Diplomarbeit

Zur Komplexität des „Shortest Vector Problem“ und seine Anwendungen in der Kryptographie

FRANK VALLENTIN

im August 1999

vorgelegt bei

Prof. Dr. INGO WEGENER

Lehrstuhl für Komplexitätstheorie und Effiziente Algorithmen
Fachbereich Informatik
Universität Dortmund

Vorwort

An dieser Stelle danke ich allen, die an der Entstehung meiner Diplomarbeit beteiligt waren.

Ich danke Prof. Dr. INGO WEGENER für die vertrauensvolle Zusammenarbeit und die gezielten Hilfestellungen. Ich danke den Mitgliedern des Instituts für Algebra und Geometrie des Fachbereichs Mathematik an der Universität Dortmund für die Schaffung einer lehrreichen und motivierenden Atmosphäre. Insbesondere danke ich Prof. Dr. RUDOLF SCHARLAU für die dauerhafte Unterstützung und Förderung. BORIS HEMKEMEIER danke ich für hilfreiche Gespräche und Ratschläge. Bei Dr. TOMAS SANDER bedanke ich mich für den Hinweis, der mich zu dem Thema meiner Diplomarbeit führte. Schließlich danke ich ANDRÉ, ANJA, ANNE-KATRIN, BERNHARD, DIRK, LUDGER, MARK, MARKUS und RALF für die zahlreichen Anmerkungen und Korrekturen.

Für die übrigen Fehler und andere Versehen bitte ich die Leserin und den Leser um Nachsicht und freue mich über konstruktive Kritik.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Die zentrale Frage dieser Diplomarbeit lautet:

Ist Kryptographie eine Geheimwissenschaft?

Oder genauer und bescheidener: Wie kann Kryptographie theoretisch fundiert werden?

Was ist Kryptographie? Die „Kryptographie“ ist ein Teilgebiet der Wissenschaft der „Kryptologie“, die sich mit Geheimschriften beschäftigt. Das besondere Merkmal der Kryptographie ist, daß chiffrierte Nachrichten öffentlich zugänglich sind. In der Kryptographie werden Verfahren entwickelt und untersucht, die es Unbefugten unmöglich machen sollen, offene, aber chiffrierte Geheimschriften zu entziffern. Ein zentraler Aspekt der Kryptographie ist die Bewertung der Sicherheit von eingesetzten Verfahren. Ein kryptographisches Verfahren gilt als sicher, wenn unbefugte Entzifferer die durch dieses Verfahren chiffrierten Nachrichten mit vertretbarem Aufwand nicht dechiffrieren können.

Militärisch geprägte Kryptographie Bis vor wenigen Jahren wurde Kryptographie fast ausschließlich für militärische Zwecke genutzt. In diesem Zeitraum war die Kryptographie eine Geheimwissenschaft: Kryptographische Forschung wurde nur von und für militärische Einrichtungen betrieben, und die Forschenden durften ihre Ergebnisse der Öffentlichkeit nicht präsentieren. Eine Ausnahme der militärischen Geheimniskrämerei bildet die Begründung der informationstheoretischen Kryptographie von CLAUDE E. SHANNON ([Sha49]), die als Ergebnis der Forschung im zweiten Weltkrieg anzusehen ist.
In SHANNONs Modell ist die Rechenkraft eines unbefugten Entzifferers nicht beschränkt. Er definiert unter dieser rigorosen Voraussetzung *perfekte Sicherheit* als die Unmöglichkeit, nützliche Informationen aus einer chiffrierten Nachricht zu berechnen, ohne den Schlüssel zu kennen. SHANNON erkannte, daß perfekte Sicherheit nur dann möglich ist, wenn die Anzahl der Bits, die Sender und Empfänger über einen öffentlichen Kommunikationskanal austauschen, höchstens so groß ist wie die Anzahl der Bits, die sie vorher über einen geheimen Kommunikationskanal vereinbart haben.

Kryptographie für die Massen Für sicherheitsrelevante Anwendungen in großen Rechnernetzen, wie z.B. dem Electronic Commerce im Internet, sind kryptographische Methoden mit perfekter Sicherheit nicht einsetzbar. Das Schlüsselmanagement ist viel zu aufwendig. Bevor zwei Teilnehmer abhörsicher kommunizieren können, müssen sie einen Schlüssel ausgetauscht haben, dessen Länge die Länge der zu sendenden Nachricht nicht unterschreitet. Für ein Rechnernetz mit n Benutzern werden $\binom{n}{2}$ geheime Schlüssel benötigt. Die Struktur des Internet macht es notwendig, daß sich die Benutzer authentifizieren und ihre Nachrichten signieren. WINFRIED DIFFIE und MARTIN E. HELLMAN initiierten 1976 in ihrem Artikel [DH76] eine neue Form von Kryptographie, die für den Einsatz in großen Rechnernetzen geeignet ist. Im Unterschied zu SHANNONs Ansatz wird realistischerweise davon ausgegangen, daß die Rechenkraft von unbefugten Entzifferern beschränkt ist.

Außerdem wird der Begriff der perfekten Sicherheit umgangen: ein kryptographisches Verfahren gilt schon als sicher, wenn unbefugte Entzifferer in einer vertretbaren Zeit nicht in der Lage sind, nützliche Informationen aus chiffrierten Nachrichten zu gewinnen.

DIFFIE und HELLMAN führten das Konzept der Public-Key-Kryptosysteme ein, die eine faszinierende Eigenschaft besitzen: Sender und Empfänger können auf einem öffentlichen Kommunikationskanal mit geheimen Nachrichten kommunizieren, ohne sich jemals auf einen gemeinsamen geheimen Schlüssel geeinigt zu haben. Das erste Public-Key-Kryptosystem wurde von RONALD L. RIVEST, ADI SHAMIR und LEONARD M. ADLEMAN ([ARS78]) entworfen. Das RSA-Verfahren wird auch noch heute vielfach eingesetzt.

Wer garantiert für die Sicherheit? Die moderne Kryptographie besitzt nicht nur Vorteile. Im Gegensatz zu SHANNONs Ansatz kann heutzutage niemand *beweisen*, daß ein Public-Key-Kryptosystem sicher ist: Es ist ein offenes Problem. Am Beispiel des RSA-Verfahrens läßt sich gut verdeutlichen, warum ausschließlich Experten die Sicherheit auf der Grundlage von *empirischen* Untersuchungen garantieren können. Mit einem effizienten Algorithmus zur Faktorisierung von ganzen Zahlen sind Nachrichten, die mit dem RSA-Verfahren chiffriert wurden, dechiffrierbar. Glücklicherweise kennt heutzutage niemand(?) einen derartigen Algorithmus. Daraus auf die Sicherheit des RSA-Verfahren zu schließen, wäre naiv. Auf der einen Seite gibt es evtl. andere Angriffspunkte. Auf der anderen Seite kann es sein, daß es einen effizienten Algorithmus gibt, der nahezu alle Zahlen in ihre Primfaktoren zerlegen kann. Für den Einsatz des RSA-Verfahrens müssen Zahlen bestimmt werden, die kein bekannter Algorithmus in annehmbarer Zeit faktorisieren kann. Doch wie sehen diese aus? Man muß sich auf Meinungen von Experten verlassen.

Benutzung von komplexitätstheoretischen Annahmen In der Komplexitätstheorie wird die Schwierigkeit untersucht, Probleme algorithmisch zu lösen. Genauer wird der Frage nachgegangen, wieviel Zeit und Speicherplatz eine TURING-Maschine zur Lösung eines Problems mindestens benötigt. Das ist für eine theoretische Formalisierung der modernen Kryptographie von enormer Wichtigkeit: Ein kryptographisches System soll einem legalen Benutzer nur einen geringen Zeitaufwand abverlangen, einem illegalen Benutzer dagegen einen ungeheuren Zeitaufwand.

Beschränkung auf grundlegende Funktionen Durch die Beschränkung auf wenige primitive kryptographische Funktionen soll die Anzahl der Angriffsmöglichkeiten gesenkt werden. One-Way-Funktionen sind die Grundbausteine von vielen kryptographischen Verfahren. Grob gesprochen sind One-Way-Funktionen Funktionen, die *immer* effizient zu berechnen, aber *fast immer* schwierig, d.h. mit hohem Aufwand, zu invertieren sind. Der aktuelle Stand der Komplexitätstheorie kann ihre Existenz nicht nachweisen. Dies ist auch kein Wunder, denn die Existenz einer One-Way-Funktion würde die Aussage „$\mathcal{P} \neq \mathcal{NP}$“ implizieren. Die Existenz von One-Way-Funktionen konnte bislang aber selbst unter der Annahme von $\mathcal{P} \neq \mathcal{NP}$ nicht nachgewiesen werden.

Probleme der komplexitätstheoretischen Kryptographie Der komplexitätstheoretischen Kryptographie sind einige Vorbehalte entgegenzubringen.

Zum einen ist die Komplexitätstheorie nur auf Worst-Case-Analysen fixiert. Man interessiert sich nur für die Instanzen eines Problems, die Rechnern maximale Leistung (Rechenzeit und Speicherplatz) abverlangen. Geheime Nachrichten sollen für unbefugte Entzifferer nicht im Worst-Case schwierig zu dechiffrieren zu sein, sondern im Average-Case und besser noch im „Most-Case“.

Zum anderen werden in der Komplexitätstheorie nur asymptotische Aussagen getroffen. Mit Grenzwertbetrachtungen kann nicht bewiesen werden, daß ein kryptographisches Verfahren bei einer verwendeten Schlüssellänge von 512 Bit sicher ist.

Ein möglicher Ausweg Kürzlich stellte MIKLÓS AJTAI in [Ajt96] einen Ansatz vor, mit dem es möglich ist, auf der Grundlage von etablierten komplexitätstheoretischen Annahmen beweisbar sichere Public-Key-Kryptosysteme zu konstruieren. Das IBM Research Magazine beschreibt in der 2. Ausgabe 1997 diesen Ansatz:

A Cryptographic Coup

When MIKLÓS AJTAI, a computer scientist at IBM's Almaden Research Center, revealed a major mathematical proof last year, he pointed the way to a significant advance in cryptography. He also set off a race to exploit his work for computer security. Now, AJTAI and his Almaden colleague CYNTHIA DWORK have emerged as leaders on the path to creating a practical public key encryption system based on his results. The new approach is the first cryptographic system that provides a high level of mathematically proven protection for computer data transmitted over networks.

Like conventional public key cryptography, the new system scrambles data sent over the Internet and other networks by encrypting it with a universally available public key. Only the recipient can decrypt the data. To do so, he or she uses software that, for each message, randomly generates a private key, known only to the recipient. To crack such a system, individuals could theoretically eavesdrop on transmissions electronically, in hopes of identifying private keys that are relatively easy to crack. Most methods of generating private keys, such as those based on factoring very large numbers, do occasionally produce keys that are simple to break. The Almaden researchers set out to remove that vulnerability.

AJTAI's advance focused on so-called lattice problems. AJTAI showed that every single randomly generated instance of a specially constructed lattice problem is equally difficult — and almost impossible — to solve. Then he and DWORK converted that knowledge into a working method of generating private keys.

According to PRABHAKAR RAGHAVAN, senior manager of computer science at Almaden, the new system has two advantages over current cryptographic techniques. Every possible private key is as difficult to crack as every other. Listening in on the processes of private key generation doesn't help. Eavesdroppers can gain no clues about how to break the private key, however often they monitor private key transactions. In addition, the approach permits users to adjust the level of security on a sliding scale, to comply with different governmental regulations.

The present form of the system is impractical. It requires encryption keys far longer than the messages that they encrypt, and it runs too slowly to be effective. "We'll need another reasonably good mathematical breakthrough to reach the point at which it's competitive with current cryptographic methods," says RAGHAVAN. Even when that occurs, RAGHAVAN warns, non-technical issues such as marketability will determine the technology's market appeal. Nevertheless, the second stage of the race to exploit AJTAI's advance has started, with Research among the early leaders.

Ziele dieser Diplomarbeit Das Ziel dieser Diplomarbeit ist es, den Ansatz von AJTAI vorzustellen und zu bewerten.

AJTAI zeigt, daß die Abbildung $(A, \boldsymbol{x}) \mapsto A\boldsymbol{x}$ für $A \in (\mathbb{Z}/q\mathbb{Z})^{d \times n}$, $\boldsymbol{x} \in \{0,1\}^n$ mit $q = d^6$ und $n = \lceil 2d \log_2 q \rceil$ eine One-Way-Funktion ist, wenn es für das sogenannte „shortest vector problem" keinen effizienten Lösungsalgorithmus gibt.

Das Besondere und Neue dieses Resultats ist die Zurückführung der Average-Case-Schwierigkeit (bzw. Most-Case-Schwierigkeit) der „Invertierung" der One-Way-Funktion auf die Worst-Case-Schwierigkeit des „shortest vector problem".

Es ist notwendig, die Komplexitätstheorie des „shortest vector problem" zu studieren, um AJTAIs Ergebnis zu bewerten. Gegeben sei eine Menge $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq \mathbb{R}^d$, wobei $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ linear unabhängige Vektoren des $\mathbb{R}^d$ sind. Geometrisch ist die Menge L ein Gitter. Das „shortest vector problem" besteht darin, einen kürzesten Vektor in der Menge $L\backslash\{\mathbf{0}\}$ zu finden.

Genauer beweist AJTAI, daß die Abbildung $(A, \boldsymbol{x}) \mapsto A\boldsymbol{x}$ eine One-Way-Funktion ist, wenn es keinen effizienten Algorithmus zur Berechnung einer d^9-Approximation für das „shortest vector problem" gibt. Eine d^9-Approximation ist ein Vektor aus $L\backslash\{\mathbf{0}\}$, der höchstens d^9 mal so lang ist, wie ein kürzester Vektor aus $L\backslash\{\mathbf{0}\}$. Der beste bekannte effiziente Algorithmus zur Approximation von kurzen Gittervektoren berechnet nur eine $2^{(d-1)/2}$-Approximation. Falls es $\mathcal{NP}$-hart ist, eine d^9-Approximation für das „shortest vector problem" zu berechnen und $\mathcal{NP} \neq \mathcal{RP}$ gilt, dann ist die moderne Kryptographie mit Hilfe einer etablierten komplexitätstheoretischen Annahme begründet.

DANIELE MICCIANCIO zeigte in [Mic98a], daß die Berechnung einer $\sqrt{2}-\varepsilon$-Approximation für das „shortest vector problem“ $\mathcal{NP}$-hart ist. ODED GOLDREICH und SHAFI GOLDWASSER bewiesen in [GG98] hingegen, daß die Berechnung einer $\sqrt{d/\ln d}$-Approximation für das „shortest vector problem“ unter üblichen komplexitätstheoretischen Voraussetzungen nicht $\mathcal{NP}$-hart sein kann.
AJTAIs Theorem ist ein beachtlicher Schritt für die theoretische Fundierung der Kryptographie, da es zeigt, wie der Worst-Case eines Problems auf den Average-Case eines evtl. anderen Problems zurückgeführt werden kann.
Die Frage „Ist Kryptographie eine Geheimwissenschaft?“ bzw. „Gibt es eine komplexitätstheoretische Fundierung der Kryptographie?“ kann noch nicht endgültig mit „NEIN!“ beantwortet werden.
Die Leserin und der Leser darf sich darauf freuen, an eine Spitze der kryptographischen Grundlagenforschung geführt zu werden.

Der Aufbau dieser Diplomarbeit Die Diplomarbeit ist in drei Hauptteile gegliedert:

- Grundlagen
- Zur Komplexität des „shortest vector problem“
- Kryptographische Anwendungen

Im ersten Hauptteil werden die Grundlagen, die für das Verständnis der weiteren Kapitel benötigt werden, gesammelt. Es werden Grundlagen aus drei Gebieten benötigt: Komplexitätstheorie, Kryptographie und diskrete Geometrie.
Im zweiten Hauptteil wird auf die Komplexitätstheorie des „shortest vector problem“ eingegangen. Es werden zunächst die wichtigsten algorithmischen Probleme der Gittertheorie erläutert und formalisiert. Danach wird das Ergebnis der $\mathcal{NP}$-Härte des „shortest vector problem“ von MICCIANCIO vorgestellt und eine Beweislücke der Originalarbeit geschlossen. Anschließend wird das Resultat von GOLDREICH und GOLDWASSER im Detail behandelt.
Im dritten Hauptteil wird die Konstruktion der One-Way-Funktion auf der Basis des Theorems der Worst-Case/Average-Case-Äquivalenz von AJTAI erklärt. Dort wird die Literatur um vollständige Beweise bereichert.

Kapitel 2

Grundlegendes zur Komplexitätstheorie und zu Approximationsalgorithmen

In diesem Kapitel diskutieren wir das Rechnermodell, das wir später benutzen werden, sowie die Komplexitätsklassen, die für unsere kryptographischen Untersuchungen relevant sind. Die effiziente Approximierbarkeit einer Optimierungsvariante des „shortest vector problem“ wird eine zentrale Rolle einnehmen, so daß wir an die komplexitätstheoretischen Grundlagen von Approximationsalgorithmen erinnern.

Wir werden für alle komplexitätstheoretischen Betrachtungen TURING-Maschinenmodelle benutzen, jedoch zur Beschreibung und Analyse von konkreten Algorithmen ein intuitives Rechnermodell, um unnötige Formalismen zu vermeiden. Im wesentlichen ist eine TURING-Maschine, die hier nicht näher definiert werden soll, eine mathematische Abstraktion eines Digitalcomputers.

Nach der wohlbekannten CHURCH-TURING-These stimmt die Klasse der Funktionen, die wir intuitiv als berechenbar ansehen, mit der Klasse der Funktionen überein, die mit Hilfe von TURING-Maschinen berechnet werden können. Die starke CHURCH-TURING-These besagt sogar, daß die Klasse der effizient berechenbaren Funktionen mit der Klasse der Funktionen, die durch TURING-Maschinen effizient berechnet werden können, übereinstimmt. Wir legen die starke Hypothese zugrunde, obwohl das Modell der Quantenrechner, mit denen man z.B. effizient ganze Zahlen in Primfaktoren zerlegen kann (siehe [Sho94]), die starke Hypothese widerlegen könnte. Da vermutlich Quantenrechner in der näheren Zukunft nicht gebaut werden können, erscheint der Ansatz, die starke Hypothese zugrunde zu legen, realistisch.

Zur verwendeten Literatur: Klassiker der Grundlagen der theoretischen Informatik sind [HU79] und [GJ79]. Neueres findet sich in [Weg93] und [Weg98]. Die im ersten Abschnitt definierten Begriffe haben sich etabliert, die im zweiten Abschnitt noch nicht. Sie sind im wesentlichen [MPS98], [Weg95] und [Aro94] entnommen.

2.1 Ein Überblick über einige für die Kryptographie relevanten Komplexitätsklassen

In der Komplexitätstheorie wird die Schwierigkeit untersucht, Probleme algorithmisch zu lösen. Genauer wird der Frage nachgegangen, wieviel Zeit und Speicherplatz eine TURING-Maschine zur Lösung eines Problems mindestens benötigt. Die Frage führt dazu, daß Probleme in Komplexitätsklassen eingeteilt werden. Das ist für eine theoretische Formalisierung der modernen Kryptographie, wie sie in Kapitel 3 vorgenommen wird, von enormer Wichtigkeit: Ein kryptographisches System soll einem legalen Benutzer nur einen geringen Zeitaufwand abverlangen, einem illegalen Benutzer dagegen einen ungeheuren Zeitaufwand.

Um Probleme leichter in Klassen einteilen zu können, werden sie in zwei Schritten in eine Normalform gebracht, ohne daß ihre Komplexität dadurch erheblich verändert wird. Im ersten Schritt wird ein Problem als Entscheidungsproblem — als Entscheidungsfrage — formuliert. Ein Beispiel: Es seien ein Graph G und eine natürliche Zahl k gegeben. Die Frage „Besitzt G eine Clique mit k Knoten?“ ist das Entscheidungsproblem, das zu dem Suchproblem „Finde in G eine Clique der Größe k.“ gehört. Ein Suchproblem ist nicht leichter als sein zugehöriges Entscheidungsproblem. Oft besitzen beide dieselbe Schwierigkeit, wie im Beispiel des Cliquenproblems. Im zweiten Schritt wird ein Entscheidungsproblem als ein Spracherkennungsproblem betrachtet. Von nun an sei Σ ein endliches Alphabet. Die Menge sämtlicher endlicher Buchstabenfolgen über Σ wird mit Σ^* bezeichnet und $|x|$ bezeichnet die Länge einer Buchstabenfolge $x \in \Sigma^*$. Eine Sprache L über Σ ist eine Teilmenge von Σ^*. Das Spracherkennungsproblem von L besteht darin, für eine Buchstabenfolge $x \in \Sigma^*$ herauszufinden, ob sie zu L gehört oder nicht. Um ein Entscheidungsproblem als ein Spracherkennungsproblem aufzufassen, müssen die Probleminstanzen als Buchstabenfolgen codiert werden, d.h. es muß eine bijektive Abbildung σ von der Menge der Probleminstanzen zu Σ^* gefunden werden. Die zu erkennende Sprache ist dann $L = \{x \in \Sigma^* : \text{Die Frage } \sigma^{-1}(x) \text{ läßt sich mit „Ja.“ beantworten}\}$.

Eine TURING-Maschine, die ein Problem mit geringem Aufwand lösen kann, heißt effizient. Geringer Aufwand bedeutet, daß die Zeit der Berechnung immer polynomiell von der Länge der Eingabe, einer kompakten Codierung einer Probleminstanz, abhängt.

Definition 2.1.1. Eine TURING-Maschine M mit dem Eingabealphabet Σ heißt polynomiell zeitbeschränkt, falls es ein Polynom $p \in \mathbb{R}[X]$ gibt, so daß die Berechnung von M bei Eingabe von $x \in \Sigma^*$ höchstens $p(|x|)$ Schritte benötigt.

2.1.1 Deterministische und nichtdeterministische Komplexitätsklassen

Wir unterscheiden zwischen zwei TURING-Maschinenmodellen, dem herkömmlichen deterministischen und dem probabilistischen Modell, dem zusätzlich Münzwürfe zur Entscheidungsfindung erlaubt sind.

Definition 2.1.2. Die Klasse $\mathcal{P}$ besteht aus allen Sprachen $L \subseteq \Sigma^*$, für die es eine deterministische polynomiell zeitbeschränkte TURING-Maschine M gibt, so daß für alle Eingaben $x \in \Sigma^*$ gilt:

- Falls $x \in L$ ist, akzeptiert M die Eingabe x.
- Falls $x \notin L$ ist, akzeptiert M die Eingabe x nicht.

Die nächste interessante Komplexitätsklasse ist $\mathcal{NP}$. Die Klasse $\mathcal{NP}$ besteht aus allen Sprachen L, für die es für alle $x \in L$ ein Zertifikat $y(x)$ gibt, mit dessen Hilfe effizient deterministisch

bestätigt werden kann, daß x zur Sprache L gehört. So ist z.B. die explizite Angabe einer k-elementigen Teilmenge von Knoten eines Graphen G, die eine Clique bilden, ein Zertifikat dafür, daß der Graph G eine Clique mit k Knoten besitzt. Außerdem ist dieses Zertifakt effizient deterministisch überprüfbar. Also ist die Entscheidungsvariante des Cliquenproblems in $\mathcal{NP}$ enthalten.

Definition 2.1.3. Die Klasse $\mathcal{NP}$ besteht aus allen Sprachen $L \subseteq \Sigma^*$, für die es eine deterministische polynomiell zeitbeschränkte TURING-Maschine M und ein Polynom $p \in \mathbb{R}[X]$ gibt, so daß für alle $x \in \Sigma^*$ gilt:

- Falls $x \in L$ ist, gibt es ein $y \in \Sigma^*$, $|y| \leq p(|x|)$, und M akzeptiert die Eingabe (x, y).
- Falls $x \notin L$ ist, akzeptiert M die Eingabe (x, y) bei einem beliebigen $y \in \Sigma^*$, $|y| \leq p(|x|)$, nicht.

Es ist *das* Problem der theoretischen Informatik, die allgemein nicht bezweifelte Vermutung $\mathcal{P} \neq \mathcal{NP}$ zu beweisen. Kürzlich wurde diesem Problem eine ganze Seite in der Wochenzeitung „DIE ZEIT“ gewidmet [Beh99]. Man hat die schwierigsten Spracherkennungsprobleme aus $\mathcal{NP}$ zu einer eigenen Klasse, die $\mathcal{NP}$-vollständigen Sprachen, zusammengefaßt. Die Klasse der $\mathcal{NP}$-vollständigen Sprachen ist so definiert, daß, falls eine $\mathcal{NP}$-vollständige Sprache zu $\mathcal{P}$ gehört, sofort $\mathcal{P} = \mathcal{NP}$ folgt. Genauso folgt natürlich $\mathcal{P} \neq \mathcal{NP}$, wenn eine $\mathcal{NP}$-vollständige Sprache nicht zu $\mathcal{P}$ gehört. Im folgenden beschreiben und benutzen wir das Konzept der polynomiellen Reduktionen, um die schwierigsten Sprachen aus $\mathcal{NP}$ zu definieren.

Definition 2.1.4. Eine polynomielle Reduktion („many-to-one reduction“) einer Sprache $L \subseteq \Sigma^*$ auf eine Sprache $L' \subseteq \Sigma^*$ ist eine Funktion $f : \Sigma^* \to \Sigma^*$, für die gilt:

- Die Funktion f ist von einer deterministischen polynomiell zeitbeschränkten TURING-Maschine berechenbar.
- Für alle $x \in \Sigma^*$ ist $x \in L$ genau dann, wenn $f(x) \in L'$.

Definition 2.1.5. Eine Sprache L heißt $\mathcal{NP}$-hart, falls es für jede Sprache aus $\mathcal{NP}$ eine polynomielle Reduktion auf L gibt. Eine Sprache L heißt $\mathcal{NP}$-vollständig, falls sie in $\mathcal{NP}$ liegt und $\mathcal{NP}$-hart ist. Die Klasse der $\mathcal{NP}$-vollständigen Sprachen wird mit $\mathcal{NPC}$ bezeichnet.

Eine stufenweise Erweiterung der Klasse $\mathcal{NP}$ um sogenannte $\mathcal{NP}$-Orakel, führt zur polynomiellen Hierarchie. Dabei besteht die 0-te Stufe aus der Klasse $\mathcal{P}$ und die 1-te Stufe aus der Klasse $\mathcal{NP}$. Das Wort „Hierarchie“ deutet an, daß vermutlich die k-te Stufe eine echte Teilklasse der $(k+1)$-ten Stufe ist.

Definition 2.1.6. Es sei $k \in \mathbb{Z}_{\geq 0}$. Die k-te Stufe der polynomiellen Hierarchie Σ_k besteht aus allen Sprachen L, für die es eine Sprache L' aus der Klasse $\mathcal{P}$ und ein Polynom $p \in \mathbb{R}[X]$ gibt, so daß

$$L = \{x : \exists y_1, |y_1| \leq p(|x|), \forall y_2, |y_2| \leq p(|x|), \ldots, Q y_k, |y_k| \leq p(|x|) : (x, y_1, \ldots, y_k) \in L'\}.$$

Hierbei ist $Q = \forall$, falls k gerade und $Q = \exists$, falls k ungerade ist.
Die polynomielle Hierarchie $\mathcal{PH}$ besteht aus sämtlichen Stufen der polynomiellen Hierarchie: $\mathcal{PH} = \bigcup_{k \in \mathbb{Z}_{\geq 0}} \Sigma_k$.

2.1.2 Probabilistische Komplexitätsklassen

Wir nehmen die liberale Sichtweise an, daß effiziente Berechnungen die sind, die probabilistische polynomiell zeitbeschränkte TURING-Maschinen durchführen können. Vermutlich ist die Klasse der effizient lösbaren Probleme eine echte Oberklasse von $\mathcal{P}$.

Wenn wir den Begriff der polynomiellen Reduktion etwas erweitern, kommen wir (natürlich) leichter zu komplexitätstheoretischen Aussagen. Zuweilen können wir für eine Sprache L „nur“ beweisen, daß sich alle Sprachen der Klasse $\mathcal{NP}$ auf L durch eine *randomisierte* polynomielle Reduktion zurückführen lassen:

Definition 2.1.7. Eine randomisierte polynomielle Reduktion einer Sprache $L \subseteq \Sigma^*$ auf eine Sprache $L' \subseteq \Sigma^*$ ist eine zufällige Funktion $f : \Sigma^* \to \Sigma^*$, die von einer probabilistischen polynomiell zeitbeschränkten TURING-Maschine berechnet werden kann und für die gilt:

- Falls $x \in L$ ist, beträgt die Wahrscheinlichkeit für das Ereignis „$f(x) \in L'$“ mindestens $\frac{1}{2}$.
- Falls $x \notin L$ ist, ist auch $f(x) \notin L'$.

Bislang haben wir Komplexitätsklassen nur mit Hilfe von deterministischen und nichtdeterministischen TURING-Maschinen definiert. Wir wenden uns nun probabilistischen Komplexitätsklassen zu. Die Definitionen von $\mathcal{P}$ und $\mathcal{NP}$ werden adaptiert: Wir ersetzen die deterministischen TURING-Maschinen durch probabilistische und die Quantoren durch Wahrscheinlichkeitsaussagen. Die konkreten Wahrscheinlichkeitswerte in Definition 2.1.8 sind zu einem gewissen Grad willkürlich gewählt. Durch wiederholtes Anwenden einer probabilistischen TURING-Maschinenberechnung und anschließender Majoritätsentscheidung läßt sich die Wahrscheinlichkeit eines Irrtums senken („probability amplification“), ohne die jeweilige Komplexitätsklasse zu verändern.

Definition 2.1.8. Die Klasse $\mathcal{RP}$ besteht aus allen Sprachen $L \subseteq \Sigma^*$, für die es eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M gibt, so daß für alle Eingaben $x \in \Sigma^*$ gilt:

- Falls $x \in L$ ist, beträgt die Wahrscheinlichkeit, daß M die Eingabe x akzeptiert, mindestens $\frac{1}{2}$.
- Falls $x \notin L$ ist, akzeptiert M die Eingabe x nicht.

Die Klasse $\mathcal{BPP}$ besteht aus allen Sprachen $L \subseteq \Sigma^*$, für die es eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M gibt, so daß für alle Eingaben $x \in \Sigma^*$ gilt:

- Falls $x \in L$ ist, beträgt die Wahrscheinlichkeit, daß M die Eingabe x akzeptiert, mindestens $\frac{3}{4}$.
- Falls $x \notin L$ ist, beträgt die Wahrscheinlichkeit, daß M die Eingabe x akzeptiert, höchstens $\frac{1}{4}$.

Die Klasse $\mathcal{AM}$ besteht aus allen Sprachen $L \subseteq \Sigma^*$, für die es eine polynomiell zeitbeschränkte TURING-Maschine M und ein Polynom $p \in \mathbb{R}[X]$ gibt, so daß für alle $x \in \Sigma^*$ gilt:

- Falls $x \in L$ ist, gibt es ein $y \in \Sigma^*$, $|y| \leq p(|x|)$, und die Wahrscheinlichkeit, daß M die Eingabe (x, y) akzeptiert, beträgt mindestens $\frac{3}{4}$.
- Falls $x \notin L$ ist, beträgt die Wahrscheinlichkeit, daß M die Eingabe (x, y) akzeptiert für jedes $y \in \Sigma^*$, $|y| \leq p(|x|)$, höchstens $\frac{1}{4}$.

2.1.3 Interaktive Beweissysteme

Wir charakterisieren nun die Klasse $\mathcal{AM}$ mit Hilfe von interaktiven Beweissystemen. Ein interaktives Beweissystem ist ein Spiel zwischen einem in seiner Rechenzeit beschränkten Spieler „Victor“ und einer in ihrer Rechenzeit unbeschränkten Spielerin „Peggy“. Ziel des Spieles ist es, daß Peggy Victor von der Richtigkeit einer Aussage (z.B. der Graph G besitzt eine Clique der Größe k) überzeugen möchte. Falls die Aussage richtig ist, soll Victor mit einer hohen Wahrscheinlichkeit überzeugt werden können. Falls sie falsch ist, soll Victor nur mit einer kleinen Wahrscheinlichkeit überzeugt werden können, ganz gleich, welche Argumente Peggy ihm vorträgt.

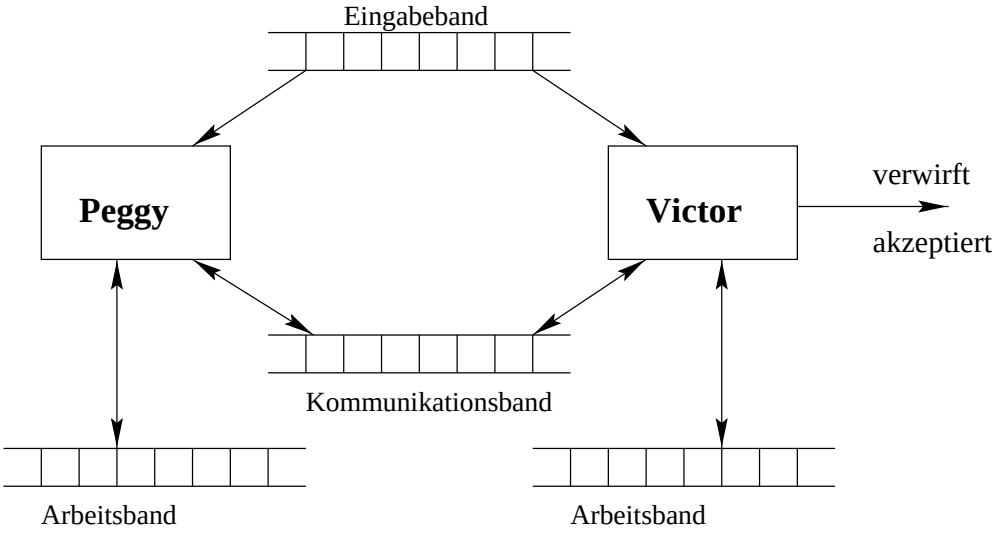

Abbildung 2.1: Ein interaktives Beweissystem.

Eine Sprache L aus der Klasse $\mathcal{NP}$ kann mit Hilfe eines interaktiven Beweissystems erkannt werden. Es sei $p \in \mathbb{R}[X]$ ein zu L gehörendes Polynom (siehe Definition 2.1.3). Es sei $x \in \Sigma^*$ eine Eingabe. Peggy berechnet $y \in \Sigma^*$, $|y| \leq p(x)$, und schickt y zu Victor, der die TURING-Maschine M aus Definition 2.1.3 besitzt. Victor akzeptiert die Eingabe x genau dann, wenn M die Eingabe (x, y) akzeptiert. Falls $x \in L$ ist, kann Peggy ein y berechnen, so daß Victor mit Hilfe von M das Paar (x, y) akzeptiert. Falls $x \notin L$ ist, akzeptiert Victor mit Hilfe von M das Paar (x, y) auf keinen Fall.

Definition 2.1.9. Die Klasse $\mathcal{IP}$ besteht aus allen Sprachen $L \subseteq \Sigma^*$, die von interaktiven Beweissystemen erkannt werden können, d.h. es gibt eine polynomiell zeitbeschränkte probabilistische TURING-Maschine V und eine in ihrer Rechenzeit unbeschränkten TURING-Maschine P, so daß für alle Eingaben $x \in \Sigma^*$ gilt:

- Falls $x \in L$ ist, beträgt die Wahrscheinlichkeit, daß V nach Kommunikation mit P die gemeinsame Eingabe x akzeptiert, mindestens $\frac{3}{4}$.
- Falls $x \notin L$ ist, beträgt die Wahrscheinlichkeit, daß V nach Kommunikation mit einer beliebigen TURING-Maschine P' die gemeinsame Eingabe x akzeptiert, höchstens $\frac{1}{4}$.

Es sei n eine nicht-negative ganze Zahl. Die Klasse $\mathcal{IP}(n)$ besteht aus allen Sprachen, die von interaktiven Beweissystemen erkannt werden können, die mit höchstens n Nachrichten zwischen P und V, bzw. zwischen V und P, auskommen.

Es ist intuitiv einsichtig, daß die Mächtigkeit von interaktiven Beweissystemen von der Anzahl der erlaubten Kommunikationsrunden abhängt.

Theorem 2.1.10. Es gelten die nachfolgenden Klassenbeziehungen:

i) $\mathcal{IP}(0) = \mathcal{BPP}$.

ii) $\mathcal{IP}(1) \supseteq \mathcal{NP}$.

iii) $\mathcal{IP}(2) = \mathcal{AM}$.

iv) Die Klasse $\mathcal{IP}(\infty) = \mathcal{IP}$ stimmt mit der Klasse der Sprachen überein, die von deterministischen polynomiell platzbeschränkten TURING-Maschinen erkannt werden können.

Die erste Aussage folgt unmittelbar aus den Definitionen. Die zweite Aussage haben wir schon weiter oben eingesehen. Für den Beweis der dritten Aussage ist es wichtig zu erkennen, daß private Münzwürfe durch öffentliche Münzwürfe simuliert werden können. Dies haben GOLDWASSER und SIPSER in [GS86] gezeigt. Die vierte Aussage ist ein nicht minder überraschendes Theorem von SHAMIR ([Sha92]).

2.1.4 Zusammenbruch von Hierarchien

Für eine Sprache $L \subseteq \Sigma^*$ definieren wir die komplementäre Sprache $\overline{L} := \Sigma^* \setminus L$. Für eine Komplexitätsklasse $\mathcal{C}$ definieren wir die komplementäre Klasse co-$\mathcal{C}$. Sie besteht aus allen Sprachen $\overline{L} \subseteq \Sigma^*$, für die es eine Sprache L aus $\mathcal{C}$ gibt. Eine Klasse $\mathcal{C}$ heißt symmetrisch, wenn $\mathcal{C} =$ co-$\mathcal{C}$ gilt. So sind z.B. $\mathcal{P}$ und $\mathcal{BPP}$ symmetrisch.

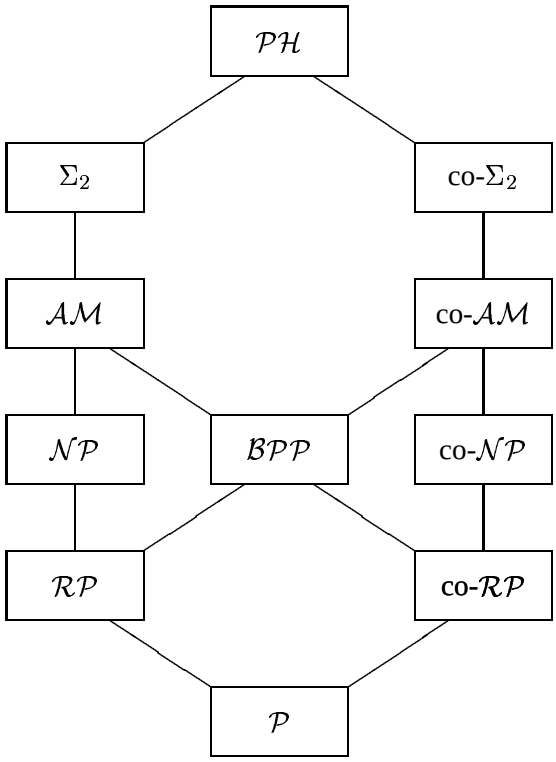

Abbildung 2.2: Inklusionsbeziehungen zwischen den angesprochenen Komplexitätsklassen.

Zwischen den angesprochenen Komplexitätsklassen gibt es viele Inklusionsbeziehungen, die bekannten sind in Abbildung 2.2 abzulesen. Nach dem heutigen Stand der Komplexitätstheorie ist es ein offenes Problem, ob zwei der Klassen sich unterscheiden. Es wird angenommen, daß je zwei der angesprochenen Klassen $\mathcal{C}_1, \mathcal{C}_2$ unterschiedlich sind, so daß Aussagen der Form „$\mathcal{C}_1 \neq \mathcal{C}_2 \Longrightarrow A$“ ein starkes Indiz für die Wahrheit der Aussage A liefern. Hierfür ist die polynomielle Hierarchie ein besonders geeignetes Werkzeug. Sie gibt uns eine unendliche Menge von Arbeitshypothesen.

Theorem 2.1.11. *(Zusammenbruch der polynomiellen Hierarchie)*
Es sei $k \in \mathbb{Z}_{\geq 0}$. Die polynomielle Hierarchie bricht zu Σ_k zusammen, d.h. es ist $\mathcal{PH} = \Sigma_k$, falls $\Sigma_k = \Sigma_{k+1}$ gilt. Sie bricht zu Σ_2 zusammen, falls $\mathcal{IP}(2) = \text{co-}\mathcal{NP}$ gilt.

2.2 Komplexitätstheorie und Approximationsalgorithmen

Wir haben im vorhergehenden Abschnitt Komplexitätsklassen für Entscheidungsprobleme behandelt. Die Beschränkung auf Entscheidungsprobleme hatte den Vorteil, daß diese als Spracherkennungsprobleme aufgefaßt werden können. Der Nachteil ist, daß sie in der Praxis weniger relevant sind als Suchprobleme. Optimierungsprobleme sind spezielle Suchprobleme, deren Lösungen nicht nur gültig, sondern sogar bestmöglich sein sollen. Bei schwierigen Optimierungsproblemen muß in der Praxis evtl. auf das Auffinden einer optimalen Lösung verzichtet und mit Approximationen gearbeitet werden.

Definition 2.2.1. Ein Optimierungsproblem besteht aus einer Menge I von Probleminstanzen, einer Funktion S, die einer Probleminstanz $w \in I$ die Menge der gültigen Lösungen zuordnet, einer Funktion v, die einer gültigen Lösung $s \in S(w)$ eine positive reelle Zahl zuordnet, und einer Variablen $g \in \{\min, \max\}$, die angibt, ob das Optimierungsproblem ein Minimierungs- oder ein Maximierungsproblem ist.
Der Wert der optimalen Lösung einer Probleminstanz $w \in I$ wird mit $\text{opt}(w)$ bezeichnet und es gilt[1] $\text{opt}(w) = g\{v(s) : s \in S(w)\}$. Eine gültige Lösung $s \in S(w)$ mit $v(s) = \text{opt}(w)$ heißt optimale Lösung.

In natürlicher Weise kann einem Optimierungsproblem (I, S, v, g) ein Entscheidungsproblem zugeordnet werden. Falls $g = \min$ ist, wird die Probleminstanz $w \in I$ zusammen mit einer positiven reellen Zahl k als Entscheidungsfrage „Gibt es $s \in S(w)$ mit $v(s) \leq k$?“ formuliert. Falls $g = \max$ ist, wird eine analoge Frage gestellt. Ein Optimierungsproblem heißt $\mathcal{NP}$-hart, wenn es das zugeordnete Entscheidungsproblem ist.

Falls ein Optimierungsproblem $\mathcal{NP}$-hart ist, gibt es unter der Voraussetzung $\mathcal{P} \neq \mathcal{NP}$ keinen effizienten deterministischen Optimierungsalgorithmus für dieses Problem. Wir können allenfalls die Existenz eines effizienten Approximationsalgorithmus erwarten. Ein Approximationsalgorithmus berechnet bei Eingabe $w \in I$ eine gültige Lösung $s \in S(w)$ und der Wert $v(s)$ approximiert den optimalen Wert $\text{opt}(w)$.

Definition 2.2.2. Es seien (I, S, v, g) ein Optimierungsproblem, $w \in I$ eine Probleminstanz und $a \in \mathbb{R}_{\geq 1}$. Eine gültige Lösung $s \in S(w)$ heißt a-Approximation für w, falls die folgenden Ungleichungen erfüllt sind:

$$\max\left\{\frac{v(s)}{\text{opt}(w)}, \frac{\text{opt}(w)}{v(s)}\right\} \leq a.$$

[1] bei naturgemäßem Mißbrauch der Notation

Die Qualität eines Approximationsalgorithmus wird durch die sogenannte Worst-Case-Güte gemessen.

Definition 2.2.3. Es sei $a \in \mathbb{R}_{\geq 1}$. Eine TURING-Maschine M berechnet eine a-Approximation für ein Optimierungsproblem, falls sie bei jeder Eingabe eine a-Approximation ausgibt. Die Worst-Case-Güte von M ist das Infimum aller $a \in \mathbb{R}_{\geq 1}$, so daß M eine a-Approximation berechnet.

Im weiteren Verlauf betrachten wir ausschließlich Minimierungsprobleme. Die angegebenen Konzepte können ohne Probleme auf Maximierungsprobleme übertragen werden.

Für ein Minimierungsproblem und ein $a \in \mathbb{R}_{\geq 1}$ kann es immer noch sehr schwierig sein, a-Approximationen zu berechnen. Es sei Π ein Minimierungsproblem. Wir möchten definieren, daß die Berechnung einer a-Approximation für Π $\mathcal{NP}$-hart ist. Die Definition muß gewährleisten, daß die Existenz eines polynomiellen deterministischen Algorithmus, der eine a-Approximation für Π berechnet, die Aussage $\mathcal{P} = \mathcal{NP}$ impliziert.

Definition 2.2.4. Es seien $c \in \mathbb{R}_{>0}$, $a \in \mathbb{R}_{\geq 1}$ und $\Pi = (I, S, v, \min)$ ein Minimierungsproblem. Wir definieren das Entscheidungsproblem Gap-(c, a)-Π wie folgt: Es sei $w \in I$ eine Instanz von Π und es gilt

- $w \in$ Gap-(c, a)-Π, wenn $\mathrm{opt}(w) \leq c$.
- $w \notin$ Gap-(c, a)-Π, wenn $\mathrm{opt}(w) > ca$.

Das Entscheidungsproblem Gap-(c, a)-Π ist ein sogenanntes Promise-Problem. Falls $a > 1$ ist, stimmt die Menge der Instanzen von Gap-(c, a)-Π nicht notwendigerweise mit der Menge der Instanzen von Π überein. Falls für eine Instanz w von Gap-(c, a)-Π die Ungleichung $\mathrm{opt}(w) > c$ gilt, kann versprochen werden, daß $\mathrm{opt}(w) > ca$ gilt. Dies ist bei der Konstruktion von polynomiellen Reduktionen zwischen Promise-Problemen zu beachten. Es dürfen nur Instanzen verwendet werden, die das Versprechen einhalten.

Definition 2.2.5. Es seien $a \in \mathbb{R}_{\geq 1}$ und $\Pi = (I, S, v, \min)$ ein Minimierungsproblem. Die Berechnung einer a-Approximation für Π ist $\mathcal{NP}$-hart, falls es ein $c \in \mathbb{R}_{>0}$ gibt, so daß das Entscheidungsproblem Gap-(c, a)-Π $\mathcal{NP}$-vollständig ist.

Proposition 2.2.6. Es seien $a \in \mathbb{R}_{\geq 1}$ und $\Pi = (I, S, v, \min)$ ein Minimierungsproblem. Falls die Berechnung einer a-Approximation für Π $\mathcal{NP}$-hart ist und es eine deterministische polynomiell zeitbeschränkte TURING-Maschine M gibt, die eine a-Approximation für Π berechnet, dann folgt $\mathcal{P} = \mathcal{NP}$.

Beweis. Es sei $c \in \mathbb{R}_{>0}$ so, daß Gap-(c, a)-Π $\mathcal{NP}$-vollständig ist. Es sei w eine Eingabe von Gap-(c, a)-Π. Die TURING-Maschine M berechne bei Eingabe von w die a-Approximation s.
1. Fall: Es gilt $v(s) \leq ca$.
Dann ist $w \in$ Gap-(c, a)-Π, denn aus $w \notin$ Gap-(c, a)-Π folgt $\mathrm{opt}(w) > ca$ und weiter $v(s) \geq \mathrm{opt}(w) > ca$.
2. Fall: Es gilt $v(s) > ca$.
Dann ist $w \notin$ Gap-(c, a)-Π, denn aus $w \in$ Gap-(c, a)-Π folgt $\mathrm{opt}(w) \leq c$ und weiter $v(s) \leq a\,\mathrm{opt}(w) \leq ca$.
Die Zugehörigkeit von w zur Sprache Gap-(c, a)-Π kann also durch eine deterministische polynomiell zeitbeschränkte TURING-Maschine entschieden werden. ◇

Ein aktuelles Resultat von HÅSTAD [Hås97] besagt z.B., daß es für jedes $\varepsilon > 0$ $\mathcal{NP}$-hart ist, eine $n^{1-\varepsilon}$-Approximationen für die Maximierungsvariante des Cliquenproblems zu berechnen (n bezeichnet die Anzahl der Knoten eines Graphen, bei dem eine Clique maximaler Größe gesucht wird). Der Versuch, effiziente deterministische Algorithmen für das Cliquenproblem zu entwerfen, ist also unter der Voraussetzung $\mathcal{P} \neq \mathcal{NP}$ aussichtslos. Aussagen dieser Art lassen sich mit Hilfe von approximationserhaltenden Reduktionen (siehe Definition 2.2.7), die in Anlehnung zu polynomiellen Reduktionen definiert sind, auf die effiziente Nicht-Approximierbarkeit anderer Optimierungsprobleme übertragen.

Wir benötigen das Konzept der approximationserhaltenden Reduktionen nur zwischen Minimierungsproblemen, so daß wir es auch nur für sie definieren. Das Konzept funktioniert zwischen Maximierungsproblemen bzw. Mischformen vollkommen analog.

Definition 2.2.7. Es seien $\Pi = (I, S, v, \min)$ und $\Pi' = (I', S', v', \min)$ Minimierungsprobleme. Eine approximationserhaltende Reduktion (eigentlich besser „gap preserving reduction“) zu den Parametern (c, a) und (c', a') von Π auf Π' ist eine Abbildung $f : I \to I'$, so daß für alle $w \in I$ gilt:

- falls $\mathrm{opt}(w) \leq c$ ist, so ist $\mathrm{opt}(f(w)) \leq c'$,
- falls $\mathrm{opt}(w) > ca$ ist, so ist $\mathrm{opt}(f(w)) > c'a'$.

Außerdem muß f durch eine deterministische polynomiell zeitbeschränkte TURING-Maschine berechnet werden können.

Proposition 2.2.8. Es seien $\Pi = (I, S, v, \min)$ und $\Pi' = (I', S', v', \min)$ Minimierungsprobleme, wobei es für ein $a \in \mathbb{R}_{\geq 1}$ $\mathcal{NP}$-hart ist eine a-Approximation für Π zu berechnen. Es sei $c \in \mathbb{R}_{>0}$ so gewählt, daß Gap-(c, a)-$\Pi \in \mathcal{NPC}$. Falls eine approximationserhaltende Reduktion $f : I \to I'$ zu den Parametern (c, a) und (c', a') von Π auf Π' existiert, so ist es ebenfalls $\mathcal{NP}$-hart, eine a'-Approximation für Π' zu berechnen.

Beweis. Die Abbildung f liefert eine polynomielle Reduktion von der Sprache Gap-(c, a)-Π auf die Sprache Gap-(c', a')-Π'. ◇

Kapitel 3

Einige Grundbegriffe der modernen Kryptographie

Die „Kryptographie“ ist ein Teilgebiet der Wissenschaft der „Kryptologie“, die sich mit Geheimschriften beschäftigt. Das besondere Merkmal der Kryptographie ist, daß chiffrierte Nachrichten öffentlich zugänglich sind, d.h. in der Kryptographie werden Verfahren entwickelt und untersucht, die es Unbefugten unmöglich machen sollen, offene, aber chiffrierte Geheimschriften zu entziffern.

Ein Szenario, das in der Kryptographie eine wichtige Rolle spielt, ist die Kommunikation in Rechnernetzwerken: Nachrichten, ob geheim oder nicht, können, wenn sie vom Sender zum Empfänger geschickt werden, von vielen Teilnehmern des Rechnernetzwerks abgehört werden.

Ein zentraler Aspekt der Kryptographie ist die Bewertung der Sicherheit von eingesetzten Verfahren. Ein kryptographisches Verfahren gilt als sicher, wenn unbefugte Entzifferer die durch dieses Verfahren chiffrierten Nachrichten mit vertretbarem Aufwand nicht dechiffrieren können. Was als vertretbarer Aufwand angesehen wird, ist der Unterschied zwischen der klassischen, informationstheoretischen und der modernen, komplexitätstheoretischen Kryptographie.

In der klassischen, informationstheoretischen Kryptographie, die SHANNON in [Sha49] begründete, besitzt der unbefugte Entzifferer unbeschränkte Rechenkraft. SHANNON definiert unter dieser rigorosen Voraussetzung *perfekte Sicherheit* als die Unmöglichkeit, nützliche Informationen aus einer chiffrierten Nachricht zu berechnen, ohne den Schlüssel zu kennen. Es ist eine bedeutende Erkenntnis von SHANNON, daß perfekte Sicherheit nur dann möglich ist, wenn die Anzahl der Bits, die Sender und Empfänger über einen öffentlichen Kommunikationskanal austauschen, höchstens so groß ist, wie die Anzahl der Bits, die sie vorher über einen geheimen Kommunikationskanal vereinbart haben.

In der modernen Kryptographie, die DIFFIE und HELLMAN in ihrem Artikel [DH76] initiierten, wird die Rechenkraft von unbefugten Entzifferern als beschränkt angesehen. Heutzutage wird realistischerweise davon ausgegangen, daß unbefugte Entzifferer nur eine probabilistische polynomiell zeitbeschränkte TURING-Maschine besitzen. Außerdem wird der Begriff der perfekten Sicherheit nicht behandelt: ein kryptographisches Verfahren gilt schon als sicher, wenn unbefugte Entzifferer in einer vertretbaren Zeit nicht in der Lage sind, nützliche Informationen aus chiffrierten Nachrichten zu gewinnen oder sie sogar zu dechiffrieren. Was unter „vertretbarer Zeit“ zu verstehen ist, ist vom Stand der Technik und von der Wichtigkeit der zu sendenden geheimen Nachricht, die in der Regel sehr schnell veraltet, abhängig. So besitzen moderne kryptographische Verfahren oft einen Sicherheitsparameter, der in Abhängigkeit von der erwarteten Rechenleistung eines unbefugten Entzifferers und von der Wichtigkeit der geheimen Nachricht, gewählt werden kann.

Eine Einführung in die moderne Kryptographie bieten [GB97], [Gol97] und [Gol95], an denen sich dieses Kapitel orientiert. Wir gehen hier nur auf die zentralen Begriffe „One-Way-Funktion“ und „Public-Key-Kryptosystem“ ein. Eine Standardreferenz für Kryptographie ist das „Handbook of applied cryptography“ [MVV97].

3.1 One-Way-Funktionen

Grob gesprochen sind One-Way-Funktionen Funktionen, die effizient zu berechnen, aber schwierig zu invertieren sind. One-Way-Funktionen sind wichtige Grundbausteine von vielen kryptographischen Verfahren. Da die Existenz von One-Way-Funktionen die Aussage „$\mathcal{P} \neq \mathcal{NP}$“ impliziert, kann man heute mit Hilfe der Komplexitätstheorie die Existenz nicht nachweisen. Bislang kann man das selbst dann nicht, wenn die Hypothese „$\mathcal{P} \neq \mathcal{NP}$“ bzw. sogar „$\mathcal{RP} \neq \mathcal{NP}$“ angenommen.

Wie schon angedeutet, gehen wir davon aus, daß unbefugte Entzifferer nur eine probabilistische polynomiell zeitbeschränkte (das beschränkende Polynom ist beliebig, aber fest) TURING-Maschine besitzen. Falls unbefugte Entzifferer mit einer positiven Wahrscheinlichkeit nützliche Informationen über chiffrierte Nachrichten gewinnen können — und das ist immer schon durch Raten möglich —, können sie die Wahrscheinlichkeit durch polynomiell viele Berechnungsversuche erhöhen. Diese Wahrscheinlichkeit muß bei einer sicheren kryptographischen Funktion unerheblich sein.

Definition 3.1.1. Eine von $n \in \mathbb{N}$ abhängige nicht-negative Funktion $\nu : \mathbb{N} \to \mathbb{R}_{\geq 0}$ heißt unerheblich („negligible“), wenn sie für $n \to \infty$ schneller gegen 0 konvergiert als der Quotient $1/|p(n)|$ für jedes Polynom $p \in \mathbb{R}[X]$, d.h. für genügend großes n gilt stets $\nu(n) < 1/|p(n)|$.

Ist die Erfolgswahrscheinlichkeit eines probabilistischen Algorithmus unerheblich, so ist sie nach polynomiell vielen Wiederholungen des Algorithmus immer noch unerheblich.

Eine einfache Anwendung von One-Way-Funktionen findet sich in der Login-Prozedur für einen geschützten Systemzugang (z.B. bei Rechnern, die mehreren Personen zugänglich sind, Geld-Automaten, etc.): Alice gibt ihr Paßwort an einem abhörsicheren Terminal ein. Das Paßwort wird mit Hilfe einer One-Way-Funktion chiffriert. Die Systemzentrale sendet das Paßwörterverzeichnis über einen öffentlichen Kommunikationskanal an das Terminal. Das Paßwörterverzeichnis enthält eine Liste der chiffrierten Paßwörter sämtlicher Personen, die Zugang zu dem System erhalten dürfen. Alice' chiffrierte Eingabe wird mit dem entsprechenden Eintrag im Paßwörterverzeichnis verglichen und bei Übereinstimmung erhält sie einen Zugang zum System. Durch dieses Protokoll wird nur eine sichere Authentifikation von Systemzugängen ermöglicht, andere Sicherheitsprobleme werden aber dadurch nicht gelöst. Es ist z.B. nicht klar, wie Alice' chiffriertes Paßwort sicher in das Paßwörterverzeichnis gelangt.

Definition 3.1.2. Sei Σ ein endliches Alphabet mit $1 \in \Sigma$. Eine Funktion $f : \Sigma^* \to \Sigma^*$ heißt starke One-Way-Funktion, falls sie die folgenden Bedingungen erfüllt:

i) (Effizient zu berechnen:) Es gibt eine probabilistische polynomiell zeitbeschränkte TURING-Maschine, die bei Eingabe von $x \in \Sigma^*$ den Funktionswert $f(x)$ berechnet.

ii) (Schwierig zu invertieren:) Für jede probabilistische polynomiell zeitbeschränkte TURING-Maschine M gibt es eine unerhebliche Funktion ν_M, so daß für genügend großes n gilt:

$$\Pr[f(z) = y \mid y = f(x) \text{ und } M \text{ berechnet bei Eingabe } (1^n, y) \text{ den Wert } z] \leq \nu_M(n),$$

wobei die Wahrscheinlichkeit über die zufällige Wahl von $x \in \Sigma^n$ und über die Münzwürfe von M genommen wird.

Ein paar Bemerkungen zur Definition 3.1.2: Es wird nicht verlangt, die Funktion f zu invertieren, sondern es sollen Urbilder bzgl. f berechnet werden. Die Eingabe 1^n sichert der TURING-Maschine M zu, daß sie genügend Zeit besitzt, um bei Eingabe y ein Urbild x abzuspeichern.

Wenn z.B. für $x \in \Sigma^n$ immer $|f(x)| \in o(\log n)$ gilt, kann M bei Eingabe von $f(x)$ in polynomieller Zeit x nicht berechnen.

Einen anderen Ansatz, One-Way-Funktionen zu definieren, bieten die sogenannten schwachen One-Way-Funktionen. Wie der Name schon suggeriert, müssen schwache im Gegensatz zu starken One-Way-Funktionen schwächeren Bedingungen genügen. Schwache One-Way-Funktionen werden betrachtet, weil ihre Existenz möglicherweise leichter als die Existenz von starken One-Way-Funktionen nachweisbar ist.

Definition 3.1.3. Sei Σ ein endliches Alphabet mit $1 \in \Sigma$. Eine Funktion $f : \Sigma^* \to \Sigma^*$ heißt schwache One-Way-Funktion, falls sie die folgenden Bedingungen erfüllt:

i) (Effizient zu berechnen:) Es gibt eine probabilistische polynomiell zeitbeschränkte TURING-Maschine, die bei Eingabe von $x \in \Sigma^*$ den Funktionswert $f(x)$ berechnet.

ii) (Ein polynomieller Anteil der Eingaben ist schwierig zu invertieren:) Es gibt ein Polynom $p \in \mathbb{R}[X]$, so daß für jede probabilistische polynomiell zeitbeschränkte TURING-Maschine M bei genügend großem n gilt:

$$\Pr[f(z) \neq y \mid y = f(x) \text{ und } M \text{ berechnet bei Eingabe } (1^n, y) \text{ den Wert } z] \geq \frac{1}{|p(n)|},$$

wobei die Wahrscheinlichkeit über die zufällige Wahl von $x \in \Sigma^n$ und über die Münzwürfe von M genommen wird.

Der wesentliche Unterschied zwischen starken und schwachen One-Way-Funktionen ist der, daß starke One-Way-Funktionen auf allen bis auf einen unerheblichen Anteil der möglichen Eingaben schwierig zu invertieren sein müssen, dagegen schwache One-Way-Funktionen nur auf einem polynomiellen Anteil der möglichen Eingaben. Obwohl die Anforderungen an schwache One-Way-Funktionen weniger rigoros als die Anforderungen an starke One-Way-Funktionen erscheinen, läßt sich aus jeder schwachen One-Way-Funktion eine starke konstruieren.

Theorem 3.1.4. Aus jeder schwachen One-Way-Funktion läßt sich eine starke One-Way-Funktion konstruieren. Insbesondere existieren genau dann starke One-Way-Funktionen, wenn schwache One-Way-Funktionen existieren.

Hier wollen wir uns noch nicht anstrengen, deswegen nur ein paar Worte zur Beweisidee (siehe [Gol95]): Da eine starke One-Way-Funktion auch eine schwache One-Way-Funktion ist, ist eine der Implikationen sofort klar. Für die andere Implikation wird mit einer schwachen One-Way-Funktion f eine starke One-Way-Funktion g durch $g(x_1 \circ \cdots \circ x_n) := f(x_1) \circ \cdots \circ f(x_n)$, das Symbol $\circ$ steht für die Konkatenation von Buchenstabenfolgen, konstruiert. Dabei sind n und die Länge von $x_i \in \Sigma^*$ geeignet gewählt, $i = 1, \ldots, n$. Die richtige Wahl dieser Parameter erfordert einigen technischen Aufwand, genau wie der Nachweis, daß g schwierig zu invertieren ist. Dies geschieht durch eine „kryptographische Reduktion“: Angenommen es gibt eine probabilistische polynomiell zeitbeschränkte TURING-Maschine, die g in nicht unerheblich vielen Fällen invertieren kann, dann läßt sich mit deren Hilfe eine probabilistische polynomiell zeitbeschränkte TURING-Maschine konstruieren, die f auf einem größeren als einen polynomiellen Anteil der möglichen Eingaben invertieren kann.
Im folgenden werden wir die Begriffe starke One-Way-Funktion und schwache One-Way-Funktion unter dem Begriff One-Way-Funktion zusammenfassen, falls die getroffenen Aussagen für beide Begriffe zutreffen.

3.2 Beispiele

Streng genommen ist die Überschrift dieses Abschnitts nur eine Vermutung: die heutige Komplexitätstheorie ist nicht in der Lage, auch nur ein Beispiel für eine One-Way-Funktion zu liefern. Selbst mit Hilfe der allgemein nicht bezweifelten Vermutung „$\mathcal{P} \neq \mathcal{NP}$“ konnte die Existenz von One-Way-Funktionen bislang nicht nachgewiesen werden.

Es wird jedoch vermutet bzw. gehofft, daß das Potenzieren im Restklassenkörper $\mathbb{Z}/p\mathbb{Z}$, p eine Primzahl, und die Multiplikation von zwei ganzen Zahlen One-Way-Funktionen sind. Gleichzeitig sind dies die prominentesten Kandidaten für One-Way-Funktionen und werden in der Praxis in kryptographischen Verfahren eingesetzt. Daß die beiden Funktionen One-Way-Funktionen sind, ist wichtig, weil sie dann im Rahmen der hier vorgestellten komplexitätstheoretischen Formalisierung der modernen Kryptographie eingesetzt werden können, um beweisbar sichere kryptographische Funktionen zu realisieren. Die Funktionen werden hier behandelt, damit die in diesem Kapitel definierten Begriffe nicht blutleer bleiben.

3.2.1 Das Problem des diskreten Logarithmus

Ein weiteres zahlentheoretisches Problem, von dem vermutet wird, daß es zu seiner Lösung keinen effizienten Algorithmus gibt, ist das Problem des diskreten Logarithmus in Restklassenkörpern. Das Problem des diskreten Logarithmus besteht darin, bei gegebenem (p, g, x), p eine Primzahl, $g \in (\mathbb{Z}/p\mathbb{Z})^\times = (\mathbb{Z}/p\mathbb{Z})\backslash\{0\}$ ein erzeugendes Element von $(\mathbb{Z}/p\mathbb{Z})^\times$ und $x \in (\mathbb{Z}/p\mathbb{Z})^\times$ die Zahl $\alpha \in \{1, 2, \ldots, p-1\}$ mit $g^\alpha = x$ zu finden. Der schnellste bekannte Algorithmus zur Lösung des Problems ist der Index-Calculus-Algorithmus [Odl85], der bei gegebener Primzahl p eine erwartete subexponentielle Laufzeit von $O(e^{\sqrt{\ln p \ln\ln p}})$ besitzt. Das Problem des diskreten Logarithmus hat sich nicht nur im Worst-Case als schwierig erwiesen, sondern auch im Average-Case. Diese Aussage präzisiert die nachfolgende Vermutung (Strong Discrete Logarithm Assumption, [GB97]).

Vermutung 3.2.1. Für jede probabilistische polynomiell zeitbeschränkte TURING-Maschine M, für jedes Polynom $q \in \mathbb{R}[X]$, und für genügend großes n gilt stets

$$\Pr[M \text{ berechnet bei der Eingabe } (p, g, x) \text{ das } \alpha \in \{1, \ldots, p-1\} \text{ mit } g^\alpha = x] < \frac{1}{|q(n)|},$$

wobei die Wahrscheinlichkeit über alle Primzahlen p mit $p \leq n$, alle erzeugenden Elemente $g \in (\mathbb{Z}/p\mathbb{Z})^\times$ von $(\mathbb{Z}/p\mathbb{Z})^\times$ und die Münzwürfe von M genommen wird.

Unter Vermutung 3.2.1, ist das Potenzieren in Restklassenkörpern $(p, g, \alpha) \mapsto (p, g, g^\alpha)$, p prim, $g \in (\mathbb{Z}/p\mathbb{Z})^\times$ erzeugt $(\mathbb{Z}/p\mathbb{Z})^\times$ und $\alpha \in \{1, \ldots, p-1\}$, eine starke One-Way-Funktion. Weiterführendes zum Problem des diskreten Logarithmus findet sich in dem Übersichtsartikel [Odl85] von ODLYZKO. Ergänzend soll erwähnt werden, daß das Problem des diskreten Logarithmus in jeder endlichen abelschen Gruppe formuliert werden kann. Dort ist es im allgemeinen nicht leichter zu lösen als das Problem des diskreten Logarithmus in Restklassenkörpern.

3.2.2 Faktorisieren von ganzen Zahlen

In den letzten Jahrzehnten wurde intensiv nach einem effizienten Algorithmus für das Faktorisieren von ganzen Zahlen geforscht. Eine ganze Zahl $n \in \mathbb{Z}$ zu faktorisieren bedeutet, die eindeutig bestimmten paarweise verschiedenen Primzahlen $p_1, \ldots, p_r \in \mathbb{N}$ und die eindeutig bestimmten Exponenten $e_1, \ldots, e_r \in \mathbb{N}$ zu finden, so daß die Gleichung $\pm n = p_1^{e_1} \cdots p_r^{e_r}$ erfüllt ist. Das

„Zahlkörpersieb“ [Len93] ist zur Zeit der effizienteste bekannte Faktorisierungsalgorithmus. Unter realistischen zahlentheoretischen Annahmen konnte gezeigt werden, daß das Zahlkörpersieb eine Zahl $n \in \mathbb{N}$ in einer erwarteten Laufzeit von $O(e^{\sqrt[3]{\ln n(\ln\ln n)^2}(C+o(1))})$ mit $C = \sqrt[3]{64/9}$ faktorisieren kann. Die Laufzeit ist zwar subexponentiell, aber von polynomiell weit entfernt.

Wir wollen eine Funktion definieren, die schwierig zu „invertieren“ ist, wenn das Faktorisieren von ganzen Zahlen schwierig ist. Wenn wir die Funktion $f : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$, $f(x, y) := xy$ betrachten, ist es klar, daß f auf mehr als der Hälfte der ganzen Zahlen effizient zu „invertieren“ ist. Also ist f keine starke One-Way-Funktion.

Wenn wir zusätzlich annehmen, daß das Faktorisieren eines Produkts von zwei verschiedenen, etwa gleich großen Primzahlen p, q (die Bitlängen von p und q stimmen überein, d.h. $\lceil \log_2 p \rceil = \lceil \log_2 q \rceil$), schwierig ist, dann ist f eine schwache One-Way-Funktion. Für den Beweis dieser Aussage wird eine Variante des Primzahlsatzes benötigt.

Theorem 3.2.2. Für $n \geq 17$ gelten für die Anzahl der Primzahlen kleiner oder gleich n die Ungleichungen ([MVV97], Fact 2.96):

$$\frac{n}{\ln n} < \pi(n) := |\{p \in \mathbb{N} : p \text{ eine Primzahl}, p \leq n\}| < 1,25506\frac{n}{\ln n}.$$

Damit läßt sich die Wahrscheinlichkeit abschätzen, daß eine Zahl $n \geq 17$ der Bitlänge $2k$, $k \in \mathbb{N}$, aus zwei Primzahlen p, q der Bitlänge k zusammengesetzt ist. Es gilt

$$\begin{aligned}
&\Pr_{\lceil \log_2 n \rceil = 2k}[n = pq, p, q \text{ prim}, \lceil \log_2 p \rceil = \lceil \log_2 q \rceil = k] \\
= \;& \frac{1}{2} \Pr_{\lceil \log_2 p \rceil = k}[p \text{ prim}] \cdot \Pr_{\lceil \log_2 q \rceil = k}[q \text{ prim}] \\
= \;& \frac{1}{2}\left(\frac{\pi(2^k) - \pi(2^{k-1})}{2^{k-1}}\right)^2 \\
\geq \;& \frac{1}{2^{2k-1}}\left(\frac{2^k}{k \ln 2} - \frac{1,25506 \cdot 2^{k-1}}{(k-1)\ln 2}\right)^2 \\
\geq \;& \frac{1}{2}\left(\frac{0,77494k - 2}{k(k-1)}\right)^2 \\
\geq \;& \frac{1}{2k^4}.
\end{aligned}$$

Durch eine genauere Analyse läßt sich eine bessere Schranke der Größenordnung $1/k^2$ angeben, aber die obige Schranke ist für unsere Zwecke vollkommen ausreichend. Da die Wahrscheinlichkeit $\Pr_{n \in \mathbb{N}}[2 \text{ teilt } \lceil \log_2 n \rceil] = \frac{1}{2}$ ist, beträgt die Wahrscheinlichkeit, daß die Ausgabe von f eine Zahl ist, die Produkt zweier Primzahlen gleicher Bitlänge ist, wenigstens $\frac{1}{4k^4}$. Also ist f unter der Voraussetzung, daß alle probabilistischen polynomiell zeitbeschränkten TURING-Maschinen nur mit einer unerheblichen Erfolgswahrscheinlichkeit Zahlen faktorisieren können, die Produkt zweier Primzahlen gleicher Bitlänge sind, eine schwache One-Way-Funktion.

3.3 Public-Key-Kryptosysteme

Ein Public-Key-Kryptosystem besteht aus drei Komponenten: einem Verfahren zur Erzeugung von Schlüsselpaaren, einer Chiffrier- und einer DechiffrierFunktion. Public-Key-Kryptosysteme besitzen eine faszinierende Eigenschaft: Sender und Empfänger können auf einem öffentlichen

Kommunikationskanal mit geheimen Nachrichten kommunizieren, ohne sich jemals auf einen gemeinsamen geheimen Schlüssel geeinigt zu haben. Diese Eigenschaft ist in der informationstheoretischen Kryptographie unmöglich, da sie SHANNONs Forderung nach perfekter Sicherheit widerspricht. Aber auch hier gilt wieder, daß die Existenz eines Public-Key-Kryptosystems bislang nicht nachgewiesen werden konnte.

Definition 3.3.1. Ein Public-Key-Kryptosystem besteht aus drei probabilistischen TURING-Maschinen (G, E, D), die die folgenden Eigenschaften besitzen. Gegeben seien ein endliches Alphabet Σ mit $1 \in \Sigma$ und ein Sicherheitsparameter $n \in \mathbb{N}$.

i) (Schlüsselerzeugung, „key generation“:) Die probabilistische polynomiell platzbeschränkte TURING-Maschine G produziert bei Eingabe von 1^n in erwarteter polynomieller Zeit ein Schlüsselpaar $(e, d) \in \Sigma^* \times \Sigma^*$, wobei e der öffentliche und d der private Schlüssel genannt werden (Notation: $(e, d) \in G(1^n)$).

ii) (Chiffrierung, „encryption“:) Sei $(e, d) \in G(1^n)$ ein Schlüsselpaar. Die probabilistische TURING-Maschine berechnet bei Eingabe von $(1^n, e, m)$, $m \in \Sigma^*$ eine Nachricht, in polynomieller Zeit ein Kryptogramm $c \in \Sigma^*$ (Notation: $c \in E(1^n, e, m)$).

iii) (Dechiffrierung, „decryption“:) Sei $(e, d) \in G(1^n)$ ein Schlüsselpaar und $c \in E(1^n, e, m)$ ein Kryptogramm. Die probabilistische TURING-Maschine D berechnet bei Eingabe von $(1^n, d, c)$ in polynomieller Zeit eine Nachricht $m' \in \Sigma^*$. Dabei gilt für alle Schlüsselpaare $(e, d) \in G(1^n)$, für alle Nachrichten $m \in \Sigma^*$ und für jedes $c \in E(1^n, e, m)$, daß die Wahrscheinlichkeit $\Pr[m \neq m']$ unerheblich ist (Notation: $m' \in D(1^n, d, c)$).

Wenn ein Public-Key-Kryptosystem (G, E, D) gegeben ist, kann es unmittelbar zur abhörsicheren Kommunikation zwischen Teilnehmern in einem Rechnernetzwerk eingesetzt werden. Zuerst vereinbaren sämtliche Teilnehmer einen Sicherheitsparameter $n \in \mathbb{N}$. Die Teilnehmerin Alice benutzt die TURING-Maschine G mit Eingabe 1^n, um ihr Schlüsselpaar (e_A, d_A) zu erhalten. Ihren öffentlichen Schlüssel e_A veröffentlicht sie in einem für jeden Teilnehmer zugänglichen Schlüsselverzeichnis und ihren privaten Schlüssel d_A legt sie so ab, daß er nur für sie zugänglich ist. Wenn nun Bob die Nachricht m zu Alice senden möchte, schaut er im öffentlichen Schlüsselverzeichnis, von dessen Richtigkeit er überzeugt ist, nach Alice' öffentlichem Schlüssel e_A. Dann chiffriert er die Nachricht mit der TURING-Maschine E bei Eingabe von $(1^n, e_A, m)$, erhält $c \in E(1^n, e_A, m)$ und sendet das Kryptogramm c zu Alice. Nachdem Alice c erhalten hat, kann sie mit der TURING-Maschine D bei Eingabe von $(1^n, d_A, c)$ mit hoher Wahrscheinlichkeit die ursprüngliche Nachricht m dechiffrieren.

Zur Definition 3.3.1 ist zu bemerken, daß Chiffrierung und Dechiffrierung nicht deterministisch sein müssen. Außerdem schwächen wir die übliche Anforderung an ein kryptographisches System, daß für alle $n \in \mathbb{N}$, $(e, d) \in G(1^n)$ und $m \in \Sigma^*$ die Gleichung $D(1^n, d, E(1^n, e, m)) = m$ gilt, ab. Es wird nur gefordert, daß sie, bzw. eine analoge Aussage (E und D realisieren nicht notwendig Abbildungen), mit hoher Wahrscheinlichkeit gilt. Daß der Chiffrieralgorithmus E deterministisch arbeitet, ist sogar unerwünscht, weil bei deterministischem E eine Eingabe immer auf dasselbe Kryptogramm abgebildet wird und ein unbefugter Entzifferer damit nützliche Informationen bekommen kann. Insbesondere kann probabilistische Chiffrierung die Anwendung von „chosen plain text attacks“ (ein unbefugter Entzifferer chiffriert selbstgewählte Nachrichten, um daraus Informationen über abgehörte chiffrierte Nachrichten zu gewinnen) erschweren. Da G polynomiell platzbeschränkt ist, sind es die Längen von d und e auch, d.h. es gibt ein Polynom $p \in \mathbb{R}[X]$, so daß für alle $n \in \mathbb{N}$ und für alle $(e, d) \in G(1^n)$ stets $|(e, d)| \leq p(n)$ gilt. Wir können also für alle komplexitätstheoretischen Betrachtungen den Sicherheitsparameter n als Eingabelänge zugrunde legen.

In Definition 3.3.1 ist noch offen geblieben, was ein *sicheres* Public-Key-Kryptosystem ist. Bevor wir diese Lücke schließen, stellen wir informale Anforderungen zusammen, die ein sicheres Public-Key-Kryptosystem erfüllen muß:

- Aus einem öffentlichen Schlüssel darf der zugehörige private Schlüssel nicht effizient berechenbar sein.
- Aus einer chiffrierten Nachricht dürfen keine nützlichen Eigenschaften der Nachricht effizient berechenbar sein, wobei der öffentliche Schlüssel, mit dem die Nachricht chiffriert wurde, als bekannt vorausgesetzt wird.
- Unabhängig von der Wahrscheinlichkeitsverteilung auf dem Nachrichtenraum dürfen Teile von chiffrierten Nachrichten nicht effizient dechiffriert werden können.
- Ein unbefugter Entzifferer darf keine nützliche Information durch das Abhören mehrerer chiffrierter Nachrichten effizient berechnen können. So soll er z.B. nicht effizient erkennen können, ob zweimal dieselbe Nachricht gesendet wurde.

Man kann sich die Anforderungen an ein sicheres Public-Key-Kryptosystem mit einem alltäglichen Beispiel verdeutlichen. Ein sicheres Public-Key-Kryptosystem besitzt im wesentlichen die Eigenschaften eines undurchsichtigen Briefumschlags. Alice schreibt eine Nachricht auf ein Blatt Papier, steckt es in einen undurchsichtigen Briefumschlag und sendet den kompletten Brief zu Bob. In diesem Modell kann nur Bob den Brief öffnen und dann die Nachricht lesen. Sichere Public-Key-Kryptosysteme besitzen also sogar eine weitaus größere Sicherheit als der alltägliche Briefverkehr.

Definition 3.3.2. Ein Public-Key-Kryptosystem (G, E, D) heißt sicher, wenn für alle probabilistischen polynomiell zeitbeschränkten TURING-Maschinen A und M und für alle Polynome $p \in \mathbb{R}[X]$ und genügend großes n stets

$$\Pr[\ \begin{array}{l} M \text{ berechnet bei Eingabe } (1^n, e, m_1, m_2, c) \text{ die Nachricht } m \mid \\ (e,d) \in G(1^n),\ M \text{ berechnet bei Eingabe von } (1^n) \text{ die} \\ \text{Nachrichten } (m_1, m_2),\ m \in \{m_1, m_2\},\ c \in E(1^n, e, m) \end{array} \ ] < \tfrac{1}{2} + \tfrac{1}{|p(n)|}$$

gilt. Dabei wird die Wahrscheinlichkeit über die Münzwürfe von G, M, E und A, sowie über die zufällige Wahl von m aus $\{m_1, m_2\}$ genommen.

Die TURING-Maschine M erzeugt Nachrichten m_1, m_2 mit einer nicht vorher festgelegten Wahrscheinlichkeitsverteilung. Diese Nachrichten kann die TURING-Maschine A, nachdem sie durch E chiffriert wurden, nicht unterscheiden.

An dieser Stelle muß gewarnt werden! Bislang haben wir uns nur um die Sicherheit eines Public-Key-Kryptosystems gegenüber passives Abhören gekümmert. Das Management der öffentlichen Schlüssel und aktive Angriffe (von physikalischen Angriffen mal abgesehen), wie z.B. das Abhören und *Verändern* von Nachrichten, sind natürlich auch zu berücksichtigen, wenn es darum geht, ein wirklich sicheres Public-Key-Kryptosystem zu entwerfen ...

Kapitel 4

Einige Grundbegriffe der diskreten Geometrie

In diesem Kapitel werden grundlegende Definitionen und Aussagen der diskreten und kombinatorischen Geometrie gesammelt, und die Notation wird festgelegt. Es werden die Grundbegriffe der Geometrie der Zahlen und der Theorie der konvexen Polytope vorgestellt.

Damit dieses Kapitel nicht unverhältnismäßig lang wird, sind viele der aufgelisteten Ergebnisse nur zitiert. Für eine ausführliche Darstellung sei auf die Standardwerke der Geometrie der Zahlen [GL87], [CS88] bzw. den Artikel [Lag95] aus dem „Handbook of Combinatorics“, sowie auf ein Standardwerk der Theorie der konvexen Polytope [Zie95] verwiesen.

Im folgenden sei E ein d-dimensionaler $\mathbb{R}$-Vektorraum, der ein Skalarprodukt $(\cdot,\cdot) : E \times E \to \mathbb{R}$ besitzt und durch $\|\cdot\| := \sqrt{(\cdot,\cdot)}$ normiert ist, d.h. das Paar $(E,(\cdot,\cdot))$ ist ein euklidischer Vektorraum und das Paar $(E,\|\cdot\|)$ ist ein BANACH-Raum. Modellhaft kann man sich $E = \mathbb{R}^d$ mit dem Skalarprodukt $(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{d} x_i y_i$ vorstellen. Die Gitter, die wir betrachten, besitzen die nicht genauer spezifizierte Dimension $n \leq d$. Manchmal ist es jedoch notwendig, daß das betrachtete Gitter die volle Dimension d besitzt.

4.1 Elementare Eigenschaften von Gittern

Zunächst wird der Begriff des Gitters definiert, sowie Begriffe, die die wichtigsten geometrischen Eigenschaften eines Gitters beschreiben. Insbesondere wird festgestellt, daß Gitter nur endlich viele nicht-triviale kürzeste Vektoren besitzen. Das Problem, kurze Vektoren in einem gegebenen Gitter zu finden, wird in den nächsten Kapiteln ein zentrales Thema sein.
Anschließend werden die MINKOWSKIschen Gitterpunktsätze angesprochen, die Abschätzungen für die Länge der kürzesten Vektoren eines Gitters liefern. Das Problem, eine Basis eines Gitters zu finden, die aus möglichst kurzen Vektoren besteht, ist ein Hauptproblem der Reduktionstheorie von Gittern. Wir werden die Grundlagen der Reduktionstheorie von KORKINE und ZOLOTAREV kennenlernen.

4.1.1 Gitter, Gitterbasen und Gitterprojektionen

Definition 4.1.1. Eine Teilmenge $L \subseteq E$ heißt Gitter, falls es $0 \leq n \leq d$ linear unabhängige Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ gibt, so daß sich L schreiben läßt als

$$L = \left\{ \sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i : \alpha_i \in \mathbb{Z} \right\} = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_n.$$

Das n-Tupel $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ heißt Basis von L. Man sagt: das Gitter L ist n-dimensional.

Die obige Definition bezieht sich auf die Wahl linear unabhängiger Vektoren, die aber alles andere als eindeutig ist. Es sei $n \in \mathbb{N}$ eine natürliche Zahl. Mit $\mathsf{GL}_n(\mathbb{Z}) := \{A \in \mathbb{Z}^{n\times n} : \det A = \pm 1\}$ wird die Gruppe der ganzzahligen unimodularen Transformationen bezeichnet. Es seien $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in E$ linear unabhängige Vektoren, $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_n$ das entsprechende Gitter und sei $A = (\alpha_{ij})_{1\leq i,j\leq n} \in \mathsf{GL}_n(\mathbb{Z})$ eine ganzzahlige unimodulare Transformation, dann ist $\sum_{j=1}^{n} \alpha_{ij}\boldsymbol{b}_j$, $i = 1, \ldots, n$, eine weitere Basis von L. Jede Basis von L entsteht aus $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ durch eine ganzzahlige unimodulare Transformation.

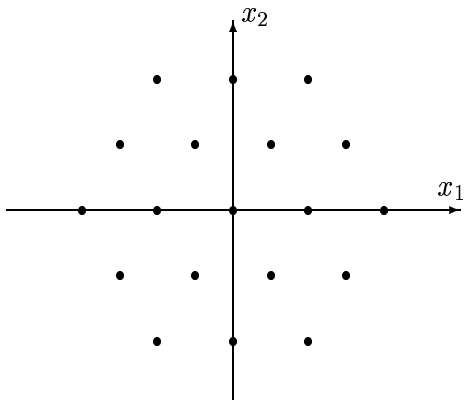

Abbildung 4.1: Das zweidimensionale hexagonale Gitter $\mathsf{A}_2 = \mathbb{Z}\binom{1}{0} \oplus \mathbb{Z}\binom{-1/2}{\sqrt{3}/2}$.

Die nachfolgende Proposition zeigt, daß es möglich ist, Gitter ohne Angabe einer Basis zu definieren.

Proposition 4.1.2. Eine Untergruppe $(L, +)$ von $(E, +)$ ist genau dann ein Gitter, wenn sie diskret ist, d.h. wenn L keinen Häufungspunkt in E besitzt.

Beweis. Siehe [Neu92], Satz 4.2. ◇

Notation 4.1.3. Es sei $F \subseteq E$ ein Untervektorraum von E. Es gilt $E = F \oplus F^\perp$ mit $F^\perp := \{\boldsymbol{x} \in E : (\boldsymbol{x}, \boldsymbol{y}) = 0 \text{ für alle } \boldsymbol{y} \in E\}$, d.h. jedes $\boldsymbol{x} \in E$ läßt sich eindeutig schreiben als $\boldsymbol{x} = \boldsymbol{x}_1 + \boldsymbol{x}_2$, wobei $\boldsymbol{x}_1 \in F$ und $\boldsymbol{x}_2 \in F^\perp$. Mit $\pi_F : E \to F$ wird die orthogonale Projektion von E auf F bezeichnet: $\pi_F(\boldsymbol{x}) = \pi_F(\boldsymbol{x}_1 + \boldsymbol{x}_2) = \boldsymbol{x}_1$.

Anders als in der Vektorraumtheorie sind die Bilder von Gittern unter linearen Abbildungen im allgemeinen keine Gitter. Als einfachstes pathologisches Beispiel ist das Bild des Gitters $\mathbb{Z}^2$ unter der linearen Abbildung $f : \mathbb{R}^2 \to \mathbb{R}$ mit $f(\boldsymbol{x}) = x_1 + \sqrt{2}x_2$ zu nennen: $f(\mathbb{Z}^2)$ ist nicht diskret. Die Situation sieht deutlich besser aus, wenn orthogonale Projektionen auf orthogonale Vektorraumkomplemente betrachtet werden, wie Proposition 4.1.4 zeigt.

Proposition 4.1.4. Es sei $L \subseteq E$ ein n-dimensionales Gitter und seien $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{n'} \in L$ linear unabhängige Vektoren. Dann ist $M := \pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(L)$ ein $(n - n')$-dimensionales Gitter.

Beweis. Es wird gezeigt, daß M eine diskrete Untergruppe von E ist. Daß M eine Untergruppe von E ist, ist offensichtlich. Angenommen $\boldsymbol{x} \in E$ ist eine Häufungspunkt von M. Betrachte die Folge $(\pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(\boldsymbol{v}_i))_{i \in \mathbb{N}}$ von paarweise verschiedenen Vektoren von M, die gegen $\boldsymbol{x}$ konvergiert. Es gilt $\pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(\boldsymbol{v}_i) = \boldsymbol{v}_i - \sum_{j=1}^{n'} \frac{(\boldsymbol{v}_i, \boldsymbol{b}_j)}{(\boldsymbol{b}_j, \boldsymbol{b}_j)} \boldsymbol{b}_j$. Betrachte die Folge $\boldsymbol{w}_i := \boldsymbol{v}_i - \sum_{j=1}^{n'} \lfloor \frac{(\boldsymbol{v}_i, \boldsymbol{b}_j)}{(\boldsymbol{b}_j, \boldsymbol{b}_j)} \rfloor \boldsymbol{b}_j$, die aus paarweise verschiedenen Vektoren von L besteht, da $\pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(\boldsymbol{v}_i) = (\pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(\boldsymbol{w}_i)$ gilt. Für alle $i \in \mathbb{N}$ besitzt die Ungleichung $\|\boldsymbol{w}_i - \pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(\boldsymbol{v}_i)\| \leq \sum_{j=1}^{n'} \|\boldsymbol{b}_j\|$ Gültigkeit, so daß die Folge $(\boldsymbol{w}_i - \pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(\boldsymbol{v}_i))_{i \in \mathbb{N}}$ beschränkt ist. Nach dem Satz von BOLZANO und WEIERSTRASS besitzt sie eine konvergente Teilfolge $(\boldsymbol{w}_{i_j} - \pi_{(\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_{n'})^\perp}(\boldsymbol{v}_{i_j}))_{j \in \mathbb{N}}$. Insbesondere ist die Folge $(\boldsymbol{w}_{i_j})_{j \in \mathbb{N}}$ konvergent. Dies steht im Widerspruch zur Diskretheit von L.
Aus der Dimensionsformel für lineare Abbildungen folgt, daß L' die Dimension $n - n'$ besitzt. $\diamond$

Es stellt sich bei einem Gitter $L \subseteq E$ und einem Gittervektor $v \in L$ die Frage, ob v zu einer Basis von L ergänzt werden kann. Ein Gittervektor $\boldsymbol{v} \in L$ läßt sich offensichtlich nur dann zu einer Basis von L ergänzen, wenn für alle $\boldsymbol{w} \in L$, $\alpha \in \mathbb{R}$ mit $\boldsymbol{v} = \alpha \boldsymbol{w}$ die Bedingung $\alpha = \pm 1$ gilt. Also muß $\boldsymbol{v}$ in Richtung $\mathbb{R}\boldsymbol{v}$ der kürzeste nicht-triviale Vektor von L sein. Diese Eigenschaft nennt man Primitivität:

Definition 4.1.5. Es sei $L \subseteq E$ ein Gitter. Ein Gittervektor $\boldsymbol{v} \in L$ heißt primitiv, wenn $L \cap \mathbb{R}\boldsymbol{v} = \mathbb{Z}\boldsymbol{v}$ gilt.

Um bei einem gegebenen Gitter zu testen, ob ein Gittervektor primitiv ist, stellt man ihn als Linearkombination einer Basis des Gitters dar und überprüft, ob der größte gemeinsame Teiler aller Koeffizienten der Linearkombination Eins ist. Es gilt sogar, daß man einen Gittervektor dann und nur dann zu einer Basis ergänzen kann, wenn er primitiv ist. Wie man eine Basis finden kann, die einen vorgegebenen primitiven Vektor enthält, ist Inhalt der nachfolgenden Proposition.

Proposition 4.1.6. Es sei $L \subseteq E$ ein d-dimensionales Gitter und sei $\boldsymbol{b}_1 \in L$ ein primitiver Vektor. Dann ist die Menge $L' := \pi_{(\mathbb{R}\boldsymbol{b}_1)^\perp}(L)$ ein $(d-1)$-dimensionales Gitter. Wenn für $\boldsymbol{b}_2, \ldots, \boldsymbol{b}_d \in L$ gilt, daß $\pi_{(\mathbb{R}\boldsymbol{b}_1)^\perp}(\boldsymbol{b}_2), \ldots, \pi_{(\mathbb{R}\boldsymbol{b}_1)^\perp}(\boldsymbol{b}_d)$ eine Basis von L' ist, dann ist $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ eine Basis von L.

Beweis. Nach Proposition 4.1.4 ist die Menge L' ein Gitter. Für die zweite Behauptung genügt es zu zeigen, daß die Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ das Gitter L erzeugen. Es sei $\boldsymbol{v} \in L$. Es gibt $\alpha_i \in \mathbb{Z}$, $i = 2, \ldots, d$ mit

$$\pi_{(\mathbb{R}\boldsymbol{b}_1)^\perp}(\boldsymbol{v}) = \sum_{i=2}^{d} \alpha_i \pi_{(\mathbb{R}\boldsymbol{b}_1)^\perp}(\boldsymbol{b}_i) = \pi_{(\mathbb{R}\boldsymbol{b}_1)^\perp}\left(\sum_{i=2}^{d} \alpha_i \boldsymbol{b}_i\right).$$

Es folgt aufgrund der Primitivität von $\boldsymbol{b}_1$, daß $\boldsymbol{v} - \sum_{i=2}^{d} \alpha_i \boldsymbol{b}_i \in L \cap \operatorname{Kern} \pi_{(\mathbb{R}\boldsymbol{b}_1)^\perp} = L \cap \mathbb{R}\boldsymbol{b}_1 = \mathbb{Z}\boldsymbol{b}_1$ ist, also $\boldsymbol{v} \in \mathbb{Z}\boldsymbol{b}_1 + \cdots + \mathbb{Z}\boldsymbol{b}_d$. ◇

4.1.2 Geometrische Invarianten von Gittern

In diesem Abschnitt studieren wir geometrische Eigenschaften von Gittern, die nicht von einer speziellen Basiswahl abhängen. Solche Eigenschaften werden als geometrische Invarianten bezeichnet.
Die wichtigste geometrische Invariante eines Gitters $L \subseteq E$ sind die kürzesten nicht-trivialen Vektoren von L, die Minimalvektoren von L. Wenn L durch eine Basis gegeben ist, ist es ein schwieriges Problem, die Minimalvektoren von L zu finden. Mit diesem Problem werden wir uns ausgiebig in den nächsten Kapiteln auseinandersetzen.

Proposition 4.1.7. Es sei $L \subseteq E$ ein d-dimensionales Gitter und $\mu \in \mathbb{R}$ eine reelle Konstante, dann gibt es nur endlich viele Gittervektoren $\boldsymbol{v} \in L$ mit $\|\boldsymbol{v}\| \leq \mu$.

Beweis. Der Beweis ist äußerst einfach: In der Kugel $\overline{B}(\mathbf{0}, \mu) = \{\boldsymbol{y} \in E : d(\boldsymbol{y}, \mathbf{0}) \leq \mu\}$ können nur endlich viele Gittervektoren liegen, da ansonsten ein Häufungspunkt existiert. ◇

Einen konstruktiven Beweis für die Tatsache, daß ein Gitter nur endlich viele Vektoren unterhalb einer vorgegebenen Längenschranke besitzt, werden wir im Beweis von Proposition 5.1.2 kennenlernen.

Definition 4.1.8. Es sei $L \subseteq E$ ein Gitter. Die Norm eines kürzesten Vektors von $L \backslash \{\mathbf{0}\}$ heißt Minimum von L, $\min L := \min\{\|\boldsymbol{v}\| : \boldsymbol{v} \in L \backslash \{\mathbf{0}\}\}$. Die Gittervektoren von L, die das Minimum von L realisieren, heißen Minimalvektoren von L, $\operatorname{Min} L := \{\boldsymbol{v} \in L : \|\boldsymbol{v}\| = \min L\}$.

Neben den Minimalvektoren eines Gitters sind die Gittervektoren interessant, die einerseits möglichst kurz sind und andererseits eine Basis ergeben.

Definition 4.1.9. Es sei $L \subseteq E$ ein n-dimensionales Gitter. Unter der Basislänge einer Basis von L versteht man die Norm des längsten Basisvektors. Unter der minimalen Basislänge von L versteht man das Minimum der Basislängen aller Basen von L (Notation: $\operatorname{bl}(L)$).

Definition 4.1.10. Es sei $L \subseteq E$ ein d-dimensionales Gitter. Eine Menge $F \subseteq E$ heißt ein Fundamentalbereich von L, wenn F meßbar[1] ist, $E = \bigcup_{\boldsymbol{v} \in L} (\boldsymbol{v} + F)$ und für alle $\boldsymbol{v} \in L \backslash \{\mathbf{0}\}$ die Gleichung $\operatorname{vol}(F \cap (\boldsymbol{v} + F)) = 0$ gilt.

Es sei $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq E$ ein Gitter, dann ist die Menge $F = \{\sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i : \alpha_i \in [0, 1]\}$ ein Fundamentalbereich von L. Alle Fundamentalbereiche von L besitzen das gleiche Volumen, das, wenn L durch eine Basis gegeben ist, effizient berechenbar ist, wie wir im folgenden sehen werden.

Definition 4.1.11. Es sei $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq E$ ein d-dimensionales Gitter und es sei $G_{(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)} = ((\boldsymbol{b}_i, \boldsymbol{b}_j))_{1 \leq i,j \leq d}$ die GRAM-Matrix von L bzgl. $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$. Die Determinante von L wird definiert als die Determinante einer GRAM-Matrix: $\det L := \det G_{(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)}$.

Da die Determinante eines Gitters von der Basiswahl unabhängig ist (zwei Basen unterscheiden sich nur durch eine unimodulare, ganzzahlige Transformation), ist sie eine geometrische Invariante des Gitters. Sie ist das Quadrat des Volumens eines Fundamentalbereichs des Gitters.

[1] Eine Menge $M \subseteq E$ heißt meßbar, wenn ihre charakteristische Funktion $1_M(x) = \begin{cases} 1, & \text{falls } \boldsymbol{x} \in M, \\ 0, & sonst. \end{cases}$, $\boldsymbol{x} \in E$, LEBESGUE-integrierbar ist. Jedoch genügt uns hier schon ein intuitiver Volumenbegriff.

4.1.3 Untergitter und Dualität von Gittern

Bei der Untersuchung von algebraischen Strukturen ist es immer interessant, Unterstrukturen zu finden, z.B. ist es interessant, welche Untergruppen eine Gruppe besitzt. Dieses algebraische Meta-Konzept beschreiben wir nun für Gitter.

Definition 4.1.12. Es sei $L \subseteq E$ ein Gitter. Eine Teilmenge M von L heißt Untergitter von L, wenn M selbst ein Gitter ist.

Wenn zwei Gitter $M \subseteq L \subseteq E$ gegeben sind, ist $(M, +)$ eine Untergruppe von $(L, +)$ und es ist sinnvoll, den Index von M in L, d.h. die Kardinalität der Faktorgruppe L/M zu betrachten.

Proposition 4.1.13. Es seien $M \subseteq L \subseteq E$ Gitter der gleichen Dimension n. Wenn $A \in \mathbb{Z}^{n\times n}$ die Matrix eines Basiswechsels einer Basis von L zu einer von M ist, dann gilt

$$[L : M] = |L/M| = \det A.$$

Korollar 4.1.14. (*Determinanten-Index-Formel*)
Es seien $M, L \subseteq E$ Gitter der gleichen Dimension und M sei ein Untergitter von L, dann gilt

$$\det M = [L : M]^2 \det L.$$

Eine Operation, die bei vielen mathematischen Objekten im unterschiedlichsten Kontext angewendet werden kann, ist das Dualisieren. Dies ist auch bei Gittern möglich, wobei das Dualisieren mit Hilfe des Skalarproduktes geschieht.

Proposition 4.1.15. Es sei $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq E$ ein d-dimensionales Gitter. Die Menge $L^\# := \{\boldsymbol{x} \in E : (\boldsymbol{x}, \boldsymbol{v}) \in \mathbb{Z} \text{ für alle } \boldsymbol{v} \in L\}$ ist ebenfalls ein d-dimensionales Gitter. Die Vektoren $\boldsymbol{b}_i^\# \in L^\#$ mit der Eigenschaft $(\boldsymbol{b}_i^\#, \boldsymbol{b}_j) = \delta_{ij}$, $1 \le i, j \le d$, bilden eine Basis von $L^\#$. Außerdem gilt $\det L^\# = (\det L)^{-1}$ und $(L^\#)^\# = L$.

Definition 4.1.16. Es sei $L \subseteq E$ ein d-dimensionales Gitter. Dann heißt $L^\#$ das zu L duale Gitter.

4.2 Sukzessive Minima und Reduktionstheorie von Gittern

Im letzten Abschnitt haben wir das Minimum, die Basislänge und die Determinante eines Gitters definiert. Dort haben wir gesehen, daß diese Begriffe geometrische Invarianten eines Gitters beschreiben. Darüber hinaus werden wir in diesem Abschnitt zeigen, daß sie in einer engen Beziehung zueinander stehen. Die Aufgabe, bei einem gegebenen Gitter eine möglichst kurze bzw. gute Basis zu finden, ist eine der Hauptaufgaben der Reduktionstheorie von Gittern. Was eine kurze bzw. gute Basis ist, ist nicht eindeutig definierbar. Wir begnügen uns hier zunächst mit einem Zitat von COHEN [Coh93] „Among all the $\mathbb{Z}$ bases of a lattice L, some are better than others. The ones whose elements are the shortest are called *reduced*.“

4.2.1 Die Gitterpunktsätze von MINKOWSKI

Dreh- und Angelpunkt der Reduktionstheorie von Gittern sind die Gitterpunktsätze von HERMANN MINKOWSKI (1864–1909), dem Begründer der Geometrie der Zahlen. In seiner Gedächtnisrede zu MINKOWSKI wird der Beweis des sogenannten MINKOWSKIschen Gitterpunktsatzes von HILBERT besonders gelobt: „Dieser Beweis eines tiefliegenden zahlentheoretischen Satzes ohne rechnerische Hilfsmittel wesentlich auf Grund einer geometrischen anschaulichen Betrachtung ist eine Perle MINKOWSKIscher Erfindungskunst ... “. Diese Perle wollen wir uns nicht entgehen lassen.

Theorem 4.2.1. (MINKOWSKI*scher Gitterpunktsatz, erster Hauptsatz von* MINKOWSKI)
Es sei $L \subseteq E$ ein d-dimensionales Gitter und es sei $K \subseteq E$ eine konvexe, zentralsymmetrische ($K = -K$) Menge. Wenn die Ungleichung $\operatorname{vol} K > 2^d\sqrt{\det L}$ erfüllt ist, so enthält K wenigstens zwei Gittervektoren von L.

Beweis. Es sei F ein Fundamentalbereich von L. Falls für alle $\boldsymbol{v} \in L\backslash\{\mathbf{0}\}$ die Mengen K und $\boldsymbol{v}+K$ disjunkt sind, folgt $\operatorname{vol} K \leq \operatorname{vol} F = \sqrt{\det L}$: Es sei K so in Teilmengen $K = \bigcup K_i$ zerlegt, daß für irgendwelche $\boldsymbol{v}_i \in L$ gilt $\boldsymbol{v}_i + K_i \subseteq F$. Nach Voraussetzung sind je zwei verschiedene Teilmengen disjunkt, so daß sich für das Volumen von K ergibt $\operatorname{vol} K = \operatorname{vol}(\bigcup K_i) = \operatorname{vol}(\bigcup(\boldsymbol{v}_i + K_i)) \leq \operatorname{vol} F$.
Es ist $\operatorname{vol} K > 2^d\sqrt{\det L}$, also $\operatorname{vol} \frac{1}{2}K > \det L$. Somit gibt es ein $\boldsymbol{v} \in L\backslash\{\mathbf{0}\}$ mit $(\boldsymbol{v} + \frac{1}{2}K) \cap \frac{1}{2}K \neq \emptyset$, d.h. es gibt $\frac{1}{2}\boldsymbol{x}, \frac{1}{2}\boldsymbol{y} \in \frac{1}{2}K$ mit $\boldsymbol{v} = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{y})$. Da K zentralsymmetrisch ist, liegt mit $\boldsymbol{y}$ auch $-\boldsymbol{y}$ in K. Da sich $\boldsymbol{v}$ als konvexe Linearkombination von $\boldsymbol{x}$ und $-\boldsymbol{y}$ schreiben läßt, muß aufgrund der Konvexität von K auch $\boldsymbol{v}$ in K liegen. ◇

Aus dem MINKOWSKIschen Gitterpunktsatz folgt als Korollar eine obere Schranke für das Minimum eines Gitters.

Korollar 4.2.2. Es sei $L \subseteq E$ ein d-dimensionales Gitter. Es gibt einen Gittervektor $\boldsymbol{v} \in L\backslash\{\mathbf{0}\}$, der nicht länger als $\sqrt{d}(\sqrt{\det L})^{1/d}$ ist, d.h. es gilt $\min L \leq \sqrt{d}(\sqrt{\det L})^{\frac{1}{d}}$.

Beweis. Setze $r := \sqrt{d}(\sqrt{\det L})^{1/d}$. Wir zeigen, daß die Kugel $\overline{B}(\mathbf{0}, r) = \{\boldsymbol{x} \in \mathbb{R}^d : \|x\| \leq r\}$ mindestens das Volumen $2^d\sqrt{\det L}$ besitzt. Dann folgt die Behauptung mit Theorem 4.2.1. Es gilt

$$\operatorname{vol}\overline{B}(\mathbf{0}, r) = r^d \operatorname{vol}\overline{B}(\mathbf{0}, 1) = r^d \frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)} = d^{d/2}\sqrt{\det L}\frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)}.$$

Falls d gerade ist, gilt $\Gamma(\frac{d}{2}+1) = \frac{2}{2}\cdot\frac{4}{2}\cdots\frac{d}{2} \leq \frac{1}{2^{d/2}}d^{d/2}$ und weiter

$$d^{d/2}\frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)} \geq d^{d/2}\frac{\pi^{d/2}}{\frac{1}{2^{d/2}}d^{d/2}} = (2\pi)^{d/2} \geq 2^d.$$

Falls d ungerade ist, gilt $\Gamma(\frac{d}{2}+1) = \frac{1}{2}\cdot\frac{3}{2}\cdots\frac{d}{2}\sqrt{\pi} \leq \frac{1}{2^{(d+1)/2}}d^{d/2}\sqrt{\pi}$ und weiter

$$d^{d/2}\frac{\pi^{d/2}}{\Gamma(\frac{d}{2}+1)} \geq d^{d/2}\frac{\pi^{d/2}}{\frac{1}{2^{(d+1)/2}}d^{d/2}\sqrt{\pi}} = \sqrt{\frac{2}{\pi}}(2\pi)^{d/2} \geq 2^d.$$

◇

Definition 4.2.3. Es sei $L \subseteq E$ ein n-dimensionales Gitter. Für $m \in \{1, \ldots, n\}$ ist das m-te sukzessive Minimum von L definiert als der Radius der kleinsten Kugel, die m linear unabhängige Gittervektoren enthält, d.h.

$$\lambda_m(L) := \min\{\mu \in \mathbb{R} : \exists \boldsymbol{v}_1, \ldots, \boldsymbol{v}_m \in L \text{ linear unabhängig mit } \|\boldsymbol{v}_i\| \leq \mu, i = 1, \ldots, m\}.$$

Es sei $L \subseteq E$ ein n-dimensionales Gitter. Obwohl der Gedanke reizvoll ist, gilt die Gleichung $\lambda_n(L) = \operatorname{bl}(L)$ im allgemeinen nicht. Das in der Dimension minimale Gegenbeispiel für diese Gleichung ist das Gitter $\mathsf{D}_5^{\#} = \mathbb{Z}^5 + \mathbb{Z}(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})^t$, denn es gilt einerseits $\operatorname{Min}\mathsf{D}_5^{\#} = \{\pm e_1, \ldots, \pm e_5\}$, wobei e_i der i-te Einheitsvektor ist, so daß $\lambda_5(\mathsf{D}_5^{\#}) = 1$ gilt, andererseits erzeugen die Minimalvektoren das Gitter nicht, d.h. es ist $\operatorname{bl}(\mathsf{D}_5^{\#}) > 1$.

Eines der Haupttheoreme der Geometrie der Zahlen ist die Abschätzung der sukzessiven Minima von MINKOWSKI. Dieses Theorem enthält den Gitterpunktsatz als Spezialfall.

Theorem 4.2.4. (*Zweiter Hauptsatz von* MINKOWSKI)
Es sei $L \subseteq E$ ein d-dimensionales Gitter mit den sukzessiven Minima $\lambda_1(L), \ldots, \lambda_d(L)$. Dann gelten die Ungleichungen

$$\frac{2^d}{d!}\sqrt{\det L} \leq \lambda_1(L) \cdots \lambda_d(L) \cdot \operatorname{vol} \overline{B}(\mathbf{0}, 1) \leq 2^d \sqrt{\det L}.$$

Überdies gibt es einen Zusammenhang zwischen den sukzessiven Minima eines Gitters und denen des zu diesem Gitter dualen Gitters.

Theorem 4.2.5. (BANASZCZYK [Ban93])
Es sei $L \subseteq E$ ein d-dimensionales Gitter. Dann gilt für alle $i \in \{1, \ldots, d\}$ die Ungleichung

$$1 \leq \lambda_i(L)\lambda_{d-i+1}(L^{\#}) \leq d.$$

4.2.2 Gitterbasisreduktion im Sinne von KORKINE und ZOLOTAREV

Wie wir in Proposition 4.1.4 gesehen haben, sind orthogonale Projektionen von Gittern auf orthogonale Komplemente von Vektorraumerzeugnissen von Untergittern ebenfalls Gitter. Orthogonale Projektionen beherrscht man am besten, wenn man Orthogonalbasen von den entsprechenden Untervektorräumen kennt. Diese lassen sich effizient mit Hilfe der aus der Linearen Algebra bekannten GRAM-SCHMIDT-Orthogonalisierung berechnen, die hier noch einmal ins Gedächtnis gerufen wird.

Proposition 4.2.6. (GRAM-SCHMIDT-*Orthogonalisierung*)
Es seien $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in E$ linear unabhängige Vektoren. Definiere induktiv

$$\boldsymbol{b}_i^* := \boldsymbol{b}_i - \sum_{j=1}^{i-1} \mu_{ij}\boldsymbol{b}_j^* \quad \text{mit} \quad \mu_{ij} := \frac{(\boldsymbol{b}_i, \boldsymbol{b}_j^*)}{(\boldsymbol{b}_j^*, \boldsymbol{b}_j^*)}, \quad 1 \leq j < i \leq n.$$

Dann bilden $\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_n^*$ eine Orthogonalbasis von $\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_n$. Sie besitzt die folgenden Eigenschaften:

i) Für $i \in \{1, \ldots, n\}$ gilt $\mathbb{R}\boldsymbol{b}_1 + \cdots + \mathbb{R}\boldsymbol{b}_i = \mathbb{R}\boldsymbol{b}_1^* + \cdots + \mathbb{R}\boldsymbol{b}_i^*$,

ii) für $i \in \{1, \ldots, n\}$ gilt $\pi_{(\mathbb{R}\boldsymbol{b}_1^* + \cdots + \mathbb{R}\boldsymbol{b}_{i-1}^*)^\perp}(\boldsymbol{b}_i) = \boldsymbol{b}_i^*$,

iii) die Transformationsmatrix der $\boldsymbol{b}_i$ auf die $\boldsymbol{b}_i^*$ besitzt Determinante Eins,

iv) es ist $\det((\boldsymbol{b}_i, \boldsymbol{b}_j)_{1 \leq i,j \leq n}) = \prod_{i=1}^n (\boldsymbol{b}_i^*, \boldsymbol{b}_i^*)$.

Eine Basis eines Gitters, deren Vektoren paarweise möglichst orthogonal zueinander sind, heißt längenreduziert:

Definition 4.2.7. Es sei $L \subseteq E$ ein n-dimensionales Gitter. Eine Basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ von L heißt längenreduziert, wenn für die zugehörigen GRAM-SCHMIDT-Koeffizienten $|\mu_{ij}| \leq \frac{1}{2}$, $1 \leq j < i \leq n$, gilt.

Wie schon in der Einleitung angedeutet wurde, ist nicht klar, wie eine „gute“ Gitterbasis aussieht. Dies hat zur Folge, daß es verschiedene Ansätze gibt, Gitterbasen als reduziert anzusehen. Wir werden den Reduktionsbegriff von KORKINE und ZOLOTAREV benutzen, da er zwei Vorteile besitzt:

i) Das Auffinden von kürzesten Gittervektoren ist ein Teilproblem der Gitterbasisreduktion.

ii) Es gibt einen offensichtlichen Algorithmus (siehe Proposition 4.1.6), der eine beliebige Basis eines gegebenen Gitters L in eine Basis von L transformiert, die im Sinne von KORKINE und ZOLOTAREV reduziert ist.

Ein entscheidender Nachteil ist, daß bislang nur Algorithmen zur Berechnung von KORKINE-ZOLOTAREV-reduzierten Gitterbasen bekannt sind, die eine in der Eingabelänge exponentielle Laufzeit besitzen, was höchstwahrscheinlich (siehe Kapitel 6) nicht verbessert werden kann. Jetzt aber endlich zur Definition des Reduktionsbegriffs von KORKINE und ZOLOTAREV:

Definition 4.2.8. Es sei $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq E$ ein d-dimensionales Gitter. Setze $L_i := \pi_{(\mathbb{R}\boldsymbol{b}_1+\cdots+\mathbb{R}\boldsymbol{b}_i)^\perp}(L)$. Die geordnete Basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ heißt reduziert im Sinne von KORKINE und ZOLOTAREV, falls die folgenden Bedingungen erfüllt sind:

i) Für alle $i \in \{1, \ldots, d\}$ gilt $\|\boldsymbol{b}_i^*\| = \min L_i$.

ii) Die Basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ ist längenreduziert.

Es sei $L \subseteq E$ ein d-dimensionales Gitter und $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ eine Basis von L, die im Sinne von KORKINE und ZOLOTAREV reduziert ist. Dann sind die Längen der Basisvektoren durch die entsprechenden sukzessiven Minima sowohl nach oben als auch nach unten beschränkt, wie die nachfolgende Proposition präzisiert.

Proposition 4.2.9. (LAGARIAS, LENSTRA, SCHNORR [LLS90])
Es sei $L \subseteq E$ ein d-dimensionales Gitter und $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ eine Basis von L, die im Sinne von KORKINE und ZOLOTAREV reduziert ist. Dann gilt die Ungleichung

$$\sqrt{\frac{4}{i+3}}\lambda_i(L) \leq \|\boldsymbol{b}_i\| \leq \sqrt{\frac{i+3}{4}}\lambda_i(L), \quad i = 1, \ldots, d.$$

4.3 Elementare Eigenschaften von konvexen Polytopen

Konvexe Polytope sind Grundobjekte der diskreten und kombinatorischen Geometrie, sie sind Verallgemeinerungen von zweidimensionalen Polygonen. Wir beschäftigen uns zunächst mit allgemeinen konvexen Polytopen und sammeln die wichtigsten Fakten. Anschließend wenden wir uns speziell den konvexen Parallelotopen zu, die besonders einfache konvexe Polytope sind.
In diesem Abschnitt werden nur Ergebnisse vorgestellt. Für Beweise, die zwar nicht schwierig, aber dennoch manchmal langwierig sind, sowie für einen tiefergehenden Einblick in die Theorie der konvexen Polytope sei auf das schöne Buch von ZIEGLER [Zie95] verwiesen.

4.3.1 Konvexe Polytope allgemein

Ein konvexes Polytop P im d-dimensionalen euklidischen Vektorraum $(E, (\cdot,\cdot))$ ist die konvexe Hülle einer endlichen Teilmenge $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ von E, d.h. ein konvexes Polytop ist eine Menge der Form

$$P = \operatorname{conv} X := \left\{ \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i : \alpha_i \in \mathbb{R}_{\geq 0}, \sum_{i=1}^{n} \alpha_i = 1 \right\}.$$

Genausogut können konvexe Polytope als beschränkte Lösungsmengen von endlich vielen linearen Ungleichungen beschrieben werden: Einerseits besitzt ein konvexes Polytop $P \subseteq E$ eine Darstellung der Form

$$P = \{\boldsymbol{x} \in E : (\boldsymbol{y}_i, \boldsymbol{x}) \leq \boldsymbol{b}_i, i = 1, \ldots, m\},\ \boldsymbol{y}_1, \ldots, \boldsymbol{y}_m \in E,\ \boldsymbol{b} \in \mathbb{R}^m,$$

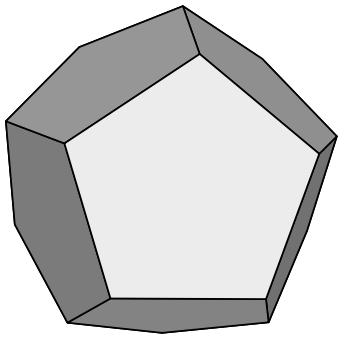

Abbildung 4.2: Der Dodekaeder: ein sehr regelmäßiges Polytop.

andererseits läßt sich jede solche beschränkte Lösungsmenge als konvexe Hülle von endlich vielen Punkten schreiben.
Das d-dimensionale Analogon zum zweidimensionalen Quadrat $C_2 := [0,1] \times [0,1]$ bzw. zum dreidimensionalen Würfel $C_3 := [0,1]^3$ ist der d-dimensionale Würfel $C_d := [0,1]^d$, der hier im wesentlichen das einzige relevante konvexe Polytop ist.

Als nächstes wollen wir das Konzept der Ecken, Kanten und Wände eines konvexen Polytops, das im $\mathbb{R}^3$ intuitiv einsichtig ist, in den euklidischen Vektorraum $(E, (\cdot,\cdot))$ übertragen.
Es sei A eine Teilmenge von E. Die affine Hülle von A ist definiert als die Menge $\operatorname{aff} A := \{\sum_{i=1}^n \alpha_i \boldsymbol{x}_i : x_i \in A,\ \alpha_i \in \mathbb{R},\ \sum_{i=1}^n \alpha_i = 1,\ n \in \mathbb{N}\}$, sie ist der kleinste affine Unterraum, der A enthält. Außerdem wird $\dim A := \dim \operatorname{aff} A$ definiert.
Eine k-dimensionale Seite F eines konvexen Polytops $P \subseteq E$ ist eine Teilmenge von P der folgenden Form

$$F = P \cap \{\boldsymbol{x} \in E : (\boldsymbol{x}, \boldsymbol{v}) \leq b\},\ \boldsymbol{v} \in E,\ b \in \mathbb{R},$$

wobei $(\boldsymbol{x}, \boldsymbol{v}) \leq b$ eine lineare Ungleichung ist, die für alle $\boldsymbol{x} \in P$ erfüllt ist, und wobei k die Dimension der affinen Hülle von F ist.
Offensichtlich ist jede Seite von P ebenfalls ein konvexes Polytop, insbesondere sind P selbst und die leere Menge, deren Dimension mit -1 definiert wird, Seiten von P. Eine Seite von P, die weder die leere Menge noch P selbst ist, heißt eigentliche Seite von P. Die nulldimensionalen Seiten von P werden als Ecken, die eindimensionalen Seiten von P werden als Kanten, und die $(\dim P - 1)$-dimensionalen Seiten von P werden als Wände von P bezeichnet.
Der d-dimensionale Würfel C_d besitzt 2^d Ecken, $d \cdot 2^{d-1}$ Kanten und $2d$ Wände.
Die durch die Inklusion halbgeordnete Menge $\mathcal{F}(P)$ der Seiten eines konvexen Polytops P besitzt eine besondere kombinatorische Struktur, die in der nachfolgenden Proposition beschrieben wird.

Proposition 4.3.1. (*Der Seitenverband eines konvexen Polytops*)
Es sei $P \subseteq E$ ein konvexes Polytop. Die Halbordnung $(\mathcal{F}(P), \subseteq)$ besitzt die folgenden Eigenschaften.

i) Jede maximale Kette $\emptyset = F_{-1} \subset F_0 \subset F_1 \subset \ldots \subset F_{\dim P} = P$ besitzt dieselbe Länge $\dim P$.

ii) Sie ist ein Verband, d.h. je zwei Seiten $F, G \in \mathcal{F}(P)$ besitzen ein Infimum und ein Supremum.

iii) Sie besitzt ein kleinstes und ein größtes Element.

iv) Der Verband ist atomar, d.h. jede eigentliche Seite ist Supremum einer geeigneten Menge von Ecken.

v) Der Verband ist coatomar, d.h. jede eigentliche Seite ist Infimum einer geeigneten Menge von Wänden.

Zwei konvexe Polytope, deren Seitenverbände bis auf Isomorphie übereinstimmen, d.h. es gibt eine Bijektion zwischen den Seiten der beiden konvexen Polytope, die die Inklusion respektiert, heißen kombinatorisch äquivalent. Zwei konvexe Polytope, die durch eine affine Abbildung ineinander überführt werden können, heißen affin äquivalent. Offensichtlich sind zwei konvexe Polytope, die affin äquivalent sind, auch kombinatorisch äquivalent.
Zwei konvexe Polytope P und Q, deren Seitenverbände bis auf eine Anti-Isomorphie übereinstimmen, d.h. es gibt eine Bijektion $\Phi : \mathcal{F}(P) \to \mathcal{F}(Q)$ mit der Eigenschaft, daß für Seiten $F, G \in \mathcal{F}(P)$ mit $F \subseteq G$ gilt $\Phi(F) \supseteq \Phi(G)$, heißen kombinatorisch dual. Zu einem Polytop P gibt es immer ein zu P kombinatorisch duales Polytop, z.B. ist das zu P polare Polytop $P^\vee := \{\boldsymbol{y} \in E : (\boldsymbol{x}, \boldsymbol{y}) \leq 1 \text{ für alle } \boldsymbol{x} \in P\}$ zu P kombinatorisch dual. Der Oktaeder $\operatorname{conv}\{\pm\boldsymbol{e}_1, \ldots, \pm\boldsymbol{e}_d\}$ ist zu dem Würfel C_d kombinatorisch dual.

4.3.2 Konvexe Parallelotope speziell

Konvexe Parallelotope (Paralleleppipede) sind besonders einfache konvexe Polytope. Ein d-dimensionales konvexes Parallelotop ist das Bild des d-dimensionalen Würfels C_d unter einer bijektiven, affinen Abbildung.

Definition 4.3.2. Es seien $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_d \in E$ linear unabhängige Vektoren und $\boldsymbol{b} \in E$, dann ist die Menge

$$P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) := \boldsymbol{b} + \left\{ \sum_{i=1}^{d} \alpha_i \boldsymbol{x}_i : \alpha_i \in [0,1] \right\}$$

affin äquivalent zu dem d-dimensionalen Würfel C_d. Die Menge $P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$ heißt das von $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_d$ aufgespannte konvexe Parallelotop, das um den Vektor $\boldsymbol{b}$ verschoben ist.

Die minimale Breite eines konvexen Parallelotops $P = P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$ (Notation: $\operatorname{width} P$) ist der maximale Durchmesser einer Kugel, die vollständig in P enthalten ist. Dies kann auch so formuliert werden, daß eine effiziente Berechnungsmethode für die Breite eines konvexen Parallelotops direkt ablesbar ist: Die minimale Breite von P ist der minimale Abstand von $\boldsymbol{x}_i$ zur Hyperebene $\mathbb{R}\boldsymbol{x}_1 + \cdots + \mathbb{R}\boldsymbol{x}_{i-1} + \mathbb{R}\boldsymbol{x}_{i+1} + \cdots + \mathbb{R}\boldsymbol{x}_d$, $i = 1, \ldots, d$.

Algorithmus 4.3.3 Berechnung der minimalen Breite eines konvexen Parallelotops

Eingabe: $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_d \in E$ linear unabhängige Vektoren und $\boldsymbol{b} \in E$.
Ausgabe: $w = \operatorname{width} P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$.
Berechne mit der GRAM-SCHMIDT-Orthogonalisierung $(\boldsymbol{x}_1^*, \ldots, \boldsymbol{x}_d^*)$.
$w \leftarrow \infty$.
for $i = 1, \ldots, d$ **do**
  $\boldsymbol{x} \leftarrow \boldsymbol{x}_i - \sum_{j=1, j\neq i}^{d} \frac{(\boldsymbol{x}_i, \boldsymbol{x}_j^*)}{(\boldsymbol{x}_j^*, \boldsymbol{x}_j^*)} \boldsymbol{x}_j^*$.
  if $w > \|\boldsymbol{x}\|$ **then**
    $w \leftarrow \|\boldsymbol{x}\|$.
  end if
end for

Eine wichtige Operation für konvexe Parallelotope ist das Skalieren um einen Faktor vom Mittelpunkt des gegebenen konvexen Parallelotops aus, wofür eine Notation eingeführt wird.

Definition 4.3.4. Es sei $\alpha \in \mathbb{R}_{>0}$ der Faktor, um den das konvexe Parallelotop $P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$ skaliert werden soll. Schreibe

$$\alpha \bullet P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) := \boldsymbol{b} + \frac{1}{2}\sum_{i=1}^{d} \boldsymbol{x}_i + \alpha \left(P(\boldsymbol{0}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) - \frac{1}{2}\sum_{i=1}^{d} \boldsymbol{x}_i \right).$$

Wenn ein konvexes Parallelotop skaliert wird, ändert sich seine minimale Breite. Die nachfolgende Proposition zeigt, daß der Zusammenhang zwischen dem Skalierungsfaktor und der Breite so ist, wie man es erwartet. Außerdem werden dort offensichtliche Eigenschaften des Skalierungsoperators gesammelt.

Proposition 4.3.5. Es seien $\alpha, \beta \in \mathbb{R}_{>0}$. Dann gilt

i) $1 \bullet P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) = P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$.

ii) $\alpha \bullet (\beta \bullet P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)) = (\alpha\beta) \bullet P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$.

iii) Falls $\alpha \in \mathbb{R}_{\geq 1}$: $P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) \subseteq \alpha \bullet P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$.

iv) $\operatorname{width}(\alpha \bullet P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)) = \alpha \operatorname{width} P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$.

Es sei $L \subseteq E$ ein d-dimensionales Gitter, das durch die Angabe der Gitterbasis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ gegeben ist. Jeder Vektor $\boldsymbol{x} \in E$ läßt sich so durch einen Gittervektor verschieben, so daß das Translat $\boldsymbol{x}'$ im Fundamentalbereich $P(\boldsymbol{0}; \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ liegt. Wenn wir die Menge $P^-(\boldsymbol{0}; \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) := \boldsymbol{0} + \left\{ \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i : \alpha_i \in [0,1) \right\}$ betrachten, ist dieser Gittervektor sogar eindeutig.
Diese Operation nennen wir Reduktion von $\boldsymbol{x}$ modulo der Gitterbasis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ (Notation: $\boldsymbol{x}' = \boldsymbol{x} \mod (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$). Falls die Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ und $\boldsymbol{x}$ durch eine endliche Eingabe codiert werden können, ist Algorithmus 4.3.6 ein effizienter deterministischer Algorithmus zur Berechnung von Reduktionen modulo einer Gitterbasis.

Algorithmus 4.3.6 Reduktion modulo einer Gitterbasis

Eingabe: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in E$ linear unabhängige Vektoren und $\boldsymbol{x} \in E$.
Ausgabe: $\boldsymbol{x}' = \boldsymbol{x} \mod (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$.
Bestimme die eindeutige Lösung $\boldsymbol{\alpha}$ des linearen Gleichungssystems $\boldsymbol{x} = \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i$.
$\boldsymbol{x}' \leftarrow \sum_{i=1}^{d} (\alpha_i - \lfloor \alpha_i \rfloor) \boldsymbol{b}_i$.

Im übrigen gelten alle Aussagen, die wir schon für das konvexe Parallelotop $P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$ getroffen haben mutas mutandis für $P^-(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) = \boldsymbol{b} + \{\sum_{i=1}^{d} \alpha_i \boldsymbol{x}_i : \alpha_i \in [0,1)\}$.

Kapitel 5

Gitterprobleme

Die wichtigsten algorithmischen Probleme der Geometrie der Zahlen sind das „shortest vector problem“ (SVP) und das „closest vector problem“ (CVP). Gegeben sei ein Gitter L im normierten Vektorraum $\mathbb{Q}^d$. Das SVP besteht darin, einen kürzesten nicht-trivialen Gittervektor von L zu finden. Es sei zusätzlich ein Vektor $\boldsymbol{x} \in \mathbb{Q}^d$ gegeben. Das CVP besteht darin, einen Gittervektor von L zu finden, der $\boldsymbol{x}$ am nächsten liegt. Für beide Probleme sind keine effizienten Algorithmen bekannt. CVP ist $\mathcal{NP}$-hart, und es ist ein offenes Problem, ob SVP ebenfalls $\mathcal{NP}$-hart ist. In Kapitel 6 werden wir die $\mathcal{NP}$-Härte von SVP unter Annahme einer plausiblen zahlentheoretischen Vermutung beweisen.

In den letzten Jahren wurden die Gitterprobleme SVP und CVP wegen ihrer vielfältigen Anwendungen ausgiebig studiert. So konnten mit Hilfe des effizienten LLL-Algorithmus ([LLL82]), der Gitterbasen reduziert und gleichzeitig eine Approximation für SVP berechnet, mehrere neuartige effiziente Algorithmen gefunden werden. Als Beispiele seien hier nur der Entwurf eines effizienten Algorithmus für ganzzahlige Optimierungsaufgaben mit einer festen Anzahl von Variablen (siehe [GLS88]), der Entwurf eines Algorithmus zur Faktorisierung von Polynomen über algebraischen Zahlkörpern und über endlichen Körpern (siehe [Coh93]) sowie SHAMIRs erfolgreiche Kryptanalyse ([Sha84]) des Public-Key-Kryptosystems von MERKLE und HELLMAN, das auf der algorithmischen Schwierigkeit des Knapsack-Problems beruhen sollte, genannt.

Neben den Anwendungen des effizienten LLL-Algorithmus wurden Einsichten in die Komplexitätstheorie von SVP und CVP gewonnen. So zeigten ARORA, BABAI, STERN und SWEEDYK in [ABSS93] unter Anwendung des $\mathcal{PCP}$-Theorems, daß die Berechnung einer Approximation für CVP um jeden konstanten Faktor $\mathcal{NP}$-hart ist. DINER, KINDLER und SAFRA verschärften in [DKS98] dieses Ergebnis. Auf die Komplexitätstheorie von SVP werden wir in den nächsten beiden Kapiteln ausführlich eingehen.

In diesem Kapitel werden die Gitterprobleme SVP und CVP vorgestellt. Wir gehen auf die komplexitätstheoretische Beziehung zwischen den beiden Problemen ein und zeigen mit einem einfachen und eleganten Beweis von GOLDREICH, MICCIANCIO, SAFRA und SEIFERT ([GMSS99]), daß die Existenz eines effizienten Approximationsalgorithmus für CVP die Existenz eines effizienten Approximationsalgorithmus für SVP mit gleicher Worst-Case-Güte impliziert. SVP ist also nicht schwerer zu approximieren als CVP. Anschließend werden zwei weitere Gitterprobleme definiert. Für deren Lösung sind deterministische Algorithmen mit polynomieller Laufzeit bekannt.

5.1 Das „shortest vector problem“ (SVP)

Es kann behauptet werden, daß das „shortest vector problem“ schon seit zweihundert Jahren die besten Mathematiker und Informatiker beschäftigt. GAUSS[1] konnte einen effizienten Algorithmus für das „shortest vector problem“ für Gitter der Dimension 2 angeben. Später wurde das allgemeine „shortest vector problem“ von DIRICHLET, HERMITE, KORKINE und ZOLOTAREV studiert, wobei Abschätzungen der Form von Korollar 4.2.2 im Vordergrund des Interesses standen. MINKOWSKI stellte das Problem in das Zentrum seiner Theorie der Geometrie der Zahlen.

Der erste Durchbruch zur algorithmischen Lösung des „shortest vector problem“ kam spät. Der effiziente LLL-Algorithmus berechnet eine $2^{(d-1)/2}$-Approximation für kürzeste Vektoren in Gittern der Dimension d. Es hat sich jedoch empirisch herausgestellt, daß der LLL-Algorithmus für Gitter geringer Dimension (etwa in den Dimension $1 \leq d \leq 100$) deutlich bessere Approximationen liefert.

Wir definieren das „shortest vector problem“ für Gitter, die im Vektorraum $\mathbb{Q}^d$ liegen. Der $\mathbb{Q}^d$ wird mit dem euklidischen Standardskalarprodukt versehen, so daß die betrachtete Norm die wohlbekannte 2-Norm ist. Ein analoges Problem läßt sich bei jeder anderen Norm auf $\mathbb{Q}^d$ definieren. So hat VAN EMDE BOAS in [vEB81] nachgewiesen, daß das SVP bzgl. der Maximumsnorm $\mathcal{NP}$-hart ist, was für das SVP bzgl. der 2-Norm ein offenes Problem ist.

Definition 5.1.1. (*„shortest vector problem“* (SVP))
Gegeben sind d linear unabhängige Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$. Es sei $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ das von ihnen erzeugte Gitter.
Variante 1: Gibt es zu einer gegebenen Konstante $\mu \in \mathbb{R}_{>0}$ einen Gittervektor $\boldsymbol{v} \in L\backslash\{\boldsymbol{0}\}$ mit $\|\boldsymbol{v}\| \leq \mu$?
Variante 2: Berechne das Minimum von L.
Variante 3: Berechne einen Minimalvektor von L.

Offensichtlich ist Variante 1 eine Abschwächung von Variante 2 und Variante 2 eine Abschwächung von Variante 3. Umgekehrt gilt, daß wenn Variante 1 effizient lösbar ist, dann ist es auch Variante 2. Binäre Suche ermöglicht eine Reduktion von Variante 2 auf Variante 1. Es ist aber ein offenes Problem, ob aus der Existenz eines effizienten Algorithmus für Variante 2 die Existenz eines effizienten Algorithmus für Variante 3 folgt.

Das „shortest vector problem“ läßt sich als Minimierungsproblem entsprechend Definition 2.2.1 auffassen. Es ist dann $\mathsf{SVP} = (I, S, v, \min)$ mit

$$
\begin{aligned}
I &= \{(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \in \mathbb{Q}^{d\times d} : \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \text{ linear unabhängig}, d \in \mathbb{N}\},\\
S &: \begin{cases} I & \to \quad 2^{\mathbb{Q}^d} \\ (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) & \mapsto \quad \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d\backslash\{\boldsymbol{0}\}, \end{cases}\\
v &: \begin{cases} S(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) & \to \quad \mathbb{R}_{>0} \\ \boldsymbol{v} & \mapsto \quad \|\boldsymbol{v}\|. \end{cases}
\end{aligned}
$$

Im Gegensatz zu vielen Problemen der Klasse $\mathcal{NP}$ ist der Beweis dafür, daß die Entscheidungsvariante des SVP in der Klasse $\mathcal{NP}$ liegt, nicht völlig trivial.

Proposition 5.1.2. Die Entscheidungsvariante des SVP liegt in der Klasse $\mathcal{NP}$.

Beweis. Wir überzeugen uns zuerst davon, daß SVP überhaupt berechenbar ist, indem wir zeigen, daß für ein $\boldsymbol{v} = \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i \in L$ die Ungleichung $\|\boldsymbol{v}\| \leq \mu$ nur dann gelten kann, wenn jeder Koeffizient α_i, $i = 1, \ldots, d$, betragsmäßig durch einen Wert, der ausschließlich von $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$

[1] Manche Leute bezeichnen ihn als den größten deutschen Informatiker.

und μ abhängt, beschränkt ist. Es sei G die GRAM-Matrix von L bzgl. der Basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$. Die GRAM-Matrix von $L^\#$ bzgl. der Basis $(\boldsymbol{b}_1^\#, \ldots, \boldsymbol{b}_d^\#)$ ist G^{-1}. Für einen Gittervektor $\boldsymbol{v} = \sum_{i=1}^d \alpha_i \boldsymbol{b}_i$ von L mit $\|\boldsymbol{v}\| \leq \mu$ ergibt die Ungleichung von CAUCHY-SCHWARZ

$$|\alpha_i| = |(\boldsymbol{v}, \boldsymbol{b}_i^\#)| \leq (\boldsymbol{v}, \boldsymbol{v})(\boldsymbol{b}_i^\#, \boldsymbol{b}_i^\#) \leq g'_{ii}\mu, \quad i = 1, \ldots, d, \tag{5.1}$$

wobei g'_{ii} das i-te Diagonalelement von G^{-1} ist. Das Verfahren, das sich aus Ungleichung (5.1) ablesen läßt, besteht darin, sämtliche $\boldsymbol{\alpha} \in \mathbb{Z}^d$ mit $|\alpha_i| \leq g'_{ii}\mu$, $i = 1, \ldots, d$, auf die Bedingung $\boldsymbol{\alpha}^t G \boldsymbol{\alpha} \leq \sqrt{\mu}$ hin zu überprüfen. In der Menge $\{\boldsymbol{x} \in \mathbb{R}^d : |x_i| \leq g'_{ii}\mu, i = 1, \ldots, d\}$, die ein konvexes Parallelotop ist, sind exakt $\prod_{i=1}^d (2\lfloor g'_{ii}\mu \rfloor) - 1$ Gitterpunkte des Gitters $\mathbb{Z}^d$ enthalten, die auf die obige Bedingung hin zu überprüfen sind, so daß der hier angegebene Algorithmus eine in der Eingabelänge exponentielle Laufzeit besitzt.

Angenommen es gilt $\min L \leq \mu$, d.h. die Eingabe $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \mu)$ ist eine Ja-Instanz der Entscheidungsvariante des SVP. Aus Ungleichung (5.1) und der Tatsache, daß die Bitlängen der Einträge der Matrix G^{-1} polynomiell in den Bitlängen der Einträge von G beschränkt sind (dies ist ein Satz von EDMONDS, siehe z.B. [GLS88] für Details), folgt die Existenz des Zertifakts $\boldsymbol{\alpha}$ für die Tatsache, daß es einen Gittervektor $\boldsymbol{v} \in L \backslash \{\boldsymbol{0}\}$ mit $\|\boldsymbol{v}\| \leq \mu$ gibt, dessen Länge in der Bitlänge der Eingabe $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \mu)$ polynomiell beschränkt ist. Das Zertifikat ist außerdem effizient verifizierbar. ◇

Der im Beweis angegebene Algorithmus wurde von FINCKE und POHST in [FP85] so modifiziert, daß er polynomielle Laufzeit besitzt, wenn die Dimension und die Längenschranke fest gewählt sind. Der FINCKE-POHST-Algorithmus ist in Theorie und Praxis der effizienteste bekannte Algorithmus zur Berechnung sämtlicher Gittervektoren eines vorgegebenen d-dimensionalen Gitters $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$, die eine vorgegebene Längenschranke $\mu \in \mathbb{R}_{>0}$ nicht überschreiten.
Mit G sei die GRAM-Matrix von L bzgl. der Basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ bezeichnet. Die Aufgabe, alle $\boldsymbol{v} \in L$ mit $\|\boldsymbol{v}\| \leq \mu$ zu bestimmen, ist äquivalent zur Aufgabe, sämtliche $\boldsymbol{\alpha} \in \mathbb{Z}^d$ mit $\boldsymbol{\alpha}^t G \boldsymbol{\alpha} \leq \mu^2$ zu bestimmen, da für $\boldsymbol{v} = \sum_{i=1}^d \alpha_i \boldsymbol{b}_i$ gilt

$$\|\boldsymbol{v}\|^2 = (v, v) = \left(\sum_{i=1}^d \alpha_i \boldsymbol{b}_i, \sum_{i=1}^d \alpha_i \boldsymbol{b}_i \right) = \sum_{i=1}^d \sum_{j=1}^d \alpha_i \alpha_j (\boldsymbol{b}_i, \boldsymbol{b}_j) = \boldsymbol{\alpha}^t G \boldsymbol{\alpha}.$$

Die Menge $E = \{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{\alpha}^t G \boldsymbol{\alpha} \leq \mu^2\}$ ist ein Ellipsoid, das ein deutlich geringeres Volumen als das konvexe Parallelotop aus dem Beweis von Proposition 5.1.2 besitzt. Der FINCKE-POHST-Algorithmus ist ein Backtracking-Algorithmus, der sukzessiv sämtliche Vektoren der Menge $\mathbb{Z}^d \cap E$ bestimmt.

Es sei $G = R^t R$ die CHOLESKY-Zerlegung (siehe [GL96]) von G, wobei $R = (r_{ij}) \in \mathbb{R}^{d \times d}$ eine rechte obere Dreiecksmatrix ist, d.h. $r_{ij} = 0$ für $1 \leq j < i \leq d$. Es gilt für $\boldsymbol{\alpha} \in \mathbb{Z}^d$ die Gleichung $R\boldsymbol{\alpha} = \left(\sum_{i=1}^d r_{1i}\alpha_i, \sum_{i=2}^d r_{2i}\alpha_i, \ldots, r_{dd}\alpha_d \right)^t$ und weiter

$$\boldsymbol{\alpha}^t G \boldsymbol{\alpha} = (R\boldsymbol{\alpha})^t (R\boldsymbol{\alpha}) = \sum_{i=1}^d \left(r_{ii}\alpha_i + \sum_{j=i+1}^d r_{ij}\alpha_j \right)^2 = \sum_{i=1}^d r_{ii}^2 \left(\alpha_i + \sum_{j=i+1}^d \frac{r_{ij}}{r_{ii}} \alpha_j \right)^2.$$

Wenn man nacheinander $\alpha_d, \alpha_{d-1}, \ldots, \alpha_1$ wählt, ergeben sich obere Schranken für $|\alpha_i|$

$$r_{dd}^2 \alpha_d^2 \leq \mu \iff |\alpha_d| \leq \sqrt{\frac{\mu}{r_{dd}^2}}$$

und

$$r_{d-1,d-1}^2\left(\alpha_{d-1}+\frac{r_{d-1,d}}{r_{d-1,d-1}}\right)^2 \leq \mu - r_{dd}^2\alpha_d^2 \Longleftrightarrow |\alpha_{d-1}| \leq \sqrt{\frac{\mu - r_{dd}^2\alpha_d^2}{r_{d-1,d-1}^2}} - \frac{r_{d-1,d}}{r_{d-1,d-1}}\alpha_d,$$

und allgemein für $i \in \{1,\ldots,d\}$

$$|\alpha_i| \leq \sqrt{\frac{\mu - T_i}{r_{ii}^2}} - \sum_{j=i+1}^{d}\frac{r_{ij}}{r_{ii}}\alpha_j, \quad \text{mit} \quad T_i = \sum_{k=i+1}^{d} r_{kk}^2\left(\alpha_k + \sum_{j=k+1}^{d}\frac{r_{kj}}{r_{kk}}\alpha_j\right)^2.$$

Somit erhalten wir den nachfolgenden Algorithmus.

Algorithmus 5.1.3 Abzählen von kurzen Gittervektoren

Eingabe: $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_d \in \mathbb{Q}^d$ linear unabhängig und $\mu \in \mathbb{R}_{>0}$.
Ausgabe: alle $\boldsymbol{\alpha} \in \mathbb{Z}^d\backslash\{\boldsymbol{0}\}$ mit $\|\sum_{i=1}^d \alpha_i\boldsymbol{b}_i\| \leq \mu$, $\alpha_j \leq 0$, $j = \max\{i \in \{1,\ldots,d\} : \alpha_i \neq 0\}$.
Berechne die GRAM-Matrix G bzgl. $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_d)$. Berechne die CHOLESKY-Zerlgung $G = R^tR$ von G.
$i \leftarrow d$. *new_bound* $\leftarrow$ *true*. *finished* $\leftarrow$ *false*.
while $\neg$*finished* **do**
  if *new_bound* **then**
    $T_i \leftarrow \sum_{k=i+1}^{d} r_{kk}^2\left(\alpha_k + \sum_{j=k+1}^{d}\frac{r_{kj}}{r_{kk}}\alpha_j\right)^2$.
    $\alpha_i \leftarrow \left\lceil \sqrt{\frac{\mu-T_i}{r_{ii}^2}} - \sum_{j=i+1}^{d}\frac{r_{ij}}{r_{ii}}\alpha_j \right\rceil - 1$.
    $U_i \leftarrow \left\lfloor \sqrt{\frac{\mu-T_i}{r_{ii}^2}} - \sum_{j=i+1}^{d}\frac{r_{ij}}{r_{ii}}\alpha_j \right\rfloor$.
  end if
  $\alpha_i \leftarrow \alpha_i + 1$.
  if $\alpha_i \leq U_i$ **then**
    if $i = 1$ **then**
      if $\boldsymbol{\alpha} = \boldsymbol{0}$ **then**
        finished $\leftarrow$ *true*.
      else
        output $\boldsymbol{\alpha}$.
      end if
    else
      $i \leftarrow i - 1$. *new_bound* $\leftarrow$ *true*.
    end if
  else
    $i \leftarrow i + 1$.
  end if
end while

Proposition 5.1.4. (*[FP85]*) Es seien $d \in \mathbb{N}$ und $\mu \in \mathbb{R}_{>0}$ fest gewählt. Die Laufzeit von Algorithmus 5.1.3 ist dann polynomiell in der Länge der Eingabe $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_d)$ beschränkt.

Dies soll zunächst genügen. Wir wenden uns nun weniger ausgiebig dem anderen wichtigen Gitterproblem CVP zu.

5.2 Das „closest vector problem“ (CVP)

Im Gegensatz zum SVP besitzt das „closest vector problem“ keine erwähnenswerte mathematische Geschichte. Dennoch ist es in der Praxis nicht weniger wichtig. Anwendungen finden sich in der Diskretisierung von Objekten im $\mathbb{R}^d$, wie sie in der graphischen oder in der akkustischen Datenverarabeitung bei Analog/Digital-Umwandlern vorkommen.

Definition 5.2.1. (*„closest vector problem“* (CVP))[2]
Das „closest vector problem“ läßt sich als Minimierungsproblem auffassen. Es ist dann CVP $= (I, S, v, \min)$ mit

$$
\begin{aligned}
I &= \{((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x}) \in \mathbb{Q}^{d\times d} \times \mathbb{Q}^d : \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \text{ linear unabhängig, } d \in \mathbb{N}\},\\
S &: \begin{cases} I & \to \quad 2^{\mathbb{Q}^d} \\ ((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x}) & \mapsto \quad \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d, \end{cases}\\
v &: \begin{cases} S((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x}) & \to \quad \mathbb{R}_{\geq 0} \\ \boldsymbol{v} & \mapsto \quad d(\boldsymbol{v}, \boldsymbol{x}) = \|\boldsymbol{v} - \boldsymbol{x}\|. \end{cases}
\end{aligned}
$$

Falls $\boldsymbol{x}$ ein Gittervektor von $\mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ ist, ist $\mathrm{opt}((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x}) = 0$, was nach Definition 2.2.1 nicht erlaubt ist. Dies soll uns nicht stören, da es einen effizienten Algorithmus gibt (siehe Kapitel 5.4), mit dem dieser Fall ausgeschlossen werden kann. Weil die Notation nicht verkompliziert werden soll, verzichten wir auf eine offensichtliche Umformulierung des CVP.

Wir resümieren aktuelle Ergebnisse der Komplexitätstheorie des CVP, die vor allem im Umfeld des $\mathcal{PCP}$-Theorems entstanden sind.

VAN EMDE BOAS bewies in [vEB81], daß die Entscheidungsvariante des CVP $\mathcal{NP}$-vollständig ist. Unter Anwendung des $\mathcal{PCP}$-Theorems, das zu diesem Zeitpunkt noch nicht so hieß, zeigten ARORA, BABAI, STERN und SWEEDYK, daß es unter der Voraussetzung „$\mathcal{P} \neq \mathcal{NP}$“ keinen effizienten deterministischen Approximationsalgorithmus mit konstanter Worst-Case-Güte für CVP gibt. DINUR, KINDLER und SAFRA verschärften in [DKS98] mit verfeinerten, aber ähnlichen Methoden, daß es für ein $\alpha \in (0, \frac{1}{2})$ $\mathcal{NP}$-hart ist, eine $2^{(\log_2 d)^{\varepsilon}}$-Approximation, $\varepsilon = (\log_2 \log_2 d)^{-\alpha}$, für das CVP bei Gittern der Dimension d, zu berechnen.

ARORA, BABAI, STERN und SWEEDYK bemerkten ausdrücklich, daß die Instanzen des CVP, die in ihrem Beweis vorkommen, eine sehr spezielle Form besitzen. Es ist möglich, die gültigen Lösungen auf Gittervektoren zu beschränken, die bzgl. der eingegebenen Gitterbasis nur Koeffizienten aus der Menge $\{0, 1\}$ besitzen. Das entsprechend eingeschränkte Problem heißt $0/1$-CVP. Man kann sogar noch einen Schritt weiter gehen.

Definition 5.2.2. Es seien $c \in \mathbb{R}_{>0}$ und $a \in \mathbb{R}_{\geq 1}$. Wir definieren das Entscheidungsproblem Gap'-(c, a)-CVP wie folgt: Es sei $w = ((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x}) \in I$ eine Instanz von CVP und es gelte:

- $w \in$ Gap'-(c, a)-CVP, wenn es ein $\boldsymbol{\alpha} \in \{0, 1\}^d$ mit $d(\sum_{i=1}^d \alpha_i \boldsymbol{b}_i, \boldsymbol{x}) \leq c$ gibt.
- $w \notin$ Gap'-(c, a)-CVP, wenn für alle $\boldsymbol{\alpha} \in \mathbb{Z}^d$ und alle $\lambda \in \mathbb{Z}\backslash\{\boldsymbol{0}\}$ die Ungleichung $d(\sum_{i=1}^d \alpha_i \boldsymbol{b}_i, \lambda\boldsymbol{x}) > ca$ gilt.

Gap'-(c, a)-CVP ist ein Promise-Problem. Wenn für alle Gittervektoren, deren Koeffizienten bzgl. der eingegebenen Gitterbasis nur 0 oder 1 sind, der Abstand von $\boldsymbol{x}$ größer als c ist, kann versprochen werden, daß erstens der Abstand sämtlicher Gittervektoren von $\boldsymbol{x}$ größer als ca ist und zweitens dies für alle ganzzahligen Vielfachen von $\boldsymbol{x}$ gilt. Ein solches Versprechen kann für $0/1$-CVP nicht gegeben werden.

Aus dem Beweis von ARORA, BABAI, STERN und SWEEDYK läßt sich das folgende Theorem ablesen.

[2] In der Literatur wird das CVP manchmal auch als „nearest vector problem“ (NV) bezeichnet.

Theorem 5.2.3. Zu jeder Konstanten $a \in \mathbb{R}_{\geq 1}$ gibt es eine rationale Zahl $c \in \mathbb{Q}_{>0}$ so, daß das Promise-Problem Gap'-(c, a)-CVP $\mathcal{NP}$-vollständig ist.

5.3 Beziehungen zwischen dem SVP und dem CVP

Die Gitterprobleme SVP und CVP sind komplexitätstheoretisch eng miteinander verknüpft. Es ist natürlich über den Umweg der $\mathcal{NP}$-Vollständigkeit des CVP möglich, eine polynomielle Reduktion von SVP auf CVP anzugeben. Sie sind aber viel enger verbunden. In diesem Abschnitt geben wir eine TURING-Reduktion[3] von SVP auf CVP an, die außerordentlich simpel und elegant ist.

Als erster zeigte HENK in [Hen97] eine TURING-Reduktion von SVP auf CVP auf. GOLDREICH, MICCIANCIO, SAFRA und SEIFERT geben in [GMSS99] eine Reduktion mit elementaren Methoden an, bei der die Gitterdimension von Eingabe- und Anfragegittern gleich ist und die Aussagen über Approximierbarkeit zuläßt.

Theorem 5.3.1. SVP ist auf CVP TURING-reduzierbar, es gilt SVP $\leq_T$ CVP.
In der im Beweis angegebenen TURING-Reduktion werden für eine Probleminstanz des SVP, die aus einem Gitter der Dimension d besteht, d Anfragen an das CVP-Orakel mit Gittern der gleichen Dimension gestellt, deren Disjunktion das Ergebnis liefert.

Beweis. Algorithmus 5.3.2 gibt die TURING-Reduktion von SVP auf CVP an. Offensichtlich ist die Laufzeit von Algorithmus 5.3.2 in der Länge der Eingabe polynomiell beschränkt, wobei für eine Anfrage an das CVP-Orakel konstante Zeit berechnet wird.

Algorithmus 5.3.2 TURING-Reduktion von SVP auf CVP

Eingabe: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$ linear unabhängig, $L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \boldsymbol{b}_d$, $\mu \in \mathbb{R}_{>0}$.
Ausgabe:
„Ja“, falls es ein $\boldsymbol{v} \in L \backslash \{\boldsymbol{0}\}$ mit $\|\boldsymbol{v}\| \leq \mu$ gibt.
„Nein“, falls es ein solches $\boldsymbol{v}$ nicht gibt.
for $i = 1, \ldots, d$ **do**
Aufruf des CVP-Orakels mit der Eingabe $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1}, 2\boldsymbol{b}_i, \boldsymbol{b}_{i+1}, \ldots, \boldsymbol{b}_d), \boldsymbol{b}_i, \mu)$ liefert die Antwort a_i.
end for
$a \leftarrow a_1 \vee \ldots \vee a_d$.
output a.

Es sei $L_i := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_{i-1} \oplus \mathbb{Z}(2\boldsymbol{b}_i) \oplus \mathbb{Z}\boldsymbol{b}_{i+1} \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ das Gitter der i-ten Anfrage an das CVP-Orakel. Aus den folgenden zwei Behauptungen folgt die Korrektheit des Algorithmus.

i) Für jeden Minimalvektor $\boldsymbol{v} \in L \backslash \{\boldsymbol{0}\}$ gibt es ein $i \in \{1, \ldots, d\}$, so daß $\boldsymbol{v} + \boldsymbol{b}_i \in L_i$ ist.

Falls die Eingabe $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \mu)$ eine Ja-Instanz der Entscheidungsvariante des SVP ist, ist auch $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1}, 2\boldsymbol{b}_i, \boldsymbol{b}_{i+1}, \ldots, \boldsymbol{b}_d), \boldsymbol{b}_i, \mu)$ eine Ja-Instanz der Entscheidungsvariante des CVP.

[3] Eine TURING-Reduktion einer Sprache $L \subseteq \Sigma^*$ auf eine Sprache $L' \subseteq \Sigma^*$ ist eine Abbildung, die von einer polynomiell zeitbeschränkten deterministischen TURING-Maschine realisiert werden kann, die Anfragen an ein L'-Orakel stellen darf, die jeweils mit konstanter Zeit berechnet werden, und die genau die Wörter von L auf Wörter von L' abbildet. Im Gegensatz zu einer polynomiellen Reduktion sind einer TURING-Reduktion mehrere Anfragen an ein L'-Orakel erlaubt (Notation: $L' \leq_T L$).

Beweis. Es sei $\boldsymbol{v} = \sum_{j=1}^{d} \alpha_j \boldsymbol{b}_j \in L \backslash \{\mathbf{0}\}$ ein Minimalvektor von L. Es existiert ein $i \in \{1, \ldots, d\}$, so daß α_i ungerade ist, denn sind alle Koeffizienten von $\boldsymbol{v}$ gerade, folgt $0 < \|\frac{1}{2}\boldsymbol{v}\| < \|\boldsymbol{v}\|$ im Widerspruch zur Minimalität von $\boldsymbol{v}$. Somit ist $\boldsymbol{v} + \boldsymbol{b}_i = \left(\frac{\alpha_i+1}{2}\right)(2\boldsymbol{b_i}) + \sum_{j=1, j\neq i}^{d} \alpha_j \boldsymbol{b}_j \in L_i$.

ii) Für jeden Gittervektor $\boldsymbol{v} \in L_i$ gilt $\boldsymbol{v} - \boldsymbol{b}_i \in L \backslash \{\mathbf{0}\}$, $i = 1, \ldots, d$.

Falls die Eingabe $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1}, 2\boldsymbol{b}_i, \boldsymbol{b}_{i+1}, \ldots, \boldsymbol{b}_d), \boldsymbol{b}_i, \mu)$ eine Ja-Instanz des CVP ist, ist $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \mu)$ eine Ja-Instanz des SVP.

Beweis. Es sei $\boldsymbol{v} = \alpha_i(2\boldsymbol{b}_i) + \sum_{j=1, j\neq i}^{d} \alpha_j \boldsymbol{b}_j$ ein Gittervektor von L_i. Es gilt $\boldsymbol{v} \neq \boldsymbol{b_i}$ und es ist $\boldsymbol{v} - \boldsymbol{b}_i = \sum_{j=1}^{d} \alpha_j \boldsymbol{b}_j \in L \backslash \{\mathbf{0}\}$, wie gewünscht. ◇

Der Beweis von Theorem 5.3.1 zeigt, daß ein effizienter Algorithmus zur Approximation des CVP sich zu einem effizienten Algorithmus zur Approximation des SVP transformieren läßt. Die Worst-Case-Güte beider Approximationsalgorithmen ist gleich.

Korollar 5.3.3. SVP ist nicht schwerer approximierbar als CVP.

Im nächsten Kapitel geben wir umgekehrt eine approximationserhaltende Reduktion des Gap'-(c, a)-CVP auf SVP an, wobei wir jedoch an eine zahlentheoretische Vermutung glauben müssen.

5.4 Weitere Gitterprobleme

Die Probleme SVP und CVP sind nicht die einzigen algorithmischen Probleme, die in Verbindung mit Gittern auftreten. Wir betrachten hier zwei weitere Probleme, für deren Lösung deterministische Algorithmen mit polynomieller Laufzeit bekannt sind. Somit können diese Algorithmen zur Konstruktion von polynomiellen Reduktionen eingesetzt werden.

Das erste Problem ist das *Zugehörigkeitsproblem*. Gegeben sei ein Gitter L durch eine Gitterbasis $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{Q}^d$ sowie ein Vektor $\boldsymbol{x} \in \mathbb{Q}^d$. Es ist zu entscheiden, ob $\boldsymbol{x}$ ein Gittervektor von L ist. Im wesentlichen kann dieses Problem auf das Lösen eines linearen Gleichungssystems mit zusätzlichen Ganzzahligkeitstests zurückgeführt werden.

Das zweite Problem ist das *Gitterbasisproblem*. Gegeben seien Vektoren $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m \in \mathbb{Q}^d$ und gesucht ist eine Basis des Gitters $\mathbb{Z}\boldsymbol{v}_1 + \cdots + \mathbb{Z}\boldsymbol{v}_m$. Dieses Problem kann z.B. mit Hilfe der sogenannten HERMITE-Normalform der Matrix $V = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m) \in \mathbb{Q}^{d \times m}$ effizient gelöst werden (siehe [Coh93]).

Kapitel 6

Über die $\mathcal{NP}$-Härte des SVP

Die Frage, ob das „shortest vector problem“ $\mathcal{NP}$-vollständig ist, ist ein bislang ungelöstes Problem der theoretischen Informatik.

Der Beweis der Aussage „SVP $\in \mathcal{NP}$“ ist schon seit langem bekannt und wurde hier in Kapitel 5 vorgestellt. Ein Durchbruch gelang AJTAI in [Ajt98], als er zeigte, daß sich jedes Problem aus der Klasse $\mathcal{NP}$ mittels einer *randomisierten* polynomiellen Reduktion auf SVP zurückführen läßt. AJTAI reduziert eine $\mathcal{NP}$-vollständige Variante des bekannten „subset sum problem“ auf SVP. Sein Beweis beruht auf zwei Ideen:

- einer konstruktiven, probabilistischen Aussage über Hypergraphen, die SAUER 1972 in einer nicht-konstruktiven Version vorstellte,
- einer effizienten Konstruktion eines Gitters, das exponentiell viele Gittervektoren in einer kleinen Kugel besitzt.

Darüber hinaus beweist AJTAI, daß die Berechnung einer $(1 + 1/2^{d^\varepsilon})$-Approximation für SVP für ein $\varepsilon \in \mathbb{R}_{>0}$ $\mathcal{NP}$-hart ist. CAI und NERURKAR verbessern in [CN98] den Faktor auf $1+1/d^\varepsilon$. Wir folgen in diesem Kapitel der Darstellung von MICCIANCIO ([Mic98a] und [Mic98b]), der den Faktor auf $\sqrt{2}$ anheben konnte. Eine weitere Abweichung von AJTAIs Originalbeweis ist die Ersetzung der Anwendung der probabilistischen Aussage über Hypergraphen durch die Anwendung einer zahlentheoretischen Vermutung. Wenn sie wahr ist, ergibt sich sogar eine *deterministische* polynomielle Reduktion einer $\mathcal{NP}$-vollständigen Variante von CVP auf SVP.

6.1 Eine zahlentheoretische Vermutung

Schon in der Einleitung dieses Kapitels wurde erwähnt, daß wir SVP $\in \mathcal{NPC}$ nur unter der Annahme der Richtigkeit einer zahlentheoretischen Vermutung beweisen können. Wir wenden uns dieser von MICCIANCIO aufgestellten Vermutung zu und überlegen, warum sie plausibel ist.

Vermutung 6.1.1. Für jedes $\varepsilon \in \mathbb{R}_{>0}$ gibt es ein $s \in \mathbb{R}_{>0}$, so daß für alle genügend großen Zahlen $q \in \mathbb{N}$ stets eine Zahl im Intervall $[q, q + q^{\varepsilon}]$ existiert, deren Primfaktoren nur einfach auftreten und alle kleiner als $(\ln q)^s$ sind. Eine solche Zahl heißt quadratfreie, $(\ln q)^s$-glatte Zahl.

MICCIANCIO merkt in [Mic98b] an, daß einfache Methoden der analytischen Zahlentheorie ausreichen, um die Aussage „Die mittlere Anzahl von quadratfreien, $(\ln q)^s$-glatten Zahlen im Intervall $[q, q + q^{\varepsilon}]$ übertrifft $q^{\varepsilon - 1/s}$." zu zeigen. Wenn quadratfreie Zahlen, die ausschließlich kleine Primfaktoren besitzen, genügend gleichverteilt auftreten, dann folgt die Vermutung. Es ist aber durchaus möglich, daß zur Zeit ein Beweis ein nicht angreifbares Ziel ist.

6.2 Effiziente Konstruktion eines Gitters

Ziel dieses Abschnitts ist die Angabe eines effizienten deterministischen Algorithmus, der bei der Eingabe von 1^d ein Gitter bestimmt, dessen Minimum nicht zu klein ist und das gleichzeitig exponentiell viele Gittervektoren in einer kleinen Kugel um einen Punkt, der ebenfalls von dem Algorithmus berechnet wird, konzentriert. Diese beiden Eigenschaften des zu konstruierenden Gitters sind für die im nächsten Abschnitt dargestellte Reduktion einer Variante des CVP auf SVP notwendig.
Nachdem wir das Ziel durch exakte Formeln ausgedrückt haben, beschreiben wir ein parametrisiertes Gitter und einen parametrisierten Punkt und beweisen für sie die oben genannten Eigenschaften. Dies geschieht jeweils in zwei Schritten: zuerst für ein reelles Gitter und anschließend für eine ganzzahlige Approximation einer Skalierung des reellen Gitters. Danach erhalten wir den Algorithmus durch geschickte Parameterwahl und durch die Vermutung 6.1.1.

Proposition 6.2.1. Für genügend großes $d \in \mathbb{N}$ und für jede Zahl $\tilde{a} \in [1, \sqrt{2})$ gibt es einen effizienten deterministischen Algorithmus, der bei der Eingabe von 1^d Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{Z}^{n+1}$, einen Vektor $\boldsymbol{x} \in \mathbb{Z}^{n+1}$, eine Matrix $A \in \mathbb{Z}^{d \times n}$, sowie eine rationale Zahl $r \in \mathbb{Q}$ berechnet, so daß folgendes gilt:

i) Für das Gitter $L := \mathbb{Z}\boldsymbol{b}_1 + \cdots + \mathbb{Z}\boldsymbol{b}_n$ ist $\min L > \tilde{a}r$.

ii) Für alle $\boldsymbol{\beta} \in \{0,1\}^d$ gibt es ein $\boldsymbol{\alpha} \in \mathbb{Z}^n$ mit $A\boldsymbol{\alpha} = \boldsymbol{\beta}$ und $d(\sum_{i=1}^n \alpha_i \boldsymbol{b}_i, \boldsymbol{x}) \leq r$.

Das im nächsten Lemma definierte Gitter L wurde zuerst von ADLEMAN untersucht, um einen direkten komplexitätstheoretischen Zusammenhang zwischen SVP und der Faktorisierung von ganzen Zahlen herzustellen. Wir zeigen, daß das Minimum von L nicht zu klein und durch die Wahl des Parameters δ steuerbar ist.

Lemma 6.2.2. Es sei $\delta \in \mathbb{R}_{\geq 1}$ und es seien $m_1, \ldots, m_n \in \mathbb{Z}_{\geq 3}$ ganze Zahlen ($m_i \geq 3$ wird erst in Lemma 6.2.4 benötigt), die paarweise prim zueinander sind, d.h. es gilt $\mathrm{ggT}(m_i, m_j) = 1$ für

$1 \leq i < j \leq n$. Die Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ seien definiert als

$$\boldsymbol{b}_1 := \begin{pmatrix} \sqrt{\ln m_1} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \delta \ln m_1 \end{pmatrix}, \boldsymbol{b}_2 := \begin{pmatrix} 0 \\ \sqrt{\ln m_2} \\ 0 \\ \vdots \\ 0 \\ 0 \\ \delta \ln m_2 \end{pmatrix}, \ldots, \boldsymbol{b}_n := \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \sqrt{\ln m_n} \\ \delta \ln m_n \end{pmatrix} \in \mathbb{R}^{n+1}.$$

Dann ist die Menge $L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_n$ ein Gitter mit $\min L \geq \sqrt{2 \ln \delta}$.

Beweis. Es sei $\boldsymbol{\alpha} \in \mathbb{Z}^n \backslash \{\mathbf{0}\}$. Wir führen die folgende Notation ein:

$$g^+ := \prod_{\substack{i=1 \\ \alpha_i > 0}}^{n} m_i^{\alpha_i}, \quad g^- := \prod_{\substack{i=1 \\ \alpha_i < 0}}^{n} m_i^{-\alpha_i}, \quad g := g^+ g^- = \prod_{i=1}^{n} m_i^{|\alpha_i|}.$$

Für das Quadrat der Länge des Gittervektors $\sum_{i=1}^n \alpha_i \boldsymbol{b}_i$ in L gilt die Ungleichung

$$\begin{aligned} \left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i \right\|^2 &= \sum_{i=1}^{n} \left(\alpha_i \sqrt{\ln m_i} \right)^2 + \left(\sum_{i=1}^{n} \alpha_i \delta \ln m_i \right)^2 \\ &\geq \sum_{i=1}^{n} |\alpha_i| \ln m_i + \delta^2 \left(\sum_{i=1}^{n} \alpha_i \ln m_i \right)^2 \\ &= \ln g + \delta^2 |\ln g^+ - \ln g^-|^2 \\ &= \ln g + \delta^2 \left(\ln \max\{g^+, g^-\} - \ln \min\{g^+, g^-\} \right)^2 \\ &= \ln g + \delta^2 \left(\ln \left(1 + \frac{|g^+ - g^-|}{\min\{g^+, g^-\}} \right) \right)^2. \end{aligned}$$

Da $\boldsymbol{\alpha} \neq \mathbf{0}$ ist und die m_i paarweise prim zueinander sind, gilt $|g^+ - g^-| \geq 1$. Außerdem ist dann $g^+ \neq g^-$ und $\min\{g^+, g^-\} < \sqrt{g}$. Damit erhalten wir

$$\ln g + \delta^2 \left(\ln \left(1 + \frac{|g^+ - g^-|}{\min\{g^+, g^-\}} \right) \right)^2 \geq \ln g + \delta^2 \left(ln \left(1 + \frac{1}{\sqrt{g}} \right) \right)^2.$$

Die Funktion $x \mapsto \ln(1 + x)$ ist konkav, d.h. für $0 \leq x \leq y$ und $\tau \in [0, 1]$ gilt

$$\ln(1 + (1 - \tau)x + \tau y) \geq (1 - \tau) \ln(1 + x) + \tau \ln(1 + y).$$

Dies mit $x = 0$, $y = 1$ und $\tau = \frac{1}{\sqrt{g}}$ ergibt

$$\ln g + \delta^2 \left(ln \left(1 + \frac{1}{\sqrt{g}} \right) \right)^2 \geq \ln g + \delta^2 \frac{(\ln 2)^2}{g}.$$

Wir betrachten die Funktion $g \mapsto \ln g + \delta^2 \frac{(\ln 2)^2}{g}$ und sehen nach Anwendung von elementaren analytischen Methoden, daß die Funktion ein absolutes Minimum in $(\delta \ln 2)^2$ besitzt. Somit folgt die Behauptung:

$$\left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i \right\|^2 \geq \ln g + \delta^2 \frac{(\ln 2)^2}{g} \geq \ln(\delta \ln 2)^2 + \frac{(\delta \ln 2)^2}{(\delta \ln 2)^2} \geq 2 \ln \delta.$$

$\diamond$

Das Gitter L aus Lemma 6.2.2 kann nicht für komplexitätstheoretische Betrachtungen herangezogen werden, da es im allgemeinen nicht mit Hilfe von rationalen Zahlen darstellbar ist. Dies ist nicht weiter tragisch, weil es durch ein ganzzahliges Gitter L' „approximiert“ werden kann, das ähnlich gute Eigenschaften besitzt, wie im nachfolgenden Lemma belegt wird. Für $x \in \mathbb{R}$ bezeichnet $\lfloor x \rceil$ die nächstliegende ganze Zahl, d.h. es gilt stets $|x - \lceil x \rceil| \leq \frac{1}{2}$.

Lemma 6.2.3. Es seien $m_1, \ldots, m_n \in \mathbb{Z}_{\geq 3}$, $\delta \in \mathbb{R}_{\geq 1}$, $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^{n+1}$ und $L \subseteq \mathbb{R}^{n+1}$ wie in Lemma 6.2.2. Desweiteren sei $\kappa \in [0, 1)$. Die ganzzahligen Vektoren $\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n \in \mathbb{Z}^{n+1}$ approximieren die skalierten reellen Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ komponentenweise:

$$b'_{ij} := \lfloor \frac{n}{\kappa} b_{ij} \rceil \in \mathbb{Z}, \;\; i = 1, \ldots, n, \;\; j = 1, \ldots, n+1.$$

Dann ist die Menge $L' := \mathbb{Z}\boldsymbol{b}'_1 + \cdots + \mathbb{Z}\boldsymbol{b}'_n \subseteq \mathbb{Z}^{n+1}$ ein Gitter und es gilt für alle $\boldsymbol{\alpha} \in \mathbb{Z}^n$:

$$\left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i \right\| \geq n \left(\frac{1}{\kappa} - 1 \right) \left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i \right\|.$$

Insbesondere gilt $\min L' \geq n \left(\frac{1}{\kappa} - 1 \right) \min L$.

Beweis. Da $\boldsymbol{b}'_i \in \mathbb{Z}^{n+1}$, $i = 1, \ldots, n$, ist die Menge L' ist eine diskrete Untergruppe von $(\mathbb{R}^{n+1}, +)$ und somit laut Proposition 4.1.2 ein Gitter. Zur Vereinfachung der Notation definieren wir die Matrizen $B := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \in \mathbb{R}^{(n+1)\times n}$ und $B' := (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n) \in \mathbb{Z}^{(n+1)\times n}$. Nach der 2. Dreiecksungleichung ($\|\boldsymbol{x}\| - \|\boldsymbol{y}\| \leq \|\boldsymbol{x} - \boldsymbol{y}\|$) gilt

$$\left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i \right\| = \left\| B'\boldsymbol{\alpha} \right\| = \left\| \frac{n}{\kappa} B\boldsymbol{\alpha} + \left(B' - \frac{n}{\kappa} B \right) \boldsymbol{\alpha} \right\| \geq \frac{n}{\kappa} \left\| B\boldsymbol{\alpha} \right\| - \left\| \left(\tilde{B} - \frac{n}{\kappa} B \right) \boldsymbol{\alpha} \right\|.$$

Die 2-Norm auf $\mathbb{R}^n$ ist mit der Quadratsummennorm auf $\mathbb{R}^{(n+1)\times n}$ (die für $A = (a_{ij}) \in \mathbb{R}^{(n+1)\times n}$ durch $\|A\|_2 := \sqrt{\sum_{i=1}^{n+1} \sum_{j=1}^{n} a_{ij}^2}$ definiert ist) verträglich, d.h. für alle $A \in \mathbb{R}^{(n+1)\times n}$ und für alle $\boldsymbol{x} \in \mathbb{R}^n$ gilt $\|A\boldsymbol{x}\|_2 \leq \|A\|_2 \cdot \|\boldsymbol{x}\|_2$ (siehe [Heu91], §114). Da die Einträge der Matrix $B' - \frac{n}{\kappa} B$ im Intervall $[-\frac{1}{2}, \frac{1}{2}]$ liegen, läßt sich die Quadratsummennorm von $B' - \frac{n}{\kappa} B$ durch

$$\left\| B' - \frac{n}{\kappa} B \right\|_2 \leq \sqrt{\sum_{i=1}^{n+1} \sum_{j=1}^{n} \left(\frac{1}{2} \right)^2} = \sqrt{\frac{(n+1)n}{4}} \leq n$$

abschätzen, es gilt also $\|(B' - \frac{n}{\kappa} B)\boldsymbol{\alpha}\| \leq n\|\boldsymbol{\alpha}\|$. Da sämtliche Einträge der Hauptdiagonalen der Matrix B größer als 1 sind (dafür wird $m_i \geq 3$, $i = 1, \ldots n$, benötigt), folgt

$$\begin{aligned} \|B\boldsymbol{\alpha}\| &= \sqrt{\sum_{i=1}^{n} (\sqrt{\ln m_i}\alpha_i)^2 + \left(\sum_{i=1}^{n} \delta \ln m_i \alpha_i \right)^2} \\ &\geq \sqrt{\sum_{i=1}^{n} (\sqrt{\ln m_i}\alpha_i)^2} \\ &\geq \|\boldsymbol{\alpha}\|, \end{aligned}$$

und damit die Behauptung. ◇

Wir haben gefordert, daß es einen Punkt $\boldsymbol{x}$ und eine kleine Kugel um $\boldsymbol{x}$ gibt, die exponentiell viele Gittervektoren von L enthält. Zur Vorbereitung des Beweises dieser Aussage betrachten wir Gittervektoren mit Koeffizienten aus der Menge $\{0,1\}$, die diese Eigenschaft bei geeigneter Parameterwahl von γ und δ besitzen.

Lemma 6.2.4. Es sei $n \geq 2$. Wir benutzen erneut die Bezeichnungen aus Lemma 6.2.2. Es sei $\gamma \in \mathbb{N}$ eine natürliche Zahl. Der Vektor $\boldsymbol{x} \in \mathbb{R}^{n+1}$ ist definiert als $\boldsymbol{x} := (0, \ldots, 0, \delta \ln \gamma)^t$. Falls für einen Vektor $\boldsymbol{\alpha} \in \{0,1\}^n$ die Bedingung $g := \prod_{i=1}^{n} m_i^{\alpha_i} \in [\gamma, \gamma + \frac{\gamma}{\delta}]$ erfüllt ist, gilt

$$d\left(\sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i, \boldsymbol{x}\right) \leq \sqrt{\ln \gamma + 2}.$$

Beweis. Da $g \in [\gamma, \gamma + \frac{\gamma}{\delta}]$ ist, gilt die Abschätzung

$$\ln g \leq \ln\left(\gamma\left(1 + \frac{1}{\delta}\right)\right) = \ln \gamma + \ln\left(1 + \frac{1}{\delta}\right) \leq \ln \gamma + \frac{1}{\delta}.$$

Diese verwenden wir zweimal und erhalten

$$\begin{aligned} d\left(\sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i, \boldsymbol{x}\right)^2 &= \left\|\sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i - \boldsymbol{x}\right\|^2 \\ &= \sum_{i=1}^{n} \left(\alpha_i \sqrt{\ln m_i}\right)^2 + \delta^2 \left(\sum_{i=1}^{n} \alpha_i \ln m_i - \ln \gamma\right)^2 \\ &= \ln g + \delta^2 (\ln g - \ln \gamma)^2 \\ &\leq \ln \gamma + \frac{1}{\delta} + \delta^2 \left(\frac{1}{\delta}\right)^2 \\ &= \ln \gamma + \frac{1}{\delta} + 1. \end{aligned}$$

Da $\delta \in \mathbb{R}_{\geq 1}$ ist, folgt die Behauptung. ◇

Auch hier müssen wir uns um eine ganzzahlige „Approximation“ kümmern, damit wir Lemma 6.2.4 für komplexitätstheoretische Betrachtungen anwenden können.

Lemma 6.2.5. Es seien $m_1, \ldots, m_n \in \mathbb{Z}_{\geq 3}$, $\delta \in \mathbb{R}_{\geq 1}$, $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^{n+1}$, $L \subseteq \mathbb{R}^{n+1}$, $\gamma \in \mathbb{N}$ und $\boldsymbol{x} \in \mathbb{R}^{n+1}$ wie in Lemma 6.2.4. Es seien $\kappa \in [0,1)$, $\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n \in \mathbb{Z}^{n+1}$ und $L' \subseteq \mathbb{Z}^{n+1}$ wie in Lemma 6.2.3. Der ganzzahlige Vektor $\boldsymbol{x}' \in \mathbb{Z}^{n+1}$ approximiert den skalierten reellen Vektor $\boldsymbol{x}$ komponentenweise: $x'_j := \lfloor \frac{n}{\kappa} x_j \rceil \in \mathbb{Z}$, $j = 1, \ldots, n+1$. Für alle $\boldsymbol{\alpha} \in \mathbb{Z}^n$ gilt die Ungleichung

$$d\left(\sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i, \boldsymbol{x}'\right) \leq n\left(\frac{1}{\kappa} + 1\right) d\left(\sum_{i=1}^{n} \alpha_i \boldsymbol{b}_i, \boldsymbol{x}\right).$$

Beweis. Dieser Beweis verläuft nahezu genauso wie der von Lemma 6.2.3. Auch hier definieren wir die Matrizen $B := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \in \mathbb{R}^{(n+1)\times n}$ und $B' := (\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n) \in \mathbb{Z}^{(n+1)\times n}$ zur Vereinfachung der Notation. Anwendung der Dreiecksungleichung liefert

$$d\left(\sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i, \boldsymbol{x}'\right) \leq \frac{n}{\kappa} \|B\boldsymbol{\alpha} - \boldsymbol{x}\| + \left\|\left(B' - \frac{n}{\kappa} B\right)\boldsymbol{\alpha} - \left(\boldsymbol{x}' - \frac{n}{\kappa}\boldsymbol{x}\right)\right\|.$$

Weiter ergibt sich für $\boldsymbol{\alpha} \neq \mathbf{0}$

$$\begin{aligned}\left\|\left(B' - \frac{n}{\kappa}B\right)\boldsymbol{\alpha} - \left(\boldsymbol{x}' - \frac{n}{\kappa}\boldsymbol{x}\right)\right\| &\leq \left\|B' - \frac{n}{\kappa}B\right\| \|\boldsymbol{\alpha}\| + \left\|\boldsymbol{x}' - \frac{n}{\kappa}\boldsymbol{x}\right\| \\ &\leq \sqrt{\frac{(n+1)n}{4}}\|\boldsymbol{\alpha}\| + \frac{1}{2} \\ &\leq n\|\boldsymbol{\alpha}\|.\end{aligned}$$

Da sämtliche Einträge der Hauptdiagonalen der Matrix B größer als 1 sind, folgt

$$\begin{aligned}\|B\boldsymbol{\alpha} - \boldsymbol{x}\| &= \sqrt{\sum_{i=1}^{n}(\sqrt{\ln m_i}\alpha_i)^2 + \left(\sum_{i=1}^{n}\delta \ln m_i \alpha_i - \delta \ln \gamma\right)^2} \\ &\geq \sqrt{\sum_{i=1}^{n}(\sqrt{\ln m_i}\alpha_i)^2} \\ &\geq \|\boldsymbol{\alpha}\|.\end{aligned}$$

Nun ergibt sich zusammenfassend die Behauptung, wobei der Fall $\boldsymbol{\alpha} = \mathbf{0}$ aufgrund der speziellen Wahl von speziellen Wahl von $\boldsymbol{x} = (0, \ldots, 0, \delta \ln \gamma)^t$ und von $x' = (0, \ldots, 0, \lceil \delta \ln \gamma \rceil)^t$ offensichtlich ist. ◇

Wir haben mit Lemma 6.2.3 und 6.2.5 alle benötigten Aussagen beisammen, um Proposition 6.2.1 zu belegen. Die Einzelaussagen werden durch geschickte Parameterwahl und mit Hilfe der Vermutung 6.1.1 zusammengesetzt.

Beweis. (*von Proposition 6.2.1*)
Es seien $0 < \varepsilon < 1 - \frac{\tilde{a}^2}{2} < 1$ (damit $\frac{\tilde{a}}{\sqrt{2(1-\varepsilon)}} < 1$) und $0 < \kappa < (1 - \frac{\tilde{a}}{\sqrt{2(1-\varepsilon)}})/(1 + \frac{\tilde{a}}{\sqrt{2(1-\varepsilon)}})$ (damit $\frac{1-\kappa}{1+\kappa} > \frac{\tilde{a}}{\sqrt{2(1-\varepsilon)}}$) fest gewählt.
Es sei $s \in \mathbb{R}_{>0}$ entsprechend Vermutung 6.1.1 so gewählt, daß für alle genügend großen Zahlen $q \in \mathbb{N}$ eine quadratfreie, $(\ln q)^s$-glatte Zahl im Intervall $[q, q + q^{\varepsilon/2}]$ existiert. Desweiteren sei d so groß, daß dies für alle $q > 2^d$ gilt.
Definiere die Zahlen $\gamma := 2^{2d^2/\varepsilon}$, $\delta := \gamma^{1-\varepsilon}$ und $n := \lceil d + (\ln \gamma)^s \rceil$.
Es seien $m_0, m_{d+1}, \ldots, m_n, m_1, \ldots, m_d$ (in dieser Reihenfolge!) die ersten $n+1$ Primzahlen größer 2. Nach dem Primzahlsatz 3.2.2 ist $\pi(n^2) > \frac{n^2}{2 \ln n} > n + 2$ ($\pi(n^2)$ bezeichnet die Anzahl der Primzahlen, die höchstens n^2 sind). Also ist die Bitlänge von m_n höchstens $\lceil 2 \log_2 n \rceil$, demnach polynomiell in d beschränkt. Somit können $m_0, \ldots, m_n$ durch einen effizienten deterministischen Algorithmus bei Eingabe von 1^d z.B. mit dem Sieb des Eratosthenes berechnet werden.
Durch die hier angegebenen Parameter seien die Vektoren $\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n \in \mathbb{Z}^{n+1}$ aus Lemma 6.2.3 und der Vektor $\boldsymbol{x}' \in \mathbb{Z}^{n+1}$ aus Lemma 6.2.5 definiert. Desweiteren definiere die Matrix $A \in \mathbb{Z}^{d \times n}$ durch $A := (E_d | \mathbf{0}_{d \times (n-d)})$, wobei E_d die $(d \times d)$-Einheitsmatrix bezeichne, und die Zahl $r := \lceil n(\frac{1}{\kappa} + 1)\sqrt{\ln \gamma + 2} \rceil$.
Es ist klar, daß es einen deterministischen Algorithmus gibt, der dies alles bei Eingabe von 1^d in polynomieller Zeit berechnen kann. Man muß sich nur vor Augen halten, daß $\tilde{a}$ eine Konstante und nicht Teil der Eingabe ist.
Nun ist nachzuweisen, daß das so definierte Gitter $L' := \mathbb{Z}\boldsymbol{b}'_1 + \cdots + \mathbb{Z}\boldsymbol{b}'_n$ die gewünschten Eigenschaften besitzt.

i) Nach Lemma 6.2.3 gilt $\min L' \geq n(\frac{1}{\kappa}-1)\sqrt{2\ln\delta}$ und wir müssen die Ungleichung

$$n\left(\frac{1}{\kappa}-1\right)\sqrt{2\ln\delta} > \tilde{a}r = \tilde{a}\left\lceil n\left(\frac{1}{\kappa}+1\right)\sqrt{\ln\gamma+2}\right\rceil$$

bzw.

$$\begin{aligned} & n\left(\frac{1}{\kappa}-1\right)\sqrt{2(1-\varepsilon)\ln\gamma} > \tilde{a}\left(n\left(\frac{1}{\kappa}+1\right)\sqrt{\ln\gamma+2}+1\right) \\ \iff\ & n\left(\left(\frac{1}{\kappa}-1\right)\sqrt{2(1-\varepsilon)\ln\gamma}-\tilde{a}\left(\frac{1}{\kappa}+1\right)\sqrt{\ln\gamma+2}\right) > \tilde{a} \end{aligned}$$

einsehen. Falls für genügend großes d stets

$$\left(\frac{1}{\kappa}-1\right)\sqrt{2(1-\varepsilon)\ln\gamma}-\tilde{a}\left(\frac{1}{\kappa}+1\right)\sqrt{\ln\gamma+2} \ \geq\ 1 \tag{6.1}$$

gilt, folgt die Behauptung unmittelbar. Es ist

$$\begin{aligned} & \sqrt{\ln\gamma}\left(\left(\frac{1}{\kappa}-1\right)\sqrt{2(1-\varepsilon)}-\tilde{a}\left(\frac{1}{\kappa}+1\right)\sqrt{1+\frac{2}{\ln\gamma}}\right) \\ =\ & d\sqrt{\frac{2}{\varepsilon}\ln 2}\left(\left(\frac{1}{\kappa}-1\right)\sqrt{2(1-\varepsilon)}-\tilde{a}\left(\frac{1}{\kappa}+1\right)\sqrt{1+\frac{2}{\ln\gamma}}\right). \end{aligned}$$

Da $\sqrt{2/\varepsilon\ln 2}>1$ ist, muß nur noch $(\frac{1}{\kappa}-1)\sqrt{2(1-\varepsilon)}-\tilde{a}(\frac{1}{\kappa}+1)\sqrt{1+2/\ln\gamma}>1/d$ bestätigt werden:

$$\begin{aligned} & \left(\frac{1}{\kappa}-1\right)\sqrt{2(1-\varepsilon)} > \tilde{a}\left(\frac{1}{\kappa}+1\right)\sqrt{1+\frac{2}{\ln\gamma}}+\frac{1}{d} \\ \iff\ & \frac{1-\kappa}{1+\kappa} > \frac{\tilde{a}}{\sqrt{2(1-\varepsilon)}}\sqrt{1+\frac{2}{\ln\gamma}}+\frac{1}{d}\cdot\frac{1}{\sqrt{2(1-\varepsilon)}}\cdot\frac{1}{\frac{1}{\kappa}-1}. \end{aligned}$$

Nach der Wahl von ε und κ ergibt sich für $d\to\infty$ die Ungleichung (6.1).

ii) Es sei $\boldsymbol{\beta}\in\{0,1\}^d$ vorgegeben und ein entsprechendes $\boldsymbol{\alpha}\in\mathbb{Z}^n$ gesucht. Setze $\alpha_i := \beta_i$, $i=1,\ldots,d$, und $g_1 := \prod_{i=1}^d m_i^{\alpha_i}$. Weil $\lceil\log_2 m_i\rceil \leq d$, $i=1,\ldots,d$, ist, ergibt sich $g_1 < 2^{d^2} = \gamma^{\varepsilon/2}$ und weiter $q := \gamma/g_1 > \gamma^{1-\varepsilon/2} > \gamma^{1/2} = 2^{d^2/\varepsilon} > 2^d$. Nach Vermutung 6.1.1 und der Wahl von d existiert im Intervall $[q, q+q^{\varepsilon/2}]$ eine quadratfreie, $(\ln q)^s$-glatte Zahl g_2. Sie läßt sich als $g_2 = \prod_{i=d+1}^n m_i^{\alpha_i}$ schreiben, wobei $\alpha_{d+1},\ldots,\alpha_n\in\{0,1\}$ so definiert werden sollen (Holzhammermethode: die $(\ln\gamma)^s$-te Primzahl m_n ist größer als $(\ln\gamma)^s \geq (\ln q)^s$). Es ist $g_2 = q+q'$ mit einem $q' < q^{\varepsilon/2} \leq \gamma^{\varepsilon/2}$. Für $g := g_1g_2 = \prod_{i=1}^n m_i^{\alpha_i}$ gilt

$$g-\gamma = g_1g_2-g_1q = g_1(g_2-q) < \gamma^{\varepsilon/2}\gamma^{\varepsilon/2} = \gamma^\varepsilon = \gamma/\delta.$$

Mit Lemma 6.2.5 folgt wie gewünscht

$$d\left(\sum_{i=1}^n \alpha_i\boldsymbol{b}_i', \boldsymbol{x}'\right) \leq n\left(\frac{1}{\kappa}+1\right)\sqrt{\ln\gamma+2} \leq r.$$

$\diamond$

6.3 Eine Reduktion von CVP auf SVP

Für den Beweis der $\mathcal{NP}$-Vollständigkeit des SVP geben wir eine approximationserhaltende Reduktion von dem Promise-Problem Gap'-(c', a')-CVP (siehe Definition 5.2.2) auf das Promise-Problem Gap-(c, a)-SVP an.

Theorem 6.3.1. Unter Annahme der Richtigkeit der Vermutung 6.1.1 ist es für jeden Approximationfaktor $a \in [1, \sqrt{2})$ $\mathcal{NP}$-hart, eine a-Approximation für das „shortest vector problem“ zu berechnen.

Beweis. Es seien $\tilde{a} \in (a, \sqrt{2}) \cap \mathbb{Q}$ und $a' \in \mathbb{Z}$ mit $a' \geq \sqrt{\frac{a^2\tilde{a}^2}{\tilde{a}^2 - a^2}}$ gewählt.
Nach Theorem 5.2.3 gibt es eine Zahl $c' \in \mathbb{Q}_{>0}$, so daß das Promise-Problem Gap'-(c', a')-CVP $\mathcal{NP}$-vollständig ist. Im folgenden geben wir eine polynomielle Reduktion von Gap'-(c', a')-CVP auf Gap-$(\frac{c'a'}{a}, a)$-SVP, bzw. eine approximationserhaltende Reduktion von dem entsprechend zu modifizierenden Optimierungsproblem CVP auf SVP zu den Parametern (c', a') und $(\frac{c'a'}{a}, a)$ an. Es sei $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x})$ eine Eingabe für das Promise-Problem Gap'-(c', a')-CVP. Falls d zu klein ist, um Proposition 6.2.1 anzuwenden, muß die Eingabe mit entsprechend konstant vielen Vektoren künstlich verlängert werden. Nach Proposition 6.2.1 gibt es zu $\tilde{a}$ einen Algorithmus, der bei Eingabe von 1^d die Vektoren $\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_n \in \mathbb{Z}^{n+1}$, den Vektor $\boldsymbol{x}' \in \mathbb{Z}^{n+1}$, die Matrix $A \in \mathbb{Z}^{d \times n}$, sowie die Zahl $r \in \mathbb{Q}$, mit den in Proposition 6.2.1 angegebenen Eigenschaften berechnet. Zur Vereinfachung der Notation wählen wir $B \in \mathbb{Q}^{d \times d}$ als die Matrix, die die Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ als Spaltenvektoren enthält. Definiere den Skalierungsfaktor $\kappa := \frac{c'a'}{\tilde{a}r}$ und die Vektoren $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{n+1} \in \mathbb{Q}^{d+(n+1)}$:

$$\boldsymbol{v}_i := \begin{pmatrix} BA\boldsymbol{e}_i \\ \kappa \boldsymbol{b}'_i \end{pmatrix},\ i = 1, \ldots, n, \quad \boldsymbol{v}_{n+1} := \begin{pmatrix} \boldsymbol{x} \\ \kappa \boldsymbol{x}' \end{pmatrix},$$

wobei $\boldsymbol{e}_i$, $i = 1, \ldots, n$, wie üblich, den i-ten Standardbasisvektor von $\mathbb{R}^n$ bezeichnet. So erhalten wir ein Gitter $M := \mathbb{Z}\boldsymbol{v}_1 + \cdots + \mathbb{Z}\boldsymbol{v}_{n+1}$, das wir nach Anwendung einer effizient berechenbaren, auf M isometrisch wirkenden linearen Abbildung als Eingabe von SVP auffassen können.

1. Behauptung: Falls ein $\boldsymbol{\beta} \in \{0, 1\}^d$ mit $d(\sum_{i=1}^d \beta_i \boldsymbol{b}_i, \boldsymbol{x}) \leq c'$ existiert, d.h. $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x}) \in$ Gap'-(c', a')-CVP, ist $\min M \leq \frac{c'a'}{a}$.

Beweis. (*der 1. Behauptung*)
Von Proposition 6.2.1 wissen wir, daß es ein $\boldsymbol{\alpha} \in \mathbb{Z}^n$ mit $A\boldsymbol{\alpha} = \boldsymbol{\beta}$ und $d(\sum_{i=1}^n \alpha_i \boldsymbol{b}'_i, \boldsymbol{x}') \leq r$ gibt. Setze $\boldsymbol{\gamma} := \binom{\boldsymbol{\alpha}}{-1} \in \mathbb{Z}^{n+1}$. Dann gilt

$$\left\| \sum_{i=1}^{n+1} \gamma_i \boldsymbol{v}_i \right\|^2 = \left\| \sum_{i=1}^{d} \beta_i \boldsymbol{b}_i - \boldsymbol{x} \right\|^2 + \kappa^2 \left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i - \boldsymbol{x}' \right\|^2 \leq c'^2 + \kappa^2 r^2 = c'^2 \left(1 + \frac{a'^2}{\tilde{a}^2}\right).$$

Es gilt $1 + \frac{a'^2}{\tilde{a}^2} \leq \frac{a'^2}{a^2}$, weil a' gerade so gewählt wurde. Damit ist die 1. Behauptung bewiesen. $\diamond$

2. Behauptung: Falls $((\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \boldsymbol{x}) \notin$ Gap'-(c', a')-CVP, ist $\min M > \frac{c'a'}{a} a = c'a'$.

Beweis. (*der 2. Behauptung*)
Es seien $\gamma = \binom{\boldsymbol{\alpha}}{\lambda} \in \mathbb{Z}^{n+1} \backslash \{\boldsymbol{0}\}$ und $\boldsymbol{\beta} := A\boldsymbol{\alpha} \in \mathbb{Z}^d$. Da

$$\left\| \sum_{i=1}^{n+1} \gamma_i \boldsymbol{v}_i \right\|^2 = \left\| \sum_{i=1}^{d} \beta_i \boldsymbol{b}_i + \lambda \boldsymbol{x} \right\|^2 + \kappa^2 \left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i + \lambda \boldsymbol{x}' \right\|^2$$

gilt, genügt es zu zeigen, daß einer der Summanden größer als $(c'a')^2$ ist. Falls $\lambda = 0$ ist, muß $\boldsymbol{\alpha} \neq \mathbf{0}$ sein. Dann erhalten wir wegen Proposition 6.2.1

$$\kappa \left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i + \lambda \boldsymbol{x}' \right\| = \kappa \left\| \sum_{i=1}^{n} \alpha_i \boldsymbol{b}'_i \right\| \geq \kappa \min L > \kappa \tilde{a} r = c'a'.$$

Falls $\lambda \neq 0$ ist, ist schon nach Voraussetzung $\| \sum_{i=1}^{d} \beta_i \boldsymbol{b}_i + \lambda \boldsymbol{x} \| > c'a'$. $\diamond$

Kapitel 7

Grenzen der $\mathcal{NP}$-Härte der Approximierbarkeit von kurzen Gittervektoren

Die schwerfällige Kapitelüberschrift vermittelt bereits, daß wir in diesem Kapitel eine Grenze a angeben werden, ab der die Berechnung einer a-Approximation für SVP nicht $\mathcal{NP}$-hart sein kann, wenn wir übliche komplexitätstheoretische Annahmen zugrunde legen.

Im vorangehenden Kapitel haben wir bereits gesehen, daß $a \geq \sqrt{2}$ sein muß, falls die Vermutung 6.1.1 zutrifft. Davor haben wir über den LLL-Algorithmus berichtet, der effizient eine $2^{(d-1)/2}$-Approximation für SVP berechnet. Also muß $a \leq 2^{(d-1)/2}$ sein. Zwischen den beiden Schranken klafft eine exponentielle Lücke. Die Grenze der $\mathcal{NP}$-Härte der Approximierbarkeit des SVP ist aber im Vergleich zur Worst-Case-Güte von bekannten effizienten Approximationsalgorithmen sehr niedrig. LAGARIAS, LENSTRA und SCHNORR zeigen in [LLS90] unter Benutzung einer schwächeren Form von Theorem 4.2.5, daß $\mathcal{NP} = \text{co-}\mathcal{NP}$ folgt, falls die Berechnung einer d-Approximation für SVP $\mathcal{NP}$-hart ist. GOLDREICH und GOLDWASSER senkten in [GG98] die Grenze auf $\sqrt{d/\ln d}$.

Wir schauen uns in diesem Kapitel den Beweis von GOLDREICH und GOLDWASSER an. Wir beweisen, daß die polynomielle Hierarchie zu Σ_2 zusammenbricht, wenn die Berechnung einer $\sqrt{d/\ln d}$-Approximation für SVP $\mathcal{NP}$-hart ist. Dabei arbeiten wir eine technische Verbesserung von MICCIANCIO ([Mic99]) ein. Der Beweis besteht im wesentlichen in der Angabe eines interaktiven Beweissystems für die Tatsache, daß ein gegebenes Gitter nur „lange“ Gittervektoren besitzt.

7.1 Ein interaktives Beweissystem für das Komplement von SVP

Ein Highlight der theoretischen Informatik ist, daß die polynomielle Hierarchie zu Σ_2 zusammenbricht, falls das Graph-Isomorphie-Problem $\mathcal{NP}$-vollständig ist. (siehe [Weg95]). Das Komplement des Graph-Isomorphie-Problems liegt in der Klasse $\mathcal{AM}$. Unter der Annahme, daß das Graph-Isomorphie-Problem $\mathcal{NP}$-vollständig ist, kann Theorem 2.1.11 angewendet werden. Es folgt der Zusammenbruch der polynomiellen Hierarchie zu Σ_2.
Mit einer analogen Idee zeigen wir, daß die polynomielle Hierarchie zu Σ_2 zusammenbricht, wenn die Berechnung einer $\sqrt{d/\ln d}$-Approximation für SVP bei d-dimensionalen Gittern $\mathcal{NP}$-hart ist. Da nach Theorem 2.1.10 die Klassen $\mathcal{IP}(2)$ und $\mathcal{AM}$ übereinstimmen, müssen wir nur einsehen, daß es eine Konstante $c \in \mathbb{R}_{>0}$ gibt, so daß Gap-$(c, \sqrt{d/\ln d})$-SVP in $\mathcal{IP}(2)$ liegt.

Theorem 7.1.1. Das Komplement Gap-$(2, \sqrt{d/\ln d})$-SVP liegt in $\mathcal{IP}(2)$. Dabei gibt d die Dimension der eingegebenen Gitter an.

Beweis. Zum Beweis geben wir ein *konkretes* interaktives Beweissystem für das Komplement von Gap-$(2, \sqrt{d/\ln d})$-SVP an, das mit zwei Kommunikationsrunden auskommt.

Zuerst geben wir ein interaktives Beweissystem an, das die wichtigsten Ideen verdeutlicht. Wir analysieren es und erkennen, daß es noch verschiedene technische Schwächen besitzt. So ist Victors Irrtumswahrscheinlichkeit zu groß, und die Analyse setzt $d \geq 9$ voraus. Diese Schwächen können aber durch Standardtechniken ausgemerzt werden.

Es seien $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$ linear unabhängige Vektoren, die die gemeinsame Eingabe für Peggy und Victor darstellen. Es sei $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ das zu betrachtende Gitter.

Victor: Wählt zufällig $\boldsymbol{\alpha} \in \{0,1\}^d$ und $\boldsymbol{y} \in \overline{B}(\mathbf{0}, \sqrt{d/\ln d})$†, berechnet mit Algorithmus 4.3.6 den Vektor $\boldsymbol{x} = (\boldsymbol{y} + \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i) \mod (2\boldsymbol{b}_1, \ldots, 2\boldsymbol{b}_d)$ und sendet $\boldsymbol{x}$ zu Peggy.

Peggy: Bestimmt ein $\boldsymbol{\beta} \in \{0,1\}^d$, so daß für alle $\boldsymbol{v} \in 2L$ der Abstand $d(\sum_{i=1}^{d} \beta_i \boldsymbol{b}_i, \boldsymbol{x} + \boldsymbol{v})$ minimal ist und sendet $\boldsymbol{\beta}$ zu Victor.

Victor: Akzeptiert genau dann, wenn $\boldsymbol{\alpha} = \boldsymbol{\beta}$ ist.

†Ein Algorithmus, der probabilistischen TURING-Maschinen die zufällige gleichverteilte Wahl eines Vektors aus einer Kugel ermöglicht, wird in [Knu69] unter 3.4.1.E beschrieben.

1. Behauptung: Falls $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \notin$ Gap-$(2, \sqrt{d/\ln d})$-SVP ist, d.h. falls $\min L > 2\sqrt{d/\ln d}$ ist, dann akzeptiert Victor die Eingabe immer.

Beweis. (*der 1. Behauptung*)
Victor sendet den Vektor $\boldsymbol{x} = \boldsymbol{y} + \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i \mod (2\boldsymbol{b}_1, \ldots, 2\boldsymbol{b}_d)$ zu Peggy. Es sei $\boldsymbol{v} \in 2L$ mit $\boldsymbol{y} + \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i - \boldsymbol{v} = \boldsymbol{y} + \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i \mod (2\boldsymbol{b}_1, \ldots, 2\boldsymbol{b}_d)$. Einerseits gilt die Ungleichung

$$\begin{aligned} d\left(\sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i, \boldsymbol{x} + \boldsymbol{v}\right) &= \left\| \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i - \boldsymbol{y} - \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i + \boldsymbol{v} - \boldsymbol{v} \right\| \\ &\leq \sqrt{d/\ln d}, \end{aligned}$$

und andererseits gilt für alle $\boldsymbol{w} \in 2L$ und für alle $\boldsymbol{\beta} \in \{0,1\}^d \backslash \{\boldsymbol{\alpha}\}$ die Ungleichung

$$\begin{aligned} d\left(\sum_{i=1}^{d} \beta_i \boldsymbol{b}_i, \boldsymbol{x} + \boldsymbol{w}\right) &= \left\| \sum_{i=1}^{d} \beta_i \boldsymbol{b}_i - \boldsymbol{y} - \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i + \boldsymbol{v} - \boldsymbol{w} \right\| \\ &\geq \left\| \sum_{i=1}^{d} (\beta_i - \alpha_i) \boldsymbol{b}_i + \boldsymbol{v} - \boldsymbol{w} \right\| - \|\boldsymbol{y}\| \\ &> 2\sqrt{d/\ln d} - \sqrt{d/\ln d} \\ &= \sqrt{d/\ln d}. \end{aligned}$$

Da $\sum_{i=1}^{d} (\beta_i - \alpha_i)\boldsymbol{b}_i \notin 2L$ ist, folgt $\sum_{i=1}^{d} (\beta_i - \alpha_i)\boldsymbol{b}_i + \boldsymbol{v} - \boldsymbol{w} \neq \boldsymbol{0}$. Peggy kann immer das ursprüngliche $\boldsymbol{\alpha}$ finden, weil $d(\sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i, \boldsymbol{x} + \boldsymbol{v}) < d(\sum_{i=1}^{d} \beta_i \boldsymbol{b}_i, \boldsymbol{x} + \boldsymbol{w})$ für alle $\boldsymbol{w} \in 2L$ und für alle $\boldsymbol{\beta} \in \{0,1\}^d \backslash \{\boldsymbol{\alpha}\}$ gilt. ⋄

2. Behauptung: Es gibt ein Polynom $p \in \mathbb{R}[X]$ mit der Eigenschaft, daß falls $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \in$ Gap-$(2, \sqrt{d/\ln d})$-SVP ist, d.h. falls $\min L \leq 2$ ist, dann akzeptiert Victor die Eingabe mit einer Wahrscheinlichkeit von höchstens $1 - \frac{1}{|p(d)|}$, ganz gleich, welche Strategie Peggy anwendet. Dabei muß jedoch $d \geq 9$ sein.

Beweis. (*der 2. Behauptung*)
Victor sendet den Vektor $\boldsymbol{x} = \boldsymbol{y} + \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i \mod (2\boldsymbol{b}_1, \ldots, 2\boldsymbol{b}_d)$ zu Peggy. Es sei $\boldsymbol{v} = \sum_{i=1}^{d} \beta_i \boldsymbol{b}_i$ ein Minimalvektor von L. Definiere $\boldsymbol{x'} := \boldsymbol{y} + \sum_{i=1}^{d} \alpha_i \boldsymbol{b}_i + \boldsymbol{v} \mod (2\boldsymbol{b}_1, \ldots, 2\boldsymbol{b}_d)$. Weil $\boldsymbol{v}$ ein Minimalvektor von L ist, gilt $\boldsymbol{x} \neq \boldsymbol{x'}$ (siehe auch den Beweis von 5.3.1). Es sei $\boldsymbol{\alpha'} \in \{0,1\}^d$ mit $\alpha_i' := \alpha_i + \beta_i \mod 2$, $i = 1, \ldots, d$. Peggy kann mit einer hohen Wahrscheinlichkeit $\boldsymbol{x}$ und $\boldsymbol{x'}$ den zugehörigen Vektoren $\boldsymbol{\alpha}$ und $\boldsymbol{\alpha'}$ nicht eindeutig zuordnen, ganz gleich welche Strategie sie anwendet. So kann keine TURING-Machine Victor mit hoher Wahrscheinlichkeit von $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \notin$ Gap-$(2, \sqrt{d/\ln d})$-SVP überzeugen. Es gilt nämlich

$$\begin{aligned} \Pr[\text{Victor akzeptiert}] &= 1 - \Pr[\text{Victor verwirft}] \\ &\leq 1 - \Pr[\text{Peggy kann } \boldsymbol{x} \text{ und } \boldsymbol{x'} \text{ nicht unterscheiden}] \\ &= 1 - \frac{\operatorname{vol}\left(\overline{B}\left(\sum_{i=1}^{d} \alpha_i \boldsymbol{b_i}, \sqrt{\frac{d}{\ln d}}\right) \cap \overline{B}\left(\sum_{i=1}^{d} \alpha_i \boldsymbol{b_i} + \boldsymbol{v}, \sqrt{\frac{d}{\ln d}}\right)\right)}{2 \cdot \operatorname{vol} \overline{B}(\boldsymbol{0}, \sqrt{d/\ln d})} \\ &= 1 - \frac{\operatorname{vol}\left(\overline{B}(\boldsymbol{0}, 1) \cap \overline{B}\left(\frac{\boldsymbol{v}}{\sqrt{d/\ln d}}, 1\right)\right)}{2 \cdot \operatorname{vol} \overline{B}(\boldsymbol{0}, 1)} \end{aligned}$$

Zur Abschätzung des Quotienten benötigen wir ein geometrisches Lemma, das mit Abbildung 7.1 illustriert wird.

Lemma 7.1.2. Es sei $\boldsymbol{x} \in \mathbb{R}^d$ ein Vektor mit $\|\boldsymbol{x}\| < 1$. Dann gilt für das Volumen des Durchschnittes der d-dimensionalen Einheitskugeln $\overline{B}(\boldsymbol{0}, 1)$ und $\overline{B}(\boldsymbol{x}, 1) = \{\boldsymbol{y} \in \mathbb{R}^d : d(\boldsymbol{x}, \boldsymbol{y}) \leq 1\}$ die Ungleichung

$$\frac{\operatorname{vol}(\overline{B}(\boldsymbol{0}, 1) \cap \overline{B}(\boldsymbol{x}, 1))}{\operatorname{vol} \overline{B}(\boldsymbol{0}, 1)} \geq \|\boldsymbol{x}\| \left(\sqrt{1 - \|\boldsymbol{x}\|^2}\right)^{d-1} \sqrt{\frac{d}{2\pi}}.$$

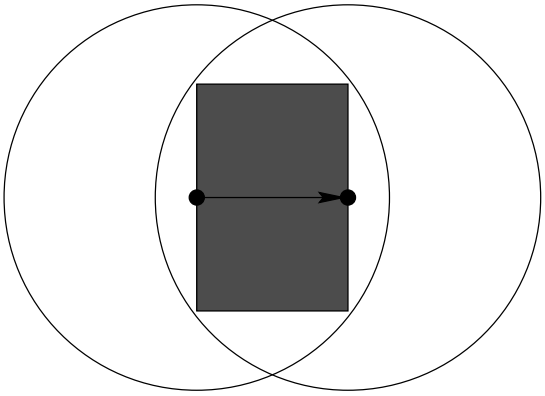

Abbildung 7.1: Illustration zu Lemma 7.1.2

Beweis. Wir schätzen das Volumen von $\overline{B}(\mathbf{0},1) \cap \overline{B}(\boldsymbol{x},1)$ durch das Volumen des Zylinders der Höhe $\|\boldsymbol{x}\|$ und der Breite $2\sqrt{1-\|\boldsymbol{x}\|^2}$, der vollständig in der Schnittmenge $\overline{B}(\mathbf{0},1) \cap \overline{B}(\boldsymbol{x},1)$ liegt, nach unten hin ab. Das Volumen des Zylinders beträgt

$$\|\boldsymbol{x}\| \cdot \operatorname{vol} \overline{B}(\mathbf{0}_{\mathbb{R}^{d-1}}, \sqrt{1-\|\boldsymbol{x}\|^2}) = \|\boldsymbol{x}\| \left(\sqrt{1-\|\boldsymbol{x}\|^2}\right)^{d-1} \frac{\pi^{(d-1)/2}}{\Gamma(\frac{d-1}{2}+1)}.$$

Dann gilt

$$\frac{\operatorname{vol}(\overline{B}(\mathbf{0},1) \cap \overline{B}(\boldsymbol{x},1))}{\operatorname{vol} \overline{B}(\mathbf{0},1)} \geq \frac{\|\boldsymbol{x}\| \left(\sqrt{1-\|\boldsymbol{x}\|^2}\right)^{d-1}}{\sqrt{\pi}} \cdot \frac{\Gamma(\frac{d}{2}+1)}{\Gamma(\frac{d-1}{2}+1)}.$$

Eine einfache analytische Tatsache, die auch für den Beweis der WALLISschen Produktdarstellung von $\frac{\pi}{2} = \lim\limits_{k\to\infty} \frac{2^2 \cdot 4^2 \cdots (2k)^2}{1^2 \cdot 3^2 \cdots (2k-1)^2 \cdot 2k}$ benötigt wird (siehe [Heu94], §94), ist die Ungleichungskette

$$\frac{2 \cdot 4 \cdots (2k)}{1 \cdot 3 \cdots (2k-1)} \cdot \frac{1}{\sqrt{2k+1}} \leq \sqrt{\frac{\pi}{2}} \leq \frac{2 \cdot 4 \cdots (2k)}{1 \cdot 3 \cdots (2k-1)} \cdot \frac{1}{\sqrt{2k}}.$$

Wie schon im Beweis von Korollar 4.2.2 unterscheiden wir zwischen zwei Fällen. Anschließend folgt die Behauptung unmittelbar.

1. Fall: d ist gerade.
Dann gilt

$$\frac{\Gamma(\frac{d}{2}+1)}{\Gamma(\frac{d-1}{2}+1)} = \frac{\frac{2}{2} \cdot \frac{4}{2} \cdots \frac{d}{2}}{\frac{1}{2} \cdot \frac{3}{2} \cdots \frac{d-1}{2}} \cdot \frac{1}{\sqrt{\pi}} = \frac{2}{1} \cdot \frac{4}{3} \cdots \frac{d}{d-1} \cdot \frac{1}{\sqrt{\pi}} \geq \sqrt{\frac{d}{2}}.$$

2. Fall: d ist ungerade.
Dann gilt

$$\frac{\Gamma(\frac{d}{2}+1)}{\Gamma(\frac{d-1}{2}+1)} = \frac{\frac{1}{2} \cdot \frac{3}{2} \cdots \frac{d}{2}}{\frac{2}{2} \cdot \frac{4}{2} \cdots \frac{d-1}{2}} \cdot \sqrt{\pi} = \frac{1}{2} \cdot \frac{3}{4} \cdots \frac{d-2}{d-1} \cdot \sqrt{d}^2 \cdot \sqrt{\pi} \geq \sqrt{\frac{2}{\pi}} \cdot \sqrt{d} \cdot \sqrt{\pi} = \sqrt{2d}.$$

◇

Falls $d \geq 9$ ist, ist $\|\frac{\boldsymbol{v}}{\sqrt{d/\ln d}}\| \leq \frac{2}{\sqrt{d/\ln d}} < 1$. Damit können wir das obige Lemmas anwenden. Es liefert

$$\begin{aligned}\frac{\operatorname{vol}\left(\overline{B}(\mathbf{0},1) \cap \overline{B}(\frac{\boldsymbol{v}}{\sqrt{d/\ln d}},1)\right)}{\operatorname{vol}\overline{B}(\mathbf{0},1)} &\geq \frac{2}{\sqrt{d/\ln d}} \cdot \left(1 - \frac{4\ln d}{d}\right)^{\frac{d-1}{2}} \cdot \sqrt{\frac{d}{2\pi}} \\ &= \sqrt{\frac{4d\ln d}{2\pi d}} \left(1 - \frac{4\ln d}{d}\right)^{\frac{d-1}{2}} \\ &\geq \sqrt{\frac{2\ln d}{\pi}} \left(1 - \frac{4\ln d}{2d/2}\right)^{d/2}.\end{aligned}$$

Nach Proposition B.3 aus [MR95] gilt für alle $t, n \in \mathbb{R}$ mit $n \geq 1$ und $|t| \leq n$ die Ungleichung

$$e^t \geq \left(1 + \frac{t}{n}\right)^n \geq e^t \left(1 - \frac{t^2}{n}\right).$$

Dies mit $n = \frac{d}{2}$ und $t = -\ln d$ (es ist $|\ln d| \leq \frac{d}{2}$, da $d \geq 9$) auf die letzte Ungleichung angewendet ergibt

$$\begin{aligned}\sqrt{\frac{2\ln d}{\pi}} \left(1 - \frac{4\ln d}{2d/2}\right)^{d/2} &\geq \sqrt{\frac{2\ln d}{\pi}} e^{-\ln d} \left(1 - \frac{(2\ln d)^2}{d}\right) \\ &= \sqrt{\frac{2\ln d}{\pi}} \left(\frac{1}{d} - \frac{4\ln^2 d}{d^2}\right).\end{aligned}$$

Demnach gibt es ein Polynom $p \in \mathbb{R}[X]$, so daß die Ungleichung

$$1 - \frac{1}{2} \cdot \frac{\operatorname{vol}\left(\overline{B}(\mathbf{0},1) \cap \overline{B}(\frac{\boldsymbol{v}}{\sqrt{d/\ln d}},1)\right)}{\operatorname{vol}\overline{B}(\mathbf{0},1)} \leq \frac{1}{|p(d)|}$$

für alle $d \geq 9$ gilt. Die Bedingung $d \geq 9$ ist erforderlich, damit die Voraussetzungen von Lemma 7.1.2 $\|\frac{\boldsymbol{v}}{\sqrt{d/\ln d}}\| < 1$ und Proposition B.3 aus [MR95] $|lnd| \leq \frac{d}{2}$ erfüllt sind. ◇

Es gibt ein Polynom $q \in \mathbb{R}[X]$, so daß für alle $d \in \mathbb{N}$ die Ungleichung $\left(1 - \frac{1}{|p(d)|}\right)^{|q(d)|} \leq \frac{1}{4}$ gilt. Es sei $d \geq 9$. Victor sendet nicht nur einen Vektor zu Peggy, sondern $|q(d)|$ viele. Anschließend akzeptiert er genau dann, wenn Peggy bei allen Anfragen richtig antwortet. Dann beträgt die Wahrscheinlichkeit höchstens $\frac{1}{4}$, daß Victor eine Instanz akzeptiert, die zu Gap-$(2, \sqrt{d/\ln d})$-SVP gehört, wie es für ein interaktives Beweissystem gefordert ist. Wenn $d < 9$ ist, muß Victor auf die Hilfe von Peggy verzichten. Dies erfordert nicht zu viel Rechenzeit, denn der Algorithmus 5.1.3 benötigt nach Proposition 5.1.4 für das Abzählen sämtlicher Gittervektoren eines Gitters fester Dimension unterhalb einer festen Längenschranke nur eine in d polynomielle Zeit.

Somit ergibt sich das nachfolgende interaktive Beweissystem für das Komplement von Gap-$(2, \sqrt{d/\ln d})$-SVP, das mit zwei Kommunikationsrunden auskommt.

Es seien $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$ linear unabhängige Vektoren, die die gemeinsame Eingabe für Peggy und Victor darstellen. Es sei $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ das zu betrachtende Gitter.

Victor: Falls $d < 9$ ist, berechnet er mit Hilfe von Algorithmus 5.1.3 sämtliche Gittervektoren, deren Länge 2 nicht überschreitet. Ist dies nur der Nullvektor, dann akzeptiert er, ansonsten verwirft er.

Falls $d \geq 9$ ist, wählt er zufällig $\boldsymbol{\alpha}_i \in \{0,1\}^d$ und $\boldsymbol{y}_i \in \overline{B}(\mathbf{0}, \sqrt{d/\ln d})$ und berechnet mit Algorithmus 4.3.6 die Vektoren $\boldsymbol{x}_i = (\boldsymbol{y}_i + \sum_{j=1}^{d} \alpha_{ij}\boldsymbol{b}_j) \bmod (2\boldsymbol{b}_1, \ldots, 2\boldsymbol{b}_d)$, $i = 1, \ldots, |q(d)|$, und sendet $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{|q(d)|})$ zu Peggy.

Peggy: Bestimmt ein $\boldsymbol{\beta}_i \in \{0,1\}^d$, so daß für alle Vektoren $\boldsymbol{v} \in 2L$ der Abstand $d(\sum_{j=1}^{d} \beta_{ij}\boldsymbol{b}_j, \boldsymbol{x}_i + \boldsymbol{v})$ minimal ist, $i = 1, \ldots, |q(d)|$, und sendet $(\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{|q(d)|})$ zu Victor.

Victor: Akzeptiert genau dann, wenn für alle $i \in \{1, \ldots, |q(d)|\}$ die Gleichung $\boldsymbol{\alpha}_i = \boldsymbol{\beta}_i$ erfüllt ist.

◇

Das im Beweis vorgestellte interaktive Beweissystem besitzt die sogenannte zero-knowledge-Eigenschaft. Falls eine Instanz $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ nicht zu Gap-$(2, \sqrt{d/\ln d})$-SVP gehört, lernt Victor durch die Kommunikation mit Peggy nur, daß $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \notin$ Gap-$(2, \sqrt{d/\ln d})$-SVP gilt, und nicht *mehr*.

Das Hauptergebnis des Kapitels ist in unmittelbarer Reichweite.

Korollar 7.1.3. Wenn die Berechnung einer $\sqrt{d/\ln d}$-Approximation für SVP bei d-dimensionalen Gittern $\mathcal{NP}$-hart ist, dann fällt die polynomielle Hierarchie zu Σ_2 zusammen.

Beweis. Angenommen es gibt eine Konstante $c \in \mathbb{R}_{>0}$, so daß das Entscheidungsproblem Gap-$(c, \sqrt{d/\ln d})$-SVP $\mathcal{NP}$-vollständig ist. Durch Multiplikation mit dem Faktor $\frac{2}{c}$ können Probleminstanzen von Gap-$(c, \sqrt{d/\ln d})$-SVP auf Probleminstanzen von Gap-$(2, \sqrt{d/\ln d})$-SVP polynomiell reduziert werden. Mithin ist auch Gap-$(2, \sqrt{d/\ln d})$-SVP $\mathcal{NP}$-vollständig.

Das Komplement von Gap-$(2, \sqrt{d/\ln d})$-SVP liegt in der Klasse $\mathcal{IP}(2)$, also gilt nach Theorem 2.1.10 Gap-$(2, \sqrt{d/\ln d})$-SVP $\in$ co-$\mathcal{AM}$. Letztendlich folgt mit Theorem 2.1.11 die Behauptung. ◇

Kapitel 8

Worst-Case/Average-Case-Äquivalenz

In der modernen Kryptographie ist es wichtig, daß das für ein kryptographisches Verfahren herangezogene Problem im Average-Case schwierig ist, um Sicherheit garantieren zu können. Dabei ist es wünschenswert, eine Klasse von Probleminstanzen zu kennen, deren Lösung genauso schwierig zu berechnen ist, wie die Lösung einer beliebigen Probleminstanz. Eine solche Klasse von Probleminstanzen gibt das Theorem der Worst-Case/Average-Case-Äquivalenz an, das AJTAI in seinem bahnbrechenden Artikel [Ajt96] vorstellte.

Das Theorem der Worst-Case/Average-Case-Äquivalenz besagt, daß Problem A im Average-Case nicht einfacher zu lösen ist als Problem B im Worst-Case. In diesem Sinne sind der Average-Case von Problem A und der Worst-Case von Problem B äquivalent schwierig. Problem A ist, bei gegebener Matrix $A \in (\mathbb{Z}/q\mathbb{Z})^{d\times n}$ einen Vektor $\boldsymbol{x} \in \{-1,0,1\}^n\backslash\{\mathbf{0}\}$ mit $A\boldsymbol{x} = \mathbf{0} \in (\mathbb{Z}/q\mathbb{Z})^d$ zu finden. Problem B ist das Entscheidungsproblem Gap-$(1, 72d^8\sqrt{(d+3)/4})$-SVP.

In diesem Kapitel stellen wir das Theorem der Worst-Case/Average-Case-Äquivalenz vor und geben einen vollständigen Beweis an. Der Beweis ist technisch sehr anspruchsvoll. Er ist eine freie Ausführung der Beweisskizze von AJTAI.

Auf dem Theorem aufbauend geben wir eine One-Way-Funktion

$$f : \begin{cases} ((\mathbb{Z}/q\mathbb{Z})^{d\times n}, \{0,1\}^n\backslash\{\mathbf{0}\}) & \to & ((\mathbb{Z}/q\mathbb{Z})^{d\times n}, (\mathbb{Z}/q\mathbb{Z})^d) \\ (A, \boldsymbol{x}) & \mapsto & (A, A\boldsymbol{x}) \end{cases}$$

an. Sie ist im Average-Case schwierig zu invertieren, wenn es im Worst-Case schwierig ist, das Minimum eines Gitters bis auf einen Faktor von $O(d^9)$ zu approximieren.

8.1 Abzählen von Gittervektoren in einem Parallelotop

Es seien ein Gitter $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq \mathbb{R}^d$ und ein Parallelotop P gegeben. Wir möchten die Anzahl der Gitterpunkte in P zählen. Der Fundamentalbereich $F^- = P^-(\mathbf{0}; \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) = \{\sum_{i=1}^d \alpha_i \boldsymbol{b}_i : \alpha_i \in [0,1)\}$ des Gitters enthält genau einen Gitterpunkt. Der Quotient des Volumens von P und des Volumens von F gibt dann ungefähr die Anzahl der Gitterpunkte in P an. Diese vage Vorstellung wird in Proposition 8.1.2 durch konkrete Zahlen manifestiert. Doch vorher benötigen wir ein Lemma. Wesentliches technisches Hilfsmittel ist der Skalierungsoperator $\bullet$, den wir in Definition 4.3.4 kennengelernt haben.

Lemma 8.1.1. Es seien $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq \mathbb{R}^d$ ein Gitter, $M := \max_{i=1,\ldots,d} \|\boldsymbol{b}_i\|$ eine Konstante und $F := P(\mathbf{0}; \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ ein Fundamentalbereich. Ferner seien $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_d \in \mathbb{R}^d$ linear unabhängige Vektoren und $\boldsymbol{b} \in \mathbb{R}^d$. Für das Parallelotop $P := P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$ gelten die folgenden Aussagen:

i) Wenn $\boldsymbol{v} \in L$ und $(\boldsymbol{v} + F) \cap P \neq \emptyset$ ist, dann gilt $\boldsymbol{v} + F \subseteq (1 + \frac{2Md}{\operatorname{width} P}) \bullet P$.

ii) Falls die Bedingungen $2Md < \operatorname{width} P$, $\boldsymbol{v} \in L$ und $(\boldsymbol{v} + F) \cap (1 - \frac{2Md}{\operatorname{width} P}) \bullet P \neq \emptyset$ erfüllt sind, gilt $\boldsymbol{v} + F \subseteq P$.

Beweis.

i) Als erstes wird gezeigt, daß für alle $\boldsymbol{x} \in P$ gilt:

$$\overline{B}(\boldsymbol{x}, Md) = \{\boldsymbol{y} \in \mathbb{R}^d : \|\boldsymbol{x} - \boldsymbol{y}\| \leq Md\} \subseteq \left(1 + \frac{2Md}{\operatorname{width} P}\right) \bullet P.$$

Jedes $\boldsymbol{y} \in \overline{B}(\boldsymbol{x}, Md)$ läßt sich schreiben als $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{x}'$ mit $\boldsymbol{x}' \in \overline{B}(\mathbf{0}, Md)$. Es gilt nach der Definition der minimalen Breite von P (der maximale Durchmesser einer Kugel, die komplett in P enthalten ist) und nach Proposition 4.3.5:

$$\overline{B}(\mathbf{0}, Md) \subseteq \frac{2Md}{\operatorname{width} P} \bullet P\left(-\frac{1}{2}\sum_{i=1}^d \boldsymbol{x}_i; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d\right),$$

also

$$\begin{aligned}
\boldsymbol{y} &\in P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) + \overline{B}(\mathbf{0}, Md) \\
&\subseteq P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) + \frac{2Md}{\operatorname{width} P} \bullet P\left(-\frac{1}{2}\sum_{i=1}^d \boldsymbol{x}_i; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d\right) \\
&= \boldsymbol{b} + P(\mathbf{0}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) \\
&\quad -\frac{1}{2}\sum_{i=1}^d \boldsymbol{x}_i + \frac{1}{2}\sum_{i=1}^d \boldsymbol{x}_i + \frac{2Md}{\operatorname{width} P}\left(P(\mathbf{0}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) - \frac{1}{2}\sum_{i=1}^d \boldsymbol{x}_i\right) \\
&= \boldsymbol{b} + \frac{1}{2}\sum_{i=1}^d \boldsymbol{x}_i + \left(1 + \frac{2Md}{\operatorname{width} P}\right)\left(P(\mathbf{0}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d) - \frac{1}{2}\sum_{i=1}^d \boldsymbol{x}_i\right) \\
&= \left(1 + \frac{2Md}{\operatorname{width} P}\right) \bullet P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d).
\end{aligned}$$

Als nächstes wird die Behauptung gezeigt. Nach Voraussetzung gibt es ein $\boldsymbol{x} \in (\boldsymbol{v} + F) \cap P$. Dieses $\boldsymbol{x}$ läßt sich schreiben als $\boldsymbol{x} = \boldsymbol{v} + \sum_{i=1}^d \alpha_i \boldsymbol{b}_i$ mit $\alpha_i \in [0,1]$. Es sei $\boldsymbol{x}' \in \boldsymbol{v} + F$,

wobei $\boldsymbol{x}' = \boldsymbol{v} + \sum_{i=1}^{d} \alpha_i' \boldsymbol{b}_i$, $\alpha_i' \in [0,1]$. Wir zeigen, daß $\boldsymbol{x}' \in \overline{B}(\boldsymbol{x}, Md) \subseteq (1 + \frac{2Md}{\operatorname{width} P}) \bullet P$ gilt. Es ist nämlich

$$\|\boldsymbol{x}' - \boldsymbol{x}\| = \left\| \sum_{i=1}^{d} (\alpha_i' - \alpha_i) \boldsymbol{b}_i \right\| \leq \sum_{i=1}^{d} |\alpha_i' - \alpha_i| \|\boldsymbol{b}_i\| \leq Md.$$

ii) Wenn man $P' := (1 - \frac{2Md}{\operatorname{width} P}) \bullet P$ setzt, dann folgt aus i) $\boldsymbol{v} + F \subseteq (1 + \frac{2Md}{\operatorname{width} P'}) \bullet P'$. Da sich mit Proposition 4.3.5 $\operatorname{width} P' = (1 - \frac{2Md}{\operatorname{width} P}) \cdot \operatorname{width} P$ ergibt, erhalten wir

$$\begin{aligned} \left(1 + \frac{2Md}{\operatorname{width} P'}\right) \bullet P' &= \left(1 + \frac{2Md}{\left(1 - \frac{2Md}{\operatorname{width} P}\right) \cdot \operatorname{width} P}\right) \left(1 - \frac{2Md}{\operatorname{width} P}\right) \bullet P \\ &= \left(1 - \frac{2Md}{\operatorname{width} P} + \frac{2Md}{\operatorname{width} P}\right) \bullet P \\ &= P. \end{aligned}$$

◇

Falls die minimale Breite des Parallelotops nicht zu klein ist und seine Kantenlänge im Vergleich zur Basislänge des Gitters (siehe Definition 4.1.9) groß ist, befindet sich in allen Translaten des Parallelotops in etwa die gleiche Anzahl von Gitterpunkten. Die Anzahl ist ungefähr proportional zum Volumen des Parallelotops. Außerdem liegen die Gitterpunkte im Parallelotop gleichmäßig verteilt: Es gibt keine affine Hyperebene, die dort hervorstechend viele Gitterpunkte enthält.

Proposition 8.1.2. Es seien $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq \mathbb{R}^d$ ein Gitter und $M := \max_{i=1,\ldots,d} \|\boldsymbol{b}_i\|$ eine Konstante, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_d \in \mathbb{R}^d$ linear unabhängige Vektoren und $\boldsymbol{b} \in \mathbb{R}^d$. Für das Parallelotop $P := P(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$ gelte $2Md < \operatorname{width} P$. Desweiteren sei H eine affine Hyperebene. Dann gelten die Ungleichungen:

i) $(1 - \frac{2Md}{\operatorname{width} P})^d \frac{\operatorname{vol} P}{\sqrt{\det L}} \leq |L \cap P| \leq (1 + \frac{2Md}{\operatorname{width} P})^d \frac{\operatorname{vol} P}{\sqrt{\det L}}$,

ii) $(1 - \frac{2Md}{\operatorname{width} P})^d \frac{\operatorname{vol} P}{\sqrt{\det L}} \leq |L \cap P^-| \leq (1 + \frac{2Md}{\operatorname{width} P})^d \frac{\operatorname{vol} P}{\sqrt{\det L}}$, mit $P^- = P^-(\boldsymbol{b}; \boldsymbol{x}_1, \ldots, \boldsymbol{x}_d)$,

iii) $|H \cap P \cap L| \leq 2Md(1 + \frac{2Md}{\operatorname{width} P})^{d-1} \frac{\operatorname{vol} \partial P}{\sqrt{\det L}}$, dabei bezeichnet $\operatorname{vol} \partial P$ das $(d-1)$-dimensionale Volumen der Oberfläche von P.

Beweis. In diesem Beweis verwenden wir die folgenden Bezeichnungen:

$$F := P(\boldsymbol{0}; \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d), \quad P' := \left(1 - \frac{2Md}{\operatorname{width} P}\right) \bullet P, \quad P'' := \left(1 + \frac{2Md}{\operatorname{width} P}\right) \bullet P.$$

i) Definiere

$$\mathcal{F} := \{\boldsymbol{v} + F : (\boldsymbol{v} + F) \cap P \neq \emptyset, \boldsymbol{v} \in L\}, \quad \mathcal{F}' := \{\boldsymbol{v} + F : (\boldsymbol{v} + F) \cap P' \neq \emptyset, \boldsymbol{v} \in L\}.$$

Nach Lemma 8.1.1 ii) gilt für $\boldsymbol{v} \in L$ mit $\boldsymbol{v} + F \in \mathcal{F}'$ auch $\boldsymbol{v} \in P$, also ist $|\mathcal{F}'| \leq |L \cap P|$. Da F ein Fundamentalbereich von L ist, gilt $P' \subseteq \bigcup_{\boldsymbol{v}+F \in \mathcal{F}'} (\boldsymbol{v} + F)$, was zur Ungleichung $\operatorname{vol} P' \leq \sum_{\boldsymbol{v}+F \in \mathcal{F}'} \operatorname{vol}(\boldsymbol{v} + F) = \sum_{\boldsymbol{v}+F \in \mathcal{F}'} \operatorname{vol}(F) = |\mathcal{F}'| \sqrt{\det L}$ führt. Dies zusammen ergibt die erste Ungleichung der Aussage.

Mit einem analogen Argument bekommt man die zweite Ungleichung der Aussage: Für $\boldsymbol{v} \in L \cap P$ gilt insbesondere $\boldsymbol{v} + F \in \mathcal{F}$, d.h. $|L \cap P| \leq |\mathcal{F}|$. Nach Lemma 8.1.1 i) ist $\boldsymbol{v} + F \subseteq P''$, d.h. $\bigcup_{\boldsymbol{v}+F \in \mathcal{F}} (\boldsymbol{v} + F) \subseteq P''$. Damit ist auch die zweite Ungleichung $|L \cap P| \leq |\mathcal{F}| \leq \operatorname{vol} P'' / \sqrt{\det L}$ bewiesen.

ii) Aus der Tatsache $P^- \subseteq P$ folgt unmittelbar $|L \cap P^-| \leq |L \cap P| \leq (1 + \frac{2Md}{\operatorname{width} P})^d \frac{\operatorname{vol} P}{\sqrt{\det L}}$.

Zur ersten Ungleichung der Aussage: Wir können aufgrund eines Stetigkeitarguments ein $\varepsilon \in (0,1)$ so wählen, daß $\lceil (1 - \frac{2Md}{\operatorname{width}(\varepsilon \bullet P)})^d \frac{\operatorname{vol}(\varepsilon \bullet P)}{\sqrt{\det L}} \rceil = \lceil (1 - \frac{2Md}{\operatorname{width} P})^d \frac{\operatorname{vol} P}{\sqrt{\det L}} \rceil$ und $\operatorname{width}(\varepsilon \bullet P) > 2Md$ gilt (Zwischenwertsatz, der linke Ausdruck (natürlich ohne die oberen GAUSS-Klammern) hängt jeweils stetig von ε ab). Nach i) ist dann

$$(1 - \tfrac{2Md}{\operatorname{width}(\varepsilon \bullet P)})^d \tfrac{\operatorname{vol}(\varepsilon \bullet P)}{\sqrt{\det L}} \leq \left\lceil (1 - \tfrac{2Md}{\operatorname{width}(\varepsilon \bullet P)})^d \tfrac{\operatorname{vol}(\varepsilon \bullet P)}{\sqrt{\det L}} \right\rceil \leq |L \cap (\varepsilon \bullet P)|.$$

Da $\varepsilon \bullet P \subseteq P^-$ und so $|L \cap (\varepsilon \bullet P)| \leq |L \cap P^-|$ ist, ergibt sich die Behauptung.

iii) Die Anzahl der Gitterpunkte in $H \cap P$ ist höchstens so groß wie die Kardinalität der Menge $\mathcal{G} := \{\boldsymbol{v} + F : (\boldsymbol{v} + F) \cap P \cap H \neq \emptyset, \boldsymbol{v} \in L\}$. Es sei $\boldsymbol{v} \in L$ ein Gitterpunkt mit $(\boldsymbol{v} + F) \cap P \cap H \neq \emptyset$. Dann ist gemäß Lemma 8.1.1 i) $\boldsymbol{v} + F \subseteq P''$. Außerdem ist $\boldsymbol{v} + F$ in einem Streifen der Breite Md um H enthalten (d.h. für ein $\boldsymbol{x} \in \boldsymbol{v} + F$ gilt $d(\boldsymbol{x}, H) \leq Md$, da der Abstand von zwei Punkten in F immer kleiner als Md ist). Das Volumen von $\bigcup_{\boldsymbol{v}+F \in \mathcal{G}} (\boldsymbol{v} + F)$ ist demnach höchstens

$$2Md \operatorname{vol}(H \cap P'') \leq 2Md \operatorname{vol} \partial P'' \leq 2Md(1 - \tfrac{2Md}{\operatorname{width} P})^{d-1} \operatorname{vol} P.$$

Also ist $|\mathcal{G}| \leq (\operatorname{vol} \bigcup_{\boldsymbol{v}+F \in \mathcal{G}} (\boldsymbol{v} + F))/\sqrt{\det L}$, und Einsetzen liefert die Behauptung.

◇

8.2 Berechnung eines Pseudowürfels

Der erste Schritt von Algorithmus 8.5.3 besteht darin, ein Parallelotop zu bestimmen, dessen Eckpunkte Gitterpunkte sind und das nahezu ein Würfel, ein sogenannter Pseudowürfel, ist. Dazu nimmt man sich einen großen Würfel Q ($Q := P(\mathbf{0}; d^3 M \boldsymbol{e}_1, \ldots, d^3 M \boldsymbol{e}_d)$, e_i bezeichnet dabei den i-ten Standardbasisvektor von $\mathbb{R}^d$, $i = 1, \ldots, d$) her und approximiert effizient seine Eckpunkte durch Gitterpunkte.

Algorithmus 8.2.1 Berechnung eines Pseudowürfels

Eingabe: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{R}^d$ linear unabhängig, $M := \max_{j=1,\ldots,d} \|\boldsymbol{b}_j\|_2$.

Ausgabe: $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d \in \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ linear unabhängig mit den in Proposition 8.2.2 beschriebenen Eigenschaften.

$B \leftarrow (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \in \mathsf{GL}_d(\mathbb{R})$ (B besitzt $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ als Spaltenvektoren).
Berechne $B^{-1} = (\beta_{ij})_{1 \leq i,j \leq d}$.
$\boldsymbol{v}_i \leftarrow \sum_{j=1}^d \lceil d^3 M \beta_{ij} \rfloor \boldsymbol{b}_j$ für $i = 1, \ldots, d$.

Proposition 8.2.2. Die Vektoren $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d$, die Algorithmus 8.2.1 bei der Eingabe von linear unabhängigen Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{R}^d$ und von $M := \max_{j=1,\ldots,d} \|\boldsymbol{b}_j\|_2$ berechnet, definieren ein d-dimensionales Parallelotop $P := P(\mathbf{0}; \boldsymbol{v}_1, \ldots, \boldsymbol{v}_d)$. Falls $d \geq 6$ ist, gilt:

i) $\|\boldsymbol{v}_i\|_2 \leq (d^3 + \frac{1}{2}d)M$ für $i = 1, \ldots, d$,

ii) $\frac{1}{3}(d^3 M)^d \leq \operatorname{vol} P \leq 3(d^3 M)^d$,

iii) $\operatorname{width} P \geq \frac{1}{3} d^3 M$,

iv) $\operatorname{vol} \partial P \leq 6d(d^3 M)^{d-1}$.

Wenn das Gitter $L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ rational ist, d.h. wenn $L \subseteq \mathbb{Q}^d$ gilt, dann ist die Laufzeit von Algorithmus 8.2.1 polynomiell in der Eingabelänge beschränkt.

Beweis. Im Gegensatz zur sonstigen Konvention $\|\cdot\| = \|\cdot\|_2$, geben wir in diesem Beweis an, welche Norm ($\|\cdot\|_2$ oder $\|\cdot\|_\infty$) wir verwenden.
Da BB^{-1} die Einheitsmatrix ist, gilt für alle $i \in \{1,\dots,d\}$: $\boldsymbol{e}_i = \sum_{j=1}^d \beta_{ij}\boldsymbol{b}_j$ und weiter

$$\begin{aligned}\|\boldsymbol{v}_i - d^3M\boldsymbol{e}_i\|_2 &= \left\|\sum_{j=1}^d \lceil d^3M\beta_{ij}\rfloor \boldsymbol{b}_j - \sum_{j=1}^d d^3M\beta_{ij}\boldsymbol{b}_j\right\|_2 \\ &\le \sum_{j=1}^d \left|\lceil d^3M\beta_{ij}\rfloor - d^3M\beta_{ij}\right| \cdot \max_{j=1,\dots,d}\|\boldsymbol{b}_j\|_2 \\ &\le \frac{1}{2}dM. \end{aligned} \tag{8.1}$$

Unmittelbar folgt $\|\boldsymbol{v}_i\|_2 = \|\boldsymbol{v}_i - d^3M\boldsymbol{e}_i + d^3M\boldsymbol{e}_i\|_2 \le d^3M + \frac{1}{2}dM = \left(d^3 + \frac{1}{2}d\right)M$.
Außerdem läßt sich mit Ungleichung (8.1) erkennen, daß $\boldsymbol{v}_1,\dots,\boldsymbol{v}_d$ linear unabhängig sind und damit P ein d-dimensionales Parallelotop ist: Angenommen $\boldsymbol{v}_1,\dots,\boldsymbol{v}_d$ sind linear abhängig und ohne Einschränkung der Allgemeinheit sei $\boldsymbol{v}_1 = \sum_{i=2}^d \alpha_i\boldsymbol{v}_i$ eine Linearkombination der $\boldsymbol{v}_2,\dots,\boldsymbol{v}_d$. Es gilt (man beachte die Ungleichung $\|\cdot\|_\infty \le \|\cdot\|_2$)

$$\begin{aligned}\max_{i=2,\dots,d}\{1,|\alpha_i|\}\cdot d^3M &= \left\|\sum_{i=2}^d \alpha_i d^3M\boldsymbol{e}_i - d^3M\boldsymbol{e}_1\right\|_\infty \\ &= \left\|\sum_{i=2}^d \alpha_i d^3M\boldsymbol{e}_i - \sum_{i=2}^d \alpha_i\boldsymbol{v}_i + \boldsymbol{v}_1 - d^3M\boldsymbol{e}_1\right\|_\infty \\ &\le \left\|\sum_{i=2}^d \alpha_i(d^3M\boldsymbol{e}_i - \boldsymbol{v}_i)\right\|_2 + \left\|\boldsymbol{v}_1 - d^3M\boldsymbol{e}_1\right\|_2 \\ &\le \sum_{i=2}^d |\alpha_i|\|d^3M\boldsymbol{e}_i - \boldsymbol{v}_i\|_2 + \|\boldsymbol{v}_1 - d^3M\boldsymbol{e}_1\|_2. \end{aligned}$$

Mit (8.1) folgt

$$\begin{aligned}\sum_{i=2}^d |\alpha_i|\|d^3M\boldsymbol{e}_i - \boldsymbol{v}_i\|_2 + \|\boldsymbol{v}_1 - d^3M\boldsymbol{e}_1\|_2 &\le (d-1)\cdot\max_{i=2,\dots,d}\{|\alpha_i|\}\cdot\frac{1}{2}dM + \frac{1}{2}dM \\ &\le \max_{i=2,\dots,d}\{1,|\alpha_i|\}\cdot d^2M,\end{aligned}$$

was im Widerspruch zu $d \ge 2$ steht.

Mit Hilfe von Ungleichung (8.1) können wir auch beweisen, daß die Eckpunkte von P nahe an den entsprechenden Eckpunkten des Würfels $Q := P(0; d^3M\boldsymbol{e}_1,\dots,d^3M\boldsymbol{e}_d)$ liegen. Es sei $\boldsymbol{x} = \sum_{i=1}^d \varepsilon_i(d^3M\boldsymbol{e}_i)$, $\boldsymbol{\varepsilon} \in \{0,1\}^d$, ein Eckpunkt von Q und $\boldsymbol{v} = \sum_{i=1}^d \varepsilon_i\boldsymbol{v}_i$ der entsprechende Eckpunkt von P. Dann gilt die Ungleichung

$$\|\boldsymbol{v} - \boldsymbol{x}\|_2 = \left\|\sum_{i=1}^d \varepsilon_i(\boldsymbol{v}_i - d^3M\boldsymbol{e}_i)\right\|_2 \le \sum_{i=1}^d \|\boldsymbol{v}_i - d^3M\boldsymbol{e}_i\|_2 \le d\left(\frac{1}{2}dM\right) = \frac{1}{2}d^2M.$$

Ausgehend von Q definiere die Parallelotope $Q' := (1-\frac{1}{d})\bullet Q$ und $Q'' := (1+\frac{1}{d})\bullet Q$. Offensichtlich gelten für Q' und Q'' die Gleichungen $\operatorname{vol} Q' = (1-\frac{1}{d})^d(d^3M)^d$, $\operatorname{vol} Q'' = (1+\frac{1}{d})^d(d^3M)^d$

und $\operatorname{width} Q' = \frac{1}{2}(1-\frac{1}{d})d^3M$. Im folgenden werden die Inklusionen $Q' \subseteq P \subseteq Q''$ gezeigt, so daß die Aussagen ii) und iii) direkt folgen, da für $d \geq 6$ stets $\frac{1}{3} \leq (1-\frac{1}{d})^d$, $(1+\frac{1}{d})^d \leq 3$ und $\frac{1}{2}(1-\frac{1}{d}) \geq \frac{1}{3}$ gilt. Die geometrische Situation ist in Abbildung 8.1 dargestellt.

$\underline{P \subseteq Q''}$: Wir zeigen zuerst, daß alle Eckpunkte von P in Q'' liegen. Es genügt zu zeigen, daß für die Eckpunkte $\boldsymbol{v}$ von P die Ungleichung $\|\boldsymbol{v} - \frac{1}{2}\sum_{i=1}^{d} d^3 M \boldsymbol{e}_i\|_\infty \leq \frac{1}{2}(1+\frac{1}{d})d^3M$ gilt, da sich Q'' schreiben läßt als $Q'' = \{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x} - \frac{1}{2}\sum_{i=1}^{d} d^3 M \boldsymbol{e}_i\|_\infty \leq \frac{1}{2}(1+\frac{1}{d})d^3M\}$.
Es sei $\boldsymbol{v}$ ein Eckpunkt von P und $\boldsymbol{x}$ der entsprechende Eckpunkt von Q. Es gilt die gewünschte Ungleichung ($\|\cdot\|_\infty \leq \|\cdot\|_2$):

$$\begin{aligned}
\left\| \boldsymbol{v} - \frac{1}{2}\sum_{i=1}^{d} d^3 M \boldsymbol{e}_i \right\|_\infty &= \left\| \boldsymbol{v} - \boldsymbol{x} + \boldsymbol{x} - \frac{1}{2}\sum_{i=1}^{d} d^3 M \boldsymbol{e}_i \right\|_\infty \\
&\leq \|\boldsymbol{v} - \boldsymbol{x}\|_2 + \left\| \boldsymbol{x} - \frac{1}{2}\sum_{i=1}^{d} d^3 M \boldsymbol{e}_i \right\|_\infty \\
&\leq \frac{1}{2}d^2M + \frac{1}{2}d^3M \\
&= \frac{1}{2}\left(1+\frac{1}{d}\right)d^3M.
\end{aligned}$$

Weil alle Punkte von P eine konvexe Linearkombination der Eckpunkte von P sind und Q'' konvex ist, folgt $P \subseteq Q''$.

$\underline{Q' \subseteq P}$: Die Kantenlänge des Würfels Q beträgt d^3M, die von Q' ist hingegen nur $(1-\frac{1}{d})d^3M = d^3M - d^2M$. Jeder Eckpunkt $\boldsymbol{x}$ von P liegt in einem Würfel der Kantenlänge d^2M (sogar in einer in diesem Würfel liegenden Kugel mit Durchmesser d^2M), der in einem Eckpunkt außen an den Würfel Q' in dem zu $\boldsymbol{x}$ entsprechenden Eckpunkt stößt (siehe Abbildung 8.1). Es liegt somit jeder Eckpunkt von Q' in der konvexen Hülle der Eckpunkte von P. Da sowohl Q' als auch P konvex sind, folgt $Q' \subseteq P$.

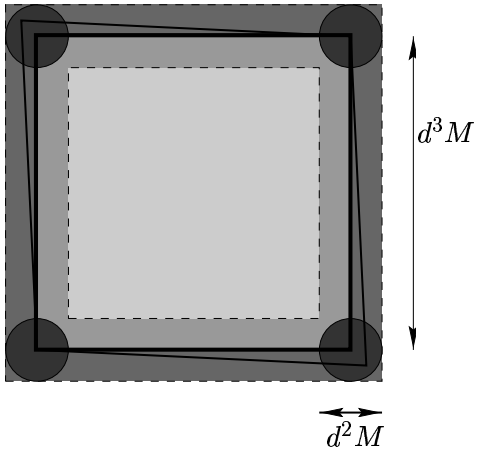

Abbildung 8.1: Geometrische Lage des in Algorithmus 8.2.1 konstruierten Pseudowürfels.

Um das $(d-1)$-dimensionale Volumen der Oberfläche von P abzuschätzen, schätzen wir zuerst das $(d-1)$-dimensionale Volumen der $2d$ Wände von P ab. Jede Wand von P besitzt das gleiche Volumen und genau $2(d-1)$ Kanten, die alle Kopien der Strecken $[0, \boldsymbol{v}_i]$, $i = 1, \ldots, d$, sind. Keine Strecke ist länger als $(d^3 + \frac{1}{2}d)M$. Daraus folgt, daß das $(d-1)$-dimensionale Volumen einer Wand von P höchstens $((d^3 + \frac{1}{2}d)M)^{d-1}$ beträgt und damit

$$\operatorname{vol} \partial P \leq 2d((d^3 + \tfrac{1}{2}d)M)^{d-1} = 2d(d^3M)^{d-1}(1 + \tfrac{1}{2d^2})^{d-1} \leq 6d(d^3M)^{d-1}.$$

$\diamond$

8.3 Zufällige Gitterpunktwahl in einem Parallelotop

Später wird ein Algorithmus benötigt der Gitterpunkte, die im Pseudowürfel liegen, zufällig gemäß der Gleichverteilung wählen kann. Ein probabilistischer Algorithmus hat nur Zugriff auf eine zufällige Bitfolge, bzw. auf zufällige ganze Zahlen. Es muß also ein Algorithmus gefunden werden, der die zufällige Bitfolge auf einen zufälligen Gitterpunkt im Pseudowürfel abbildet. Dies geschieht in Algorithmus 8.3.1, der die Gleichverteilung der nur knapp verfehlt.

Algorithmus 8.3.1 Zufällige Gitterpunktwahl in einem Parallelotop

Eingabe: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Z}^d$ linear unabhängig und $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d \in L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ linear unabhängige Gittervektoren mit $\max_{i=1,\ldots,d}\{\|\boldsymbol{b}_i\|, \|\boldsymbol{v}_i\|\} \leq 2^{d^\varepsilon}$ für ein $\varepsilon > 0$.
Es sei $P := P^-(\boldsymbol{0}; \boldsymbol{v}_1, \ldots, \boldsymbol{v}_d)$.

Ausgabe: $\boldsymbol{v} \in P \cap L$ mit $\sum_{\boldsymbol{v}' \in P \cap L} \left| \Pr[\boldsymbol{v}' = \boldsymbol{v}] - \frac{1}{|P \cap L|} \right| \leq 2^{-d^\varepsilon}$.
Wähle δ in Abhängigkeit von ε genügend groß (siehe Sublemma 8.3.3).
Wähle $\alpha_1, \ldots, \alpha_d \in \{0, 1, \ldots, 2^{d^\delta} - 1\}$ unabhängig voneinander, zufällig gemäß der Gleichverteilung.
$\boldsymbol{w} \leftarrow \sum_{i=1}^d \alpha_i \boldsymbol{b}_i$.
Wähle $\boldsymbol{w}' \in L$, so daß $\boldsymbol{w} - \boldsymbol{w}' \in P$ gilt (siehe Algorithmus 4.3.6).
$\boldsymbol{v} \leftarrow \boldsymbol{w} - \boldsymbol{w}'$.

Proposition 8.3.2. Der Algorithmus 8.3.1 ist korrekt: Er berechnet bei der Eingabe von linear unabhängigen Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Z}^d$ und linear unabhängigen Gittervektoren $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d \in L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \boldsymbol{b}_d$ mit $\max_{i=1,\ldots,d}\{\|\boldsymbol{b}_i\|, \|\boldsymbol{v}_i\|\} \leq 2^{d^\varepsilon}$ einen Gittervektor $\boldsymbol{v} \in L \cap P$, so daß $\sum_{\boldsymbol{v}' \in P \cap L} \left| \Pr[\boldsymbol{v}' = \boldsymbol{v}] - \frac{1}{|P \cap L|} \right| \leq 2^{-d^\varepsilon}$ gilt. Seine Laufzeit ist polynomiell in der Eingabelänge beschränkt.

Beweis. Setze $P' := P^-(0; 2^{d^\delta}\boldsymbol{b}_1, \ldots, 2^{d^\delta}\boldsymbol{b}_d)$. Offensichtlich ist der Gitterpunkt $\boldsymbol{w}$ zufällig gemäß der Gleichverteilung aus der Menge $P' \cap L$ gewählt. Offensichtlich ist $|P' \cap L| = 2^{d^{\delta+1}}$. Definiere die Mengen

$$\mathcal{F} := \{\boldsymbol{v} + P : \boldsymbol{v} + P \subseteq P', \boldsymbol{v} \in L\}, \quad \mathcal{F}' := \{\boldsymbol{v} + P : (\boldsymbol{v} + P) \cap P' \neq \emptyset, \boldsymbol{v} \in L\},$$

außerdem definiere das Ereignis E : „$\boldsymbol{w} \in \bigcup_{\boldsymbol{v}+P \in \mathcal{F}}(\boldsymbol{v} + P)$“. Unter der Bedingung, daß Ereignis E eingetreten ist, gilt — aufgrund der Gleichverteilung von $\boldsymbol{w}$ — daß $\boldsymbol{v}$ in $P \cap L$ gleichverteilt ist, d.h. für alle $\boldsymbol{v}' \in P \cap L$ gilt $\Pr[\boldsymbol{v} = \boldsymbol{v}' \mid E] = \frac{1}{|P \cap L|}$. Damit später der Satz von der totalen Wahrscheinlichkeit angewendet werden kann, ist eine obere Schranke für die Wahrscheinlichkeit

$$\Pr[\neg E] = \frac{\left| \bigcup_{\boldsymbol{v}+P \in \mathcal{F}' \setminus \mathcal{F}} (\boldsymbol{v} + P) \cap L \right|}{|P' \cap L|}$$

interessant, die das folgende Sublemma liefert.

Sublemma 8.3.3. Es gilt die Ungleichung

$$\left| \bigcup_{\boldsymbol{v}+P \in \mathcal{F}' \setminus \mathcal{F}} (\boldsymbol{v}+P) \cap L \right| \leq 2^{-2d^{\varepsilon+1}} 2^{d^{\delta+1}}.$$

Beweis. (von Sublemma 8.3.3)
Dieser Beweis kann mit den üblichen Methoden (Anwendung von Lemma 8.1.1, Proposition 8.1.2) und einiger mühseliger Rechnerei geführt werden (siehe [Ajt96], Lemma 8). Es muß δ in Abhängigkeit von ε so groß gewählt werden, daß die Ungleichung richtig ist. Da der einzige Erkenntnisgewinn die explizite Bestimmung von δ wäre, und δ nicht erneut explizit auftritt, verzichten wir hier auf die Details. Aus AJTAIs Beweis läßt sich ablesen, daß δ polynomiell von ε abhängt. ◇

Mit Sublemma 8.3.3 folgt $\Pr[\neg E] \leq \frac{2^{-2d^{\varepsilon+1}} 2^{d^{\delta+1}}}{2^{d^{\delta+1}}} = 2^{-2d^{\varepsilon+1}}$. Die behauptete Ungleichung betrachten wir jetzt summandenweise: für alle $\boldsymbol{v}' \in P \cap L$ gilt

$$\begin{aligned}
& \left|\Pr[\boldsymbol{v}' = \boldsymbol{v}] - \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid E]\right| \\
= \; & \left|(\Pr[E] \cdot \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid E]) + (\Pr[\neg E] \cdot \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid \neg E]) - \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid E]\right| \\
= \; & \left|((\Pr[E] - 1) \cdot \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid E]) + (\Pr[\neg E] \cdot \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid \neg E])\right| \\
= \; & \left|\Pr[\neg E] \cdot (\Pr[\boldsymbol{v}' = \boldsymbol{v} \mid \neg E] - \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid E])\right| \\
\leq \; & \Pr[\neg E].
\end{aligned}$$

Da $L \subseteq \mathbb{Z}^d$ ist, ist $\det L \in \mathbb{N}$ und es ergibt sich $|P \cap L| \leq \frac{\operatorname{vol} P}{\sqrt{\det L}} \leq \operatorname{vol} P \leq \prod_{i=1}^{d} \|\boldsymbol{v}_i\| \leq 2^{d^{\varepsilon+1}}$ und die Behauptung folgt:

$$\sum_{\boldsymbol{v}' \in P \cap L} |\Pr[\boldsymbol{v}' = \boldsymbol{v}] - \Pr[\boldsymbol{v}' = \boldsymbol{v} \mid E]| \leq \sum_{\boldsymbol{v}' \in P \cap L} 2^{-2d^{\varepsilon+1}} \leq 2^{-d^{\varepsilon+1}} \leq 2^{-d^{\varepsilon}}.$$

◇

8.4 Unterteilung des Pseudowürfels

Es sei q eine natürliche Zahl. Der Pseudowürfel $P^-(\mathbf{0}; \boldsymbol{v}_1, \ldots, \boldsymbol{v}_d)$ wird in q^d gleich große, paarweise disjunkte Teilpseudowürfel $P^-_{\boldsymbol{x}}$, $\boldsymbol{x} \in (\mathbb{Z}/q\mathbb{Z})^d$, unterteilt. Ein Vektor $\boldsymbol{x} \in (\mathbb{Z}/q\mathbb{Z})^d$ wird im folgenden als Vektor im $\mathbb{Z}^d$ mit Koeffizienten aus der Menge $\{0, \ldots, q-1\}$ identifiziert. Der Punkt $\boldsymbol{O}_{\boldsymbol{x}} := \sum_{i=1}^{d} \frac{\boldsymbol{x}_i}{q} \boldsymbol{v}_i$ ist der Ursprung von $P^-_{\boldsymbol{x}} := P^-(\boldsymbol{O}_{\boldsymbol{x}}; \frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_d)$. Entsprechend wird $P_{\boldsymbol{x}}$ definiert.

Intuitiv besagt die nachfolgende Proposition: Wenn man Gitterpunkte zufällig gleichverteilt in $P^-(\mathbf{0}; \boldsymbol{v}_1, \ldots, \boldsymbol{v}_d)$ wählen kann, dann kann man auch zuerst zufällig gleichverteilt ein $\boldsymbol{x}$ aus $(\mathbb{Z}/q\mathbb{Z})^d$ und danach zufällig gleichverteilt einen Gitterpunkt in $P^-_{\boldsymbol{x}}$ wählen.

Diese Aussage erfordert einen Beweis, da nicht alle Teilpseudowürfel die gleiche Anzahl von Gitterpunkten enthalten.

Proposition 8.4.1. Es seien $q \in \mathbb{N}$, $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subseteq \mathbb{R}^d$ ein d-dimensionales Gitter, $M := \max_{i=1,\ldots,d} \|\boldsymbol{b}_i\|$ eine Konstante und $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d \in L$ linear unabhängige Gittervektoren. Für das Parallelotop $P := P^-(\mathbf{0}; \boldsymbol{v}_1, \ldots, \boldsymbol{v}_d)$ gelten die Ungleichungen $\frac{qM}{\operatorname{width} P} < \frac{1}{12d^4}$ und $4Mdq \operatorname{vol} \partial P < (\frac{1}{2} - \frac{1}{d^2}) \operatorname{vol} P$. Falls sich $\boldsymbol{v}$ zufällig gemäß der Gleichverteilung aus der Menge $L \cap P$ wählen läßt, dann läßt sich $\boldsymbol{x} \in (\mathbb{Z}/q\mathbb{Z})^d$ in Abhängigkeit von $\boldsymbol{v}$ so wählen, daß gilt:

i) Für alle $\boldsymbol{y} \in (\mathbb{Z}/q\mathbb{Z})^d$ ist $\Pr[\boldsymbol{x} = \boldsymbol{y}] = \frac{1}{q^d}$.

ii) $\Pr[\boldsymbol{v} \in P_{\boldsymbol{x}}^-] > 1 - \frac{1}{d^2}$.

iii) Für jede affine Hyperebene $H \subseteq \mathbb{R}^d$ ist $\Pr[\boldsymbol{v} - \boldsymbol{O}_{\boldsymbol{x}} \in H] < \frac{1}{2}$.

Beweis. Nach Proposition 8.1.2 ii)gilt für jedes $\boldsymbol{x} \in (\mathbb{Z}/q\mathbb{Z})^d$

$$\left(1 - \frac{2Md}{\operatorname{width} P_{\boldsymbol{x}}^-}\right)^d \frac{\operatorname{vol} P_{\boldsymbol{x}}^-}{\sqrt{\det L}} \leq |L \cap P_{\boldsymbol{x}}^-| \leq \left(1 + \frac{2Md}{\operatorname{width} P_{\boldsymbol{x}}^-}\right)^d \frac{\operatorname{vol} P_{\boldsymbol{x}}^-}{\sqrt{\det L}}.$$

Da $\frac{M}{\operatorname{width} P_{\boldsymbol{x}}^-} = \frac{qM}{\operatorname{width} P} \leq \frac{1}{12d^4}$ ist, folgt

$$\left(1 + \frac{2Md}{\operatorname{width} P_{\boldsymbol{x}}^-}\right)^d \leq \left(1 + \frac{1}{6d^3}\right)^d \leq e^{\frac{1}{6d^2}},$$

dabei wurde die Ungleichung $(1 + \frac{t}{n})^n \leq e^t$ mit $n = d$, $t = \frac{1}{6d^2}$ (siehe [MR95], Proposition B.3) verwendet. Logarithmieren ergibt (die Abbildung $x \mapsto \ln(1 + x)$ ist konkav):

$$\frac{1}{6d^2} \leq (1 - \frac{1}{6d^2}) \ln(1 + 0) + \frac{1}{6d^2} \ln(1 + 2) \leq \ln\left(1 + 1(-\frac{1}{6d^2}) \cdot 0 + \frac{1}{6d^2} \cdot 2\right) = \ln(1 + \frac{1}{3d^2}).$$

Also gilt $(1 + \frac{2Md}{\operatorname{width} P_{\boldsymbol{x}}^-})^d \leq 1 + \frac{1}{3d^2}$.

Da $\frac{M}{\operatorname{width} P_{\boldsymbol{x}}^-} = \frac{qM}{\operatorname{width} P} \leq \frac{1}{12d^4}$ ist, folgt

$$\left(1 - \frac{2Md}{\operatorname{width} P_{\boldsymbol{x}}^-}\right)^d \geq 1 - \frac{1}{3d^2} \geq \left(1 - \frac{1}{6d^3}\right)^d \geq e^{-\frac{1}{6d^2}} \left(1 - \frac{1}{36d^5}\right),$$

dabei wurde die Ungleichung $(1 + \frac{t}{n})^n \geq e^t(1 - \frac{t^2}{n})$ (siehe [MR95], Proposition B.3) mit $n = d$ und $t = -\frac{1}{6d^2}$ verwendet. Weiter ergibt sich mit der meistunterschätzten Ungleichung der Analysis $e^t \geq 1 + t$:

$$e^{-\frac{1}{6d^2}} \left(1 - \tfrac{1}{36d^5}\right) \geq \left(1 - \tfrac{1}{6d^2}\right) \left(1 - \tfrac{1}{36d^5}\right) \geq 1 - \tfrac{1}{3d^2}.$$

Also gilt $(1 - \frac{2Md}{\operatorname{width} P_{\boldsymbol{x}}^-})^d \geq 1 - \frac{1}{3d^2}$.

Demnach enthält jeder Teilpseudowürfel wenigstens $\alpha := \left\lceil (1 - \frac{1}{3d^2}) \frac{1}{q^d} \frac{\operatorname{vol} P}{\sqrt{\det L}} \right\rceil$ Gitterpunkte. Für jeden Vektor $\boldsymbol{x} \in (\mathbb{Z}/q\mathbb{Z})^d$ bestimme irgendeine Teilmenge $P'_{\boldsymbol{x}}$ von $P_{\boldsymbol{x}}^-$, die aus genau α Gitterpunkten besteht. Wähle zufällig gemäß der Gleichverteilung $\boldsymbol{z}$ aus der Menge $(\mathbb{Z}/q\mathbb{Z})^d$. Falls $\boldsymbol{v} \in \bigcup_{\boldsymbol{y} \in (\mathbb{Z}/q\mathbb{Z})^d} P'_{\boldsymbol{y}}$ ist, setze $\boldsymbol{x}$ so, daß $\boldsymbol{v} \in P'_{\boldsymbol{x}}$ ist. Ansonsten setze $\boldsymbol{x} = \boldsymbol{z}$. Da die Wahl von $\boldsymbol{z}$ unabhängig von der Wahl von $\boldsymbol{v}$ geschieht und alle $P'_{\boldsymbol{y}}$, $\boldsymbol{y} \in (\mathbb{Z}/q\mathbb{Z})^d$, dieselbe Kardinalität besitzen, folgt die erste Behauptung.

Für den Beweis der zweiten Behauptung stellen wir fest

$$\Pr[\boldsymbol{v} \in P_{\boldsymbol{x}}^-] \geq \Pr[\boldsymbol{v} \in P'_{\boldsymbol{x}}] \geq \frac{\left(1 - \frac{1}{3d^2}\right) \frac{\operatorname{vol} P'_{\boldsymbol{x}}}{\sqrt{\det L}}}{\left(1 + \frac{1}{3d^2}\right) \frac{\operatorname{vol} P'_{\boldsymbol{x}}}{\sqrt{\det L}}} = \frac{1 - \frac{1}{3d^2}}{1 + \frac{1}{3d^2}} \geq 1 - \frac{1}{d^2}.$$

Aus dieser Ungleichung folgt u.a. $\Pr[\boldsymbol{v} \notin P'_{\boldsymbol{x}}] \leq \frac{1}{d^2}$, was für den Beweis der dritten Behauptung hilfreich ist. Da

$$\begin{aligned}
\Pr[\boldsymbol{v} - \boldsymbol{O}_{\boldsymbol{x}} \in H] &= \Pr[\boldsymbol{v} \in P'_{\boldsymbol{x}}] \cdot \Pr[\boldsymbol{v} - \boldsymbol{O}_{\boldsymbol{x}} \in H \mid \boldsymbol{v} \in P'_{\boldsymbol{x}}] + \\
&\quad \Pr[\boldsymbol{v} \notin P'_{\boldsymbol{x}}] \cdot \Pr[\boldsymbol{v} - \boldsymbol{O}_{\boldsymbol{x}} \in H \mid \boldsymbol{v} \notin P'_{\boldsymbol{x}}] \\
&\leq \Pr[\boldsymbol{v} \in P'_{\boldsymbol{x}}] \cdot \Pr[\boldsymbol{v} - \boldsymbol{O}_{\boldsymbol{x}} \in H \mid \boldsymbol{v} \in P'_{\boldsymbol{x}}] + \frac{1}{d^2} \cdot 1 \\
&\leq \Pr[\boldsymbol{v} - \boldsymbol{O}_{\boldsymbol{x}} \in H | \boldsymbol{v} \in P'_{\boldsymbol{x}}] + \frac{1}{d^2}
\end{aligned}$$

gilt, genügt es, $\Pr[\boldsymbol{v} \in \boldsymbol{O}_{\boldsymbol{x}} + H \mid \boldsymbol{v} \in P'_{\boldsymbol{x}}] < \frac{1}{2} - \frac{1}{d^2}$ einzusehen, um die dritte Behauptung zu beweisen. Unter der Bedingung $\boldsymbol{v} \in P'_{\boldsymbol{x}}$ ist $\boldsymbol{v}$ zufällig gemäß der Gleichverteilung aus $P'_{\boldsymbol{x}}$ gewählt. Dann erhalten wir mit Proposition 8.1.2

$$\begin{aligned}
\Pr[\boldsymbol{v} \in \boldsymbol{O}_{\boldsymbol{x}} + H \mid \boldsymbol{v} \in P'_{\boldsymbol{x}}] &\leq \frac{|(\boldsymbol{O}_{\boldsymbol{x}} + H) \cap P'_{\boldsymbol{x}} \cap L|}{|P'_{\boldsymbol{x}}|} \\
&\leq \frac{2Md\left(1 + \frac{2Mdq}{\operatorname{width} P}\right)^{d-1} \frac{1}{q^{d-1}} \frac{\operatorname{vol} \partial P}{\sqrt{\det L}}}{\left(1 - \frac{1}{3d^2}\right) \frac{1}{q^d} \frac{\operatorname{vol} P}{\sqrt{\det L}}} \\
&\leq 2Mdq \cdot \frac{\left(1 + \frac{1}{3d^2}\right)}{\left(1 - \frac{1}{3d^2}\right)} \cdot \frac{\operatorname{vol} \partial P}{\operatorname{vol} P} \\
&\leq 4Mdq \cdot \frac{\operatorname{vol} \partial P}{\operatorname{vol} P}.
\end{aligned}$$

Und nach Voraussetzung ist $4Mdq \cdot \frac{\operatorname{vol} \partial P}{\operatorname{vol} P} < \frac{1}{2} - \frac{1}{d^2}$. ◇

8.5 Das Theorem der Worst-Case/Average-Case-Äquivalenz

Im folgenden sei d eine genügend große natürliche Zahl. Desweiteren seien die natürlichen Zahlen $q := d^6$ und $n := \lceil 2d \log_2 q \rceil$ in Abhängigkeit von d gewählt. Diese spezielle Wahl werden wir im Anschluß von Lemma 8.5.2 diskutieren.

Theorem 8.5.1. (*Das Theorem der Worst-Case/Average-Case-Äquivalenz*)
Angenommen es gibt eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M und ein Polynom $p \in \mathbb{R}[X]$, so daß die Wahrscheinlichkeit

$$\Pr[M \text{ berechnet aus } A \in (\mathbb{Z}/q\mathbb{Z})^{d \times n} \text{ ein } \boldsymbol{x} \in \{-1, 0, 1\}^n \backslash \{\mathbf{0}\} \text{ mit } A\boldsymbol{x} = \mathbf{0}] > \frac{1}{|p(d)|}$$

ist. Dabei wird die Wahrscheinlichkeit über die zufällige Wahl von $A \in (\mathbb{Z}/q\mathbb{Z})^{d \times n}$ und über die Münzwürfe von M genommen. Dann ist Gap-$(1, 72d^8\sqrt{(d+3)/4})$-SVP $\in \mathcal{RP}$, d.h. dann gibt es einen effizienten Algorithmus, der Gitterminima bis auf einen Faktor von $O(d^9)$ approximiert.

Der Beweis dieses Theorems geschieht in drei Schritten. Im ersten Schritt wird eine TURING-Maschine M_1 angegeben, die ein System von linear unabhängigen Gittervektoren in ein anderes transformiert, so daß die Länge des längsten Gittervektors des Ursprungsystems mindestens halbiert wird. Im zweiten Schritt wird eine TURING-Maschine M_2 angegeben, die ein System von linear unabhängigen Gittervektoren in eines transformiert, dessen Vektoren nicht viel länger als die Basislänge des Eingabegitters ist. Im dritten Schritt wird mit Hilfe der Approximation der Basislänge (siehe Definition 4.1.9, Notation: bl) von M_2 eine Approximation des Minimums des Eingabegitters berechnet.

Lemma 8.5.2. Angenommen es gibt ein Polynom $p \in \mathbb{R}[X]$ und eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M, so daß die Wahrscheinlichkeit

$$\Pr[M \text{ berechnet aus } A \in (\mathbb{Z}/q\mathbb{Z})^{d\times n} \text{ ein } \boldsymbol{x} \in \{-1,0,1\}^n\backslash\{\mathbf{0}\} \text{ mit } A\boldsymbol{x} = \mathbf{0}] > \frac{1}{|p(d)|}$$

ist. Dabei wird die Wahrscheinlichkeit über die zufällige Wahl von $A \in (\mathbb{Z}/q\mathbb{Z})^{d\times n}$ und über die Münzwürfe von M genommen. Dann gibt es eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M_1, die bei der Eingabe von linear unabhängigen Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$ und linear unabhängigen Gittervektoren $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d \in L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ mit $\|\boldsymbol{v}_1\| \leq \ldots \leq \|\boldsymbol{v}_d\| =: S$ und $36d^7\,\mathrm{bl}(L) \leq S$ einen Gittervektor $\boldsymbol{v} \in L$ berechnet, so daß die Wahrscheinlichkeit dafür, daß $\|\boldsymbol{v}\| \leq \frac{S}{2}$ gilt und daß $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d-1}, \boldsymbol{v}$ linear unabhängig sind, mindestens $\frac{1}{2}$ beträgt.

Beweis. Im ersten Schritt beschreiben wir einen Algorithmus, der für die Berechnung der TURING-Maschine M_1 herangezogen werden kann. Der einzige Haken an diesem Algorithmus ist, daß seine Erfolgswahrscheinlichkeit zu niedrig ist. Durch polynomiell viele Wiederholungen einiger seiner Schritte wird dies später behoben.
Die im Algorithmus 8.5.3 vorkommenden Bezeichnungen sind an die Bezeichnungen von Abschnitt 8.4 angelehnt. Algorithmus 8.2.1 berechnet aus den Vektoren $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d$ einen Pseudowürfel $P := P^-(\mathbf{0}; \boldsymbol{w}_1, \ldots, \boldsymbol{w}_d)$. Für $\boldsymbol{a}_i \in (\mathbb{Z}/q\mathbb{Z})^d$ ist $\boldsymbol{O}_{\boldsymbol{a}_i} := \sum_{i=j}^d a_{ij}\boldsymbol{w}_j$ der Ursprung des Teilpseudowürfels $P^-_{\boldsymbol{a}_i} := P^-(\boldsymbol{O}_{\boldsymbol{a}_i}; \frac{1}{q}\boldsymbol{w}_1, \ldots, \frac{1}{q}\boldsymbol{w}_d)$.
Wir gehen im folgenden davon aus, daß Algorithmus 8.3.1 Gittervektoren $\boldsymbol{u}_i$, $i = 1, \ldots, n$, unabhängig voneinander und zufällig gemäß der Gleichverteilung aus dem Pseudowürfel P wählt. Nach Proposition 8.3.2 trifft unsere Annahme mit einer Wahrscheinlichkeit von mindestens 2^{-d^ε} für ein fest gewähltes $\varepsilon \in \mathbb{R}_{>0}$ mit $\max\{\|\boldsymbol{b}_i\|, \|\boldsymbol{v}_i\| : i = 1, \ldots, d\} \leq 2^{d^\varepsilon}$ zu.

Algorithmus 8.5.3

Eingabe: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$ linear unabhängig, $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d \in L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ linear unabhängig mit $\|\boldsymbol{v}_1\| \leq \ldots \leq \|\boldsymbol{v}_d\| =: S$ und $36d^7\,\mathrm{bl}(L) \leq S$.

Ausgabe: (probabilistisch, mit möglichem Irrtum:) $\boldsymbol{v} \in L$ mit $\|\boldsymbol{v}\| \leq S/2$ und $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d-1}, \boldsymbol{v}$ linear unabhängig.

i) Anwendung von Algorithmus 8.2.1 bei Eingabe von den Vektoren $(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d)$ liefert linear unabhängige $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_d \in \mathbb{Z}\boldsymbol{v}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{v}_d$. Es sei $P := P^-(\mathbf{0}; \boldsymbol{w}_1, \ldots, \boldsymbol{w}_d)$ der zugehörige Pseudowürfel.

ii) Ein n-maliges Wiederholen von Algorithmus 8.3.1 bei der Eingabe von den Vektoren $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ und $(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_d)$ liefert zufällige Gittervektoren $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n \in P \cap L$.

iii) Bestimme $\boldsymbol{a}_i \in (\mathbb{Z}/q\mathbb{Z})^d$, so daß $\boldsymbol{u}_i \in P^-_{\boldsymbol{a}_i}$, $i = 1, \ldots, n$, durch Lösen des entsprechenden linearen Gleichungssystems.

iv) $A \leftarrow (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \in (\mathbb{Z}/q\mathbb{Z})^{d\times n}$.

v) Anwendung von M mit Eingabe A liefert $\boldsymbol{x} \in \{-1,0,1\}^n\backslash\{\mathbf{0}\}$.

vi) $\boldsymbol{v} \leftarrow \sum_{i=1}^n x_i\boldsymbol{u}_i - \sum_{i=1}^n x_i\boldsymbol{O}_{a_i}$.

Nun soll Proposition 8.4.1 angewendet werden. Dafür müssen die Ungleichungen $\frac{q\,\mathrm{bl}(L)}{\mathrm{width}\,P} \leq \frac{1}{12d^4}$ und $4\,\mathrm{bl}(L)dq\,\mathrm{vol}\,\partial P < (\frac{1}{2} - \frac{1}{d^2})\,\mathrm{vol}\,P$ erfüllt sein. Minimale Breite und Volumen von P können

mit Hilfe von Proposition 8.2.2 abgeschätzt werden. Wir erhalten für alle $d \in \mathbb{N}$

$$\frac{q \operatorname{bl}(L)}{\operatorname{width} P} \leq \frac{d^6 \operatorname{bl}(L)}{\frac{1}{3} d^3 S} = \frac{3d^3}{S} \operatorname{bl}(L) \leq \frac{1}{12d^4} \iff 36d^7 \operatorname{bl}(L) \leq S$$

und

$$4 \operatorname{bl}(L) dq \operatorname{vol} \partial P \leq 4 \operatorname{bl}(L) dq (6d(d^3 S)^{d-1}) \leq \left(\frac{1}{2} - \frac{1}{d^2}\right) \left(\frac{1}{3}(d^3 S)^d\right) \leq \left(\frac{1}{2} - \frac{1}{d^2}\right) \operatorname{vol} P,$$

da für genügend großes d gilt

$$\begin{aligned} & 4 \operatorname{bl}(L) dq (6d(d^3 S)^{d-1}) \leq \left(\frac{1}{2} - \frac{1}{d^2}\right) \left(\frac{1}{3}(d^3 S)^d\right) \\ \iff \quad & 72 \operatorname{bl}(L) d^2 q \leq \left(\frac{1}{2} - \frac{1}{d^2}\right) d^3 S \\ \iff \quad & 72 \operatorname{bl}(L) d^5 \leq \frac{1}{4} S \\ \impliedby \quad & 36 d^7 \operatorname{bl}(L) \leq S. \end{aligned}$$

Im folgenden betrachten wir die guten Fälle und schätzen nachher die Irrtumswahrscheinlichkeit von Algorithmus 8.5.3 ab.
Mit Hilfe von Algorithmus 8.3.1 und Proposition 8.4.1 bekommen wir die Möglichkeit, zufällig gemäß der Gleichverteilung Vektoren aus der Menge $(\mathbb{Z}/q\mathbb{Z})^d$ zu wählen. Dabei gilt für jedes $i \in \{1, \ldots, n\}$: $\Pr[\boldsymbol{u}_i \in P^-_{\boldsymbol{a}_i}] \geq 1 - \frac{1}{d^2}$. Die Wahrscheinlichkeit, daß für alle $i \in \{1, \ldots, n\}$ gleichzeitig $\boldsymbol{u}_i \in P^-_{\boldsymbol{a}_i}$ ist, beträgt somit wenigstens $(1 - \frac{1}{d^2})^n$.
Wir nehmen jetzt an, daß die $\boldsymbol{a}_i$ unabhängig voneinander, zufällig gemäß der Gleichverteilung aus $(\mathbb{Z}/q\mathbb{Z})^d$ gewählt wurden, d.h. für alle $i \in \{1, \ldots, d\}$ gleichzeitig $\boldsymbol{u}_i \in P^-_{\boldsymbol{a}_i}$ gilt. Demnach ist die Matrix A zufällig gemäß der Gleichverteilung aus $(\mathbb{Z}/q\mathbb{Z})^{d \times n}$ gewählt. Anwendung der TURING-Maschine M mit der Eingabe A liefert $\boldsymbol{x} \in \{-1, 0, 1\}^n \backslash \{\boldsymbol{0}\}$, so daß die Wahrscheinlichkeit für das Ereignis „$A\boldsymbol{x} = \boldsymbol{0}$“ mindestens $\frac{1}{|p(d)|}$ ist.
Es sei $A\boldsymbol{x} = \boldsymbol{0}$. Dann ist $\sum_{i=1}^n x_i \boldsymbol{O}_{\boldsymbol{a}_i} \in L$, denn es gilt

$$\sum_{i=1}^n x_i \boldsymbol{O}_{\boldsymbol{a}_i} = \sum_{i=1}^n x_i \sum_{j=1}^d \frac{a_{ij}}{q} \boldsymbol{w}_j = \sum_{j=1}^d \sum_{i=1}^n \frac{x_i a_{ij}}{q} \boldsymbol{w}_j,$$

und da $A\boldsymbol{x} = \boldsymbol{0} \in (\mathbb{Z}/q\mathbb{Z})^d$, ist $\sum_{i=1}^n \frac{x_i a_{ij}}{q} \in \mathbb{Z}$, $j = 1, \ldots, d$. Also ist auch das von Algorithmus 8.5.3 in Schritt vi) berechnete $\boldsymbol{v} \in L$.
Weil $\boldsymbol{u}_i \in P^-_{\boldsymbol{a}_i}$ ist, gilt $\boldsymbol{u}_i - \boldsymbol{O}_{\boldsymbol{a}_i} \in P^-_{\boldsymbol{0}}$, somit $\|\boldsymbol{u}_i - \boldsymbol{O}_{\boldsymbol{a}_i}\| \leq \frac{d}{q} \max_{j=1,\ldots,d} \|\boldsymbol{w}_j\|$, $i = 1, \ldots, d$. Proposition 8.2.2 liefert $\max_{j=1,\ldots,d} \|\boldsymbol{w}_j\| \leq (d^3 + \frac{1}{2}d)S$. Dies zusammen mit $|x_i| \leq 1$, $i = 1, \ldots, n$, ergibt für genügend großes d stets

$$\|\boldsymbol{v}\| = \left\| \sum_{i=1}^n x_i (\boldsymbol{u}_i - \boldsymbol{O}_{\boldsymbol{a}_i}) \right\| \leq n \|\boldsymbol{u}_i - \boldsymbol{O}_{\boldsymbol{a}_i}\| \leq n \frac{d}{q} \left(d^3 + \frac{1}{2}d\right) S \leq \frac{S}{2}.$$

Es bleibt noch zu zeigen, daß $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d-1}, \boldsymbol{v}$ mit genügend hoher Wahrscheinlichkeit linear unabhängig sind. Da $\boldsymbol{x} \neq \boldsymbol{0}$ ist, können wir ohne Einschränkung $x_1 \neq 0$ annehmen. Dann gilt

$$\begin{aligned} \boldsymbol{v} \in \mathbb{R}\boldsymbol{v}_1 \oplus \cdots \oplus \mathbb{R}\boldsymbol{v}_{d-1} \quad & \iff \quad \sum_{i=1}^n x_i (\boldsymbol{u}_i - \boldsymbol{O}_{\boldsymbol{a}_i}) \in \mathbb{R}\boldsymbol{v}_1 \oplus \cdots \oplus \mathbb{R}\boldsymbol{v}_{d-1} \\ & \iff \quad x_1 (\boldsymbol{u}_1 - \boldsymbol{O}_{\boldsymbol{a}_1}) \in -\sum_{i=2}^n x_i (\boldsymbol{u}_i - \boldsymbol{O}_{\boldsymbol{a}_i}) + \mathbb{R}\boldsymbol{v}_1 \oplus \cdots \oplus \mathbb{R}\boldsymbol{v}_{d-1}. \end{aligned}$$

Proposition 8.4.1 besagt, daß für *jede* affine Hyperebene H $\Pr[\boldsymbol{u}_1 - \boldsymbol{O}_{\boldsymbol{a}_1} \in H] < \frac{1}{2}$ ist. Dies gilt auch speziell für die affine Hyperebene $-\frac{1}{x_1}\sum_{i=2}^{n} x_i(\boldsymbol{u}_i - \boldsymbol{O}_{\boldsymbol{a}_i}) + \mathbb{R}\boldsymbol{v}_1 \oplus \cdots \oplus \mathbb{R}\boldsymbol{v}_{d-1}$. Also sind $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d-1}, \boldsymbol{v}$ mit einer Wahrscheinlichkeit von wenigstens $\frac{1}{2}$ linear unabhängig.
Wir resümieren, an welchen Stellen der Erfolg von Algorithmus 8.5.3 vom Zufall abhängt, und berechnen seine Irrtumswahrscheinlichkeit:

- Abweichung von der Gleichverteilung durch Algorithmus 8.3.1 führt zu einer Irrtumswahrscheinlichkeit von höchstens 2^{-d^ε}.

- Es gilt $n = \lceil 2\log_2 d^6 \rceil$. Für genügend großes d gilt stets $d^2 \geq \lceil 2\log_2 d^6 \rceil$. Damit ist (wieder Anwendung von $e^t(1 - \frac{t^2}{n}) \leq (1 + \frac{t}{n})^n$)

$$\Pr[\boldsymbol{u}_1 \in P^-_{\boldsymbol{a}_1}, \ldots, \boldsymbol{u}_n \in P^-_{\boldsymbol{a}_n}] \geq \left(1 - \frac{1}{d^2}\right)^n \geq \left(1 - \frac{1}{d^2}\right)^{d^2} \geq e^{-1}\left(1 - \frac{1}{d^2}\right) \geq \frac{1}{2e}.$$

  Dies führt zu einer Irrtumswahrscheinlichkeit von höchstens $1 - \frac{1}{2e}$.

- Anwendung der TURING-Maschine M führt zu einer Irrtumswahrscheinlichkeit von höchstens $1 - \frac{1}{|p(d)|}$.

- $\Pr[\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d-1}, \boldsymbol{v}$ linear unabhängig$] > \frac{1}{2}$, führt zu einer Irrtumswahrscheinlichkeit von höchstens $\frac{1}{2}$.

D.h. die Irrtumswahrscheinlichkeit von Algorithmus 8.5.3 beträgt bei k-maliger Wiederholung der Schritte ii) – v) mit anschließendem Erfolgstest ($A\boldsymbol{x} = \boldsymbol{0}$?) höchstens

$$2^{-d^\varepsilon} + \left(1 - \frac{1}{2e}\right)^k + \left(1 - \frac{1}{|p(d)|}\right)^k + \frac{1}{2}.$$

Es kann k (polynomiell in d) so gewählt werden, daß der obige Ausdruck kleiner als $\frac{3}{4}$ wird. Wenn wir den gesamten Algorithmus, der die Schritte ii) – v) k-mal ausführt, dreimal mit anschließendem Erfolgstest ($\|\boldsymbol{v}\| \leq \frac{S}{2}$ und $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d-1}, \boldsymbol{v}$ linear unabhängig?) anwenden, erhalten wir eine Berechnungsvorschrift für die TURING-Maschine M_1, die eine Erfolgswahrscheinlichkeit von mindestens $\frac{1}{2}$ besitzt. ◇

Im vorhergehenden Beweis ist ersichtlich, warum $q = d^6$ und $n = \lceil 2d\log_2 q \rceil$ gewählt wurden. Es müssen die Ungleichungen $\frac{q\,\mathrm{bl}(L)}{\mathrm{width}\,P} \leq \frac{1}{12d^4}$ und $4\,\mathrm{bl}(L)dq\,\mathrm{vol}\,\partial P < (\frac{1}{2} - \frac{1}{d^2})\,\mathrm{vol}\,P$ erfüllt sein, um Proposition 8.4.1 anwenden zu können. Außerdem muß für die Reduktion des längsten Vektors von $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d$ die Ungleichung $n\frac{d}{q}(d^3 + \frac{1}{2}d) \leq \frac{1}{2}$ erfüllt sein. Ein nützlicher Nebeneffekt der Wahl von q und n ist, daß die durch die Matrix $A \in (\mathbb{Z}/q\mathbb{Z})^{d\times n}$ gegebene Abbildung mit Urbildmenge $\{0,1\}^n\backslash\{\boldsymbol{0}\}$ nicht injektiv ist, da die Zahl der Urbilder $2^n - 1$ die Zahl der möglichen Bilder q^d übersteigt. Also gibt es ein $\boldsymbol{x} \in \{-1,0,1\}^n\backslash\{\boldsymbol{0}\}$ mit $A\boldsymbol{x} = \boldsymbol{0}$.

Die TURING-Maschine M_1 aus Lemma 8.5.2 ermöglicht die Reduktion eines Gittervektors. Wenn die TURING-Maschine M_1 mehrfach angewendet wird, wobei das Ergebnis einer erfolgreichen Reduktion jeweils die neue Eingabe darstellt, dann entsteht ein System von kurzen, linear unabhängigen Gittervektoren und damit eine Approximation der Basislänge des Gitters.

Zur Bestimmung der Anzahl der benötigten Wiederholungen benutzen wir die Ungleichung von CHERNOFF.

Theorem 8.5.4. (CHERNOFF-*Ungleichung, siehe [MR95], Theorem 4.2*)
Es seien $X_1, \ldots, X_n$ unabhängige Zufallsvariablen aus der Menge $\{0,1\}$. Es sei $\Pr[X_i = 1] = p_i$ mit $0 < p_i < 1$, $i = 1, \ldots, n$. Dann gilt für $X = \sum_{i=1}^{n} X_i$, $\mu = E[X] = \sum_{i=1}^{n} p_i$ und $0 < \delta \leq 1$ die CHERNOFF-Ungleichung

$$\Pr[X < (1-\delta)\mu] < e^{-\frac{\mu\delta^2}{2}}.$$

Lemma 8.5.5. Angenommen es gibt ein Polynom $p \in \mathbb{R}[X]$ und eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M, so daß die Wahrscheinlichkeit

$$\Pr[M \text{ berechnet aus } A \in (\mathbb{Z}/q\mathbb{Z})^{d\times n} \text{ ein } \boldsymbol{x} \in \{-1,0,1\}^n \backslash \{\mathbf{0}\} \text{ mit } A\boldsymbol{x} = \mathbf{0}] > \frac{1}{|p(d)|}$$

ist. Dabei wird die Wahrscheinlichkeit über die zufällige Wahl von $A \in (\mathbb{Z}/q\mathbb{Z})^{d\times n}$ und über die Münzwürfe von M genommen. Dann gibt es eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M_2, die bei Eingabe von linear unabhängigen Vektoren $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$ mit einer Wahrscheinlichkeit von wenigstens $\frac{1}{2}$ Vektoren $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d \in L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$ berechnet, so daß $\max_{i=1,\ldots,d} \|\boldsymbol{v}_i\| \leq 72d^7 \operatorname{bl}(L)$ gilt.

Beweis. Zunächst werden die $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ mit dem LLL-Algorithmus zu $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_d$ reduziert. Danach wird durch mehrfache Anwendung der TURING-Maschine M_1 aus Lemma 8.5.2 die Länge des längsten Vektors der jeweiligen Eingabe Schritt für Schritt halbiert. Bei einem erfolgreichen Reduktionsschritt wird die Ausgabe zur Eingabe der nächsten Berechnung (Selbstreduktion). Es sind zur Erreichung des gewünschten Ergebnisses $\max_{i=1,\ldots,d} \|\boldsymbol{v}_i\| \leq 72d^7 \operatorname{bl}(L)$ weniger als d^2 erfolgreiche Reduktionsschritte notwendig, da die vorhergehende Anwendung des LLL-Algorithmus schon $\|\boldsymbol{w}_i\| \leq 2^{(d-1)/2}\lambda_i(L) \leq 2^{(d-1)/2} \operatorname{bl}(L)$, $i = 1, \ldots, d$, garantiert. Diese Ungleichung ist eine Eigenschaft des LLL-Algorithmus, der in [Coh93] ausführlich beschrieben wird. Die Ungleichung wird hier benötigt, da sie garantiert, daß die Länge von jedem Vektor $\boldsymbol{w}_i$ höchstens d-mal halbiert werden muß, um das gewünschte Ergebnis zu erreichen.
Falls das gewünschte Ergebnis nicht erreicht ist, ist die Wahrscheinlichkeit für einen erfolgreichen Reduktionsschritt mindestens $\frac{1}{2}$. Die CHERNOFF-Ungleichung gibt eine obere Schranke dafür an, wie viele Reduktionsschritte durchgeführt werden müssen, damit die Wahrscheinlichkeit für die Erreichung des gewünschten Ergebnisses wenigstens $\frac{1}{2}$ ist. Es sei $d \geq 2$. Wir wenden $4d^2$ mal die TURING-Maschine M_1 an. Es bezeichne X die Anzahl der erfolgreichen Reduktionsschritte. Wir gehen idealisierend davon aus, daß in jedem Reduktionsschritt die Bedingungen aus Lemma 8.5.2 erfüllt sind und daß $E[X] = 2d^2$ ist. Dies führt nur zu einer schlechteren Abschätzung: Wenn die Voraussetzung aus Lemma 8.5.2 nicht erfüllt sind, dann ist das gewünschte Ergebnis bereits erreicht; außerdem gilt $E[X] \geq 2d^2$.
Die CHERNOFF-Ungleichung lautet dann

$$\Pr[X < d^2] = \Pr[X < (1-1/2)2d^2] < e^{-\frac{1}{4}d^2} \leq \frac{1}{2}.$$

$\diamond$

Es seien $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ eine Eingabe und $L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$. Mit Hilfe der TURING-Maschine M_2 kann das d-te sukzessive Minimum (siehe Definition 4.2.3) des zu L dualen Gitters $L^\#$ approximiert werden. Nach Theorem 4.2.5, das einen Zusammenhang zwischen $\min L = \lambda_1(L)$ und $\lambda_d(L^\#)$ herstellt, kann die Approximation von $\lambda_d(L^\#)$ zur Approximation von $\min L$ verwendet werden.

Beweis. (von Theorem 8.5.1)
Der nachfolgende Algorithmus kann probabilistisch entscheiden, ob eine Eingabe zur Sprache Gap-$(1, 72d^8\sqrt{\frac{d+3}{4}})$-SVP gehört.

Algorithmus 8.5.6 Probabilistischer Entscheidungsalgorithmus für Gap-$(1, 72d^8\sqrt{\frac{d+3}{4}})$-SVP

Eingabe: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Q}^d$ linear unabhängig, $L := \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d$.
Ausgabe: (probabilistisch, mit möglichem einseitigen Irrtum:)
„Ja.“, falls $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \in$ Gap-$(1, 72d^8\sqrt{\frac{d+3}{4}})$-SVP.
„Nein.“, falls $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \notin$ Gap-$(1, 72d^8\sqrt{\frac{d+3}{4}})$-SVP.
Berechnung der zu $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ dualen Basis $(\boldsymbol{b}_1^\#, \ldots, \boldsymbol{b}_d^\#)$ des Gitters $L^\#$ durch Lösung des linearen Gleichungssystems $(\boldsymbol{b}_i, \boldsymbol{b}_j^\#) = \delta_{ij}$, $1 \leq i, j \leq d$.
Anwendung der probabilistischen TURING-Maschine M_2 auf $(\boldsymbol{b}_1^\#, \ldots, \boldsymbol{b}_d^\#)$ liefert eine Basis $(\boldsymbol{v}_1^\#, \ldots, \boldsymbol{v}_d^\#)$ von $L^\#$.
$\lambda \leftarrow \max_{i=1,\ldots,d} \|\boldsymbol{v}_i^\#\|$.
$\lambda' \leftarrow 72d^8\sqrt{\frac{d+3}{4}}/\lambda$.
if $\lambda' \leq 72d^8\sqrt{\frac{d+3}{4}}$ **then**
  output „Ja.“
else
  output „Nein.“
end if

Die durch die Berechnung der probabilistischen polynomiell zeitbeschränkten TURING-Maschine M_2 gewonnene Zahl λ ist mit einer Wahrscheinlichkeit von mindestens $\frac{1}{2}$ eine Approximation für die Basislänge von $L^\#$. Es gilt

$$\mathrm{bl}(L^\#) \leq \lambda \leq 72d^7\,\mathrm{bl}(L^\#).$$

Außerdem ist dann λ eine Approximation für das d-te sukzessive Minimum von $L^\#$. Einerseits ist $\lambda_d(L^\#) \leq \mathrm{bl}(L^\#) \leq \lambda$ und andererseits ergibt sich mit Proposition 4.2.9 die Ungleichung $\mathrm{bl}(L^\#) \leq \sqrt{\frac{d+3}{4}}\lambda_d(L^\#)$. Wir erhalten somit

$$\lambda_d(L^\#) \leq \lambda \leq 72d^7\sqrt{\frac{d+3}{4}}\lambda_d(L^\#).$$

Theorem 4.2.5 liefert: $1 \leq \lambda_1(L)\lambda_d(L^\#) \leq d$. Auf der einen Seite gilt

$$\lambda_1(L) \leq \frac{d}{\lambda_d(L^\#)} \leq 72d^8\sqrt{\frac{d+3}{4}}\lambda = \lambda',$$

und auf der anderen Seite gilt

$$\lambda' = \frac{72d^8\sqrt{\frac{d+3}{4}}}{\lambda} \leq \frac{72d^8\sqrt{d+3}4}{\lambda_d(L^\#)} \leq 72d^8\sqrt{\frac{d+3}{4}}\lambda_1(L).$$

Falls $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \in$ Gap-$(1, 72d^8\sqrt{(d+3)/4})$-SVP, dann akzeptiert Algorithmus 8.5.6 mit einer Wahrscheinlichkeit von mindestens $\frac{1}{2}$. Falls $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) \notin$ Gap-$(1, 72d^8\sqrt{(d+3)/4})$-SVP, dann akzeptiert Algorithmus 8.5.6 niemals. ◇

8.6 Konstruktion einer One-Way-Funktion

GOLDREICH, GOLDWASSER und HALEVI bemerkten in [GGH96], daß das Theorem der Worst-Case/Average-Case-Äquivalenz so umformuliert werden kann, daß es die Konstruktion einer One-Way-Funktion ermöglicht. Eine hinreichende Voraussetzung der Konstruktion ist, daß es keinen effizienten Approximationsalgorithmus für SVP mit Worst-Case-Güte d^9 gibt.
Bislang kann für keine andere Funktion eine derartige Aussage getroffen werden. Alle anderen potentiellen One-Way-Funktionen basieren auf Hypothesen der Form von Vermutung 3.2.1 über das Problem des diskreten Logarithmus: Es wird vermutet, daß das Problem des diskreten Logarithmus schon im Average-Case schwierig ist.

Theorem 8.6.1. Es seien die natürlichen Zahlen d, q und n so wie im vorhergehenden Abschnitt. Falls Gap-$(1, 72d^8\sqrt{(d+3)/4})$-SVP $\notin \mathcal{RP}$ ist, dann ist die Abbildung

$$f : \begin{cases} ((\mathbb{Z}/q\mathbb{Z})^{d\times n}, \{0,1\}^n\backslash\{\mathbf{0}\}) & \to & ((\mathbb{Z}/q\mathbb{Z})^{d\times n}, (\mathbb{Z}/q\mathbb{Z})^d) \\ (A, \boldsymbol{x}) & \mapsto & (A, A\boldsymbol{x}) \end{cases}$$

eine (starke) One-Way-Funktion.

Beweis. Angenommen f ist keine starke One-Way-Funktion. Dann gibt es ein Polynom $p \in \mathbb{R}[X]$ und eine probabilistische polynomiell zeitbeschränkte TURING-Maschine M, die bei Eingabe von $(A, A\boldsymbol{x})$ ein $\boldsymbol{y} \in \{0,1\}^n\backslash\{\mathbf{0}\}$ berechnet, so daß mit einer Wahrscheinlichkeit von wenigstens $\frac{1}{|p(d)|}$ die Gleichung $A\boldsymbol{x} = A\boldsymbol{y}$ gilt. Dabei wird die Wahrscheinlichkeit über die zufällige Wahl von A, die zufällige Wahl von $\boldsymbol{x}$ und die Münzwürfe von M genommen. Wir werden sehen, daß der nachfolgende Algorithmus die Voraussetzung von Theorem 8.5.1 erfüllt.

Algorithmus 8.6.2

Eingabe: $A \in (\mathbb{Z}/q\mathbb{Z})^{d\times n}$.

Ausgabe: (probabilistisch, mit möglichem Irrtum:) $\boldsymbol{z} \in \{-1,0,1\}^n$ mit $A\boldsymbol{z} = \mathbf{0} \in (\mathbb{Z}/q\mathbb{Z})^d$.

Wähle $\boldsymbol{x} \in \{0,1\}^n\backslash\{\mathbf{0}\}$ zufällig gemäß der Gleichverteilung.
Anwendung der TURING-Maschine M auf $(A, A\boldsymbol{x})$ liefert $\boldsymbol{y}$.
$\boldsymbol{z} \leftarrow \boldsymbol{x} - \boldsymbol{y}$.

Es folgt unmittelbar $\Pr[A\boldsymbol{z} = \mathbf{0}] \geq \frac{1}{|p(d)|}$. Dabei wird die Wahrscheinlichkeit über die zufällige Wahl von A und die Münzwürfe von M genommen.
Es verbleibt zu zeigen, daß unter der Bedingung $A\boldsymbol{z} = \mathbf{0}$ mit hoher Wahrscheinlichkeit $\boldsymbol{z} \neq \mathbf{0}$, d.h. $\boldsymbol{x} \neq \boldsymbol{y}$ ist. Dazu betrachten wir den „schlimmsten“ Fall: alle bis auf einer der Vektoren aus $(\mathbb{Z}/q\mathbb{Z})^d$ besitzen unter der durch A beschriebenen linearen Abbildung genau ein Urbild aus der Menge $\{0,1\}^n\backslash\{\mathbf{0}\}$. Da $q = d^6$ und $n = \lceil 2d\log_2 q\rceil$ ist, gilt

$$\begin{aligned} \Pr[\boldsymbol{x} \neq \boldsymbol{y} | A\boldsymbol{x} = A\boldsymbol{y}] &\geq \frac{1}{2}\cdot\frac{(2^n-1)-(q^d-1)}{2^n-1} \\ &\geq \frac{1}{2} - \frac{q^d}{2^{n+1}} \\ &= \frac{1}{2} - 2^{d\log_2 q-(n+1)} \\ &= \frac{1}{2} - 2^{d\log_2 q-\lceil 2d\log_2 q\rceil-1} \\ &\geq \frac{1}{2} - \frac{1}{2^{d\log_2 q-2}}. \end{aligned}$$

Somit gibt es ein Polynom $p_1 \in \mathbb{R}[X]$, so daß die Berechnung von Algorithmus 8.6.2 mit Wahrscheinlichkeit

$$\Pr[A\boldsymbol{z} = \mathbf{0} \text{ und } \boldsymbol{z} \neq \mathbf{0}] = \Pr[A\boldsymbol{z} = \mathbf{0}] \cdot \Pr[\boldsymbol{z} \neq \mathbf{0} | A\boldsymbol{z} = \mathbf{0}] \geq \frac{1}{|p_1(d)|}$$

erfolgreich ist. Wenn die Voraussetzung von Theorem 8.6.1 richtig ist, kann dies nach Theorem 8.5.1 nicht sein. Widerspruch! ◇

Da bislang der LLL-Algorithmus mit $2^{(d-1)/2}$ den besten effizient berechenbaren Approximationsfaktor für SVP bietet, ist die Voraussetzung Gap-$(1, 72d^8\sqrt{(d+3)/4})$-SVP $\notin \mathcal{RP}$ von Theorem 8.6.1 nicht abwegig.

Die Wunschvorstellung ist, daß Gap-$(1, 72d^8\sqrt{(d+3)/4})$-SVP $\mathcal{NP}$-vollständig ist. Unter dieser Voraussetzung und unter der Voraussetzung $\mathcal{RP} \neq \mathcal{NP}$ ist f eine One-Way-Funktion. Damit wäre eine theoretische Fundierung der modernen Kryptographie mit Hilfe von üblichen komplexitätstheoretischen Annahmen gelungen.

Wir haben jedoch in Kapitel 7 gesehen, daß schon Gap-$(2, \sqrt{d/\ln d})$-SVP nicht $\mathcal{NP}$-vollständig sein kann, da sonst die polynomielle Hierarchie zu Σ_2 zusammenfällt.

Um dieses Dilemma zu umgehen, kann man versuchen, den Faktor $72d^8\sqrt{(d+3)/4}$ zu senken. Dies haben auch CAI und NERURKAR in einer Folge von Artikeln [CN97], [Cai98a] und [Cai98b] erreicht. Der aktuelle Stand ist der Faktor $d^{4+\varepsilon}$.

In Kapitel 6 haben wir gesehen, daß die Berechnung einer $(\sqrt{2} - \varepsilon)$-Approximation für SVP $\mathcal{NP}$-hart ist, falls eine zahlentheoretische Vermutung zutrifft. Somit ist der Faktor $\sqrt{2} - \varepsilon$ erstrebenswert. Die Ansätze von AJTAI, CAI und NERURKAR beruhen alle auf dem Theorem 4.2.5: $1 \leq \lambda_1(L)\lambda_d(L^\#) \leq d$. Es ist bekannt, daß die obere Schranke bis auf eine Konstante optimal ist. Also muß ein anderer Ansatz gefunden werden, um den Faktor unter d zu senken.

Es verbleibt die theoretische Fundierung der modernen Kryptographie mit Hilfe von üblichen komplexitätstheoretischen Annahmen als ein offenes Problem.

Literaturverzeichnis

[ABSS93] SANJEEV ARORA, LÁSZLÓ BABAI, JACQUES STERN, ELIZABETH SWEEDYK. *The hardness of approximate optima in lattices, codes, and systems of linear equations. 34th Annual IEEE Symposium on Foundations of Computer Science*, 724–733, 1993.

[Ajt96] MIKLÓS AJTAI. *Generating hard instances of lattice problems (extended abstract). 28th Annual ACM Symposium on the Theory of Computing*, 99–108, 1996.

[Ajt98] MIKLÓS AJTAI. *The shortest vector problem in L_2 is $\mathcal{NP}$-hard for randomized reductions. 30th Annual ACM Symposium on the Theory of Computing*, 10–19, 1998.

[Aro94] SANJEEV ARORA. *Probabilistic checking of proofs and hardness of approximation problems.* PhD thesis, Princeton University, 1994.

[ARS78] LEONARD M. ADLEMAN, RONALD L. RIVEST, ADI SHAMIR. *A method for obtaining digital signature and public-key cryptosystems. Communication of the ACM* **21** (1978), 120–126.

[Ban93] WOJCIECH BANASZCZYK. *New bounds in some transference theorems in the geometry of numbers. Mathematische Annalen* **296** (1993), 625–635.

[Beh99] EHRHARD BEHRENDS. *$\mathcal{P} = \mathcal{NP}$?* DIE ZEIT, 4. März 1999.

[Cai98a] JIN-YI CAI. *A new transference theorem and applications to* AJTAI*'s connection factor. Electronic Colloquium on Computational Complexity*, 1998.

[Cai98b] JIN-YI CAI. *A relation of primal-dual lattices and the complexity of shortest lattice vector problem, Preprint*, 1998.

[CN97] JIN-YI CAI, AJAY P. NERURKAR. *An improved worst-case to average-case connection for lattice problems (extended abstract). 38th Annual IEEE Symposium on Foundations of Computer Science*, 468–477, 1997.

[CN98] JIN-YI CAI, AJAY P. NERURKAR. *Approximation the SVP to within a factor $(1+\frac{1}{d^\varepsilon})$ is $\mathcal{NP}$-hard under randomized reductions (extended abstract), Preprint*, 1998.

[Coh93] HENRI COHEN. *A course in computational algebraic number theory. Graduate Texts in Mathematics* **138**. Springer-Verlag, 1993.

[CS88] JOHN H. CONWAY, NEIL J.A. SLOANE. *Sphere Packing, Lattices and Groups. Grundlehren der math. Wiss.* **290**. Springer-Verlag, 1988.

[DH76] WINFRIED DIFFIE, MARTIN E. HELLMAN. *New directions in cryptography. IEEE Transations on Information Theory* **22** (1976), 644–654.

[DKS98] IRIT DINUR, GUY KINDLER, SHMUEL SAFRA. *Approximating CVP to within almost-polynomial factors is $\mathcal{NP}$-hard. 39th Annual IEEE Symposium on Foundations of Computer Science*, 1998.

[FP85] ULRICH FINCKE, MICHAEL POHST. *Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. Mathematics of Computation* **44** (1985), 463–471.

[GB97] SHAFI GOLDWASSER, MIHIR BELLARE. *Lecture notes on cryptography.* Erhältlich unter `http://www-cse.ucsd.edu/users/mihir/papers/gb.html`, 1997.

[GG98] ODED GOLDREICH, SHAFI GOLDWASSER. *On the limits of non-approximability of lattice problems. 30th Annual ACM Symposium on Theory of Computing*, 1–9, 1998.

[GGH96] ODED GOLDREICH, SHAFI GOLDWASSER, SHAI HALEVI. *Collision-free hashing from lattice problems. Electronic Colloquium on Computational Complexity*, 1996.

[GJ79] MICHAEL R. GAREY, DAVID S. JOHNSON. *Computers and intractibility, a guide to the theory of $\mathcal{NP}$-completeness.* W.H. Freeman and Company, 1979.

[GL87] PETER M. GRUBER, CORNELIS G. LEKKERKERKER. *Geometry of numbers*. North-Holland, 1987.

[GL96] GENE H. GOLUB, CHARELES F. VAN LOAN. *Matrix computations.* Johns Hopkins University Press, 1996.

[GLS88] MARTIN GRÖTSCHEL, LÁZLÓ LOVÁSZ, ALEXANDER SCHRIJVER. *Geometric algorithms and combinatorial optimization.* Springer-Verlag, 1988.

[GMSS99] ODED GOLDREICH, DANIELE MICCIANCIO, SHMUEL SAFRA, JEAN-PIERRE SEIFERT. *Approximating shortest lattice vectors is not harder than approximating closest lattice vectors. Electronic Colloquium on Computational Complexity*, 1999.

[Gol95] ODED GOLDREICH. *Foundations of cryptography, fragments of a book.* Erhältlich unter `http://www.wisdom.weizmann.ac.il/oded/`, 1995.

[Gol97] ODED GOLDREICH. *On the foundations of modern cryptography. Advances in Cryptology—CRYPTO '97*, 46–74, 1997.

[GS86] SHAFI GOLDWASSER, MICHAEL SIPSER. *Private coins versus public coins in interactive proof systems. 18th Annual ACM Symposium on the Theory of Computing*, 59-68, 1996.

[Hås97] JOHANN HÅSTAD. *Clique is hard to approximate within $n^{1-\varepsilon}$. Electronic Colloquium on Computational Complexity*,1997.

[Hen97] MARTIN HENK. *Note on shortest und nearest lattice vectors. Inform. Process. Lett.* **61** (1997), 183–188.

[Heu91] HARRO HEUSER. *Lehrbuch der Analysis, Teil 2*, 7. Auflage. Teubner Verlag, 1991.

[Heu94] HARRO HEUSER. *Lehrbuch der Analysis, Teil 1*, 11. Auflage. Teubner Verlag, 1994.

[HU79] JOHN E. HOPCROFT, JEFFREY D. ULLMAN. *Introduction to automata theory, languages and computation.* Addison-Wesley, 1979.

[Knu69] DONALD E. KNUTH. *Seminumerical Algorithms.* Addison Wesley, 1969.

[Lag95] JEFFREY C. LAGARIAS. *Chapter „Point Lattices“. Handbook of Combinatorics.* North Holland, 1995.

[Len93] AARJEN K. LENTRA, editor. *The development of the number field sieve, Lecture Notes Mathematics* **1554**. Springer-Verlag, 1993.

[LLL82] AARJEN K. LENSTRA, HENDRIK W. LENSTRA, LAZLO LOVÁSZ. *Factoring polynomials with rational coefficients. Mathematische Annalen* **261** (1982), 515–534.

[LLS90] JEFFREY C. LAGARIAS, HENDRIK W. LENSTRA, CLAUS P. SCHNORR. KORKINE-ZOLOTAREV *bases and successive minima of a lattice and its reciprocal lattice. Combinatorica* **10** (1990), 334–348.

[Mic98a] DANIELE MICCIANCIO. *The shortest vector in a lattice is hard to approximate to within some constant. 39th Annual IEEE Symposium on Foundations of Computer Science*, 1998.

[Mic98b] DANIELE MICCIANCIO. *The shortest vector in a lattice is hard to approximate to within some constant.* PhD thesis, MIT, 1998.

[Mic99] DANIELE MICCIANCIO. *Lattice based cryptography: a global improvement. Theory of Cryptography Library*, 1999.

[MPS98] ERNST W. MAYR, HANS J. PRÖMEL, ANGELIKA STEGER. *Lectures on proof verification and approximation algorithms.* Springer-Verlag, 1998.

[MR95] RAJEEV MOTWANI, PRABHAKAR RAGHAVAN. *Randomized algorithms.* Cambridge University Press, 1995.

[MVV97] ALFRED J. MENEZES, PAUL C. VAN OORSCHOT, SCOTT A. VANSTONE. *Handbook of applied cryptography*. CRC Press, 1997.

[Neu92] JÜRGEN NEUKIRCH. *Algebraische Zahlentheorie.* Springer-Verlag, 1992.

[Odl85] ANDREW M. ODLYZKO. *The discrete logarithm problem and its cryptographic significance. Advances in Cryptology: Proceedings of Eurocrypt '84*, 224–314, 1985.

[Sha49] CLAUDE E. SHANNON. *Communication theory of secrecy systems. Bell System Technical Journal* **28** (1949), 656–715.

[Sha84] ADI SHAMIR. *A polynomial time algorithm for breaking the basic* MERKLE-HELLMAN *cryptosystem. IEEE Transactions on Information Theory* **30** (1984), 699–704.

[Sha92] ADI SHAMIR. $\mathcal{IP} = \mathcal{P}$*-SPACE. Journal of the ACM* **39** (1992), 869–977.

[Sho94] PETER W. SHOR. *Algorithms for quantum computation: discrete logarithms and factoring.* 35th Annual IEEE Symposium on Foundations of Computer Science, 124–134, 1994.

[vEB81] PETER VAN EMDE BOAS. *Another* $\mathcal{NP}$*-complete partition problem and the complexity of computing short vectors in a lattice. Technical Report*, Universität Amsterdam, 1981.

[Weg93] INGO WEGENER. *Theoretische Informatik.* B.G. Teubner, 1993.

[Weg95] INGO WEGENER. *Skript zur Spezialvorlesung Komplexitätstheorie 2, SS 95*. Universität Dortmund, 1995.

[Weg98] INGO WEGENER. *Skript zur Stammvorlesung Komplexitätstheorie 1, WS 98/99.* Universität Dortmund, 1998.

[Zie95] GÜNTER M. ZIEGLER. *Lectures on Polytopes. Graduate Texts in Mathematics* **152**. Springer-Verlag, 1995.

Erklärung

Hiermit erkläre ich, FRANK VALLENTIN, daß ich die Diplomarbeit mit dem Titel: „Zur Komplexität des „Shortest Vector Problem“ und seine Anwendungen in der Kryptographie“ selbständig und nur unter Verwendung der angegebenen Hilfsmittel angefertigt habe.

Dortmund, den 13. August 1999

Einverständniserklärung des Urhebers

Ich, FRANK VALLENTIN, erkläre mich einverstanden, daß meine Diplomarbeit nach §6 (1) des URG der Öffentlichkeit durch die Übernahme in die Bereichsbibliothek zugänglich gemacht wird. Damit können Leser der Bibliothek die Arbeit einsehen und zu persönlichen wissenschaftlichen Zwecken Kopien aus dieser Arbeit anfertigen. Weiter Urheberrechte werden nicht berührt.

Dortmund, den 13. August 1999