrt{\Delta}$. Hence Lemma 3.5.25 implies that the binary size of each minimum μ of \mathfrak{A} with $|\mu|_i \leq B_i$ for $1 \leq i \leq m$ is bounded by

$$n \log \left(\sqrt{\Delta} \sqrt{n} \max\{B_1, \dots, B_n, B_1^2, \dots, B_n^2\} \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \log(\sqrt{\Delta}).$$

This proves the second assertion. \square

Corollary 5.1.16 *The number of neighbors of a minimum in a reduced ideal is bounded by $4^n (m \log(\Delta))^r$.*

Proof. By Proposition 5.1.3 it is sufficient to prove the assertion for the number of neighbors of 1. Since $r = m - 1$ we have $m \geq 2$ for $r \geq 1$. Hence, the assertion follows from Corollary 5.1.9 and Corollary 5.1.15. \square

Using similar methods, in [5] it is also shown

Lemma 5.1.17 *The number of reduced ideals in the equivalence class of an ideal \mathfrak{A} is bounded by $6^n R_{\mathcal{O}}$.*

In fact, the bounds given in [5] are more precise. But for our applications and estimations these sharper bounds would not yield an improvement.

5.2 The Reduction Algorithm

By means of the algorithms for computing approximations we now describe an algorithm that on input of an ideal \mathfrak{A} of \mathcal{O} computes a minimum $\mu \in \mathfrak{A}$ and a reduced ideal \mathfrak{B} with $\mathfrak{B} = (1/\mu)\mathfrak{A}$. The main idea was already presented in [13]. But here we shall use other techniques and give a more precise analysis, that especially considers the dependence of the running time on the degree n of the number field.

The main idea is based on the following results.

Proposition 5.2.1 *Let \mathfrak{A} be an ideal of \mathcal{O} with Minkowski lattice $\underline{\mathfrak{A}}$. Then each $\mu \in \mathfrak{A}$ such that $\underline{\mu}$ is a shortest vector of $\underline{\mathfrak{A}}$ is a minimum of \mathfrak{A} .*

Proof. Suppose that we have $\mu \in \mathfrak{A}$, where $\underline{\mu}$ is a shortest vector of $\underline{\mathfrak{A}}$. If μ is not a minimum of \mathfrak{A} then there exists $\beta \in \mathfrak{A}$, $\beta \neq 0$, such that $|\beta|_i < |\mu|_i$ for $1 \leq i \leq m$. But this implies

$$\|\underline{\beta}\|_2^2 = \sum_{i=1}^s |\beta|_i^2 + \sum_{i=s+1}^m |\beta|_i < \sum_{i=1}^s |\mu|_i^2 + \sum_{i=s+1}^m |\mu|_i = \|\underline{\mu}\|_2^2,$$

which clearly is a contradiction. \square

In our algorithms we can only compute with approximations to vectors of the Minkowski lattice. Thus, we need a stronger version of the above proposition. We present a result that is a refinement of the results of [13].

Lemma 5.2.2 *Let \mathfrak{A} be an ideal of \mathcal{O} with basis $(\alpha_1, \dots, \alpha_n)$. Let $q \in \mathbb{N}$, and let $\Lambda' \subseteq \mathbb{R}^n$ be the lattice with basis $(\mathbf{a}'_1, \dots, \mathbf{a}'_n)$, where \mathbf{a}'_i is a q -approximation to $\underline{\alpha}_i$ for $1 \leq i \leq n$. Finally, let $\mathbf{t} \in \mathbb{Z}^n$, such that $\sum_{i=1}^n \mathbf{t}_i \mathbf{a}'_i$ is a shortest vector of Λ' . If*

$$q > 2 \log(\Delta) + (n+1)^2 (6 \log(n) + \log(n^4 \|\text{MT}(\Omega)\|_\infty) + \text{size}(\mathfrak{A}) + \log(d(\mathfrak{A}) + 2)) \quad (5.1)$$

then $\alpha = \sum_{i=1}^n \mathbf{t}_i \alpha_i$ is a minimum of \mathfrak{A} such that for $1 \leq i \leq m$ we have

$$|\alpha|_i \leq 2\sqrt{n} \left(\sqrt{\Delta} N_{\mathcal{O}}(\mathfrak{A}) \right)^{\frac{1}{n}}. \quad (5.2)$$

Proof. We assume that α is not a minimum of \mathfrak{A} . Hence, there must exist a minimum $\beta \in \mathfrak{A}$ such that

$$|\beta|_i < |\alpha|_i \quad (5.3)$$

for $1 \leq i \leq m$. Applying Lemma 4.2.12, we see that for

$$q > n \log\left(\frac{3}{2}\right) + \frac{3 \log(n)}{2} + \log\left(\frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})}\right) - \log(c) + 3, \quad (5.4)$$

where $c \in \mathbb{R}$, $0 < c < 1$, and $\mathbf{A} = (\underline{\alpha}_1, \dots, \underline{\alpha}_n)$, we have

$$\|\underline{\alpha}\|_2^2 \leq (1+c)^2 \lambda_i^2(\Lambda) \leq (1+c)^2 \|\underline{\beta}\|_2^2. \quad (5.5)$$

We obtain

$$\sum_{i=1}^s (|\alpha|_i^2 - |\beta|_i^2) + \sum_{i=s+1}^m (|\alpha|_i - |\beta|_i) \leq c(2+c) \|\underline{\beta}\|_2^2,$$

and by (5.3)

$$0 < |\alpha|_i - |\beta|_i \leq c' \quad (5.6)$$

for $1 \leq i \leq m$, where $c' = \max \left\{ c(2+c) \|\underline{\beta}\|_2, \sqrt{c(2+c) \|\underline{\beta}\|_2} \right\}$. By (5.3) and (5.5) we have

$$\|\underline{\beta}\|_2 \leq 4\lambda_1(\underline{\mathfrak{A}}). \quad (5.7)$$

Thus, if

$$c < \frac{1}{12\lambda_1(\underline{\mathfrak{A}})} \quad (5.8)$$

then $c(2+c)\|\underline{\beta}\|_2 \leq 3c\|\underline{\beta}\|_2 \leq 1$, and therefore $c' = c(2+c)\|\underline{\beta}\|_2$. Now we can rewrite (5.6) as

$$\frac{|\alpha|_i}{|\beta|_i} \leq 1 + \frac{c'}{|\beta|_i}. \quad (5.9)$$

Using the arithmetic-mean-geometric-mean inequality and Hölders inequality we have

$$|\beta|_i = \frac{|\mathbf{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|}{\prod_{\substack{1 \leq j \leq n \\ j \neq i}} |\beta|_j} \geq \frac{|\mathbf{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|(m-1)^{m-1}}{m \|\underline{\beta}\|_2^m}.$$

Inserting in (5.9) yields

$$\frac{|\alpha|_i}{|\beta|_i} \leq 1 + \frac{c'm \|\underline{\beta}\|_2^m}{|\mathbf{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|(m-1)^{m-1}}. \quad (5.10)$$

Next, we observe that since $d(\underline{\mathfrak{A}})\underline{\mathfrak{A}} \subseteq \mathcal{O}$, we have by Lemma 3.2.7 and Proposition 3.1.7

$$d(\underline{\mathfrak{A}})^n \prod_{i=1}^m |\alpha|_i = \mathbf{N}_{\mathbb{F}/\mathbb{Q}}(d(\underline{\mathfrak{A}})\alpha) \geq \mathbf{N}_{\mathbb{F}/\mathbb{Q}}(d(\underline{\mathfrak{A}})\beta) + 1 = d(\underline{\mathfrak{A}})^n \prod_{i=1}^m |\beta|_i + 1.$$

By (5.10) this gives us

$$1 + \frac{1}{d(\underline{\mathfrak{A}})^n |\mathbf{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|} \leq \prod_{i=1}^m \frac{|\alpha|_i}{|\beta|_i} \leq \left(1 + \frac{c'm \|\underline{\beta}\|_2^m}{|\mathbf{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|(m-1)^{m-1}} \right)^m \quad (5.11)$$

$$\begin{aligned}
&= 1 + \sum_{i=1}^m \binom{m}{i} \left(\frac{c' m \|\underline{\beta}\|_2^m}{|\mathbb{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|(m-1)^{m-1}} \right)^i \\
&\leq 1 + \sum_{i=1}^m \binom{m}{i} \left(\frac{c' m \|\underline{\beta}\|_2^m}{|\mathbb{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|(m-1)^{m-1}} \right) \\
&\leq 1 + 2^m \left(\frac{c' m \|\underline{\beta}\|_2^m}{|\mathbb{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|(m-1)^{m-1}} \right).
\end{aligned}$$

We note that the inequality in the third line of (5.11) is correct, provided that

$$c \leq \frac{(m-1)^{m-1} \mathcal{N}_{\mathcal{O}}(\mathfrak{A})}{3m(4\lambda_1(\mathfrak{A}))^m}, \quad (5.12)$$

since by Lemma 3.4.15 this implies that

$$c' \leq \frac{|\mathbb{N}_{\mathbb{F}/\mathbb{Q}}(\beta)|(m-1)^{m-1}}{m \|\underline{\beta}\|_2^m}.$$

Now, from (5.11) we obtain

$$\frac{(m-1)^{m-1}}{d(\mathfrak{A})^n m 2^m \|\underline{\beta}\|_2^m} \leq c' = c(2+c) \|\underline{\beta}\|_2. \quad (5.13)$$

Applying (5.7) we see that for

$$c < \frac{(m-1)^{m-1}}{3d(\mathfrak{A})^n m 2^m (4\lambda_1(\mathfrak{A}))^{m+1}}, \quad (5.14)$$

we have a contradiction to (5.13). Thus, if c satisfies (5.8), (5.12), and (5.14) then α is a minimum of \mathfrak{A} . But by Theorem 3.5.7, (3.5) and Proposition 3.4.11 and Corollary 3.4.13 this is certainly true if

$$c = \frac{2^{\frac{1}{n^2}(\text{size}(\mathfrak{A}) - (n^2+1)(\log(d(\mathfrak{A})+2))})}{3d(\mathfrak{A})^n m 2^{3m} m^{m/2} \sqrt{\Delta} 2^{\text{size}(\mathfrak{A})}}. \quad (5.15)$$

Since by Lemma 3.5.23 and Proposition 3.4.11 and Corollary 3.4.13

$$\frac{\text{dft}(\mathbf{A})}{\lambda_1(\mathfrak{A})} \leq \frac{2^t \prod_{i=1}^n \|\underline{\alpha}_i\|_2}{\sqrt{\Delta} \mathcal{N}_{\mathcal{O}}(\mathfrak{A}) \lambda_1(\mathfrak{A})} \leq \frac{2^{n+1} (n^{7/2} \|\text{MT}(\Omega)\|_{\infty} 2^{\text{size}(\mathfrak{A})})^n}{\sqrt{\Delta} \sqrt{n} 2^{(1+1/n) \frac{1}{n^2} (\text{size}(\mathfrak{A}) - (n^2+1)(\log(d(\mathfrak{A})+2))}}},$$

the lower bound on q follows from (5.4).

To proof (5.2) we note that by (5.5) $\|\underline{\alpha}\|_2 \leq 2\lambda_1(\mathfrak{A})$. Thus the assertion is an immediate consequence of Theorem 3.5.7 and Proposition 3.5.21. \square

Clearly, Lemma 5.2.2 implies the correctness of the following Algorithm 5.2.3.

Algorithm 5.2.3 (REDUCE)

Input : an order \mathcal{O} of the number field \mathbb{F} ; an ideal \mathfrak{A} of \mathcal{O}
Output : a reduced ideal \mathfrak{B} of \mathcal{O} and a minimum α of \mathfrak{A} such that
 $\mathfrak{B} = (1/\alpha)\mathfrak{A}$ and $|\alpha|_i \leq 2\sqrt{n} \left(\sqrt{\Delta} N_{\mathcal{O}}(\mathfrak{A}) \right)^{1/n}$ for $1 \leq i \leq n$

- (1) **procedure** REDUCE ($\mathcal{O}, \mathfrak{A}$)
- (2) /* Let $(d(\mathfrak{A}), (a_{i,j}) \in \mathbb{Z}^{n \times n})$ be the standard representation of \mathfrak{A} */
- (3) **for** ($i := 1$ **to** n **step** 1) **do**
- (4) Compute approximation \mathbf{a}'_i to $\underline{\alpha}_i = (1/d(\mathfrak{A})) \sum_{j=1}^n a_{j,i} \underline{\omega}_j$ of precision q , where q satisfies (5.1)
- (5) **od**
- (6) Compute $\mathbf{t} \in \mathbb{Z}^n$ such that $\sum_{i=1}^n \mathbf{t}_i \mathbf{a}'_i$ is a shortest vector of the lattice with basis $(\mathbf{a}'_1, \dots, \mathbf{a}'_n)$;
- (7) $\alpha := \sum_{i=1}^n \mathbf{t}_i \alpha_i$;
- (8) $\mathfrak{B} := (1/\alpha)\mathfrak{A}$
- (9) **end procedure**

To complete the description of Algorithm 5.2.3 we have to describe how to perform step (6), that means how to compute a shortest vector of a lattice. While the algorithm of Fincke and Pohst in [23] for finding such a vector is very effective in practice, it is for our theoretical purpose more convenient to use the ideas of [30] and [33], since the algorithm presented there has a worst case running time that is exponential only in the dimension n . We start by describing lattice bases of a special form.

Definition 5.2.4 Let Λ be a lattice in \mathbb{R}^n of dimension n . A basis $(\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n)$ of Λ is said to be *Korkine-Zolotaref reduced* if, for $1 \leq i \leq n$, we have

$$\|\mathbf{b}_i\|_2 = \lambda_1(\Gamma_i),$$

where Γ_i is the projection of Λ onto $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_{i-1})^\perp$, and, for $1 \leq j < i \leq n$, we have

$$\left| \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \right| \leq \frac{1}{2}.$$

In [37] the authors study many properties of Korkine-Zolotaref reduced bases. We are interested in the results of [37, Theorem 2.1 and Theorem 2.3] which we summarize in the following

Proposition 5.2.5 *Let Λ be a lattice in \mathbb{R}^n of dimension n . If $(\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n)$ is a Korkine-Zolotaref reduced basis of Λ , then*

$$\prod_{i=1}^n \|\mathbf{b}_i\|_2 \leq \left(\gamma_n^n \prod_{i=1}^n \frac{i+3}{4} \right)^{\frac{1}{2}} \det(\Lambda),$$

and for $1 \leq i \leq n$ we have

$$\frac{4}{i+3} \lambda_i(\Lambda)^2 \leq \|\mathbf{b}_i\|_2^2 \leq \frac{i+3}{4} \lambda_i(\Lambda)^2.$$

Theorem 5.2.6 *There is an algorithm that given a basis $\mathbf{A} \in \mathbb{Z}^{n \times n}$ of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$ determines a Korkine-Zolotaref reduced basis \mathbf{B} of Λ and a matrix $\mathbf{T} \in \mathbb{Z}^{n \times n}$ with $\mathbf{B} = \mathbf{A}\mathbf{T}$ and*

$$\|\mathbf{T}\|_\infty \leq \frac{1}{|\det(\mathbf{A})|} (\sqrt{n} \|\mathbf{A}\|_\infty)^n \quad (5.16)$$

in time

$$n^{o(n) + \frac{n}{2}} (\log(n \|\mathbf{A}\|_\infty))^3.$$

Proof. From [30, Proposition 3.4] and Lemma 3.5.12 it follows that there is an algorithm that given \mathbf{A} determines \mathbf{B} in time

$$n^{o(n) + \frac{n}{2}} (\log(n \|\mathbf{A}\|_\infty))^3.$$

(More details and a slightly sharper time bound can be found in [1, Sect. 2.6].) Therefore, we only have to show how to compute \mathbf{T} in the stated time. Suppose that $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]$ and $\mathbf{T} = [\mathbf{t}_1, \dots, \mathbf{t}_n]$. Then for $1 \leq i \leq n$ we can find \mathbf{t}_i by solving the linear system $\mathbf{b}_i = \mathbf{A}\mathbf{t}_i$. This can be done in time $O\left(n^3 \log\left((n \|\mathbf{A}\|_\infty)^{n-1} \|\mathbf{b}_i\|_2\right)\right)$, as is shown for example in [43, Sect. 1.4]. Thus, given \mathbf{A} and \mathbf{B} , we can compute \mathbf{T} in time

$$O\left(n^4 \log\left((n \|\mathbf{A}\|_\infty)^{n-1} \sqrt{n} \|\mathbf{B}\|_\infty\right)\right). \quad (5.17)$$

Next, we observe that by Proposition 5.2.5 and Lemma 3.5.12 we have

$$\|\mathbf{B}\|_\infty \leq \left(\frac{n+3}{4}\right)^{\frac{1}{2}} \lambda_m(\Lambda) \leq \left(\frac{n(n+3)}{4}\right)^{\frac{1}{2}} \|\mathbf{A}\|_\infty \leq n \|\mathbf{A}\|_\infty. \quad (5.18)$$

Combining (5.17) and (5.18) and applying some elementary algebraic transformations we see that given \mathbf{A} and \mathbf{B} we can determine \mathbf{T} in time

$$O\left(n^5 \log(n \|\mathbf{A}\|_\infty)\right).$$

But this time bound is surely dominated by the time needed for the computation of \mathbf{B} . This proves the first assertion of the theorem.

Finally, we have to prove the estimation of $\|\mathbf{T}\|_\infty$. Since for $1 \leq i \leq n$ we have $\mathbf{b}_i = \mathbf{A}\mathbf{t}_i$ we obtain from Corollary 3.5.13, Corollary 3.5.18 and (5.18)

$$\|\mathbf{t}_i\|_\infty \leq \frac{1}{|\det(\mathbf{A})|} (\sqrt{n} \|\mathbf{A}\|_\infty)^n.$$

This completes the proof of the theorem. \square

Corollary 5.2.7 *There is an algorithm that given a basis $\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_n) \in \mathbb{Z}^{n \times n}$ of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$ determines a shortest vector \mathbf{v} of Λ and a vector $\mathbf{t} \in \mathbb{Z}^n$ with $\mathbf{v} = \sum_{i=1}^n \mathbf{t}_i \mathbf{a}_i$ in time*

$$n^{o(n) + \frac{n}{2}} (\log(n \|\mathbf{A}\|_\infty))^3.$$

Proof. Let $(\mathbf{b}_1, \dots, \mathbf{b}_n)$ be a Korkine-Zolotaref reduced basis of Λ , let $\mathbf{T} = (\mathbf{y}_1, \dots, \mathbf{y}_n)$ be a matrix such that $\mathbf{B} = \mathbf{A}\mathbf{T}$. Then Definition 5.2.4 implies $\|\mathbf{b}_1\|_2 = \lambda_1(\Lambda)$. Furthermore, since $\mathbf{b}_1 = \mathbf{A}\mathbf{y}_1$ we can set $\mathbf{t} = \mathbf{y}_1$. Thus the assertion follows from Theorem 5.2.6. \square

Theorem 5.2.8 *On input of an ideal \mathfrak{A} of an order \mathcal{O} Algorithm 5.2.3 determines a minimum α of \mathfrak{A} , such that $|\alpha|_i \leq 2\sqrt{n} \left(\sqrt{\Delta} N_{\mathcal{O}}(\mathfrak{A}) \right)^{1/n}$ for $1 \leq i \leq m$, and the reduced ideal $\mathfrak{B} = (1/\alpha)\mathfrak{A}$ in time*

$$n^{o(n) + \frac{n}{2}} (\log(\Delta) + \log \|\mathbf{MT}(\Omega)\|_\infty + \text{size}(\mathfrak{A}))^{O(1)}.$$

Proof. We use the assumptions and the notation of Lemma 5.2.2 and Algorithm 5.2.3. Let $\mathbf{A}' = (\mathbf{a}'_1, \dots, \mathbf{a}'_n)$ and let Λ' be the lattice with basis \mathbf{A}' . W.l.o.g. we may assume that $\mathbf{C} = 2^q \mathbf{A}' \subseteq \mathbb{Z}^{n \times n}$. We let \mathbf{v} be a shortest vector in the lattice Λ'' with basis \mathbf{C} . Then $\mathbf{v} \in \mathbb{Z}^n$ and $2^{-q}\mathbf{v}$ is a shortest vector of Λ' .

From Corollary 5.2.7 it follows that given \mathbf{C} we can compute a shortest vector \mathbf{v} of Λ'' and $\mathbf{t} \in \mathbb{Z}^n$ such that $\mathbf{v} = \mathbf{C}\mathbf{t}$, in time

$$n^{o(n) + \frac{n}{2}} (\log(n \|\mathbf{C}\|_\infty))^3. \quad (5.19)$$

By Lemma 3.5.14 we know that

$$\|\mathbf{C}\|_\infty \leq n2^q \|\mathbf{A}'\|_\infty. \quad (5.20)$$

Since q satisfies (5.4), by Lemma 4.2.12 and (4.28) we have for $1 \leq i \leq n$

$$\|\mathbf{a}_i\|_\infty \leq \|\mathbf{a}_i\|_2 \leq \frac{4}{3} \|\underline{\alpha}_i\|_2 \leq \frac{4\sqrt{n}}{3} \|\underline{\alpha}_i\|_\infty,$$

and since Lemma 3.5.23 implies that

$$\|\underline{\alpha}_i\|_\infty \leq n^2 \|\mathbf{MT}(\Omega)\|_\infty 2^{\text{size}(\alpha_i)} \leq n^2 \|\mathbf{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{A})},$$

we obtain

$$\|\mathbf{A}'\|_\infty \leq \frac{4n^3}{3} \|\mathbf{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{A})}. \quad (5.21)$$

Thus, combining (5.19), (5.20) and (5.21) and Lemma 5.2.2 yield that we can compute $\mathbf{t} \in \mathbb{Z}^n$ such that $\sum_{i=1}^n \mathbf{t}_i \mathbf{a}_i$ is a shortest vector of Λ' and therefore $\alpha := \sum_{i=1}^n \mathbf{t}_i \alpha_i$ is a minimum of α satisfying (5.2) in time

$$n^{o(n) + \frac{n}{2}} \left(\log \left(2^q \|\mathbf{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{A})} \right) \right)^{O(1)}, \quad (5.22)$$

provided we already know the approximations $\mathbf{a}_1, \dots, \mathbf{a}_n$ of precision q . But the term (5.22) dominates the running time that according to Theorem 4.3.1 is necessary to compute these approximations. Since q has to satisfy (5.1) this implies that we can compute \mathbf{t} in time

$$n^{\alpha(n) + \frac{n}{2}} (\log(\Delta) + \log \|\text{MT}(\Omega)\|_\infty + \text{size}(\mathfrak{A}))^{O(1)},$$

where we have used that $\text{size}(\mathfrak{A}) \geq \log(d(\mathfrak{A}))$ and $\log(d(\mathfrak{A}) + 2) \leq \log(d(\mathfrak{A})) + 2$. The same time suffices to compute $\alpha = \sum_{i=1}^n \mathbf{t}_i \alpha_i$ since $\text{size}(\alpha) \leq \text{size}(\mathfrak{A})$.

By Lemma 3.5.25 we know that

$$\text{size}(\alpha) \leq n \log \left(d(\mathfrak{A}) \sqrt{n} \max\{\mathbf{H}(\alpha), (\mathbf{H}(\alpha))^2\} \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \log(d(\mathfrak{A})).$$

Applying (5.2) and Corollary 3.4.13 and using that by Lemma 3.4.7 we can compute $(1/\alpha)\mathfrak{A}$ in polynomial time we obtain that there is an algorithm that computes \mathfrak{B} in time

$$(\text{size}(\mathfrak{A}) + \log(\Delta) + \log \|\text{MT}(\Omega)\|_\infty + n)^{O(1)}.$$

Since this term is dominated by the running time of computing α and by using the results of [51] we can compute an appropriate value for q in the same running time this concludes the proof. \square

5.3 Computing Neighbors

Next, we want to explain the calculation of all neighbors of a minimum β in a reduced ideal \mathfrak{C} of \mathcal{O} . Again, we especially pay attention to the influence of the degree of the actual number field. We start by giving a short presentation of the theory of *minimal sets* introduced in [4]. Note, that by Proposition 5.1.3 it is sufficient to explain how to compute all neighbors of 1 in the reduced ideal $\mathfrak{D} = (1/\beta)\mathfrak{C}$.

Now, let \mathfrak{D} be a reduced ideal, and let S be a finite non-empty subset of \mathfrak{D} . Then we set for $1 \leq i \leq m$

$$|S|_i = \max\{|\alpha|_i : \alpha \in S\}$$

and

$$S(i) = \{\alpha : \alpha \in S, |\alpha|_i = |S|_i, |\alpha|_j < |S|_j \text{ for all } 1 \leq j \leq m, j \neq i\}.$$

Definition 5.3.1 Let \mathfrak{D} be a reduced ideal of an order \mathcal{O} , and let S be a finite non-empty subset of \mathfrak{D} . The set S is called *minimal*, if $S = \{\alpha : \alpha \in \mathfrak{D}, 0 < |\alpha|_i \leq |S|_i \text{ for all } 1 \leq i \leq m\}$ and if there is no $\alpha \neq 0$ in \mathfrak{D} with $|\alpha|_i < |S|_i$ for $1 \leq i \leq m$.

Clearly, if α is an element of a minimal set that contains the element 1, then α is a neighbor of 1, and each neighbor of 1 belongs to a minimal subset containing 1. Thus, we can determine all neighbors of 1 by computing all minimal sets containing 1. By [6, Theorem 12.2] we have

Proposition 5.3.2 *The number of elements of a minimal subset of \mathfrak{D} is at most $2^s 6^t$.*

Hence, Corollary 5.1.16 implies

Proposition 5.3.3 *The number of minimal sets containing 1 is at most $4^n (m \log(\Delta))^r$.*

For every minimal set S and for every $i \in \mathbb{N}$, $1 \leq i \leq m$, there is precisely one minimal set S' with $|S'|_j = |S|_j$ for $1 \leq j \leq m$, $j \neq i$, and $S'(i) \neq \emptyset$. We call S' the i -th expansion of S , and we write $S' = e_i(S)$. The i -th compression of a minimal set S is defined to be $k_i(S) = \{\alpha : \alpha \in S \mid |\alpha|_i < |S|_i\}$. We call a minimal set S' neighbor of a minimal set S if S' is either an expansion or a compression of S . We write $S N S'$.

Lemma 5.3.4 *Let \mathfrak{D} be a reduced ideal of an order \mathcal{O} , and let S and S' be minimal sets of \mathfrak{D} containing 1. Then there is a sequence $S = S_1, S_2, \dots, S_{\ell-1}, S_\ell = S'$ ($\ell \in \mathbb{N}$) of minimal sets of \mathfrak{D} with $S_1 N S_2 N \dots N S_{\ell-1} N S_\ell$ and $1 \in S_j$ for $1 \leq j \leq \ell$.*

Proof. As in [4] we see that if $S \subset S'$ we can construct the connecting sequence by applying only compressions. If $S \not\subset S'$ then we can apply Lemma 3.5 in [4]. We obtain a sequence $S = S_1, S_2, \dots, S_{\ell-1}, S_\ell = S'$ with $S_1 N S_2 N \dots N S_{\ell-1} N S_\ell$, where for $2 \leq j \leq \ell$ we have

$$1 \in \mathcal{N}(S) \cap \mathcal{N}(S') \subset \mathcal{N}(S_j).$$

(Here by $\mathcal{N}(S)$ we denote the norm body of S as defined in Definition 5.1.4.) This implies the assertion. \square

Using the above lemma we construct the procedure NEIGHBORS, that on input of a reduced ideal \mathfrak{C} and $\alpha \in \mathfrak{C}$ finds all neighbors of α . Suppose that the i -th expansion and compression of a minimal set S are computed by the procedures EXPAN and COMP on input of the corresponding index i and the set S . NEIGHBORS also uses the subroutine FILL that on input of algebraic numbers $\alpha_1, \dots, \alpha_m$, where $|\alpha_i|_i = |S|_i$ for a minimal set S and $1 \leq i \leq m$, computes the set S . Then NEIGHBORS works as follows:

Algorithm 5.3.5 (NEIGHBORS)

Input : an order \mathcal{O} ; a reduced ideal \mathfrak{C} of \mathcal{O} ; a minimum α of \mathfrak{C}
Output : the set N of all neighbors of α in \mathfrak{C}

```

(1) procedure NEIGHBORS ( $\mathcal{O}, \mathfrak{C}, \alpha$ )
(2)    $\mathfrak{D} := (1/\alpha)\mathfrak{C}$ ;
(3)    $\mathcal{S} := \text{FILL}(\mathcal{O}, \mathfrak{D}, 1, \dots, 1)$ ;
(4)    $\mathcal{S} := \{\mathcal{S}\}$ ;
(5)    $N' := \mathcal{S}$ ;
(6)   for (every set  $S \in \mathcal{S}$ ) do
(7)     for ( $i := 1$  to  $m$  step 1) do
(8)        $\mathcal{S} := \mathcal{S} \cup \{\text{EXPAN}(\mathcal{O}, \mathfrak{D}, i, S)\}$ ;
(9)        $N' := N' \cup \text{EXPAN}(\mathcal{O}, \mathfrak{D}, i, S)$ ;
(10)      if ( $1 \in \text{COMP}(\mathcal{O}, \mathfrak{D}, i, S)$ ) then
(11)         $\mathcal{S} := \mathcal{S} \cup \{\text{COMP}(\mathcal{O}, \mathfrak{D}, i, S)\}$ ;
(12)         $N' := N' \cup \text{COMP}(\mathcal{O}, \mathfrak{D}, i, S)$ ;
(13)      fi
(14)    od
(15)  od
(16)   $N := \{\nu : \nu = \alpha\beta, \beta \in N'\}$ ;
(17) end procedure

```

By Lemma 5.3.4 we have

Proposition 5.3.6 *If FILL, EXPAN and COMP are correct then NEIGHBORS is correct.*

Finally we have to describe the implementation of FILL, COMP and EXPAN. We use the results of section 4.3. COMP uses the procedure VAPPR that uses the algorithm described in Theorem 4.3.3 to compute approximations to the archimedean valuations of algebraic integers. For an algebraic number α , an rational integer $1 \leq i \leq m$ and $q \in \mathbb{N}$ we denote by $\text{VAPPR}(\mathcal{O}, \alpha, i, q)$ the q -approximation to $|\alpha|_i$ computed by VAPPR. Clearly, the compression can be computed as follows:

Algorithm 5.3.7 (COMP)

Input : an order \mathcal{O} ; a reduced invertible ideal \mathfrak{D} of \mathcal{O} ; a minimal set S ; $i \in \mathbb{N}$ with $1 \leq i \leq m$
Output : the i -th compression $k_i(S)$ of S

- (1) **procedure** COMP ($\mathcal{O}, \mathfrak{D}, i, S$)
- (2) $k_i(S) := \emptyset$;
- (3) Compute $p \in \mathbb{N}$, $p > n^2 \log(4\Delta^{3/2}) + 2$;
- (4) Compute $q \in \mathbb{N}$, $q > \log(\sqrt{\Delta}) + \log(\sqrt{\Delta} + 1) + p + 3$;
- (5) **for** (every $\alpha \in S$) **do**
- (6) **if** ($\text{VAPPR}(\mathcal{O}, \alpha, i, q) < \max\{\text{VAPPR}(\mathcal{O}, \beta, i, q) : \beta \in S\} - 2^{-p}$) **then**
- (7) $k_i(S) := k_i(S) \cup \{\alpha\}$;
- (8) **fi**
- (9) **od**
- (10) **end procedure**

Proposition 5.3.8 *COMP (Algorithm 5.3.7) is correct. Moreover, given \mathcal{O} , a reduced ideal \mathfrak{D} , a minimal set S containing 1 and $i \in \mathbb{N}$, $1 \leq i \leq m$, COMP computes the i -th compression of S in time*

$$(4^n + \log(\Delta) + \log(\text{MT}(\Omega)))^{O(1)} .$$

Proof. Since \mathfrak{D} is reduced by Lemma 5.1.11 we have $d(\mathfrak{A}) \leq \sqrt{\Delta}$. Since S only contains neighbors of 1, we have by Corollary 5.1.9 for every $\alpha \in S$

$$H(\alpha) \leq \sqrt{\Delta} . \tag{5.23}$$

Thus, Theorem 4.3.4 implies that COMP is correct.

Clearly, we can find appropriate p and q in the stated time. By Proposition 5.3.2 the set S contains at most $2^{s_6 t} \leq 4^n$ elements. From (5.23) and Lemma 3.5.25 it follows that for each $\alpha \in S$ we have

$$\text{size}(\alpha) \leq (n+1) \log(\Delta) + 4 \log(n) + 2n^2 + n + \log(\text{MT}(\Omega)) + 2 . \tag{5.24}$$

By Theorem 4.3.3 the running time of each call up of $\text{VAPPR}(\mathcal{O}, \beta, i, q)$ is

$$O \left(n^8 (q + \text{size}(\alpha) + (\log(n \|\text{MT}(\Omega)\|_\infty))^3) (\log(q + \text{size}(\alpha) + \log \|\text{MT}(\Omega)\|_\infty))^2 \right) . \tag{5.25}$$

Thus, (5.24) and (5.25) imply the bound of the running time. \square

Before we start with the description of FILL and EXPAN, we need a technical result that shows that we can perform all computations in approximations to Minkowski lattices of appropriate precision.

Theorem 5.3.9 *Let \mathfrak{D} be a reduced ideal of \mathcal{O} with basis $(\delta_1, \dots, \delta_n)$. Let $q \in \mathbb{N}$, and let $\mathbf{D}' = (\mathbf{d}_1, \dots, \mathbf{d}_n) \in \mathbb{Q}^{n \times n}$, where \mathbf{d}_i is a q -approximation to δ_i for $1 \leq i \leq n$. Let $\beta \in \mathfrak{D}$, $\beta = \sum_{i=1}^n x_i \delta_i$ with $x_i \in \mathbb{Z}$ for $1 \leq i \leq n$, and set $\mathbf{b} = \sum_{i=1}^n x_i \mathbf{d}_i$. Finally, for $1 \leq j \leq m$ let $\alpha_j \in \mathfrak{D}$ with $\alpha_j = \sum_{i=1}^n \mathbf{y}_{i,j} \delta_i$, where $\mathbf{y}_j \in \mathbb{Z}^n$, $H(\alpha_j) \leq \sqrt{\Delta}$, and set $\mathbf{a}_j = \sum_{i=1}^n \mathbf{y}_{i,j} \mathbf{d}_i$. If*

$$q > \log(\sqrt{\Delta}) + \log \left(n^4 \Delta \left(\frac{4\sqrt{n\Delta}}{3} + 1 \right) 2^{n+1} \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^{n+1} + 1 \right) + p + 3,$$

where $p \in \mathbb{N}$ with

$$p > n^2 \log \left(4n^4 \Delta^2 \left(\frac{4\sqrt{n\Delta}}{3} + 1 \right) 2^{n+1} \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^{n+1} \right) + 2,$$

then we have

$$|\beta|_i \leq |\alpha_i|_i \text{ for all } 1 \leq i \leq m \iff |\mathbf{b}|_i \leq |\mathbf{a}_i|_i + 2^{-p} \text{ for all } 1 \leq i \leq m, \quad (5.26)$$

$$|\beta|_i < |\alpha_i|_i \text{ for all } 1 \leq i \leq m \iff |\mathbf{b}|_i < |\mathbf{a}_i|_i - 2^{-p} \text{ for all } 1 \leq i \leq m. \quad (5.27)$$

Proof. Clearly for all $\beta \in \mathfrak{D}$ such that $|\beta|_i \leq |\alpha_i|_i$ for all i with $1 \leq i \leq m$, we have $H(\beta) \leq \sqrt{\Delta}$. Next, we apply Corollary 3.5.18, Corollary 3.5.24, and Proposition 3.5.21 and Proposition 3.5.22, and see that for $1 \leq i \leq n$ we have

$$|x_i| \leq n\Delta \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^n.$$

Thus, by Proposition 4.1.8 the vector \mathbf{b} is an approximation to $\underline{\beta}$ of precision

$$q - \log(n) - \log \left(n\Delta \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^n \right).$$

Thus by our conditions on q and p we can apply Theorem 4.3.4 which proves one direction of the equivalences (5.26) and (5.27).

Next, we show the other direction for (5.26). The proof for (5.27) is completely analogous. Suppose that $|\mathbf{b}|_i \leq |\mathbf{a}_i|_i + 2^{-p}$ for all $1 \leq i \leq m$. In the following we want to find an upper bound on $|\beta^{(i)}|$ that allows us to apply Theorem 4.3.4.

By Corollary 3.5.18 we know that

$$|x_i| \leq \|\mathbf{b}\|_2 \frac{\text{dft}(\mathbf{D}')}{\lambda(\mathbf{D}')} . \quad (5.28)$$

The condition on q and Corollary 4.2.6 imply that $\|\mathbf{d}_i\|_2 \leq (4/3) \|\underline{\delta}_i\|_2$ and $(5/4) \|\mathbf{d}_i\|_2 \geq \|\underline{\delta}_i\|_2$. Thus from (5.28) it follows that

$$|x_i| \leq \|\mathbf{b}\|_2 \frac{5}{4} \left(\frac{4}{3} \right)^n \frac{\prod_{i=1}^n \|\underline{\delta}_i\|_2}{\lambda(\underline{\mathfrak{D}})} .$$

Applying Proposition 3.5.22, Corollary 3.5.24 and Corollary 5.1.10 we obtain

$$|x_i| \leq \sqrt{\Delta} \|\mathbf{b}\|_2 2^{n+1} \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^n.$$

Therefore, for $1 \leq i \leq n$

$$|\beta^{(i)}| \leq n\sqrt{\Delta} \|\mathbf{b}\|_2 2^{n+1} \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^n \max \left\{ |\delta_1^{(i)}|, \dots, |\delta_n^{(i)}| \right\}. \quad (5.29)$$

Finally, we need an upper bound on $\|\mathbf{b}\|_2$. First we observe that

$$\|\mathbf{b}\|_i \leq \max \{ |\alpha|_i + 2^{-p} : \alpha \in T \}. \quad (5.30)$$

Note, that for all α_j we have $H(\alpha_j) \leq \sqrt{\Delta}$, and therefore $\|\alpha_j\|_2 \leq \sqrt{n\Delta}$. Again, we apply Corollary 3.5.18, Corollary 3.5.24, Proposition 3.5.21 and Proposition 3.5.22 and obtain

$$\|\mathbf{y}_j\|_2 \leq n\Delta \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^n.$$

Thus, \mathbf{a}_j is an approximation to α_j of precision

$$q - \log(n) - \log \left(n\Delta \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^n \right).$$

Moreover, q satisfies the conditions of Lemma 4.2.5, and we obtain

$$\|\alpha_j\|_2 \leq \frac{4}{3} \|\alpha\|_2 \leq \frac{4\sqrt{n\Delta}}{3},$$

and therefore for $1 \leq i \leq m$

$$|\alpha_j|_i \leq \frac{4\sqrt{n\Delta}}{3}.$$

Inserting in (5.30) and simple estimations yield

$$\|\mathbf{b}\|_2 \leq \sqrt{n} \left(\frac{4\sqrt{n\Delta}}{3} + 2^{-p} \right),$$

and the combination with (5.29) and Lemma 3.5.23 and some crude inequalities give us

$$H(\beta) \leq n^4 \Delta \left(\frac{4\sqrt{n\Delta}}{3} + 1 \right) 2^{n+1} \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^{n+1}.$$

Now, it follows from Theorem 4.3.4 that $|\beta|_i \leq |\alpha_i|_i$ for $1 \leq i \leq m$. \square

Corollary 5.3.10 *Let \mathfrak{A} , $(\delta_1, \dots, \delta_n)$, $D' = (\mathbf{d}_1, \dots, \mathbf{d}_n)$, \mathbf{b} , and $\alpha_1, \dots, \alpha_n$ be as in Theorem 5.3.9. Furthermore, for $1 \leq i \leq n$ let \mathbf{a}_j be a q -approximation to $\underline{\alpha}_j$. If*

$$q > \log(\sqrt{\Delta}) + \log \left(n^4 \Delta \left(\frac{4\sqrt{n\Delta}}{3} + 1 \right) 2^{n+1} \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^{n+1} + 1 \right) + p + 3,$$

where $p \in \mathbb{N}$ with

$$p > n^2 \log \left(4n^4 \Delta^2 \left(\frac{4\sqrt{n\Delta}}{3} + 1 \right) 2^{n+1} \left(n^3 \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})} \right)^{n+1} \right) + 2,$$

then we have

$$\begin{aligned} |\beta|_i \leq |\alpha_i|_i \text{ for all } 1 \leq i \leq m &\iff |\mathbf{b}|_i \leq |\mathbf{a}_i|_i + 2^{-p} \text{ for all } 1 \leq i \leq m, \\ |\beta|_i < |\alpha_i|_i \text{ for all } 1 \leq i \leq m &\iff |\mathbf{b}|_i < |\mathbf{a}_i|_i - 2^{-p} \text{ for all } 1 \leq i \leq m. \end{aligned}$$

Proof. Clearly, the proof of the above theorem also works if we assume in a straightforward way that the vectors \mathbf{a}_i are q -approximations to the α_j . \square

Next, we describe the procedure FILL:

Algorithm 5.3.11 (FILL)

Input : an order \mathcal{O} ; a reduced ideal \mathfrak{D} of \mathcal{O} ; $\alpha_1, \dots, \alpha_m \in \mathfrak{D}$, where
 $|\alpha_i|_i = |S|_i$ for a minimal set S of \mathfrak{D} and $1 \leq i \leq m$
Output : the minimal set S

- (1) **procedure** FILL ($\mathcal{O}, \mathfrak{D}, \alpha_1, \dots, \alpha_m$)
- (2) /* Let $(d(\mathfrak{A}), (c_{i,j}) \in \mathbb{Z}^{n \times n})$ be the standard representation of \mathfrak{A} */
- (3) Compute p, q satisfying the conditions of Theorem 5.3.9
- (4) **for** ($j := 1$ **to** n **step** 1) **do**
- (5) Compute q -approximation \mathbf{d}_j to $\underline{\delta}_j = (1/d(\mathfrak{A})) \sum_{k=1}^n c_{j,k} \underline{\omega}_k$;
- (6) Compute q -approximation \mathbf{a}_j to α_j ;
- (7) **od**
- (8) Compute the set T of all $\mathbf{t} \in \mathbb{Z}^n$ such that $\mathbf{b} = \sum_{i=1}^n \mathbf{t}_i \mathbf{d}_i$ satisfies $|\mathbf{b}|_i \leq |\mathbf{a}_i|_i + 2^{-p}$ for $1 \leq i \leq m$;
- (9) $S := \{\alpha : \alpha = \sum_{i=1}^n \mathbf{t}_i \delta_i, \mathbf{t} \in T\}$;
- (10) **end procedure**

Proposition 5.3.12 *FILL (Algorithm 5.3.11) is correct.*

Proof. By Corollary 5.1.9 the height of every element of a minimal set is bounded by $\sqrt{\Delta}$. Hence, from Theorem 5.3.9 and Corollary 5.3.10 the assertion follows. \square

To complete the description of FILL we have to show how step (8) can be performed. We use the notation of FILL (Algorithm 5.3.11). Let $\Lambda' \subseteq \mathbb{Q}^n$ be the lattice with basis $(\mathbf{d}_1, \dots, \mathbf{d}_n)$. The task is to find the set T of all $\mathbf{t} \in \mathbb{Z}^n$ such that $\mathbf{b} = \sum_{i=1}^n \mathbf{t}_i \mathbf{d}_i \in \Lambda'$ satisfies $|\mathbf{b}|_i \leq |\mathbf{a}_i|_i + 2^{-p}$ for $1 \leq i \leq m$. But the input of FILL is a reduced ideal \mathfrak{D} and elements $\alpha_1, \dots, \alpha_m \in \mathfrak{D}$, where $|\alpha_i|_i = |S|_i$ for a minimal set S of \mathfrak{D} and $1 \leq i \leq m$.

Hence, by Theorem 5.3.9 and Corollary 5.3.10 we know that for every $\mathbf{b} \in \Lambda'$ there exists $u \in \mathbb{N}$, $1 \leq u \leq m$ with $|\mathbf{b}|_u > (|\mathbf{a}_i|_i + 2^{-p})/2$. We show how to solve the corresponding problem in an arbitrary lattice $\Lambda \in \mathbb{Z}^n$. Mainly, we follow the strategy presented in [6, Sect. 15], but using Korkine-Zolotaref reduced lattice bases and more precise estimations we obtain better upper bounds for the running time. We also note, that similar ideas were already used in [31].

Theorem 5.3.13 *There is an algorithm that given a basis \mathbf{A} of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$ and $c_1, \dots, c_n \in \mathbb{N}$ with $c_u = c_{u+t}$ for $s+1 \leq u \leq s+t$, with the property that for every lattice vector $\mathbf{v} \in \Lambda$ there exists $1 \leq u \leq m$ with*

$$|\mathbf{v}|_u > \frac{c_u}{2}, \quad (5.31)$$

computes the set of all $\mathbf{v} \in \Lambda$ satisfying

$$|\mathbf{v}|_u \leq c_u \text{ for } 1 \leq u \leq m. \quad (5.32)$$

For each of these vectors \mathbf{v} the algorithm computes $\mathbf{y} \in \mathbb{Z}^n$ such that $\mathbf{v} = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i$. With $C = \prod_{u=1}^n c_u$ the algorithm has running time

$$n^{o(n)+n} (\log(nC \|\mathbf{A}\|_\infty))^5.$$

We shall prove this theorem by transforming the lattice Λ into a new lattice Λ' such that every $\mathbf{v} \in \Lambda$ satisfying (5.32) corresponds to a “short” vector in Λ' . We start by setting

$$w_u = \left\lceil c_u^{1/e_u} \right\rceil$$

for $1 \leq u \leq n$, where

$$e_u = \begin{cases} 1 & \text{if } 1 \leq u \leq s, \\ 2 & \text{if } s+1 \leq u \leq n. \end{cases}$$

Then we have $w_u = c_u$ for $1 \leq u \leq s$ and $c_u \leq w_u^2 \leq 2c_u$ for $s+1 \leq u \leq n$. Hence, for every solution \mathbf{v} of (5.32) we have

$$|\mathbf{v}|_u \leq w_u^{e_u} \text{ for } 1 \leq u \leq m. \quad (5.33)$$

On the other hand, for every $\mathbf{v} \in \Lambda$ there exists $1 \leq u \leq m$ with

$$|\mathbf{v}|_u > \left(\frac{w_u}{2}\right)^{e_u}. \quad (5.34)$$

Now let $W = \prod_{u=1}^n w_u$, $W_u = W/w_u$ for $1 \leq u \leq n$. Then we define Λ' to be the lattice which has the basis $\mathbf{A}' = (\mathbf{a}'_1, \dots, \mathbf{a}'_n)$, where for $1 \leq j \leq n$ we have

$$\mathbf{a}'_j = (W_1 \mathbf{a}_{1,j}, \dots, W_n \mathbf{a}_{n,j})^T \in \mathbb{Z}^n.$$

Rather than looking for the solutions of (5.33) we can as well look for vectors $\mathbf{v}' \in \Lambda'$ with

$$|\mathbf{v}'|_u \leq W^{e_u} \text{ for } 1 \leq u \leq m. \quad (5.35)$$

Since every nonzero vector \mathbf{v} of Λ satisfies (5.34) we have $\lambda_1(\Lambda') \geq W/2$. Therefore, if $\mathbf{v}' \in \Lambda'$ satisfies (5.35), then

$$\|\mathbf{v}'\|_2 \leq \sqrt{m}W \leq 2\sqrt{m}\lambda_1(\Lambda'). \quad (5.36)$$

Suppose for a moment that we can find the set

$$T = \left\{ \mathbf{y} : \mathbf{y} \in \mathbb{Z}^n, \left\| \sum_{i=1}^n \mathbf{y}_i \mathbf{a}'_i \right\|_2 \leq 2\sqrt{m}\lambda_1(\Lambda') \right\}$$

in time

$$(2n^2)^{o(n) + \frac{n}{2}} \left(\log \left(2n^{\frac{3}{2}} C \|A\|_\infty \right) \right)^3,$$

where $C = \prod_{u=1}^n c_u$. Then from (5.45) it follows that for each $\mathbf{y} \in T$ we can compute $\mathbf{v} = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i$ and check whether \mathbf{v} is a solution of (5.32) performing $O(n)$ arithmetical operations on rational integers of binary length $O(n \log(nC \|A\|_\infty))$. Thus, the assertion follows from elementary transformations.

It remains to proof that we can find T in the stated time. But this is an immediate consequence of the following Algorithm 5.3.14 and Lemma 5.3.15 and Lemma 5.3.16.

Algorithm 5.3.14 (ENUM)

Input : a Korkine-Zolotaref reduced basis $(\mathbf{b}_1, \dots, \mathbf{b}_n)$ of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$; radius $r \geq \|\mathbf{b}_1\|_2$

Output : $S = \{\mathbf{x} : \mathbf{x} \in \mathbb{Z}^n, \|\sum_{i=1}^n \mathbf{x}_i \mathbf{b}_i\|_2 \leq r\}$

- (1) **procedure** ENUM (j)
- (2) $t_j := \sum_{i=j+1}^n \mathbf{x}_i \mu_{i,j}$;
- (3) **if** ($j > 1$) **then**
- (4) **for** ($\mathbf{x}_j := \lfloor -t_j - r/\|\mathbf{b}_j^*\|_2 \rfloor$ **to** $\lceil -t_j + r/\|\mathbf{b}_j^*\|_2 \rceil$ **step** 1) **do**
- (5) ENUM ($j - 1$)
- (6) **od**
- (7) **else**
- (8) **for** ($\mathbf{x}_1 := \lfloor -t_1 - r/\|\mathbf{b}_1^*\|_2 \rfloor$ **to** $\lceil -t_1 + r/\|\mathbf{b}_1^*\|_2 \rceil$ **step** 1) **do**
- (9) $\mathbf{b} := \sum_{i=1}^n \mathbf{x}_i \mathbf{b}_i$;
- (10) **if** ($\|\mathbf{b}\|_2^2 \leq r^2$) **then**
- (11) $S := S \cup \{\mathbf{b}\}$
- (12) **fi**
- (13) **od**
- (14) **fi**
- (15) **end procedure**

- (16) $S := \emptyset$;
- (17) Compute the Gram-Schmidt vectors $\mathbf{b}_1^*, \dots, \mathbf{b}_n^*$ and the coefficients $\mu_{i,j} := \langle \mathbf{b}_i, \mathbf{b}_j^* \rangle / \langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle$ for $1 \leq j < i \leq n$;
- (18) ENUM (n);

Lemma 5.3.15 *Given a Korkine-Zolotaref reduced basis $\mathbf{B} = (\mathbf{b}_1, \dots, \mathbf{b}_n)$ of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$, and a constant $r > \|\mathbf{b}_1\|_2$, Algorithm 5.3.14 finds in time*

$$(c^2 n)^{o(n) + \frac{n}{2}} (\log(cn \|\mathbf{B}\|_\infty))^3,$$

where $c = r / \|\mathbf{b}_1\|_2$, the set $S = \{\mathbf{x} : \mathbf{x} \in \mathbb{Z}^n, \|\sum_{i=1}^n \mathbf{x}_i \mathbf{b}_i\|_2 \leq r\}$. The set S contains $(c^2 n)^{o(n) + n/2}$ elements each of binary size $O(n^2 \log(cn \|\mathbf{B}\|_\infty))$.

Proof. We first claim that there exists a set $T \subseteq \mathbb{Z}^n$ of $(c^2 n)^{o(n) + n/2}$ elements that contains every $\mathbf{x} \in \mathbb{Z}^n$ with

$$\left\| \sum_{i=1}^n \mathbf{x}_i \mathbf{b}_i \right\|_2 \leq c \|\mathbf{b}_1\|_2. \quad (5.37)$$

Let

$$\mathbf{v} = \sum_{j=1}^n \mathbf{x}_j \mathbf{b}_j,$$

where $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_n) \in \mathbb{Z}^n$. Then Lemma 3.5.9 implies that

$$\mathbf{v} = \sum_{j=1}^n y_j \mathbf{b}_j^*,$$

where for $1 \leq j \leq n$

$$y_j = \frac{\langle \mathbf{v}, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} = \sum_{k=1}^n \mathbf{x}_k \frac{\langle \mathbf{b}_k, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} = \mathbf{x}_j + \sum_{k=j+1}^n \mathbf{x}_k \frac{\langle \mathbf{b}_k, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle}. \quad (5.38)$$

Now suppose that $\|\mathbf{v}\|_2 \leq c \|\mathbf{b}_1\|_2$. Then

$$\|\mathbf{v}\|_2^2 = \sum_{j=1}^n y_j^2 \|\mathbf{b}_j^*\|_2^2 \leq (c \|\mathbf{b}_1\|_2)^2, \quad (5.39)$$

and from (5.38) it follows that

$$\left| \mathbf{x}_j + \sum_{k=j+1}^n \mathbf{x}_k \frac{\langle \mathbf{b}_k, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \right| = |y_j| \leq c \frac{\|\mathbf{b}_1\|_2}{\|\mathbf{b}_j^*\|_2}. \quad (5.40)$$

Hence,

$$\left[- \sum_{k=j+1}^n \mathbf{x}_k \frac{\langle \mathbf{b}_k, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} - c \frac{\|\mathbf{b}_1\|_2}{\|\mathbf{b}_j^*\|_2} \right] \leq \mathbf{x}_j \leq \left[- \sum_{k=j+1}^n \mathbf{x}_k \frac{\langle \mathbf{b}_k, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} + c \frac{\|\mathbf{b}_1\|_2}{\|\mathbf{b}_j^*\|_2} \right]. \quad (5.41)$$

We let T be the set of all $(\mathbf{x}_1, \dots, \mathbf{x}_n) \in \mathbb{Z}^n$ that for $1 \leq j \leq n$ satisfy (5.41). Then every element $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ of T also satisfies (5.37). Clearly, $|T|$ is bounded by

$$\prod_{j=1}^n \left(2c \frac{\|\mathbf{b}_1\|_2}{\|\mathbf{b}_j^*\|_2} + 2 \right). \quad (5.42)$$

Since Definition 5.2.4 and Corollary 3.5.10 imply that $c \|\mathbf{b}_1\|_2 / \|\mathbf{b}_j^*\|_2 > 1$ for $1 \leq j \leq n$, we obtain

$$|T| < (4c)^n \prod_{j=1}^n \left(\frac{\|\mathbf{b}_1\|_2}{\|\mathbf{b}_j^*\|_2} \right),$$

and by (3.9) and (3.6) we have $|T| < (4c)^n \gamma_n^{n/2}$. Applying (3.5) we conclude that $|T| \leq (16c^2n)^{\frac{n}{2}}$, which proves our claim.

Clearly, by construction Algorithm 5.3.14 enumerates all elements of the set T and collects only the vectors $\mathbf{x} \in T$ with $\|\sum_{i=1}^n \mathbf{x}_i \mathbf{b}_i\|_2 \leq c \|\mathbf{b}_1\|_2$. Thus, Algorithm 5.3.14 is correct and collects $(c^2n)^{o(n)+n/2}$ lattice vectors. For each of those vectors it performs $n^{O(1)}$ arithmetical operations, but this factor can be omitted because of our use of the o -notation. The Gram-Schmidt vectors and the values $\mu_{i,j}$ can be computed by $O(n^3)$ operations (see [33]). Finally, the bounds in the for loops (step (4) and step (8)) can be computed in $O(\ell)$ operations, where ℓ is the binary size of the numbers involved. For more details we refer for example to [51]. We conclude, that Algorithm 5.3.14 performs $(c^2n)^{o(n)+\frac{n}{2}} + O(n^3 + n\ell)$ arithmetical operations on rational numbers.

To estimate the binary length of these numbers we mainly use Lemma 3.5.16. For $1 \leq i \leq n$ let $d_i = \prod_{j=1}^i \|\mathbf{b}_j^*\|_2$. Then, by Lemma 3.5.12 we conclude that

$$d_i \leq (m \|\mathbf{B}\|_\infty)^n. \quad (5.43)$$

Thus, from Lemma 3.5.16 it follows that the numbers involved in the computation of the Gram-Schmidt vectors and the coefficients $\mu_{i,j}$ are rationals of which the numerators and the denominators are integers of size $O(n \log(n \|\mathbf{B}\|_\infty))$.

Next, we have to estimate the size of the integers \mathbf{x}_i and t_i ($1 \leq i \leq n$). First, we observe that by (3.12) we have $d_{i-1} \mathbf{b}_i^* \in \Lambda$, hence, $d_{i-1} \|\mathbf{b}_i^*\|_2 \geq \lambda_1(\Lambda) = \|\mathbf{b}_1\|_2$, and therefore $\|\mathbf{b}_1\|_2 / \|\mathbf{b}_i^*\|_2 \leq d_{i-1}$. Applying (5.43) we conclude that $\|\mathbf{b}_1\|_2 / \|\mathbf{b}_i^*\|_2 \leq (m \|\mathbf{B}\|_\infty)^n$. Therefore, by induction on i and using (5.41) we obtain that the binary size of \mathbf{x}_i and t_i is

$$O(\log(n) + n \log(cn \|\mathbf{B}\|_\infty)) = O(n \log(cn \|\mathbf{B}\|_\infty)), \quad (5.44)$$

where the last equality follows from $\|\mathbf{B}\|_\infty \geq 1$. Thus, the binary size of each $\mathbf{x} \in S$ is $O(n^2 \log(cn \|\mathbf{B}\|_\infty))$. Furthermore, since all arithmetical operations can be done in quadratic time, we see by ‘‘brute force’’ and by the properties of the o -notation that Algorithm 5.3.14 has running time

$$(c^2n)^{o(n)+\frac{n}{2}} (\log(cn \|\mathbf{B}\|_\infty))^3. \quad \square$$

Lemma 5.3.16 *There is an algorithm that given a basis $\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_n)$ of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$, and a constant $r > \lambda_1(\Lambda)$, determines in time*

$$(c^2 n)^{o(n) + \frac{n}{2}} (\log(cn \|\mathbf{A}\|_\infty))^3,$$

where $c = r/\lambda_1(\Lambda)$, the set $T = \{\mathbf{y} : \mathbf{y} \in \mathbb{Z}^n, \|\sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i\|_2 \leq r\}$. The binary size of any $\mathbf{y} \in T$ is

$$O(n^2 \log(cn \|\mathbf{A}\|_\infty)). \quad (5.45)$$

Proof. To find the set T the algorithm first determines a Korkine-Zolotaref reduced basis $\mathbf{B} = (\mathbf{b}_1, \dots, \mathbf{b}_n)$ of Λ and a matrix $\mathbf{T} \in \mathbb{Z}^{m \times m}$ such that $\mathbf{B} = \mathbf{A}\mathbf{T}$. Then it uses Algorithm 5.3.14 to compute the set $S = \{\mathbf{x} : \mathbf{x} \in \mathbb{Z}^n, \|\sum_{i=1}^n \mathbf{x}_i \mathbf{b}_i\|_2 \leq r\}$. Note that $\lambda_1(\Lambda) = \|\mathbf{b}_1\|_2$. By Theorem 5.2.6, Lemma 5.3.15 and (5.18) this can be done in time

$$(c^2 n)^{o(n) + \frac{n}{2}} (\log(cn^2 \|\mathbf{A}\|_\infty))^3. \quad (5.46)$$

For every $\mathbf{x} \in S$ we have $\mathbf{B}\mathbf{x} = \mathbf{A}\mathbf{y}$, where $\mathbf{y} = \mathbf{T}\mathbf{x}$. Thus, we obtain the set T by multiplying \mathbf{T} with every element of S . Since $|S| = (c^2 n)^{o(n) + \frac{n}{2}}$ from (5.16), (5.44) and (5.18) it follows that these multiplications can be performed in time

$$\begin{aligned} (c^2 n)^{o(n) + \frac{n}{2}} \log(cn^2 \|\mathbf{A}\|_\infty) \log\left(\frac{1}{|\det(\mathbf{A})|} (\sqrt{n} \|\mathbf{A}\|_\infty)^n\right) \\ = (c^2 n)^{o(n) + \frac{n}{2}} (\log(cn^2 \|\mathbf{A}\|_\infty))^2, \end{aligned}$$

where we use that $|\det(\mathbf{A})| \in \mathbb{N}$, $\|\mathbf{A}\|_\infty \geq 1$ and $c \geq 1$. Thus the running time of the algorithm is dominated by (5.46). Also, the estimate of the binary size of $\mathbf{y} \in T$ is an immediate consequence of (5.16), (5.44) and (5.18). Using the properties of the o - and O -notation we obtain the stated result. \square

This completes the proof of Theorem 5.3.13.

Using the above results we can now give an upper bound for the running time of the algorithm FILL (Algorithm 5.3.11).

Lemma 5.3.17 *On input of a reduced ideal \mathfrak{D} of an order \mathcal{O} , and $\alpha_1, \dots, \alpha_m \in \mathfrak{D}$, where $|\alpha_i|_i = |S|_i$ for a minimal set S of \mathfrak{D} and $1 \leq i \leq m$, FILL (Algorithm 5.3.11) computes the set S in time*

$$n^{O(n)} (\log(\|\mathbf{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(1)}.$$

Proof. By Proposition 5.3.12 we only have to prove the stated running time. We use the notation of Algorithm 5.3.11 and Theorem 5.3.9. Let Λ' be the lattice with basis \mathbf{D}' , where $\mathbf{D}' = (\mathbf{d}_1, \dots, \mathbf{d}_n)$. We may assume that $\mathbf{C} = 2^q \mathbf{D}' \subseteq \mathbb{Z}^n$, and let Λ'' be the lattice with basis \mathbf{C} . For $1 \leq u \leq s$ let $c_u = 2^q(|\mathbf{a}_u|_u + 2^{-p})$, and for $s+1 \leq u \leq m$ let

$c_u = c_{u+t} = 2^{2q}(|\mathbf{a}_u|_u + 2^{-p})$. We apply Theorem 5.3.13 and see that we can compute the set of all $\mathbf{v} \in \Lambda''$ satisfying

$$|\mathbf{v}|_u \leq c_u \text{ for } 1 \leq u \leq m,$$

and for each of these \mathbf{v} the vector $\mathbf{y} \in \mathbb{Z}^n$ such that $\mathbf{v} = 2^q \sum_{i=1}^n \mathbf{y}_i \mathbf{d}_i$, in time

$$n^{o(n)+n} (\log(nC \|\mathbf{C}\|_\infty))^5,$$

where by Corollary 5.1.9 we have that $C \leq (2^{2q}(\sqrt{\Delta} + 1))^m$. Clearly, T is the set of these \mathbf{y} . It remains to find an upper bound on $\|\mathbf{C}\|_\infty$. As in the proof of Theorem 5.2.8 we see that

$$\|\mathbf{C}\|_\infty \leq n2^q \|\mathbf{D}'\|_\infty$$

and

$$\|\mathbf{D}'\|_\infty \leq \frac{4n^3}{3} \|\text{MT}(\Omega)\|_\infty 2^{\text{size}(\mathfrak{D})}.$$

Thus, applying the estimations of Corollary 5.1.12, we conclude that step (8) of FILL needs time

$$n^{o(n)+n} (n + q + \log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(1)}. \quad (5.47)$$

By Theorem 4.3.1 and the usual estimations of the sizes it follows that this time also suffices to compute all needed approximations in step (5) and (6). Finally, step (9) has a running time that is bounded by at most the square of (5.47). Since p and q satisfy Theorem 5.3.9 this concludes the proof. \square

Finally, we describe EXPAN. Computing the i -th expansion needs a more complicated strategy than those used in COMP or the algorithm FILL: We first compute $\alpha \in \mathfrak{D}$ with $|\alpha|_i = |e_i(S)|_i$. Then we can compute the set $e_i(S)$ using the procedure FILL. As in the case of the reduction algorithm in section 5.2 we shall use approximations of the Minkowski lattice of the ideal under consideration.

Algorithm 5.3.18 (EXPAN)

Input : an order \mathcal{O} ; a reduced invertible ideal \mathfrak{D} of \mathcal{O} ; a minimal set S ; $i \in \mathbb{N}$ with $1 \leq i \leq m$
Output : the i -th expansion E_i of S

(1) **procedure** EXPAN ($\mathcal{O}, \mathfrak{D}, i, S$)

- (2) /* Let $(d(\mathfrak{A}), (c_{k,j}) \in \mathbb{Z}^{n \times n})$ be the standard representation of \mathfrak{A} */
- (3) Compute p, q satisfying the conditions of Theorem 5.3.9
- (4) **for** $(j := 1$ **to** n **step** 1) **do**
- (5) Compute q -approximation \mathbf{d}_j to $\underline{\delta}_j = (1/d(\mathfrak{A})) \sum_{k=1}^n c_{j,k} \underline{\omega}_k$;
- (6) **od**
- (7) **for** $(j := 1$ **to** m **step** 1) **do**
- (8) Compute $\alpha_j \in S$ with $\text{VAPPR}(\mathcal{O}, \alpha_j, j, q) = \max\{\text{VAPPR}(\mathcal{O}, \beta, j, q) : \beta \in S\}$ and compute q -approximation \mathbf{a}_j to $\underline{\alpha}_j$
- (9) **od**
- (10) Compute $\mathbf{t} \in \mathbb{Z}^n$ such that $|\mathbf{a}|_j \leq |\mathbf{a}_j|_j + 2^{-p}$ for $1 \leq j \leq m$, $j \neq i$, where $\mathbf{a} = \sum_{k=1}^n \mathbf{t}_k \mathbf{d}_k$ and $|\mathbf{a}|_i$ is minimal;
- (11) $\alpha := \sum_{k=1}^n \mathbf{t}_k \delta_k$;
- (12) $S := \text{FILL}(\mathcal{O}, \mathfrak{A}, \alpha_1, \dots, \alpha_{i-1}, \alpha, \alpha_{i+1}, \dots, \alpha_m)$;
- (13) **end procedure**

Again, Theorem 5.3.9 and Corollary 5.3.10 imply the correctness of our algorithm.

Proposition 5.3.19 *EXPAN (Algorithm 5.3.18) is correct.*

To complete the description of Algorithm 5.3.11 and Algorithm 5.3.18 we have to explain the implementation of step (10) in EXPAN. Obviously, we can not apply Theorem 5.3.13 without checking whether its condition are satisfied. But by the following lemma we are also able to check them.

Lemma 5.3.20 *There is an algorithm that given a basis $\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_n)$, of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$ and $c_1, \dots, c_n \in \mathbb{N}$ with $c_u = c_{u+t}$ for $s+1 \leq u \leq s+t$, decides whether there exists a lattice vector $\mathbf{v} \in \Lambda$ such that*

$$|\mathbf{v}|_u \leq c_u \text{ for } 1 \leq u \leq m. \quad (5.48)$$

If a vector \mathbf{v} satisfying (5.48) exists the algorithm determines such a vector and $\mathbf{y} \in \mathbb{Z}^n$ such that $\mathbf{v} = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i$. With $C = \prod_{u=1}^n c_u$ the running time of the algorithm is

$$n^{o(n)+n} (\log(nC \|\mathbf{A}\|_\infty))^5.$$

Proof. We use the same notation as in the proof of Theorem 5.3.13. Then with the help of the algorithm described in Corollary 5.2.7 we can compute a shortest vector \mathbf{b}' of Λ' and $\mathbf{y} \in \mathbb{Z}^n$ such that $\mathbf{b}' = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i$. If $\lambda_1(\Lambda') = \|\mathbf{b}'\|_2 \geq W/2$ then we can proceed as in the proof Theorem 5.3.13, compute the set T and check if T contains $(\mathbf{y}_1, \dots, \mathbf{y}_n) \in \mathbb{Z}^n$ such that $\mathbf{v} = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i$ is a solution of (5.48). If $\lambda_1(\Lambda') < W/2$ then clearly, \mathbf{b}' is a solution of (5.48). As in the proof of Theorem 5.3.13 the assertion follows. \square

The results of Theorem 5.3.13 and Lemma 5.3.20 are now used to solve the following problem: Let $\Lambda \subseteq \mathbb{Z}^n$ be a n -dimensional lattice, and fix $i \in \{1, \dots, m\}$. Suppose that

there is no $\mathbf{v} \neq \mathbf{0}$ in Λ satisfying $|\mathbf{v}|_i = 0$ and

$$|\mathbf{v}|_u \leq c_u \text{ for } 1 \leq u \leq m, u \neq i. \quad (5.49)$$

We are interested in finding a vector $\mathbf{v} \neq \mathbf{0}$ in Λ satisfying (5.49) with minimal $|\mathbf{v}|_i$. We apply the following strategy:

Algorithm 5.3.21

Input : a basis $(\mathbf{a}_1, \dots, \mathbf{a}_n)$ of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$; $i \in \mathbb{N}$, $1 \leq i \leq m$; $c_1, \dots, c_n \in \mathbb{N}$ with $c_u = c_{u+t}$ for $s+1 \leq u \leq s+t$
Output : $\mathbf{y} \in \mathbb{Z}^n$ such that $\mathbf{v} = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i \neq \mathbf{0}$, $|\mathbf{v}|_u \leq c_u$ for $1 \leq u \leq m$, $u \neq i$, where $|\mathbf{v}|_i \neq 0$ is minimal

- (1) $C_i := \prod_{u=1, u \neq i}^n c_u$;
- (2) $c_i := 2^t \det(\Lambda) / C_i$;
- (3) **while** (there exists $\mathbf{v} \in \Lambda$ with $|\mathbf{v}|_u \leq c_u$ for $1 \leq u \leq m$) **do**
- (4) $c_i := |\mathbf{v}|_i / 2$;
- (5) **od**
- (6) $c_i := c_i * 2$;
- (7) Compute the set T of all $\mathbf{y} \in \mathbb{Z}^n$ such that $\mathbf{v} = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i$ satisfies $|\mathbf{v}|_u \leq c_u$ for $1 \leq u \leq m$;
- (8) Find $\mathbf{y} \in T$ such that $|\mathbf{v}|_i \neq 0$ is minimal, where $\mathbf{v} = \sum_{i=1}^n \mathbf{y}_i \mathbf{a}_i$;

Lemma 5.3.22 *Algorithm 5.3.21 is correct. On input of a basis $\mathbf{A} \in \mathbb{Z}^{n \times n}$ of a n -dimensional lattice $\Lambda \subseteq \mathbb{Z}^n$, its running time is*

$$n^{o(n)+n} (\log(n \|\mathbf{A}\|_\infty))^6.$$

Proof. Let $\mathbf{v} \in \Lambda$. Since by Lemma 5.1.6 for $1 \leq i \leq m$ the volume of the body $\{\mathbf{w} \in \mathbb{R}^n : |\mathbf{w}|_u < c_u \text{ for } 1 \leq u \leq m, u \neq i, |\mathbf{w}|_i < |\mathbf{v}|_i\}$ is $2^s \pi^t C_i |\mathbf{v}|_i$ with $C_i = \prod_{u=1, u \neq i}^n c_u$, from Theorem 3.5.4 it follows that there is a nonzero solution \mathbf{v} of (5.49) with

$$|\mathbf{v}|_i \leq 2^t \det(\Lambda) / C_i.$$

We set $c_i = 2^t \det(\Lambda) / C_i$ and $C = \prod_{u=1}^n c_u = 2^t \det(\Lambda)$. From [43, Sect. 1.4] it follows that given \mathbf{A} we can compute $\det(\Lambda) = |\det(\mathbf{A})|$ in time $O(n^4 \log(n \|\mathbf{A}\|_\infty)^2)$. Thus, we can perform step (2) and (3) in time $O(\log(C) n^5 \log(n \|\mathbf{A}\|_\infty)^2)$.

In the steps (4)–(6) of the algorithm we search for a nonzero solution of (5.49) with $|\mathbf{v}|_i \leq c_i$. Then we substitute c_i by $|\mathbf{v}|_i / 2$ and repeat the search. In case of success we again replace c_i by $|\mathbf{v}|_i / 2$. This procedure is iterated until the search fails. Clearly, the search must fail for $c_i < 1$ and thus, the number of successful searches is $O(n \log \det(\Lambda))$.

Since at any time we have $\prod_{u=1}^n c_u \leq 2^t \det(\Lambda)$ the total time used performing the steps (1)–(6) is, by proposition Lemma 5.3.20,

$$n^{o(n)+n} (\log (n 2^t \det(\Lambda) \|\mathbf{A}\|_\infty))^5 O(n \log \det(\Lambda)),$$

which by (3.9) and Corollary 3.5.13 and some elementary properties of the o- and O-notation equals

$$n^{o(n)+n} (\log (n \|\mathbf{A}\|_\infty))^6 .$$

If the search fails, in step (6) we set $c_i = 2c_i$. Then the assumption of proposition Theorem 5.3.13 is satisfied and by the same arguments as above we see that the set T in step (7) can be determined in time

$$n^{o(n)+n} (\log (n \|\mathbf{A}\|_\infty))^5 .$$

While determining all those solutions we can also find \mathbf{y} with minimal corresponding $|\mathbf{v}|_i$. \square

Lemma 5.3.23 *Given \mathcal{O} , a reduced ideal \mathfrak{D} , a minimal set S containing 1 and $i \in \mathbb{N}$, $1 \leq i \leq m$, EXPAN computes the i -th expansion of S in time*

$$n^{O(n)} (\log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(1)} .$$

Proof. To prove this lemma we act in the same way as in the proof of of Theorem 5.2.8 or Lemma 5.3.17. The correctness of EXPAN was already shown in Proposition 5.3.19.

We use the same notation as in Algorithm 5.3.18. Again, we first concentrate our interest on the computations in the lattice Λ' with basis $\mathbf{D}' = (\mathbf{d}_1, \dots, \mathbf{d}_n)$. We let $\mathbf{C} = 2^q \mathbf{D}' \subseteq \mathbb{Z}^n$, and let Λ'' be the lattice with basis \mathbf{C} . For $1 \leq u \leq s$ let $c_u = 2^q(|\mathbf{a}_u|_u + 2^{-p})$, and for $s+1 \leq u \leq m$ let $c_u = c_{u+t} = 2^{2q}(|\mathbf{a}_u|_u + 2^{-p})$. Then we apply Lemma 5.3.22. Thus, step (10) of EXPAN can be performed by Algorithm 5.3.21 in time

$$n^{o(n)+n} (\log (n \|\mathbf{C}\|_\infty))^6 .$$

Now, by the same estimates as in the proof of Lemma 5.3.17 and since there is also a call up of the procedure FILL, the assertion follows. \square

Lemma 5.3.24 *On input of an order \mathcal{O} , a reduced ideal \mathfrak{C} of \mathcal{O} and a minimum α of \mathfrak{C} the algorithm NEIGHBORS (Algorithm 5.3.5) determines the set of all neighbors of α in time*

$$(n \log(\Delta))^{O(n)} (\text{size}(\alpha) + \log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(1)} .$$

Proof. By Proposition 5.3.6 we only have to prove that NEIGHBORS works in the stated running time. First, we note that the number of iterations of the for-loop from step (6) to step (15) is bounded by the number of minimal sets containing 1, which by Proposition 5.3.3 is at most $4^n(m \log(\Delta))^r$. In each iterations, there are m call ups of

the procedures EXPAN and COMP. Hence, the total time for performing that for-loop is

$$(n \log(\Delta))^{O(n)} (\log(\|\text{MT}(\Omega)\|_\infty) + \log(D))^{O(1)}$$

Clearly, that time also suffices to perform the steps (1)–(5). Therefore, now we only have to estimate the time that is needed in step (16) for computing the set N . To do this we need upper bounds on the binary size of the elements of N' . We shall use Lemma 3.5.25.

Clearly, each element β of the set N' is a neighbor of 1, and thus a minimum of the reduced ideal \mathfrak{D} . Thus, by Lemma 3.5.25, Corollary 5.1.9 and Lemma 5.1.11, we have

$$\text{size}(\beta) \leq n \log(\sqrt{n}\Delta 2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n) + 2n + \log(\Delta).$$

Hence, we can compute the set N in time

$$(n \log(\Delta))^{O(n)} (\text{size}(\alpha) + \log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(1)}.$$

This concludes the proof. □

Chapter 6

Binary Multiplicative Representations of Algebraic Numbers

6.1 Definitions and Preliminaries

By Lemma 3.5.25 the binary size of an algebraic number is polynomially bounded by its height. For our further computations it is more convenient to introduce a representation of algebraic numbers which is shorter and hence different from the standard representation. In the following sections, let \mathbb{F} be a number field of degree n and of signature (s, t) , and set $m = s + t$ and $r = s + t - 1$. Let \mathcal{O} be an order of \mathbb{F} , and for convenience, let $\Delta = |\Delta_{\mathcal{O}}|$. Furthermore, we assume that \mathcal{O} is given by the multiplication table $\text{MT}(\Omega)$, where $\Omega = (\omega_1, \dots, \omega_n)$ is a \mathbb{Z} -basis of \mathcal{O} .

A *multiplicative representation of an algebraic number* $\alpha \in \mathbb{F}$ is a pair

$$M = ((\beta_1, \dots, \beta_\ell), (e_1, \dots, e_\ell)),$$

where $\ell \in \mathbb{N}$ and $\beta_i \in \mathbb{F}$ and $e_i \in \mathbb{Z}$ for $1 \leq i \leq \ell$, such that

$$\alpha = \prod_{i=1}^{\ell} \beta_i^{e_i}.$$

The algebraic numbers β_i are given in standard and the rational integers e_i in binary representation.

For convenience, we write $\alpha = ((\beta_1, \dots, \beta_k), (e_1, \dots, e_\ell))$ if $((\beta_1, \dots, \beta_k), (e_1, \dots, e_\ell))$ is a multiplicative representation of α . This shall always be clear by the context.

Clearly, the multiplicative representation of an algebraic number is not unique. Before we describe a way how to determine multiplicative representations with special properties we explain how to compute with multiplicative representations. If $M = ((\beta_1, \dots, \beta_\ell), (e_1, \dots, e_\ell))$ and $N = ((\gamma_1, \dots, \gamma_k), (f_1, \dots, f_k))$ ($k \in \mathbb{N}$) are multiplicative representations of the algebraic numbers μ and ν then

$$MN = ((\beta_1, \dots, \beta_\ell, \gamma_1, \dots, \gamma_k), (e_1, \dots, e_\ell, f_1, \dots, f_k))$$

is a multiplicative representation of $\mu\nu$. It is also easy to see that for $z \in \mathbb{Z}$

$$M^z = ((\beta_1, \dots, \beta_\ell), (ze_1, \dots, ze_\ell))$$

is a multiplicative representation of the number μ^z and

$$\frac{M}{N} = MN^{-1}$$

is a multiplicative representation of μ/ν .

Proposition 6.1.1 *There is an algorithm that given an order \mathcal{O} of \mathbb{F} by a multiplication table $\text{MT}(\Omega)$ of a \mathbb{Z} -basis Ω , a multiplicative representation M of $\alpha \in \mathbb{F}$ and a natural number q determines a q -approximation to the logarithm vector $\text{Log } \alpha$ in time*

$$O\left(n^8 \text{size}(M) (q + \text{size}(M) + (\log(n \|\text{MT}(\Omega)\|_\infty))^3) \cdot (\log(q + \text{size}(M) + \log \|\text{MT}(\Omega)\|_\infty))^2\right).$$

Proof. Let $M = ((\beta_1, \dots, \beta_k), (e_1, \dots, e_k))$. Then we have

$$\text{Log } \alpha = \sum_{i=1}^k e_i \text{Log } \beta_i.$$

Therefore, if \mathbf{b}_i is an approximation of precision $p = q + 2k + \lceil \log(e_i) \rceil$ to $\text{Log } \beta_i$ for $1 \leq i \leq k$ then by Proposition 4.1.8

$$\mathbf{a} = \sum_{i=1}^k e_i \mathbf{b}_i$$

is a q -approximation to $\text{Log } \alpha$. By Theorem 4.3.2 we can compute the vectors $\mathbf{b}_1, \dots, \mathbf{b}_k$ in time

$$O\left(kn^8 (q + 2k + 2 + \text{size}(M) + (\log(n \|\text{MT}(\Omega)\|_\infty))^3) \cdot (\log(q + 2k + 2 + \text{size}(M) + \log \|\text{MT}(\Omega)\|_\infty))^2\right).$$

Clearly, the same running time suffices to compute \mathbf{a} . \square

The algorithm described in Proposition 6.1.1 is called LAPPROX, and for a multiplicative representation $M = ((\beta_1, \dots, \beta_k), (e_1, \dots, e_k))$ of $\alpha \in \mathbb{F}$, and a natural number q we denote by $\text{LAPPROX}(\mathcal{O}, M, q)$ the approximation of precision q to $\text{Log } \alpha$ that is determined by LAPPROX on the corresponding input.

In the following, we shall work with multiplicative representations of a special shape.

Definition 6.1.2 A multiplicative representation $((\beta_1, \dots, \beta_k), (e_1, \dots, e_k))$ ($k \in \mathbb{N}$) of a minimum α of an ideal \mathfrak{A} of \mathcal{O} is called *binary* if $e_i = 2^{k-i}$ for all $1 \leq i \leq k$ and if $((\beta_1, \dots, \beta_i), (e_1, \dots, e_i))$ is a minimum of \mathfrak{A} for $1 \leq i \leq k$. Such a binary multiplicative representation is also written as $M = (\beta_1, \dots, \beta_k)$. The binary size of M is defined to be $\text{size}(M) = \sum_{i=1}^k \text{size}(\beta_k)$.

For convenience, we often write $\alpha = (\beta_1, \dots, \beta_k)$ if $(\beta_1, \dots, \beta_k)$ is a binary representation of α . For convenience, we shall also sometimes identify the algebraic number α given in standard representation with a binary representation of the form (α) .

6.2 Finding Minima Close to a Given Point

Definition 6.2.1 Let \mathfrak{A} be an ideal of \mathcal{O} , and let $\mathbf{s} \in \mathbb{R}^r$. Let $c \in \mathbb{R}_{>0}$. A minimum α of \mathfrak{A} is called *c-close to \mathbf{s}* if $\|\mathbf{s} - \text{Log } \alpha\|_\infty < (\log(\Delta))/4 + c$.

We now present the algorithm CLOSE which on input of \mathcal{O} , a reduced invertible ideal \mathfrak{A} , and $\mathbf{s} \in \mathbb{Q}^r$ finds the binary multiplicative representation of a minimum α of \mathfrak{A} which is (3/4)-close to \mathbf{s} . Our algorithm is a generalization of the algorithm CLOSE presented in [3, Sect. 6] which solves the corresponding problem for orders of real quadratic number fields. It is based on our work in [60, Sect. 4].

The main tool, which is used in CLOSE, is the procedure DOUBLE, which on input of $\mathbf{t} \in \mathbb{Q}^r$ and a binary multiplicative representation $(\beta_1, \dots, \beta_{k-1})$ of a minimum of \mathfrak{A} which is (3/4)-close to \mathbf{t} , determines a binary multiplicative representation $(\beta_1, \dots, \beta_{k-1}, \beta_k)$ of a minimum that is (3/4)-close to $2\mathbf{t}$, where $H(\beta_k) \leq 4\Delta^{(3/4)(m+2)}$. Using DOUBLE as a subroutine, CLOSE works as follows.

Algorithm 6.2.2 (CLOSE)

Input : an order \mathcal{O} ; a reduced invertible ideal \mathfrak{A} of \mathcal{O} ; $\mathbf{s} \in \mathbb{Q}^r$
Output : the binary multiplicative representation $(\beta_1, \dots, \beta_\ell)$ of a minimum α of \mathfrak{A} which is (3/4)-close to \mathbf{s} , where $\ell = \lceil \log \|\mathbf{s}\|_\infty \rceil + 1$, and $H(\beta_i) \leq (4\Delta^{3/4})^{m+1}$ for $1 \leq i \leq \ell$

- (1) **procedure** CLOSE ($\mathcal{O}, \mathfrak{A}, \mathbf{s}$)
- (2) $\ell := \max\{0, \lceil \log \|\mathbf{s}\|_\infty \rceil + 1\}$;
- (3) $\beta_0 := 1$;
- (4) $\mathbf{t} := \mathbf{s}/2^\ell$;
- (5) **for** ($k := 1$ **to** ℓ **step** 1) **do**
- (6) $(\beta_1, \dots, \beta_k) := \text{DOUBLE}(\mathcal{O}, \mathfrak{A}, \mathbf{t}, (\beta_0, \dots, \beta_{k-1}))$;
- (7) $\mathbf{t} := 2\mathbf{t}$;
- (8) **od**
- (9) **end procedure**

Proposition 6.2.3 *Algorithm 6.2.2 is correct provided DOUBLE is correct.*

Proof. The assertion follows from the fact that in step (3) of the algorithm we have $\|\mathbf{t}\|_\infty \leq 1$. Thus, the minimum α_0 is close to $\mathbf{0}$. \square

Next we present the procedure DOUBLE. It uses the subroutine NEAREST which on input of an ideal \mathfrak{B} of \mathcal{O} and a vector $\mathbf{u} \in \mathbb{Q}^r$ finds a minimum β in \mathfrak{B} which is (1/4)-close to \mathbf{u} . Also note, that we use the algorithm LAPPROX introduced in section 6.1.

Algorithm 6.2.4 (DOUBLE)

Input : an order \mathcal{O} ; a reduced invertible ideal \mathfrak{A} of \mathcal{O} ; $\mathbf{t} \in \mathbb{Q}^r$;
the binary multiplicative representation $(\beta_1, \dots, \beta_{k-1})$ of a
minimum α of \mathfrak{A} which is $(3/4)$ -close to \mathbf{t}
Output : the binary multiplicative representation $(\beta_1, \dots, \beta_{k-1}, \beta_k)$
of a minimum of \mathfrak{A} which is $(3/4)$ -close to $2\mathbf{t}$, where β_k is
a minimum of $(1/\alpha^2)\mathfrak{A}$ and $H(\beta_k) \leq (4\Delta^{3/4})^{m+1}$

- (1) **procedure** DOUBLE ($\mathcal{O}, \mathfrak{A}, \mathbf{t}, (\beta_1, \dots, \beta_{k-1})$)
- (2) $\mathfrak{B} := (1/\alpha^2)\mathfrak{A}$;
- (3) $\mathbf{u} := 2(\mathbf{t} - \text{LAPPROX}(\mathcal{O}, (\beta_1, \dots, \beta_{k-1}), 3))$;
- (4) $\beta_k := \text{NEAREST}(\mathcal{O}, \mathfrak{B}, \mathbf{u})$;
- (5) **end procedure**

Proposition 6.2.5 *If NEAREST works correctly then Algorithm 6.2.4 is correct.*

Proof. Clearly, β_k is a minimum of $(1/\alpha^2)\mathfrak{A}$. Now, let $\alpha' = (\beta_1, \dots, \beta_{k-1}, \beta_k)$. Then we have $\alpha' = \alpha^2\beta_k$, and since β_k is a minimum of $\mathfrak{B} = (1/\alpha^2)\mathfrak{A}$ it follows from Proposition 5.1.3 that α' is a minimum of \mathfrak{A} . Let $\mathbf{a} = \text{LAPPROX}(\mathcal{O}, (\beta_1, \dots, \beta_{k-1}), 3)$. Then we have

$$\begin{aligned} \|2\mathbf{t} - \log \alpha'\|_\infty &= \|2\mathbf{t} - 2\text{Log } \alpha - \text{Log } \beta_k\|_\infty \\ &\leq \|2(\mathbf{t} - \mathbf{a}) - \text{Log } \beta_k\|_\infty + 2\|\mathbf{a} - \text{Log } \alpha\|_\infty \\ &\leq \frac{\log(\Delta)}{4} + \frac{1}{4} + \frac{1}{8} \leq \frac{\log(\Delta) + 3}{4}. \end{aligned}$$

Thus, α' is $(3/4)$ -close to $2\mathbf{t}$. Next, we estimate $H(\beta_k)$. Since α is $(3/4)$ -close to \mathbf{t} we know that

$$\|\mathbf{t} - \text{Log } \alpha\|_\infty < \frac{\log(\Delta) + 3}{4}.$$

Hence, by the properties of LAPPROX we have

$$\|\mathbf{u}\|_\infty = \|2(\mathbf{t} - \mathbf{a})\|_\infty \leq 2\|\mathbf{t} - \text{Log } \alpha\|_\infty + \frac{1}{4} \leq \frac{\log(\Delta)}{2} + \frac{7}{2}, \quad (6.1)$$

and since β_k is $(1/4)$ -close to \mathbf{u} , (6.1) implies

$$\|\text{Log } \beta_k\|_\infty \leq \|\text{Log } \beta_k - \mathbf{u}\|_\infty + \|\mathbf{u}\|_\infty < \frac{\log(\Delta) + 1}{4} + \frac{\log(\Delta)}{2} + \frac{7}{2} = \frac{3\log(\Delta)}{4} + 2.$$

Thus we have for $1 \leq i \leq r$

$$\frac{1}{4\Delta^{3/4}} < |\beta_k|_i < 4\Delta^{3/4}.$$

Applying Proposition 3.1.7, Proposition 5.1.8 and Corollary 5.1.10 we obtain

$$|\beta_k|_m \leq \frac{N_{\mathcal{O}}(\mathfrak{B})\sqrt{\Delta}}{\prod_{i=1}^r |\beta_k|_i} \leq \frac{N_{\mathcal{O}}(\mathfrak{A})\sqrt{\Delta} \left(4\Delta^{3/4}\right)^r}{(N_{\mathbb{F}/\mathbb{Q}}(\alpha))^2} \leq \frac{\sqrt{\Delta} \left(4\Delta^{3/4}\right)^r}{(N_{\mathcal{O}}(\mathfrak{A}))^2} \leq \left(4\Delta^{3/4}\right)^{m+1}.$$

This implies the assertion. \square

The algorithm NEAREST finds, on input of an ideal \mathfrak{B} of \mathcal{O} and a vector $\mathbf{u} \in \mathbb{Q}^r$, a minimum β in \mathfrak{B} which is $(1/4)$ -close to \mathbf{u} . It uses the subroutine TARGET that on input of a reduced ideal \mathfrak{C} and a vector $\mathbf{v} \in \mathbb{Q}^r$ finds a minimum μ of \mathfrak{C} that is $(1/16)$ -close to \mathbf{v} . NEAREST also uses REDUCE (see Algorithm 5.2.3) that on input of \mathcal{O} and an ideal \mathfrak{B} determines a minimum γ of \mathfrak{B} and the ideal $\mathfrak{C} = (1/\gamma)\mathfrak{B}$.

Algorithm 6.2.6 (NEAREST)

Input : An order \mathcal{O} ; an ideal \mathfrak{B} of \mathcal{O} ; $\mathbf{u} \in \mathbb{Q}^r$
Output : a minimum β of \mathfrak{B} which is $(1/4)$ -close to \mathbf{u}

- (1) **procedure** NEAREST ($\mathcal{O}, \mathfrak{B}, \mathbf{u}$)
- (2) $(\mathfrak{C}, \gamma) := \text{REDUCE}(\mathcal{O}, \mathfrak{B});$
- (3) $\mathbf{v} := \mathbf{u} - \text{LAPPROX}(\mathcal{O}, (\gamma), 5);$
- (4) $\mu := \text{TARGET}(\mathcal{O}, \mathfrak{C}, \mathbf{v});$
- (5) $\beta := \gamma\mu;$
- (6) **end procedure**

Analogously to Proposition 6.2.5 we obtain

Proposition 6.2.7 *If TARGET works correctly then NEAREST is correct.*

Now, we have to describe the procedure TARGET. This procedure uses the subroutine COLLECT that on input of a reduced invertible ideal \mathfrak{C} and positive rational numbers B_1, \dots, B_m finds a set M that contains all minima ν of \mathfrak{C} with $|\nu|_i \leq B_i$ for $1 \leq i \leq m$.

Algorithm 6.2.8 (TARGET)

Input : An order \mathcal{O} ; a reduced ideal \mathfrak{C} of \mathcal{O} ; $\mathbf{v} \in \mathbb{Q}^r$
Output : a minimum μ of \mathfrak{C} which is $(1/16)$ -close to \mathbf{v}

- (1) **procedure** TARGET ($\mathcal{O}, \mathfrak{C}, \mathbf{v}$)
- (2) $M := \text{COLLECT}(\mathcal{O}, \mathfrak{C}, 2^{2v_1+1}\sqrt{\Delta}, \dots, 2^{2v_r+1}\sqrt{\Delta}, \Delta^{m/2}2^{2m\|\mathbf{v}\|_\infty});$
- (3) For μ choose $\nu \in M$ such that $\|\mathbf{v} - \text{LAPPROX}(\mathcal{O}, (\nu), 6)\|_\infty$
 is minimal;
- (4) **end procedure**

Proposition 6.2.9 *Algorithm 6.2.8 is correct provided that COLLECT is correct.*

Proof. We use the notation of Algorithm 6.2.8. Let μ be a minimum of \mathfrak{C} that is $(1/32)$ -close to \mathbf{v} . By Lemma 5.1.14 such a minimum exists. Then we first show that $\mu \in M$. By our assertion we have

$$\|\text{Log } \mu - \mathbf{v}\|_\infty \leq \frac{\log(\Delta)}{4} + \frac{1}{32},$$

and therefore for $1 \leq i \leq r$

$$\frac{2^{\log(e)\mathbf{v}_i}}{2^{\log(e)(\log(\Delta)/4+1/32)}} \leq |\mu|_i \leq 2^{\log(e)(\mathbf{v}_i+\log(\Delta)/4+1/32)} \leq 2^{2\mathbf{v}_i+1} \Delta^{\frac{1}{2}}, \quad (6.2)$$

where e is the Euler constant. From Proposition 3.1.7 and Corollary 5.1.10 we obtain that

$$\prod_{i=1}^m |\mu|_i \leq \sqrt{\Delta}. \quad (6.3)$$

Thus combining (6.2) and (6.3) and some simple estimations yield

$$|\mu|_m \leq \sqrt{\Delta} 2^{(m-1)\log(e)(\log(\Delta)/4+1/32)} 2^{(m-1)2\|\mathbf{v}\|_\infty} \leq \Delta^{\frac{m}{2}} 2^{2m\|\mathbf{v}\|_\infty}.$$

This implies that $\mu \in M$.

Next, we have to show that each $\nu \in M$ such that $\|\mathbf{v} - \text{LAPPROX}(\mathcal{O}, (\nu), 6)\|_\infty$ is minimal, also is $(1/16)$ -close to \mathbf{v} . Clearly, since M contains a $(1/32)$ -close minimum μ we have

$$\|\mathbf{v} - \text{LAPPROX}(\mathcal{O}, (\nu), 6)\|_\infty \leq \|\mathbf{v} - \text{LAPPROX}(\mathcal{O}, (\mu), 6)\|_\infty \leq \frac{\log(\Delta)}{4} + \frac{3}{64}.$$

Thus we obtain

$$\|\mathbf{v} - \text{Log } \nu\|_\infty \leq \frac{\log(\Delta)}{4} + \frac{1}{16},$$

which proves the assertion. \square

Let us just remark that the bounds for the elements of the set M are rather crude. But because we shall later give the running times of the procedures of this section in terms using O-notation any more subtle analysis would yield no better result.

Before we explain how the routine COLLECT works we quote the following lemma from [6, Sect. 10].

Lemma 6.2.10 *Let μ and μ' be minima of an ideal \mathfrak{A} . Then there exists a sequence $\mu_1 = \mu, \mu_2, \mu_3, \dots, \mu_{\ell-1}, \mu_\ell = \mu'$ ($\ell \in \mathbb{N}$) of minima of \mathfrak{A} such that μ_{i+1} is a neighbor of μ_i for $1 \leq i \leq \ell - 1$ and $|\mu_j|_i < \max\{|\mu|_i, |\mu'|_i\}$ for $1 \leq j \leq \ell$ and $1 \leq i \leq m$.*

By the above lemma we obtain the following description for COLLECT that is similar to [6, Algorithm 10.1]. Note that the procedure COLLECT uses the procedure NEIGHBORS that on input of a reduced ideal \mathfrak{C} and a minimum α on \mathfrak{C} finds the set of all neighbors of α .

Algorithm 6.2.11 (COLLECT)

Input : An order \mathcal{O} ; a reduced invertible ideal \mathfrak{C} of \mathcal{O} ; $B_1, \dots, B_m \in \mathbb{Q}_{>0}$
Output : a set M containing all minima ν of \mathfrak{C} with $|\nu|_i \leq B_i$ for $1 \leq i \leq m$

```

(1) procedure COLLECT ( $\mathcal{O}, \mathfrak{C}, B_1, \dots, B_m$ )
(2)   for ( $i := 1$  to  $m$  step 1) do
(3)      $B_i := B_i + 2^{-10}$ ;
(4)   od
(5)    $M := \emptyset$ ;
(6)   for (every  $\nu \in \text{NEIGHBORS}(\mathcal{O}, \mathfrak{C}, 1)$ ) do
(7)     if ( $\text{VAPPR}(\mathcal{O}, \nu, i, 10) \leq B_i$  for all  $1 \leq i \leq m$ ) then
(8)        $M := M \cup \{\nu\}$ ;
(9)     fi
(10)  od
(11)  for (every  $\alpha \in M$ ) do
(12)    for (every  $\nu \in \text{NEIGHBORS}(\mathcal{O}, \mathfrak{C}, \alpha)$ ) do
(13)      if ( $(\text{VAPPR}(\mathcal{O}, \nu, i, 10) \leq B_i$  for all  $1 \leq i \leq m$ ) and ( $\nu \notin M$ ))
(14)        then
(15)           $M := M \cup \{\nu\}$ ;
(16)        fi
(17)      od
(18)    od
end procedure

```

Proposition 6.2.12 *COLLECT (Algorithm 6.2.11) is correct. Given an order \mathcal{O} , a reduced invertible ideal \mathfrak{C} of \mathcal{O} , and numbers $B_1, \dots, B_m \in \mathbb{Q}_{>0}$, the algorithm finds a set M containing all minima ν of \mathfrak{C} with $|\nu|_i \leq B_i$ for $1 \leq i \leq m$ in time*

$$(\log(B) + n + \log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(n)},$$

where $B = \max\{B_1, \dots, B_n, B_1^2, \dots, B_n^2\} + 2^{-9}$. The set M contains only minima of \mathfrak{C} , it contains at most

$$4^n (\log(\Delta) + m|\ln(B)|)^r$$

elements, and for each $\mu \in M$ we have $|\mu|_i \leq B_i + 2^{-9}$ for $1 \leq i \leq m$.

Proof. We use the notation of Algorithm 6.2.11. Then from Lemma 6.2.10 it follows that the set M computed by COLLECT contains all minima ν of \mathfrak{C} with $\text{VAPPR}(\mathcal{O}, \nu, i, 10) \leq B_i + 2^{-10}$ for $1 \leq i \leq m$, and therefore all minima ν of \mathfrak{C} with $|\nu|_i \leq B_i$ for $1 \leq i \leq m$. Clearly, for each $\mu \in M$ we have $|\mu|_i \leq B_i + 2^{-9}$ for $1 \leq i \leq m$. This shows that Algorithm 6.2.11 is correct.

Clearly, the number of iterations of the for-loop from step (9) to step (15) is bounded by the size of the set M . If $\text{VAPPR}(\mathcal{O}, \nu, i, 10) \leq B_i + 2^{-10}$ then $|\nu|_i \leq B_i + 2^{-9}$. Hence,

by Corollary 5.1.15 the set M contains at most

$$4^n \left(\log(\Delta) + \sum_{j=1}^m |\ln(B_j + 2^{-9})| \right)^r$$

elements. The number of iterations of the two other for-loops is by Corollary 5.1.16 not greater than

$$4^n (m \log(\Delta))^r .$$

Thus, there are at most

$$4^{2n} n \left(\log(\Delta) + \sum_{j=1}^m |\ln(B_j + 2^{-9})| \right)^r (m \log(\Delta))^r \quad (6.4)$$

different calls of the procedures VAPPR and NEIGHBORS. By Corollary 5.1.15 the binary size of each minimum α involved in a call of VAPPR or NEIGHBORS is at most

$$n \log \left(\sqrt{n\Delta} B \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \log(\sqrt{\Delta}) .$$

Hence, comparing the running times given in Theorem 4.3.3 and Lemma 5.3.24 we observe that the time used for each call of VAPPR or NEIGHBORS is bounded by

$$(n \log(\Delta))^{O(n)} (\log(B) + \log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(1)} . \quad (6.5)$$

Multiplying (6.4) and (6.5) shows that the total running time of COLLECT is

$$(n \log(\Delta))^{O(n)} (\log(B) + \log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(n)} . \quad \square$$

Proposition 6.2.12 allows us to estimate the running time of the other algorithms described in this section.

Lemma 6.2.13 *On input of an order \mathcal{O} , a reduced ideal \mathfrak{C} of \mathcal{O} and a vector $\mathbf{v} \in \mathbb{Q}^r$, the algorithm TARGET (Algorithm 6.2.8) finds a minimum μ of \mathfrak{C} which is $(1/16)$ -close to \mathbf{v} in time*

$$(n + \log(\Delta) + \|\mathbf{v}\|_\infty + \|\text{MT}(\Omega)\|_\infty)^{O(n)} .$$

Moreover, we have

$$\text{size}(\mu) \leq n \log \left(\sqrt{n\Delta} \Delta^m 2^{(4m\|\mathbf{v}\|_\infty+2)} \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \log(\sqrt{\Delta}) .$$

Proof. The running time of the call of the procedure COLLECT in step (2) of the algorithm is

$$(n + \|\mathbf{v}\|_\infty + \log(\|\text{MT}(\Omega)\|_\infty) + \log(\Delta))^{O(n)},$$

as can easily be seen by applying Proposition 6.2.12. In step (3) we have to start LAPPROX for each $\mu \in M$. Again by Proposition 6.2.12, the set M contains at most

$$4^n (\log(\Delta) + m |\ln(\Delta^{m/2} 2^{2m\|\mathbf{v}\|_\infty + 1})|)^r \quad (6.6)$$

elements. The same lemma implies that for each $\mu \in M$ we have $|\mu|_i \leq \Delta^{m/2} 2^{2m\|\mathbf{v}\|_\infty + 1} + 2^{-9}$ for $1 \leq i \leq m$. Hence, we obtain from Corollary 5.1.15

$$\text{size}(\mu) \leq n \log \left(\sqrt{n\Delta} \Delta^m 2^{4m\|\mathbf{v}\|_\infty + 2} \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \log(\sqrt{\Delta}). \quad (6.7)$$

Thus, by Proposition 6.1.1 and (6.6) and (6.7), we can compute all needed approximations in step (3) in time

$$(n + \log(\Delta) + \|\mathbf{v}\|_\infty + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

This concludes the proof. \square

Lemma 6.2.14 *On input of an order \mathcal{O} , an ideal \mathfrak{B} of \mathcal{O} and a vector $\mathbf{u} \in \mathbb{Q}^r$, the algorithm NEAREST (Algorithm 6.2.6) finds a minimum β of \mathfrak{B} which is $(1/4)$ -close to \mathbf{u} in time*

$$(n + \log(\Delta) + \|\mathbf{u}\|_\infty + \text{size}(\mathfrak{B}) + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

Proof. By Theorem 5.2.8 the algorithm REDUCE determines a minimum γ of \mathfrak{B} , such that

$$|\gamma|_i \leq 2\sqrt{n} \left(\sqrt{\Delta} N_{\mathcal{O}}(\mathfrak{B}) \right)^{1/n} \quad (6.8)$$

for $1 \leq i \leq m$, and the reduced ideal $\mathfrak{C} = (1/\gamma)\mathfrak{B}$ in time

$$n^{o(n) + \frac{n}{2}} (\log(\Delta) + \log \|\text{MT}(\Omega)\|_\infty + \text{size}(\mathfrak{B}))^{O(1)}.$$

By Proposition 3.1.7 and Lemma 3.4.15 we have

$$\prod_{i=1}^m |\gamma|_i \geq N_{\mathcal{O}}(\mathfrak{B}),$$

and therefore for $1 \leq i \leq m$

$$|\gamma|_i \geq \frac{N_{\mathcal{O}}(\mathfrak{B})}{\left(2\sqrt{n} \left(\sqrt{\Delta} N_{\mathcal{O}}(\mathfrak{B}) \right)^{\frac{1}{n}} \right)^{m-1}}.$$

This implies that

$$\begin{aligned} \ln(N_{\mathcal{O}}(\mathfrak{B})) - (m-1) \left(1 + \frac{1}{2} \ln(n) + \frac{1}{n} \left(\frac{1}{2} \ln(\Delta) + \ln(N_{\mathcal{O}}(\mathfrak{B})) \right) \right) \\ \leq \ln |\gamma|_i \leq 1 + \frac{1}{2} \ln(n) + \frac{1}{n} \left(\frac{1}{2} \ln(\Delta) + \ln(N_{\mathcal{O}}(\mathfrak{B})) \right), \end{aligned}$$

and

$$\|\mathbf{u} - \text{LAPPROX}(\mathcal{O}, (\gamma), 5)\|_{\infty} \leq \|\mathbf{u}\|_{\infty} + |\ln(N_{\mathcal{O}}(\mathfrak{B}))| + n(1 + \ln(n) + \ln(\Delta)) + 1.$$

Thus, by Lemma 6.2.13 and Corollary 3.4.13 step (4) in NEAREST has running time

$$(n + \log(\Delta) + \|\mathbf{u}\|_{\infty} + \text{size}(\mathfrak{B}) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)}.$$

By Lemma 3.5.25, Corollary 3.4.13 and Lemma 6.2.13 this time is also sufficient to compute the product $\gamma\mu$ in step (5) of the algorithm. This concludes the proof. \square

In order to estimate the running time of the algorithm DOUBLE we need

Lemma 6.2.15 *There is an algorithm that given an order \mathcal{O} , an invertible ideal \mathfrak{A} of \mathcal{O} , and the binary multiplicative representation M of a minimum α of \mathfrak{A} determines the ideal $(1/\alpha)\mathfrak{A}$ in time*

$$(n + \log(\Delta) + \text{size}(M) + \text{size}(\mathfrak{A}))^{O(1)}.$$

Proof. Suppose that $M = (\beta_1, \dots, \beta_k)$ ($k \in \mathbb{N}$). Then the algorithm works as follows.

Input : an order \mathcal{O} ; an invertible ideal \mathfrak{A} of \mathcal{O} ; the binary multiplicative representation $(\beta_1, \dots, \beta_k)$ of a minimum α of \mathfrak{A}
Output : the ideal $\mathfrak{B} = (1/\alpha^2)\mathfrak{A}$

(1) $\mathfrak{A}_1 := (1/\beta_1)\mathfrak{A}$;
(2) **for** ($i := 2$ **to** k **step 1**) **do**
(3) $\mathfrak{A}_i := (1/\beta_i)\mathfrak{A}_{i-1}^2\mathfrak{A}^{-1}$
(4) **od**

Since for $1 \leq i \leq k$ we have

$$\mathfrak{A}_i = \left(\frac{1}{\prod_{j=1}^i \beta_j^{2^{i-j}}} \right) \mathfrak{A} = \left(\frac{1}{(\beta_1, \dots, \beta_i)} \right) \mathfrak{A},$$

the algorithm is correct. Next, we have to estimate the running time of the algorithm. From Lemma 3.1.2 and Lemma 3.4.7 it follows that the ideal \mathfrak{A}_1 can be computed in time $(\text{size}(\mathfrak{A}) + \text{size}(\beta_1))^{O(1)}$. By the assertions of the lemma the ideals \mathfrak{A}_i are reduced for $1 \leq i \leq k$. Hence by Corollary 5.1.12 their binary size is at most $(n^2 + 1) \log(\sqrt{\Delta})$. Thus applying Lemma 3.1.2 and Lemma 3.4.7 it is easy to see that the steps (2)–(4) can be performed in time $((n^2 + 1) \log(\sqrt{\Delta}) + \text{size}(M) + \text{size}(\mathfrak{A}))^{O(1)}$. \square

Corollary 6.2.16 *There is an algorithm that given an order \mathcal{O} , an invertible ideal \mathfrak{A} of \mathcal{O} , and the binary multiplicative representation M of a minimum α of \mathfrak{A} determines the ideal $(1/\alpha^2)\mathfrak{A}$ in time*

$$(n + \log(\Delta) + \text{size}(M) + \text{size}(\mathfrak{A}))^{O(1)}.$$

Proof. We compute the ideal

$$\mathfrak{B} := ((1/\alpha)\mathfrak{A})^2 \mathfrak{A}^{-1}.$$

Then we have $\mathfrak{B} = (1/\alpha^2)\mathfrak{A}$ and the assertion immediately follows from Lemma 6.2.15. \square

Lemma 6.2.17 *On input of an order \mathcal{O} , a reduced invertible ideal \mathfrak{A} of \mathcal{O} , $\mathfrak{t} \in \mathbb{Q}^r$, and a binary multiplicative representation $M = (\beta_1, \dots, \beta_{k-1})$ of a minimum α of \mathfrak{A} which is $(3/4)$ -close to \mathfrak{t} , the algorithm *DOUBLE* finds the binary multiplicative representation $(\beta_1, \dots, \beta_{k-1}, \beta_k)$ of a minimum of \mathfrak{A} which is $(3/4)$ -close to $2\mathfrak{t}$, where $H(\beta_k) \leq (4\Delta^{3/4})^{m+1}$ in time*

$$(n + \log(\Delta) + \text{size}(M) + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

Proof. By Corollary 6.2.16 and Corollary 5.1.12 we can compute the ideal \mathfrak{B} in step (2) of the algorithm in time

$$(n + \log(\Delta) + \text{size}(M) + \text{size}(\mathfrak{A}))^{O(1)} = (n + \log(\Delta) + \text{size}(M))^{O(1)}.$$

Clearly, the same term also describes the binary size of \mathfrak{B} . By Proposition 6.1.1 we can compute \mathfrak{u} in step (3) in time

$$(n + \log(\Delta) + \log(\text{MT}(\Omega)) + \text{size}(M))^{O(1)},$$

and by (6.1) we know that

$$\|\mathfrak{u}\|_\infty \leq \frac{\log(\Delta)}{2} + \frac{7}{2}.$$

Thus, from Lemma 6.2.14 it follows that the running time of step (4) is

$$(n + \log(\Delta) + \text{size}(M) + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

This implies the assertion. \square

Before we prove an upper bound for the running time of *CLOSE* we need

Lemma 6.2.18 *Using the notation of Algorithm 6.2.4 and Lemma 6.2.17 we have*

$$\text{size}(\beta_k) \leq n \log \left(\Delta^{\frac{n-1}{2}} \sqrt{n} \left(8\Delta^{\frac{3}{2}} \right)^{m+1} \frac{2^n (n^3 \|\text{MT}(\Omega)\|_\infty 2^{2n})^n}{\sqrt{\Delta}} \right) + 2n + \frac{n-1}{2} \log(\Delta).$$

Proof. To be able to apply Lemma 3.5.25 we have to find upper bounds on $H(\beta_k)$ and $d(\mathfrak{B})$. Clearly, we have

$$H(\beta_k) \leq (4\Delta^{3/4})^{m+1}. \quad (6.9)$$

Hence, to apply Lemma 3.5.25 we have to find an upper bound of $d(\mathfrak{B})$. We write $\mathfrak{B} = \mathfrak{B}_1 \mathfrak{B}_2$ with $\mathfrak{B}_1 = (1/\alpha)\mathfrak{A}$ and $\mathfrak{B}_2 = (1/\alpha)\mathcal{O}$. Then we have $d(\mathfrak{B}) \leq d(\mathfrak{B}_1)d(\mathfrak{B}_2)$. Since α is a minimum of \mathfrak{A} the ideal \mathfrak{B}_1 is reduced. Applying Lemma 5.1.11 we see that $d(\mathfrak{B}_1) \leq \sqrt{\Delta}$. Next, we observe that $d(\mathfrak{A})\alpha \in \mathcal{O}$. Thus Lemma 3.2.8 implies that

$$\frac{N_{\mathbb{F}/\mathbb{Q}}(d(\mathfrak{A})\alpha)}{d(\mathfrak{A})\alpha} = d(\mathfrak{A})^{n-1}N_{\mathbb{F}/\mathbb{Q}}(\alpha) \left(\frac{1}{\alpha} \right) \in \mathcal{O},$$

and therefore $d(\mathfrak{B}_2) \leq d(\mathfrak{A})^{n-1}N_{\mathbb{F}/\mathbb{Q}}(\alpha)$. Since \mathfrak{A} is reduced, we finally obtain from Lemma 5.1.11 and Proposition 5.1.8

$$d(\mathfrak{B}) \leq \Delta^{\frac{n-1}{2}}. \quad (6.10)$$

Combining Lemma 3.5.25 with (6.9) and (6.10) yields the assertion. \square

Lemma 6.2.19 *Given an order \mathcal{O} , a reduced invertible ideal \mathfrak{A} of \mathcal{O} , and a vector $\mathbf{s} \in \mathbb{Q}^r$ the algorithm CLOSE determines the binary multiplicative representation $(\beta_1, \dots, \beta_k)$ of a minimum α of \mathfrak{A} which is $(3/4)$ -close to \mathbf{s} , where $k = \lceil \log \|\mathbf{s}\|_\infty \rceil + 1$ and $H(\beta_i) \leq (4\Delta^{3/4})^{m+1}$ for $1 \leq i \leq k$ in time*

$$(n + \log(\Delta) + \log \|\mathbf{s}\|_\infty + \|\text{MT}(\Omega)\|_\infty)^{O(n)}$$

Proof. Clearly, the number of iterations of the for-loop from step (5) to step (8) is $\ell = \max\{0, \lceil \log \|\mathbf{s}\|_\infty \rceil + 1\}$. From Lemma 6.2.17 we know that the running time of the k -th iteration ($1 \leq k \leq \ell$) is

$$(n + \log(\Delta) + \text{size}(\beta_1, \dots, \beta_{k-1}) + \|\text{MT}(\Omega)\|_\infty)^{O(n)},$$

which by Lemma 6.2.18 proves the assertion. \square

Finally, we describe an algorithm that given a vector finds the set of all minima that are $(3/4)$ -close to the vector.

Algorithm 6.2.20 (CUBOID)

Input : An order \mathcal{O} ; a reduced invertible ideal \mathfrak{A} of \mathcal{O} ; $\mathbf{s} \in \mathbb{Q}^r$

Output : a set W containing binary multiplicative representations $(\beta_1, \dots, \beta_\ell)$ of all minima α of \mathfrak{A} which are $(3/4)$ -close to \mathbf{s} , where $\ell = \lceil \log \|\mathbf{s}\|_\infty \rceil + 1$, and $H(\beta_i) \leq (4\Delta^{3/4})^{2(m+1)}$ for $1 \leq i \leq \ell$

- (1) **procedure** CUBOID ($\mathcal{O}, \mathfrak{A}, \mathbf{s}$)
- (2) $\beta := (\beta'_1, \dots, \beta'_\ell) := \text{CLOSE}(\mathcal{O}, \mathfrak{A}, \mathbf{s});$

- (3) $\mathfrak{C} := (1/\beta)\mathfrak{B}'$;
- (4) $M := \text{COLLECT}(\mathcal{O}, \mathfrak{C}, 4\sqrt{\Delta}, \dots, 4\sqrt{\Delta}, (4\Delta)^{m/2})$;
- (5) $W := \{(\beta'_1, \dots, \beta'_{\ell-1}, \beta_\ell \nu) : \nu \in M\}$;
- (6) **end procedure**

Lemma 6.2.21 *CUBOID (Algorithm 6.2.20) is correct. On input of an order \mathcal{O} , a reduced invertible ideal \mathfrak{A} of \mathcal{O} , and $\mathbf{s} \in \mathbb{Q}^r$ the algorithm computes a set W containing binary multiplicative representations $(\beta_1, \dots, \beta_\ell)$ of all minima α of \mathfrak{A} which are $(3/4)$ -close to \mathbf{s} , where $\ell = \lceil \log \|\mathbf{s}\|_\infty \rceil + 1$, and $H(\beta_i) \leq (4\Delta^{3/4})^{2(m+1)}$ for $1 \leq i \leq \ell$ in time*

$$(n + \log(\Delta) + \log \|\mathbf{s}\|_\infty + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

The set contains only binary representations of minima of \mathfrak{A} that are 1-close to \mathbf{s} and all representations satisfy the conditions on the height.

Proof. The proof of the correctness of the algorithm is analogous to the proofs of the correctness of the algorithms TARGET, COLLECT and CLOSE. The same holds for the proof of the running time. \square

6.3 Compact Representations of Algebraic Integer

In algebraic number fields there are many integers of small absolute norm and very big height. For example, it is conjectured that for infinitely many algebraic number fields \mathbb{F} the number of bits needed to write down the standard representations of a system of fundamental units or of a generator of a principal ideal with respect to a \mathbb{Z} -basis of an order \mathcal{O} of \mathbb{F} is $\Delta^{1/2+o(1)}$. Using CLOSE we can construct a variant of multiplicative representations, that allows us to represent such elements by only polynomially many bits. Given such a representation of a system of fundamental units we can determine in polynomial time an approximation to the regulator $R_{\mathcal{O}}$. Note, that this answers an open question suggested by H. W. Lenstra in [32, Problem 5.2].

Finally, we want to note that this new kind of representation is a generalization of the *compact representation* of algebraic integers introduced in our work in [18].

Definition 6.3.1 A *compact representation* B of an algebraic number $\alpha \in \mathbb{F}$ is a pair $B = (\gamma, (\beta_1, \dots, \beta_\ell))$ ($\ell \in \mathbb{N}$), where $\gamma \in \mathbb{F}$ with

$$\text{size}(\gamma) \leq 2n \log |\mathbb{N}_{\mathbb{F}/\mathbb{Q}}(\alpha)| + (3n^2 + n)(\log(\Delta) + \log(\text{MT}(\Omega)) + 5 \log(n) + 1), \quad (6.11)$$

and where $(\beta_1, \dots, \beta_\ell)$ is a binary multiplicative representation of a minimum ρ of \mathcal{O} with $\alpha = \gamma/\rho$ such that

$$\ell \leq \log(\log(n) + (n-1) \log(H(\alpha))) + 2$$

and

$$\text{size}(\beta_k) \leq 4n^3(\log(\Delta) + \log(\text{MT}(\Omega)) + 5 \log(n) + 1) \quad \text{for } 1 \leq k \leq \ell. \quad (6.12)$$

The total binary size of B is $\text{size}(B) = \text{size}(\gamma) + \sum_{k=1}^{\ell} \text{size}(\beta_k)$.

Before we show how to compute compact representations we first collect some simple properties of them. Clearly, from Lemma 6.2.15 and Lemma 3.4.7 it immediately follows

Proposition 6.3.2 *There is a polynomial time algorithm that given an order \mathcal{O} and a compact binary representation of an algebraic integer $\alpha \in \mathcal{O}$ computes the ideal $\alpha\mathcal{O}$.*

Lemma 3.4.4 implies

Corollary 6.3.3 *There is a polynomial time algorithm that given an order \mathcal{O} and a compact representation of $\alpha \in \mathcal{O}$ decides whether α is a unit of \mathcal{O} .*

Corollary 6.3.4 *There is an algorithm that given an order \mathcal{O} , compact representations of r units $\varepsilon_1, \dots, \varepsilon_r$ of \mathcal{O} , and a natural number q decides whether the vectors $\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r$ are linearly independent, and in that case computes a q -approximation to $\det(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r)$ in time*

$$(n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty} + q)^{\mathcal{O}(1)} .$$

Proof. If $\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r$ are linearly independent, then the lattice with basis $\mathbf{R} = (\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r)$ is a sublattice of $\text{Log } \mathcal{O}^*$. Hence, Proposition 3.5.3 implies that $\det(\mathbf{R})$ is an integral multiple of $R_{\mathcal{O}}$. We describe an algorithm that approximates that determinant. First, the algorithm determines an approximation \mathbf{R}' of precision 1 to \mathbf{R} and determines

$$\ell = \lceil \|\mathbf{R}'\|_{\infty} \rceil + 1 ,$$

which implies that $\ell > \|\mathbf{R}\|_{\infty}$. Next, the algorithm computes an approximation \mathbf{R}'' to \mathbf{R} of precision

$$p \geq q + (n + 1) \left(1 + \frac{n}{2} + \log(\ell) + \log(n) \right) + 10 .$$

From Lemma 4.2.8 it follows that $\det(\mathbf{R}'')$ is a q -approximation to $\det(\mathbf{R})$. From [63] we know that $R_{\mathcal{O}} \geq R_{\mathcal{O}_{\mathbb{F}}} > 0.05$. Hence, the logarithm vectors of the units are independent if and only if $\det(\mathbf{R}'') > 0.05 - 2^{-10}$. Clearly, from Theorem 3.4.9, Definition 6.3.1 and Proposition 6.1.1 the assertion follows. \square

Next, we describe an algorithm that given an arbitrary multiplicative representation of $\alpha \in \mathbb{F}$ and the ideal $\mathfrak{A} = \alpha\mathcal{O}$ computes a compact representation of $\alpha \in \mathcal{O}$.

Algorithm 6.3.5 (COMPACT)

Input : an order \mathcal{O} ; a multiplicative representation of $\alpha \in \mathbb{F}$; the ideal $\mathfrak{A} = \alpha\mathcal{O}$

Output : a compact representation $B = (\gamma, (\beta_1, \dots, \beta_\ell))$ of α

- (1) **procedure** COMPACT ($\mathcal{O}, \alpha, \mathfrak{A}$)
- (2) $(\mathfrak{B}, \gamma) := \text{REDUCE}(\mathcal{O}, \mathfrak{A}); \mathbf{v} = \text{LAPPROX}(\mathcal{O}, \alpha, 3);$
- (3) $\mathbf{s} := \text{LAPPROX}(\mathcal{O}, (\gamma), 3) - \mathbf{v};$
- (4) $\beta := (\beta'_1, \dots, \beta'_\ell) := \text{CLOSE}(\mathcal{O}, \mathcal{O}, \mathbf{s});$
- (5) $\mathfrak{C} := (1/\beta)\mathfrak{B}';$
- (6) $M := \text{COLLECT}(\mathcal{O}, \mathfrak{C}, 4\sqrt{\Delta}, \dots, 4\sqrt{\Delta}, (4\Delta)^{m/2});$
- (7) $q := \lceil \log(n) \rceil + n + 8;$
- (8) Find $\nu \in M$ such that $\beta\nu\alpha/\gamma = 1;$
- (9) $(\beta_1, \dots, \beta_\ell) := (\beta'_1, \dots, \beta'_{\ell-1}, \beta_\ell\nu);$
- (10) **end procedure**

Lemma 6.3.6 *The algorithm COMPACT is correct. On input of an order \mathcal{O} , a multiplicative representation M of $\alpha \in \mathbb{F}$ and the ideal $\mathfrak{A} = \alpha\mathcal{O}$ it computes a compact representation $B = (\gamma, (\beta_1, \dots, \beta_\ell))$ of α in time*

$$(n + \log(\Delta) + \text{size}(M) + \log(\log(H(\alpha))) + \log |N_{\mathbb{F}/\mathbb{Q}}(\alpha)| + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

Proof. Most parts of the proof are analogous to the proofs of section 6.2. First, we note that by Theorem 5.2.8 and Corollary 3.4.13 we can compute \mathfrak{B} and γ in time

$$n^{o(n) + \frac{n}{2}} (\log(\Delta) + \log \|\text{MT}(\Omega)\|_\infty + \log |N_{\mathbb{F}/\mathbb{Q}}(\alpha)|)^{O(1)}, \quad (6.13)$$

where we have also used the fact that by part (b) of Lemma 3.4.15 we have $|N_{\mathbb{F}/\mathbb{Q}}(\alpha)| = N_{\mathcal{O}}(\mathfrak{A})$. It further follows from Theorem 5.2.8, Lemma 3.5.25 and Lemma 3.4.15 that γ satisfies (6.11).

By the same techniques as in the proof of Proposition 6.2.9 we obtain that

$$\|\mathbf{s} - (\text{Log } \gamma - \text{Log } \alpha)\|_\infty \leq \frac{\log(\Delta) + 1}{4}, \quad (6.14)$$

and therefore β is 1-close to $\text{Log } \gamma - \text{Log } \alpha$. Moreover, it also follows that the set M must contain an element ν such that $\text{Log } \beta + \text{Log } \nu = \text{Log } \gamma - \text{Log } \alpha$ and $\beta\nu = \gamma/\alpha$.

By the definition of binary representations and by the properties of CLOSE we immediately obtain that $\beta\nu = (\beta_1, \dots, \beta_\ell)$ is a minimum of \mathcal{O} , and that the numbers $\beta_1, \dots, \beta_{\ell-1}$ satisfy (6.12). By a computation similar to the one in the prove of Lemma 6.2.18 it can easily be seen that β_ℓ satisfies (6.12) too.

By [25, Theorem 1.1] we can test whether a multiplicative representation represents the element 1 in time polynomially bounded by the size of the representation. Hence, we can perform step (8) in time

$$(n + \log(\Delta) + \text{size}(M) + \log \|\mathbf{s}\|_\infty + \log |N_{\mathbb{F}/\mathbb{Q}}(\alpha)| + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

Therefore, by Proposition 6.1.1, Proposition 6.2.12, Lemma 6.2.19 and (6.13) the running time of the algorithm is given by

$$(n + \log(\Delta) + \text{size}(M) + \log \|\mathbf{s}\|_\infty + \log |N_{\mathbb{F}/\mathbb{Q}}(\alpha)| + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

But by (6.14) and Theorem 5.2.8 and Lemma 3.4.15 we have

$$\begin{aligned} \|\mathbf{s}\|_\infty &\leq \|\text{Log } \gamma\|_\infty + \|\text{Log } \alpha\|_\infty + \frac{\log(\Delta) + 1}{4} \\ &\leq 2(\log(n) + \log(\Delta) + \log |N_{\mathbb{F}/\mathbb{Q}}(\alpha)| + \log(H(\alpha))) + \frac{\log(\Delta) + 1}{4}. \end{aligned}$$

This proves the assertion. \square

Corollary 6.3.7 *There is an algorithm that given an order \mathcal{O} and compact representations $B(\alpha)$ and $B(\beta)$ of two algebraic integers $\alpha, \beta \in \mathcal{O}$ determines a compact representation of the product $\alpha\beta$ in time*

$$(n + \log(\Delta) + \log(\log(H(\alpha\beta)))) + \log |N_{\mathbb{F}/\mathbb{Q}}(\alpha\beta)| + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

Proof. The algorithm works as follows: It determines the ideal $(\alpha\mathcal{O})(\beta\mathcal{O}) = \alpha\beta\mathcal{O}$ and calls $\text{COMPACT}(\mathcal{O}, \alpha\beta, \alpha\beta\mathcal{O})$. Here we mean by $\alpha\beta$ a multiplicative representation.

By Proposition 6.1.1 and Proposition 6.3.2 the ideal $\mathfrak{A} = \alpha\beta\mathcal{O}$ can be computed in polynomial time. Finally, by Lemma 6.3.6 and Definition 6.3.1 the running time of the algorithm COMPACT on input of \mathcal{O} , $\alpha\beta\mathcal{O}$ and $\mathbf{a} + \mathbf{b}$ is

$$(n + \log(\Delta) + \log(\log(H(\alpha\beta)))) + \log |N_{\mathbb{F}/\mathbb{Q}}(\alpha\beta)| + \|\text{MT}(\Omega)\|_\infty)^{O(n)}.$$

This concludes the proof. \square

Next, we are interested in compact representations of fundamental units and generators of principal ideals of an order \mathcal{O} . Therefore, we need upper bounds of the height of fundamental units. We are only interested in showing polynomially bounds. Note, that similar bounds are already presented in our work in [59].

Proposition 6.3.8 *Any order \mathcal{O} contains a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that*

$$\ln(H(\varepsilon_i)) \leq (r-1) \left(\frac{r(r+3)}{4}\right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16}\right)^{r-1} \quad \text{for } 1 \leq i \leq r. \quad (6.15)$$

Proof. By Proposition 3.5.30 there exists a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that for all $1 \leq i \leq r$ we have

$$\|\text{Log } \varepsilon_i\|_2 \leq \left(\frac{r(r+3)}{4}\right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16}\right)^{r-1}.$$

This implies that for all $1 \leq i \leq r$ we have

$$\left| \ln |\varepsilon_i|_j \right| \leq \left(\frac{r(r+3)}{4} \right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16} \right)^{r-1} \quad \text{for } 1 \leq j \leq r.$$

Since by Proposition 3.3.1 the norm of each unit is 1 we obtain from Proposition 3.1.7 that for all $1 \leq i \leq r$ we also have

$$|\ln |\varepsilon_i|_m| \leq (r-1) \left(\frac{r(r+3)}{4} \right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16} \right)^{r-1} \quad \text{for } 1 \leq i \leq r.$$

This proves the assertion. \square

From Definition 6.3.1, Proposition 3.3.1 and Proposition 6.3.8 we immediately obtain

Corollary 6.3.9 *Any order \mathcal{O} contains a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that the binary size of a compact representation of each ε_i ($1 \leq i \leq r$) is*

$$(n + \log(\Delta) + \log \|\text{MT}(\Omega)\|_{\infty})^{O(1)}.$$

Corollary 6.3.10 *For every principal ideal \mathfrak{A} of an order \mathcal{O} there exists a generator $\alpha \in \mathcal{O}$ of \mathfrak{A} such that the binary size of a compact representation of α is*

$$(n + \log(\Delta) + \log(\text{size}(\mathfrak{A})) + \log \|\text{MT}(\Omega)\|_{\infty})^{O(1)}.$$

Proof. Let β be a generator of \mathfrak{A} , let $\varepsilon_1, \dots, \varepsilon_r$ be a system of fundamental units of \mathcal{O} satisfying (6.15), and let

$$T = \left\{ \mathbf{z}: \mathbf{z} = \sum_{i=1}^r x_i \text{Log } \varepsilon_i, x_i \in \mathbb{R}, 0 \leq x_i < 1 \text{ for } 1 \leq i \leq r \right\}$$

be the corresponding fundamental parallelepiped in $\text{Log } \mathcal{O}^*$. Then there is a vector $\mathbf{c} = \sum_{i=1}^r y_i \text{Log } \varepsilon_i \in \text{Log } \mathcal{O}^*$, where $y_i \in \mathbb{Z}$ for $1 \leq i \leq r$, such that $\mathbf{c}' = \text{Log } \beta - \mathbf{c} \in T$. Thus we have

$$\|\mathbf{c}'\|_2 \leq r \max \{ \ell: \ell = \|\text{Log } \varepsilon_i\|_2, 1 \leq i \leq r \}.$$

If we set

$$\alpha = \frac{\beta}{\prod_{i=1}^r \varepsilon_i^{y_i}},$$

then α is a generator of \mathfrak{A} with $\text{Log } \alpha = \mathbf{c}'$. Therefore, we have by (6.15)

$$|\ln |\alpha|_i| \leq r(r-1) \left(\frac{r(r+3)}{4} \right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16} \right)^{r-1} \quad \text{for } 1 \leq i \leq r.$$

From Proposition 3.1.7 and Lemma 3.4.15 we obtain

$$\begin{aligned} |\ln |\alpha|_m| &= |\ln |\text{N}_{\mathcal{O}}(\mathfrak{A})| - \sum_{i=1}^r \ln |\alpha|_i| \\ &\leq |\log |\text{N}_{\mathcal{O}}(\mathfrak{A})|| + r^2(r-1) \left(\frac{r(r+3)}{4} \right)^{\frac{r}{2}} R_{\mathcal{O}} \left(\frac{17^r 4^{2+t}}{16} \right)^{r-1}. \end{aligned}$$

Since $r \leq n$, the assertion follows from Definition 6.3.1, Corollary 3.4.13 and Theorem 3.4.9. \square

6.4 Applications in Complexity Theory

We now apply the results of the previous sections for examining the structural properties of some important problems in algorithmic algebraic number theory.

Let \mathbb{F} be an algebraic number field, i.e. a finite field extension of the field \mathbb{Q} of the rational numbers. In [39] and [16] it was shown that the problem of computing the value of the class number h of \mathbb{F} belongs to the complexity class \mathcal{NP} if \mathbb{F} is of degree 2 and if a certain Generalized Riemann Hypothesis (GRH) holds. That is, assuming the GRH, there exists a nondeterministic polynomial time algorithm that accepts the set of all pairs (Δ, h) , where $\Delta \in \mathbb{Z}$ is square free and h is the class number of $\mathbb{Q}(\sqrt{\Delta})$. To be more precise, we denote by \mathcal{NP} the class of all languages that are accepted by nondeterministic polynomial time algorithms. For more details we refer to [2].

We will generalize this result by showing that under the assumption of the GRH the problem of computing the value of the class number $h_{\mathcal{O}_{\mathbb{F}}}$ of the maximal order of an arbitrary algebraic number field \mathbb{F} of arbitrary degree belongs to \mathcal{NP} . We shall also prove that computing the compact representations of a system of fundamental units of the maximal order is in \mathcal{NP} . Note that this result is already published in our work [60].

The main tool in our proofs is the compact representation of algebraic integers introduced in the previous section. We will use it to show that there exists a short proof for the principality of a given ideal. As an application of this result we will show that under the assumption of the generalized Riemann Hypothesis (GRH) there exists a short proof that a certain integer H is divisible by the class number $h_{\mathcal{O}_{\mathbb{F}}}$. As described in the previous section we can use short representations of elements of a system of fundamental units to compute a rational R^* which is an approximation of $mR_{\mathcal{O}_{\mathbb{F}}}$ for some $m \in \mathbb{Z}$. Making use of the analytic class number formula, we here will show, how to compute a number Θ such that $(6/5)h_{\mathcal{O}_{\mathbb{F}}}R_{\mathcal{O}_{\mathbb{F}}} < \Theta < (15/8)h_{\mathcal{O}_{\mathbb{F}}}R_{\mathcal{O}_{\mathbb{F}}}$. Then we can easily verify that $H/h_{\mathcal{O}_{\mathbb{F}}} = m = 1$, because $HR^* \leq \Theta$ if and only if $H/h_{\mathcal{O}_{\mathbb{F}}} = m = 1$.

We first show

Theorem 6.4.1 *The set PRI of all pairs $(\mathcal{O}, \mathfrak{A})$, where \mathcal{O} is an order of an algebraic number field and \mathfrak{A} is a principal ideal in \mathcal{O} , belongs to \mathcal{NP} .*

Proof. Let $(\mathcal{O}, \mathfrak{A}) \in \text{PRI}$, where \mathcal{F} is a number field of degree n . Then by Corollary 6.3.10 and Lemma 6.3.6 there exists a generator α of the ideal \mathfrak{A} and a compact representation of α such that the binary size of this compact representation is

$$(n + \log(\Delta_{\mathcal{O}}) + \log(\text{size}(\mathfrak{A})) + \log \|\text{MT}(\Omega)\|_{\infty})^{O(1)}.$$

By Proposition 6.3.2 there is a polynomial time algorithm, which given the compact representation of α computes $\alpha\mathcal{O}$. Thus a nondeterministic polynomial time algorithm, that accepts PRI, only has to guess the compact representation of α , to compute $\alpha\mathcal{O}$ and to compare this ideal with \mathfrak{A} . \square

By this theorem and Proposition 6.1.1 we obtain the following complexity result for discrete logarithm problems of orders.

Corollary 6.4.2 *The set of all tuples $(\mathcal{O}, \mathfrak{A}, p, \mathbf{a})$ where \mathfrak{A} is a principal ideal of \mathcal{O} and \mathbf{a} is a $\lfloor \log(p) \rfloor$ -approximation to the logarithm vector of a generator of \mathfrak{A} belongs to \mathcal{NP} .*

Proposition 6.4.3 *There is a polynomial time algorithm that given an order \mathcal{O} , algebraic integers $\beta_1, \dots, \beta_i \in \mathcal{O}$ and ideals $\mathfrak{B}_1, \dots, \mathfrak{B}_i$ of \mathcal{O} , where $i \in \mathbb{N}$, decides whether $\mathfrak{B}_j = (1/\beta_j)(\mathfrak{B}_{j-1})^2$ for $1 \leq j \leq i$.*

Proof. Given an ideal we can compute the square of it in polynomial time. Given $\beta \in \mathcal{O}$ we can compute $1/\beta$ in polynomial time, as stated by Lemma 3.1.2. Hence the assertion follows. \square

Lemma 6.4.4 *Let \mathcal{O} be an order, let \mathfrak{A} be an ideal of \mathcal{O} and let $x = 2^i$, $i \in \mathbb{N}$. Then there are $\beta_0, \beta_1, \dots, \beta_i \in \mathcal{O}$ with*

$$\text{size}(\beta_j) = (n + \log(\Delta_{\mathcal{O}}) + \log(\|\text{MT}(\Omega)\|_{\infty}))^{O(1)} \quad \text{for } 1 \leq j \leq i$$

and ideals $\mathfrak{B}_1, \dots, \mathfrak{B}_i$ of \mathcal{O} with

$$\text{size}(\mathfrak{B}_j) = (\log(\Delta_{\mathcal{O}}) + n)^{O(1)} \quad \text{for } 1 \leq j \leq i,$$

such that $\mathfrak{B}_j = (1/\beta_j)(\mathfrak{B}_{j-1})^2$ and $\mathfrak{A}^x \sim \mathfrak{B}_i$.

Proof. We set $\mathfrak{B}_0 = 1/\beta_0\mathfrak{A}$ and $\mathfrak{B}_j = (1/\beta_j)(\mathfrak{B}_{j-1})^2$ for $j \geq 1$, where β_0 is a minimum of \mathfrak{A} , and β_j is a minimum of $(\mathfrak{B}_{j-1})^2$, and $\text{Log } \beta_0$ and each $\text{Log } \beta_j$ belong to

$$\mathcal{W}(\mathbf{0}) = \{\mathbf{x}: \mathbf{x} \in \mathbb{R}^r, |\mathbf{x}_i| \leq (\log(\Delta_{\mathcal{O}}))/4 \text{ for } 1 \leq i \leq r\}.$$

Let $\mathfrak{B} = \mathfrak{B}_i$. Clearly, \mathfrak{B} is reduced and $\mathfrak{A}^x \sim \mathfrak{B}$. Using the techniques of the proof of Proposition 6.2.9 we obtain that

$$H(\beta_j) \leq (4\Delta_{\mathcal{O}})^{3(m+1)/4}.$$

Thus, applying Lemma 3.5.25 we obtain for $1 \leq j \leq i$

$$\text{size}(\beta_j) = (n + \log(\Delta_{\mathcal{O}}) + \log(\|\text{MT}(\Omega)\|_{\infty}))^{O(1)}.$$

Since each \mathfrak{B}_j is reduced, from Corollary 5.1.12 it follows that

$$\text{size}(\mathfrak{B}_j) = (\log(\Delta_{\mathcal{O}}) + n)^{O(1)}. \quad \square$$

Lemma 6.4.5 *Let \mathcal{O} be an order, let $\mathfrak{A}_0, \dots, \mathfrak{A}_{\ell}$ be reduced ideals of \mathcal{O} , $\ell \in \mathbb{N}$. Then there are $\beta_0, \beta_1, \dots, \beta_{\ell} \in \mathcal{O}$ with $\text{size}(\beta_j) = (n + \log(\Delta_{\mathcal{O}}) + \log(\|\text{MT}(\Omega)\|_{\infty}))^{O(1)}$ and ideals $\mathfrak{B}_1, \dots, \mathfrak{B}_{\ell}$ with $\text{size}(\mathfrak{B}_j) = (\log(\Delta_{\mathcal{O}}) + n)^{O(1)}$ for $1 \leq j \leq \ell$ such that $\mathfrak{B}_j = (1/\beta_j)(\mathfrak{A}_j\mathfrak{B}_{j-1})$ and $\prod_{i=1}^{\ell} \mathfrak{A}_i \sim \mathfrak{B}_{\ell}$.*

Proof. Let $\mathfrak{B}_0 = 1/\beta_0\mathfrak{A}_0$ and $\mathfrak{B}_j = (1/\beta_j)(\mathfrak{A}_j\mathfrak{B}_{j-1})$ for $1 \leq j \leq \ell$, where β_0 is a minimum of \mathfrak{A} and β_j is a minimum in $\mathfrak{A}_j\mathfrak{B}_{j-1}$ with $\text{Log } \beta_j \in \mathcal{W}(\mathbf{0})$. We set $\mathfrak{B} = \mathfrak{B}_\ell$. Then the assertion follows analogously to the proof of Lemma 6.4.4. \square

Theorem 6.4.6 *The set of all tuples $(\mathcal{O}, \mathfrak{A}_0, \dots, \mathfrak{A}_\ell, x_0, \dots, x_\ell)$ where \mathcal{O} is an order of an algebraic number field and $\mathfrak{A}_0, \dots, \mathfrak{A}_\ell$ are ideals of \mathcal{O} and $x_0, \dots, x_\ell \in \mathbb{Z}$, $\ell \in \mathbb{N}$, such that $\prod_{i=0}^{\ell} \mathfrak{A}_i^{x_i} \sim \mathcal{O}$, belongs to \mathcal{NP} .*

Proof. We may assume that $x_i \geq 0$ for $1 \leq i \leq \ell$. Otherwise, we replace \mathfrak{A}_i by \mathfrak{A}_i^{-1} , which by Lemma 3.4.7 can be computed in polynomial time. For $1 \leq i \leq \ell$ let

$$x_i = \sum_{j=0}^{y_i} x_{i,j} 2^j,$$

where $x_{i,j} \in \{0, 1\}$ for $0 \leq j \leq y_i$ with $y_i = \lceil \log(x_i) \rceil + 1$. Then we have

$$\prod_{i=0}^{\ell} \mathfrak{A}_i^{x_i} = \prod_{i=0}^{\ell} \prod_{j=0}^{y_i} (\mathfrak{A}_i^{x_{i,j}})^{2^j}.$$

Thus, from Proposition 6.4.3 and Lemma 6.4.4 and Lemma 6.4.5 it follows that there is a nondeterministic polynomial time algorithm that can guess an ideal \mathfrak{B} , whose binary size is polynomially bounded by the size of the input tuple, and test if $\prod_{i=0}^{\ell} \mathfrak{A}_i^{x_i} \sim \mathfrak{B}$. Hence, the assertion follows from Theorem 6.4.1. \square

Before we show that computing class numbers of maximal orders belongs to \mathcal{NP} we need

Definition 6.4.7 Let $\mathcal{O}_{\mathbb{F}}$ be the maximal order of an algebraic number field \mathbb{F} . An ideal \mathfrak{P} of $\mathcal{O}_{\mathbb{F}}$ is called *prime*, if for all $\alpha, \beta \in \mathbb{F}$ with $\alpha\beta \in \mathfrak{P}$ we have $\alpha \in \mathfrak{P}$ or $\beta \in \mathfrak{P}$.

Theorem 6.4.8 *If GRH holds then the set of all pairs (\mathbb{F}, H) , where \mathbb{F} is an algebraic number field and where H is a multiple of $h_{\mathcal{O}_{\mathbb{F}}}$, belongs to \mathcal{NP} .*

Proof. By [11] we can test whether a given order is a maximal order by finding the prime factorization of the discriminant of the given order. Thus, there is a nondeterministic polynomial time algorithm that accepts a multiplication table $\text{MT}(\Omega)$ if and only if it encodes the maximal order of a number field.

Now, we may suppose that we know a multiplication table $\text{MT}(\Omega_{\mathbb{F}})$ of a \mathbb{Z} -basis $\Omega_{\mathbb{F}}$ of a maximal order $\mathcal{O}_{\mathbb{F}}$ of an algebraic number field \mathbb{F} of degree n . Let

$$F = \{\mathfrak{P}_1, \dots, \mathfrak{P}_{k(\mathbb{F})}\},$$

where $k(\mathbb{F}) \in \mathbb{N}$, be the set of all prime ideals of the maximal order $\mathcal{O}_{\mathbb{F}}$ whose norms do not exceed $12(\log \Delta_{\mathbb{F}})^2 + 1$. Then $k(\mathbb{F}) \leq n(12(\log \Delta_{\mathbb{F}})^2 + 1)$ and by means of the methods

presented in [10] and [62] the set F can be computed in time $O(\log(\|\text{MT}(\Omega_{\mathbb{F}})\|_{\infty}) + n + \log(\Delta))^{O(1)}$. We set

$$A = \left\{ \alpha : \alpha \in \mathbb{F} - \{0\}, \alpha \mathcal{O}_{\mathbb{F}} = \prod_{i=1}^{k(\mathbb{F})} \mathfrak{P}_i^{e_i}, e_i \in \mathbb{Z} \text{ for } 1 \leq i \leq k(\mathbb{F}) \right\}.$$

and finally, as in [7], we define the function

$$\varphi' : A \rightarrow \mathbb{Z}^{k(\mathbb{F})}, \quad \alpha \mapsto \varphi'(\alpha) = (e_1, \dots, e_{k(\mathbb{F})}).$$

If we assume the GRH then by [7] the image $L' = \varphi'(A)$ is a $k(\mathbb{F})$ -dimensional lattice of determinant $h_{\mathbb{F}}$.

If we get a matrix $\mathbf{E} = (e_{i,j}) \in \mathbb{Z}^{k(\mathbb{F}) \times k(\mathbb{F})}$, such that

$$\prod_{j=1}^{k(\mathbb{F})} \mathfrak{P}_j^{e_{i,j}} \sim \mathcal{O} \quad \text{for all } 1 \leq i \leq k(\mathbb{F}). \quad (6.16)$$

then the vectors $\mathbf{e}_1, \dots, \mathbf{e}_{k(c, \mathbb{F})}$, where $\mathbf{e}_i = (e_{i,1}, \dots, e_{i,k(c, \mathbb{F})})$ for $1 \leq i \leq k(\mathbb{F})$, form a sublattice Λ of L'_c . Thus from Proposition 3.5.3 we know that $|\det(E)|$ is a multiple of $h_{\mathbb{F}}$. Without loss of generality we may assume that \mathbf{E} is in Hermite normal form (see Lemma 2.1.1). Then we have

$$\|\mathbf{E}\|_{\infty} \leq |\det(\mathbf{E})|.$$

Conversely, for each multiple H of the class number $h_{\mathcal{O}_{\mathbb{F}}}$ there exists a matrix \mathbf{E} of the form described above. Thus a nondeterministic algorithm which accepts the multiple H of the class number only has to guess a matrix $\mathbf{E} = (e_{i,j}) \in \mathbb{Z}^{k(\mathbb{F}) \times k(\mathbb{F})}$ and to verify condition (6.16). By Theorem 6.4.6 this can be done by a nondeterministic algorithm in polynomial time. \square

Theorem 6.4.9 *If GRH holds then there is a polynomial time algorithm that given an algebraic number field \mathbb{F} and the number w of roots of unity in \mathbb{F} computes a number $\Theta \in \mathbb{R}_{>0}$ such that $(101/96)h_{\mathcal{O}_{\mathbb{F}}}R_{\mathcal{O}_{\mathbb{F}}} \leq \Theta \leq (1515/768)h_{\mathcal{O}_{\mathbb{F}}}R_{\mathcal{O}_{\mathbb{F}}}$.*

Proof. We use the results of [15]. The product $h_{\mathcal{O}_{\mathbb{F}}}R_{\mathcal{O}_{\mathbb{F}}}$ can be expressed by means of the analytic class number formula

$$h_{\mathbb{F}}R_{\mathbb{F}} = C_{\mathbb{F}} \prod_{p \in \mathbb{P}} E(p),$$

where

$$C_{\mathbb{F}} = \frac{w\sqrt{\Delta}}{2^s(2\pi)^t},$$

\mathbb{P} is the set of rational primes and $E(p)$ is the Euler factor belonging to p . Since s and t can be determined in polynomial time from a generating polynomial f of \mathbb{F} , which can be computed in polynomial time too (see for example [43]), we only have to describe a

method for computing an approximation of $\prod_{p \in \mathbb{P}} E(p)$. For this purpose let \mathbb{L} be the normal closure of \mathbb{F} . As in [15] we choose $Q \in \mathbb{Z}_{>0}$ and split $\prod_{p \in \mathbb{P}} E(p) = F(Q)T(Q)$, where

$$F(Q) = \prod_{p \in \mathbb{P}, p \leq Q} E(p) \quad \prod_{\substack{p \in \mathbb{P}, p > Q \\ p \text{ ramified in } \mathbb{L}}} E(p)$$

and

$$T(Q) = \prod_{\substack{p \in \mathbb{P}, p > Q \\ p \text{ unramified in } \mathbb{L}}} E(p).$$

Then we have $h_{\mathcal{O}_{\mathbb{F}}} R_{\mathcal{O}_{\mathbb{F}}} = C_F F(Q) T(Q)$. From [15] (cf. inequality (3.4) and Theorem 3.1) we obtain that

$$|\log(T(Q))| \leq (c_3 \log(\Delta)) / \sqrt{Q},$$

where $c_3 = n^{O(1)}$. Therefore, if we set

$$Q = (c_3 \log(\Delta) / \log(5/4))^2 \quad \text{and} \quad \Theta' = (3/2) C_{\mathbb{F}} F(Q),$$

then

$$\frac{6}{5} h_{\mathcal{O}_{\mathbb{F}}} R_{\mathcal{O}_{\mathbb{F}}} < \Theta' < \frac{15}{18} h_{\mathcal{O}_{\mathbb{F}}} R_{\mathcal{O}_{\mathbb{F}}}.$$

By the methods of [15] the value $E(p)$ can be computed in time polynomially bounded by the input size and p for every $p \leq Q$. Now, let

$$p = \lceil -\log\left(\frac{5}{96} C_{\mathbb{F}}\right) \rceil + 1,$$

and let C' be a p -approximation to $C_{\mathbb{F}}$. Then by [51] we can determine an approximation C' in time $(s + t + \log(\Delta_{\mathbb{F}}))^{O(1)}$, and $\Theta = (3/2) C' F(Q)$ satisfies the stated bounds. This concludes the proof. \square

Theorem 6.4.10 *If GRH holds then the set \mathcal{H} of all pairs $(\mathbb{F}, h_{\mathcal{O}_{\mathbb{F}}})$ belongs to \mathcal{NP} .*

Proof. We describe a nondeterministic algorithm that accept the set \mathcal{H} . We know that there is a nondeterministic polynomial time algorithm which guesses on input of \mathbb{F} the representation of the maximal order $\mathcal{O}_{\mathbb{F}}$ of \mathbb{F} and verifies it. This assertion holds because we may assume that the binary size of the given multiplication table is polynomially bounded by $(2 + \log_2(\Delta_{\mathbb{F}}))^{O(1)}$ and since the set of all pairs $(\mathbb{F}, \mathcal{O}_{\mathbb{F}})$ belongs to \mathcal{NP} . There also exists a nondeterministic polynomial time algorithm that guesses the compact representations of r elements $\epsilon_1, \dots, \epsilon_r$ such that the binary size of their compact representations satisfies (6.3.9). By Corollary 6.3.3, the algorithm can test in polynomial time, whether the ϵ_i are units of $\mathcal{O}_{\mathbb{F}}$. Finally, we note that there is a nondeterministic algorithm which guesses a number $w \leq n(n+1)/2$ and w elements of \mathbb{F} and tests whether they are roots of unity. Since the height of that elements is 1 and hence by Lemma 3.5.25 their binary size is polynomial in the input size and $\log(\Delta)$, this can be done in polynomial time too. The algorithm that accepts the set \mathcal{H} first starts the above

algorithms. After that initialization, it checks by the method of Corollary 6.3.4 whether the logarithm vectors of the units ϵ_i are independent, and in that case it computes an approximation R^* of precision 10 to the absolute value of the determinant of the matrix $(\text{Log } \epsilon_1, \dots, \text{Log } \epsilon_r)$. Then we know that R^* is an approximation of precision 10 to a multiple of $R_{\mathcal{O}_{\mathbb{F}}}$. Using the algorithm of Theorem 6.4.9 it computes an approximation Θ of $h_{\mathcal{O}_{\mathbb{F}}} R_{\mathcal{O}_{\mathbb{F}}}$ such that $(101/96)h_{\mathcal{O}_{\mathbb{F}}} R_{\mathcal{O}_{\mathbb{F}}} < \Theta < (1515/768)h_{\mathcal{O}_{\mathbb{F}}} R_{\mathcal{O}_{\mathbb{F}}}$. Then the algorithm verifies according to Theorem 6.4.8 that H is a multiple of the class number $h_{\mathcal{O}_{\mathbb{F}}}$. Clearly, $H = h_{\mathcal{O}_{\mathbb{F}}}$ if and only if $HR^* \leq \Theta$. \square

By essentially the same proof and noting that by Theorem 3.4.9 the binary size of the class number is $(\log(\Delta))^{O(1)}$ we have

Theorem 6.4.11 *If GRH holds then the set \mathcal{R} of all tuples $(\mathbb{F}, B(\varepsilon_1), \dots, B(\varepsilon_r))$, where $B(\varepsilon_1), \dots, B(\varepsilon_r)$ are compact representations of a system of fundamental units of $\mathcal{O}_{\mathbb{F}}$ belongs to \mathcal{NP} .*

From Theorem 6.4.6 and Theorem 6.4.10 also immediately follows

Theorem 6.4.12 *If GRH holds then the set of all tuples $(\mathbb{F}, \mathfrak{A}, \mathfrak{D}, y)$ where \mathbb{F} is an algebraic number field, \mathfrak{A} and \mathfrak{D} are ideals of $\mathcal{O}_{\mathbb{F}}$, and $y = \log_{[\mathfrak{D}]}([\mathfrak{A}])$ belongs to \mathcal{NP} .*

Let ORD be the set of all triples $(\mathbb{F}, \mathfrak{A}, \ell)$ where \mathbb{F} is an algebraic number field and ℓ is the order of the ideal class $[\mathfrak{A}]$ of the ideal \mathfrak{A} in the class group $Cl_{\mathcal{O}_{\mathbb{F}}}$ of \mathbb{F} . Let GEN be the set of all tuples $(\mathbb{F}, \mathfrak{A}_1, \dots, \mathfrak{A}_k)$, where \mathbb{F} is an algebraic number field and $\mathfrak{A}_1, \dots, \mathfrak{A}_k$ are ideals such that the class group $Cl_{\mathcal{O}_{\mathbb{F}}}$ is generated by the classes $[\mathfrak{A}_1], \dots, [\mathfrak{A}_k]$. Then from our theorem and by proofs that are absolutely analogous to the proofs of [17] we immediately obtain

Corollary 6.4.13 *Under the assumption of the GRH the sets \mathcal{H} , PRI, ORD and GEN belong to the complexity class $\mathcal{NP} \cap \text{co-NP}$.*

Chapter 7

Computing Units and Discrete Logarithms in Class Groups

7.1 The Main Idea

In the following sections, let \mathbb{F} be a number field of degree n and of signature (s, t) , and set $m = s + t$ and $r = s + t - 1$. Let \mathcal{O} be an order of \mathbb{F} , and for convenience, let $\Delta = |\Delta_{\mathcal{O}}|$. Furthermore, we assume that \mathcal{O} is given by the multiplication table $\text{MT}(\Omega)$, where $\Omega = (\omega_1, \dots, \omega_n)$ is a \mathbb{Z} -basis of \mathcal{O} .

As a further application of our previous results we want to describe and analyze algorithms for solving the discrete logarithm problem of orders and in the class group of an order \mathcal{O} . We shall also describe algorithms for computing a system of fundamental units and approximations to the regulator as well as principal ideal testing and computing relative generators of ideals. Our work can be seen as a generalization of [3].

We start with the algorithm BOUNDED for solving the *bounded discrete logarithm problem* in the class group, that is to say, given an order \mathcal{O} , two invertible ideals \mathfrak{A} and \mathfrak{B} , and $u \in \mathbb{N}_{\geq 2}$ the algorithm decides if there exists $y \in \mathbb{N}$, $1 \leq y < u$, such that $[\mathfrak{A}] = [\mathfrak{B}]^y$, and in that case finds the minimal such y . We use the well known Baby-step–Giant-step strategy discovered by Shanks (see [53], [54]) for solving discrete logarithm problems in a finite abelian group.

Algorithm 7.1.1 (BOUNDED)

Input : an order \mathcal{O} ; invertible ideals \mathfrak{A} and \mathfrak{D} of \mathcal{O} ; $u \in \mathbb{N}_{\geq 2}$
Output : if there exists $y \in \mathbb{N}$, $1 \leq y < u$, with $[\mathfrak{A}] = [\mathfrak{D}]^y$, then the minimal such y , else $y = 0$

- (1) **procedure** BOUNDED ($\mathcal{O}, \mathfrak{A}, \mathfrak{D}, u$)
- (2) $T := \emptyset$; $r := 1$; $y := 0$;
- (3) $(\mathfrak{A}, \gamma) := \text{REDUCE}(\mathcal{O}, \mathfrak{A})$; $(\mathfrak{D}, \delta) := \text{REDUCE}(\mathcal{O}, \mathfrak{D})$;
- (4) $\mathfrak{C} := \mathcal{O}$;
- (5) **repeat**
- (6) $(\mathfrak{C}, \gamma) := \text{REDUCE}(\mathcal{O}, \mathfrak{C}\mathfrak{D})$; $T := T \cup \{(\mathfrak{C}, r)\}$;
- (7) $r := r + 1$;
- (8) **until** $((r^2 \geq u) \text{ or } ([\mathfrak{C}] = [\mathfrak{A}]) \text{ or } ([\mathfrak{C}] = [\mathfrak{D}]));$

```

(9)   if ( $[\mathfrak{C}] = [\mathfrak{A}]$ ) then
(10)      $y := r$ ;
(11)   else
(12)     if ( $[\mathfrak{C}] \neq [\mathfrak{D}]$ ) then
(13)        $q := 0$ ;  $\mathfrak{B} := \mathfrak{A}\mathfrak{C}$ ;
(14)     repeat
(15)        $\mathfrak{B} := \text{REDUCE}(\mathcal{O}, \mathfrak{B}\mathfrak{C}^{-1})$ ;
(16)       if ( $([\mathfrak{B}], r) \in T$  for some  $r$ ) then
(17)          $y := q\lfloor\sqrt{u}\rfloor + r$ 
(18)       else
(19)          $q := q + 1$ 
(20)       fi
(21)     until ( $y \neq 0$  or  $q^2 > u$ )
(22)   fi
(23) fi
(24) end procedure

```

In the description of the algorithm we make the assumption that we can decide whether two ideals are in the same ideal class. This can be seen as a special case of the following problem: given ideals $\mathfrak{B}_1, \dots, \mathfrak{B}_k$, $k \in \mathbb{N}$, and an ideal \mathfrak{B} we have to decide whether $[\mathfrak{B}] \in \{[\mathfrak{B}_1], \dots, [\mathfrak{B}_k]\}$. This problem is called the *containment problem* or *containment test*. In fact, it will be one of the major problems of this chapter to show how such a containment test can be implemented. Also note that we have not used the element γ , since we are only interested in the corresponding reduced ideals. By the ideas described in [53], [54] and [3] we have

Theorem 7.1.2 *BOUNDED is correct.*

Proof. Let $y \in \mathbb{N}$ with $y < u$ such that $[\mathfrak{A}] = [\mathfrak{D}]^y$ be minimal. Then there exist $a \in \mathbb{N}$ with $a^2 < u$ and $b \in \mathbb{N}_0$ with $b^2 \leq u$ such that $y = b\lfloor\sqrt{u}\rfloor + a$. In the first repeat-loop we have for each pair (\mathfrak{C}, r) that $[\mathfrak{C}] = [\mathfrak{D}]^r$. In the second one we always have $[\mathfrak{B}] = [\mathfrak{A}][[\mathfrak{D}]]^{-q\lfloor\sqrt{u}\rfloor}$. Hence for $r = a$ an element (\mathfrak{C}, a) is inserted in T . Since for $q = b$ we have $[\mathfrak{B}] = [\mathfrak{D}]^r$ the algorithm shall correctly compute y . Clearly, if there do not exist $y \in \mathbb{N}$ with $y < u$ such that $[\mathfrak{A}] = [\mathfrak{D}]^y$ then the output of the algorithm is $y = 0$. \square

Using BOUNDED the algorithm DISCRETE for computing discrete logarithms in the class group works as follows:

Algorithm 7.1.3 (DISCRETE)

Input : an order \mathcal{O} ; invertible ideals \mathfrak{A} and \mathfrak{D} of \mathcal{O}
Output : if there exists $y \in \mathbb{N}$ with $[\mathfrak{A}] = [\mathfrak{D}]^y$, then the minimal such y , else $y = 0$

```

(1) procedure DISCRETE ( $\mathcal{O}, \mathfrak{A}, \mathfrak{D}$ )
(2)    $u := 2$ ;

```

```

(3)   repeat
(4)      $y := \text{BOUNDED}(\mathcal{O}, \mathfrak{A}, \mathfrak{D}, u);$ 
(5)      $o := \text{BOUNDED}(\mathcal{O}, \mathcal{O}, \mathfrak{D}, u);$ 
(6)      $u := 2u;$ 
(7)   until  $(y + o \neq 0)$ 
(8) end procedure

```

Again, by [53], [54] and [3] we have

Theorem 7.1.4 *DISCRETE is correct.*

In the following we have to explain how to implement the containment test. An important tool for our implementation shall be an algorithm for computing a system of fundamental units of an order. This algorithm shall be described in the next section (section 7.2) in full detail. Then in section 7.3 we finally can describe the containment test and complete the analysis of DISCRETE (Algorithm 7.1.3).

7.2 Computing a System of Fundamental Units

We consider the following problem : Given \mathcal{O} we want to find a system of fundamental units of \mathcal{O} , or to be more precise, compact representations of these units such that their logarithm vectors form a basis of $\text{Log } \mathcal{O}^*$ of a special shape, a so-called (ϵ, δ) -constructable basis. To solve the problem we shall use the strategy described in [6] for computing a basis of the unit lattice of \mathcal{O} . Clearly, given such a basis we can find the wanted compact representations by means of the algorithm COMPACT (Algorithm 6.3.5). While in [6] the running time of the algorithm is only given for a fixed degree n of the number field we shall here show the dependence of the running time on n . For this purpose we especially have to consider the various cases where approximations take influence on the behavior of the algorithm.

We assume that minima and units are represented by compact representations. For convenience, we proceed as in section 6.1 and identify each element $\xi \in \mathbb{F}$ with its compact representation. Finally, let k always be a natural number.

We start by introducing the notion of a (ϵ, δ) -constructable basis of an arbitrary lattice $\Lambda \in \mathbb{R}^r$ of dimension r . First, we need

Definition 7.2.1 Let B be a finite sequence of vectors in \mathbb{R}^r , and let $\mathbf{c} \in \mathbb{R}^r$. Then we call $|\mathbf{c}|_B = \max \{z : z = |\langle \mathbf{c}, \mathbf{a} \rangle|, \mathbf{a} \in B\}$ the *height* of \mathbf{c} with respect to B .

Definition 7.2.2 Let A be an arbitrary subset of \mathbb{R}^r , let B be a nonempty and finite sequence of vectors in \mathbb{R}^r , and let $\epsilon \in \mathbb{R}$, $0 < \epsilon \leq 1$. Then we denote by $\mathcal{S}(A, B, \epsilon)$ the set of all vectors $\mathbf{a} \in A$ with $|\mathbf{a}|_B > 0$ such that for all $\mathbf{c} \in A$ satisfying $|\mathbf{c}|_B > 0$ we have $|\mathbf{c}|_B \geq \epsilon |\mathbf{a}|_B$.

Definition 7.2.3 Let $\delta \in \mathbb{R}$, $0 < \delta \leq 1$. Let $\mathbf{a}_1, \dots, \mathbf{a}_k$ be pairwise orthogonal vectors in \mathbb{R}^r and let B_k be a basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Then we define $\mathcal{P}_\delta(\mathbf{a}_1, \dots, \mathbf{a}_k)$ to be the set of all vectors $\mathbf{v} \in \mathbb{R}^r$ of the shape

$$\mathbf{v} = \sum_{j=1}^k x_j \mathbf{a}_j + \sum_{\mathbf{a} \in B_k} x_{\mathbf{a}} \mathbf{a}, \text{ where } x_j \in \mathbb{R}, |x_j| \leq \delta \text{ for } 1 \leq j \leq k, x_{\mathbf{a}} \in \mathbb{R} \text{ for } \mathbf{a} \in B_k.$$

Note that $\mathcal{P}_\delta(\mathbf{a}_1, \dots, \mathbf{a}_k)$ is independent of the choice of the basis B_k and hence is well defined.

Definition 7.2.4 Let $\epsilon, \delta \in \mathbb{R}$, $0 < \epsilon \leq 1, 0 < \delta \leq 1$. A sequence $(\mathbf{a}_1, \dots, \mathbf{a}_\ell)$ ($1 \leq \ell \leq r$) of lattice vectors in a r -dimensional lattice Λ in \mathbb{R}^r is called (ϵ, δ) -constructable, if for all k with $0 \leq k \leq \ell - 1$ we have $\mathbf{a}_{k+1} \in \mathcal{S}(\Lambda, B_k, \epsilon) \cap \mathcal{P}_\delta(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$, where B_k is an orthonormal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$.

In what follows we shall show that for every r -dimensional lattice Λ in \mathbb{R}^r and for every $\epsilon, \delta \in \mathbb{R}$, $0 < \epsilon \leq 1, 0 < \delta \leq 1$ there exists a basis of Λ that is (ϵ, δ) -constructable. The proof of this claim shall be constructive and lead to an algorithm for computing such a basis. We start with the following observations:

Let $\mathbf{a}_1, \dots, \mathbf{a}_k$ be linearly independent vectors in \mathbb{R}^r . Then for any orthogonal basis B_k of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$ the combination of $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ and B_k is a basis of the space \mathbb{R}^r . Thus for every $\mathbf{b} \in \mathbb{R}^r$ there are uniquely determined $x_i \in \mathbb{R}$ ($1 \leq i \leq k$) and $x_{\mathbf{a}} \in \mathbb{R}$ ($\mathbf{a} \in B_k$) such that $\mathbf{b} = \sum_{i=1}^k x_i \mathbf{a}_i + \sum_{\mathbf{a} \in B_k} x_{\mathbf{a}} \mathbf{a}$. Hence,

$$|\mathbf{b}|_{B_k} = \max \{z : z = |x_{\mathbf{a}}|, \mathbf{a} \in B_k\}.$$

If $\pi_k(\mathbf{b})$ is the projection of \mathbf{b} onto $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$, then $\pi_k(\mathbf{b}) = \sum_{\mathbf{a} \in B_k} x_{\mathbf{a}} \mathbf{a}$, and

$$|\mathbf{b}|_{B_k} = |\pi_k(\mathbf{b})|_{B_k} \quad \text{and} \quad \|\pi_k(\mathbf{b})\|_2 = \left(\sum_{\mathbf{a} \in B_k} x_{\mathbf{a}}^2 \right)^{\frac{1}{2}}.$$

From this observation and from Lemma 3.5.12 we obtain

Proposition 7.2.5 Let $\{\mathbf{a}_1, \dots, \mathbf{a}_k\}$ ($0 \leq k \leq r - 1$) be linearly independent vectors in \mathbb{R}^r . Let B_k be an orthogonal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Let $\mathbf{b} \in \mathbb{R}^r$, and let $\pi_k(\mathbf{b})$ be the projection of \mathbf{b} onto $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Then we have

$$\sqrt{r-k} |\mathbf{b}|_{B_k} \geq \|\pi_k(\mathbf{b})\|_2 \geq |\mathbf{b}|_{B_k} = |\pi_k(\mathbf{b})|_{B_k}.$$

A simple consequence of the above proposition is

Corollary 7.2.6 Let $B \subseteq \mathbb{R}^r$ be a sequence of pairwise orthogonal vectors, and let $\mathbf{c} \in \mathbb{R}^r$. Then $|\mathbf{c}|_B = 0$ if and only if $\mathbf{c} \in \text{span}(B)^\perp$.

After these preliminaries we can describe the construction of a (ϵ, δ) -constructable basis of a lattice. We proceed in several steps. In each step we already know some vectors of a basis and try to find a new one.

Definition 7.2.7 A sequence $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ ($0 \leq k \leq r$) of vectors in a r -dimensional lattice $\Lambda \subseteq \mathbb{R}^r$ can be *extended* to a basis of Λ if there exists a basis of Λ of the form $(\mathbf{a}_1, \dots, \mathbf{a}_{k-1}, \mathbf{a}_k, \dots, \mathbf{a}_r)$.

Clearly, the empty sequence ($k = 0$) and every basis of Λ can be extended to a basis. We shall also use the following characterization given in [19].

Lemma 7.2.8 Let Λ be a r -dimensional lattice in \mathbb{R}^r . A sequence $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ ($0 \leq k \leq r$) of lattice vectors in Λ can be extended to a basis of Λ if and only if every vector $\mathbf{c} \in \Lambda - \{\mathbf{0}\}$ of the shape

$$\mathbf{c} = \sum_{i=1}^k u_i \mathbf{a}_i$$

with real u_1, \dots, u_k necessarily has u_1, \dots, u_k integral.

Lemma 7.2.9 Let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ ($0 \leq k \leq r - 1$) be a sequence of vectors in a r -dimensional lattice Λ in \mathbb{R}^r , that can be extended to a basis of Λ , and let B_k be an orthonormal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Then for all $\epsilon \in \mathbb{R}$, $1/2 \leq \epsilon \leq 1$, the set $\mathcal{S}(\Lambda, B_k, \epsilon)$ is not empty, and for all $\mathbf{a}_{k+1} \in \mathcal{S}(\Lambda, B_k, \epsilon)$ the sequence $(\mathbf{a}_1, \dots, \mathbf{a}_k, \mathbf{a}_{k+1})$ can be extended to a basis of Λ .

Proof. First we have to show that $\mathcal{S}(\Lambda, B_k, \epsilon)$ is not the empty set. For this we note that by Corollary 7.2.6 there is a vector $\mathbf{b} \in \Lambda$ with $|\mathbf{b}|_{B_k} > 0$. Otherwise the dimension of Λ would be less than r . Hence by Proposition 7.2.5 we see that the infimum of the set

$$\left\{ z : z = |\mathbf{b}|_{B_k}, \mathbf{b} \in \Lambda, |\mathbf{b}|_{B_k} > 0 \right\}$$

is greater than $\lambda_1(\Gamma)/\sqrt{r-k}$, where Γ is the lattice that we obtain by projecting all vectors of Λ onto the space $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. This implies that $\mathcal{S}(\Lambda, B_k, \epsilon) \neq \emptyset$.

Let \mathbf{a}_{k+1} belong to $\mathcal{S}(\Lambda, B_k, \epsilon)$ and let us assume that the sequence $(\mathbf{a}_1, \dots, \mathbf{a}_k, \mathbf{a}_{k+1})$ can not be extended to a basis of Λ . Then by Lemma 7.2.8 there is a vector $\mathbf{c} \in \Lambda$, $\mathbf{c} \neq \mathbf{0}$, of the shape $\mathbf{c} = \sum_{i=1}^{k+1} u_i \mathbf{a}_i$, where $u_1, \dots, u_{k+1} \in \mathbb{R}$ and for some j , $1 \leq j \leq k+1$, we have

$$u_j \notin \mathbb{Z}. \tag{7.1}$$

Since $\mathbf{a}_1, \dots, \mathbf{a}_{k+1}$ and \mathbf{c} are vectors in Λ , the vector

$$\mathbf{c}' = \sum_{i=1}^{k+1} u_i \mathbf{a}_i - \sum_{i=1}^{k+1} \lceil u_i \rceil \mathbf{a}_i = \sum_{i=1}^{k+1} (u_i - \lceil u_i \rceil) \mathbf{a}_i$$

belongs to Λ . We show that $|\mathbf{c}'|_{B_k} > 0$. Suppose, in contrary, that $|\mathbf{c}'|_{B_k} = 0$. Then by Corollary 7.2.6 we have $\mathbf{c}' \in \text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k) \cap \Lambda$. Since $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ can be extended to a basis of Λ , by Lemma 7.2.8 it follows that $u_i - \lceil u_i \rceil \in \mathbb{Z}$, and therefore $u_i \in \mathbb{Z}$ for $1 \leq i \leq k+1$. But this is in contradiction to (7.1). Hence $|\mathbf{c}'|_{B_k} > 0$. Therefore we have $|\mathbf{c}'|_{B_k} > \epsilon |\mathbf{a}_{k+1}|_{B_k}$. On the other hand we obtain

$$\begin{aligned} |\mathbf{c}'|_{B_k} &= \max \{z: z = |\langle \mathbf{c}', \mathbf{b} \rangle|, \mathbf{b} \in B_k\} \\ &= \max \left\{ z: z = \left| \sum_{i=1}^{k+1} (u_i - \lceil u_i \rceil) \langle \mathbf{a}_i, \mathbf{b} \rangle \right|, \mathbf{b} \in B_k \right\} \\ &= |u_{k+1} - \lceil u_{k+1} \rceil| \max \{z: z = |\langle \mathbf{a}_{k+1}, \mathbf{b} \rangle|, \mathbf{b} \in B_k\} \\ &\leq \frac{1}{2} \max \{z: z = |\langle \mathbf{a}_{k+1}, \mathbf{b}_i \rangle|, \mathbf{b} \in B_k\} \\ &\leq \epsilon |\mathbf{a}_{k+1}|_{B_k}, \end{aligned}$$

which is clearly a contradiction. \square

Lemma 7.2.10 *Let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ ($0 \leq k \leq r-1$) be a sequence of vectors in a r -dimensional lattice Λ in \mathbb{R}^r , that can be extended to a basis of Λ , and let B_k be an orthonormal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Let $\delta \in \mathbb{R}$, $1/2 \leq \delta \leq 1$. Then for every vector \mathbf{b} in Λ there exists a vector \mathbf{d} in Λ such that $|\mathbf{b}|_{B_k} = |\mathbf{d}|_{B_k}$ and $\mathbf{d} \in \mathcal{P}_\delta(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$.*

Proof. Let $\mathbf{b} \in \Lambda \subseteq \mathbb{R}^r$. Since the combination of $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ and B_k is a basis of \mathbb{R}^r from (3.8) it follows that there are uniquely determined $x_{\mathbf{a}} \in \mathbb{R}$ for all $\mathbf{a} \in B_k$ such that

$$\mathbf{b} = \sum_{i=1}^k \frac{\langle \mathbf{b}, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \mathbf{a}_i^* + \sum_{\mathbf{a} \in B_k} x_{\mathbf{a}} \mathbf{a}. \quad (7.2)$$

If we set

$$\mathbf{d} = \mathbf{b} - \sum_{i=1}^k \left[\frac{\langle \mathbf{b}, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \right] \mathbf{a}_i,$$

then \mathbf{d} belongs to Λ . Moreover, we also know that \mathbf{d} is of the shape

$$\mathbf{d} = \sum_{i=1}^k \frac{\langle \mathbf{d}, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \mathbf{a}_i^* + \sum_{\mathbf{a} \in B_k} x_{\mathbf{a}} \mathbf{a},$$

where $x_{\mathbf{a}} \in \mathbb{R}$ for all $\mathbf{a} \in B_k$. Hence, $|\mathbf{b}|_{B_k} = \max\{z: z = x_{\mathbf{a}}, \mathbf{a} \in B_k\} = |\mathbf{d}|_{B_k}$.

Furthermore, by (7.2) and (3.7) (first part) we have for $1 \leq j \leq k$

$$\left| \frac{\langle \mathbf{d}, \mathbf{a}_j^* \rangle}{\langle \mathbf{a}_j^*, \mathbf{a}_j^* \rangle} \right| = \left| \frac{\langle \mathbf{b}, \mathbf{a}_j^* \rangle}{\langle \mathbf{a}_j^*, \mathbf{a}_j^* \rangle} - \sum_{i=1}^k \left[\frac{\langle \mathbf{b}, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_i^*, \mathbf{a}_i^* \rangle} \right] \frac{\langle \mathbf{a}_j, \mathbf{a}_i^* \rangle}{\langle \mathbf{a}_j^*, \mathbf{a}_j^* \rangle} \right| = \left| \frac{\langle \mathbf{b}, \mathbf{a}_j^* \rangle}{\langle \mathbf{a}_j^*, \mathbf{a}_j^* \rangle} - \left[\frac{\langle \mathbf{b}, \mathbf{a}_j^* \rangle}{\langle \mathbf{a}_j^*, \mathbf{a}_j^* \rangle} \right] \right| \leq \frac{1}{2}. \quad \square$$

Combining Lemma 7.2.9 and Lemma 7.2.10 we obtain

Corollary 7.2.11 *Let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ ($0 \leq k \leq r - 1$) be a sequence of vectors in a r -dimensional lattice Λ in \mathbb{R}^r , that can be extended to a basis of Λ , and let B_k be an orthonormal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Then for all $\epsilon, \delta \in \mathbb{R}$, $1/2 \leq \epsilon \leq 1$, $1/2 \leq \delta \leq 1$, the intersection $\mathcal{S}(\Lambda, B_k, \epsilon) \cap \mathcal{P}_\delta(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$ is not empty, and for all $\mathbf{a}_{k+1} \in \mathcal{S}(\Lambda, B_k, \epsilon) \cap \mathcal{P}_\delta(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$ the sequence $(\mathbf{a}_1, \dots, \mathbf{a}_k, \mathbf{a}_{k+1})$ can be extended to a basis of the lattice Λ .*

Clearly, Corollary 7.2.11 immediately implies

Theorem 7.2.12 *For every r -dimensional lattice Λ in \mathbb{R}^r and for every $\epsilon, \delta \in \mathbb{R}$, $1/2 \leq \epsilon \leq 1$, $1/2 < \delta \leq 1$ there exists a basis of Λ that is (ϵ, δ) -constructable.*

We also have by Corollary 7.2.11

Corollary 7.2.13 *Let $\epsilon, \delta \in \mathbb{R}$, $1/2 \leq \epsilon \leq 1$, $1/2 \leq \delta \leq 1$. Then each (ϵ, δ) -constructable sequence of a r -dimensional lattice Λ in \mathbb{R}^r can be extended to a basis of Λ .*

The above ideas lead to the following algorithm FUNDAMENTAL that on input of an order \mathcal{O} , and $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1$, $1/2 < \delta < 1$ finds a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that $(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r)$ is a (ϵ, δ) -constructable basis of $\text{Log } \mathcal{O}^*$. FUNDAMENTAL uses the subroutine NEXTVECTOR that on input of $k \in \mathbb{N}$, $0 \leq k \leq r - 1$, ϵ, δ , and k units $\varepsilon_1, \dots, \varepsilon_k \in \mathcal{O}^*$ of \mathcal{O} such that $(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_k)$ is a (ϵ, δ) -constructable sequence in $\text{Log } \mathcal{O}^*$, determines a unit ε_{k+1} such that $\text{Log } \varepsilon_{k+1} \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, \epsilon) \cap \mathcal{P}_\delta((\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*)$, where B_k is an arbitrary orthonormal basis of $\text{span}(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_k)^\perp$ and $(\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*$ are the Gram-Schmidt vectors of the vectors $\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_k$.

Algorithm 7.2.14 (FUNDAMENTAL)

Input : an order \mathcal{O} ; $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1$, $1/2 < \delta < 1$
Output : a system $\{\varepsilon_1, \dots, \varepsilon_r\}$ of fundamental units of \mathcal{O} such that
 $(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r)$ is a (ϵ, δ) -constructable basis of $\text{Log } \mathcal{O}^*$

- (1) **procedure** FUNDAMENTAL ($\mathcal{O}, \epsilon, \delta$)
- (2) **for** ($k := 0$ **to** $r - 1$ **step** 1) **do**
- (3) $\varepsilon_{k+1} := \text{NEXTVECTOR}(\mathcal{O}, k, \epsilon, \delta, \varepsilon_1, \dots, \varepsilon_k)$
- (4) **od**
- (5) **end procedure**

Applying induction and Corollary 7.2.11 yield

Corollary 7.2.15 *Algorithm 7.2.14 is correct, provided that NEXTVECTOR is correct.*

We note that in [6] a $(1/2, 1/2)$ -constructable basis of a lattice in \mathbb{R}^r is described. Since in real computations one can only find approximations of the real coefficients of the

lattice vectors one can only compute (ϵ, δ) -constructable bases where $\epsilon - 1/2$ and $\delta - 1/2$ is arbitrary small but always greater than 0. This shall be explained in more detail in the description of the procedure NEXTVECTOR. But before we investigate those problems in greater detail we summarize, for further reference, some technical but also important properties of (ϵ, δ) -constructable sequences.

Theorem 7.2.16 *Let $\epsilon, \delta \in \mathbb{R}$, $1/2 \leq \epsilon \leq 1, 1/2 \leq \delta \leq 1$, and let $(\mathbf{a}_1, \dots, \mathbf{a}_\ell)$ ($1 \leq \ell \leq r$) be a (ϵ, δ) -constructable sequence of a r -dimensional lattice Λ in \mathbb{R}^r . For k with $0 \leq k \leq \ell - 1$ let Λ_k be the lattice with basis $(\mathbf{a}_1, \dots, \mathbf{a}_k)$, and denote by Γ_k the projection of Λ onto the space $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Also, for $\mathbf{a} \in \Lambda$ let $\pi_k(\mathbf{a})$ be the projection of \mathbf{a} onto $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Then we have*

$$\epsilon^2 \|\pi_k(\mathbf{a}_{k+1})\|_2^2 \leq (r - k)\lambda_1(\Gamma_k)^2 \leq (r - k)\lambda_{k+1}(\Lambda)^2, \quad (7.3)$$

$$\epsilon^2 \|\mathbf{a}_{k+1}\|_2^2 \leq \delta^2 r(k + 2)\lambda_{k+1}(\Lambda)^2, \quad (7.4)$$

$$\epsilon^{k+1} \prod_{j=1}^{k+1} \|\mathbf{a}_j\|_2 \leq (\delta^2 r \gamma_{k+1})^{\frac{k+1}{2}} \det(\Lambda_{k+1}) \prod_{j=1}^{k+1} (j + 2)^{\frac{1}{2}}, \quad (7.5)$$

$$\lambda_1(\Gamma_k) \sqrt{r - k} (\delta^2 r \gamma_{k+1})^{\frac{k+1}{2}} \prod_{j=1}^{k+1} (j + 2)^{\frac{1}{2}} \geq \epsilon^{k+2} \lambda_1(\Lambda). \quad (7.6)$$

Proof. Let \mathbf{b} be a vector in Λ such that $\|\pi_k(\mathbf{b})\|_2 = \lambda_1(\Gamma_k)$. Then by Proposition 7.2.5 we have

$$\lambda_1(\Gamma_k) = \|\pi_k(\mathbf{b})\|_2 \geq \|\mathbf{b}\|_{B_k} \geq \epsilon \|\mathbf{a}_{k+1}\|_{B_k} \geq \frac{\epsilon}{\sqrt{r - k}} \|\pi_k(\mathbf{a}_{k+1})\|_2.$$

Next, we note that in Λ there are $k + 1$ linearly independent vectors in Λ of length bounded by $\lambda_{k+1}(\Lambda)$. At least one of them is of height greater than 0 with respect to B_k , since otherwise, by Corollary 7.2.6 there would be $k + 1$ linearly independent vectors in $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)$. Thus, there exists a vector $\mathbf{c} \in \Lambda$, such that

$$\lambda_1(\Gamma_k) \leq \|\pi_k(\mathbf{c})\|_2 \leq \|\mathbf{c}\|_2 \leq \lambda_{k+1}(\Lambda).$$

This proves (7.3).

To prove (7.4) we combine Definition 7.2.3 with (7.3) and Lemma 3.5.9. Then we obtain

$$\begin{aligned} \epsilon^2 \|\mathbf{a}_{k+1}\|_2^2 &\leq \epsilon^2 \|\pi_k(\mathbf{a}_{k+1})\|_2^2 + \epsilon^2 \delta^2 \sum_{i=1}^k \|\mathbf{a}_i^*\|_2^2 \\ &= \epsilon^2 \|\pi_k(\mathbf{a}_{k+1})\|_2^2 + \epsilon^2 \delta^2 \sum_{i=2}^k \|\pi_{i-1}(\mathbf{a}_i)\|_2^2 + \epsilon^2 \delta^2 \|\pi_0(\mathbf{a}_1)\|_2^2 \\ &\leq (r - k)\lambda_{k+1}(\Lambda)^2 + \delta^2 \sum_{i=2}^k (r - i)\lambda_i(\Lambda)^2 + \delta^2 r \lambda_1(\Lambda)^2 \\ &\leq \delta^2 r(k + 2)\lambda_{k+1}(\Lambda)^2. \end{aligned}$$

Since for $1 \leq j \leq k+1$ obviously $\lambda_j(\Lambda) \leq \lambda_j(\Lambda_{k+1})$, we can prove (7.5) by the following chain of inequalities,

$$(\epsilon)^{k+1} \prod_{j=1}^{k+1} \|\mathbf{a}_j\|_2 \leq \prod_{j=1}^{k+1} (\delta^2 r (j+2))^{\frac{1}{2}} \lambda_j(\Lambda_{k+1}) \leq (\delta^2 r \gamma_{k+1})^{\frac{k+1}{2}} \det(\Lambda_{k+1}) \prod_{j=1}^{k+1} (j+2)^{\frac{1}{2}},$$

where the last inequality is a consequence of Theorem 3.5.7.

Finally, we note that by (7.3) and (7.5) and by the properties of the determinant (see [29]) we have

$$\begin{aligned} \lambda_1(\Gamma_k) (\delta^2 r \gamma_{k+1})^{\frac{k+1}{2}} \prod_{j=1}^{k+1} (j+2)^{\frac{1}{2}} &\geq \epsilon \|\pi_k(\mathbf{a}_{k+1})\|_2 \frac{(\delta^2 r \gamma_{k+1})^{\frac{k+1}{2}}}{\sqrt{r-k}} \prod_{j=1}^{k+1} (j+2)^{\frac{1}{2}} \\ &= \epsilon \frac{\det(\Lambda_{k+1})}{\det(\Lambda_k)} \frac{(\delta^2 r \gamma_{k+1})^{\frac{k+1}{2}}}{\sqrt{r-k}} \prod_{j=1}^{k+1} (j+2)^{\frac{1}{2}} \geq \frac{\epsilon^{k+2}}{\sqrt{r-k}} \|\mathbf{a}_{k+1}\|_2 \geq \frac{\epsilon^{k+2}}{\sqrt{r-k}} \lambda_1(\Lambda). \end{aligned}$$

This proves inequality (7.6). \square

Next we have to describe the implementation of the subroutine NEXTVECTOR hat on input of $k \in \mathbb{N}$, $0 \leq k \leq r-1$, $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1$, $1/2 < \delta < 1$, and k units $\varepsilon_1, \dots, \varepsilon_k \in \mathcal{O}^*$ such that $(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_k)$ is a (ϵ, δ) -constructable sequence in $\text{Log } \mathcal{O}^*$, determines $\varepsilon_{k+1} \in \mathcal{O}^*$ with $\text{Log } \varepsilon_{k+1} \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, \epsilon) \cap \mathcal{P}_\delta((\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*)$, where B_k is an arbitrary orthonormal basis of $\text{span}(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_k)^\perp$ and where the vectors $(\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*$ are the Gram-Schmidt vectors of $\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_k$.

The main problem is that we can not compute B_k or the Gram-Schmidt vectors $(\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*$, but only approximations to them. So we have to show how to compute an appropriate unit $\varepsilon_{k+1} \in \mathcal{O}^*$ only knowing those approximations. To be more precise we shall show a related result for more general lattices.

We start with the following simple observation: Since lattices are discrete sets we have by Lemma 3.5.12

Proposition 7.2.17 *If $q \in \mathbb{N}$, $q > -\log(\lambda_1(\Lambda)/(4\sqrt{r}))$, then each approximation Λ' of precision q to Λ is a discrete set such that the minimal distance between two elements of Λ' is greater than $\lambda_1(\Lambda)/2$. Moreover, each approximating function of precision q restricted on Λ is one-to-one.*

In what follows we shall always implicitly assume that the studied approximating functions are one-to-one on the given lattices. We can do this without loss of generality since the actual precisions will always be larger than the one suggested by the above proposition. For convenience, we introduce the following definition.

Definition 7.2.18 Let Λ be a r -dimensional lattice Λ in \mathbb{R}^r , and let $1 \leq k \leq r-1$. Let $\epsilon, \delta, \sigma, \eta \in \mathbb{R}_{>0}$. Then we define

$$q_2(\Lambda, k, \epsilon, \delta, \sigma, \eta) = q_1(\Lambda, \epsilon, \delta, \sigma) - \log \left(\frac{1}{2} \min \left\{ \sqrt{\eta}, \frac{\eta^2 \epsilon^{k+2} \lambda_1(\Lambda)}{\sigma + (1 + 2(\delta + \eta)) \frac{\sqrt{r-k}}{\epsilon} \lambda_{k+1}(\Lambda)} \right\} \right),$$

where

$$q_1(\Lambda, k, \epsilon, \delta) = -\log \left(\frac{\epsilon^{k+2} \lambda_1(\Lambda)}{\sqrt{r-k} (\delta^2 r \gamma_{k+1})^{\frac{k+1}{2}} \prod_{j=1}^{k+1} (j+2)^{\frac{1}{2}}} \right).$$

In what follows let $\epsilon, \delta \in \mathbb{R}$, $1/2 < \epsilon < 1, 1/2 < \delta < 1$, and let $(\mathbf{a}_1, \dots, \mathbf{a}_\ell)$ ($1 \leq \ell \leq r$) be a (ϵ, δ) -constructable sequence of a r -dimensional lattice Λ in \mathbb{R}^r . First we describe the effect of approximations to the set $\mathcal{P}_\delta(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$.

Lemma 7.2.19 Let $\epsilon, \delta \in \mathbb{R}$, $1/2 < \epsilon < 1, 1/2 < \delta < 1$, and let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ ($1 \leq k \leq r-1$) be a (ϵ, δ) -constructable sequence of a r -dimensional lattice Λ in \mathbb{R}^r . Let $\sigma \in \mathbb{R}_{>0}$ and $\mathbf{b} \in \Lambda$, with $\|\mathbf{b}\|_2 \leq \sigma$. Let $\eta \in \mathbb{R}$, $\eta > 0$. Finally, let $f: \mathbb{R}^r \rightarrow \mathbb{Q}^r$ be an approximating function of precision $q \in \mathbb{N}$ such that the vectors $f(\mathbf{a}_1^*), \dots, f(\mathbf{a}_k^*)$ are pairwise orthogonal. If $q > q_2(\Lambda, k, \epsilon, \delta, \sigma, \eta)$ then we have:

- (a) If $\mathbf{b} \in \mathcal{P}_{1/2}(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$ then $f(\mathbf{b}) \in \mathcal{P}_{1/2+\eta}(f(\mathbf{a}_1^*), \dots, f(\mathbf{a}_k^*))$.
- (b) If $f(\mathbf{b}) \in \mathcal{P}_{1/2+\eta}(f(\mathbf{a}_1^*), \dots, f(\mathbf{a}_k^*))$ then $\mathbf{b} \in \mathcal{P}_{1/2+2\eta}(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$.

Proof. Let B_k be an orthonormal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$ and let B'_k be an orthonormal basis of $\text{span}(f(\mathbf{a}_1), \dots, f(\mathbf{a}_k))^\perp$. Let $\mathbf{b} \in \mathcal{P}_{1/2}(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$, i.e.

$$\mathbf{b} = \sum_{j=1}^k x_j \mathbf{a}_j^* + \sum_{\mathbf{a} \in B_k} x_{\mathbf{a}} \mathbf{a}, \quad (7.7)$$

where $x_j \in \mathbb{R}$, $|x_j| \leq 1/2$ for $1 \leq j \leq k$, $x_{\mathbf{a}} \in \mathbb{R}$ for $\mathbf{a} \in B_k$. Since $B'_k \cup \{f(\mathbf{a}_1), \dots, f(\mathbf{a}_k)\}$ is a basis of \mathbb{R}^r , we also have

$$f(\mathbf{b}) = \sum_{j=1}^k y_j f(\mathbf{a}_j^*) + \sum_{\mathbf{a}' \in B'_k} x_{\mathbf{a}'} \mathbf{a}',$$

where $y_j \in \mathbb{R}$ for $1 \leq j \leq k$, $x_{\mathbf{a}'} \in \mathbb{R}$ for $\mathbf{a}' \in B'_k$. Thus, to prove (a), we only have to estimate y_j .

From (3.8) it follows that for $1 \leq j \leq k$

$$y_j = \frac{\langle f(\mathbf{b}), f(\mathbf{a}_j^*) \rangle}{\langle f(\mathbf{a}_j^*), f(\mathbf{a}_j^*) \rangle}. \quad (7.8)$$

We set $\mathbf{e} = f(\mathbf{b}) - \mathbf{b}$, and for $1 \leq j \leq k$ we set $\mathbf{e}_j = f(\mathbf{a}_j^*) - \mathbf{a}_j^*$. Then, using (7.8) we have

$$y_j = \frac{\langle \mathbf{b}, \mathbf{a}_j^* \rangle + \langle \mathbf{b}, \mathbf{e}_j \rangle + \langle \mathbf{e}, \mathbf{a}_j^* \rangle + \langle \mathbf{e}, \mathbf{e}_j \rangle}{\langle \mathbf{a}_j^* + \mathbf{e}_j, \mathbf{a}_j^* + \mathbf{e}_j \rangle}.$$

From (7.7) it follows that $\langle \mathbf{b}, \mathbf{a}_j^* \rangle = x_j \langle \mathbf{a}_j^*, \mathbf{a}_j^* \rangle$. Applying (3.11) we thus obtain

$$|y_j| \leq \frac{x_j \|\mathbf{a}_j^*\|_2^2 + \|\mathbf{b}\|_2 \|\mathbf{e}_j\|_2 + \|\mathbf{e}\|_2 \|\mathbf{a}_j^*\|_2 + \|\mathbf{e}\|_2 \|\mathbf{e}_j\|_2}{\left(\|\mathbf{a}_j^*\|_2 - \|\mathbf{e}_j\|_2\right)^2}. \quad (7.9)$$

On the other hand, Definition 7.2.18 and Theorem 7.2.16 imply that both $\|\mathbf{e}_j\|_2$ and $\|\mathbf{e}\|_2$ are at most

$$\frac{1}{2} \min \left\{ \sqrt{\eta} \|\mathbf{a}_j^*\|_2, \frac{\eta \|\mathbf{a}_j^*\|_2^2}{\|\mathbf{b}\|_2 + (1 + 2(\delta + \eta)) \|\mathbf{a}_j^*\|_2} \right\}. \quad (7.10)$$

Combining (7.9) and (7.10) we see that $y_j \leq 1/2 + \eta$. This proves part (a) of the theorem. The proof of (b) is absolutely analogous. \square

We also have to examine what happens with the set $\mathcal{S}(\Lambda, B_k, \epsilon)$ when we work with approximations. For further reference, we introduce a new abbreviation.

Definition 7.2.20 Let Λ be a r -dimensional lattice Λ in \mathbb{R}^r , and let $1 \leq k \leq r - 1$. Let $\epsilon, \delta, \sigma \in \mathbb{R}_{>0}$. Then we define

$$q_3(\Lambda, k, \epsilon, \delta, \sigma) = q_1(\Lambda, k, \epsilon, \delta) - \log \left(\frac{1 - \epsilon}{\sqrt{r} 2(\sigma + 2)} \right).$$

Lemma 7.2.21 Let $\epsilon, \delta \in \mathbb{R}$, $1/2 < \epsilon < 1$, $1/2 < \delta < 1$, and let $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ ($1 \leq k \leq r - 1$) be a (ϵ, δ) -constructable sequence of a r -dimensional lattice Λ in \mathbb{R}^r . Let B_k be an orthonormal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Furthermore, let $\sigma \in \mathbb{R}_{>0}$, $\sigma > (r + k + 2)^2 \lambda_{k+1}(\Lambda)^2 / \epsilon^2$. Let $f: \mathbb{R}^r \rightarrow \mathbb{Q}^r$ be an approximating function of precision $q \in \mathbb{N}$, let $B'_k = f(B_k)$ and let $\Lambda' = \{\mathbf{v}: \mathbf{v} = f(\mathbf{a}), \mathbf{a} \in \Lambda, \|\mathbf{a}\|_2 \leq \sigma\}$. Finally, let \mathbf{a}_{k+1} be a vector of Λ such that $\|\mathbf{a}_{k+1}\|_2 \leq \sigma$. If $q > q_3(\Lambda, k, \epsilon^2, \delta, \sigma)$ then we have:

- (a) If $\mathbf{a}_{k+1} \in \mathcal{S}(\Lambda, B_k, 1)$, then $f(\mathbf{a}_{k+1}) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon})$.
- (b) If $f(\mathbf{a}_{k+1}) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon})$, then $\mathbf{a}_{k+1} \in \mathcal{S}(\Lambda, B_k, \epsilon)$.

Proof. First, we show that for all $\mathbf{a} \in \Lambda$ with $\|\mathbf{a}\|_2 \leq \sigma$ we have

$$\frac{2}{1 + \epsilon} |f(\mathbf{a})|_{B'_k} > |\mathbf{a}|_{B_k} \geq \frac{2\sqrt{\epsilon}}{1 + \epsilon} |f(\mathbf{a})|_{B'_k}. \quad (7.11)$$

To do so, let $B'_k = (\mathbf{a}'_{k+1}, \dots, \mathbf{a}'_r)$ and $B_k = (\widehat{\mathbf{a}}_{k+1}, \dots, \widehat{\mathbf{a}}_r)$. Applying the triangular inequality and Schwarz inequality we obtain for $k+1 \leq i \leq r$

$$\begin{aligned} |\langle f(\mathbf{a}), \mathbf{a}'_i \rangle| &\geq |\langle \mathbf{a}, \widehat{\mathbf{a}}_i \rangle| - |\langle \mathbf{a}, \mathbf{a}'_i - \widehat{\mathbf{a}}_i \rangle| - |\langle f(\mathbf{a}) - \mathbf{a}, \mathbf{a}'_i \rangle| \\ &\geq |\langle \mathbf{a}, \widehat{\mathbf{a}}_i \rangle| - \|\mathbf{a}\|_2 \|\mathbf{a}'_i - \widehat{\mathbf{a}}_i\|_2 - \|f(\mathbf{a}) - \mathbf{a}\|_2 \|\mathbf{a}'_i\|_2. \end{aligned} \quad (7.12)$$

By our assumptions and Lemma 4.2.1 we may assume that for $k+1 \leq i \leq r$ we have $\|\widehat{\mathbf{a}}_i - \mathbf{a}'_i\|_2 \leq \sqrt{r}2^{-q}$. If we insert this estimation in (7.12), then we can derive from the definition of the height (see Definition 7.2.1) and from Theorem 7.2.16 and Definition 7.2.20 that

$$|f(\mathbf{a})|_{B'_k} > |\mathbf{a}|_{B_k} - \|\mathbf{a}\|_2 \sqrt{r}2^{-q} - \sqrt{r}2^{-q}(1 + \sqrt{r}2^{-q}) \geq \frac{1+\epsilon}{2} |\mathbf{a}|_{B_k}.$$

This proves the left inequality of (7.11); the right one can analogously be proven.

To prove (a) we first assume that $f(\mathbf{a}_{k+1}) \notin \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon})$. That means that there exists $\mathbf{a} \in \Lambda$, $\|\mathbf{a}\|_2 \leq s$ such that $|f(\mathbf{a})|_{B'_k} < \sqrt{\epsilon} |f(\mathbf{a}_{k+1})|_{B'_k}$. Thus by (7.11) we have

$$\frac{1+\epsilon}{2} |\mathbf{a}_{k+1}|_{B_k} \geq \sqrt{\epsilon} |f(\mathbf{a}_{k+1})|_{B'_k} \geq |f(\mathbf{a})|_{B'_k} > \frac{1+\epsilon}{2} |\mathbf{a}|_{B_k}.$$

Therefore, \mathbf{a}_{k+1} does not belong to $\mathcal{S}(\Lambda, B_k, 1)$.

To prove (b), let $f(\mathbf{a}_{k+1}) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon})$, and suppose that $\mathbf{a}_{k+1} \notin \mathcal{S}(\Lambda, B_k, \epsilon)$. That means that there exists a vector \mathbf{a} of Λ with $\epsilon |\mathbf{a}_{k+1}|_{B_k} > |\mathbf{a}|_{B_k} > 0$. By Lemma 7.2.10 and Lemma 3.5.9 we may assume that

$$\|\mathbf{a}\|_2^2 \leq \frac{1}{4} \sum_{j=1}^k \|\pi_{j-1}(\mathbf{a}_j)\|_2^2 + \|\pi_k(\mathbf{a})\|_2^2, \quad (7.13)$$

where $\pi_j(\mathbf{a})$ is the projection of \mathbf{a} onto $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_j)^\perp$ ($0 \leq j \leq k$). Then applying Theorem 7.2.16 and Proposition 7.2.5 we obtain

$$\begin{aligned} \|\mathbf{a}\|_2^2 &\leq \frac{1}{4\epsilon^2} \sum_{j=2}^k (r-j)\lambda_j(\Lambda)^2 + \frac{r}{4\epsilon^2} \lambda_1(\Lambda)^2 + \sqrt{r-k} |\mathbf{a}|_{B_k} \\ &\leq \frac{1}{4\epsilon^2} \sum_{j=2}^k (r-j)\lambda_j(\Lambda)^2 + \frac{r}{4\epsilon^2} \lambda_1(\Lambda)^2 + \epsilon \sqrt{r-k} \|\pi_k(\mathbf{a}_{k+1})\|_2^2 \\ &\leq \frac{1}{4\epsilon^2} \sum_{j=2}^k (r-j)\lambda_j(\Lambda)^2 + \frac{r}{4\epsilon^2} \lambda_1(\Lambda)^2 + \frac{(r-k)^{\frac{3}{2}}}{\epsilon} \lambda_{k+1}(\Lambda)^2 \\ &\leq \left(\frac{r+k+2}{\epsilon} \right)^2 \lambda_{k+1}(\Lambda)^2 \leq \sigma. \end{aligned} \quad (7.14)$$

By (7.11), it follows that

$$\frac{2\sqrt{\epsilon}}{1+\epsilon} |f(\mathbf{a})|_{B'_k} \geq \frac{2\sqrt{\epsilon}}{1+\epsilon} |f(\mathbf{a}_{k+1})|_{B'_k} > \epsilon |\mathbf{a}_{k+1}|_{B_k} \geq |\mathbf{a}|_{B_k} \geq \frac{2\sqrt{\epsilon}}{1+\epsilon} |f(\mathbf{a})|_{B'_k}.$$

But this is a contradiction. \square

For convenience, we introduce the following notation.

Definition 7.2.22 For a matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$ of rank r and for $c \in \mathbb{R}$, $0 < c < 1$, we define the values

$$q_4(\mathbf{A}, c) = \log(\|\mathbf{A}^{-1}\|_f) + 3 \log\left(\frac{3}{\sqrt{2}}(r+6)\right) - \min\left\{0, \log\left(\frac{r+6}{\|\mathbf{A}\|_f}\right)\right\} - \log(c)$$

and

$$q_5(\mathbf{A}, c) = q_4(\mathbf{A}, c) + \log(2 \|\mathbf{A}\|_f + 1).$$

Using the above lemmata we can now prove that the following implementation of the algorithm NEXTVECTOR is correct. NEXTVECTOR uses the procedure GIANT that on input of an order \mathcal{O} , $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1$, $1/2 < \delta < 1$, $p \in \mathbb{N}$, the Gram-Schmidt vectors $\mathbf{b}_1^*, \dots, \mathbf{b}_k^* \in \mathbb{Q}^r$ of a p -approximation $(\mathbf{b}_1, \dots, \mathbf{b}_k)$ of a (ϵ, δ) -constructable sequence $(\text{Log } \epsilon_1, \dots, \text{Log } \epsilon_k)$ in $\text{Log } \mathcal{O}^*$, and an orthogonal basis B'_k of $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_k)^\perp$ that is a p -approximation to an orthonormal basis of $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_k)^\perp$, looks for a unit $\epsilon_{k+1} \in \mathcal{O}^*$ with $\text{LAPPROX}(\mathcal{O}, \epsilon_{k+1}, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, where $\Lambda' = \{\mathbf{v}: \mathbf{v} = \text{LAPPROX}(\mathcal{O}, \epsilon, p), \epsilon \in \mathcal{O}^*, \|\text{Log } \epsilon\|_2 \leq \sigma\}$ with $\sigma \in \mathbb{N}$, $2(r+k+2)^2 \lambda_{k+1}(\text{Log } \mathcal{O}^*)^2 / \epsilon^2 > \sigma > (r+k+2)^2 \lambda_{k+1}(\text{Log } \mathcal{O}^*)^2 / \epsilon^2$.

Algorithm 7.2.23 (NEXTVECTOR)

Input : an order \mathcal{O} ; $k \in \mathbb{N}$ with $0 \leq k \leq r-1$; $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1$, $1/2 < \delta < 1$; $\epsilon_1, \dots, \epsilon_k \in \mathcal{O}^*$ such that $(\text{Log } \epsilon_1, \dots, \text{Log } \epsilon_k)$ is a (ϵ, δ) -constructable sequence in $\text{Log } \mathcal{O}^*$

Output : $\epsilon_{k+1} \in \mathcal{O}^*$ with $\text{Log } \epsilon_{k+1} \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, \epsilon) \cap \mathcal{P}_\delta((\text{Log } \epsilon_1)^*, \dots, (\text{Log } \epsilon_k)^*)$, where B_k is an arbitrary orthonormal basis of $\text{span}(\text{Log } \epsilon_1, \dots, \text{Log } \epsilon_k)^\perp$ and where the vectors $(\text{Log } \epsilon_1)^*, \dots, (\text{Log } \epsilon_k)^*$ are the Gram-Schmidt vectors of the vectors $\text{Log } \epsilon_1, \dots, \text{Log } \epsilon_k$

- (1) **procedure** NEXTVECTOR($\mathcal{O}, k, \epsilon, \delta, \epsilon_1, \dots, \epsilon_k$)
- (2) Compute $p \in \mathbb{N}$ with $p = \lceil 4(5n+9)(16n+5 \log |\Delta_{\mathcal{O}}| - (2n + \log |\Delta_{\mathcal{O}}| + 1) n \log(\epsilon)) - \log(\delta - 1/2) - \log(1 - \epsilon) + 50 \rceil$;
- (3) **for** ($i := 1$ **to** k **step** 1) **do**
- (4) $\mathbf{b}_i := \text{LAPPROX}(\mathcal{O}, \epsilon_i, p)$
- (5) **od**
- (6) Compute the Gram-Schmidt vectors $\mathbf{b}_1^*, \dots, \mathbf{b}_k^*$ and an orthogonal basis B'_k of $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_k)^\perp$ that is a p -approximation to an orthonormal basis of $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_k)^\perp$;
- (7) $\epsilon_{k+1} := \text{GIANT}(\mathcal{O}, k, \epsilon, \delta, p, \mathbf{b}_1^*, \dots, \mathbf{b}_k^*, B'_k)$;
- (8) **end procedure**

Lemma 7.2.24 NEXTVECTOR (Algorithm 7.2.23) is correct, provided that GIANT is correct.

Proof. Let the notations be as in Algorithm 7.2.23. For convenience, let $\mathbf{a}_i = \text{Log } \varepsilon_i$ for $1 \leq i \leq k$. Suppose that $B'_k = (\mathbf{b}'_{k+1}, \dots, \mathbf{b}'_r)$, and set $\widehat{\mathbf{b}}_i = \mathbf{b}'_i / \|\mathbf{b}'_i\|_2$ for $k+1 \leq i \leq r$. Then the vectors $\widehat{\mathbf{b}}_i$ form an orthonormal basis of $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_k)^\perp$. Let $\sigma \in \mathbb{N}$ with $2(r+k+2)^2 \lambda_{k+1} (\text{Log } \mathcal{O}^*)^2 / \epsilon^2 > \sigma > (r+k+2)^2 \lambda_{k+1} (\text{Log } \mathcal{O}^*)^2 / \epsilon^2$. Let $\mathbf{A} = [\mathbf{b}_1, \dots, \mathbf{b}_k, \widehat{\mathbf{b}}_{k+1}, \dots, \widehat{\mathbf{b}}_r] \in \mathbb{R}^{r \times r}$ and suppose for the moment that

$$p > q_5(\mathbf{A}, c), \quad (7.15)$$

where $c \in \mathbb{R}_{>0}$ is a real number such that

$$\log(c) = -\max\{q_2(\text{Log } \mathcal{O}^*, k, \epsilon, \delta, \sigma, \delta/2 - 1/4), q_3(\text{Log } \mathcal{O}^*, k, \epsilon^2, \delta, \sigma)\} - 2. \quad (7.16)$$

We show that there exists $\varepsilon_{k+1} \in \mathcal{O}^*$ with $\text{LAPPROX}(\mathcal{O}, \varepsilon_{k+1}, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, where $\Lambda' = \{\mathbf{v}: \mathbf{v} = \text{LAPPROX}(\mathcal{O}, \varepsilon, p), \varepsilon \in \mathcal{O}^*, \|\text{Log } \varepsilon\|_2 \leq \sigma\}$, and that each such unit satisfies $\text{Log } \varepsilon_{k+1} \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, \epsilon) \cap \mathcal{P}_\delta((\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*)$. Then it follows that if the algorithm GIANT is correct, it will find such a unit which implies that NEXTVECTOR is correct.

The matrix \mathbf{A} is of full rank. Hence, there exists a QR factorization $\mathbf{A} = \mathbf{Q}\mathbf{R}$, where $\mathbf{R} = (r_{i,j}) \in \mathbb{R}^{r \times r}$ and $\mathbf{Q} \in \mathbb{R}^{r \times r}$. We denote the i -th column of \mathbf{Q} by \mathbf{q}_i ($1 \leq i \leq r$). Then for $1 \leq i \leq k$, we have $\mathbf{q}_i = \mathbf{b}_i^* / \|\mathbf{b}_i^*\|_2$, and $r_{i,i} = \|\mathbf{b}_i^*\|_2$; thus we obtain

$$r_{i,i} \mathbf{q}_i = \mathbf{b}_i^*. \quad (7.17)$$

For $k+1 \leq i \leq r$ we have $\mathbf{q}_i = \widehat{\mathbf{b}}_i$ (see [27]).

By our assumptions the matrix $\mathbf{A}' = [\mathbf{a}_1, \dots, \mathbf{a}_k, \widehat{\mathbf{b}}_{k+1}, \dots, \widehat{\mathbf{b}}_r] \in \mathbb{R}^{r \times r}$ is an approximation to \mathbf{A} of precision at least $q_5(\mathbf{A}, c)$. Thus, by Theorem 4.2.4 there exist matrices $\mathbf{W} \in \mathbb{R}^{r \times r}$ and $\mathbf{F} = (f_{i,j}) \in \mathbb{R}^{r \times r}$ with

$$\|\mathbf{W}\|_f \leq \frac{c}{2 \|\mathbf{A}\|_f + 1}$$

and

$$\|\mathbf{F}\|_f \leq \frac{c}{2 \|\mathbf{A}\|_f + 1},$$

such that the QR factorization of \mathbf{A}' is of the form $\mathbf{A}' = (\mathbf{Q} + \mathbf{W})(\mathbf{R} + \mathbf{F})$. For $1 \leq i \leq r$ we denote the i -th column of \mathbf{W} by \mathbf{w}_i . Then the vectors $\mathbf{a}_i^* = (r_{i,i} + f_{i,i})(\mathbf{q}_i + \mathbf{w}_i)$, $1 \leq i \leq k$, are the Gram-Schmidt vectors of the vectors \mathbf{a}_i . Let $\mathbf{v}_i = f_{i,i}(\mathbf{q}_i + \mathbf{w}_i) + r_{i,i} \mathbf{w}_i$. Then (7.17) implies that $\mathbf{b}_i^* + \mathbf{v}_i = (r_{i,i} + f_{i,i})(\mathbf{q}_i + \mathbf{w}_i) = \mathbf{a}_i^*$, and

$$\|\mathbf{v}_i\|_2 \leq c \frac{2 \|\mathbf{A}\|_f + c}{2 \|\mathbf{A}\|_f + 1} \leq c.$$

Thus \mathbf{a}_i^* is a q -approximation to \mathbf{b}_i^* where $q = -\lfloor \log(c) \rfloor$.

Clearly, the sequence $B_k = (\widehat{\mathbf{b}}_{k+1} + \mathbf{w}_{k+1}, \dots, \widehat{\mathbf{b}}_r + \mathbf{w}_r)$, where \mathbf{w}_i is the i -th column vector of \mathbf{W} ($k+1 \leq i \leq r$), forms an orthonormal basis of $\text{span}(\mathbf{a}_1, \dots, \mathbf{a}_k)^\perp$. Moreover,

for $k+1 \leq i \leq r$ the vector $\widehat{\mathbf{b}}_i + \mathbf{w}_i$ is a q -approximation to $\widehat{\mathbf{b}}_i$. Thus B_k is a $(q-1)$ -approximation to B'_k .

Now, let $f: \mathbb{R}^r \rightarrow \mathbb{Q}^r$ be an approximating function of precision $q-1$ with $f(B_k) = B'_k$ and $f(\mathbf{a}_i^*) = \mathbf{b}_i^*$ ($1 \leq i \leq k$). (By the above observations such a function exists.) Let $\Lambda' = \{\mathbf{v}: \mathbf{v} = f(\mathbf{a}), \mathbf{a} \in \Lambda, \|\mathbf{a}\|_2 \leq \sigma\}$. Since we may assume that $(\mathbf{a}_1, \dots, \mathbf{a}_k)$ is (ϵ, δ) -constructable, we can apply Corollary 7.2.11, which implies that there exists a vector $\mathbf{a}_{k+1} \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, 1) \cap \mathcal{P}_{1/2}(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*)$. Then from Lemma 7.2.19 and Lemma 7.2.21 it follows that

$$f(\mathbf{a}_{k+1}) = g(\mathbf{a}_{k+1}) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(f(\mathbf{a}_1^*), \dots, f(\mathbf{a}_k^*)).$$

On the other hand, the same lemmata imply that for each vector $\mathbf{a}_{k+1} \in \text{Log } \mathcal{O}^*$ with $f(\mathbf{a}_{k+1}) = g(\mathbf{a}_{k+1}) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(f(\mathbf{a}_1^*), \dots, f(\mathbf{a}_k^*))$ we have

$$\mathbf{a}_{k+1} \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, \epsilon) \cap \mathcal{P}_\delta(\mathbf{a}_1^*, \dots, \mathbf{a}_k^*).$$

This implies that GIANT can always find an appropriate unit $\varepsilon_{k+1} \in \mathcal{O}^*$ with $\text{Log } \varepsilon_{k+1} = \mathbf{a}_{k+1} \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, \epsilon) \cap \mathcal{P}_\delta((\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*)$, provided that (7.15) is correct.

To verify (7.15) we shall compute an upper bound of $q_5(\mathbf{A}, c)$, that is smaller than p . Our bound shall be very crude, but sufficient for our purposes. By Corollary 3.5.19 we have

$$\begin{aligned} q_5(\mathbf{A}, c) &\leq \log \left(r \frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})} \right) + 3 \log(3(r+6)) \\ &\quad - \min \left\{ 0, \log \left(\frac{r+6}{\|\mathbf{A}\|_f} \right) \right\} - \log(c) + \log(2\|\mathbf{A}\|_f + 1). \end{aligned} \quad (7.18)$$

Thus we can find an upper bound of $q_5(\mathbf{A}, c)$ by computing upper bounds of $\|\mathbf{A}\|_f$, $\frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})}$, and $\log(c)$. We start by estimating $\|\mathbf{A}\|_f$. Clearly, from Lemma 4.2.1 and Theorem 7.2.16 it follows that

$$\begin{aligned} \|\mathbf{A}\|_f &\leq \sum_{i=1}^k \|\mathbf{b}_i\|_2 + \sum_{i=k+1}^r \left\| \widehat{\mathbf{b}}_i \right\|_2 \\ &\leq \sum_{i=1}^k (\|\mathbf{a}_i\|_2 + \sqrt{r}) + (r-k) \\ &\leq \left(\frac{\delta}{\epsilon} \right) k \sqrt{r(k+2)} \lambda_{k+1}(\text{Log } \mathcal{O}^*) + r^2. \end{aligned}$$

Thus, Proposition 3.5.30 implies that

$$\|\mathbf{A}\|_f \leq \left(\frac{\delta}{\epsilon} \right) (2r)^{r+2} 2^{(n+1)r} 4n^2 \sqrt{|\Delta_{\mathcal{O}}|} (\log |\Delta_{\mathcal{O}}|)^n (\log \log |\Delta_{\mathcal{O}}|)^{n/2} + r^2. \quad (7.19)$$

By (7.5), Proposition 3.5.29, and (3.5) we know that

$$\frac{\text{dft}(\mathbf{A}')}{\lambda(\mathbf{A}')} \leq \frac{1}{\epsilon^{k+1}} (\delta^2 r (k+1))^{\frac{k+1}{2}} \prod_{j=1}^{k+1} (j+2)^{\frac{1}{2}} \frac{17r}{16} \frac{1}{4^{2+t}}.$$

Hence, we have

$$p > r \log \left(\frac{3}{2} \right) + \frac{3 \log(r)}{2} + \log \left(\frac{\text{dft}(\mathbf{A}')}{\lambda(\mathbf{A}')} \right) + 3,$$

and therefore by (4.29)

$$\frac{\text{dft}(\mathbf{A})}{\lambda(\mathbf{A})} \leq \left(\frac{3}{2} \right)^r \frac{\text{dft}(\mathbf{A}')}{\lambda(\mathbf{A}')} \leq \left(\frac{3}{2} \right)^r \frac{1}{\epsilon^{k+1}} (\delta^2 r (k+1))^{\frac{k+1}{2}} \prod_{j=1}^{k+1} (j+2)^{\frac{1}{2}} \frac{17r}{16} \frac{1}{4^{2+t}}. \quad (7.20)$$

By Proposition 3.5.29, Proposition 3.5.30, and Theorem 7.2.16 and straightforward computations we can find bounds on $q_2(\text{Log } \mathcal{O}^*, k, \epsilon, \delta, \sigma, \delta/2 - 1/4)$ and $q_3(\text{Log } \mathcal{O}^*, k, \epsilon, \delta, \sigma)$, and therefore by (7.16) on $-\log(c)$. We obtain

$$\begin{aligned} q_2(\text{Log } \mathcal{O}^*, k, \epsilon, \delta, \sigma, \delta/2 - 1/4) &< 2(5n+8)(n+1 - \log(\epsilon)) \\ &\quad - \log(\delta - 1/2) + (3n+2)(2n + \log |\Delta_{\mathcal{O}}|)(2 - n \log(\epsilon)) + 7 \end{aligned}$$

and

$$q_3(\text{Log } \mathcal{O}^*, k, \epsilon^2, \delta, \sigma) < 2(5n+8)(n+1 - \log(\epsilon)) - \log(1 - \epsilon) + (3n+1) \log |\Delta_{\mathcal{O}}| + 9,$$

and finally by (7.16)

$$\begin{aligned} -\log(c) &\leq 2(5n+8)(n+1 - \log(\epsilon)) - \log(\delta - 1/2) \\ &\quad - \log(1 - \epsilon) + (3n+2)(2n + \log |\Delta_{\mathcal{O}}|)(2 - n \log(\epsilon)) + 11. \quad (7.21) \end{aligned}$$

Thus (7.18) and (7.19), (7.20), and (7.21) imply that

$$\begin{aligned} q_5(\mathbf{A}, c) &< 4(5n+9)(16n+5 \log |\Delta_{\mathcal{O}}| - (2n + \log |\Delta_{\mathcal{O}}| + 1)n \log(\epsilon)) \\ &\quad - \log(\delta - 1/2) - \log(1 - \epsilon) + 50. \quad (7.22) \end{aligned}$$

Clearly, this proves (7.15). \square

The implementation of the procedure GIANT uses the strategy described in [6, Sect. 9].

Before we describe GIANT in more detail we consider the following problem: Suppose that we have a finite set A of ideals of an order \mathcal{O} and an ideal \mathfrak{A} of \mathcal{O} , and we want to know whether \mathfrak{A} belongs to A . We can solve this problem by first “sorting” the set A and then using techniques like *binary search* (see [35]). To do so, we agree on the standard lexicographical order on the set of all ideals (or to be more precise, on the set of standard representations of the ideals). Then given two ideals, we can lexicographically compare

them in time linear bounded by their sizes. If the given set A contains a elements then we can sort it (with respect to the lexicographical order) using $O(a \log(a))$ comparisons. If \mathfrak{A} has been sorted then we can decide whether $\mathfrak{A} \in A$ using additionally $O(\log(a))$ comparisons (see [35]). Since these running times shall always be dominated by the running time of other parts of the following algorithms that contain such tests, we shall always implicitly assume that computed sets of ideals are sorted (with respect to the lexicographical order).

GIANT calls the procedure BABY, that works in the following way: Let us use the notation of Algorithm 7.2.23 and Lemma 7.2.24. Let $\mathbf{c}_1, \dots, \mathbf{c}_r \in \mathbb{Q}^r$ be vectors such that for $1 \leq j \leq k$ the vector \mathbf{c}_j has the form $\mathbf{c}_j = U_j \mathbf{b}_j^*$, where $U_j \in \mathbb{Q}$, and for $k+1 \leq j \leq r$ the vector \mathbf{c}_j has the form $\mathbf{c}_j = U \mathbf{b}'_j$, where $U \in \mathbb{Q}$. Let $D_0 := \lceil 2\sqrt{r}(\log |\Delta_{\mathcal{O}}| + 3)/4 + 2^{-p} \rceil$. For $1 \leq i \leq k$ let $\bar{\mathbf{b}}_i$ be a vector of the form $u_i \mathbf{b}_i^* / \|\mathbf{b}_i^*\|_2$, where u_i is a p -approximation to D_0 , and for $k+1 \leq j \leq r$ let $\bar{\mathbf{b}}_j := D_0 \mathbf{b}'_j$. Given \mathcal{O} , k , ϵ , δ , p , $\mathbf{c}_1, \dots, \mathbf{c}_r$, and $\bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_r$, the procedure BABY determines a pair (S, ε_{k+1}) . If there exists a unit $\varepsilon \in \text{Log } \mathcal{O}^*$ with $\text{LAPPROX}(\mathcal{O}, \varepsilon, p) \in \{\mathbf{w} : \mathbf{w} = \sum_{i=1}^r x_i (\mathbf{c}_i - \bar{\mathbf{b}}_i), x_i \in \mathbb{R}, |x_i| \leq 1/2 \text{ for } 1 \leq i \leq r\}$ and $\text{LAPPROX}(\mathcal{O}, \varepsilon, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, then ε_{k+1} is such a unit; otherwise we have $\varepsilon_{k+1} = 0$. In that case S is the set of all pairs $((1/\nu)\mathcal{O}, \nu)$ with $\nu \in \mathcal{O}$ being a minimum in binary multiplicative representation such that $\text{LAPPROX}(\mathcal{O}, \nu, p+1) \in \{\mathbf{w} : \mathbf{w} = \sum_{j=1}^r x_j \mathbf{c}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r\}$.

Algorithm 7.2.25 (GIANT)

Input : an order \mathcal{O} ; $k \in \mathbb{N}$ with $0 \leq k \leq r - 1$; $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1$, $1/2 < \delta < 1$; $p \in \mathbb{N}$ as computed in step (2) of Algorithm 7.2.23; the Gram-Schmidt vectors $\mathbf{b}_1^*, \dots, \mathbf{b}_k^*$ of a p -approximation $(\mathbf{b}_1, \dots, \mathbf{b}_k)$ of a (ϵ, δ) -constructable sequence $(\text{Log } \epsilon_1, \dots, \text{Log } \epsilon_k)$ in $\text{Log } \mathcal{O}^*$; an orthogonal basis $B'_k = (\mathbf{b}'_{k+1}, \dots, \mathbf{b}'_r)$ of $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_k)^\perp$ that is a p -approximation to an orthonormal basis of $\text{span}(\mathbf{b}_1, \dots, \mathbf{b}_k)^\perp$

Output : $\epsilon_{k+1} \in \mathcal{O}^*$ with $\text{LAPPROX}(\mathcal{O}, \epsilon_{k+1}, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, where $\Lambda' = \{\mathbf{v} : \mathbf{v} = \text{LAPPROX}(\mathcal{O}, \epsilon, p), \epsilon \in \mathcal{O}^*, \|\text{Log } \epsilon\|_2 \leq \sigma\}$ with $\sigma \in \mathbb{N}$, $2(r+k+2)^2 \lambda_{k+1}(\text{Log } \mathcal{O}^*)^2 / \epsilon^2 > \sigma > (r+k+2)^2 \lambda_{k+1}(\text{Log } \mathcal{O}^*)^2 / \epsilon^2$

- (1) **procedure** GIANT($\mathcal{O}, k, \epsilon, \delta, p, \mathbf{b}_1^*, \dots, \mathbf{b}_k^*, B'_k$)
- (2) $T := \emptyset; \epsilon_{k+1} := 0;$
- (3) $\delta' := \delta/2 + 1/4;$
- (4) $D_0 := \lceil 2\sqrt{r}(\log |\Delta_{\mathcal{O}}| + 3)/4 + 2^{-p} \rceil;$
- (5) **for** ($j := 1$ **to** k **step** 1) **do**
- (6) $M_j := \lceil (2\delta' \|\mathbf{b}_j^*\|_2)^{1/2} \rceil;$
- (7) $U_j := 2\delta' / (2M_j + 1);$
- (8) Compute a vector $\bar{\mathbf{b}}_j$ of the form $\bar{\mathbf{b}}_j = u_j \mathbf{b}_j^* / \|\mathbf{b}_j^*\|_2$, where u_j is a p -approximation to D_0 ;
- (9) $\tilde{\mathbf{b}}_j := U_j \mathbf{b}_j^*;$
- (10) **od**
- (11) **for** ($j := k + 1$ **to** r **step** 1) **do**
- (12) $\bar{\mathbf{b}}_j := D_0 \mathbf{b}'_j;$
- (13) **od**

```

(14)  $\ell := -1;$ 
(15) while  $(\varepsilon_{k+1} = 0)$  do
(16)    $\ell := \ell + 1;$ 
(17)    $D := 2^\ell D_0;$ 
(18)   for  $(j := k + 1$  to  $r$  step  $1)$  do
(19)      $U_j := D;$ 
(20)      $\tilde{\mathbf{b}}_j := U_j \mathbf{b}'_j;$ 
(21)   od
(22)    $(S, \varepsilon_{k+1}) := \text{BABY}(\mathcal{O}, k, \varepsilon, \delta, \tilde{\mathbf{b}}_1 + \bar{\mathbf{b}}_1, \dots, \tilde{\mathbf{b}}_r + \bar{\mathbf{b}}_r, \bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_r);$ 
(23)   if  $(\varepsilon_{k+1} = 0)$  then
(24)      $N := -1;$ 
(25)     while  $(N < D$  and  $\varepsilon_{k+1} = 0)$  do
(26)        $N := N + 1;$ 
(27)       for (every  $\mathbf{z} \in \mathbb{Z}^r$  with  $|\mathbf{z}_j| \leq M_j$  for  $1 \leq j \leq k$ ,  $\max\{a : a = |\mathbf{z}_j|, k + 1 \leq j \leq r\} = N$ ) do
(28)          $\mu := \text{CLOSE}(\mathcal{O}, \mathcal{O}, \sum_{j=1}^r \mathbf{z}_j \tilde{\mathbf{b}}_j);$ 
(29)         if (there exists  $(\mathfrak{B}, \nu) \in S$  with  $(1/\mu)\mathcal{O} = \mathfrak{B}$ ) then
(30)            $E := E \cup \{\text{COMPACT}(\mathcal{O}, \mu/\nu, \mathcal{O})\};$ 
(31)         fi
(32)       od
(33)       if  $(E \neq \emptyset)$  then
(34)         Choose  $\varepsilon \in E$  with  $\text{LAPPROX}(\mathcal{O}, \varepsilon, p) \in \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$ ,
           such that  $|\text{LAPPROX}(\mathcal{O}, \varepsilon, p)|_{B'_k} > 0$  is minimal;
(35)          $\varepsilon_{k+1} := \varepsilon;$ 
(36)       fi
(37)     od
(38)   fi
(39) od
(40) end procedure

```

Before we can show that the algorithm GIANT is correct we need

Lemma 7.2.26 *Let us use the notation of Algorithm 7.2.25. Fix ℓ , let $D = 2^\ell D_0$. Let*

$$\tilde{\mathcal{P}} = \left\{ \mathbf{u} : \mathbf{u} = \sum_{j=1}^r x_j \tilde{\mathbf{b}}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\},$$

and for $\mathbf{z} \in \mathbb{R}^r$ let

$$\tilde{\mathcal{P}}(\mathbf{z}) = \left\{ \mathbf{v} : \mathbf{v} = \mathbf{u} + \sum_{j=1}^r \mathbf{z}_j \tilde{\mathbf{b}}_j, \mathbf{u} \in \tilde{\mathcal{P}} \right\}.$$

Also, let

$$\mathcal{P}' = \left\{ \mathbf{u} : \mathbf{u} = \sum_{j=1}^r x_j (\tilde{\mathbf{b}}_j + \bar{\mathbf{b}}_j), x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\}.$$

Let ε be a unit of \mathcal{O} and $\widehat{\varepsilon}$ be a p -approximation to ε . Finally, let $\mathbf{z} \in \mathbb{Z}^r$, let μ be a minimum of \mathcal{O} that is $(3/4)$ -close to $\mathbf{m} = \sum_{j=1}^r \mathbf{z}_j \widetilde{\mathbf{b}}_j$, and let \mathbf{w} be a $(p+1)$ -approximation to $\text{Log}(\mu/\varepsilon)$. If $\widehat{\varepsilon} \in \widetilde{\mathcal{P}}(\mathbf{z})$ then we have $\mathbf{w} \in \mathcal{P}'$.

Proof. Since μ is $(3/4)$ -close to \mathbf{m} there exists a vector $\mathbf{f} \in \mathbb{Q}^r$ with $\|\mathbf{f}\|_\infty \leq (\log |\Delta_{\mathcal{O}}| + 3)/4 + 2^{p-1}$ such that $\mathbf{w} = \mathbf{m} - \widehat{\varepsilon} + \mathbf{f}$. Clearly, $\mathbf{m} - \widehat{\varepsilon} \in \widetilde{\mathcal{P}}$. To show, that $\mathbf{w} \in \mathcal{P}'$ we only have to show that there exist $x_1, \dots, x_r \in \mathbb{R}$ with $|x_j| \leq 1/2$ for $1 \leq j \leq r$ and $\mathbf{f} = \sum_{j=1}^r x_j \widetilde{\mathbf{b}}_j$. But this follows from (3.9) and Lemma 3.5.15 using estimations that are analogous to those used in the proofs in section 6.2. \square

Lemma 7.2.27 *If BABY works correctly then GIANT (Algorithm 7.2.25) is correct.*

Proof. We use the notation of Algorithm 7.2.25 and Lemma 7.2.26. We have to show that GIANT finds a unit $\varepsilon_{k+1} \in \text{Log } \mathcal{O}^*$ with $\text{LAPPROX}(\mathcal{O}, \varepsilon_{k+1}, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$.

Suppose that BABY is correct. If BABY determines in step (21) an element $\varepsilon_{k+1} \neq 0$ then GIANT shall output this element and the correctness follows from the correctness of BABY.

Thus, let us suppose that ε_{k+1} computed in step (21) equals zero. First we show that in that case GIANT terminates: By the proof of Algorithm 7.2.23 we know that there exists a unit $\varepsilon \in \mathcal{O}^*$ such that for an arbitrary p -approximation $\widehat{\varepsilon}$ to $\text{Log } \varepsilon$ we have $\widehat{\varepsilon} \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, and that $\text{Log } \varepsilon \in \mathcal{S}(\text{Log } \mathcal{O}^*, B_k, \epsilon) \cap \mathcal{P}_\delta((\text{Log } \varepsilon_1)^*, \dots, (\text{Log } \varepsilon_k)^*)$. Thus from (7.3) and (7.4), Proposition 7.2.5, (7.11) and (4.2) it follows that

$$|\widehat{\varepsilon}|_{B'_k} \leq \left(\frac{1+\epsilon}{2\sqrt{\epsilon}} \right) \left(\frac{\sqrt{r-k}\lambda_{k+1}(\text{Log } \mathcal{O}^*)}{\epsilon} \right). \quad (7.23)$$

(Note that (7.4) has allowed us to apply (7.11).)

Now, suppose that we are in the while-loop of Algorithm 7.2.25 from step (15) to step (39), where in step (17) we have

$$\frac{D^2}{4} = 4^{\ell-1} D_0^2 \leq 2 \left(\frac{1+\epsilon}{2\sqrt{\epsilon}} \right) \left(\frac{\sqrt{r-k}\lambda_{k+1}(\text{Log } \mathcal{O}^*)}{\epsilon} \right) < 4^\ell D_0^2 = D^2. \quad (7.24)$$

Since we have for $1 \leq j \leq k$

$$\widetilde{\mathbf{b}}_j = U_j \mathbf{b}_j^* = \frac{2\delta' \|\mathbf{b}_j^*\|_2}{2M_j + 1} \left(\frac{\mathbf{b}_j^*}{\|\mathbf{b}_j^*\|_2} \right),$$

we know that

$$\mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*) = \bigcup_{\substack{\mathbf{z} \in \mathbb{Z}^r \\ |\mathbf{z}_j| \leq M_j, 1 \leq j \leq k}} \widetilde{\mathcal{P}}(\mathbf{z}). \quad (7.25)$$

From (7.23), (7.24), and (7.25) it follows that there exists a vector $\mathbf{z} \in \mathbb{Z}^r$ with $|\mathbf{z}_j| \leq M_j$ for $1 \leq j \leq k$ and $|\mathbf{z}_j| \leq D$ for $k+1 \leq j \leq r$, such that $\widehat{\mathbf{e}} \in \widetilde{\mathcal{P}}(\mathbf{z})$.

Suppose that in step (27) of Algorithm 7.2.25 the variable ℓ satisfies (7.24) and that $N = \max\{a: a = |\mathbf{z}_j|, k+1 \leq j \leq r\}$. Then during the execution of the for-loop from step (27) to step (32) a minimum μ of \mathcal{O} is computed, which is $(3/4)$ -close to $\mathbf{m} = \sum_{j=1}^r \mathbf{z}_j \widetilde{\mathbf{b}}_j$. Clearly, from Lemma 3.4.4 and Proposition 5.1.3 it follows that the element $\nu = \mu/\varepsilon$ is a minimum of \mathcal{O} with $(1/\nu)\mathcal{O} = (1/\mu)\mathcal{O}$. Let \mathbf{w} be an arbitrary $(p+1)$ -approximation to $\text{Log } \nu$. Then by Lemma 7.2.26 we know that $\mathbf{w} \in \mathcal{P}'$, and therefore the set S contains the pair (\mathfrak{A}, ν) . Let $\mathbf{e} = \text{LAPPROX}(\mathcal{O}, \varepsilon, p)$. Since we have $\|\text{Log } \varepsilon\|_2 > 0$, from Lemma 4.2.1 and the choice of p it follows that $\|\mathbf{e}\|_2 \geq \sqrt{r}2^{-p}$. Thus, if the algorithm until now did not insert a unit in the set E then in step (30) the element $\text{COMPACT}(\mathcal{O}, \mu/\nu, \mathcal{O})$ is inserted in the set E . Hence the algorithm terminates. Next, we have to show that the output of GIANT always is correct.

Clearly from Lemma 3.4.4 it follows that each element that is output of GIANT is a unit of \mathcal{O} . By the above arguments we also know that only units ε are inserted in the set E with $\|\text{Log } \varepsilon\|_2 \leq \sigma$. Now, suppose that GIANT outputs the unit ε . We may assume that ε is found by the algorithm while executing the while-loop from step (15) to step (38) with ℓ having the value ℓ' and N having the value N_0 .

If $\text{LAPPROX}(\mathcal{O}, \varepsilon_{k+1}, p) \notin \mathcal{S}(\Lambda', B'_k, \sqrt{\varepsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, then we must have $\text{LAPPROX}(\mathcal{O}, \varepsilon, p) \notin \mathcal{S}(\Lambda', B'_k, \sqrt{\varepsilon})$. Thus, there exists a unit ε' with

$$|\text{LAPPROX}(\mathcal{O}, \varepsilon', p)|_{B'_k} < \varepsilon |\text{LAPPROX}(\mathcal{O}, \varepsilon, p)|_{B'_k}. \quad (7.26)$$

Let $\mathbf{z}' \in \mathbb{Z}^r$ be such that $\text{LAPPROX}(\mathcal{O}, \varepsilon', p) \in \mathcal{P}(\mathbf{z}')$. Then we must have

$$N_0 \geq \max\{a: a = |\mathbf{z}'_j|, k+1 \leq j \leq r\},$$

since otherwise we had a contradiction to (7.26). If

$$N_0 > \max\{a: a = |\mathbf{z}'_j|, k+1 \leq j \leq r\},$$

then the algorithm would stop with $N = \max\{a: a = |\mathbf{z}'_j|, k+1 \leq j \leq r\}$, as can be seen by the same argument that was used for showing that GIANT terminates. But this would be a contradiction to the definition of N_0 . On the other hand we can not have

$$N_0 = \max\{a: a = |\mathbf{z}_j|, k+1 \leq j \leq r\},$$

as can be seen as follows: If $\mathbf{z} \neq \mathbf{z}'$ then this would contradict step (34). But if $\mathbf{z} = \mathbf{z}'$ then we have $\text{LAPPROX}(\mathcal{O}, \varepsilon'/\varepsilon, p) \in \mathcal{P}'$ and BABY determines a correct unit. Thus we have shown that $\text{LAPPROX}(\mathcal{O}, \varepsilon, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\varepsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$. \square

Finally, we explain how the routine BABY works. For convenience, we use the notation of Algorithm 7.2.25. Note, that BABY uses CUBOID (Algorithm 6.2.20).

Algorithm 7.2.28 (BABY)

Input : an order \mathcal{O} ; k ; ϵ ; δ ; p ; $\mathbf{c}_1, \dots, \mathbf{c}_r \in \mathbb{Q}^r$ such that for $1 \leq j \leq k$ the vector \mathbf{c}_j has the form $\mathbf{c}_j = U_j \mathbf{b}_j^*$, where $U_j \in \mathbb{Q}$, and for $k+1 \leq j \leq r$ the vector \mathbf{c}_j has the form $\mathbf{c}_j = U \mathbf{b}_j'$, where $U \in \mathbb{Q}$; the vectors $\bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_r$;

Output : pair (S, ϵ_{k+1}) , that can be described as follows: if there exists a unit $\epsilon \in \text{Log } \mathcal{O}^*$ with $\text{LAPPROX}(\mathcal{O}, \epsilon, p) \in \{\mathbf{w} : \mathbf{w} = \sum_{i=1}^r x_i (\mathbf{c}_i - \mathbf{b}_i), x_i \in \mathbb{R}, |x_i| \leq 1/2 \text{ for } 1 \leq i \leq r\}$ and $\text{LAPPROX}(\mathcal{O}, \epsilon, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, then ϵ_{k+1} is such a unit; otherwise we have $\epsilon_{k+1} = 0$. In that case S is the set of all pairs $((1/\nu)\mathcal{O}, \nu)$ with $\nu \in \mathcal{O}$ being a minimum in binary multiplicative representation such that $\text{LAPPROX}(\mathcal{O}, \nu, p+1) \in \{\mathbf{w} : \mathbf{w} = \sum_{j=1}^r x_j \mathbf{c}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r\}$.

- (1) **procedure** BABY($\mathcal{O}, k, \epsilon, \delta, \mathbf{c}_1, \dots, \mathbf{c}_r, \bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_r$)
- (2) $S := \emptyset$; $\epsilon_{k+1} := 0$;
- (3) $h := -1$;
- (4) **while** ($h < \lceil (1/2)(\|\mathbf{c}_r\|_2 / \|\bar{\mathbf{b}}_r\|_2 - 1) \rceil$ and $\epsilon_{k+1} = 0$) **do**
- (5) $h := h + 1$;
- (6) **for** (every $\mathbf{z} \in \mathbb{Z}^r$ with $|\mathbf{z}_j| \leq \lceil (1/2)(\|\mathbf{c}_j\|_2 / \|\bar{\mathbf{b}}_j\|_2 - 1) \rceil$ for $1 \leq j \leq k$ and $\max\{a : a = |\mathbf{z}_j|, k+1 \leq j \leq r\} = h$) **do**
- (7) **for** (every $\nu \in \text{CUBOID}(\mathcal{O}, \mathcal{O}, \sum_{i=1}^r z_i \bar{\mathbf{b}}_i)$ with $\text{LAPPROX}(\mathcal{O}, \nu, p+1) \in \{\mathbf{w} : \mathbf{w} = \sum_{j=1}^r x_j \mathbf{c}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r\}$) **do**
- (8) **if** ($\mathcal{O} = (1/\nu)\mathcal{O}$) **then**
- (9) $E := E \cup \{\nu\}$
- (10) **else**
- (11) $S := S \cup \{(1/\nu)\mathcal{O}, \nu\}$;
- (12) **fi**
- (13) **od**
- (14) **if** ($E \neq \emptyset$) **then**
- (15) Choose $\epsilon \in E$ with $\text{LAPPROX}(\mathcal{O}, \epsilon, p) \in \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$, such that $|\text{LAPPROX}(\mathcal{O}, \epsilon, p)|_{B'_k} > 0$ is minimal;
- (16) $\epsilon_{k+1} := \epsilon$;
- (17) **fi**
- (18) **od**
- (19) **od**
- (20) **end procedure**

Lemma 7.2.29 *BABY (Algorithm 7.2.28) is correct. Its running time is*

$$\prod_{j=1}^k \left(2 \left\lceil \left(\frac{1}{2} \right) \left(\frac{\|\mathbf{c}_j\|_2}{\|\bar{\mathbf{b}}_j\|_2} - 1 \right) \right\rceil + 1 \right) \left(2 \left\lceil \left(\frac{1}{2} \right) \left(\frac{\|\mathbf{c}_r\|_2}{\|\bar{\mathbf{b}}_r\|_2} - 1 \right) \right\rceil + 1 \right)^{r-k} \left(n + p + \log(\Delta) + \log \left(\sum_{i=1}^r \|\mathbf{c}_i\|_2 \right) + \|\text{MT}(\Omega)\|_\infty \right)^{O(n)}.$$

Proof. It is easy to see that BABY terminates. Let the pair (S, ε_{k+1}) be the output of BABY. First, let us assume that $\varepsilon_{k+1} = 0$. Then S is the set of all pairs of the shape $((1/\nu)\mathcal{O}, \nu)$, where ν is found by a call of $\text{CUBOID}(\mathcal{O}, \mathcal{O}, \sum_{i=1}^r z_j \bar{\mathbf{b}}_j)$, where $\mathbf{z} \in \mathbb{Z}^r$ and $\|\mathbf{z}\|_\infty \leq \lceil (1/2)(\|\mathbf{c}\|_2/\|\bar{\mathbf{b}}_j\|_2 - 1) \rceil$. For $\mathbf{z} \in \mathbb{Z}^r$ let

$$\mathcal{Q}(\mathbf{z}) = \left\{ \mathbf{v} : \mathbf{v} = \sum_{i=1}^r (x_j + z_j) \bar{\mathbf{b}}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\},$$

and let

$$\mathcal{Q}' = \left\{ \mathbf{v} : \mathbf{v} = \sum_{i=1}^r x_j \mathbf{c}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\}.$$

Then we have

$$\mathcal{Q}' = \bigcup_{\substack{\mathbf{z} \in \mathbb{Z}^r \\ \|\mathbf{z}_j\| \leq \lceil (1/2)(\|\mathbf{c}\|_2/\|\bar{\mathbf{b}}_j\|_2 - 1) \rceil}} \mathcal{Q}(\mathbf{z}).$$

Fix $\mathbf{z} \in \mathbb{Z}^r$ and let S' be the set of all minima ν of \mathcal{O} such that $\text{LAPPROX}(\mathcal{O}, \nu, p+1) \in \mathcal{Q}(\mathbf{z})$. Then using (3.9) and Lemma 3.5.15 and estimations as in the proofs in section 6.2 we see that S' is contained in the set of units computed by the call of $\text{CUBOID}(\mathcal{O}, \mathcal{O}, \sum_{i=1}^r z_j \bar{\mathbf{b}}_j)$. Thus S is the set of all pairs $((1/\nu)\mathcal{O}, \nu)$ with ν being a minimum of \mathcal{O} such that $\text{LAPPROX}(\mathcal{O}, \nu, p+1) \in \left\{ \mathbf{w} : \mathbf{w} = \sum_{j=1}^r x_j \mathbf{c}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\}$.

If BABY computes an element $\varepsilon_{k+1} = \varepsilon \neq 0$ then by Lemma 3.4.4 we know that ε_{k+1} is a unit of \mathcal{O} , and it is also easy to see that $\text{LAPPROX}(\mathcal{O}, \varepsilon, p) \in \left\{ \mathbf{w} : \mathbf{w} = \sum_{i=1}^r x_i \mathbf{c}_i, x_i \in \mathbb{R}, |x_i| \leq 1/2 \text{ for } 1 \leq i \leq r \right\}$. Thus we only have to show that we have $\text{LAPPROX}(\mathcal{O}, \varepsilon, p) \in \mathcal{S}(\Lambda', B'_k, \sqrt{\epsilon}) \cap \mathcal{P}_{\delta/2+1/4}(\mathbf{b}_1^*, \dots, \mathbf{b}_k^*)$. This can be done by ideas similar to those used in the proof of Lemma 7.2.27. Hence, BABY is correct.

For estimating the running time we need an upper bound of the number of vectors \mathbf{z} enumerated in the algorithm. Clearly, that number is bounded by

$$\prod_{j=1}^k \left(2 \left\lceil \left(\frac{1}{2} \right) \left(\frac{\|\mathbf{c}_j\|_2}{\|\bar{\mathbf{b}}_j\|_2} - 1 \right) \right\rceil + 1 \right) \left(2 \left\lceil \left(\frac{1}{2} \right) \left(\frac{\|\mathbf{c}_r\|_2}{\|\bar{\mathbf{b}}_r\|_2} - 1 \right) \right\rceil + 1 \right)^{r-k}.$$

For each of the corresponding vectors $\mathbf{d} = \sum_{i=1}^r z_j \bar{\mathbf{b}}_j$ we have $\|\mathbf{d}\|_\infty \leq \sum_{i=1}^r \|\mathbf{c}_i\|_2$. Hence, by Lemma 6.2.21 the running time of each call up of CUBOID is

$$\left(n + \log(\Delta) + \log \left(\sum_{i=1}^r \|\mathbf{c}_i\|_2 \right) + \|\text{MT}(\Omega)\|_\infty \right)^{O(n)}.$$

Hence, applying Proposition 6.1.1 and some simple estimations we obtain that the running time of the whole algorithm is dominated by

$$\prod_{j=1}^k \left(2 \left[\binom{1}{2} \left(\frac{\|\mathbf{c}_j\|_2}{\|\bar{\mathbf{b}}_j\|_2} - 1 \right) \right] + 1 \right) \left(2 \left[\binom{1}{2} \left(\frac{\|\mathbf{c}_r\|_2}{\|\bar{\mathbf{b}}_r\|_2} - 1 \right) \right] + 1 \right)^{r-k} \left(n + p + \log(\Delta) + \log \left(\sum_{i=1}^r \|\mathbf{c}_i\|_2 \right) + \|\text{MT}(\Omega)\|_\infty \right)^{O(n)}.$$

□

Lemma 7.2.30 *The running time of GIANT (Algorithm 7.2.25) is*

$$\sqrt{R_{\mathcal{O}}} \left(n + p + \log(\Delta) + \|\text{MT}(\Omega)\|_\infty + \lceil \sqrt{2\delta'} \rceil + \left(\frac{1+\epsilon}{2\epsilon\sqrt{\epsilon}} \right)^{O(n)} \right).$$

Proof. As in the proof of Lemma 7.2.24 we see that \mathbf{b}_i^* is a q -approximation to $(\text{Log } \varepsilon_i)^*$ and B'_k is a q -approximation to B_k where $q = -\lceil \log(c) \rceil$ with $\log(c)$ as in the proof of Lemma 7.2.24. Hence by (7.21), Lemma 4.2.5 and Proposition 3.5.29 we have for $1 \leq i \leq k$

$$\|\mathbf{b}_i^*\|_2 \leq 2 \|(\text{Log } \varepsilon_i)^*\|_2, \quad (7.27)$$

and for all $\mathbf{b}' \in B'_k$

$$\frac{1}{2} \leq \|\mathbf{b}'\|_2 \leq 2. \quad (7.28)$$

We also have for $1 \leq j \leq r$

$$\frac{D_0}{2} \leq \|\bar{\mathbf{b}}_j\|_2 \leq 2D_0. \quad (7.29)$$

Now, we fix ℓ and use the notation of Algorithm 7.2.25. For $1 \leq j \leq k$ we have by (7.29)

$$\frac{\|\tilde{\mathbf{b}}_j + \bar{\mathbf{b}}_j\|_2}{\|\bar{\mathbf{b}}_j\|_2} \leq \frac{2\delta'}{2M_j + 1} \|\mathbf{b}_j^*\|_2 \frac{2}{D_0} + 1 \leq \frac{1}{D_0} \sqrt{2\delta' \|\mathbf{b}_j^*\|_2} + 1.$$

If $\sqrt{\|\mathbf{b}_j^*\|_2} \geq 1$ then we have

$$\lceil \sqrt{\|\mathbf{b}_j^*\|_2} \rceil \leq \sqrt{\|\mathbf{b}_j^*\|_2} + 1 \leq 2\sqrt{\|\mathbf{b}_j^*\|_2}.$$

But if $\sqrt{\|\mathbf{b}_j^*\|_2} \leq 1$ then we have by Proposition 3.5.29

$$\left\lceil \sqrt{\|\mathbf{b}_j^*\|_2} \right\rceil \leq \sqrt{\|\mathbf{b}_j^*\|_2} + 1 = \frac{1}{\sqrt{\|\mathbf{b}_j^*\|_2}} \left(\sqrt{\|\mathbf{b}_j^*\|_2} + \|\mathbf{b}_j^*\|_2 \right) \leq 4n\sqrt{\|\mathbf{b}_j^*\|_2}.$$

Hence, we always have

$$\left\lceil \sqrt{\|\mathbf{b}_j^*\|_2} \right\rceil \leq 4n\sqrt{\|\mathbf{b}_j^*\|_2}, \quad (7.30)$$

and therefore

$$\prod_{j=1}^k \left(2 \left\lceil \left(\frac{1}{2} \right) \left(\frac{\|\tilde{\mathbf{b}}_j + \bar{\mathbf{b}}_j\|_2}{\|\bar{\mathbf{b}}_j\|_2} - 1 \right) \right\rceil + 1 \right) \leq (8n)^k \left[\frac{1}{2D_0} \sqrt{2\delta'} \right]^k \prod_{j=1}^k \sqrt{\|\mathbf{b}_j^*\|_2}. \quad (7.31)$$

Next, we note that by (7.29) and (7.28) we have $k+1 \leq j \leq r$

$$\frac{\|\tilde{\mathbf{b}}_j + \bar{\mathbf{b}}_j\|_2}{\|\bar{\mathbf{b}}_j\|_2} \leq 2^{\ell+1} + 2, \quad (7.32)$$

and therefore

$$\left(2 \left\lceil \frac{1}{2} \left(\frac{\|\tilde{\mathbf{b}}_r + \bar{\mathbf{b}}_r\|_2}{\|\bar{\mathbf{b}}_r\|_2} - 1 \right) \right\rceil + 1 \right)^{r-k} \leq 2^{(r-k)(\ell+1)}. \quad (7.33)$$

Hence applying Lemma 7.2.29, (7.31), and (7.33) we see that the running time of the call $\text{BABY}(\mathcal{O}, k, \epsilon, \delta, \bar{\mathbf{b}}_1 + \bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_r + \bar{\mathbf{b}}_r, \bar{\mathbf{b}}_1, \dots, \bar{\mathbf{b}}_r)$ in step (22) is

$$(8n)^k \left[\frac{1}{2D_0} \sqrt{2\delta'} \right]^k \prod_{j=1}^k \sqrt{\|\mathbf{b}_j^*\|_2} 2^{(r-k)(\ell+1)}. \quad (7.34)$$

Thus from Lemma 7.2.27 (see (7.24)), (7.34), and Theorem 7.2.16 it follows that the running time of step (22) is

$$\sqrt{R_{\mathcal{O}}} \left(n + \log(\Delta) + \left\lceil \sqrt{2\delta'} \right\rceil + \left(\frac{1+\epsilon}{2\epsilon\sqrt{\epsilon}} \right)^{O(n)} (n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} \right), \quad (7.35)$$

where we have used that by (7.27), Theorem 7.2.16, and Theorem 3.5.7 we have

$$\begin{aligned} \prod_{j=1}^k \sqrt{\|\mathbf{b}_j^*\|_2} \left(\sqrt{\lambda_{k+1}} \right)^{r-k} &\leq 2^k \prod_{j=1}^k \sqrt{\|(\text{Log } \varepsilon_j)^*\|_2} \left(\sqrt{\lambda_{k+1}(\text{Log } \mathcal{O}^*)} \right)^{r-k} \\ &\leq 2^k \prod_{j=1}^r (\sqrt{r} \lambda_j(\text{Log } \mathcal{O}^*)) \leq 2^k n^{\frac{n}{2}} R_{\mathcal{O}}. \end{aligned}$$

Next, we have to estimate the number of calls of the procedure CLOSE , which equals the number of $\mathbf{z} \in \mathbb{Z}^r$ enumerated by the algorithm in step (27). Clearly,

$$\prod_{j=1}^k (2M_j + 1) 2^{(\ell+1)(r-k)}$$

is an upper bound of this number. By estimations analogous to the above ones we see that this number is also bounded by

$$\sqrt{R_{\mathcal{O}}} \left(n + \log(\Delta) + \lceil \sqrt{2\delta'} \rceil + \left(\frac{1+\epsilon}{2\epsilon\sqrt{\epsilon}} \right) \right)^{O(n)},$$

and that for each of this vectors \mathbf{z} we have $\log \|\sum_{j=1}^r \mathbf{z}_j \tilde{\mathbf{b}}_j\|_{\infty}$ is

$$(\log(\Delta) + n)^{O(n)}.$$

Hence, by Lemma 6.2.19, Lemma 6.3.6, and Proposition 6.1.1 the running time of the while-loop from step (25) to step (37) is

$$\left(p + n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty} + \lceil \sqrt{2\delta'} \rceil + \left(\frac{1+\epsilon}{2\epsilon\sqrt{\epsilon}} \right) \right)^{O(n)}. \quad (7.36)$$

(Here we used that by Proposition 5.1.8 the norm of a minimum of \mathcal{O} is bounded by $O(\Delta_{\mathcal{O}})$.)

From (7.35), (7.36), and Proposition 6.1.1 it follows that each round of the while-loop from step (15) to (39) takes time

$$\sqrt{R_{\mathcal{O}}} \left(n + p + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty} + \lceil \sqrt{2\delta'} \rceil + \left(\frac{1+\epsilon}{2\epsilon\sqrt{\epsilon}} \right) \right)^{O(n)}$$

By Corollary 3.5.31 and (7.24) we obtain that the number of rounds is at most

$$\ell \leq 3 + \log \left(\left(\frac{1+\epsilon}{2\epsilon\sqrt{\epsilon}} \right) \left((2r)^{r+1} 2^{(n+1)r} 4n^2 \sqrt{|\Delta_{\mathcal{O}}|} (\log |\Delta_{\mathcal{O}}|)^{n-1} (\log \log |\Delta_{\mathcal{O}}|)^{n/2} \right) \right).$$

This concludes the proof, since the above running times dominates the running times of the remaining steps of the algorithm. \square

Finally we can estimate the running time of FUNDAMENTAL (Algorithm 7.2.14).

Theorem 7.2.31 *Given an order \mathcal{O} , and $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1$, $1/2 < \delta < 1$ the algorithm FUNDAMENTAL determines a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that $(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r)$ is a (ϵ, δ) -constructable basis of $\text{Log } \mathcal{O}^*$ in time*

$$\sqrt{R_{\mathcal{O}}} \left(-\log(\delta - 1/2) - \log(1 - \epsilon) + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty} + \left(\frac{1+\epsilon}{2\sqrt{\epsilon}} \right) \right)^{O(n)}.$$

Proof. By Lemma 7.2.30 the running time of each call of GIANT is

$$\sqrt{R_{\mathcal{O}}} \left(n + p + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty} + \left(\frac{1+\epsilon}{2\epsilon\sqrt{\epsilon}} \right) \right)^{O(n)}. \quad (7.37)$$

Gram-Schmidt vectors of a given sequence of vectors can be computed in polynomial time. Thus by Theorem 7.2.16 and by Proposition 6.1.1 the running time of each call of NEXTVECTOR is dominated by (7.37). The assertion follows from Proposition 3.2.6, the choice of p and the fact that $\delta < 1$. \square

From Corollary 6.3.4 and Proposition 3.2.6 we immediately obtain

Corollary 7.2.32 *There is an algorithm that on input of an order \mathcal{O} and a number $q \in \mathbb{N}$ computes a q -approximation to $R_{\mathcal{O}}$ in time*

$$\sqrt{R_{\mathcal{O}}} (q + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} .$$

If we assume that the orders are given by short multiplication tables (see section 3.2) then we obtain more incisive formulas:

Corollary 7.2.33 *Given an order \mathcal{O} by a short multiplication table, and $\epsilon, \delta \in \mathbb{Q}$ with $1/2 < \epsilon < 1, 1/2 < \delta < 1$ the algorithm *FUNDAMENTAL* determines (in compact representation) a system of fundamental units $\{\varepsilon_1, \dots, \varepsilon_r\}$ of \mathcal{O} such that $(\text{Log } \varepsilon_1, \dots, \text{Log } \varepsilon_r)$ is a (ϵ, δ) -constructable basis of $\text{Log } \mathcal{O}^*$ in time*

$$\sqrt{R_{\mathcal{O}}} \left(-\log(\delta - 1/2) - \log(1 - \epsilon) + \log(\Delta) + \left(\frac{1 + \epsilon}{2\sqrt{\epsilon}} \right) \right)^{O(n)} .$$

Corollary 7.2.34 *There is an algorithm that on input of a short multiplication table of an order \mathcal{O} computes (in compact representation) a system of fundamental units in time*

$$\sqrt{R_{\mathcal{O}}} (\log(\Delta))^{O(n)} .$$

Corollary 7.2.35 *There is an algorithm that on input of a short multiplication table of an order \mathcal{O} and a number $q \in \mathbb{N}$ computes a q -approximation to $R_{\mathcal{O}}$ in time*

$$\sqrt{R_{\mathcal{O}}} (q + \log(\Delta))^{O(n)} .$$

7.3 The Containment Problem

In this section we want to show how to solve the *containment problem*: Given a set T of reduced invertible ideals and a reduced invertible ideal \mathfrak{B} we wish to find out whether \mathfrak{B} is equivalent to some $\mathfrak{C} \in T$.

Using a strategy similar to Algorithm 7.2.25 we first show how to implement the algorithm *EQUIV* that decides whether two given reduced ideals are equivalent. We follow the ideas of [6]. Note that here we work with approximations where in [6] the author uses real numbers in his algorithms.

Algorithm 7.3.1 (EQUIV)

Input : an order \mathcal{O} ; two reduced invertible ideals $\mathfrak{B}, \mathfrak{C}$ of \mathcal{O} ;
Output : if $\mathfrak{B} \sim \mathfrak{C}$ then the compact representation of a generator α of \mathfrak{B} relative to \mathfrak{C} , else $\alpha = 0$

```

(1) procedure EQUIV ( $\mathcal{O}, \mathfrak{B}, \mathfrak{C}$ )
(2)    $\alpha := 0$ ;
(3)    $(\varepsilon_1, \dots, \varepsilon_r) := \text{FUNDAMENTAL}(\mathcal{O}, 3/4, 3/4)$ ;
(4)   Compute  $p \in \mathbb{N}$  with  $p = \lceil 4(5n + 9)(16n + 5 \log(\Delta) + 2(2n + \log(\Delta) + 1)n) + 54 \rceil$ ;
(5)   for ( $i := 1$  to  $r$  step 1) do
(6)      $\mathbf{b}_i := \text{LAPPROX}(\mathcal{O}, \varepsilon_i, p)$ 
(7)   od
(8)   Compute the Gram-Schmidt vectors  $\mathbf{b}_1^*, \dots, \mathbf{b}_r^*$ ;
(9)    $\delta' := 5/8$ ;
(10)   $D_0 := \lceil 2\sqrt{r}(\log(\Delta) + 3)/4 + 2^{-p} \rceil$ ;
(11)  for ( $j := 1$  to  $r$  step 1) do
(12)     $M_j := \lceil (2\delta' \|\mathbf{b}_j\|_2)^{1/2} \rceil$ ;
(13)    Compute  $U_j \in \mathbb{Q}$  such that  $U_j \mathbf{b}_j^*$  is a  $p$ -approximation to  $(\|\mathbf{b}_j\|_2 2\delta' / (2M_j + 1))(\mathbf{b}_j^* / \|\mathbf{b}_j^*\|_2)$ ;
(14)     $\tilde{\mathbf{b}}_j := U_j \mathbf{b}_j^*$ ;
(15)    Compute a vector  $\bar{\mathbf{b}}_j$  of the form  $u_j \mathbf{b}_j^* / \|\mathbf{b}_j^*\|_2$ , where  $u_j$  is a  $p$ -approximation to  $D_0$ ;
(16)     $\mathbf{b}_j := U_j \mathbf{b}_j^*$ ;
(17)  od
(18)  Compute the set  $S$  of all pairs  $((1/\nu)\mathfrak{C}, \nu)$  with  $\nu \in \mathfrak{C}$  being a minimum of  $\mathfrak{C}$  such that  $\text{LAPPROX}(\mathcal{O}, \nu, p + 1) \in \{\mathbf{w} : \mathbf{w} = \sum_{j=1}^r x_j (\tilde{\mathbf{b}}_r + \bar{\mathbf{b}}_r), x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r\}$ ;
(19)  for (all  $\mathbf{z} \in \mathbb{Z}^r$  with  $|\mathbf{z}_j| \leq M_j$  for all  $1 \leq j \leq r$ ) do
(20)     $\mu := \text{CLOSE}(\mathcal{O}, \mathfrak{B}, \sum_{j=1}^r \mathbf{z}_j \tilde{\mathbf{b}}_j)$ ;
(21)    if (there exists  $(\mathfrak{A}, \nu) \in S$  with  $(1/\mu)\mathfrak{B} = \mathfrak{A}$ ) then
(22)       $\alpha := \text{COMPACT}(\mathcal{O}, \mu/\nu, \mathfrak{B})$ ;
(23)    exit ;
(24)  fi
(25)  od
(26) end procedure

```

Lemma 7.3.2 *EQUIV (Algorithm 7.3.1) is correct. On input of an order \mathcal{O} and two reduced invertible ideals $\mathfrak{B}, \mathfrak{C}$ of \mathcal{O} it determines an algebraic number $\alpha \in \mathbb{F}$, where α is a generator of \mathfrak{B} relative to \mathfrak{C} if $\mathfrak{B} \sim \mathfrak{C}$, and where $\alpha = 0$ otherwise. The running time of EQUIV is*

$$\sqrt{R_{\mathcal{O}}} (n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} .$$

Proof. Suppose that EQUIV determines an element $\alpha \neq 0$. Then there exist a minimum ν of \mathfrak{C} and a minimum μ of \mathfrak{B} such that

$$\frac{1}{\nu}\mathfrak{C} = \frac{1}{\mu}\mathfrak{B}.$$

Thus we have $\mathfrak{B} \sim \mathfrak{C}$. It is also easy to see that α is a generator of \mathfrak{B} relative to \mathfrak{C} .

On the other hand suppose that $\mathfrak{B} \sim \mathfrak{C}$. We use an idea analogous to Lemma 7.2.26. Let

$$\tilde{\mathcal{P}} = \left\{ \mathbf{u} : \mathbf{u} = \sum_{j=1}^r x_j \tilde{\mathbf{b}}_j, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\},$$

for $\mathbf{z} \in \mathbb{R}^r$ let

$$\tilde{\mathcal{P}}(\mathbf{z}) = \left\{ \mathbf{v} : \mathbf{v} = \mathbf{u} + \sum_{j=1}^r \mathbf{z}_j \tilde{\mathbf{b}}_j, \mathbf{u} \in \tilde{\mathcal{P}} \right\}.$$

Also, let

$$\mathcal{P}' = \left\{ \mathbf{u} : \mathbf{u} = \sum_{j=1}^r x_j (\tilde{\mathbf{b}}_j + \bar{\mathbf{b}}_j), x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\}.$$

If $\mathfrak{B} \sim \mathfrak{C}$ then there exists a minimum β of \mathfrak{C} with $\mathfrak{B} = (1/\beta)\mathfrak{C}$. Clearly, we may assume w.l.o.g. that

$$\text{Log } \beta \in \left\{ \mathbf{v} : \mathbf{v} = \sum_{j=1}^r x_j \text{Log } \varepsilon, x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r \right\}$$

(cf. the proof of Corollary 6.3.10). By Lemma 3.5.15, (3.9), Lemma 7.2.19 and the choice of p we have for every p -approximation \mathbf{b} to $\text{Log } \beta$

$$\mathbf{b} \in \left\{ \mathbf{v} : \mathbf{v} = \sum_{j=1}^r x_j \mathbf{b}_j, x_j \in \mathbb{R}, |x_j| \leq \delta' \text{ for } 1 \leq j \leq r \right\} \subseteq \bigcup_{\substack{\mathbf{z} \in \mathbb{Z}^r \\ |\mathbf{z}_j| \leq M_j, 1 \leq j \leq r}} \tilde{\mathcal{P}}(\mathbf{z}).$$

Now, let $\mathbf{z} \in \mathbb{Z}^r$ with $|\mathbf{z}_j| \leq M_j$ for $1 \leq j \leq r$ such that $\mathbf{b} \in \tilde{\mathcal{P}}(\mathbf{z})$. If μ is a minimum of \mathfrak{B} that is $(3/4)$ -close to the vector

$$-\left(\sum_{j=1}^r \mathbf{z}_j \tilde{\mathbf{b}}_j \right)$$

then by Proposition 5.1.3 the element $\nu = \beta\mu$ is a minimum of \mathfrak{C} . Using Lemma 3.5.15, (3.9) and by the choice of p we see that $\text{LAPPROX}(\mathcal{O}, \nu, p+1) \in \mathcal{P}'$. Hence S shall contain the element ν and thus the algorithm EQUIV shall output a generator of \mathfrak{B} relative to \mathfrak{C} .

Step (18) of the algorithm can be implemented by using a simple variant of the algorithm BABY. Hence the proof of the running is analogous to the proof of the running

time of FUNDAMENTAL (Algorithm 7.2.14). Again, the needed approximations can be computed in the stated running time, which can be shown by our usual arguments using Theorem 7.2.16, Proposition 6.1.1 and the results of chapter 4. \square

By Lemma 7.3.2 and Proposition 6.1.1 we can give a bound for the time that is needed to solve discrete logarithm problems of an order \mathcal{O} . Also using Proposition 3.2.6 we obtain

Corollary 7.3.3 *There is an algorithm that given an order \mathcal{O} , a reduced invertible ideal \mathfrak{A} of \mathcal{O} and $q \in \mathbb{N}$ decides whether \mathfrak{A} is principal and in that case computes a generator α of \mathfrak{A} and a q -approximation to $\text{Log } \alpha$ in time*

$$\sqrt{R_{\mathcal{O}}} (q + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} .$$

Again, assuming that orders are given by short multiplication tables we obtain

Corollary 7.3.4 *There is an algorithm that given an order \mathcal{O} by a short multiplication table, a reduced invertible ideal \mathfrak{A} of \mathcal{O} and $q \in \mathbb{N}$ decides whether \mathfrak{A} is principal and in that case computes a generator α of \mathfrak{A} and a q -approximation to $\text{Log } \alpha$ in time*

$$\sqrt{R_{\mathcal{O}}} (q + \log(\Delta))^{O(n)} .$$

Finally we can show how to solve the *containment problem*: Given a set T of reduced invertible ideals and a reduced invertible ideal \mathfrak{B} we wish to find out whether \mathfrak{B} is equivalent to some $\mathfrak{C} \in T$. More precisely, we want to solve the problem for a fixed set T but many different reduced ideals \mathfrak{C} . We generalize the ideas of [3] and proceed in two steps. In a precomputation we compute for every $\mathfrak{C} \in T$ a set S as in step (18) of Algorithm 7.3.1 and determine the union of these sets. We use the notation of [3] and call that union the *expansion* of T . Then for each ideal \mathfrak{C} we only have to perform step (19)–(25).

Algorithm 7.3.5 (PRECOMPUTATION)

Input : an order \mathcal{O} ; a set T of reduced invertible ideals of \mathcal{O} ;

Output : the expansion S of T , $M_1, \dots, M_r, \tilde{\mathbf{b}}_1, \dots, \tilde{\mathbf{b}}_r$

- (1) **procedure** PRECOMPUTATION ($\mathcal{O}, \mathfrak{B}, \mathfrak{C}$)
- (2) $\alpha := 0; S := \emptyset;$
- (3) $(\varepsilon_1, \dots, \varepsilon_r) := \text{FUNDAMENTAL}(\mathcal{O}, 3/4, 3/4);$
- (4) Compute $p \in \mathbb{N}$ with $p = \lceil 4(5n + 9)(16n + 5 \log(\Delta) + 2(2n + \log(\Delta) + 1)n) + 54 \rceil;$
- (5) **for** ($i := 1$ **to** r **step 1**) **do**
- (6) $\mathbf{b}_i := \text{LAPPROX}(\mathcal{O}, \varepsilon_i, p)$
- (7) **od**
- (8) Compute the Gram-Schmidt vectors $\mathbf{b}_1^*, \dots, \mathbf{b}_r^*;$
- (9) $\delta' := 5/8;$

- (10) $D_0 := \lceil 2\sqrt{r}(\log(\Delta) + 3)/4 + 2^{-p} \rceil$;
- (11) **for** ($j := 1$ **to** r **step** 1) **do**
- (12) $M_j := \lceil (2\delta' \|\mathbf{b}_j\|_2)^{1/2} \rceil$;
- (13) Compute $U_j \in \mathbb{Q}$ such that $U_j \mathbf{b}_j^*$ is a p -approximation to $(\|\mathbf{b}_j\|_2 2\delta' / (2M_j + 1))(\mathbf{b}_j^* / \|\mathbf{b}_j^*\|_2)$;
- (14) $\tilde{\mathbf{b}}_j := U_j \mathbf{b}_j^*$;
- (15) Compute a vector $\bar{\mathbf{b}}_j$ of the form $u_j \mathbf{b}_j^* / \|\mathbf{b}_j^*\|_2$, where u_j is a p -approximation to D_0 ;
- (16) $\mathbf{b}_j := U_j \mathbf{b}_j^*$;
- (17) **od**
- (18) **for** (every $\mathfrak{C} \in T$) **do**
- (19) Compute the set S' of all pairs $((1/\nu)\mathfrak{C}, \nu)$ with $\nu \in \mathfrak{C}$ being a minimum of \mathfrak{C} such that $\text{LAPPROX}(\mathcal{O}, \nu, p+1) \in \{\mathbf{w} : \mathbf{w} = \sum_{j=1}^r x_j(\tilde{\mathbf{b}}_j + \bar{\mathbf{b}}_j), x_j \in \mathbb{R}, |x_j| \leq 1/2 \text{ for } 1 \leq j \leq r\}$;
- (20) $S := S \cup S'$;
- (21) **od**
- (22) **end procedure**

From Lemma 7.3.2 and the above remarks we immediately obtain

Proposition 7.3.6 *The running time of PRECOMPUTATION (Algorithm 7.3.5) is*

$$|T| \sqrt{R_{\mathcal{O}}} (n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} .$$

The final test for containment in the set T has the following form:

Algorithm 7.3.7 (CONTAINMENT)

- Input :** an order \mathcal{O} ; a reduced invertible ideal \mathfrak{B} ; the expansion S of a set T of reduced invertible ideals of \mathcal{O} ; $p, M_1, \dots, M_r, \tilde{\mathbf{b}}_1, \dots, \tilde{\mathbf{b}}_r$ as in Algorithm 7.3.5
- Output :** if \mathfrak{B} is equivalent to some $\mathfrak{C} \in T$ then a generator α of \mathfrak{B} relative to \mathfrak{C} , else $\alpha = 0$

- (1) **procedure** CONTAINMENT ($\mathcal{O}, \mathfrak{B}, S, M_1, \dots, M_r, \tilde{\mathbf{b}}_1, \dots, \tilde{\mathbf{b}}_r$)
- (2) $\alpha := 0$;
- (3) **for** (all $\mathbf{z} \in \mathbb{Z}^r$ with $|\mathbf{z}_j| \leq M_j$ for all $1 \leq j \leq r$) **do**
- (4) $\mu := \text{CLOSE}(\mathcal{O}, \mathfrak{B}, \sum_{j=1}^r \mathbf{z}_j \tilde{\mathbf{b}}_j)$;
- (5) **if** (there exists $(\mathfrak{A}, \nu) \in S$ with $(1/\mu)\mathfrak{B} = \mathfrak{A}$) **then**
- (6) $\alpha := \text{COMPACT}(\mathcal{O}, \mu/\nu, \mathfrak{B})$;
- (7) **exit** ;
- (8) **fi**
- (9) **od**
- (10) **end procedure**

Proposition 7.3.8 *CONTAINMENT (Algorithm 7.3.7) is correct. The running time of each call of CONTAINMENT after PRECOMPUTATION is*

$$\sqrt{R_{\mathcal{O}}}(n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} .$$

Proof. The correctness of the algorithm can be shown in the same way as the correctness of Algorithm 7.2.25 or Algorithm 7.3.1.

By Lemma 5.1.17 the set S contains at most $6^n h_{\mathcal{O}} R_{\mathcal{O}}$ reduced ideals. Hence we can search an element in S in time $O(\log(6^n h_{\mathcal{O}} R_{\mathcal{O}}))$. Hence from Theorem 3.4.9 and the proof of Lemma 7.3.2 it follows that the running time of each call of CONTAINMENT is

$$\sqrt{R_{\mathcal{O}}}(n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} . \quad \square$$

Using the above results we now complete the analysis of the algorithms BOUNDED and DISCRETE.

Lemma 7.3.9 *Given an order \mathcal{O} , invertible ideals \mathfrak{A} and \mathfrak{D} of \mathcal{O} , and $u \in \mathbb{N}_{\geq 2}$ the algorithm BOUNDED (Algorithm 7.1.1) determines the bounded discrete logarithm in time*

$$\sqrt{u R_{\mathcal{O}}}(n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} (\text{size}(\mathfrak{A}) + \text{size}(\mathfrak{D}))^{O(1)} .$$

Proof. We use the notation of Algorithm 7.1.1. As in [3] we see that BOUNDED requires $O(\sqrt{u})$ operations and tests for containment in the set T which contains $O(\sqrt{u})$ elements. This tests can be performed using one call of PRECOMPUTATION and \sqrt{u} calls of CONTAINMENT. Hence by Proposition 7.3.6 and Proposition 7.3.8 these tests need time

$$\sqrt{u R_{\mathcal{O}}}(n + \log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} .$$

The algorithm also contains $O(\sqrt{u})$ tests whether two reduced invertible ideals are equivalent, and $O(\sqrt{u})$ calls of REDUCE where the input of REDUCE is either \mathfrak{A} , \mathfrak{D} or the product of two reduced ideals. Thus, the assertion follows from Lemma 7.3.2, Algorithm 5.2.3 and Corollary 5.1.12. \square

Theorem 7.3.10 *On input of an order \mathcal{O} and invertible ideals \mathfrak{A} and \mathfrak{D} of \mathcal{O} the running time of DISCRETE (Algorithm 7.1.3) is*

$$\Delta^{\frac{1}{4}} (\log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} (\text{size}(\mathfrak{A}) + \text{size}(\mathfrak{D}))^{O(1)} .$$

Proof. We use the notation of Algorithm 7.1.3. Clearly, DISCRETE uses the subroutine BOUNDED at most $O(\log(h_{\mathcal{O}}))$ times. Since u is bounded by $2h_{\mathcal{O}}$ we have by Lemma 7.3.9 and Proposition 3.2.6 that the running time of the algorithm is

$$\sqrt{h_{\mathcal{O}} R_{\mathcal{O}}} (\log(\Delta) + \|\text{MT}(\Omega)\|_{\infty})^{O(n)} (\text{size}(\mathfrak{A}) + \text{size}(\mathfrak{D}))^{O(1)} .$$

Thus the assertion follows from Theorem 3.4.9. \square

Corollary 7.3.11 *On input of a short multiplication table of an order \mathcal{O} and invertible ideals \mathfrak{A} and \mathfrak{D} of \mathcal{O} the running time of DISCRETE (Algorithm 7.1.3) is*

$$\Delta^{\frac{1}{4}} (\log(\Delta))^{O(n)} (\text{size}(\mathfrak{A}) + \text{size}(\mathfrak{D}))^{O(1)} .$$

By Corollary 5.1.12 we thus obtain

Corollary 7.3.12 *On input of a short multiplication table of an order \mathcal{O} and two reduced invertible ideals of \mathcal{O} the algorithm DISCRETE (Algorithm 7.1.3) solves the discrete logarithm problem in the class group of \mathcal{O} in time*

$$\Delta^{\frac{1}{4}} (\log(\Delta))^{O(n)} .$$

Bibliography

- [1] S. Amendola, *Effiziente Algorithmen für Probleme in Gittern über \mathbb{R}^d* , Master's thesis, Universität des Saarlandes, 1995.
- [2] J. L. Balcázar, J. Diaz, and J. Gabarró, *Structural Complexity I*, Springer-Verlag, 1988.
- [3] I. Biehl and J. Buchmann, *Algorithms for quadratic orders*, Proceedings of Symposia in Applied Mathematics (1993), 329–347.
- [4] J. Buchmann, *On the computation of units and class numbers by a generalization of lagrange's algorithm*, J. Number Theory **26** (1987), no. 1, 8–30.
- [5] ———, *On the period length of the generalized lagrange algorithm*, J. Number Theory **26** (1987), no. 1, 31–37.
- [6] ———, *Zur Komplexität der Berechnung von Einheiten und Klassenzahlen algebraischer Zahlkörper*, Habilitationsschrift, 1987.
- [7] ———, *A subexponential algorithm for the determination of class group and regulator of algebraic number fields*, Séminaire de Théorie des Nombres, Paris 1988–1989, Birkhäuser Verlag, 1990.
- [8] ———, *Number theoretic algorithms and cryptology*, Proc. of FCT'91 (LNCS 529), Springer-Verlag, 1991, pp. 16–21.
- [9] ———, *Reducing lattice bases by means of approximations*, Proc. First ANTS (1994) (L. M. Adleman and M. Huang, eds.), Springer-Verlag, 1994.
- [10] J. Buchmann and H. W. Lenstra Jr., *Computing maximal orders and decomposing primes in number fields*, Preprint.
- [11] ———, *Approximating rings of integers in number fields*, J. de Théorie des Nombres **6** (1994), 221–260.
- [12] J. Buchmann and O. van Sprang, *On short representations of orders and number fields*, Preprint, 1992.
- [13] J. Buchmann and H. C. Williams, *On principal ideal testing in algebraic number fields*, J. Symbolic Computation **4** (1987), no. 1, 11–19.

- [14] ———, *A key exchange system based on real-quadratic fields*, Proc. of CRYPTO'89 (LNCS 435), Springer-Verlag, 1989.
- [15] ———, *On the computation of the class number of an algebraic number field*, Math. Comp. **53** (1989), 679–688.
- [16] ———, *On the existence of a short proof for the value of the class number and regulator of a real quadratic field*, NATO Advanced Science Institutes Series C, vol. 256, 327–345, NATO Advanced Science Institutes Series C, Kluwer, Dordrecht, 1989, pp. 327–345.
- [17] ———, *Some remarks concerning the complexity of computing class groups of quadratic fields*, J. Complexity **7** (1991), 311–315.
- [18] J. Buchmann, H. C. Williams, and C. Thiel, *Short representation of quadratic integers*, Proc. CANT (1992), Springer-Verlag, 1994.
- [19] J. W. S. Cassels, *An Introduction to the Geometry of Numbers*, Springer-Verlag, Berlin et al., 1959.
- [20] H. Cohen, *A Course in Computational Algebraic Number Theory*, Springer-Verlag, Berlin et al., 1993.
- [21] W. Diffie and M. Hellman, *New directions in cryptography*, IEEE Trans. Inform. Theory **22** (1976), 472–492.
- [22] T. ElGamal, *A public key cryptosystem and a signature scheme based on discrete logarithms*, IEEE Trans. Inform. Theory **31** (1985), 469–472.
- [23] U. Fincke and M. Pohst, *Improved methods for calculating vectors of short length in a lattice, including a complexity analysis*, MATHC **4** (1985), 463–471.
- [24] G. Fischer and R. Sacher, *Einführung in die Algebra*, Teubner Studienbücher Mathematik, 1981.
- [25] G. Ge, *Algorithms related to multiplicative representations of algebraic numbers*, Ph.D. thesis, University of California at Berkeley, 1993.
- [26] P. E. Gill, W. Murray, and M. H. Wright, *Numerical Linear Algebra and Optimization (Volume 1)*, Addison-Wesley Publishing Company, 1991.
- [27] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, 1983.
- [28] D. Gordon, *Discrete logarithms in $GF(p)$ using the number field sieve*, Siam Jour. on Discrete Math. **26** (1993), no. 6, 124–138.
- [29] W. H. Greub, *Linear Algebra*, Springer-Verlag, Berlin et al., 1967.

- [30] B. Helfrich, *Algorithms to construct minkowski reduced and hermite reduced lattice bases*, J. Theoretical Computer Science **41** (1985), 125–139.
- [31] H. W. Lenstra Jr., *Integer programming with a fixed number of variables*, Math. Oper. Res. **8** (1983), 538–548.
- [32] ———, *Algorithms in algebraic number theory*, Bull. Amer. Math. Soc. (N.S.) **26** (1992), no. 4, 211–244.
- [33] R. Kannan, *Minkowski's Convex Body Theorem And Integer Programming*, Math. Operations Research **12** (1987), no. 3, 415–439.
- [34] V. Kessler, *On the minimum of the unit lattice*, Preprint, 1991.
- [35] D. E. Knuth, *The art of computer programming, vol. 3: Sorting and searching*, Addison-Wesley, 1973.
- [36] ———, *The art of computer programming, vol. 2: seminumerical algorithms*, Addison-Wesley, 1981.
- [37] J. C. Lagarias, H. W. Lenstra Jr., and C. P. Schnorr, *Korkine-Zolotarev bases and successive minima of a lattice and its dual lattice*, Combinatorica **10** (1990), no. 4, 333–348.
- [38] H. R. Lewis and C. H. Papadimitriou, *Elements of the theory of computation*, Prentice-Hall, 1981.
- [39] K. S. McCurley, *Cryptographic key distribution and computation in class groups*, NATO Advanced Science Institutes Series C, vol. 256, 459–479, NATO Advanced Science Institutes Series C, Kluwer, Dordrecht, 1989, pp. 459–479.
- [40] A. Müller, *Effiziente Algorithmen für Probleme der linearen Algebra über \mathbb{Z}* , Master's thesis, Universität des Saarlandes, 1994.
- [41] W. Narkiewicz, *Number theory*, World Scientific Publishing Co, 1983.
- [42] ———, *Elementary and analytic theory of algebraic numbers*, Springer-Verlag, Berlin et al., 1990.
- [43] S. Neis, *Kurze Darstellung von Ordnungen*, Master's thesis, Universität des Saarlandes, 1994.
- [44] National Institute of Standards and Technology, *The digital signature standard, proposal and discussion*, Comm. of the ACM **35** (1992), 36–54.
- [45] M. Pohst and H. Zassenhaus, *Algorithmic algebraic number theory*, Cambridge University Press, 1989.
- [46] M. E. Pohst, *Computational Algebraic Number Theory*, Birkhäuser Verlag, Basel et al., 1993.

- [47] J. W. Sands, *Generalization of a theorem of Siegel*, Acta Arithmetica **58** (1991), 47–57.
- [48] R. Scheidler, J. Buchmann, and H.C. Williams, *Implementation of a key exchange protocol using real quadratic fields*, Proc. of EUROCRYPT'90, Springer-Verlag, 1990.
- [49] A. Schinzel and H. Zassenhaus, *A refinement of two theorems of Kronecker*, Michigan Math. J. **12** (1965), 81–85.
- [50] A. Schönhage, *Factorization of univariate integer polynomials by diophantine approximation and an improved basis reduction algorithm*, ICALP (1984).
- [51] ———, *Numerik analytischer Funktionen und Komplexität*, Jahresber. Deutsch. Math.-Verein. **92** (1990).
- [52] A. Schrijver, *Theory of linear and integer programming*, Wiley&Sons Ltd., Chichester, 1987.
- [53] D. Shanks, *Class number, a theory of factorization and genera*, Proc. Sympos. Pure Math. **20** (1970), 415–440.
- [54] ———, *The infrastructure of a real quadratic field and its applications*, Proc. 1972 Number Theory Conference (1972), 217–224.
- [55] G. W. Stewart, *Introduction to Matrix Computation*, Academic Press, 1973.
- [56] ———, *Perturbation bounds for the QR factorization of a matrix*, SIAM J. Numer. Anal. **14** (1977), no. 3, 509–518.
- [57] I. Stewart and D. Tall, *Algebraic Number Theory*, University Press, Cambridge, 1987.
- [58] J. Stoer, *Einführung in die numerische Mathematik*, Springer-Verlag, Berlin et al., 1978.
- [59] C. Thiel, *Short proofs using compact representations of algebraic integers*, to appear in J. Complexity, 1994.
- [60] ———, *Under the Assumption of the Generalized Riemann Hypothesis Verifying the Class Number Belongs to $NP \cap co-NP$* , Proc. First ANTS (1994) (L. M. Adleman and M. Huang, eds.), Springer-Verlag, 1994.
- [61] P. van Emde Boas, *Machine models, computational complexity and number theory*, Computational Methods in Number Theory (H. W. Lenstra Jr. and R. Tijdeman, eds.), Mathematisch Centrum Amsterdam, Amsterdam, 1982, pp. 111–167.
- [62] D. Weber, *Ein Algorithmus zur Zerlegung von Primzahlen in Primideale*, Master's thesis, Universität des Saarlandes, 1993.

- [63] R. Zimmert, *Ideale kleiner Norm in Idealklassen und eine Regulatorabschätzung*, Invent. Math. **62** (1981), 367–380.