

# Sicherheitsparameter für Regevs Kryptosystem



Technische Universität Darmstadt  
Fachbereich Mathematik

Diplomarbeit  
Tobias Müller  
Betreuer: Prof. Dr. J. Buchmann

7. Dezember 2004

## **Erklärung**

Hiermit erkläre ich, dass ich meine Diplomarbeit nur unter Verwendung der in der Arbeit angegebenen Quellen selbständig angefertigt habe.

Der Einsicht in die Diplomarbeit und der Ausleihe eines Exemplars stimme ich zu.

Tobias Müller

## **Danksagung**

Ich möchte mich zunächst bei Prof. Dr. Buchmann und Arthur Schmidt bedanken, die mir diese interessante Arbeit ermöglichten. Mein besonderer Dank gilt meiner Familie, die mich während meines gesamten Studiums in jedweder Hinsicht unterstützt haben.

Außerdem möchte ich den vielen neuen Freunden danken, die ich während meines Studiums gewonnen habe. Besonders ihretwegen habe ich gerne am Fachbereich Mathematik der TU Darmstadt studiert. Nennen möchte ich an dieser Stelle Claudia Appel, Martina Felber, Clemens Krauß, Stefanie Krusche, Herr Küpper, Raphael Overbeck, Marcus Prill, Daniel Weimer und Julian Wiedl.



# Inhaltsverzeichnis

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Vorwort</b>  | <b>7</b>  |
| 1.1      | Einleitung . . . . .  | 7         |
| 1.2      | Notationen . . . . .  | 9         |
| <b>2</b> | <b>Einführung in die Gittertheorie</b>                            | <b>11</b> |
| 2.1      | Grundlagen und Notationen aus der linearen Algebra . . . . .      | 11        |
| 2.2      | Grundlagen und Notationen aus der Gittertheorie . . . . .         | 12        |
| 2.3      | Das Schmidtsche Orthogonalisierungsverfahren . . . . .            | 16        |
| 2.4      | Das Duale Gitter . . . . .  | 17        |
| 2.5      | Gitterprobleme und ihre Komplexität . . . . .                     | 18        |
| <b>3</b> | <b>Grundlagen der Maß- und Integrationstheorie</b>                | <b>20</b> |
| 3.1      | Maßtheorie . . . . .  | 20        |
| 3.2      | Das Lebesgueintegral . . . . .                                    | 21        |
| 3.2.1    | Messbare Funktionen . . . . .                                     | 22        |
| 3.2.2    | Das Lebesgueintegral . . . . .                                    | 22        |
| 3.3      | Die wichtigsten Sätze der Integrationstheorie . . . . .           | 24        |
| 3.3.1    | Der Satz von Beppo Levi und der Satz von Lebesgue . . . . .       | 24        |
| 3.3.2    | Der Zusammenhang zwischen Lebesgue- und Riemannintegral . . . . . | 24        |
| 3.3.3    | Der Satz von Fubini und die Transformationsformel . . . . .       | 25        |
| 3.3.4    | Die Transformationsformel für lineare Abbildungen . . . . .       | 26        |
| 3.3.5    | Der Mittelwertsatz . . . . .                                      | 27        |
| <b>4</b> | <b>Grundbegriffe der Wahrscheinlichkeitstheorie</b>               | <b>28</b> |
| 4.1      | Wahrscheinlichkeitsräume . . . . .                                | 28        |
| 4.2      | Zufallsvariablen und Verteilungsfunktionen . . . . .              | 28        |
| 4.3      | Erwartungswert und Varianz . . . . .                              | 29        |
| 4.4      | Mehrdimensionale Zufallsvariablen, Unabhängigkeit . . . . .       | 30        |
| <b>5</b> | <b>Grundlagen der Komplexitätstheorie</b>                         | <b>33</b> |
| 5.1      | Berechenbarkeitsmodelle und Komplexität . . . . .                 | 33        |
| 5.2      | Probabilistische Polynomialzeit . . . . .                         | 35        |
| 5.3      | Reduktion von Problemen . . . . .                                 | 36        |
| 5.4      | Ununterscheidbarkeit von Verteilungen . . . . .                   | 37        |
| 5.5      | Average-Case Komplexität . . . . .                                | 39        |
| <b>6</b> | <b>Das Kryptosystem</b>   | <b>40</b> |
| 6.1      | Einige Verteilungen . . . . .                                     | 40        |
| 6.2      | Das Kryptosystem . . . . .  | 43        |
| <b>7</b> | <b>Einige technische Beweise</b>                                  | <b>44</b> |

|   |           |
|---|-----------|
| <b>8 Hauptteil</b>  | <b>57</b> |
| 8.1 Reduktion auf das dSV Problem . . . . .                     | 57        |
| 8.2 Verteilungen auf Gittern . . . . .                          | 61        |
| <b>9 Die Analyse des Verschlüsselungsverfahrens</b>             | <b>72</b> |
| <b>10 Chosen ciphertext Attacken auf das Regev-Kryptosystem</b> | <b>84</b> |
| <b>11 Grundlagen</b>  | <b>89</b> |
| 11.1 Reihen . . . . .   | 89        |
| 11.2 Funktionenfolgen . . . . .                                 | 90        |
| 11.3 g-adische Entwicklung . . . . .                            | 90        |
| 11.4 Public-Key Kryptosysteme und Sicherheitsbegriffe . . . . . | 91        |
| <b>12 Symbolverzeichnis</b>                                     | <b>93</b> |

# 1 Vorwort

## 1.1 Einleitung

Computer und Netzwerke sind in den letzten Jahren ein wichtiger Bestandteil des modernen Lebens geworden. Ihr Einsatz in sensiblen Bereichen wie zum Beispiel dem Zahlungsverkehr macht es notwendig, Verfahren zu entwickeln, die einen sicheren Datenaustausch ermöglichen. Von zentraler Bedeutung hierbei sind Public Key Kryptosysteme (PKC). Die Sicherheit eines PKC beruht auf der Schwierigkeit eines mathematischen Problems. Momentan basiert die Sicherheit fast aller gängigen PKC auf der Schwierigkeit diskrete Logarithmen zu berechnen und zu faktorisieren. Da sich die Mathematik schon seit geraumer Zeit mit diesen Problemen beschäftigt, ist man bisher davon ausgegangen, dass sie schwer zu lösen sind. Dies änderte sich erst mit der Konstruktion der Quantencomputer. Mit dem Bau des Quantencomputers wäre es möglich von Shor entwickelte Algorithmen zu implementieren, die diese Probleme in Polynomialzeit berechnen (siehe [Sho 1997]). Damit würden die gängigsten PKC unsicher. Diese Tatsache verdeutlicht die Notwendigkeit möglichst viele mathematische Probleme zu finden, die sich für die Konstruktion von Kryptosystemen eignen. Hierbei stellte sich heraus, dass Gitterprobleme eine brauchbare Alternative sind. Die Gründe hierfür sind vielschichtig, und wir werden später auf sie eingehen.

Gitter sind diskrete additive Untergruppen des  $\mathbb{R}^d$ . Man kann beweisen, dass die Elemente eines Gitters ganzzahlige Linearkombinationen von Basisvektoren sind. Die Anzahl  $n$  der Basisvektoren ist die Dimension des Gitters. Versieht man den  $\mathbb{R}^d$  mit einer Norm, üblicherweise nimmt man die euklidische, so kann man Abstände messen. Dies macht es möglich verschiedene Gitterprobleme, wie das Finden eines kürzesten Vektors (SVP) zu definieren. Obwohl Gitter schon seit Anfang des 20. Jahrhunderts untersucht wurden, ist die Relevanz für kryptographische Konstruktionen erst sehr spät erkannt worden. Zunächst lag ihr Anwendungsbereich in der Entwicklung von Angriffen (siehe [Adl 1983], [Odl 1990] u.v.m.). So musste beispielsweise die Wahl der Sicherheitsparameter von RSA und DSA aufgrund gitterbasierter Angriffe überdacht werden. Erst durch neuere Erkenntnisse über die Komplexität von Gitterproblemen wurden sie auch für die Konstruktion von Kryptosystemen interessant.

Ein Vorteil von gitterbasierten Kryptosystemen gegenüber den meisten anderen ist die vermutete oder bewiesene  $\mathcal{NP}$ -Härte vieler Gitterprobleme. Es wird davon ausgegangen, dass Probleme dieser Klasse auch im Zeitalter der Quantencomputer schwer zu lösen sind.

Der zweite große Vorteil, ist die Möglichkeit Verschlüsselungsverfahren zu entwickeln, die auf der sogenannten *worst case* Schwierigkeit eines Problems basieren. Das bedeutet, ist es jemanden möglich eine zufällige Instanz des Problems zu berechnen, so kann er jede Instanz des Gitterproblems lösen.

Obwohl bereits einige gitterbasierte Verschlüsselungsverfahren entwickelt wurden, gibt es bis jetzt nur zwei auf *worst case* Gitterproblemen basierende Verfahren. Eines wurde von Ajtai und Dwork entwickelt. Das andere wurde von Oded

Regev entwickelt und ist Thema meiner Diplomarbeit. Die Sicherheit beider Verfahren basiert auf dem Problem, einen kürzesten Vektor in einem eindeutigen Gitter zu finden dem *unique Shortest Vector Problem*(uSVP). Ein eindeutiges Gitter ist ein Gitter, in dem der kürzeste Vektor ungleich Null eindeutig bestimmt ist. Ajtai konnte in [Ajt 1996] dessen *worst case* Komplexität zeigen. Die Komplexität des uSVP hängt neben der Dimension des Gitters von dem Verhältnis der Länge des kürzesten Vektor  $v$  zu anderen, nicht zu  $v$  parallelen Vektoren, ab.

Das von Ajtai und Dwork 1997 in [Ajt 1997] vorgestellte Kryptosystem basiert auf einem uSVP, dass auf einem sogenannten  $O(n^8)$ -eindeutigem Gitter gestellt ist. Ein  $O(n^8)$ -eindeutiges Gitter ist ein Gitter, in dem alle nicht zu dem kürzesten Vektor  $v$  parallelen Vektoren mindestens um einen Faktor in der Größenordnung  $O(n^8)$  größer als  $v$  sind. Goldreich, Goldwasser und Halevi beseitigten in [Gol 1997] Entschlüsselungsfehler und verbesserten die Sicherheit auf das  $O(n^7)$  – uSVP. Obwohl Ajtai in [Ajt 1998] die NP- Härte des SVP beweisen konnte, liegen keine Ergebnisse über die Komplexität des uSVP vor.

Ein großes Sicherheitsproblem für das Ajtai-Dwork Kryptosystem stellen allerdings verbesserte Gitterreduktionsalgorithmen und die verbesserte Approximation des SVP dar. Daher gilt das  $O(n^7)$  – uSVP als nicht sicher.

Regev hat ein Verschlüsselungsverfahren konstruiert, dessen Sicherheit auf dem  $O(n^{1,5})$  – uSVP basiert. Dies bedeutet eine erhebliche Verbesserung der Sicherheit und es könnte sein, dass dieses Problem trotz der verbesserten Reduktions- und Approximationsalgorithmen schwer zu lösen ist.

Ein Nachteil von allen bisher entwickelten auf Gitterproblemen basierenden Verschlüsselungsverfahren ist, dass sie entweder unsicher oder ineffizient sind. Sie gelten daher als nicht praktikabel.

Ziel der Diplomarbeit ist es, den Beweis der Sicherheit des Kryptosystems auf Korrektheit und die Effizienz des Verfahrens zu untersuchen. Dies geschieht, indem Schranken für die Größe der in dem Beweis auftretenden Konstanten bestimmt werden, die entscheidend für die Effizienz sind. Zusätzlich konnte noch eine nichtadaptive chosen ciphertext Attacke auf das Regev Kryptosystem entwickelt werden.

## **Gliederung und Inhalt dieser Diplomarbeit**

Um den Beweis der Sicherheit des Verfahrens nachvollziehen zu können, werden Erkenntnisse aus verschiedenen mathematischen Gebieten benötigt. In den ersten Kapiteln werden die benötigten Ergebnisse der Gittertheorie, der Maß- und Integrationstheorie, der Wahrscheinlichkeitstheorie und der Komplexitätstheorie zusammengefasst.

In Kapitel 6 wird das Verschlüsselungsverfahren vorgestellt.

Im nachfolgenden Kapitel werden technische Aussagen bewiesen, die für den Beweis der Sicherheit des Verfahrens benötigt werden.

Das Kapitel 8 ist der zentrale Teil des Beweises der Sicherheit des Kryptosystems. Hier wird bewiesen, wie man mit Hilfe von Ergebnissen aus der harmoni-

schen Analysis das uSVP auf die Unterscheidbarkeit von Verteilungen reduzieren kann.

Die Reduktion erfolgt im wesentlichen in drei Schritten. Zunächst wird das uSVP mittels einer Cookreduktion auf ein Entscheidungsproblem reduziert. Angenommen der kürzeste Vektor hat die Darstellung  $a_1, \dots, a_n$  mit  $a_i \in \mathbb{Z}$  zu einer beliebigen Basis eines Gitters. Das Entscheidungsproblem fragt, ob der  $i$ -te Koeffizient von einer Primzahl  $p$  geteilt wird. Die Idee des Beweises ist, dass Gitter immer größer werden zu lassen, ohne dabei den kürzesten Vektor zu verändern. Am Ende lässt sich der kürzeste Vektor einfach ablesen.

Im zweiten Schritt wird dieses Problem mittels Ergebnissen aus der harmonischen Analysis auf die Unterscheidbarkeit zweier  $n$ -dimensionaler Verteilungen auf einem  $n$ -dimensionalen Parallelogramm reduziert. Dies geschieht mittels einer Karpreduktion.

Die Verteilungen werden im letzten Schritt ins Eindimensionale transformiert. Damit durch die Projektion nicht zu viele Informationen verlorengehen, müssen die Bereiche, die auf den gleichen Punkt abgebildet werden, sehr klein sein. Dazu werden die Punkte auf eine Linie projiziert, die das Parallelogramm  $K$ -mal durchläuft. Die Größe der Zahl  $K$  bestimmt die Größe der Elemente des öffentlichen Schlüssels. Zur Beantwortung der Frage, wie eine Gerade mehrmals ein Parallelogramm durchläuft verweisen wir auf Kapitel 8. Die Reduktion vom  $n$ -Dimensionalen ins Eindimensionale ist neuartig und könnte durchaus noch weitere Anwendungen finden.

In Kapitel 9 wird die Sicherheit des Kryptosystem bewiesen. Aus dem Beweis ergibt sich die Anzahl der Elemente des öffentlichen Schlüssels. Man nimmt an, dass ein Algorithmus  $\mathcal{A}$  mit nicht zu vernachlässigender Wahrscheinlichkeit Verschlüsselungen der Null von denen der Eins unterscheiden kann. Aus  $\mathcal{A}$  lässt sich ein Algorithmus konstruieren, der mit nicht zu vernachlässigender Wahrscheinlichkeit zwischen zwei Verteilungen unterscheiden kann, auf deren Unterscheidbarkeit man das uSVP in Kapitel 8 reduziert hat.

In Kapitel 10 wird ein Angriff auf das Kryptosystem vorgestellt und bewiesen.

## 1.2 Notationen

**Definition 1.1**  $[n]$  bezeichnet die Menge  $\{1, \dots, n\}$ .

**Definition 1.2** Seien  $x, y \in \mathbb{R}$ . Die Zahl  $x \bmod y$  ist definiert durch

$$x - \lfloor x/y \rfloor y.$$

**Definition 1.3** Sei  $x \in \mathbb{R}$ . Die Zahl  $\lfloor x \rfloor$  sei die am nächsten zu  $x$  liegende ganze Zahl. Existieren zwei solcher Zahlen so bezeichnet  $\lfloor x \rfloor$  die kleinere.

**Definition 1.4** Sei  $x \in \mathbb{R}$ . Durch

$$\text{frc}(x) := |x - \lfloor x \rfloor|$$

ist der Abstand von  $x$  zu der am nächsten an  $x$  liegenden ganzen Zahl definiert.

Offensichtlich nimmt  $\text{frc}(x)$  Werte zwischen null und  $\frac{1}{2}$  an. Eine einfache Anwendung der Dreiecksungleichung ergibt folgendes Lemma.

**Lemma 1.5** *Es gilt*  
 $\text{frc}(a + b) \leq \text{frc}(a) + \text{frc}(b)$  und  
 $\text{frc}(a - b) \geq \text{frc}(a) - \text{frc}(b)$ .

**Definition 1.6** *Mit  $\log$  bezeichnen wir den Logarithmus zur Basis 2.*

**Definition 1.7** *Man sagt, dass ein Algorithmus zwischen einer Verteilung  $A$  und einer Menge von Verteilungen  $\Upsilon$  unterscheiden kann, wenn er  $v \in \Upsilon$  von  $A$  unterscheiden kann.*

**Definition 1.8** *Sei  $B$  eine Menge und  $A \subseteq B$ . Die Funktion  $\chi_A : B \rightarrow \{0, 1\}$  ist definiert durch*

$$\chi_A(x) := \begin{cases} 1 & \text{falls } x \in A, \\ 0 & \text{sonst} \end{cases} .$$

Nun wollen wir noch die O-Notation einführen.

**Definition 1.9** *Sei  $g : \mathbb{N} \rightarrow \mathbb{R}^+$  eine Funktion. Dann ist*

$$O(g) := \{f : \mathbb{N} \rightarrow \mathbb{R}^+ \mid (\exists c, n_0 > 0)(\forall n \geq n_0) f(n) \leq c \cdot g(n)\}.$$

**Definition 1.10** *Sei  $g : \mathbb{N} \rightarrow \mathbb{R}^+$  eine Funktion. Dann ist*

$$\Omega(g) := \{f : \mathbb{N} \rightarrow \mathbb{R}^+ \mid (\exists c, n_0 > 0)(\forall n \geq n_0) f(n) \geq c \cdot g(n)\}.$$

## 2 Einführung in die Gittertheorie

### 2.1 Grundlagen und Notationen aus der linearen Algebra

**Definition 2.1** Sei  $V$  ein reeller Vektorraum und  $A$  eine nichtleere Teilmenge von  $V$ . Mit  $\text{span}(A)$  bezeichnet man alle Linearkombinationen von Elementen aus  $A$ .

$$\text{span}A := \{\lambda_1 a_1 + \lambda_2 a_2 + \dots + \lambda_n a_n \mid \lambda_i \in \mathbb{R}, a_i \in A\}$$

Man nennt  $\text{span}(A)$  das Erzeugnis oder den Spann von  $A$ .

**Definition 2.2 (Norm)** Sei  $V$  ein reeller Vektorraum. Eine Abbildung  $p : V \mapsto [0, \infty)$  heißt Norm, falls

- (a)  $p(\lambda v) = |\lambda|p(v) \quad \forall \lambda \in \mathbb{R}, v \in V$ ,
- (b)  $p(v + w) \leq p(v) + p(w) \quad \forall v, w \in V$
- (c)  $p(v) = 0 \Rightarrow v = 0$

gilt. Man nennt  $(V, p)$  einen normierten Raum.

**Definition 2.3 (Skalarprodukt)** Sei  $V$  ein reeller Vektorraum. Eine Abbildung  $\langle \cdot, \cdot \rangle : V \times V \rightarrow [0, \infty)$  heißt Skalarprodukt, falls

- (a)  $\langle v_1 + v_2, w \rangle = \langle v_1, w \rangle + \langle v_2, w \rangle \quad \forall v_i \in V, w \in V$
- (b)  $\langle \lambda v, w \rangle = \lambda \langle v, w \rangle \quad \forall v, w \in V, \lambda \in \mathbb{R}$
- (c)  $\langle v, v \rangle = 0 \Leftrightarrow v = 0$

Durch  $\|v\| = \sqrt{\langle v, v \rangle}$  erhält man zu jedem Skalarprodukt auf einem Vektorraum  $V$  eine Norm (Lemma V.1.3 in [Wer 2000]). Normierte Räume, deren Norm über ein Skalarprodukt definiert werden kann, heißen euklidische Räume.

**Definition 2.4 (Standardskalarprodukt)** Seien  $u = (u_1, u_2, \dots, u_n)$  und  $v = (v_1, v_2, \dots, v_n)$  Vektoren aus dem  $\mathbb{R}^n$ . Durch

$$\langle u, v \rangle := u^\top v = \sum_{i=1}^n u_i v_i$$

wird das Standardskalarprodukt auf dem  $\mathbb{R}^n$  definiert.

Mit dem Standardskalarprodukt wird der  $\mathbb{R}^n$  zu einem euklidischen Vektorraum. Wenn wir im Folgenden von Längen, Abständen usw. reden werden, sind diese über das Standardskalarprodukt definiert.

**Definition 2.5 (Orthogonales Komplement)** Sei  $U$  eine Teilmenge eines euklidischen Vektorraumes  $V$ . Die Menge

$$U^\perp := \{v \in V \mid \langle v, u \rangle = 0 \text{ für alle } u \in U\}$$

heißt orthogonales Komplement von  $U$ . Betrachtet man das Standardskalarprodukt auf  $V$ , so ist der Winkel zwischen einem Element aus  $U$  und einem aus dem orthogonalen Komplement 90 Grad.

Die Vorstellung, dass zwei Vektoren ein Parallelogramm aufspannen und drei Vektoren einen Quader, kann man durch folgende Definition auch auf das  $n$ -dimensionale übertragen.

**Definition 2.6 (Parallelepiped)** Seien  $v_1, v_2, \dots, v_k \in \mathbb{R}^n$ . Dann nennen wir die Menge

$$\mathcal{P}(v_1, v_2, \dots, v_k) := \left\{ \sum_{i=1}^k \lambda_i v_i \mid \lambda_i \in [0, 1] \right\}$$

das von den Vektoren  $v_1, v_2, \dots, v_k$  aufgespannte Parallelepiped.

**Definition 2.7** Sei  $\mathcal{P}(v_1, \dots, v_n)$  ein Parallelepiped und  $x \in \mathbb{R}^n$ .  $x$  modulo  $\mathcal{P}(v_1, \dots, v_n)$  bezeichnet den eindeutig bestimmten Punkt  $y \in \mathcal{P}(v_1, \dots, v_n)$ , so dass  $y - x$  eine ganzzahlige Linearkombination der Vektoren  $(v_1, \dots, v_n)$  ist.

Im Gegensatz zu einer Kugel und einem Würfel ist der Durchmesser einer beliebigen Menge  $A$  nicht kanonisch. Wir wollen den Durchmesser als die längste in  $A$  liegende Linie definieren.

**Definition 2.8 (Durchmesser)** Sei  $A \subseteq \mathbb{R}^n$  eine beschränkte Menge. Dann heißt

$$\text{diam}(A) = \sup\{\|x - y\| \mid x, y \in A\}$$

Durchmesser von  $A$ .

**Bemerkung 2.9** Da die Länge der Vektoren in  $\mathcal{P}(v_1, \dots, v_n)$  durch  $\sum_{i=1}^n \|v_i\|$  beschränkt ist, ist der Durchmesser eines Parallelepipeds höchstens  $\sum_{i=1}^n \|v_i\|$ .

**Satz 2.10 (Volumen)** Seien  $b_1, b_2, \dots, b_n \in \mathbb{R}^d$  und  $\hat{b}_1, \hat{b}_2, \dots, \hat{b}_n \in \mathbb{R}^d$  die zugehörigen Orthogonalvektoren (siehe Abschnitt 2.3). Dann gilt für das  $n$ -dimensionale Volumen  $\text{vol}_n$  des Parallelepipeds  $\mathcal{P}(b_1, b_2, \dots, b_n)$ :

$$\text{vol}_n(\mathcal{P}(b_1, b_2, \dots, b_n)) = \prod_{i=1}^n \|\hat{b}_i\|$$

Beweis: siehe [Schn].

Ein konstruktives Verfahren zur Berechnung der Vektoren  $\hat{b}_i$  ist das *Schmidtsche Orthogonalisierungsverfahren* (siehe Kapitel 2.3). Wir werden später sehen, dass das Volumen auch mit Hilfe der Determinante berechnet werden kann.

## 2.2 Grundlagen und Notationen aus der Gittertheorie

**Definition 2.11 (Gitter)** Seien  $b_1, \dots, b_n$  linear unabhängige Vektoren im  $\mathbb{R}^d$ . Die Menge

$$L := L(b_1, \dots, b_n) := \left\{ \sum_{i=1}^n t_i b_i \mid t_i \in \mathbb{Z} \right\}$$

heißt Gitter mit Basis  $b_1, b_2, \dots, b_n$  und Rang  $n$ . Die Matrix

$$B := [b_1 \ b_2 \ \dots \ b_n]$$

heißt Basismatrix.  $L$  ist das von den Spalten der Matrix erzeugte Gitter.

Da Basisvektoren linear unabhängig sein müssen, darf  $n$  nicht größer als  $d$  sein, da  $d + 1$  Vektoren im  $\mathbb{R}^d$  stets linear abhängig sind. Ein wichtiger Spezialfall sind Gitter, deren Rang mit der Dimension des Raumes übereinstimmt. Solche Gitter nennt man vollständig oder volldimensional.

Eine andere Möglichkeit ein Gitter zu definieren liefert der nächste Satz. Man kann ein Gitter nämlich auch als Untergruppe des  $\mathbb{R}^n$  ohne Häufungspunkt definieren. Allerdings bleibt dann noch zu zeigen, dass eine Gitterbasis existiert. Da man sich unter einem Gitter das Erzeugnis von Vektoren vorstellt, ist unsere Definition leichter zugänglich.

**Satz 2.12** Sei  $L \subseteq \mathbb{R}^d$  eine additive Untergruppe. Folgende Aussagen sind äquivalent:

- (i)  $L$  hat keinen Häufungspunkt.
- (ii)  $L$  ist ein Gitter.

Beweis: siehe [Schn].

**Definition 2.13 (Untergitter)** Seien  $L, L' \subseteq \mathbb{R}^d$  Gitter.  $L'$  heißt Untergitter von  $L$ , falls  $L' \subseteq L$ . Ist  $\text{Rang}(L) = \text{Rang}(L')$ , so hat  $L'$  vollen Rang.

Es stellt sich die Frage, welche Basistransformationen ohne das Gitter zu verändern durchgeführt werden können. Im Reellen dürfen Basen von Vektorräumen, ohne den Vektorraum zu verändern, mit beliebigen invertierbaren Matrizen multipliziert werden. Gitter bleiben nur unter Anwendung einer Teilmenge der invertierbaren Matrizen den unimodularen Matrizen unverändert.

**Definition 2.14 (Unimodulare Matrizen)** Die unimodularen Matrizen sind die Elemente der Menge

$$SL_n(\mathbb{Z}) := \{U \in \mathbb{Z}^{n \times n} \mid \det(U) = \pm 1\}.$$

**Satz 2.15** Die Wirkung einer unimodularen Matrix  $T$  auf eine Menge von Vektoren  $b_1, \dots, b_n$  lässt sich durch die Hintereinanderausführung von den Operationen

- (i) Ersetze den Vektor  $b_i$  durch  $-b_i$ ,
  - (ii) ersetze den Vektor  $b_i$  durch den Vektor  $b_i + b_j$  mit  $j \neq i$  und
  - (iii) vertausche den Vektor  $b_i$  und  $b_j$
- beschreiben.

Beweis: siehe [Schn].

**Satz 2.16** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter vom Rang  $n$  mit Basis  $B$ . Eine  $d \times n$  Matrix  $B'$  ist genau dann eine Basis von  $L$ , wenn eine unimodulare Matrix  $T \in GL_n(\mathbb{Z})$  mit  $B' = BT$  existiert.

Beweis: siehe [Schn].

Aus den beiden vorherigen Sätzen wissen wir, durch welche Grundoperationen wir Basen ineinander überführen können.

**Satz 2.17** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter vom Rang  $n$ . Seien  $B = (b_1, b_2, \dots, b_n)$  und  $B' = (b_1', b_2', \dots, b_n')$  Basen von  $L$ . Dann lässt sich  $B$  in  $B'$  durch folgende drei Basistransformationen überführen.

(i) Ersetze den Vektor  $b_i$  durch  $-b_i$ .

(ii) Ersetze den Vektor  $b_i$  durch den Vektor  $b_i + b_j$  mit  $j \neq i$ .

(iii) Vertausche die Vektoren  $b_i$  und  $b_j$ .

Beweis: siehe [Schn].

**Definition 2.18 (Gitterdeterminante)** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter mit Basis  $B$ . Die Gitterdeterminante ist definiert durch

$$\det(L) := (\det(B^\top B))^{\frac{1}{2}}.$$

Da die Menge der unimodularen Matrizen für  $n > 1$  unendlich groß ist, gibt es zu jedem Gitter, dessen Rang größer 1 ist, unendlich viele Basen. Die Determinantenfunktion hat die Eigenschaften  $\det(T^\top) = \det(T)$  und  $\det(AL) = \det(A)\det(L)$  (siehe [Beu 1998] 7.5 und 7.6). Damit gilt

$$((BT)^\top BT) = \det(T^\top)\det(B^\top B)\det(T).$$

Aus Satz 2.16 folgt, dass die Gitterdeterminante unabhängig von der Wahl der Basis ist.

**Definition 2.19 (Grundmasche)** Sei  $B = (b_1, \dots, b_n)$  eine Basis eines Gitters  $L$ . Das Parallelepiped

$$\mathcal{P}(b_1, \dots, b_n) = \left\{ \sum_{i=1}^n \lambda_i b_i \mid \lambda_i \in [0, 1) \right\}$$

heißt Grundmasche von  $B$ .

Die Grundmasche ist also ein  $n$ -dimensionales Parallelepiped. Folgender Satz stellt einen wichtigen Zusammenhang zwischen Gitterdeterminante und Volumen der Grundmasche her.

**Satz 2.20** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter vom Rang  $n$  und  $B = (b_1, \dots, b_n)$  eine Basis von  $L$ . Dann gilt

$$\det(L) = \text{vol}_n(\mathcal{P}(b_1, \dots, b_n)).$$

Beweis: siehe [Schn].

Für einen Vektorraum gibt es den Basisergänzungssatz. Dieser besagt, dass eine linear unabhängige Menge von Vektoren zu einer Basis ergänzt werden kann. Für Gitter kann man unter bestimmten Voraussetzungen eine analoge Aussage beweisen.

**Satz 2.21** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter vom Rang  $n$  und  $v \in L$  ein Vektor ungleich null. Dann existiert ein  $k \in \mathbb{N}$  und Vektoren  $b_2, \dots, b_n$ , so dass  $B = (\frac{1}{k}v, b_2, \dots, b_n)$  eine Basis von  $L$  ist.

Beweis: Beweis mittels vollständiger Induktion über den Rang  $n$  des Gitters.

$n = 1$ : Das Gitter wird von einem Vektor  $b$  erzeugt. Es gibt eine ganze Zahl  $k \neq 0$  mit  $b = k \cdot v$ . Dann ist  $\frac{1}{|k|}b$  eine Basis von  $L$ .

Induktionsschritt: Sei  $n > 1$ . Angenommen die Aussage stimmt für alle Gitter deren Rang kleiner als  $n$  ist.

Sei  $G = (g_1, g_2, \dots, g_n)$  eine Basis von  $L$ . Dann existieren  $k_1, \dots, k_n \in \mathbb{Z}$  mit  $v = k_1g_1 + k_2g_2 + \dots + k_n g_n$ .

Es gibt zwei verschiedene Fälle. Im ersten Fall ist einer der Koeffizienten  $k_i$  null. Dann bilden die restlichen Vektoren ein  $(n - 1)$ -dimensionales Gitter für das nach Induktionsannahme eine Basis mit den gewünschten Eigenschaften existiert. Wenn wir diese Basis um den Vektor  $g_i$  ergänzen, erhalten wir nach Induktionsannahme eine Basis  $B$  von  $L$  mit  $\frac{1}{|k_i|}v \in B$ .

Im zweiten Fall sind alle Koeffizienten  $k_i$  ungleich 0. Wir versuchen diesen Fall auf den ersten zu reduzieren in dem wir die Basis so verändern, dass der erste Koeffizient 0 ist.

Sei  $k = \text{ggT}(k_1, k_2)$ ,  $\tilde{k}_1 = \frac{k_1}{k}$  und  $\tilde{k}_2 = \frac{k_2}{k}$ . Da  $\tilde{k}_1$  und  $\tilde{k}_2$  teilerfremd sind existieren nach dem erweiterten Satz von Euklid (Satz 11.1) ganze Zahlen  $c_1$  und  $c_2$  mit

$$c_1\tilde{k}_1 + c_2\tilde{k}_2 = 1.$$

Die Matrix  $A = \begin{pmatrix} c_2 & -c_1 & 0 & \cdots & 0 \\ \tilde{k}_1 & \tilde{k}_2 & 0 & & \vdots \\ 0 & 0 & 1 & \ddots & \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & & 0 & 1 \end{pmatrix}$

hat also Determinante 1 und Einträge aus den ganzen Zahlen. Nach Satz 2.14 beschreibt  $A$  eine Basistransformation. Die transformierte Basis hat folgende Gestalt.

$$B = (b_1, \dots, b_n) = AG = (-c_1g_1 + c_2g_2, \tilde{k}_1g_1 + \tilde{k}_2g_2, g_3, \dots, g_n).$$

Der Vektor  $v$  zu dieser Basis hat die Darstellung

$$\begin{aligned} v &= k_1g_1 + k_2g_2 + \dots + k_n g_n \\ &= k(\tilde{k}_1g_1 + \tilde{k}_2g_2) + k_3g_3 + \dots + k_n g_n \\ &= 0 \cdot b_1 + kb_2 + k_3b_3 + \dots + k_nb_n. \end{aligned}$$

Wir haben also eine Basis gefunden, für die der Vektor  $v$  in einem  $(n - 1)$ -dimensionalen Unterraum liegt. Nach Induktionsannahme ist die Aussage bewiesen.

### 2.3 Das Schmidtsche Orthogonalisierungsverfahren

In der Gittertheorie ist es von Interesse spezielle Basen eines Gitters zu finden. Wichtig hierbei sind sogenannte langenreduzierte Basen. Die Idee bei der Langenreduzierung ist, eine moglichst orthogonale Basis eines Gitters zu finden. Da die Determinante eines Gitters eindeutig bestimmt ist und bei deren Berechnung die Winkel zwischen den Vektoren eingehen, erhalt man eine „kurze“ Basis des Gitters, wenn die Basisvektoren moglichst senkrecht aufeinander stehen. Mit Hilfe des *Schmidtschen Orthogonalisierungsverfahren* lassen sich Verfahren konstruieren, die Basen in langenreduzierte Basen transformieren (siehe Kapitel 7).

**Definition 2.22 (Orthogonale Projektion)** Sei  $B = (b_1, \dots, b_n) \in \mathbb{R}^d$  eine Basis von  $L$ . Die Funktionen

$$\pi_i : \mathbb{R}^d \rightarrow \text{span}(b_1, \dots, b_{i-1})^\perp \text{ mit } i = 2, \dots, n$$

seien die Projektionen in den zu  $\text{span}(b_1, \dots, b_{i-1})$  orthogonalen Raum.

Mit dem *Schmidtschen Orthogonalisierungsverfahren* ordnet man jeder Basis eines Gitters  $L \subseteq \mathbb{R}^d$  ein *Orthogonalsystem*  $\hat{b}_1, \dots, \hat{b}_n$  mit Hilfe der orthogonalen Projektionen zu. Die Berechnungsvorschrift ist

$$\hat{b}_i := \pi_i(b_i) = b_i - \sum_{j=1}^{i-1} \frac{\langle b_i, \hat{b}_j \rangle}{\|\hat{b}_j\|^2} \hat{b}_j. \quad (1)$$

Der Vektor  $\hat{b}_i$  ist also die Projektion des  $i$ -ten Basisvektor in den Raum  $\text{span}(b_1, \dots, b_{i-1})^\perp$ . Offensichtlich stehen die Vektoren  $\hat{b}_i$  und  $\hat{b}_j$  fur  $i \neq j$  senkrecht aufeinander.

Die Vektoren  $\hat{b}_i$  mit  $i = 1, \dots, n$  bilden ein Gitter  $\tilde{L}$  mit dem gleichen Rang wie das Ausgangsgitter  $L$ . Im Allgemeinen sind diese Gitter nicht gleich. Um aus der Basis von  $\tilde{B} = (\hat{b}_1, \dots, \hat{b}_n)$  wieder  $B = (b_1, \dots, b_n)$  zu berechnen, benotigt man die sogenannten *Gram-Schmidt Koeffizienten*. Diese sind definiert durch

$$\mu_{i,j} := \begin{cases} 0 & \text{falls } i < j \\ 1 & \text{falls } i = j \\ \frac{\langle b_i, \hat{b}_j \rangle}{\|\hat{b}_j\|^2} & \text{falls } i > j \end{cases}$$

Wenn man Gleichung (1) umstellt erhalt man folgende Beziehung

$$b_i = \sum_{j=1}^n \mu_{i,j} \hat{b}_j.$$

Stellt man diesen Zusammenhang mit Hilfe der Matrixschreibweise dar erhält man,

$$[b_1 \cdots b_n] = [\hat{b}_1 \cdots \hat{b}_n] \cdot \begin{pmatrix} 1 & \mu_{2,1} & \cdots & \mu_{n-1,1} & \mu_{n,1} \\ 0 & 1 & & \mu_{n-1,2} & \mu_{n,2} \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & & 0 & 1 & \mu_{n,n-1} \\ 0 & \cdots & 0 & 0 & 1 \end{pmatrix}. \quad (2)$$

## 2.4 Das Duale Gitter

Von großem Interesse für uns wird das zu  $L$  duale Gitter  $L^*$  sein. Deshalb werden in diesem Kapitel die wichtigsten Eigenschaften des dualen Gitters zusammengefasst.

**Definition 2.23 (Kroneckerdelta)** Das Kroneckerdelta  $\delta_{ij}$  bezeichnet die Abbildung

$$\delta_{ij} : [n] \times [n] \rightarrow \{0, 1\}, \quad (i, j) \mapsto \begin{cases} 1 & \text{falls } i = j \\ 0 & \text{sonst} \end{cases}.$$

**Definition 2.24 (Duale Gitter)** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter mit Basis  $B$ . Das duale Gitter  $L^*$  sind die Punkte  $x \in \text{span}(B)$  mit  $\langle x, y \rangle \in \mathbb{Z}$  für alle  $y \in L$ . Ein Gitter  $L$  mit  $L = L^*$  heißt auch selbstdual oder unimodular.

Das duale Gitter wird auch als reziprokes oder polares Gitter bezeichnet. Ein triviales Beispiel eines selbstdualen Gitters ist  $\mathbb{Z}^n$ .

**Definition 2.25 (Duale Basis)** Sei  $B = (b_1, \dots, b_n)$  eine Basis eines  $n$ -dimensionalen Gitters  $L$ . Die Spalten der Matrix  $b_1^*, \dots, b_n^*$  heißen duale Basis. Nach Lemma 2.25 bilden diese Vektoren tatsächlich eine Basis des dualen Gitters.

**Satz 2.26** Sei  $B$  eine Basis eines  $n$ -dimensionalen Gitters  $L$ . Dann ist  $B^* = (B^{-1})^\top$  eine Basis des dualen Gitters  $L^*$ .

Beweis: siehe [Schn].

Wegen  $B^\top (B^{-1})^\top = E$  muss das Skalarprodukt der  $i$ -ten Spalte von  $B$  und der  $j$ -ten Spalte von  $(B^{-1})^\top$  Eins falls  $i = j$  und Null sonst sein. Aus Satz 2.26 folgt also:

**Lemma 2.27** Sei  $B = (b_1, \dots, b_n)$  eine Basis eines  $n$ -dimensionalen Gitters  $L$ . Dann bilden die Vektoren  $b_1^*, \dots, b_n^*$  mit  $\langle b_i, b_j^* \rangle = \delta_{ij}$  eine Basis des dualen Gitters.

**Satz 2.28** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter. Dann gilt

- (i)  $\det(L^*) = \frac{1}{\det(L)}$
- (ii)  $(L^*)^* = L$
- (iii)  $L$  und  $L^*$  haben den gleichen Rang.

Beweis: siehe [Schm].

**Bemerkung 2.29** Sei  $L$  ein volldimensionalen Gitters. Aus Satz 2.21 folgt, dass für jeden Vektor  $v \neq 0 \in L$  eine Basis  $B$  und eine natürliche Zahl  $k$  so existieren, dass  $\frac{1}{k}v$  ein Basisvektor von  $B$  ist. Sei  $B^*$  die zu  $B$  duale Basis. Dann stehen alle Vektoren bis auf den Vektor  $(\frac{1}{k}v)^*$  aus  $B^*$  senkrecht auf dem Vektor  $v$ . Also existiert eine Basis von  $L^*$  bestehend aus einem Vektor  $w \in L^*$  mit  $\langle w, v \rangle = k \in \mathbb{N}$  und Vektoren  $b_2^*, \dots, b_n^*$ , die alle senkrecht auf  $v$  stehen.

## 2.5 Gitterprobleme und ihre Komplexität

In diesem Kapitel werden wir eine kurze Zusammenfassung der wichtigsten Gitterprobleme angeben und das Gitterproblem vorstellen, auf dem die Sicherheit unseres Kryptosystems beruht.

**Definition 2.30** Sei  $L \subseteq \mathbb{R}^d$  ein Gitter und  $v$  ein kürzester Vektor ungleich 0 in  $L$ . Mit  $\lambda_1(L)$  bezeichnen wir die Länge von  $v$ .

Die wichtigsten Gitterprobleme sind:

### Shortest Vector Problem (SVP):

Dieses Problem wurde erstmals von Dirichlet im Jahre 1842 formuliert. Die Problemstellung lautet folgendermaßen:

Gegeben sei eine Basis  $B = (b_1, b_2, \dots, b_n)$  von  $L \subseteq \mathbb{Q}^d$ . Finde einen Vektor  $v \in L$  mit  $\|v\| = \lambda_1(L)$ . Man kann dieses Problem natürlich wie alle anderen auch zu jeder beliebigen Norm betrachten.

### Closest Vector Problem (CVP):

Gegeben sei eine Basis eines Gitters  $L \subseteq \mathbb{Q}^d$  und ein Vektor  $v \in \mathbb{R}^d$ . Finde einen Gittervektor  $u \in L$  mit  $\|u - v\| = \min\{\|u - w\| \mid w \in L\}$ .

### Shortest Basis Problem (SBP):

Was eine kürzeste Basis ist, ist nicht kanonisch. Deshalb gibt es verschiedene SBP Varianten. Die Gängigste ist:

Sei  $L \subseteq \mathbb{Q}^d$  ein Gitter und  $M$  die Menge aller Basen von  $L$ . Finde eine Gitterbasis  $B = (b_1, b_2, \dots, b_n)$ , so dass

$$\max_{1 \leq i \leq n} \|b_i\| = \inf_{B'=(b'_1, \dots, b'_n) \in M} \max_{1 \leq i \leq n} \|b'_i\|.$$

Ajtai hat in [Ajt 1998] gezeigt, dass das SVP unter randomisierter Reduktion NP-hart ist. Bereits länger bekannt ist die NP-Härte der anderen beiden Probleme (siehe dazu [Cai 1999]).

Nun wollen wir zu der Problemstellung kommen auf der die Sicherheit des von Regev in [Reg 2003] vorgestellte Kryptosystem basiert. Es handelt sich hierbei um ein SVP, dass auf einer bestimmten Klasse von Gittern, den so genannten  $f(n)$ -eindeutigen Gittern, gestellt ist. Man geht davon aus, dass dieses Problem schwierig zu lösen ist.

**Definition 2.31 (*Eindeutigkeit*)** Ein Gitter  $L$  heißt *eindeutig*, wenn sein kürzester Vektor eindeutig bestimmt ist. Diesen werden wir mit  $\tau(L)$  bezeichnen.

**Definition 2.32 ( *$f(n)$ -Eindeutigkeit*)** Ein eindeutiges Gitter  $L$  heißt  $f(n)$ -eindeutig, falls sein eindeutig bestimmter kürzester Vektor  $\tau(L)$  mindestens um den Faktor  $f(n)$  kürzer als alle nicht zu  $\tau(L)$  parallelen Vektoren ist.

**Definition 2.33 (*unique Shortest Vector Problem (uSVP)*)** Gegeben sei eine Basis eines  $f(n)$ -eindeutigen Gitters  $L \subseteq \mathbb{Q}^d$ . Finde einen Vektor  $v \in L$  mit  $\|v\| = \lambda_1(L)$ .

In Kapitel 8 werden wir beweisen, dass es genauso schwierig ist das uSVP zu lösen, wie zwischen zwei Mengen von Verteilungen zu unterscheiden. Dieser Beweis geht in zwei Schritten. Im ersten Schritt das uSVP auf folgendes Problem reduziert.

**Definition 2.34 (*dSVP<sub>p</sub>*)** Das decision SVP mit Parameter  $p$  ( $\text{dSVP}_p$ ): Sei  $p \in \mathbb{N}$ . Gegeben sei eine Basis  $B$  eines  $f(n)$ -eindeutigen Gitter  $L \subseteq \mathbb{Q}^d$  und eine Zahl  $\alpha$  mit  $\lambda_1(L) < \alpha \leq 2\lambda_1(L)$ . Seien  $a_1, a_2, \dots, a_n$  die Koeffizienten des kürzesten Vektors. Entscheide ob  $p$  den ersten Koeffizienten  $a_1$  teilt oder nicht.

Es ist klar, dass es genauso schwierig ist dieses Problem zu lösen, wie zu entscheiden, ob  $p$  einen anderen Koeffizienten des kürzesten Vektors teilt. Es muss nur die Reihenfolge der Basisvektoren verändert und dann das  $\text{dSVP}_p$  gelöst werden.

### 3 Grundlagen der Maß- und Integrationstheorie

In den späteren Kapiteln wollen wir den Wert von Integralen bestimmen. Um dies tun zu können, benötigen wir einige zentrale Ergebnisse aus der Maß- und Integrationstheorie. In diesem Kapitel wollen wir diese Ergebnisse herleiten und den Zusammenhang zwischen Riemann- und Lebesgueintegral klären.

Mit  $\mathbb{K}$  bezeichnen wir entweder  $\mathbb{C}$  oder  $\mathbb{R}$ .

#### 3.1 Maßtheorie

Um das Lebesgueintegral definieren zu können müssen wir kurz die zentralen Ergebnisse der Maßtheorie zusammenfassen. Es gibt zwei zentrale Ziele in der Maßtheorie.

- Das erste Ziel ist es Inhalte ausrechnen, d.h. Längen für  $n = 1$ , Flächen für  $n = 2$  und Volumina für  $n = 3$ .
- Man will Integrale von Funktionen ausrechnen. Um dies sinnvoll machen zu können, müssen Mengen einen Inhalt zugeordnet werden.

Seit Ende des 19. Jahrhunderts beschäftigt man sich mit allgemeinen Mengen und Funktionen. 1905 fand Vitali eine Teilmenge der reellen Zahlen, der kein vernünftiger Inhalt zugeordnet werden kann.

Deshalb ist es eine zentrale Aufgabe Mengen zu finden, denen man Inhalte zuzuordnen kann. Borel und Lebesgue gingen hierbei von Quadern aus, deren Inhalt bekannt ist. Ihr Ziel war es, aus diesen Grundmengen ein möglichst großes System von sogenannten messbaren Mengen zu definieren, mit Hilfe derer man das Volumen einer Menge bestimmen kann. Dieses Ziel steckt hinter der Definition einer  $\sigma$ -Algebra. Ihre Elemente bilden die messbaren Mengen.

**Definition 3.1 ( $\sigma$ -Algebra)** Sei  $X$  eine Menge. Ein System  $\Sigma$  von Teilmengen von  $X$  heißt  $\sigma$ -Algebra, falls

- (a)  $X \in \Sigma$ ,
- (b)  $A \in \Sigma \Rightarrow X \setminus A \in \Sigma$  und
- (c) Sei  $(A_k)_{k \in \mathbb{N}}$  eine Folge von Mengen aus  $\Sigma$ . Dann ist auch die Menge  $\bigcup_{k \in \mathbb{N}} A_k$  in  $\Sigma$ .

erfüllt sind. Das Paar  $(X, \Sigma)$  heißt messbarer Raum.

Nun wollen wir ein Maß auf einem System von messbaren Mengen definieren. Es ist klar, dass das Maß einer Vereinigung von disjunkten Elementen der Algebra die Summe ihrer Maße sein muss.

**Definition 3.2 (Maß)** Sei  $(X; \Sigma)$  ein messbarer Raum. Ein Maß ist eine Funktion  $\mu : \Sigma \rightarrow [0, \infty]$ , die folgende Eigenschaften hat:

- (a)  $\mu(\emptyset) = 0$ .
- (b)  $\mu$  ist  $\sigma$ -additiv, d.h. ist  $(A_k)_{k \in \mathbb{N}}$  eine Folge von paarweise disjunkten Mengen aus  $\Sigma$ , so gilt

$$\mu\left(\bigcup_{k \in \mathbb{N}} A_k\right) = \sum_{i=1}^{\infty} \mu(A_k).$$

Wir nennen  $(X, \Sigma, \mu)$  einen Maßraum.

Es gibt verschiedene Möglichkeiten Maße auf den reellen Zahlen zu definieren (z.B. Zählmaß, Diracmaß). Für uns ist aber hauptsächlich das Lebesguemaß von Interesse, da es das „kanonische“ Maß auf den reellen Zahlen ist. Es ordnet einem Quader seinen Inhalt zu. Um das Lebesguemaß definieren zu können, brauchen wir ein System von messbaren Mengen, die sogenannten Borelmengen. Um das Mengensystem der Borelmengen definieren zu können, benötigen wir das folgende Lemma.

**Lemma 3.3** *Sei  $X$  eine Menge und  $\mathcal{A}$  eine Familie von  $\sigma$ -Algebren über  $X$ . Der Durchschnitt*

$$\bigcap \mathcal{A} = \{A \subseteq X \mid A \in \Sigma \text{ für alle } \Sigma \in \mathcal{A}\}$$

*ist ebenfalls eine  $\sigma$ -Algebra.*

Beweis: siehe [Els 2002].

**Definition 3.4 (Borelmengen)** *Sei  $Q = \{(a_1, b_1] \times \dots \times (a_n, b_n] \subseteq \mathbb{R}^n \mid a_i < b_i\}$  die Menge aller halboffenen Quader und  $\mathcal{A}$  die Menge aller  $\sigma$ -Algebren, die  $Q$  enthalten. Dann heißt  $\mathcal{B}(\mathbb{R}^n) := \bigcap \mathcal{A}$  die Borel- $\sigma$ -Algebra und die Elemente von  $\mathcal{B}(\mathbb{R}^n)$  heißen Borelmengen.*

**Satz 3.5 (Existenz und Eindeutigkeit des Lebesgue-Maßes)**

(a) *Es gibt ein Maß  $\lambda$  auf der Borel- $\sigma$ -Algebra, das eindeutig bestimmt durch die Eigenschaft*

*$\lambda(Q) = (b_1 - a_1) \cdot \dots \cdot (b_n - a_n)$  für alle Quader  $Q = (a_1, b_1] \times \dots \times (a_n, b_n] \in \mathcal{Q}$  ist.*

Das Lebesguemaß ist das Maß, das jedem Quader den kanonischen Inhalt zuordnet. Es ist daher sinnvoll dieses Maß zu benutzen um Integrale zu definieren.

### 3.2 Das Lebesgueintegral

Es ist möglich das Riemannintegral auch für teilweise stetige Funktionen zu definieren, indem man den Definitionsbereich in Teile unterteilt auf denen die Funktion stetig ist. Ähnliches gilt auch für das Lebesgueintegral. Es kann gebildet werden, wenn die Funktion überall außer auf einer sogenannten Nullmenge lebesgueintegrierbar ist. Die lebesgueintegrierbaren Funktionen sind die Äquivalenzklassen der Funktionen, die sich nur auf einer Nullmenge unterscheiden.

**Definition 3.6 (Nullmengen)** *Eine Teilmenge  $A$  eines Maßraumes  $(X, \Sigma, \mu)$  heißt Nullmenge, falls  $\mu(A) = 0$ . Wegen Eigenschaft 3.2(b) ist die abzählbare Vereinigung von Nullmengen wieder eine Nullmenge.*

Die Grundidee des Riemannintegrals ist es, den Definitionsbereich der Funktion in Bereiche zu unterteilen und ein Ober- und Unterintegral zu bilden. Ist eine Funktion riemannintegrierbar, so unterscheiden sich die Werte der beiden Integrale nicht.

Es gibt allerdings Funktionen wie  $\chi_{\mathbb{Q}}$  bei denen diese Idee versagt. Das Oberintegral ist 1, das Unterintegral 0. Da die rationalen Zahlen aufgrund ihrer Abzählbarkeit eine Nullmenge bilden (siehe [For 1983]) sollte der Wert des Integrals allerdings Null sein. Für diese Herangehensweise müssen wir die Funktionswerte der Funktion mit dem Maß ihrer Urbilder multiplizieren. Dann wäre diese Funktion integrierbar und der Wert des Integrals 0.

- *Riemannintegral*: Unterteile den Definitionsbereich, multipliziere die Größe des Gebietes mit dem (approximierten) Funktionswert und addiere diese Werte.
- *Lebesgueintegral*: Unterteile den Wertebereich, und summiere über die (approximierten) Werte mal dem Maß des Urbilds.

### 3.2.1 Messbare Funktionen

Nun wollen wir die Funktionen beschreiben, deren Integral wir später bilden können. Hierzu benötigen wir die erweiterte Zahlengerade  $\bar{\mathbb{R}} := \mathbb{R} \cup \pm\infty$  und die  $\sigma$ -Algebra

$$\mathcal{B}(\bar{\mathbb{R}}) := \{A \cup B \mid A \subseteq \{\pm\infty\}, B \in \mathcal{B}(\mathbb{R})\}.$$

Es kann nur messbaren Mengen ein Maß zugeordnet werden. Um ein Lebesgueintegral definieren zu können, müssen also die Urbilder von messbaren Mengen wieder messbar sein. Diese Bedingung erfüllen folgende Funktionen.

**Definition 3.7** *Sei  $(X, \Sigma)$  ein messbarer Raum. Eine Funktion  $f : X \rightarrow \bar{\mathbb{R}}$  heißt messbar, falls*

$$(\forall A \in \mathcal{B}(\bar{\mathbb{R}})) f^{-1}(A) \in \Sigma$$

*gilt.*

### 3.2.2 Das Lebesgueintegral

Um das Integral definieren zu können beschäftigen wir uns zunächst mit einer besonderen Klasse der messbaren Funktionen, den sogenannten Stufenfunktionen, deren Integral eine endliche Summe ist.

**Definition 3.8 (Stufenfunktion)** *Sei  $(X, \Sigma, \mu)$  ein Maßraum. Eine Funktion  $s : (X, \Sigma) \rightarrow \mathbb{R}$  heißt Stufenfunktion, wenn  $s$  messbar ist und nur endlich viele Werte annimmt.*

Seien  $a_1, \dots, a_k$  die Funktionswerte und  $A_1 := s^{-1}(a_1), \dots, A_k = s^{-1}(a_k)$  die Mengen auf denen  $f$  die Funktionswerte annimmt. Dann lässt sich jede Stufenfunktion auf einfache Weise durch

$$s(x) = \sum_{j=1}^k a_j \chi_{A_j}(x)$$

darstellen. Ist  $s$  eine nichtnegative Stufenfunktion, so definieren wir das Integral für eine messbare Menge  $E \in \Sigma$  durch

$$\int_E s \, d\mu := \sum_{j=1}^k a_j \mu(A_j \cap E) \in [0, \infty]. \quad (3)$$

Der folgende Satz stellt sicher, dass sich Stufenfunktionen zur Approximation von Funktionen eignen.

**Satz 3.9** *Sei  $(X, \Sigma, \mu)$  ein Maßraum und  $f : X \rightarrow [0, \infty]$ . Dann gibt es eine monoton wachsende Folge von Stufenfunktionen  $s_k : X \rightarrow [0, \infty)$  mit*

$$(\forall x \in X) \lim_{k \rightarrow \infty} s_k(x) = f(x).$$

Beweis: [Els 2002] Kapitel 4 Satz 2.1.

Somit können wir das Lebesgueintegral für nichtnegative Funktionen als Grenzwert der Integrale der approximierenden Stufenfunktionen definieren.

**Definition 3.10** *Sei  $f : X \rightarrow [0, \infty]$  messbar und  $E \in \Sigma$ . Dann ist das Lebesgueintegral durch*

$$\int_E f \, d\mu := \sup \left\{ \int_E s \, d\mu \mid s : X \rightarrow [0, \infty) \text{ Stufenfkt. mit } s(x) \leq f(x) \right\}$$

definiert.

Um das Lebesgueintegral einer beliebigen Funktion definieren zu können, muss man diese in ihren negativen und positiven Teil unterteilen, diese integrieren und die Differenz der beiden Werte bilden. Hierzu wird eine messbare Funktion  $f$  in zwei nichtnegative Funktionen

$$f_{\pm} : X \rightarrow [0, \infty], \quad f_+ := \max(0, f) \quad \text{und} \quad f_- := \max(0, -f)$$

unterteilt. Das Integral ist definiert, wenn für eine der beiden Funktionen der Wert des Integrals endlich ist (siehe [Els 2002]).

**Definition 3.11 (Lebesgueintegral)** *Sei  $(X, \Sigma, \mu)$  ein Maßraum,  $E \in \Sigma$  und  $f : X \rightarrow \bar{\mathbb{R}}$  messbar. Ist eines der beiden Integrale  $\int_E f_+ \, d\mu$  und  $\int_E f_- \, d\mu$  endlich, so ist das Lebesgueintegral von  $f$  definiert durch*

$$\int_E f \, d\mu := \int_E f_+ \, d\mu - \int_E f_- \, d\mu.$$

Man zeigt relativ leicht, dass stetige Funktionen auch messbar sind. Existiert also das Riemannintegral so existiert in der Regel auch das Lebesgueintegral, umgekehrt gilt das aber nicht (z.B. die Funktion  $\chi_{\mathbb{Q}}$ ). Die einzigen Ausnahmen sind bestimmte uneigentliche Integrale wie  $\frac{\cos(x)}{x}$ . Bei diesen Funktionen ist der Wert des Lebesgueintegrals von  $f_+$  und von  $f_-$  unendlich, das uneigentliche Riemannintegral existiert aber.

### 3.3 Die wichtigsten Sätze der Integrationstheorie

In diesem Kapitel werden die benötigten Sätze aus der Integrationstheorie formuliert. Wir werden die Beweise nicht ausführen. Außerdem wollen wir falls Riemann- und Lebesgueintegral existieren, ihre Gleichheit (bzgl. dem Maßraum  $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n), \lambda)$ ) zeigen. Dies macht klar, warum wir uns Lebesgueintegrale ebenso wie Riemannintegrale als Fläche unter einem Graphen vorstellen können. Ein weiterer wichtiger Punkt ist, dass wir später Funktionen betrachten werden, die riemann- und lebesgueintegrierbar sind. Die Gleichheit macht es uns möglich zur Berechnung der Integrale sowohl Sätze zu verwenden, die im allgemeinen nur für riemannintegrierbare Funktionen (z.B. den Mittelwertsatz) gelten, als auch die Konvergenzsätze zu benutzen, die nur für lebesgueintegrierbare Funktionen gelten.

#### 3.3.1 Der Satz von Beppo Levi und der Satz von Lebesgue

Als erstes werden wir den Satz über die monotone Konvergenz oder auch Satz von Beppo Levi(1905) aufführen. Sei  $(X, \Sigma, \mu)$  ein Maßraum.

**Theorem 3.12 (Satz von Beppo Levi)**

Seien  $f_1, f_2, \dots : X \rightarrow \mathbb{R}$  messbare Funktionen mit  $0 \leq f_1 \leq f_2 \leq \dots$ . Es sei  $f(t) := \lim_{n \rightarrow \infty} f_n(t) \in [0, \infty)$ . Dann ist  $f$  messbar mit

$$\lim_{n \rightarrow \infty} \int f_n d\mu = \int f d\mu \in [0, \infty).$$

Beweis: siehe [Els 2002] Kapitel 4 Satz 2.7.

Ein weiteres zentraler Ergebnis aus der Integrationstheorie ist der Satz von der majorisierten Konvergenz oder auch Satz von Lebesgue.

**Theorem 3.13 (Konvergenzsatz von Lebesgue(1910))**

Seien  $f_1, f_2, \dots : X \rightarrow \mathbb{R}$  integrierbar und es existiere eine messbare Funktion  $f$  mit  $f(t) = \lim_{n \rightarrow \infty} f_n(t)$  fast überall. Es existiere ferner eine integrierbare Funktion  $g$  mit  $|f_n| \leq g$  fast überall für alle  $n \in \mathbb{N}$ . Dann ist  $f$  integrierbar, und es gilt

$$f(t) = \lim_{n \rightarrow \infty} \int f_n d\mu = \int f d\mu.$$

Beweis: siehe [Els 2002] Kapitel 4 Satz 5.2.

#### 3.3.2 Der Zusammenhang zwischen Lebesgue- und Riemannintegral

In diesem Unterkapitel wollen wir skizzieren, wie man im Eindimensionalen die Gleichheit von Lebesgue und Riemannintegral zeigen kann. Da der Satz von Fubini (Theorem 3.15) sowohl für lebesgue- als auch riemannintegrierbare Funktionen bewiesen werden kann, kann man die Gleichheit auch im  $n$ -dimensionalen leicht aus dem folgenden Satz schließen.

**Satz 3.14** Sei  $[a, b] \subseteq \mathbb{R}$  ein beschränktes Intervall. Dann ist jede auf  $[a, b]$  riemannintegrierbare Funktion  $f$  auch bzgl. des Lebesguemaßes  $\lambda$  lebesgueintegrierbar und es gilt

$$\int_{[a,b]} f \, d\lambda = \int_a^b f(x) \, dx.$$

Beweisskizze: Per Definition sind das Riemann- und das Lebesgueintegral für Stufenfunktionen identisch. Man approximiert die Funktion mit einer Folge von Treppenfunktionen die von oben approximieren und einer Folge von Treppenfunktionen, die von unten approximiert. Mit Hilfe des Satzes über die majorierten Konvergenz wird gezeigt, dass die Grenzfunktionen lebesgueintegrierbar sind und dass beide fast überall mit  $f$  übereinstimmen. Für einen genauen Beweis siehe (siehe [Els 2002] Kapitel 4 Satz 6.1).

### 3.3.3 Der Satz von Fubini und die Transformationsformel

Ein weiterer zentraler Satz der Integrationstheorie ist der Satz von Fubini. Dieser besagt im wesentlichen, dass die Integrationsreihenfolge den Wert des Integrals nicht verändert. Der Satz von Fubini gilt sowohl für lebesgue- als auch riemannintegrierbare Funktionen. Wir werden ihn aber nur für lebesgueintegrierbare Funktionen formulieren.

**Theorem 3.15 (Satz von Fubini)**

Seien  $n, m$  natürliche Zahlen und  $f : \mathbb{R}^{n+m} \rightarrow \mathbb{K}$  eine messbare Funktion. Dann gilt,

(a) Für alle  $s \in \mathbb{R}^n$  ist die Funktion  $t \mapsto f(s, t)$  messbar, und für alle  $t \in \mathbb{R}^m$  ist  $s \mapsto f(s, t)$  messbar. Ferner sind die  $[0, \infty]$ -wertigen  $s \mapsto \int |f(s, t)| d\lambda(s)$  messbar.

(b) Ist

$$\int_{\mathbb{R}^m} \left( \int_{\mathbb{R}^n} |f(s, t)| d\lambda(s) \right) d\lambda(t) < \infty,$$

so ist  $f$  integrierbar.

(c) Ist  $f$  integrierbar, so ist  $f_t : s \mapsto f(s, t)$  für fast alle  $t$  integrierbar. Die Funktion

$$h : t \mapsto \begin{cases} \int f_t d\lambda & \text{falls } f_t \text{ integrierbar} \\ 0 & \text{sonst} \end{cases}$$

ist mess- und integrierbar und es gilt

$$\int_{\mathbb{R}^{n+m}} f \, d\lambda = \int_{\mathbb{R}^n} h \, d\lambda$$

Beweis: siehe [Els 2002].

**Bemerkung 3.16** Der Satz von Fubini besagt also, dass die Integrationsreihenfolge den Wert des Integrals unbeeinflusst lässt.

**Folgerung 3.17** Ist eine Funktion sowohl riemann- als auch lebesgueintegrierbar bzgl. dem Lebesguemaß, so folgt aus Satz 3.14 und Theorem 3.15, dass der Wert der beiden Integrale übereinstimmt.

Der andere wichtige Satz, durch den viele Integrale erst berechenbar werden, ist die Transformationsformel. Sie ist die Substitutionsformel für den  $\mathbb{R}^n$ . Um diese formulieren zu können, brauchen wir den Begriff des  $C^1$ -Diffeomorphismus. Diffeomorphismen sind differenzier- und invertierbare Abbildungen, deren Umkehrabbildungen ebenfalls differenzierbar sind. Zunächst definieren wir die Räume der  $k$ -mal stetig differenzierbaren Funktionen.

**Definition 3.18** Sei  $U \subseteq \mathbb{R}^n$  offen. Wir setzen

$$C^k(U) := \{f : U \rightarrow \mathbb{R} \mid f \text{ ist } k\text{-mal stetig differenzierbar}\}.$$

**Definition 3.19 ( $C^k$ -Diffeomorphismus)** Seien  $U, V \subseteq \mathbb{R}^n$  offen. Ein  $C^k$ -Diffeomorphismus ist eine bijektive Abbildung  $\phi : U \rightarrow V \in C^k$  deren Umkehrabbildung ebenfalls in  $C^k$  ist.

**Theorem 3.20 (Transformationsformel)** Seien  $U, V \subseteq \mathbb{R}^n$  offen und  $\phi : U \rightarrow V$  ein  $C^1$ -Diffeomorphismus. Dann ist  $f$  genau dann lebesgueintegrierbar, wenn  $(f \circ \phi)|\det d\phi|$  lebesgueintegrierbar ist und es gilt

$$\int_U f \circ \phi |\det d\phi| d\lambda = \int_V f d\lambda.$$

Beweis: siehe [Els 2002] Kapitel 5 Satz 4.2.

### 3.3.4 Die Transformationsformel für lineare Abbildungen

Bijektive lineare Abbildungen zwischen endlichdimensionalen Räumen sind  $C^\infty$ -Diffeomorphismen. Ihre erste Ableitung ist die zu der linearen Abbildung gehörende Matrix. Man erhält für den Fall, dass die Abbildung  $\phi$  linear ist folgenden Spezialfall der Transformationsformel.

**Satz 3.21 (Transformationsformel im linearen Fall)** Seien  $U, V \subseteq \mathbb{R}^n$  offene Mengen und  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  eine lebesgueintegrierbare Funktion und  $A : U \rightarrow V$  eine lineare Abbildung. Dann gilt nach Theorem 3.20

$$\int_V f(y) dy = \int_U f(Ax) |\det(A)| dx.$$

### 3.3.5 Der Mittelwertsatz

Die Funktionen mit denen wir uns in den kommenden Kapiteln beschäftigen werden sind sowohl riemann- als auch lebesgueintegrierbar. Darüberhinaus sind sie differenzierbar. Um ihren Wert zu berechnen, können wir daher auch den Mittelwertsatz benutzen. Damit wir den Mittelwertsatz definieren können, müssen wir uns zunächst überlegen was Differenzierbarkeit im Mehrdimensionalen bedeutet.

**Definition 3.22 (Differenzierbarkeit)** Sei  $U$  eine offene Teilmenge des  $\mathbb{R}^n$ . Eine Funktion  $f : U \rightarrow \mathbb{R}^m$  heißt differenzierbar an der Stelle  $a \in U$ , falls eine lineare Abbildung  $T : \mathbb{R}^n \rightarrow \mathbb{R}^m$  und eine an der Stelle  $a$  stetige Abbildung  $r : U \rightarrow \mathbb{R}^m$  mit

$$f(x) = f(a) + T(x - a) + r(x)\|x - a\|$$

und  $r(a) = 0$  existiert. Die Abbildung  $T$  heißt Ableitung von  $f$  in  $a$  und wird mit  $\text{grad}(f(a))$  bezeichnet. Eine Abbildung heißt differenzierbar, wenn sie in jedem Punkt differenzierbar ist.

**Satz 3.23 (Mittelwertsatz)** Sei  $f : U \rightarrow \mathbb{R}$  differenzierbar und die Verbindungsstrecke zwischen den Punkten  $a, b \in U$  sei in  $U$  enthalten. Dann gibt es einen Punkt  $c$  auf der Verbindungsstrecke zwischen  $a$  und  $b$  mit

$$f(b) - f(a) = \langle \text{grad } f(c), b - a \rangle.$$

Beweis: siehe [Bar 1996] Seite 115.

## 4 Grundbegriffe der Wahrscheinlichkeitstheorie

Ziel dieses Kapitels ist es, die für den Beweis für die Sicherheit des Kryptosystems benötigten Begriffe aus der Statistik und Wahrscheinlichkeitstheorie zusammenzufassen.

### 4.1 Wahrscheinlichkeitsräume

Ein Wahrscheinlichkeitsraum soll ein Zufallsexperiment beschreiben. Was wird benötigt um dies zu tun?

Als erstes braucht man die Menge aller möglichen Ergebnisse. Jedem dieser Ergebnisse ist eindeutig ein Element der *Ergebnismenge*  $\Omega$  zugeordnet. Die Teilmengen von  $\Omega$  heißen Ereignisse. Die Menge aller Ereignisse  $\mathcal{A}$  bilden eine  $\sigma$ -Algebra (siehe Definition 3.1). Diesen Ereignissen werden mit Hilfe eines sogenannten Wahrscheinlichkeitsmaßes  $P$  Wahrscheinlichkeiten zugeordnet. Bei der Definition von  $P$  ist dabei darauf zu achten, dass  $P$  gewisse Voraussetzungen erfüllt. Zum Beispiel muss sichergestellt werden, dass sich Wahrscheinlichkeiten disjunkter Ereignisse addieren. Es hat sich dabei als sehr brauchbar erwiesen, dass eine Funktion, die Wahrscheinlichkeiten misst die sogenannten *Axiome von Kolmogoroff* erfüllt.

**Definition 4.1 (Wahrscheinlichkeitsmaß)** Sei  $\Omega$  eine Ergebnismenge und  $\mathcal{A}$  eine  $\sigma$ -Algebra von Ereignissen. Eine Abbildung  $P : \mathcal{A} \rightarrow \mathbb{R}$  heißt *Wahrscheinlichkeitsmaß*, wenn:

- (a)  $P(A) \geq 0$  für alle  $A \in \mathcal{A}$
- (b)  $P(\Omega) = 1$ .
- (c)  $P(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P(A_i)$  mit  $A_i \cap A_j = \emptyset$  für  $i \neq j$ .

Das Tripel  $(\Omega, \mathcal{A}, P)$  heißt *Wahrscheinlichkeitsraum*.

Die Wahrscheinlichkeit eines Ereignisses  $A_i$  ist dann der durch die Maßfunktion zugeordnete Wert  $P(A_i)$ .

### 4.2 Zufallsvariablen und Verteilungsfunktionen

Das Ergebnis eines Zufallsexperiment muss nicht unbedingt ein Zahlenwert sein. Trotzdem interessiert häufig ein mit dem Zufallsexperiment zusammenhängender Zahlenwert. Eine Zufallsvariable beschreibt diese Situation mathematisch. Sie ordnet einem Ereignis einen Zahlenwert zu.

**Definition 4.2 (Zufallsvariable)** Sei  $(\Omega, \mathcal{A}, P)$  ein Wahrscheinlichkeitsraum. Eine Abbildung  $X : \Omega \rightarrow \mathbb{R}$  heißt *Zufallsvariable über  $(\Omega, \mathcal{A}, P)$*  falls für alle reellen Intervalle  $I \subseteq \mathbb{R}$

$$\{\omega \in \Omega : X(\omega) \in I\} \in \mathcal{A}$$

gilt.

Für die Wahrscheinlichkeit des Ereignisses  $\{\omega \in \Omega : X(\omega) \in I\}$  schreiben wir abkürzend  $P(X \in I)$ . Die Wahrscheinlichkeit solcher Ereignisse lässt sich mit Hilfe der folgenden Verteilungsfunktion berechnen.

**Definition 4.3 (Verteilungsfunktion)** Sei  $X$  eine Zufallsvariable über  $(\Omega, \mathcal{A}, P)$ . Dann heißt die Abbildung  $F : \mathbb{R} \rightarrow [0, 1]$  mit

$$F(x) = P(X \leq x), x \in \mathbb{R}$$

Verteilungsfunktion der Zufallsvariablen  $X$ .

Es gibt zwei verschiedene Klassen von Zufallsvariablen.

**Definition 4.4 (Diskrete Zufallsvariable)** Eine Zufallsvariable  $X$  heißt diskret, wenn ihr Wertebereich endlich oder abzählbar unendlich ist.

Beispiele für diskrete Zufallsvariablen sind die Geometrische Verteilung und die Binomialverteilung. Das Charakteristische der Verteilungsfunktion einer diskreten Zufallsvariable ist, dass sie eine Treppenfunktion ist. Demgegenüber stehen die Zufallsvariablen deren Verteilungsfunktionen stetig sind, wie z.B. die Normalverteilung oder die Weibull-Verteilung.

**Definition 4.5** Eine Zufallsvariable heißt stetig mit Dichte  $f$  verteilt, falls sich ihre Verteilungsfunktion  $F : \mathbb{R} \rightarrow \mathbb{R}$  folgendermaßen definieren lässt:

$$F(x) = \int_{-\infty}^x f(t) dt \quad x \in \mathbb{R}.$$

### 4.3 Erwartungswert und Varianz

Es gibt verschiedene wichtige Kennzahlen mit denen man Zufallsvariablen zu beschreiben versucht. Die zwei wichtigsten sind der Erwartungswert und die Varianz. Der Erwartungswert ist so etwas wie das arithmetische Mittel, d.h. das durchschnittliche Ergebnis. Die Varianz beschreibt die Streuung um diesen Wert.

**Definition 4.6 (Erwartungswert)**

(a) Ist  $X$  eine diskrete Zufallsvariable, die die Werte  $x_1, x_2, \dots$  annimmt. Falls  $\sum_i |x_i| \cdot P(X = x_i)$  konvergiert heißt

$$E(X) = \sum_i x_i \cdot P(X = x_i)$$

der Erwartungswert von  $X$ .

(b) Ist  $X$  eine stetig verteilte Zufallsvariable mit der Dichte  $f$  und ist das Integral  $\int_{-\infty}^{\infty} |x|f(x) dx$  endlich, so heißt

$$\int_{-\infty}^{\infty} x f(x) dx$$

der Erwartungswert von  $X$ .

Bei diskreten Zufallsvariablen wird die absolute Konvergenz der Reihe gefordert, um sicherzustellen, dass die Reihe invariant gegenüber Umordnung ist. Eine ähnliche Begründung greift auch bei stetig verteilten Zufallsvariablen. Mit  $X$  ist natürlich auch  $[X - E(X)]^2$  eine Zufallsvariable und es lässt sich die Varianz folgendermaßen definieren.

**Definition 4.7 (Varianz)** Sei  $X$  eine Zufallsvariable, für die sowohl  $E(X)$  als auch  $E([X - E(X)]^2)$  existieren. Dann heißt

$$\text{Var}(X) = E([X - E(X)]^2)$$

die Varianz von  $X$ .

Wie bereits angedeutet ist die Varianz ein Indikator dafür, wie weit eine Zufallsvariable um ihren Erwartungswert „streut“. Dies besagt der folgende Satz.

**Satz 4.8 (Tschebyscheffsche Ungleichung)** Sei  $X$  eine Zufallsvariable deren Varianz existiert. Dann gilt für jedes  $c > 0$  die Ungleichung

$$P(|X - E(X)| \geq c) \leq \frac{\text{Var}(X)}{c^2}.$$

Beweis: siehe [Leh 2000].

#### 4.4 Mehrdimensionale Zufallsvariablen, Unabhängigkeit

Häufig sind bei einem Zufallsexperiment mehrere durch das Ergebnis  $\omega$  bestimmte Zahlenwerte von Interesse. Einem Ergebnis werden also mehrere reelle Zahlen zugeordnet. Wir wollen uns hier nur mit den sogenannten zweidimensionalen Zufallsvariablen beschäftigen. Die Sätze lassen sich auch auf  $n$ -dimensionale Zufallsvariablen verallgemeinern.

**Definition 4.9 (Zweidim. Zufallsvariable)** Sei  $(\Omega, \mathcal{A}, P)$  ein Wahrscheinlichkeitsraum und  $X : \Omega \rightarrow \mathbb{R}$  und  $Y : \Omega \rightarrow \mathbb{R}$  Zufallsvariablen. So wird durch

$$(X, Y) : \Omega \rightarrow \mathbb{R}^2$$

$$\omega \mapsto (X(\omega), Y(\omega))$$

eine zweidimensionale Zufallsvariable definiert. Die Funktion

$$F(x, y) = P(X \leq x, Y \leq y)$$

heißt Verteilungsfunktion von  $(X, Y)$  wobei  $P(X \leq x, Y \leq y)$  die Wahrscheinlichkeit des Ereignisses

$$\{\omega \in \Omega : X(\omega) \leq x\} \cap \{\omega \in \Omega : Y(\omega) \leq y\}$$

bezeichnet.

Mit  $F_X = P(X \leq x)$  und  $F_Y = P(Y \leq y)$  bezeichnen wir die Verteilungsfunktion der Komponenten der zweidimensionalen Zufallsvariable. Diese werden auch mit *Randverteilungsfunktionen* genannt. Wegen

$$F_X(x) = P(X \leq x, Y < \infty) = \lim_{y \rightarrow \infty} F(x, y) \quad , x \in \mathbb{R}$$

und

$$F_Y(y) = P(X < \infty, Y < y) = \lim_{x \rightarrow \infty} F(x, y) \quad , y \in \mathbb{R}$$

sind sie durch  $F$  eindeutig bestimmt. Umgekehrt lässt sich nur unter bestimmten Voraussetzungen die Verteilungsfunktion, der sogenannten Unabhängigkeit von  $X$  und  $Y$ , aus den beiden Randverteilungen die Verteilungsfunktion bestimmen.

**Definition 4.10 (Unabhängigkeit)** Die Zufallsvariablen  $X$  und  $Y$  heißen *unabhängig*, wenn für die Verteilungsfunktion der zweidimensionalen Zufallsvariable  $(X, Y)$

$$F(x, y) = F_X(x) \cdot F_Y(y)$$

für alle  $(x, y) \in \mathbb{R}^2$  gilt.

Von Interesse ist auch die Summe zweier Zufallsvariablen  $X$  und  $Y$ . Für den Erwartungswert von  $X + Y$  gilt folgender Satz.

**Satz 4.11** Sei  $(X, Y)$  eine zweidimensionale Zufallsvariable mit zugehörigen Erwartungswerten  $E(X)$  und  $E(Y)$ . Dann hat  $X + Y$  den Erwartungswert

$$E(X + Y) = E(X) + E(Y)$$

Beweis: siehe [Leh 2000].

Der Zusammenhang zwischen der Varianz der Zufallsvariablen  $X$  und  $Y$  und der Varianz der zweidimensionalen Zufallsvariable  $(X, Y)$  ist etwas komplizierter als der zwischen den Erwartungswerten zweier Zufallsvariablen. Es gilt folgender Satz.

**Satz 4.12** Sei  $(X, Y)$  eine zweidimensionale Zufallsvariable. Existieren die Varianzen  $V(X)$  und  $V(Y)$ , so existiert auch die Varianz von  $X + Y$  und der Erwartungswert von  $X \cdot Y$ , und es gilt

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) + 2 \cdot [E(X \cdot Y) - E(X)E(Y)].$$

Beweis: siehe [Leh 2000].

Eine wichtige Kenngröße einer zweidimensionalen Zufallsvariable ist der Term  $[E(X \cdot Y) - E(X)E(Y)]$ .

**Definition 4.13 (Kovarianz)** Sei  $(X, Y)$  eine zweidimensionale Zufallsvariable. Die Varianzen von  $X$  und  $Y$  mögen existieren. Dann heißt die Größe

$$\text{Cov}(X, Y) = E([X - E(X)] \cdot [Y - E(Y)]) = E(X \cdot Y) - E(X)E(Y)$$

*Kovarianz*. Falls die Kovarianz null ist, heißen die Zufallsvariablen *unkorreliert*.

**Bemerkung 4.14** Sind die beiden Zufallsvariablen unkorreliert so gilt nach Satz 4.12

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) + 2 \cdot \text{Cov}(X, Y) = \text{Var}(X) + \text{Var}(Y).$$

In der Kryptographie generiert man häufig Werte mittels unabhängiger Zufallsvariablen  $X_1, \dots, X_n$  und addiert diese. Dieses Verfahren liefert eine neue Zufallsvariable  $Y = \sum_{i=1}^n X_i$ . Mittels Satz 4.11 können wir leicht den Erwartungswert ausrechnen. Es ist aber nicht so einfach die Kovarianz einer  $n$ -dimensionalen Zufallsvariablen auszurechnen. Der folgende Satz liefert uns eine leicht ins  $n$ -dimensionale zu verallgemeinernde Möglichkeit die Varianz zu berechnen, ohne die Kovarianz bestimmen zu müssen.

**Satz 4.15** Sei  $(X, Y)$  eine zweidimensionale Zufallsvariable. Sind die beiden Zufallsvariablen  $X$  und  $Y$  unabhängig, so sind sie unkorreliert.

Beweis: siehe [Leh 2000].

Zum Abschluss unserer Einführung in die Wahrscheinlichkeitstheorie und Statistik wollen wir uns definieren, was der statistische Abstand zweier Zufallsvariablen im diskreten und im kontinuierlichen Fall ist.

**Definition 4.16** Seien  $X, Y$  zwei diskret verteilte Zufallsvariablen die Werte  $(x_i)_{i \in \mathbb{N}}$  annimmt. Dann ist der statistische Abstand durch

$$\Delta(X, Y) = \frac{1}{2} \sum_{i \in \mathbb{N}} |P(X = x_i) - P(Y = x_i)|$$

definiert.

**Definition 4.17** Seien  $X, Y$  zwei eindimensionale stetig mit den Dichtefunktionen  $f_X$  und  $f_Y$  verteilte Zufallsvariablen. Der statistische Abstand ist definiert durch

$$\Delta(X, Y) = \frac{1}{2} \int_{\mathbb{R}} |f_X(x) - f_Y(x)| dx.$$

## 5 Grundlagen der Komplexitätstheorie

Im ersten Teil dieses Kapitels wollen wir zwei verschiedene Ansätze zur Klassifizierung berechenbarer Probleme nach ihrer Effizienz präsentieren. Auf dieser Grundlage werden wir uns mit der Frage beschäftigen, wann Verteilungen praktisch ununterscheidbar sind. Ziel ist es, einen groben Überblick über die für uns interessanten Begriffe zu geben.

Um Probleme aufgrund ihrer Komplexität einordnen zu können, muss man sich zunächst mit der Frage beschäftigen, wann Probleme überhaupt berechenbar sind. Hierzu wurde der Begriff der *intuitiven Berechenbarkeit* eingeführt. Probleme heißen *intuitiv berechenbar*, wenn es einen Algorithmus gibt, der das Problem löst und nach endlich vielen Schritten hält. Um diese Formulierung zu konkretisieren, entwickelte Turing ein simples Modell einer Maschine, das er *Universal Machine* nannte und heute nach ihm *Turing-Maschine* benannt ist [Tur 1936]. Die allgemein akzeptierte Vermutung ist, dass die Klasse der intuitiv lösbaren Probleme mit der Klasse der Turing berechenbaren Probleme übereinstimmt. Ziel der Komplexitätstheorie ist es, diese Probleme bezüglich ihrer Schwierigkeit zu klassifizieren. Die These auf der die Klassifikation basiert ist, dass Algorithmen mit polynomieller Laufzeit effizient zu berechnen sind. In der Praxis ist diese These jedoch kaum haltbar, da Algorithmen mit großer polynomieller Laufzeit bereits sehr rechenintensiv sind.

### 5.1 Berechenbarkeitsmodelle und Komplexität

Um Algorithmen und damit auch Probleme bezüglich ihrer Effizienz zu klassifizieren, müssen wir ihre Laufzeit in Abhängigkeit zu der Eingabelänge bestimmen. Die Schwierigkeit eines Problems hängt von dem effizientesten Algorithmus der das Problem löst ab.

**Definition 5.1** *Ein Algorithmus hat polynomielle Laufzeit, wenn die Anzahl der Schritte, polynomiell in der Eingabelänge  $l$  beschränkt ist. D.h.,*

$$\text{Schrittzahl}(l) \leq p(l)$$

wobei  $l$  die Eingabelänge und  $p \in \mathbb{R}[X]$ .

Analog kann man exponentielle, subexponentielle usw. Algorithmen definieren. Allgemein hat ein Algorithmus Laufzeit  $O(f(l))$ , wenn eine Konstante  $K$  existiert mit

$$\text{Schrittzahl}(l) \leq K \cdot f(l).$$

Die Komplexität eines Problems wird über die Sprache in der es formuliert ist beschrieben. Wir geben hier eine vereinfachte Definition, was eine Sprache ist.

**Definition 5.2 (Sprache)** *Sei  $\mathcal{A}$  ein Menge von Zeichen (Alphabet). Ein Wort ist eine endliche Folge von Elementen aus  $\mathcal{A}$ . Sei  $\mathcal{A}^*$  die Menge aller Wörter. Eine Sprache  $A := (\mathcal{A}, W_A, F_A)$  über  $\mathcal{A}$  ist die Menge aller gültigen Wörter  $W_A \subseteq \mathcal{A}^*$  und eine Menge von Funktionen  $F_A$ .*

**Definition 5.3 (char. Funktion)** Sei  $A$  eine Sprache mit Alphabet  $\mathcal{A}$ . Die charakteristische Funktion ist definiert durch

$$\chi_A(a) = 1 \Leftrightarrow a \in W_A.$$

Man kann das Lösen eines Problems auf die Berechnung einer charakteristischen Funktion einer Sprache reduzieren (siehe [Hop 1979]).

Im folgenden werden wir nur noch Sprachen betrachten, deren Alphabet aus der Menge  $\{0, 1\}$  besteht und deren Funktionenmenge nur aus der charakteristischen Funktion besteht. Nun wollen wir die klassische Einteilung der turingberechenbare Probleme in Effizienzklassen betrachten. Die effizient lösbaren Probleme sind die Probleme, die mit Hilfe eines deterministischen Polynomialzeitalgorithmus gelöst werden können.

**Definition 5.4 (P)**  $\mathcal{P}$  ist die Klasse der Sprachen, deren charakteristische Funktion durch einen deterministischen Polynomialzeit-Turing-Maschinen-Algorithmus berechnet werden kann.

Die Klasse  $\mathcal{NP}$  sind die Sprachen, deren Lösung man durch einen Polynomialzeitalgorithmus überprüft werden kann. Dies sind die Sprachen, die von nicht-deterministischen Polynomialzeitmaschinen erkennbar sind.

**Definition 5.5 (NP)** Die Klasse der nichtdeterministischen Polynomialzeitsprachen  $A$  ist definiert durch

$$A \in \mathcal{NP} :\Leftrightarrow$$

$$\exists B \in \mathcal{P}, W_B \subseteq \{0,1\}^* \times \{0,1\}^*, P \in \mathbb{R}[X] W_A = \{x \in \{0,1\}^* \mid \exists y \in \{0,1\}^{P(|x|)} : (x,y) \in W_B\}$$

Ist  $(x,y) \in B$ , so heißt  $y$  Zeuge für  $x \in A$ .

Es wird im allgemeinen davon ausgegangen, dass die Klassen  $\mathcal{NP}$  und  $\mathcal{P}$  nicht gleich sind. Ein ungelöstes Problem der Komplexitätstheorie ist es, diese Vermutung zu beweisen.

Eine wichtige Konstruktion in der Kryptographie und der Komplexitätstheorie sind die Orakel.

**Definition 5.6 (Orakel)** Ein Orakel ist ein nicht spezifizierter Polynomialzeitalgorithmus der ein Problem  $A$  löst.

Orakel werden in der Komplexitätstheorie verwendet, um Probleme auf andere zu reduzieren. Das funktioniert folgendermaßen. Man nimmt an, dass man ein Problem  $A$  mit Hilfe eines Orakels löst. Diese Lösungen werden benutzt um einen Polynomialzeitalgorithmus zu konstruieren, der ein anderes Problem löst (siehe Cookreduktion Definition 5.10). Ein wichtiges Anwendungsgebiet in der Kryptographie liegt in der Simulation von Angriffen auf ein Kryptosystem.

## 5.2 Probabilistische Polynomialzeit

Beschränkt man sich auf die eben beschriebene klassische Einteilung in Komplexitätsklassen, so sind die Elemente aus  $\mathcal{P}$  die „effizient“ berechenbaren Probleme. Existiert aber ein Polynomialzeit-Algorithmus, der Verschlüsselungen der 0 von denen der 1 mit einer Wahrscheinlichkeit von beispielsweise  $\frac{3}{4}$  unterscheiden kann, so ist das Problem, Verschlüsselungen der Null von denen der Eins zu unterscheiden, nicht in  $\mathcal{P}$ . Im klassischen Sinn gilt das System als nicht gebrochen. Es ist also notwendig einen anderen Sicherheitsbegriff zu definieren. Die Grundlage hierfür bilden probabilistische Polynomialzeit-Algorithmen, die wir als effizient durchführbar betrachten. Genauer gesagt betrachten wir randomisierte Algorithmen (d.h. probabilistische Turing Maschinen) mit polynomieller Laufzeit als effizient.

Eine Möglichkeit zur Konstruktion solcher Algorithmen ist es, zufällige Entscheidungen sogenannte Münzwürfe zu erlauben. Sei  $x$  die Eingabe einer probabilistischen Turing-Maschine. Die Ausgabe wird durch eine Zufallsvariable  $M$  beschrieben. Mit  $P[M(x) = y]$  bezeichnen wir die Wahrscheinlichkeit, dass  $M$  den Wert  $y$  annimmt. Neben der Eingabe hängt diese Wahrscheinlichkeit von den Münzwürfen ab. Wir können ohne Beschränkung der Allgemeinheit annehmen, dass die Anzahl der Würfe  $t_M(x)$  konstant ist. Die Wahrscheinlichkeit, dass die Ausgabe einer probabilistischen Turing-Maschine den Wert  $y$  bei Eingabe des Wertes  $x$  annimmt, ist also

$$P[M(x) = y] = \frac{|\{r \in \{0, 1\}^{t_M(x)} : M_r(x) = y\}|}{2^{t_M(x)}}.$$

Mit  $M_r(x)$  wird das Ergebnis der Turing-Maschine mit Eingabe  $x$  und der Folge der Münzwürfe  $r$  bezeichnet.

Da die Ausgabe eines probabilistischen Algorithmus von den internen Münzwürfen abhängt kann es durchaus sein, dass Ausgaben nicht richtig sind. Dies geht in der folgenden Definition ein.

**Definition 5.7 (Bounded-Probability Polynomial-time-BPP):** *BPP sind die Sprachen, die durch eine probabilistische-Polynomialzeit-Turing-Maschine  $M$  erkannt werden kann. Wir sagen, dass  $M$  eine Sprache  $A = (\mathcal{A}, W_A, F_A)$  erkennt, wenn*

$$(\forall x \in \mathcal{A}^*) P[M(x) = \chi(x)] \geq \frac{2}{3}.$$

Die Konstante ist willkürlich gewählt und kann durch jede beliebige Konstante andere Konstante größer  $\frac{1}{2}$  ersetzt werden. Allgemein gilt  $A \in \mathcal{BPP}$ , wenn eine in Polynomialzeit berechenbare Funktion  $t : \mathbb{N} \rightarrow [0, 1]$ , ein positives Polynom  $p$  und eine probabilistische Polynomialzeit-Turing-Maschine  $M$  so existieren, dass

$$\begin{aligned} (\forall x \in W_A) P[M(x) = 1] &> t(l) + p(l)^{-1} \text{ und} \\ (\forall x \notin W_A) P[M(x) = 1] &< t(l) + p(l)^{-1} \end{aligned}$$

gilt, wobei  $l$  die Eingabelänge des Problems ist. Nun wollen wir noch einmal auf die Frage zurückkommen, wann ein Kryptosystem  $K$  mit Klartextraum  $\{0, 1\}$  und Sicherheitsparameter  $n$  in diesem Sinn als gebrochen gilt.

**Bemerkung 5.8** *Sei  $M$  eine probabilistische Polynomialzeit-Turing-Maschine. Sei  $p_0$  die Wahrscheinlichkeit für  $M(x) = 1$  unter der Voraussetzung, dass  $x$  eine Verschlüsselung der Null war und sei  $p_1$  analog für den Fall, dass  $x$  eine Verschlüsselung der Eins war definiert.*

*Dann kann  $M$  Verschlüsselungen der Null von denen der Eins unterscheiden, wenn*

$$|p_0 - p_1| > n^{-c}$$

*für ein  $c > 0$  gilt.*

Mit Hilfe von Chernoffs Grenze (Theorem 11.19) kann man zeigen, dass für jede Sprache  $A \in \mathcal{BPP}$  eine probabilistische Polynomialzeit-Turing-Maschine existiert, so dass

$$(\forall x \in \mathcal{A}^*) P[M(x) = \chi(x)] \geq 1 - 2^{-p(l)}$$

gilt (siehe [Gol 1999]).

Die Fehlerwahrscheinlichkeit einer probabilistische Polynomialzeit-Turing-Maschine kann also beliebig klein gemacht werden, ohne polynomielle Laufzeit zu verlieren.

### 5.3 Reduktion von Problemen

Der Begriff der Reduzibilität ist eine der wichtigsten Werkzeuge, die die Komplexitätstheorie entwickelt hat.

Für die meisten intuitiv berechenbaren Probleme wissen wir nicht viel über ihren Rechen- und Speicherplatzbedarf. Es ist noch nicht einmal klar, ob es Elemente in  $\mathcal{NP}$  gibt, die nicht in  $P$  liegen. Es hat sich daher als praktisch erwiesen, ein zu klassifizierendes Problem  $A$  auf ein anderes Problem  $B$  zu reduzieren. Das bedeutet: Kann man das Problem  $A$  lösen, so existiert ein effizienter Algorithmus, der das Problem  $B$  löst. Geht man davon aus, dass  $B$  schwer zu lösen ist, muss  $A$  damit mindestens genauso schwer zu lösen sein. Ist es möglich die Umkehrung auch zu beweisen sind die beiden Probleme gleich schwierig. Diese Herangehensweise hat sich als sehr brauchbar erwiesen, um die Komplexität von Problemen zu bestimmen.

Es gibt zwei verschiedene Wege eine solche Reduktion durchzuführen: Oft ist es möglich, die Instanz eines Problems vollständig durch die Lösung der Instanz eines anderen Problems zu reduzieren. Man bezeichnet eine solche Reduktion auch many-to-one reduzierbar. Ist die Laufzeit des Reduktionsalgorithmus polynomiell, so nennt man eine solche Reduktion Karpreduktion.

**Definition 5.9 (Karpreduktion)** *Eine Sprache  $L_1$  heißt karpreduzierbar in eine Sprache  $L_2$ , wenn eine Funktion  $f$  mit der Eigenschaft*

$$x \in L_1 \Leftrightarrow f(x) \in L_2$$

existiert, die in deterministische Polynomialzeit berechnet werden kann.

Eine andere Möglichkeit das Problem  $A$  auf ein Problem  $B$  zu reduzieren ist zu zeigen, dass man jede Instanz von  $A$  mit Hilfe der Lösung mehrerer Instanzen eines anderen Problems lösen kann. Die benötigten Lösungen werden durch ein Orakel simuliert. Man nennt eine Turing-Maschine, die ein Orakel verwendet auch Orakel-Turing-Maschine. Hat der Reduktionsalgorithmus polynomielle Laufzeit, so liegen die betrachteten Probleme in der gleichen Komplexitätsklasse. Der Index in den beiden folgenden Definition soll klar machen, dass  $M$  ein Orakel benutzt, welches die charakteristische Funktion der Sprache  $A_2$  effizient berechnen kann.

**Definition 5.10 (Cookreduktion)** Eine Sprache  $A_1$  heißt *reduzierbar auf eine Sprache  $A_2$  falls eine deterministische Polynomialzeit Orakel Turing Maschine  $M^{A_2}$  existiert, die  $A_1$  auf  $A_2$  reduziert.*

Wie wir bereits gesehen haben ist es sinnvoll, Probleme die in  $\mathcal{BPP}$  liegen als effizient berechenbar anzusehen. Eine Cookreduktion vor diesem Hintergrund wird durch folgende Definition beschrieben.

**Definition 5.11** Eine Sprache  $A_1$  heißt *reduzierbar auf eine Sprache  $A_2$  falls eine probabilistische Polynomialzeit Orakel Turing Maschine  $M$  existiert, so dass*

$$(\forall x \in \mathcal{A}^*) P[M(x) = \chi(x)] \geq \frac{2}{3}$$

*gilt.*

Die Konstante ist willkürlich gewählt und kann eine beliebige Zahl größer  $\frac{1}{2}$  sein.

**Definition 5.12 (NP-schwer)** Eine Sprache  $A \subseteq \{0, 1\}^*$  heißt *NP-schwer, falls jede Sprache  $B \subseteq \{0, 1\}^*$  auf  $A$  mittels einer Karpreduktion reduziert werden kann.*

Hat man einen Algorithmus gefunden, der ein NP-schweres Problem in Polynomialzeit löst, so kann man aus diesem einen Algorithmus konstruieren, der ein beliebiges  $\mathcal{NP}$ -Problem lösen kann. Die NP-vollständigen Sprachen sind die Sprachen die NP-schwer sind und in  $\mathcal{NP}$  liegen.

Nun wollen wir uns überlegen, wann eine probabilistische Polynomialzeit-Turing-Maschine keine Informationen aus dem Abstand zweier Verteilungen ziehen kann.

## 5.4 Ununterscheidbarkeit von Verteilungen

Eine weitere wichtige Frage ist die der Ununterscheidbarkeit von Verteilungen. Die Antwort auf die Frage wann zwei Verteilungen ununterscheidbar sind, resultiert aus der Beantwortung der Frage, wann Algorithmen berechenbar sind.

Um definieren zu können, wann zwei Funktionen ununterscheidbar sind, müssen wir die Gegebenheiten simulieren. Wir haben einen Sicherheitsparameter  $k$  (z.B. die Schlüssellänge). Zu jedem Parameter  $k$  haben wir eine Wahrscheinlichkeitsverteilung  $X_k$ . Eine Berechnung nennen wir nicht durchführbar, wenn sie in Abhängigkeit des Parameters mehr als polynomiell viele Schritte durchführt. Sei  $p$  ein Polynom. In der folgenden Definition beschreiben die Folge der Zufallsvariablen  $X := \{X_k\}_{k \in \mathbb{N}}$  und  $Y := \{Y_k\}_{k \in \mathbb{N}}$  zwei Folgen von Verteilungen in Abhängigkeit eines Sicherheitsparameters. Ein effizienter Algorithmus hat polynomiell beschränkte Kapazitäten. Also kann er nur polynomiell viele Werte zweier Verteilungen verarbeiten. Dies berücksichtigt folgende Definition für die Ununterscheidbarkeit zweier Verteilungen (siehe [Gol 1999]).

**Definition 5.13 (Ununterscheidbarkeit)** Sei  $s : \mathbb{N} \rightarrow \mathbb{N}$  polynomiell. Zwei Folgen von Zufallsvariablen  $X := \{X_k\}_{k \in \mathbb{N}}$  und  $Y := \{Y_k\}_{k \in \mathbb{N}}$  heißen ununterscheidbar, wenn für jeden probabilistischen Polynomialzeitalgorithmus  $D$  und jedes Polynom  $p(\cdot)$  ein  $K \in \mathbb{N}$  so existiert, dass für alle  $k \geq K$  die Bedingung

$$|P[D(X_k^{(1)}, \dots, X_k^{(s(k))}) = 1] - P[D(Y_k^{(1)}, \dots, Y_k^{(s(k))}) = 1]| < \frac{1}{p(k)}$$

erfüllt ist.

Die Zufallsvariablen  $X_k^{(i)}$  und  $Y_k^{(i)}$  mit  $i \in [s(k)]$  sind unabhängig und identisch  $X_k$  bzw.  $Y_k$  verteilt.

Sei  $K$  ein Kryptosystem mit Klartextraum  $\{0,1\}$ . Die Verschlüsselungen der Null und der Eins erzeugen Verteilungen  $X_k$  bzw.  $Y_k$  in Abhängigkeit des Sicherheitsparameters  $k$ . Ist der Abstand zwischen den beiden Zufallsvariablen kleiner als jedes Polynom in Abhängigkeit des Sicherheitsparameters, so kann kein Angreifer aus Schlüsseltexten Informationen über deren Klartexte erlangen. Eine Funktion, die kleiner als jedes Polynom ist wollen wir zu vernachlässigend nennen.

**Definition 5.14** Eine Funktion  $f : \mathbb{R} \rightarrow \mathbb{R}$  heißt zu vernachlässigend, wenn für jedes Polynom  $p$  und ausreichend großes  $k$  die Bedingung  $f(k) < \frac{1}{p(k)}$  erfüllt ist.

Multipliziert man solche Funktionen mit einem Polynom, bleiben sie eine zu vernachlässigende Funktion.

Aus dieser Definition resultiert der Begriff, wann die Wahrscheinlichkeit für ein Ereignis  $A$  zu vernachlässigen ist. Hierbei ist  $A$  wieder eine Folge von Ereignissen  $(A_k)_{k \in \mathbb{N}}$  in Abhängigkeit eines Sicherheitsparameters.

**Definition 5.15** Sei  $k$  ein Sicherheitsparameter. Die Wahrscheinlichkeit für ein Ereignis  $A = (A_k)_{k \in \mathbb{N}}$  heißt zu vernachlässigend, falls für jedes Polynom  $p$  ein  $K \in \mathbb{N}$  so existiert, dass für alle  $k \geq K$

$$p(A_k) < \frac{1}{p(k)}$$

gilt.

Ein Ereignis, das diese Bedingung nicht erfüllt, heißt nicht zu vernachlässigend.

Man sagt, dass zwei Folgen von Zufallsvariablen statistisch nah beieinander liegen, wenn für jedes Polynom  $p$  und ausreichend großes  $k$  der statistische Abstand (Definition 4.17) zwischen den beiden Verteilungen kleiner als  $\frac{1}{p(k)}$  ist. Es ist klar, dass diese dann auch nicht unterscheidbar sind. Allerdings gibt es Folgen, die statistisch weit auseinander liegen, und trotzdem ununterscheidbar sind (siehe dazu [Has]).

In Kapitel 8 und 9 werden wir zeigen, dass ein Angreifer der Verschlüsselungen der 0 und der 1 mit nicht zu vernachlässigender Wahrscheinlichkeit unterscheiden kann, auch mit nicht zu vernachlässigender Wahrscheinlichkeit das *uSVP* lösen kann. Dafür dürfen bestimmte Verteilungen nicht unterscheidbar sein. Um dies sicherzustellen werden Konstanten so gewählt, dass der statistische Abstand in Abhängigkeit des Sicherheitsparameters  $2^{-k}$  ist. Wir nennen einen solchen Abstand zwischen Verteilungen auch exponentiell klein.

## 5.5 Average-Case Komplexität

Bis jetzt haben wir Probleme nur nach der maximalen Dauer, die ein Algorithmus braucht, um ein Problem zu lösen, klassifiziert (worst-case Komplexität). Es ist aber nicht klar, ob ein Problem dessen schwierigster Fall schwer zu lösen ist, im allgemeinen schwer ist. In der Kryptographie ist diese Fragestellung von großem Interesse. Der Grund hierfür liegt darin, dass die Schwierigkeit eines Kryptosystem auf der Instanz eines Problems beruht. Daher ist es wichtig zu wissen, wie aufwändig es ist, eine zufällig ausgewählte Instanz eines Problems zu knacken. In [Ajt 1997] konnte Ajtai zeigen, dass es genauso schwer ist, eine zufällige Instanz des *uSVP* zu lösen, wie den schwierigsten Fall zu lösen. Da das in dieser Diplomarbeit vorgestellte Verfahren auf diesem Problem basiert, ist eine zufällig ausgewählte Instanz in der Regel schwer zu lösen. Dies ist ein großer Vorteil gegenüber anderen Systemen, bei denen schwere Instanzen eines Problems generiert werden müssen.

## 6 Das Kryptosystem

In diesem Kapitel wollen wir das Kryptosystem konstruieren. Um dies tun zu können, müssen wir uns einige Verteilungen auf dem Intervall  $I = [0, 1)$  definieren. Es handelt sich hierbei um Normalverteilungen, die man auf  $I$  reduziert. Die Idee, die hinter dieser Reduktion steht ist folgende. Hat man die Dichtefunktion  $\rho$  einer Verteilung auf  $\mathbb{R}$  gegeben, so erhält man den Funktionswert der reduzierten Verteilung  $\rho \bmod 1$  in dem Punkt  $x \in [0, 1)$ , durch die Summe aller Funktionswerte der Elemente  $y \in \mathbb{R}$ , die modulo Eins gleich  $x$  sind. Anschaulich macht es Sinn, dass man auf diese Weise eine Verteilung auf  $I$  erhält.

### 6.1 Einige Verteilungen

In diesem Kapitel werden wir die für das Kryptosystem benötigten Verteilungen konstruieren.

**Definition 6.1 (Gleichverteilung)** (i) Eine Zufallsvariable  $X$  heißt gleichverteilt auf dem Intervall  $[0, 1)$ , wenn  $X$  stetig mit der Dichte

$$f : [0, 1) \rightarrow \mathbb{R} \\ f(t) = 1$$

verteilt ist.

(ii) Eine Zufallsvariable  $X$  heißt gleichverteilt auf der Menge  $[n]$ , wenn

$$P(X = i) = \frac{1}{n}$$

gilt.

**Definition 6.2 (Normalverteilung)** Sei  $\mu \in \mathbb{R}$  und  $\sigma > 0$ . Eine Zufallsvariable  $X$  heißt normalverteilt mit Eigenwert  $\mu$  und Varianz  $\sigma^2$ , falls  $X$  stetig mit Dichte

$$f : \mathbb{R} \rightarrow \mathbb{R} \\ f(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2}$$

verteilt ist.

Wir werden in einem der späteren Kapiteln noch die folgende wichtige Eigenschaft normalverteilter Zufallsvariablen benötigen.

**Lemma 6.3** Seien  $X_1$  und  $X_2$  unabhängige, normalverteilte Zufallsvariablen mit Erwartungswerten  $\mu_i$  und Varianzen  $\sigma_i^2$ ,  $i=1,2$ , so ist die Summe  $X_1 + X_2$  normalverteilt mit Eigenwert  $\mu = \mu_1 + \mu_2$  und Varianz  $\sigma_1^2 + \sigma_2^2$ .

Beweis: siehe [Leh 2000].

Das Lemma kann auch auf  $n$  unabhängige, normalverteilte Zufallsvariablen  $X_1, \dots, X_n$  verallgemeinert werden. Ist  $X$  die Summe dieser Zufallsvariablen, so gilt

$$X = Y + Z$$

mit  $Y = X_1 + X_2$  und  $Z = X_3 + \dots + X_n$  darstellen. Auf die Zufallsvariable  $Y$  kann das Lemma angewandt. Induktiv erhält man, dass die sich Varianzen und Erwartungswerte genauso wie im Falle zweier Zufallsvariablen addieren.

Nun definieren wir uns eine Funktion, die aus einer Normalverteilung eine Verteilung auf dem Intervall  $[0,1)$  macht.

**Definition 6.4** Sei  $\beta \in \mathbb{R}^+$ . Dann ist  $Q_\beta$  definiert durch

$$Q_\beta(r) = \sum_{k=-\infty}^{\infty} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(r-k)^2}.$$

**Lemma 6.5** Die Funktion  $Q_\beta$  ist eine Dichtefunktion auf  $[0,1)$ .

Beweis: Sei  $f_n := \sum_{k=-n}^n \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(r-k)^2}$ .  $(f_n)_{n \in \mathbb{N}}$  ist eine aufsteigende Folge messbarer Funktionen  $0 \leq f_1 \leq f_2 \leq \dots$  mit  $Q_\beta(r) = \lim_{n \rightarrow \infty} f_n(r)$ .

Nach dem Satz von Beppo Levi (Theorem 3.12) ist  $f$  messbar und es gilt:

$$\begin{aligned} \int_0^1 Q_\beta(r) dr &= \lim_{n \rightarrow \infty} \int_0^1 \sum_{k=-n}^n \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(r-k)^2} dr \\ &= \lim_{n \rightarrow \infty} \sum_{k=-n}^n \int_0^1 \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(r-k)^2} dr \\ &= \lim_{n \rightarrow \infty} \int_{-n}^{n+1} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}r^2} dr \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}r^2} dr. \end{aligned}$$

Dies ist eine Normalverteilung mit Eigenwert 0 und Varianz  $\frac{\beta}{2\pi}$ . Insbesondere hat das Integral über diese Funktion den Wert 1, was zu zeigen war.

Offensichtlich ist  $Q_\beta(r+z) = Q_\beta(r)$  für  $z \in \mathbb{Z}$ . Die Funktion ist also periodisch mit Periode 1.

**Definition 6.6** Sei  $h \in \mathbb{N}$  und  $\beta \in \mathbb{R}^+$ . Dann ist die Funktion  $T_{h,\beta} : [0,1) \rightarrow \mathbb{R}$  definiert durch

$$T_{h,\beta}(r) := Q_\beta(rh \bmod 1) = \sum_{k=-\infty}^{\infty} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(rh-k)^2}.$$

**Lemma 6.7**  $T_{h,\beta}$  ist eine Dichtefunktion auf  $[0,1)$ .

Beweis: Sei  $f_n(r) := \sum_{k=-n}^n \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(rh-k)^2}$

Dann ist  $f_1 \leq f_2 \leq f_3, \dots$  eine Folge messbarer Funktionen, von  $[0,1)$  nach  $\mathbb{R}$ . Die  $f_n$  sind so gewählt, dass  $T_{h,\beta}(r) = \lim_{n \rightarrow \infty} f_n(r)$ . Nach dem Satz von Beppo Levi (Theorem 3.12) gilt folgende Gleichheit:

$$\int_0^1 T_{h,\beta}(r) dr = \lim_{n \rightarrow \infty} \int_0^1 f_n(r) dr.$$

Um den Wert des Integrals zu berechnen, substituiert man  $z = rh$  und erhält

$$\begin{aligned} \int_0^1 T_{h,\beta}(r) dr &= \lim_{n \rightarrow \infty} \int_0^1 \sum_{k=-n}^n \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(rh-k)^2} dr \\ &= \lim_{n \rightarrow \infty} \frac{1}{h} \int_0^h \sum_{k=-n}^n \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(z-k)^2} dz \\ &= \lim_{n \rightarrow \infty} \frac{1}{h} \left[ \sum_{i=0}^{h-1} \int_i^{i+1} \sum_{k=-n}^n \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(z-k)^2} dz \right] \\ &\stackrel{(1)}{=} \lim_{n \rightarrow \infty} \frac{1}{h} \left[ h \int_0^1 \sum_{k=-n}^n \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(z-k)^2} dz \right] \\ &= \lim_{n \rightarrow \infty} \int_{-n}^{n+1} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(z-k)^2} dz \\ &= 1. \end{aligned}$$

Die Gleichung (1) gilt, da die Funktion  $Q_\beta$  periodisch mit Periode Eins ist. Damit ist  $T_{h,\beta}$  ebenfalls eine Verteilung auf  $[0,1)$ .

Es gilt  $T_{h,\beta}(r + \frac{1}{h}) = T_{h,\beta}(r)$ . Die Anzahl der Perioden der Funktion  $T_{h,\beta}$  im Intervall  $[0,1)$  ist also  $h$ .

### Ein Verfahren zum Generieren von $T_{h,\beta}$ -verteilten Werten

- 1) Wähle zufällig einen Wert  $x$  aus der Menge  $\{1, \dots, h-1\}$  aus.
- 2) Wähle einen  $Q_\beta$ -verteilten Wert  $y \in [0,1)$
- 3) Gebe  $\frac{x+y}{h}$  aus.

Die Funktion  $T_{h,\beta}$  ist  $\frac{1}{h}$ -periodisch. Die Wahl von  $x$  stellt sicher, dass alle Perioden gleich wahrscheinlich sind. In einer Periode  $[\frac{i}{h}, \frac{i+1}{h})$  sind die Werte  $r \cdot h$ ,  $r \in [\frac{i}{h}, \frac{i+1}{h})$   $Q_\beta$ -verteilt. Also generiert man tatsächlich  $T_{h,\beta}$ -verteilte Werte. Dieses Verfahren ist offensichtlich effizient.

**Definition 6.8** Sei  $c_h$  die Konstante aus Lemma 8.15 und  $g(n)$  eine Funktion mit positivem Wertebereich. Die Menge  $\Upsilon_{g(n)}$  ist definiert durch

$$\Upsilon_{g(n)} := \{T_{h,\beta} \mid h \in \mathbb{N}, h \leq 2^{c_h n^2}, \beta \in [\frac{n}{(g(n))^2}, 4\frac{n}{(g(n))^2}]\}.$$

## 6.2 Das Kryptosystem

Sei  $n \in \mathbb{N}$  ein Sicherheitsparameter,  $N = 2^{c_N n^2}$  und  $m = c_m n^2$  wobei  $c_N$  und  $c_m$  zwei Konstanten sind, die später bestimmt werden. Sei  $\gamma(n)$  eine Funktion mit  $\frac{\gamma(n)}{n\sqrt{\log n}} \rightarrow \infty$  für  $n$  gegen unendlich. Wir werden später sehen, dass die Sicherheit des Verfahrens steigt, je kleiner die Funktion  $\gamma$  gewählt wird.

**Privater Schlüssel:** Sei  $H = \{h \in [\sqrt{N}, 2\sqrt{N}) \mid \text{frc}(h) < \frac{1}{16m}\}$ . Wähle  $h \in H$  zufällig und gleichverteilt. Sei  $d := \frac{N}{h}$ . Der private Schlüssel ist  $h$ .

**Öffentlicher Schlüssel:** Wähle  $\beta$  zufällig aus dem Intervall  $[4\frac{1}{(\gamma(n))^2}, 8\frac{1}{(\gamma(n))^2})$ . Es werden  $m$   $T_{h,\beta}$ -verteilte Werte  $z_1, z_2, \dots, z_m$  generiert, indem man Werte  $x_1, x_2, \dots, x_m$  und  $y_1, y_2, \dots, y_m$  wie in Teil 6.1 beschrieben wählt und addiert. Sei  $i_0$  ein Index für den  $x_{i_0}$  ungerade ist (Die Wahrscheinlichkeit dafür, dass so ein Index existiert, ist  $(1 - (\frac{1}{2})^m)$ ). Für  $i \in [m]$  sei  $a_i := \lfloor N \cdot z_i \rfloor$ . Der öffentliche Schlüssel ist  $(a_1, a_2, \dots, a_m, i_0)$ .

**Verschlüsselung:** Zur Verschlüsselung eines Bits wird zufällig eine Menge  $S \subseteq [m]$  gewählt. Ist das Bit 0, so ist der Schlüsseltext  $\sum_{i \in S} a_i \bmod N$ . Ansonsten ist er  $\sum_{i \in S} a_i + \lfloor \frac{a_{i_0}}{2} \rfloor \bmod N$ .

**Entschlüsselung:** Sei  $w \in \{0, 1, \dots, N\}$  der Schlüsseltext. Ist  $\text{frc}(\frac{w}{d}) < \frac{1}{4}$  so ist die Entschlüsselung 0, ansonsten 1.

## 7 Einige technische Beweise

Wir werden später beweisen, dass die Sicherheit des Kryptosystems auf der Sicherheit des uSVP basiert. Dazu müssen wir das uSVP auf das Problem zwei Verteilungen zu unterscheiden reduzieren. Um die Reduktion durchführen zu können, benötigen wir einige technische Eigenschaften von verschiedenen Funktionen und von LLL-reduzierten Basen, die wir in diesem Kapitel definieren und beweisen werden. In diesem und den folgenden Kapiteln beschäftigen wir uns nur noch mit volldimensionalen Gittern.

**Lemma 7.1** *Die Wahrscheinlichkeit, dass eine normalverteilte Zufallsvariable mit Varianz  $\sigma^2$  Werte annimmt, die einen größeren Abstand als  $t$  von ihrem Eigenwert haben, ist höchstens*

$$\sqrt{\frac{2}{\pi}} \cdot \frac{\sigma}{t} e^{-\frac{t^2}{2\sigma^2}}.$$

Beweis: Sei  $\mu$  der Eigenwert der Normalverteilung. Falls der Eigenwert  $\mu$  der Zufallsvariable ungleich null ist, kann man durch Substitution von  $Y = X - \mu$  die Aussage auf den Fall  $\mu = 0$  reduzieren. Es reicht also den Fall  $\mu = 0$  zu betrachten.

Es gilt

$$\begin{aligned} \int_t^\infty \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} dx &\leq \int_t^\infty \left(1 + \frac{\sigma^2}{x^2}\right) \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} dx \\ &\stackrel{(1)}{=} -\frac{1}{\sqrt{2\pi}\sigma} \cdot \frac{\sigma^2}{x} e^{-\frac{x^2}{2\sigma^2}} \Big|_t^\infty \\ &= \frac{\sigma}{\sqrt{2\pi}t} e^{-\frac{t^2}{2\sigma^2}}. \end{aligned}$$

Die Gleichheit (1) folgt aus  $\frac{d}{dx} \left(-\frac{1}{\sqrt{2\pi}\sigma} \cdot \frac{\sigma^2}{x} e^{-\frac{x^2}{2\sigma^2}}\right) = \left(1 + \frac{\sigma^2}{x^2}\right) \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$ .

Die Wahrscheinlichkeit, dass der Abstand von einer normalverteilten Zufallsvariable zu ihrem Eigenwert größer als  $t$  ist, ist also höchstens  $2 \cdot \frac{\sigma}{\sqrt{2\pi}t} e^{-\frac{t^2}{2\sigma^2}}$ . Damit ist die Behauptung bewiesen.

**Lemma 7.2**  $(\forall x, r \in \mathbb{R}) \sum_{k \in \mathbb{Z}} e^{-\pi(kr+x)^2} \leq 1 + \frac{1}{r}$

Für  $r = 0$  setze  $\frac{1}{r} = \infty$ .

Beweis: Für jedes  $r$  existiert ein  $m \in \mathbb{Z}$ , so dass  $|mr+x|$  minimal ist. Die Summe  $\sum_{k \in \mathbb{Z}} e^{-(kr+x)^2}$  ist gleich dem Integral der Stufenfunktion

$$\sum_{k \in \mathbb{Z}} e^{-\pi(kr+x)^2} \chi_{[k, k+1]}.$$

Nun betrachten wir die Funktion

$$f : \mathbb{R} \rightarrow \mathbb{R}, k \mapsto e^{-\pi(kr+x)^2}$$

Wir stellen uns die beiden Funktionen als Graphen vor. Der Graph der Stufenfunktion liegt für Werte, deren Argument kleiner als  $m$  ist, unter dem Graphen von  $f$ . Wenn wir die  $m$ -te Stufe weglassen und die restlichen Stufen nach links „verschieben“ liegt der resultierende Graph komplett unter dem Graphen von  $f$ . Da die Summanden nicht größer als 1 sind, gilt mit Substitution  $y := kr + x$

$$\begin{aligned}
\sum_{k \in \mathbb{Z}} e^{-\pi(kr+x)^2} &\leq 1 + \sum_{k \in \mathbb{Z} \setminus \{m\}} e^{-\pi(kr+x)^2} \\
&= 1 + \frac{1}{r} \left( \sum_{k \in \mathbb{Z} \setminus \{m\}} r e^{-\pi(kr+x)^2} \right) \\
&= 1 + \frac{1}{r} \int_{-\infty}^{\infty} \left( \sum_{k \in \mathbb{Z} \setminus \{m\}} r e^{-\pi(kr+x)^2} \chi_{[k, k+1]} \right) dk \\
&\leq 1 + \frac{1}{r} \int_{-\infty}^{\infty} r e^{-\pi(kr+x)^2} dk \\
&= 1 + \frac{1}{r} \int_{-\infty}^{\infty} e^{-\pi(y^2)} dy.
\end{aligned}$$

Die Funktion  $e^{-\pi(y^2)}$  ist eine Normalverteilung mit Eigenwert 0 und Varianz  $\frac{1}{\sqrt{2\pi}}$ . Somit ist der Wert des Integrals 1 und die Behauptung bewiesen.

**Lemma 7.3** Für alle  $a, x \in \mathbb{R}$  und alle  $b > \frac{1}{\sqrt{2\pi}} + 1$  existiert eine Konstante  $c$  mit  $\left| \frac{d}{dx} \sum_{k \in \mathbb{Z}} e^{-\pi(bk+ax)^2} \right| \leq ca$ .

Beweis: Um das Integral bestimmen zu können, müssen wir als erstes zeigen das Summation und Ableitung vertauscht werden können. Dazu wollen wir Satz 11.7 auf die Funktionenfolge

$$\left( \sum_{k=-n}^n e^{-\pi(bk+ax)^2} \right)_{n \in \mathbb{N}}$$

anwenden.

Sei  $t \in \mathbb{R}$  und  $[d, e]$  ein Intervall, dass den Punkt  $t$  enthält.

Sei  $f_n : [d, e] \rightarrow \mathbb{R}$ ,  $x \mapsto \sum_{k=-n}^n e^{-\pi(bk+ax)^2}$ . Die Funktionenfolge  $(f_n)_{n \in \mathbb{N}}$  konvergiert offensichtlich punktweise gegen

$$\sum_{k \in \mathbb{Z}} e^{-\pi(bk+ax)^2}.$$

Da jedes Folgenglied  $f_n$  aus einer endlichen Summe von Funktionen besteht, lässt sich Summation und Ableitung vertauschen und somit ist

$$(f'_n)_{n \in \mathbb{N}} = \left( \frac{d}{dx} \sum_{k=-n}^n e^{-\pi(bk+ax)^2} \right)_{n \in \mathbb{N}} = \left( \sum_{k=-n}^n -2a\pi(bk+ax)e^{-\pi(bk+ax)^2} \right)_{n \in \mathbb{N}}$$

die Folge der Ableitungen. Auf dem Intervall  $[d, e]$  lassen sich die Beträge der Funktionen  $e^{-\pi(bk+ax)^2}$  und  $-2a\pi(bk+ax)$  für jedes  $k$  durch ihr Supremum abschätzen und man erhält eine Abschätzung

$$l_k := \sup_{x \in [d, e]} |-2a\pi(bk+ax)| \cdot \sup_{x \in [d, e]} |e^{-\pi(bk+ax)^2}|$$

für den Betrag von  $f$ . Die Reihe  $\sum_{k=-\infty}^{\infty} l_k$  konvergiert. Also konvergiert

$$\left(\frac{d}{dx} \sum_{k=-n}^n e^{-\pi(bk+ax)^2} = \sum_{k=-n}^n -2a\pi(bk+ax)e^{-\pi(bk+ax)^2}\right)_{n \in \mathbb{N}}$$

auf dem Intervall  $[d, e]$  gleichmäßig gegen die Funktion

$$f'(x) := \sum_{k=-\infty}^{\infty} -2a\pi(bk+ax)e^{-\pi(bk+ax)^2}.$$

Nach Satz 11.7 ist  $f$  auf dem Intervall  $[d, e]$  differenzierbar mit Ableitung  $f'$ . Da  $t$  beliebig ist haben wir bewiesen, dass Differentiation und Summation vertauscht werden kann. Um den Wert des Integrals abzuschätzen, setzen wir  $z := a \cdot x$  und erhalten

$$\begin{aligned} |a| \left| \frac{d}{dz} \sum_{k \in \mathbb{Z}} e^{-\pi(bk+z)^2} \right| &= |a| \left| \sum_{k \in \mathbb{Z}} -2\pi(bk+z) e^{-\pi(bk+z)^2} \right| \\ &\leq |a| \sum_{k \in \mathbb{Z}} |-2\pi(bk+z) e^{-\pi(bk+z)^2}|. \end{aligned}$$

Also müssen wir nur noch eine Schranke für  $\sum_{k \in \mathbb{Z}} |-2\pi(bk+z) e^{-\pi(bk+z)^2}|$  finden.

Da die Funktion offensichtlich periodisch ist, reicht es sie für  $z \in [0, b]$  zu untersuchen.

In diesem Fall ist

$$\sum_{k \in \mathbb{Z}} |-2\pi(bk+z) e^{-\pi(bk+z)^2}| \leq 2 \left( \sum_{k \in \mathbb{N}_0} |-2\pi(bk+z) e^{-\pi(bk+z)^2}| \right) \quad (*).$$

Wie im Lemma zuvor stellt man die Summe als Integral einer Stufenfunktion dar. Diese Funktion wollen wir durch die Funktion  $2\pi z e^{-\pi z^2}$  majorisieren. Durch eine einfache Kurvendiskussion sieht man, dass die Funktion  $|2\pi z e^{-\pi z^2}|$  von 0 bis  $\frac{1}{\sqrt{2\pi}}$  steigt, wo sie ihr Maximum  $\frac{\sqrt{2\pi}}{\sqrt{e}}$  annimmt. Danach fällt sie.

Deshalb und da  $b$  größer als  $\frac{1}{\sqrt{2\pi}} + 1$  ist, fällt die Reihe durch das Weglassen der höchsten Stufe unter den Graphen der Funktion und es folgt

$$\begin{aligned} \sum_{k \in \mathbb{N}_0} |-2\pi(bk+z) e^{-\pi(bk+z)^2}| &= |2\pi z e^{-\pi z^2}| + \sum_{k \in \mathbb{N}} |-2\pi(bk+z) e^{-\pi(bk+z)^2}| \\ &\leq \frac{\sqrt{2\pi}}{\sqrt{e}} + \int_0^{\infty} |2\pi z e^{-\pi z^2}| dz \\ &= d. \end{aligned}$$

Mit (\*) ist die Aussage bewiesen.

Im Folgenden bezeichnen wir mit  $d(L^*)$  die Gitterdeterminante des zu  $L$  dualen Gitter  $L^*$  (siehe dazu Definition 2.18 und 2.24).

**Lemma 7.4** *Für alle  $v \in L$  mit  $v \neq 0$  gilt*

$$\int_{\mathcal{P}(L^*)} \sum_{k \in \mathbb{Z}} e^{-\pi \left( \frac{k + \langle v, x \rangle}{\|v\|} \right)^2} dx = \|v\| d(L^*).$$

Beweis: Als erstes zeigen wir, dass das Integral nicht von der Grundmasche, d.h. von der Wahl der Basis  $B = (b_1^*, \dots, b_n^*)$  (siehe Satz 2.20), abhängt. Hierzu muss bewiesen werden, dass der Wert des Integrals invariant unter Basistransformationen ist. Nach Satz 2.17 müssen wir die Invarianz unter den Transformationen  $b_i^* = -b_i^*$  und  $b_i^* = b_i^* + b_j^*$  mit  $i \neq j$  zeigen.

Sei  $f(x) := \sum_{k \in \mathbb{Z}} e^{-\pi \left( \frac{k + \langle v, x \rangle}{\|v\|} \right)^2}$ .

Man beachte, dass für jedes  $w \in L^*$  das Skalarprodukt  $\langle v, w \rangle$  eine ganze Zahl ist (siehe Definition 2.24). Also folgt für  $w \in L^*$   $f(x + w) = f(x)$ . Für den Beweis benötigen folgende Mengen:

$$\mathcal{P}_1 = \left\{ \sum_{i=1}^n \alpha_i b_i^* \mid \alpha_i \in [0, 1), \alpha_2 \geq \alpha_1 \right\}$$

$$\mathcal{P}_2 = \left\{ \sum_{i=1}^n \alpha_i b_i^* \mid \alpha_i \in [0, 1), \alpha_2 < \alpha_1 \right\}$$

$$\mathcal{P}_3 = \left\{ \sum_{i=1}^n \alpha_i b_i^* + b_2^* \mid \alpha_i \in [0, 1), \alpha_2 < \alpha_1 \right\}.$$

Die Mengen sind disjunkt und so gewählt, dass  $\mathcal{P}(L^*) = \mathcal{P}(b_1^*, b_2^*, \dots, b_n^*) = \mathcal{P}_1 \cup \mathcal{P}_2$  und  $\mathcal{P}(b_1^* + b_2^*, b_2^*, \dots, b_n^*) = \mathcal{P}_1 \cup \mathcal{P}_3$ . Also gilt wegen  $f(x + b_2^*) = f(x)$

$$\begin{aligned} \int_{\mathcal{P}(L^*)} f(x) dx &= \int_{\mathcal{P}_1} f(x) dx + \int_{\mathcal{P}_2} f(x) dx \\ &= \int_{\mathcal{P}_1} f(x) dx + \int_{\mathcal{P}_2} f(x + b_2^*) dx \\ &= \int_{\mathcal{P}_1} f(x) dx + \int_{\mathcal{P}_3} f(x) dx \\ &= \int_{\mathcal{P}(b_1^* + b_2^*, b_2^*, \dots, b_n^*)} f(x) dx. \end{aligned}$$

Nun bleibt noch zu zeigen, dass das Integral unverändert bleibt, wenn der Basisvektor  $b_i^*$  durch  $-b_i^*$  ersetzt wird.

Wie oben schon bemerkt wurde, gilt  $f(x) = f(x + w)$  für  $w \in L^*$ . Insbesondere ist  $f(x) = f(x - b_1^*)$ . Also gilt

$$\begin{aligned} \int_{\mathcal{P}(b_1^*, b_2^*, \dots, b_n^*)} f(x) dx &= \int_{\mathcal{P}(b_1^*, b_2^*, \dots, b_n^*)} f(x - b_1^*) dx \\ &= \int_{\mathcal{P}(-b_1^*, b_2^*, \dots, b_n^*)} f(x) dx. \end{aligned}$$

Damit ist die Invarianz gegenüber Basistransformationen bewiesen und wir können den Wert ausrechnen.

Nach Bemerkung 2.29 existiert eine Basis  $B^* = (w, b_2^*, \dots, b_n^*)$  von  $L^*$  mit der Eigenschaft, dass alle Vektoren  $b_i^*$  senkrecht auf dem Vektor  $v$  stehen und dass das Skalarprodukt  $\langle w, v \rangle = j$  ist. Sei  $\mathcal{P}$  die zu dieser Basis gehörende Grundmasche. Als erstes wollen wir eine Vorüberlegung machen, die wir in dem Beweis benötigen.

Da  $v$  senkrecht auf allen Vektoren außer  $w$  steht, gilt

$$d(L^*) = \left\langle \frac{v}{\|v\|}, w \right\rangle d(\mathcal{P} \cap \{y | \langle y, v \rangle = 0\}).$$

Wegen  $d(\mathcal{P} \cap \{y | \langle y, v \rangle = a\}) = d(\mathcal{P} \cap \{y | \langle y, v \rangle = 0\})$  folgt für alle  $a \in [0, j]$ ,

$$\|v\| d(L^*) = \langle w, v \rangle d(\mathcal{P} \cap \{y | \langle y, v \rangle = a\}) = j \cdot d(\mathcal{P} \cap \{y | \langle y, v \rangle = a\}).$$

Also gilt

$$\frac{\|v\|}{j} d(L^*) = d(\mathcal{P} \cap \{y | \langle y, v \rangle = 0\}). \quad (4)$$

Sei  $U := \frac{\langle v, w \rangle}{\|v\|^2} v \times \mathcal{P} \cap \{c \in \mathcal{P} | \langle c, v \rangle = 0\}$ . Da  $B^*$  eine Basis des  $\mathbb{R}^n$  ist, die Vektoren  $b_2^*, \dots, b_n^*$  senkrecht auf  $v$  stehen und weil  $\frac{\langle v, w \rangle}{\|v\|^2} v$  die Projektion des Vektors  $w$  in den Vektor  $v$  ist, existieren eindeutig bestimmte Zahlen  $\tilde{\lambda}_2, \tilde{\lambda}_3 \dots \tilde{\lambda}_n$  mit

$$w = \frac{\langle v, w \rangle}{\|v\|^2} v + \tilde{\lambda}_2 b_2^* + \dots + \tilde{\lambda}_n b_n^*.$$

Sei  $A$  die bijektive lineare Abbildung

$$A : \mathcal{P} \rightarrow U$$

$$(\lambda_1 w + \lambda_2 b_2^* + \dots + \lambda_n b_n^*) \mapsto (\lambda_1 w + \lambda_2 b_2^* + \dots + \lambda_n b_n^* - \lambda_1 (\tilde{\lambda}_2 b_2^* + \dots + \tilde{\lambda}_n b_n^*)).$$

Die Abbildung  $A$  zur Basis  $B^*$  wird durch die Matrix

$$A_{B^*} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ -\tilde{\lambda}_1 & 1 & \ddots & \vdots \\ -\tilde{\lambda}_2 & 0 & \ddots & \\ \vdots & \vdots & \ddots & 0 \\ -\tilde{\lambda}_n & 0 & \dots & 0 & 1 \end{pmatrix}$$

beschrieben. Die Determinante von  $A_B^*$  ist 1. Da die Vektoren  $b_2^*, \dots, b_n^*$  senkrecht auf  $v$  stehen, gilt

$$\begin{aligned} f(A(\lambda_1 w + \lambda_2 b_2^* + \dots + \lambda_n b_n^*)) &= f\left(\frac{\lambda_1 \langle v, w \rangle}{\|v\|^2} v + \lambda_2 b_2^* + \dots + \lambda_n b_n^*\right) \\ &= e^{-\pi \left(\frac{k + \langle v, \frac{\lambda_1 \langle v, w \rangle}{\|v\|^2} v + \lambda_2 b_2^* + \dots + \lambda_n b_n^* \rangle}{\|v\|}\right)^2} \\ &= e^{-\pi \left(\frac{k + \langle v, \frac{\lambda_1 \langle v, w \rangle}{\|v\|^2} v \rangle}{\|v\|}\right)^2} \\ &= e^{-\pi \left(\frac{k + \lambda_1 \langle v, w \rangle}{\|v\|}\right)^2} \\ &= f(\lambda_1 w + \lambda_2 b_2^* + \dots + \lambda_n b_n^*). \end{aligned}$$

Jetzt wenden wir die Transformationsformel (Theorem 3.21) mit der linearen Abbildung  $A$  an und erhalten

$$\int_{\mathcal{P}} f(x) dx = \int_{\mathcal{P}} f(Ax) |\det(A)| dx = \int_U f(x) dx.$$

Nach dem Satz von Fubini (Theorem 3.15) ist die Integrationsreihenfolge un-  
erheblich und wir können in beliebiger Reihenfolge integrieren.

$$\int_U f(x) dx = \int_{[0, \frac{\langle v, w \rangle}{\|v\|^2} v]} \int_{\mathcal{P} \cap \{y | \langle y, v \rangle = 0\}} f(z, y) dy dz.$$

Als nächstes wollen wir die Menge  $U$  parametrisieren, um wieder die Transformationsformel anzuwenden. Sei dazu

$$\begin{aligned} \phi : [0, j] \times \mathcal{P} \cap \{y | \langle y, v \rangle = 0\} &\rightarrow [0, \frac{\langle v, w \rangle}{\|v\|^2} v] \times \mathcal{P} \cap \{y | \langle y, v \rangle = 0\} \\ (\lambda, p) &\mapsto \left(\lambda \frac{v}{\|v\|^2}, p\right) \end{aligned}$$

Die Determinante von  $\phi$  ist  $\frac{1}{\|v\|}$ . Also gilt

$$\int_{[0, \frac{\langle v, w \rangle}{\|v\|^2} v]} \int_{\mathcal{P} \cap \{y | \langle y, v \rangle = 0\}} f(z, y) dy dz = \frac{1}{\|v\|} \int_{[0, j]} \int_{\mathcal{P} \cap \{y | \langle y, v \rangle = 0\}} f\left(a \frac{v}{\|v\|^2}, y\right) dy da.$$

Der Funktionswert von  $f$  an der Stelle  $x$  hängt nur von der Größe des Koeffizienten vor  $v$  ab. Mit (4) folgt

$$\begin{aligned} \frac{1}{\|v\|} \int_{[0, j]} \int_{\mathcal{P} \cap \{y | \langle y, v \rangle = 0\}} f(x) dx &= \frac{1}{\|v\|} \int_0^j \int_{\mathcal{P} \cap \{y | \langle y, v \rangle = 0\}} \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k+a}{\|v\|}\right)^2} dx da \\ &= \frac{1}{\|v\|} \int_0^j \frac{\|v\|}{j} d(L^*) \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k+a}{\|v\|}\right)^2} da. \end{aligned}$$

Da die Funktion über die integriert wird periodisch mit Periode 1 ist haben wir bis jetzt

$$\begin{aligned} \int_{\mathcal{P}} f(x) dx &= \frac{1}{\|v\|} \int_0^j \frac{\|v\|}{j} d(L^*) \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k+a}{\|v\|}\right)^2} da \\ &= \int_0^1 d(L^*) \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k+a}{\|v\|}\right)^2} da \\ &= d(L^*) \int_{-\infty}^{\infty} e^{-\pi \left(\frac{a}{\|v\|}\right)^2} da \end{aligned}$$

gezeigt. Nun zum Beweis von  $d(L^*) \int_{-\infty}^{\infty} e^{-\pi \left(\frac{a}{\|v\|}\right)^2} da = \|v\| d(L^*)$ .

Substituiere  $z = \frac{\sqrt{2\pi}}{\|v\|} a$ . Dann ist  $da = \frac{\|v\|}{\sqrt{2\pi}} dz$  und es folgt

$$d(L^*) \int_{-\infty}^{\infty} e^{-\pi \left(\frac{a}{\|v\|}\right)^2} da = d(L^*) \frac{\|v\|}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\frac{1}{2} z^2} dz.$$

Die Funktion  $\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} z^2}$  ist die Normalverteilung. Damit ist der Wert des Integrals Eins. Es folgt

$$d(L^*) \frac{\|v\|}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\frac{1}{2} z^2} dz = d(L^*) \|v\|$$

und die Behauptung ist bewiesen.

**Lemma 7.5** *Seien  $a, b \in \mathbb{Z}$  und  $p > 3$  eine Primzahl. Angenommen  $p$  teilt  $a$  nicht, dann teilt  $p$  eine der Zahlen  $(b + \frac{p-1}{2}a), \dots, b, \dots, (b - \frac{p-1}{2}a)$*

Beweis: Angenommen  $b \equiv 0 \pmod{p}$ , so ist nichts zu zeigen. Angenommen  $b \not\equiv 0 \pmod{p}$ . Dann sind  $a \pmod{p}$  und  $b \pmod{p}$  Elemente der multiplikativen Gruppe  $G = (\mathbb{Z}/p\mathbb{Z})^\times$ . Die Anzahl der Elemente von  $G$  ist  $p-1$ . Die Menge  $M = \{\pm 1, \pm 2, \dots, \pm \frac{p-1}{2}\}$  ist ein Repräsentantensystem von  $G$ .

Wegen  $[a] \in G$  existiert ein  $m \in G$  mit  $m \cdot a \equiv 1 \pmod{p}$ . Also ist  $(m \cdot b)a - b \equiv 0 \pmod{p}$ . In  $M$  muss somit ein Element  $z$  mit  $za - b$  kongruent  $0 \pmod{p}$  enthalten sein. Damit ist die Aussage bewiesen.

**Lemma 7.6** *Seien  $N = dh, d, h, m, S$  und  $a_i, i \in [m]$  so gewählt wie in Abschnitt 6.2 beschrieben und  $(a_1, \dots, a_m, i_0)$  der öffentliche Schlüssel. Sei*

$$\omega = \sum_{i \in S} a_i \pmod{N} = \sum_{i \in S} a_i + k_1 dh \quad \text{und}$$

$$\tilde{\omega} = \sum_{i \in S} a_i \pmod{d[h]} = \sum_{i \in S} a_i + k_2 d[h].$$

*Dann ist die Wahrscheinlichkeit, für das Ereignis  $k_1 \neq k_2$  höchstens  $2^{-\Omega(n)}$ .*

Beweis: Angenommen  $k_1 \neq k_2$ . Dann liegt die Summe  $\sum_{i \in S} a_i$  in einem Intervall zwischen zwei Vielfachen  $k \cdot dh$  und  $k \cdot d[h]$ .

Wegen  $\sum_{i \in S} \leq m \cdot dh$  ist die Länge eines solchen Intervalls höchstens  $|m(dh - d[h])|$ . Es gibt  $m$  viele solcher Intervalle. Nach Lemma 9.8 ist die Verteilung der zu den  $a_i$  gehörenden  $x_i$  exponentiell nah an der Gleichverteilung. Also ist die Wahrscheinlichkeit für  $k_1 \neq k_2$  höchstens

$$\frac{m^2 |dh - d[h]|}{N} \leq \frac{m}{16h} = \frac{c_m n^2}{16 \cdot 2^{c_n n^2}}$$

was zu zeigen war.

## Eigenschaften einer LLL-reduzierten Basis

Ziel dieses Abschnittes ist es Lemma 7.15 zu beweisen. Dieses Lemma wird benötigt, um die Konvergenz des Algorithmus, der das uSVP auf das dSVP<sub>p</sub> reduziert, sicherzustellen. Der LLL-Reduktionsbegriff geht auf A.K. Lenstra, H.W. Lenstra und L. Lovász zurück [Len 1982].

Um den LLL-Reduktionsalgorithmus formulieren zu können, benötigen wir einen Algorithmus, der ein gegebenes Gitter längenreduziert.

**Definition 7.7 (Längenreduktion)** Seien  $\mu_{i,j}$  die Gram-Schmidt-Koeffizienten (siehe Abschnitt 2.3). Eine Basis eines Gitters  $L \subseteq \mathbb{R}^d$  heißt längenreduziert, wenn  $|\mu_{i,j}| \leq \frac{1}{2}$  für  $1 \leq j < i \leq n$  erfüllt ist.

Eine gegebene Basis mit ihren Gram-Schmidt-Koeffizienten lässt sich durch den Algorithmus 7.8 in eine längenreduzierte Basis des gleichen Gitters transformieren.

**Algorithmus 7.8 (Längenreduktion)**

**Eingabe:** Basisvektoren  $b_1, b_2, \dots, b_n$  eines Gitters  $L$ .

```

1: for  $i = 1, 2, \dots, n$  do
2:   /*Längenreduziere  $b_i$ 
3:   for  $j = i - 1, i - 2, \dots, 1$  do
4:     Setze  $t = \lceil \mu_{i,j} \rceil$ 
5:     Setze  $b_i = b_i - t \cdot b_j$ 
6:     Setze  $\mu_{i,j} = \mu_{i,j} - t$ 
7:   end for
8: end for

```

**Ausgabe:** Eine längenreduzierte Basis von  $L$  und ihre Gram-Schmidt-Koeffizienten  $\mu_{i,j}$ .

Betrachten wir Schritt 5 des Algorithmus. Die Anweisung  $b_i^{neu} := b_i - t \cdot b_j$  bewirkt für den neuen Gram-Schmidt-Koeffizienten:

$$\mu_{i,j}^{neu} = \frac{\langle b_i^{neu}, b_j \rangle}{\|\hat{b}_j\|^2} = \frac{\langle b_i - t \cdot b_j, \hat{b}_j \rangle}{\|\hat{b}_j\|^2} = \frac{\langle b_i, \hat{b}_j \rangle}{\|\hat{b}_j\|^2} - t \cdot \frac{\langle b_j, \hat{b}_j \rangle}{\|\hat{b}_j\|^2} = \mu_{i,j} - t.$$

Wegen der Wahl von  $t$  ist  $|\mu_{i,j}^{neu}| \leq \frac{1}{2}$ .

**Lemma 7.9** Die Laufzeit von Algorithmus 7.8 ist  $O(n^2)$ .

**Definition 7.10** Sei  $\delta \in (\frac{1}{4}, 1]$ . Die Basis  $b_1, b_2, \dots, b_n$  eines Gitters  $L \subseteq \mathbb{R}^n$  heißt LLL-reduziert mit Parameter  $\delta$ , wenn

$$L_1 : |\mu_{i,j}| \leq \frac{1}{2} \quad 1 \leq j < i \leq n \quad \text{und}$$

$$L_2 : \delta \cdot \|\pi_i(b_i)\|^2 \leq \|\pi_{i+1}(b_{i+1})\|^2 + \mu_{i+1,i}^2 \|\pi_i(b_i)\|^2 \quad 1 \leq i < n$$

erfüllt sind.

Eine gegebene Basis lässt sich mit Hilfe von Algorithmus 7.11 in eine LLL-reduzierte Basis transformieren. Der Reduktionsalgorithmus wurde von L. Lovász entwickelt, dessen Verfahren auf einem Polynomialzeit-Algorithmus von Lenstra basiert.

**Algorithmus 7.11 (LLL-Reduktion)**

**Eingabe:** Basisvektoren  $b_1, b_2, \dots, b_n \in \mathbb{Z}^d$  eines Gitters  $L \subseteq \mathbb{Z}^d$ ;  
Reduktionsparameter  $\delta \in (\frac{1}{4}, 1]$

1: Setze  $k = 2$ .

2: Berechne die Gram-Schmidt-Koeffizienten  $\mu_{1,1}, \mu_{2,1}, \mu_{2,2}$

3: **while**  $k \leq n$  **do**

4: /Schleifeninvariante:  $B := b_1, b_2, \dots, b_{k-1}$  ist LLL-reduziert./

5: Längenreduziere die Basis  $b_1, \dots, b_k$  /Algorithmus!!!

6: **for**  $j = 1, 2, \dots, k-1$  **do**

7: Berechne die aktuellen Gram-Schmidt-Koeffizienten

8: **end for**

9: **if**  $(\delta \cdot \|\hat{b}_{k-1}\|^2 \leq \mu_{k,k-1}^2 \cdot \|\hat{b}_{k-1}\|^2 + \|\hat{b}_k\|^2)$

10: Setze  $k = k + 1$

11: **else**

12: Vertausche  $b_k$  und  $b_{k-1}$

13: Setze  $k = \max(k-1, 2)$

14: **end if**

15: **end while**

**Ausgabe:** Mit dem Parameter  $\delta$  LLL-reduzierte Basis  $b_1, b_2, \dots, b_n \in \mathbb{Z}^d$  des Gitters  $L$ .

**Satz 7.12 (Lenstra, Lenstra, Lovász)** Sei  $B$  eine Basis eines Gitters  $L \subseteq \mathbb{R}^d$ ,  $\delta \in (0, 1)$  und  $M := \max_i \|b_i\|^2$ . Man kann  $B$  in eine mit dem Parameter  $\delta$  reduzierte LLL-reduzierte Basis von  $L$  in  $O(dn^3 \log M)$  arithmetischen Schritten überführen.

Beweis: siehe [Len 1982].

**Lemma 7.13** Sei  $B = (b_1, b_2, \dots, b_n)$  eine mit dem Parameter  $\delta = \frac{3}{4}$  LLL-reduzierte Basis. Dann gelten die folgenden drei Aussagen:

- (i) Für  $i \in [n-1]$  gilt  $\|\hat{b}_i\| \leq 2\|\hat{b}_{i+1}\|$ .
- (ii) Für  $i < j$  gilt  $|\langle \hat{b}_i, b_j \rangle| \leq \frac{1}{2}\|\hat{b}_i\|^2$ .
- (iii)  $\min_i \|\hat{b}_i\| \leq \lambda_1(L)$ ,  $i \in [n]$ .

Beweis:

(i) Wegen der Eigenschaft  $L_2$  in Definition 7.10 gilt

$$\left(\frac{3}{4} - |\mu_{i+1,i}|^2\right)\|\hat{b}_i\|^2 \leq \|\hat{b}_{i+1}\|^2.$$

Durch Umstellen erhalten wir

$$\|\hat{b}_i\|^2 \leq \frac{1}{\frac{3}{4} - |\mu_{i+1,i}|^2} \|\hat{b}_{i+1}\|^2 \leq 4\|\hat{b}_{i+1}\|^2$$

und die Behauptung ist bewiesen.

(ii) folgt direkt aus der Definition der Gram-Schmidt-Koeffizienten.

(iii) Sei  $\tau(L) = \sum_{i=1}^n a_i b_i$ . Die Vektoren  $\hat{b}_1, \hat{b}_2, \dots, \hat{b}_n$  bilden eine Orthogonalbasis des  $\mathbb{R}^n$ . Der kürzeste Vektor zu dieser Basis hat die Darstellung

$$\tau(L) = \sum_{i=1}^n \left( \sum_{j=1}^n \mu_{j,i} a_j \right) \hat{b}_i.$$

Sei  $k \in [n]$  so gewählt, dass  $k$  die größte Zahl ist für die der Koeffizient  $a_k$  ungleich null ist. Da die Basisvektoren  $\hat{b}_i$  orthogonal zueinander sind und der Gram-Schmidt-Koeffizient für  $i = j$  Eins ist, erhalten wir

$$\begin{aligned} \lambda_1(L) &= \left\| \sum_{i=1}^n \sum_{j=1}^n \mu_{j,i} a_j \hat{b}_i \right\| \\ &= \left\| \sum_{i=1}^k \sum_{j=i}^k \mu_{j,i} a_j \hat{b}_i \right\| \\ &\geq \|\mu_{k,j} a_k \hat{b}_k\| \\ &\geq \min_i \|\hat{b}_i\| |a_k|. \end{aligned}$$

Hierbei haben wir ausgenutzt, dass der Vektor  $\hat{b}_k$  senkrecht auf dem von den Vektoren  $b_1, \dots, b_{k-1}$  aufgespannten Unterraum steht. Da der Betrag der  $a_k$  mindestens Eins ist, ist die Behauptung bewiesen.

**Lemma 7.14** Sei  $B = (b_{i,j})_{1 \leq i, j \leq n}$  eine obere Dreiecksmatrix mit der Eigenschaft

$$(\forall i < j \leq n) |b_{i,j}| \leq |b_{i,i}|.$$

Dann haben die Einträge der Matrix  $(B^\top)^{-1}$  höchstens den Betrag  $\frac{1}{\min_i |b_{i,i}|} 2^n$ .

Beweis: Die Matrix  $B^\top = (l_{i,j})_{i,j \leq n}$  ist eine untere Dreiecksmatrix mit der Eigenschaft, dass der Betrag der Einträge  $l_{i,j}$  höchstens so groß wie der Betrag der Diagonaleinträge  $l_{j,j}$  ist. Sie lässt sich durch

$$B^\top = M D$$

darstellen. Hierbei ist  $D$  eine Diagonalmatrix mit den Einträgen  $b_{i,i}$  auf der Diagonale und die Matrix  $M = m_{i,j}$  ist eine untere Dreiecksmatrix mit Einsen auf der Diagonale. Alle anderen Einträge von  $M$  sind vom Betrag höchstens Eins.

Es gilt

$$(B^\top)^{-1} = D^{-1} M^{-1}.$$

Die Matrix  $D^{-1}$  ist eine Diagonalmatrix mit den Einträgen  $\frac{1}{b_{i,i}}$ . Die Einträge der Matrix  $M^{-1} = a_{i,j}$  lassen sich rekursiv durch die Formeln

$$\begin{aligned} a_{j,j} &= 1 & 1 \leq j \leq n \\ a_{j,i} &= 0 & 1 \leq j < i \leq n \\ a_{i,j} &= \left(-\sum_{k=j}^{i-1} m_{i,k} \cdot a_{k,j}\right) & 1 \leq j < i \leq n \end{aligned}$$

berechnen. Zunächst werden die Einträge in der ersten Zeile, dann die in der zweiten usw. bestimmt. Nun beweisen wir mittels vollständiger Induktion über die Dimension der Matrix, dass  $|a_{i,j}| \leq 2^{i-j}$  für  $i \geq j$  gilt.

(IV) Handelt es sich um eine  $1 \times 1$ -Matrix, so ist die Aussage offensichtlich korrekt.

(IS)  $n \rightarrow n + 1$ :

Die Einträge auf der Diagonale von  $M^{-1}$  erfüllen offensichtlich die Bedingung. Nach Induktionsannahme ist die Aussage für  $i \leq n$  erfüllt. Es bleibt zu zeigen, dass die Aussage auch für  $i = n + 1$  stimmt.

Es gilt

$$\begin{aligned} |a_{n+1,j}| &= \left| \left(-\sum_{k=j}^n m_{n+1,k} \cdot a_{k,j}\right) \right| \\ &\leq \sum_{k=j}^n |(m_{n+1,k} \cdot 2^{k-j})| \\ &\leq \sum_{k=j}^n 2^{k-j}. \end{aligned}$$

Wir erhalten mit Satz 11.8

$$\begin{aligned} |a_{n+1,j}| &\leq 2^{-j} \sum_{k=0}^n 2^k \\ &= 2^{n+1-j}, \end{aligned}$$

was zu zeigen war.

Da die Beträge von  $D^{-1}$  durch  $\min_i |\frac{1}{b_{i,i}}|$  nach oben beschränkt sind, ist die Aussage des Lemma bewiesen.

**Lemma 7.15** Sei  $B := (b_1, b_2, \dots, b_n)$  eine mit dem Parameter  $\frac{3}{4}$  LLL-reduzierte Basis von  $L$  und  $\tau(L) = \sum_{i=1}^n a_i b_i$  sein kürzester Vektor. Dann gilt:

- (i)  $(\forall i \in [n]) |a_i| \leq 2^{2n}$
- (ii)  $\lambda_1(L) \leq \|b_1\| \leq 2^n \lambda_1(L)$
- (iii) Sei  $(b_1^*, \dots, b_n^*)$  die duale Basis. Dann gilt

$$(\forall i \in [n]) \|b_i^*\| \leq \frac{\sqrt{n}}{\lambda_1(L)} 2^n.$$

Beweis: Sei  $(\hat{b}_1, \dots, \hat{b}_n)$  die Gram-Schmidt-Orthogonalisierung von  $B$  (s. Abschnitt 2.3).

Die Gram-Schmidt-Orthogonalisierung der Basis  $B$  hat nach Lemma 7.13 folgende Eigenschaften:

- (1)  $\forall i \in [n] \|\hat{b}_i\| \leq 2\|\hat{b}_{i+1}\|.$
- (2) Für  $i < j$  gilt  $|\langle \hat{b}_i, b_j \rangle| \leq \frac{1}{2}\|\hat{b}_i\|^2.$
- (3)  $\min_{i \in [n]} \|\hat{b}_i\| \leq \lambda_1(L).$

Beweis von (ii):

Aus (1) folgt, dass  $\|\hat{b}_1\| \leq 2^{(n-1)}\|\hat{b}_n\|.$  Wegen  $\hat{b}_1 = b_1$  und (3) gilt

$$\lambda_1(L) \leq \|b_1\| \leq 2^n \lambda_1(L).$$

Beweis von (i):

Wir betrachten hierzu die Darstellung von  $B$  in der Orthonormalbasis

$$O = \left( \frac{\hat{b}_1}{\|\hat{b}_1\|}, \dots, \frac{\hat{b}_n}{\|\hat{b}_n\|} \right).$$

Der Vektor  $b_j$  zur Basis  $O$  ist die  $j$ -te Spalte der Matrix  $M = (m_{i,j})_{1 \leq i, j \leq n}$  mit  $m_{i,j} = \langle \hat{b}_i, b_j \rangle / \|\hat{b}_i\|.$  Da die Vektoren  $\hat{b}_j$  senkrecht auf dem von den Vektoren  $(b_1, \dots, b_{j-1})$  erzeugten Unterraum stehen, gilt  $m_{i,j} = 0$  für  $j < i.$  Dies hat zur Folge, dass die Matrix  $M$  eine obere Dreiecksmatrix ist. Die Einträge auf ihrer Diagonale sind  $m_{i,i} = \|\hat{b}_i\|.$  Man beachte das aus (2)  $|m_{i,j}| \leq \frac{1}{2}\|\hat{b}_i\|$  für  $i < j$  folgt.

Der kürzeste Vektor hat also die Darstellung

$$\sum_{i=1}^n a_i b_i = \sum_{i=1}^n \left( \sum_{j=i}^n a_j m_{i,j} \right) \hat{b}_i / \|\hat{b}_i\|.$$

Wegen (1) und da  $\|\hat{b}_1\| \geq \lambda_1(L)$  lässt sich

$$\lambda_1(L) \leq \|\hat{b}_1\| \leq 2^n \|\hat{b}_i\|$$

schließen. Da die Vektoren  $\hat{b}_i$  senkrecht aufeinander stehen, kann der Betrag der Koeffizienten  $(\sum_{j=1}^n a_j m_{i,j})$  höchstens den Wert  $2^n \|\hat{b}_i\|$  annehmen.

Die Behauptung (i) zeigen wir per Induktion. Betrachten wir den Fall  $i = n$ , so wissen wir, dass  $|a_n m_{n,n}| \leq 2^n \|\hat{b}_n\|$ . Also ist der Betrag des Koeffizienten  $a_n$  höchstens  $2^n$ . Wir wollen jetzt induktiv  $|a_k| \leq 2^{2n-k}$  schließen.

(IS) Angenommen die Aussage stimmt für die Werte  $a_{k+1}, \dots, a_n$ . Es gilt

$$\begin{aligned} \left| \sum_{j=k+1}^n a_j m_{k,j} \right| &\stackrel{(*)}{\leq} \frac{1}{2} \sum_{j=k+1}^n |a_j| \|\hat{b}_k\| \\ &\leq \frac{1}{2} \left( \sum_{j=k+1}^n 2^{2n-j} \right) \|\hat{b}_k\| \\ &\leq \frac{1}{2} \cdot 2^{2n-k} \|\hat{b}_k\|. \end{aligned}$$

Die Ungleichung (\*) gilt wegen (2).

Also folgt mit der Dreiecksungleichung und  $|\sum_{j=i}^n a_j m_{i,j}| \leq 2^n \|\hat{b}_i\|$

$$\begin{aligned} |a_k m_{k,k}| &\leq \left| \sum_{j=k+1}^n a_j m_{k,j} \right| + \left| \sum_{j=k}^n a_j m_{k,j} \right| \\ &\leq \left( \frac{1}{2} \cdot 2^{2n-k} + 2^n \right) \|\hat{b}_k\| \\ &\leq 2^{2n-k} \|\hat{b}_k\|. \end{aligned}$$

Damit ist nur noch (iii) zu beweisen. Die Basis des dualen Gitters ist nach Satz 2.26 durch die Spalten der Matrix  $(M^\top)^{-1}$  gegeben. Wegen  $|m_{i,j}| \leq |m_{i,i}|$  lässt sich Lemma 7.14 anwenden. Außerdem gilt

$$\min_i |m_{i,i}| = \min_i \|\hat{b}_i\| \geq \frac{\lambda_1(L)}{2^n}.$$

Die Beträge der Einträge von  $(M^\top)^{-1}$  sind nach Lemma 7.14 durch  $\frac{1}{\lambda_1(L)} 2^{2n}$  beschränkt. Also ist die Länge der Spalten durch  $\frac{\sqrt{n}}{\lambda_1(L)} 2^{2n}$  beschränkt und die Aussage ist bewiesen.

## 8 Hauptteil

Um die Sicherheit des Verfahrens zu beweisen, muss gezeigt werden, dass es genauso schwierig ist, in einem  $f(n)$ -eindeutigen Gitter das SVP zu lösen, wie zwischen der Gleichverteilung  $U$  und der Menge von Verteilungen  $\Upsilon_{g(n)}$  zu unterscheiden. Zur Erinnerung sei nochmal erwähnt, dass Unterscheiden zwischen einer Verteilung  $U$  und einer Menge von Verteilungen bedeutet, dass man mit nicht zu vernachlässigender Wahrscheinlichkeit (siehe Definition 5.9 und 5.11) zwischen jeder Verteilung der Menge und  $U$  unterscheiden kann. Dies geschieht in mehreren Schritten. Zuerst zeigen wir mittels einer Karpreduktion (Definition 5.10), dass das  $uSVP$  (Definition 2.33) auf das  $dSVP_p$  (Definition 2.34) reduziert werden kann. Danach werden wir das  $dSVP_p$  auf das Problem der Unterscheidbarkeit zwischen der Gleichverteilung und der Menge von Verteilungen  $\Upsilon_{g(n)}$  reduzieren.

In diesem Kapitel setzen wir oft voraus, dass die Größe eines Parameters (z.B. eine Primzahl) polynomiell von der Dimension des Gitters abhängt. Ist ein konkretes Gitter gegeben, so ist diese Forderung trivial. Unsere Herangehensweise ist aber anders. Wir betrachten Probleme in Abhängigkeit von der Dimension  $n$  des Gitters und wollen sicherstellen, dass wir in Abhängigkeit von  $n$  einen Polynomialzeitalgorithmus finden, der Probleme auf andere Probleme reduziert. Um polynomielle Laufzeit zu garantieren, ist es daher oft erforderlich, dass Parameter polynomiell von der Größe des Gitters abhängen. Alle Gitter die wir in diesem Kapitel betrachten sind volldimensional.

Den Abstand zweier Verteilungen exponentiell klein nennen, wenn er in der Größenordnung  $2^{-n}$  ist.

### 8.1 Reduktion auf das dSV Problem

Durch Veränderung der Reihenfolge der Basisvektoren kann man ein Orakel, dass das  $dSVP_p$  löst, nutzen, um für ein beliebiges  $i \in [n]$  herauszufinden ob  $p$  den Koeffizienten  $a_i$  des kürzesten Vektors teilt oder nicht. Diese Tatsache werden wir im Folgenden ausnutzen, ohne es nochmals zu erwähnen.

**Lemma 8.1** *Sei  $p(n) > 2$  eine Primzahl deren Größe höchstens polynomiell von  $n$  abhängt und  $L$  ein eindeutiges Gitter. Es existiert ein Algorithmus der das  $uSVP$  auf das  $dSVP_p$  reduziert.*

Die Beweisidee ist knapp gesagt die Folgende: Das Ausgangsgitter wird immer gröber gemacht, ohne dabei den kürzesten Vektor zu verändern. Am Ende des Beweises ist das Gitter so grob, dass der kürzeste Vektor abgelesen werden kann. Teilt  $p$  nämlich den  $i$ -ten Koeffizienten des kürzesten Vektor, so können wir den  $i$ -ten Basisvektor mit  $p$  multiplizieren. Das resultierende Gitter hat dann immer noch den gleichen kürzesten Vektor wie das Ausgangsgitter. Diese Tatsache werden wir im Folgenden ausnutzen.

In dem Beweis werden wir zunächst einen Algorithmus  $\mathcal{C}$  mit polynomieller Laufzeit konstruieren, den wir zur Reduktion des  $uSVP$  auf das  $dSVP_p$  benötigen.

Sei  $B = (b_1, b_2, \dots, b_n)$  eine LLL-reduzierte Basis von  $L$  und

$$\tau(L) = \sum_{i=1}^n a_i b_i$$

der kürzeste Vektor. Der Algorithmus  $\mathcal{C}(i, j), i, j \in [n]$  benutzt ein Orakel  $\mathcal{O}(B)$ , das ausgibt, ob der Koeffizienten  $a_i$  von  $p$  geteilt wird. Dies ist gleichbedeutend damit, dass es dSVP $_p$  löst.

Die Routine  $\mathcal{C}(i, j)$  besteht aus folgenden zwei Schritten:

1) Teilt  $p$  den Koeffizienten  $a_i$  verändere den  $i$ -ten Basisvektor  $b_i = p \cdot b_i$ . Das neue Gitter  $\tilde{L}$  ist ein Untergitter von  $L$ , dass den gleichen kürzesten Vektor wie  $L$  hat. Dadurch wird der  $i$ -te Koeffizient um den Faktor  $p$  verkleinert. Dieser Schritt wird wiederholt bis  $p$  den Koeffizienten  $a_i$  nicht mehr teilt oder die Anzahl der Wiederholungen  $2n$  erreicht hat.

Ist Letzteres der Fall und  $a_i \neq 0$  muss  $|a_i|$  größer als  $2^{2n}$  gewesen sein, was im Widerspruch zu Lemma 7.15 steht. Also muss  $a_i$  schon Null gewesen sein. Der Algorithmus gibt dann die unveränderte Basis und das Bit 0 zurück. Ansonsten wird Schritt zwei durchgeführt.

2) Wird Schritt 2 durchgeführt wissen wir, dass wir ein Gitter  $\tilde{L}$  gefunden haben, dass den gleichen kürzesten Vektor  $\tau(L)$  wie das Ausgangsgitter hat und in dem  $p$  den  $i$ -ten Koeffizienten nicht mehr teilt.

Sei  $\tilde{L} = L(b_1, b_2, \dots, b_n)$  das neue Gitter und  $(a_1, a_2, \dots, \tilde{a}_i, \dots, a_n)$  die Koeffizienten des kürzesten Vektors (Der Koeffizient  $a_i$  des kürzesten Vektors hat sich durch Schritt 1 eventuell verändert, der Vektor nicht). Nun betrachten wir die  $p$  verschiedenen Basen von  $\tilde{L}$ , in denen der Vektor  $b_i$  durch  $b_i - \frac{p-1}{2}b_j, \dots, b_i - b_j, b_i, b_i + b_j, \dots, b_i + \frac{p-1}{2}b_j$  ersetzt wird. Dass dies Basen von  $\tilde{L}$  sind folgt aus Satz 2.17. Nun untersuchen wir die Darstellung des kürzesten Vektors zu den verschiedenen Basen.

Offensichtlich ist  $a_j$  der einzige Koeffizient, der durch die Transformationen verändert wird. Dieser nimmt die Werte

$$a_j = a_j + \frac{p-1}{2}\tilde{a}_i, \dots, a_j = a_j + \tilde{a}_i, a_j, a_j = a_j - \tilde{a}_i, \dots, a_j = a_j - \frac{p+1}{2}\tilde{a}_i \quad (5)$$

an. Nach Lemma 7.5 existiert ein  $a_j$ , das von  $p$  geteilt wird. Mit Hilfe des Orakels finden wir die Basis, für die  $p \mid a_j$  gilt. Wir nehmen diese Basis und ersetzen den  $j$ -ten Basisvektor  $b_j$  durch  $p \cdot b_j$ . Diese Operation erhält den kürzesten Vektor des Ausgangsgitters. Der Vorgang wird  $2n$ -mal wiederholt. Die Rückgabe ist die letzte Basis und das Bit 1.

**Lemma 8.2** *Der Algorithmus  $\mathcal{C}(i, j)$  gibt eine Basis  $B$  und ein Bit zurück. Das Gitter  $L(B)$  und  $\mathcal{C}$  haben folgende Eigenschaften:*

- (i) *Ist dieses Bit 0, so ist der  $i$ -te Koeffizient  $a_i$  des kürzesten Vektors 0.*
- (ii) *Ist das Bit 1, so gilt  $a_i \neq 0$ ,  $|a_j| \leq \frac{1}{2}|a_i|$  und  $p \nmid a_i$ .*
- (iii) *Der kürzeste Vektor des übergebenen Gitters ist gleich dem kürzesten Vektor von  $L(B)$ .*
- (iv)  *$\mathcal{C}$  hat polynomielle Laufzeit in Abhängigkeit von der Dimension des Gitters.*

Beweis: Die Aussagen (i) und (iii) sind schon in der Beschreibung von  $\mathcal{C}$  bewiesen worden. Es bleibt also nur noch (ii) und (iv) zu zeigen.

(ii): Ist das Bit 1, so wurde Schritt 2 des Algorithmus durchgeführt. Wie in der Beschreibung des Algorithmus erläutert haben wir am Beginn von Schritt 2 ein Gitter, das den gleichen kürzesten Vektor wie das Ausgangsgitter hat und in dem  $p$  den  $i$ -ten Koeffizienten des kürzesten Vektors nicht teilt. In einem Teilschritt von Schritt 2 wird nur der  $i$ -te Basisvektor durch  $v_i + k \cdot v_j$  ersetzt. Der  $i$ -te Koeffizient  $a_i$  des kürzesten Vektors bleibt durch diese Transformationen unverändert. Also wird er auch am Ende von Schritt 2 nicht von  $p$  geteilt.

Nach der ersten Veränderung der Basis in Schritt 2 ist der Betrag  $|a_j^{(1)}|$  höchstens  $|a_j^{(0)}| + \frac{p-1}{2}|a_i|$  groß. Danach wird der Basisvektor  $v_j$  mit  $p$  multipliziert. Dies verändert  $a_j$  nochmal und am Ende eines Durchlaufs gilt

$$|a_j^{(1)}| \leq \left( |a_j^{(0)}| + \frac{p-1}{2}|a_i| \right) / p = \left( \frac{1}{2} - \frac{1}{2p} \right) |a_i| + \frac{|a_j^{(0)}|}{p}.$$

Nach dem nächsten Schritt ist  $|a_j^{(2)}|$  dann noch höchstens

$$\left( \frac{1}{2} - \frac{1}{2p} \right) |a_i| + \frac{1}{p} \left( \frac{1}{2} - \frac{1}{2p} \right) |a_i| + \frac{|a_j^{(0)}|}{p^2}$$

groß. Nach  $2n$  Wiederholungen gilt

$$\begin{aligned} |a_j^{(2n)}| &\leq \left( \frac{1}{2} - \frac{1}{2p} \right) \left( 1 + \frac{1}{p} + \dots + \frac{1}{p^{2n-1}} \right) |a_i| + \frac{|a_j^{(0)}|}{p^{2n}} = \\ &\left( \frac{1}{2} - \frac{1}{2p} + \frac{1}{2p} - \frac{1}{2p^2} + \dots - \frac{1}{p^{2n-1}} + \frac{1}{p^{2n-1}} - \frac{1}{p^{2n}} \right) |a_i| + \frac{|a_j^{(0)}|}{p^{2n}} < \\ &\frac{1}{2} |a_i| + \frac{|a_j^{(0)}|}{p^{2n}}. \end{aligned}$$

Da die Basis von  $L$  LLL-reduziert ist, muss  $|a_j|$  nach Lemma 7.14 kleiner als  $2^{2n}$  gewesen sein. Also ist der Bruch  $\frac{|a_j|}{p^{2n}}$  eine Zahl kleiner 1. Da die Koeffizienten eines Gittervektors ganze Zahlen sein müssen gilt die Ungleichung  $|a_j| \leq \frac{1}{2}|a_i|$ . (iv) Da die Primzahldarstellung polynomiell von der Dimension des Gitters abhängt, müssen in Schritt 2 nur polynomiell viele Basen überprüft werden. Also hat  $\mathcal{C}$  polynomielle Laufzeit.

Damit ist Lemma 8.2 bewiesen.

Beweis von Lemma 8.1: Damit wir eine Schranke für die Koeffizienten des kürzesten Vektors haben nehmen wir ohne Beschränkung der Allgemeinheit an, dass die Basis in der das Gitter gegeben ist LLL-reduziert ist. Dies können wir tun, da der LLL-Algorithmus polynomielle Laufzeit hat (siehe Satz 7.12). Nach Lemma 7.15 sind die Beträge der Koeffizienten des kürzesten Vektors höchstens  $2^{2n}$  und es gilt  $\frac{\|b_1\|}{2^n} \leq \lambda_1(L) \leq \|b_1\|$ .

Nun werden wir einen Algorithmus  $\mathcal{B}(\alpha)$  konstruieren, der für  $\lambda_1(L) < \alpha \leq$

$2\lambda_1(L)$  den kürzesten Vektor mit Hilfe der Lösung des dSVP<sub>p</sub> findet.  $\mathcal{B}(\alpha)$  wird mit  $n + 1$  verschiedenen Werten

$$\alpha_j = 2^{j-n} \cdot \|b_1\| \quad , j = 0 \dots, n + 1$$

aufgerufen. Wegen  $\alpha_0 = \frac{\|b_1\|}{2^n} \leq \lambda_1(L) \leq \|b_1\| < 2\|b_1\| = \alpha_{n+1}$  muss ein  $\alpha_j$  existieren, für das die Bedingung

$$\alpha_{j-1} \leq \lambda_1(L) < \alpha_j = 2\alpha_{j-1}$$

erfüllt ist.

Aus  $\alpha_{j-1} \leq \lambda_1(L)$  folgt, dass  $2\alpha_{j-1} = \alpha_j$  höchstens  $2\lambda_1(L)$  ist. Damit ist sichergestellt, dass es einen Aufruf  $\mathcal{B}(\alpha_j)$  gibt, für den die Ungleichung

$$\lambda_1(L) < \alpha_j \leq 2\lambda_1(L)$$

gilt. Es kann sein, dass der Algorithmus für Werte  $\alpha_j$ , die nicht die Bedingung  $\lambda_1(L) < \alpha_j \leq 2\lambda_1(L)$  erfüllen, keine Gitterpunkte liefert. Am Ende des Verfahrens muss daher überprüft werden, welcher der zurückgegebenen Vektoren der kürzeste in  $L$  liegende Vektor ist.

Im Folgenden werden wir das Orakel mit Untergittern von  $L$  aufrufen. Diese Untergitter haben den gleichen kürzesten Vektor wie  $L$ . Da  $L$  ein  $f(n)$ -eindeutiges Gitter ist, sind auch alle Untergitter von  $L$ , die den gleichen kürzesten Vektor wie  $L$  haben, ebenfalls  $f(n)$ -eindeutig. Das Orakel wird also immer mit einem  $f(n)$ -eindeutigen Gitter befragt.

Nun zur Beschreibung von  $\mathcal{B}$ :

Der Algorithmus startet mit einer Menge  $M = [n]$  von Indizes. Wenn wir wissen, dass der  $i$ -te Koeffizient des kürzesten Vektor 0 ist, entfernen wir den Index  $i$  aus der Menge  $M$ . Ziel ist es,  $M$  zu einer einelementigen Menge zu machen. So lange  $|M| \geq 2$  ist, werden folgende Operationen durchgeführt.

O.B.d.A.  $1, 2 \in M$ . Es wird abwechselnd  $\mathcal{C}(1, 2)$  und  $\mathcal{C}(2, 1)$  aufgerufen, bis das Rückgabebit 0 ist. Nach Lemma 8.2 bedeutet dies, dass entweder der Koeffizient  $a_1$  oder  $a_2$  (hängt davon ab ob  $\mathcal{C}(1, 2)$  oder  $\mathcal{C}(2, 1)$  0 zurückgibt) des kürzesten Vektors 0 ist. Den zugehörigen Index entfernen wir aus der Menge  $M$ . Nach Lemma 8.2 hat das neue Untergitters den gleichen kürzesten Vektor wie das Ausgangsgitter.

Jetzt müssen wir noch überprüfen, ob  $\mathcal{B}$  terminiert. Das Verfahren terminiert, wenn sichergestellt ist, dass  $\mathcal{C}$  nach endlich vielen Aufrufen Null zurückgibt. Nach Lemma 8.2 ist, unter der Voraussetzung, dass das Rückgabebit 1 ist, nach dem Aufruf  $\mathcal{C}(1, 2)$  die Ungleichung  $|a_2| \leq \frac{1}{2}|a_1|$  erfüllt. Danach wird  $\mathcal{C}(2, 1)$  aufgerufen, was zur Folge hat, dass das neue  $a_1$  folgende Ungleichung erfüllt:  $|a_1| \leq \frac{1}{2}|a_2| \leq \frac{1}{4}|a_{alt}|$ . Also wird der Koeffizient  $a_1$  mindestens um den Faktor 4 verkleinert. Da die Koeffizienten ganzzahlig sind und ihr Betrag nach Lemma 7.15 nicht größer als  $2^{2n}$  ist, muss  $a_1$  oder  $a_2$  spätestens nach  $2n$  Schritten 0 sein. Dies hat zur Folge, dass  $\mathcal{C}$  im nächsten Schritt 0 zurückgibt.  $\mathcal{C}$  und damit auch  $\mathcal{B}$  sind Algorithmen mit polynomieller Laufzeit.

**Beispiel 8.3** Um ein Gefühl für die Wirkungsweise von  $\mathcal{B}$  zu bekommen, ist es ganz gut sich ein Beispiel anzuschauen. Hier ist der Vektor  $v$  zwar nicht der kürzeste Vektor des Gitter. Der  $dSVP_p$ -Löser verhält sich aber so, was zwar nicht korrekt für die Wirkungsweise von  $\mathcal{B}$  aber unerheblich ist. Die Anwendung von  $\mathcal{B}$  macht  $v$  zu einem Basisvektor eines Untergitters des Ausgangsgitters  $L$ . Sei  $L$  das Gitter das von der Basis  $B := ((4, 1, 0)^\top, (0, 0, 3)^\top, (-1, -1, -1)^\top)$  erzeugt wird und  $v = (12, 3, -18)_\epsilon = (3, -6, 0)_B$ .  $dSVP_3$  gibt Eins zurück, falls 3 den Eintrag  $a_i$  von  $v_B$  teilt und Null sonst.

Am Anfang ist  $M = \{1, 2, 3\}$ .

Es wird  $\mathcal{C}(1, 2)$  aufgerufen. 3 teilt den ersten Koeffizienten von  $v_B$ . Also wird der 1. Basisvektor mit 3 multipliziert. Das neue Gitter  $\tilde{B}$  wird durch die Matrix

$$\begin{pmatrix} 12 & 0 & -1 \\ 3 & 0 & -1 \\ 0 & 3 & 1 \end{pmatrix}$$

repräsentiert. Der neue erste Koeffizient  $a_1$  von  $v_{\tilde{B}}$  ist 1 und  $\mathcal{C}$  führt Schritt 2 durch.  $v_1$  wird durch  $v_1 - v_2$  ersetzt. Dann ist  $\tilde{a}_2 = a_2 + a_1 = 5$ .  $p$  teilt  $\tilde{a}_2$  in diesem Fall nicht. Also muss die nächste Basis betrachtet werden. Hier bleibt  $b_1$  und damit auch  $a_2 = -6$  unverändert. 3 teilt -6 und der zweite Basisvektor  $b_2$  wird durch  $2 \cdot b_2 = (9, 0, -3)$  ersetzt. Die Basis des neuen Gitters ist:

$$(b_1, p \cdot b_2, b_3) = \begin{pmatrix} 12 & 0 & -1 \\ 3 & 0 & -1 \\ 0 & 9 & 1 \end{pmatrix}$$

Nun ist  $a_2 = -2$ ,  $p \mid a_2 - a_1$  und die Basis wird wieder verändert.

$$(b_1 + b_2, p \cdot b_2, b_3) = \begin{pmatrix} 12 & 0 & -1 \\ 3 & 0 & -1 \\ 9 & 27 & 1 \end{pmatrix}, a_2 = -1$$

$p \mid a_2 + a_1$ , also wird die Basis folgenderweise verändert:

$$(b_1 - b_2, p \cdot b_2, b_3) = \begin{pmatrix} 12 & 0 & 1 \\ 3 & 0 & -1 \\ -18 & 81 & 1 \end{pmatrix}, a_2 = 0$$

Da  $a_3$  schon null ist, verändern die restlichen Operationen den ersten Basisvektor nicht mehr und wir haben bereits nach dem ersten Aufruf von  $\mathcal{C}$  ein Untergitter gefunden, dass  $v$  als ersten Basisvektor hat.

## 8.2 Verteilungen auf Gittern

In diesem Unterkapitel werden wir Ergebnisse aus der harmonischen Analysis verwenden, um das  $dSVP$  auf die Ununterscheidbarkeit zweier Verteilungen zu reduzieren. Sei  $B_n$  die  $n$ -dimensionale Einheitskugel,  $A \subset \mathbb{R}^n$  eine abzählbare Menge, und  $\rho : A \rightarrow \mathbb{R}$ ,  $\rho(A) := \sum_{x \in A} e^{-\pi \|x\|^2}$ . Als erstes wollen wir die Ergebnisse vorstellen, die wir verwenden werden.

**Lemma 8.4** Für jedes Gitter  $L$  gilt

$$\rho(L \setminus \sqrt{n}B_n) < 2^{-\Omega(n)} \rho(L).$$

Beweis: [Ban 1993] Lemma 1.5(i) mit  $c=1$ .

**Korollar 8.5** Sei  $L$  ein Gitter. Dann gilt

$$\rho(L) < \frac{\rho(L \cap \sqrt{n}B_n)}{(1 - 2^{-\Omega(n)})}.$$

Beweis:

$$\begin{aligned} \rho(L) &= \rho((L \setminus \sqrt{n}B_n) \cup (L \cap \sqrt{n}B_n)) \\ &= \rho(L \setminus \sqrt{n}B_n) + \rho(L \cap \sqrt{n}B_n) \\ &< 2^{-\Omega(n)} \rho(L) + \rho(L \cap \sqrt{n}B_n) \end{aligned}$$

Also gilt die Ungleichung

$$\rho(L) - 2^{-\Omega(n)} \rho(L) < \rho(L \cap \sqrt{n}B_n).$$

Wenn man beide Seiten durch  $1 - 2^{-\Omega(n)}$  dividiert, hat man die Behauptung.

**Lemma 8.6** Für jedes Gitter  $L$  und jeden Vektor  $z \in \mathbb{R}^n$  gilt

$$\rho(L^* + z) = d(L) \sum_{x \in L} e^{2\pi i \langle x, z \rangle} \rho(\{x\}).$$

Beweis: [Ban 1993] Lemma 1.1(i) mit  $a = \pi$ ,  $b = 1$ ,  $y = 0$ .

**Definition 8.7** Sei  $L$  ein Gitter. Die Funktion  $D_{L^*}$  sei definiert durch

$$\begin{aligned} D_{L^*} &: \mathcal{P}(L^*) \rightarrow \mathbb{R} \\ D_{L^*}(x) &= \rho(L^* + x). \end{aligned}$$

**Definition 8.8** Sei  $L$  ein Gitter und  $v \in L \setminus \{0\}$ . Die Funktion  $T_{L^*,v}$  sei definiert durch

$$\begin{aligned} T_{L^*,v} &: \mathcal{P}(L^*) \rightarrow \mathbb{R} \\ T_{L^*,v}(x) &= \frac{d(L)}{\|v\|} \sum_{k \in \mathbb{Z}} e^{-\pi \left( \frac{k + \langle v, x \rangle}{\|v\|} \right)^2}. \end{aligned}$$

**Bemerkung 8.9** Die Funktionen  $D_{L^*}$  und  $T_{L^*,v}$  lassen sich auf ganz  $\mathbb{R}^n$  fortsetzen. Das Skalarprodukt eines Elementes aus  $L^*$  und eines aus  $L$  ist nach Definition eine ganze Zahl. Für jeden Vektor  $w \in L^*$  gilt  $D_{L^*}(x+w) = D_{L^*}(x)$ . Gleiches gilt für  $T_{L^*,v}$ . Also ist die Fortsetzung beider Funktionen periodisch zu der Grundmasche des dualen Gitters  $\mathcal{P}(L^*)$ .

**Definition 8.10** Sei  $U_{L^*}$  die Dichtefunktion der Gleichverteilung auf  $\mathcal{P}(L^*)$ , d.h. die Funktion für die  $U_{L^*}(x) = d(L^*) = \frac{1}{d(L)}$  (siehe Satz 2.28).

**Lemma 8.11**  $T_{L^*,v}$  und  $D_{L^*}$  sind Dichtefunktionen auf  $\mathcal{P}(L^*)$ .

Beweis: Zu zeigen ist, dass das Integral über die beiden Funktionen 1 ist.

$$\begin{aligned} \int_{\mathcal{P}(L^*)} D_{L^*}(x) dx &= \int_{\mathcal{P}(L^*)} \sum_{l \in L^*} e^{-\pi \|x+l\|^2} dx \\ &= \int_{\mathcal{P}(L^*)} \lim_{n \rightarrow \infty} \sum_{l \in L^* \cap \sqrt{n}B_n} e^{-\pi \|x+l\|^2} dx \end{aligned}$$

Nach dem Satz von Beppo Levi (Theorem 3.12) kann man Integration und Grenzwertbildung vertauschen. Also gilt

$$\begin{aligned} \int_{\mathcal{P}(L^*)} \lim_{n \rightarrow \infty} \sum_{l \in L^* \cap \sqrt{n}B_n} e^{-\pi \|x+l\|^2} dx &= \lim_{n \rightarrow \infty} \int_{\mathcal{P}(L^*)} \sum_{l \in L^* \cap \sqrt{n}B_n} e^{-\pi \|x+l\|^2} dx \\ &= \lim_{n \rightarrow \infty} \sum_{l \in L^* \cap \sqrt{n}B_n} \int_{\mathcal{P}(L^*)} e^{-\pi \|x+l\|^2} dx. \end{aligned}$$

Da für verschiedene Gitterpunkte  $x, y$  aus  $L^*$  der Schnitt der verschobenen Grundmaschen  $\mathcal{P}(L^*) + x$  und  $\mathcal{P}(L^*) + y$  leer ist und wegen  $\bigcup_{x \in L^*} \mathcal{P}(L^*) + x = \mathbb{R}^n$  gilt die folgende Gleichheit

$$\begin{aligned} \lim_{n \rightarrow \infty} \sum_{l \in L^* \cap \sqrt{n}B_n} \int_{\mathcal{P}(L^*)} e^{-\pi \|x+l\|^2} dx &= \sum_{l \in L^*} \int_{\mathcal{P}(L^*)} e^{-\pi \|x+l\|^2} dx \\ &= \int_{\mathbb{R}^n} e^{-\pi \|x\|^2} dx. \end{aligned}$$

$f(x) := e^{-\pi \|x\|^2}$  ist eine  $n$ -dimensionale Gaußsche Normalverteilung mit Eigenwert 0 und Standardabweichung  $\frac{1}{\sqrt{2\pi}}$ . Also nimmt ihr Integral den Wert 1 an, was zu zeigen war.

Nach Lemma 7.4 gilt  $\int_{\mathcal{P}(L^*)} \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k + \langle v, x \rangle}{\|v\|}\right)^2} dx = \|v\| d(L^*) = \frac{\|v\|}{d(L)}$

und es folgt

$$\int_{\mathcal{P}(L^*)} T_{L^*,v}(x) dx = \int_{\mathcal{P}(L^*)} \frac{d(L)}{\|v\|} \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k + \langle v, x \rangle}{\|v\|}\right)^2} dx = 1.$$

Die Funktionen  $T_{L^*}$ ,  $U_{L^*}$ ,  $D_{L^*}$  sind somit Dichtefunktionen auf der Grundmasche des dualen Gitters.

Wir werden ausnutzen, dass die Verteilung  $D_{L^*}$  für bestimmte Gitter nicht von  $T_{L^*}$  und für andere nicht von  $U_{L^*}$  unterscheidbar ist. Dies beweisen wir in den folgenden zwei Lemmata.

Nun werden wir den statistische Abstand verschiedener Verteilungen bestimmen. Dieser ist definiert durch

$$\Delta(X, Y) = \frac{1}{2} \int_{\mathbb{R}} |f_1(x) - f_2(x)| dx$$

wobei  $f_1$  und  $f_2$  die Dichtefunktionen sind (siehe Definition 4.17). Da wir zeigen das der Abstand im Bereich  $2^{-\Omega(n)}$  (siehe Definition 1.10) ist, können wir die Konstante  $\frac{1}{2}$  auch weglassen.

**Lemma 8.12** *Sei  $L$  ein Gitter, in dem alle Vektoren außer dem Nullvektor länger als  $\sqrt{n}$  sind. Dann gilt für den Abstand  $\Delta(D_{L^*}, U_{L^*})$  zwischen  $D_{L^*}$  und  $U_{L^*}$*

$$\Delta(D_{L^*}, U_{L^*}) < 2^{-\Omega(n)}.$$

Beweis: Sei  $y \in \mathbb{R}^n$ .

Da alle Vektoren ungleich 0 größer als  $\sqrt{n}$  sind, ist der Nullpunkt der einzige Gitterpunkt der in  $\sqrt{n}B_n$  liegt. Daraus folgt

$$\left| 1 - \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) \right| < \sum_{x \in L \setminus \sqrt{n}B_n} \rho(\{x\}) = \rho(L \setminus \sqrt{n}B_n).$$

Nun benutzen wir die Abschätzungen aus Lemma 8.4 und aus Korollar 8.5 und erhalten

$$\rho(L \setminus \sqrt{n}B_n) < 2^{-\Omega(n)} \rho(L) < 2^{-\Omega(n)} \frac{\rho(L \cap \sqrt{n}B_n)}{1 - 2^{-\Omega(n)}}.$$

Da die Menge  $\sqrt{n}B_n$  nur einen Gitterpunkt nämlich den Nullpunkt enthält, ist  $\rho(L \cap \sqrt{n}B_n) = \rho(0) = e^0 = 1$ . Also gilt die Abschätzung

$$\left| 1 - \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) \right| < 2^{-\Omega(n)}.$$

Durch Multiplikation mit  $d(L)$  erhält man

$$\left| d(L) - d(L) \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) \right| < d(L) 2^{-\Omega(n)}.$$

Diese Ungleichung verwenden wir jetzt, um  $\Delta(D_{L^*}, U_{L^*})$  abzuschätzen. Man beachte, dass nach Lemma 8.6  $\rho(L^* + y) = d(L) \cdot \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\})$  und nach

Satz 2.28  $d(L) = \frac{1}{d(L^*)}$  gilt. Man erhält

$$\begin{aligned} \Delta(D_{L^*}, U_{L^*}) &= \int_{\mathcal{P}(L^*)} |D_{L^*}(x) - U_{L^*}(x)| dx \\ &= \int_{\mathcal{P}(L^*)} \left| \rho(L^* + x) - \frac{1}{d(L^*)} \right| dx \\ &= \int_{\mathcal{P}(L^*)} \left| d(L) - d(L) \cdot \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) \right| dx \\ &< \int_{\mathcal{P}(L^*)} d(L) 2^{-\Omega(n)} dx = 2^{-\Omega(n)}. \end{aligned}$$

**Lemma 8.13** Sei  $L$  ein Gitter mit kürzestem Vektor  $u$ , in dem alle nicht zu  $u$  parallelen Vektoren länger als  $\sqrt{n}$  sind. Dann ist

$$\Delta(D_{L^*}, T_{L^*, u}) < 2^{-\Omega(n)} \left(1 + \frac{1}{\|u\|}\right).$$

Gilt zusätzlich  $\|u\| \geq n^{-c}$  für ein  $c > 0$ , so folgt

$$\Delta(D_{L^*}, T_{L^*, u}) < 2^{-\Omega(n)}.$$

Beweis: Sei  $y \in \mathbb{R}^n$ . Als erstes brauchen wir eine Abschätzung für folgenden Abstand:

$$\left| \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) - \sum_{k \in \mathbb{Z}} e^{2\pi i k \langle u, y \rangle} \rho(\{ku\}) \right|.$$

Da die Vielfachen von  $u$  nach Voraussetzung die einzigen Vektoren sind, die eventuell nicht in  $\sqrt{n}B_n$  liegen, gilt

$$\begin{aligned} \left| \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) - \sum_{k \in \mathbb{Z}} e^{2\pi i k \langle u, y \rangle} \rho(\{ku\}) \right| &\leq \left| \sum_{x \in L \setminus \sqrt{n}B_n} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) \right| \\ &\leq \sum_{x \in L \setminus \sqrt{n}B_n} \rho(\{x\}) \\ &= \rho(L \setminus \sqrt{n}B_n). \end{aligned}$$

Auf die rechte Seite wenden wir Lemma 8.4 und Korollar 8.5 an und erhalten

$$\begin{aligned} \rho(L \setminus \sqrt{n}B_n) &< 2^{-\Omega(n)} \rho(L) \\ &< 2^{-\Omega(n)} \frac{1}{1 - 2^{-\Omega(n)}} \rho(L \cap \sqrt{n}B_n) \\ &\leq 2^{-\Omega(n)} \rho(L \cap \sqrt{n}B_n). \end{aligned}$$

Die einzigen Vektoren aus  $L$  die in  $\sqrt{n}B_n$  liegen, sind Vielfache von  $u$  und sind Elemente eines eindimensionalen Untergitters von  $L$ . Deshalb lässt sich Lemma 7.2 mit  $x = 0$  auf  $\rho(L \cap \sqrt{n}B_n)$  anwenden und man erhält

$$\rho(L \cap \sqrt{n}B_n) \leq \left(1 + \frac{1}{\|u\|}\right).$$

Damit ist

$$\left| \sum_{x \in L} e^{2\pi i \langle x, y \rangle} \rho(\{x\}) - \sum_{k \in \mathbb{Z}} e^{2\pi i k \langle u, y \rangle} \rho(\{ku\}) \right| < 2^{-\Omega(n)} \left(1 + \frac{1}{\|u\|}\right)$$

gezeigt.

Multiplizieren wir beide Seiten mit  $d(L)$  erhalten wir nach Lemma 8.6

$$\left| \rho(L^* + y) - d(L) \sum_{k \in \mathbb{Z}} e^{2\pi i k \langle u, y \rangle} \rho(\{ku\}) \right| < d(L) 2^{-\Omega(n)} \left(1 + \frac{1}{\|u\|}\right). \quad (6)$$

Nun müssen wir noch den Zusammenhang zwischen  $d(L) \sum_{k \in \mathbb{Z}} e^{2\pi i k \langle u, y \rangle} \rho(\{ku\})$  und  $T_{L^*, u}(y)$  ziehen.

Die Funktion  $T_{L^*, u}(y) = \frac{d(L)}{\|u\|} \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k + \langle u, y \rangle}{\|u\|}\right)^2}$  ist eine Summe von Auswertungen der Funktion  $f = e^{-\pi \left(\frac{k + \langle u, y \rangle}{\|u\|}\right)^2}$ . Die in  $f$  eingesetzten Argumente bilden ein um  $\frac{\langle u, y \rangle}{\|u\|}$  verschobenes eindimensionales Gitter, das von dem Wert  $\frac{1}{\|u\|}$  erzeugt wird. Diese Tatsache wollen wir jetzt konkretisieren und mit  $\rho$  in Verbindung bringen.

Ziel ist es, Lemma 8.6 anzuwenden. Dazu müssen wir das duale Gitter auf die richtige Weise einbeziehen. Sei  $M$  das Gitter, das von  $\|u\|$  erzeugt wird. Dann wird  $M^*$  von  $\frac{1}{\|u\|}$  erzeugt und es gilt

$$\begin{aligned} T_{L^*, u}(y) &= \frac{d(L)}{\|u\|} \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k + \langle u, y \rangle}{\|u\|}\right)^2} \\ &= \frac{d(L)}{\|u\|} \sum_{k \in \mathbb{Z}} e^{-\pi \left(\frac{k}{\|u\|} + \frac{\langle u, y \rangle}{\|u\|}\right)^2} \\ &= \frac{d(L)}{\|u\|} \sum_{m \in \mathbb{M}^*} e^{-\pi \left(m + \frac{\langle u, y \rangle}{\|u\|}\right)^2} \\ &= \frac{d(L)}{\|u\|} \rho\left(M^* + \frac{\langle u, y \rangle}{\|u\|}\right). \end{aligned}$$

Die Anwendung von Lemma 8.6 ergibt

$$\rho(M^* + a) = d(M) \sum_{b \in M} e^{2\pi i a b} \rho(\{b\}) = \|u\| \sum_{k \in \mathbb{Z}} e^{2\pi i a k \|u\|} \rho(\{k\|u\|\}).$$

Man beachte  $\rho(\{k\|u\|\}) = \rho(\{ku\})$ . Wenn man für  $a$  den Wert  $\frac{\langle u, y \rangle}{\|u\|}$  einsetzt, erhält man

$$\begin{aligned} T_{L^*, u}(y) &= \frac{d(L)}{\|u\|} \rho\left(M^* + \frac{\langle u, y \rangle}{\|u\|}\right) \\ &= \frac{d(L)}{\|u\|} \|u\| \sum_{k \in \mathbb{Z}} e^{2\pi i \frac{\langle u, y \rangle}{\|u\|} k \|u\|} \rho(\{ku\}) \\ &= d(L) \sum_{k \in \mathbb{Z}} e^{2\pi i \langle u, y \rangle k} \rho(\{ku\}). \end{aligned} \tag{7}$$

Die Gleichungen (6) und (7) ergeben

$$\begin{aligned} \Delta(D_{L^*}, T_{L^*, u}) &= \int_{\mathcal{P}(L^*)} \left| \rho(L^* + y) - d(L) \sum_{k \in \mathbb{Z}} e^{2\pi i k \langle u, y \rangle} \rho(\{ku\}) \right| dy \\ &< \int_{\mathcal{P}(L^*)} d(L) 2^{-\Omega(n)} \left(1 + \frac{1}{\|u\|}\right) dy \\ &= 2^{-\Omega(n)} \left(1 + \frac{1}{\|u\|}\right). \end{aligned}$$

Hängt die Länge des Vektors  $u$  nur polynomiell von der Dimension des Gitters ab, gilt auch der zweite Teil der Behauptung.

Nun wollen wir die Lösung des  $dSVP_p$  auf die Unterscheidbarkeit der Verteilungen  $U_{L^*}$  und  $T_{L^*,\tau(L)}$  reduzieren. Wir nutzen dabei folgende Tatsache aus. Angenommen  $p$  teilt den  $i$ -ten Koeffizienten des kürzesten Vektors. Ersetzt man den  $i$ -ten Basisvektor  $b_i$  durch  $p \cdot b_i$  so erhält man ein Untergitter  $\tilde{L}$  von  $L$ , mit gleichem kürzesten Vektor. Teilt  $p$  den Koeffizienten nicht, so erhalten wir durch die Multiplikation mit  $p$  ein Untergitter  $\tilde{L}$  von  $L$  mit größerem kürzesten Vektor. Wir werden  $p$  und  $L$  so wählen, dass in diesem Fall alle Vektoren des neuen Gitters größer als  $\sqrt{n}$  sind. Wir wissen aus den beiden vorherigen Lemmata, dass die resultierende Verteilung  $D_{\tilde{L}^*}$  exponentiell kleinen Abstand (in Abhängigkeit von  $n$ ) zu der Gleichverteilung oder zu der Verteilung  $T_{\tilde{L}^*,u}$  hat. Damit existiert keine Polynomialzeit-Turing-Maschine, die zwischen der Verteilung  $D_{\tilde{L}^*}$  und der Gleichverteilung bzw. der Verteilung  $T_{\tilde{L}^*,u}$  unterscheiden kann. Diese Tatsache werden wir ausnutzen, um die Reduktion durchzuführen.

**Lemma 8.14** *Sei  $g(n) < p(n)$  so, dass  $p(n)$  eine Primzahl ist und polynomiell von  $n$  abhängt. Das  $dSVP_{p(n)}$  auf einem  $g(n)$ -eindeutigen Gitter, kann auf die Unterscheidbarkeit zwischen  $U_{L^*}$  und  $T_{L^*,\tau(L)}$  reduziert werden. Die Basis von  $L$  sei hierbei LLL-reduziert und die Länge des kürzesten Vektors sei im Intervall  $[\frac{\sqrt{n}}{g(n)}, 2\frac{\sqrt{n}}{g(n)}]$ .*

Beweis: Sei  $B = (b_1, b_2, \dots, b_n)$  eine Gitterbasis eines  $g(n)$ -eindeutigen Gitters  $L$  und  $\alpha$  eine Zahl mit  $\lambda_1(L) < \alpha \leq 2\lambda_1(L)$ . Sei  $L'$  das Gitter mit der Basis  $B' = (v'_1, \dots, v'_n) := \frac{2\sqrt{n}}{\alpha \cdot g(n)} B$ .

Der kürzeste Vektor  $\tau(L')$  des Gitters  $L'$  ist dann  $\frac{2\sqrt{n}}{\alpha \cdot g(n)} \cdot \tau(L)$ . Aus  $2\lambda_1(L) \geq \alpha$  folgt  $\frac{2\sqrt{n}}{\alpha \cdot g(n)} \cdot \lambda_1(L) \geq \frac{\sqrt{n}}{g(n)}$  und aus  $\lambda_1(L) < \alpha$  das  $\lambda_1(L') < \frac{2\sqrt{n}}{g(n)}$ . Also liegt  $\lambda_1(L')$  im Intervall  $[\frac{\sqrt{n}}{g(n)}, 2\frac{\sqrt{n}}{g(n)}]$ .

Da  $L$  ein  $g(n)$ -eindeutiges Gitter ist, ist die Länge jedes nicht zu  $\tau(L)$  parallelen Vektor  $v$  in  $L$  mindestens  $g(n)\lambda_1(L)$ . Das hat zur Folge, dass jeder nicht zu  $\tau(L')$  parallele Vektor  $v' (= \frac{2\sqrt{n}}{\alpha \cdot g(n)} v, v \in L)$  in  $L'$  mindestens die Länge  $\sqrt{n}$  hat, denn

$$\|v'\| = \left\| \frac{2\sqrt{n}}{\alpha \cdot g(n)} v \right\| \geq \left\| \frac{2\sqrt{n}}{\alpha \cdot g(n)} g(n) \lambda_1(L) \right\| \stackrel{(1)}{\geq} \frac{2\sqrt{n}}{\alpha} \frac{\alpha}{2} = \sqrt{n}.$$

Die Ungleichung (1) gilt wegen  $2\lambda_1(L) \geq \alpha$ .

Sei  $M \subseteq L'$  das Gitter, dass von  $(p(n)v'_1, v'_2, \dots, v'_n)$  aufgespannt wird.

Wir nehmen an, dass die Basis von  $M$  LLL-reduziert ist. Dies verändert die eben beschriebenen Eigenschaften von  $M$  natürlich nicht. Nun betrachten wir die Verteilung  $D_{M^*}$ .

1. Fall:  $p(n) \mid a_1$

Wenn  $p(n)$  den ersten Koeffizienten  $a_1$  von  $\tau(L')$  teilt, haben die beiden Gitter den gleichen kürzesten Vektor. Damit hat jeder nicht zu  $\tau(M)$  parallele Vektor in  $M$  mindestens die Länge  $\sqrt{n}$ . Nach Lemma 8.13 sind die Verteilungen  $D_{M^*}$

und  $T_{M^*, \tau(M)}$  ununterscheidbar (siehe Def 4.17).

2.Fall:  $p(n) \nmid a_1$

Angenommen  $p(n) \nmid a_1$ . Dann ist der Vektor  $p(n)\tau(L)$  der kleinste Vektor in  $M$ , der in dem von  $\tau(L)$  erzeugten Gitter liegt. Dieser hat mindestens die Länge  $p(n)\frac{\sqrt{n}}{g(n)} > \sqrt{n}$ . Da alle nicht zu  $\tau(L)$  parallelen Vektoren in  $L$  und damit auch in  $M$  mindestens die Länge  $\sqrt{n}$  haben, sind alle Vektoren in  $M$  länger als  $\sqrt{n}$ . Nach Lemma 8.12 sind die Verteilungen  $D_{M^*}$  und  $U_{M^*}$  ununterscheidbar.

Man kann effizient  $D_{M^*}$ -verteilte Werte generieren, indem man normalverteilte Werte mit Eigenwert 0 und Standardabweichung  $\frac{1}{2\pi}$  erzeugt und modulo  $\mathcal{P}(M^*)$  reduziert. Wenn es also einen Polynomialzeitalgorithmus gibt, der zwischen  $U_{M^*}$  und  $T_{M^*, \tau(M)}$  unterscheiden kann, kann man mit dessen Hilfe entscheiden, ob die Zahl  $p(n)$  den ersten Koeffizienten des kürzesten Vektors teilt oder nicht.

Nun müssen wir die Verteilungen  $U_{L^*}$  und  $T_{L^*, \tau(L)}$ , in Verteilungen auf  $[0, 1)$  transformieren. Dies tun wir mittels einer Abbildung  $f$ , die Punkte aus der Grundmasche auf Punkte einer Gerade abbildet. Wichtig ist, dass durch diese Abbildung nicht zu viele Informationen verlorengehen. Um dies sicherzustellen, konstruieren wir  $f$  so, dass die Mengen, die auf den gleichen Punkt abgebildet werden, hinreichend klein sind. Die Idee, die hinter dieser Projektion steckt, wollen wir für das Zweidimensionale skizzieren.

Angenommen es ist ein Parallelogramm  $\mathcal{P}$  mit den Eckpunkten  $0, v_1, v_2$  und  $v_1 + v_2$  und eine Gerade  $g$ , die durch den Nullpunkt geht, gegeben. Desweiteren soll  $g$  einen Punkt  $p$  enthalten, der reduziert modulo  $\mathcal{P}$  gleich  $v_1 + v_2$  ist. Wenn wir  $g$  modulo  $\mathcal{P}$  reduzieren, erhalten wir eine Menge von Linien die  $\mathcal{P}$  durchqueren. Je größer die Steigung von  $g$  ist, desto mehr Linien durchlaufen das Parallelogramm. Wenn wir also die Steigung der Geraden erhöhen, können wir die Mengen, die in den gleichen Punkt projiziert werden, verkleinern. Gleiches gilt auch im höherdimensionalen.

Die Konstante  $K$  kann man sich als Anzahl der Linien, die das Parallelepipid durchlaufen, vorstellen.

**Lemma 8.15** *Sei  $L$  ein Gitter, dessen Basis LLL-reduziert ist und für das  $\lambda_1(L) \in [\frac{\sqrt{n}}{g(n)}, \frac{2\sqrt{n}}{g(n)})$  gilt. Dann existiert eine Konstante  $c_h$ , so dass für alle  $g(n) \geq 4\sqrt{n}$ ,  $g(n) \leq \text{poly}(n)$  das Problem zwischen  $T_{L^*, \tau(L)}$  und  $U_{L^*}$  zu unterscheiden auf das Problem, zwischen  $U$  und  $\Upsilon_{g(n)}$  zu unterscheiden, reduziert werden kann.*

Beweis: Sei  $(v_1, v_2, \dots, v_n)$  eine LLL-reduzierte Basis von  $L$  und  $(v_1^*, v_2^*, \dots, v_n^*)$  die duale Basis von  $L^*$ . Für ein später gewähltes großes  $K \in \mathbb{N}$  sei  $f$  die Abbildung,

$$f : \mathcal{P}(L^*) \rightarrow [0, 1)$$

$$\sum_{i=1}^n a_i v_i^* \mapsto \frac{\lfloor K a_1 \rfloor}{K} + \frac{\lfloor K a_2 \rfloor}{K^2} + \dots + \frac{\lfloor K a_{n-1} \rfloor}{K^{n-1}} + \frac{a_n}{K^{n-1}}$$

Sei  $r \in [0, 1)$ . Wir wollen uns überlegen, welche Werte aus  $\mathcal{P}(L^*)$  auf  $r$  abgebildet werden. Wegen Existenz und Eindeutigkeit der  $g$ -adischen Darstellung (Satz 11.9) wissen wir, dass eindeutig bestimmte Zahlen  $\tilde{r}_1 \tilde{r}_2 \dots \tilde{r}_n$  so existieren, dass

$$r = \tilde{r}_1 \frac{1}{K} + \tilde{r}_2 \frac{1}{K^2} \dots + \tilde{r}_n \frac{1}{K^n}$$

gilt.

Die Zahlen  $\tilde{r}_1, \tilde{r}_2, \dots, \tilde{r}_{n-1}$  sind aus der Menge  $\{0, 1, \dots, K-1\}$  und der Wert  $\tilde{r}_n$  ist aus dem Intervall  $[0, K)$ . Um aus dieser Darstellung einen Rückschluss auf die Werte, die auf  $r$  abgebildet werden, ziehen zu können, müssen wir die Faktoren  $\tilde{r}_i$  so verändern, dass sie im Intervall  $[0, 1)$  liegen. Dies erreichen wir, indem wir  $r_i = \frac{\tilde{r}_i}{K}$  für  $i = 1, \dots, n$  setzen.

Also gibt es für jedes  $r \in [0, 1)$  eine eindeutige Darstellung

$$r = r_1 + \frac{r_2}{K} + \dots + \frac{r_{n-1}}{K^{n-2}} + \frac{r_n}{K^{n-1}},$$

wobei die  $r_i$  für  $i < n$  aus der Menge  $\{0, \frac{1}{K}, \dots, \frac{K-1}{K}\}$  sind und  $r_n$  aus dem Intervall  $[0, 1)$  ist. Man sieht leicht, dass die Menge

$$S(r) := \left\{ \sum_{i=1}^n a_i v_i^* \mid (\forall i \in [n-1]) a_i \in [r_i, r_i + \frac{1}{K}) \text{ und } a_n = r_n \right\}.$$

auf  $r$  abgebildet wird und dass  $\bigcup_{r \in [0, 1)} S(r) = \mathcal{P}(L^*)$ .

Die Menge  $S(r)$  ist ein  $(n-1)$ -dimensionales Parallelepipet, dessen Durchmesser höchstens  $\frac{1}{K} \sum_{i=1}^{n-1} \|v_i^*\|$  groß ist.

Sei  $w$  der Vektor  $v_1^* + K v_2^* + \dots + K^{n-1} v_n^*$ . Man beachte, dass  $K r_i \in \mathbb{N}$  für  $i < n$  gilt. Also folgt

$$\begin{aligned} r w \bmod \mathcal{P}(L^*) &= \left( r_1 + \frac{r_2}{K} + \dots + \frac{r_n}{K^{n-1}} \right) v_1^* \bmod \mathcal{P}(L^*) \\ &+ \left( K r_1 + r_2 + \frac{r_3}{K} + \dots + \frac{r_n}{K^{n-2}} \right) v_2^* \bmod \mathcal{P}(L^*) \\ &: \\ &+ \left( K^{n-1} r_1 + \dots + K r_{n-1} + r_n \right) v_n^* \bmod \mathcal{P}(L^*) \\ &= \left( r_1 + \frac{r_2}{K} + \frac{r_3}{K^2} + \dots + \frac{r_n}{K^{n-1}} \right) v_1^* \\ &+ \left( r_2 + \frac{r_3}{K} + \dots + \frac{r_n}{K^{n-2}} \right) v_2^* \\ &: \\ &+ r_n v_n^*. \end{aligned}$$

Da die Zahlen  $r_1, \dots, r_n$  in der oben beschriebenen Weise mit der  $K$ -adischen Entwicklung von  $r$  zusammenhängen, wissen wir, dass die Summe  $\sum_{j=i}^n K^{i-j} r_j$

für alle  $i \in \{1, \dots, n\}$  im Intervall  $[0, 1)$  liegen muss. Also gilt

$$\begin{aligned} f(rw \bmod \mathcal{P}(L^*)) &= \frac{\lfloor Kr_1 + r_2 + \dots + \frac{r_n}{K^{n-2}} \rfloor}{K} \\ &+ \frac{\lfloor Kr_2 + r_3 + \dots + \frac{r_n}{K^{n-3}} \rfloor}{K^2} \\ &: \\ &+ \frac{r_n}{K^{n-1}} \\ &= r_1 + \frac{r_2}{K} + \dots + \frac{r_n}{K^{n-1}} = r \end{aligned}$$

Nun wollen wir die Funktion  $f$  benutzen, um die mehrdimensionalen Verteilungen  $U_{L^*}$  und  $T_{L^*, \tau(L)}$  in Verteilungen auf dem Intervall  $[0, 1)$  zu transformieren. Man beachte, dass die Funktion  $f$  effizient berechnet werden kann.

Startet man mit der Gleichverteilung auf  $\mathcal{P}$  so erhält man auf dem Intervall  $[0, 1)$  wieder die Gleichverteilung, da  $\text{vol}S(r) = \text{vol}S(r')$  für alle  $r, r' \in [0, 1)$  gilt.

Also müssen wir nur noch  $T_{L^*, \tau(L)}$  betrachten. Die Dichtefunktion, die wir durch  $f$  auf  $[0, 1)$  erhalten ist:

$$T_1(r) := \frac{d(L^*)}{\text{vol}(S(r))} \int_{S(r)} T_{L^*, \tau(L)}(x) dx.$$

Nun schließen wir, dass der Abstand zwischen der Funktion  $T_1$  und der Funktion

$$T_{|\langle \tau(L), w \rangle|, \lambda_1(L)^2}(r) = \frac{1}{\lambda_1(L)} \sum_{k \in \mathbb{Z}} e^{-\pi \left( \frac{k + \langle \tau(L), w \rangle}{\lambda_1(L)} \right)^2} = d(L^*) T_{L^*, \tau(L)}(rw)$$

kleiner  $2^{-\Omega(n)}$  ist. Der erste Teil der Gleichung gilt, da  $\langle \tau(L), w \rangle$  eine ganze Zahl ist und sich die Funktion nicht ändert wenn wir das Vorzeichen von  $\langle \tau(L), w \rangle$  ändern.

Sei  $\text{diam}(S(r))$  der Durchmesser von  $S(r)$ . Sei  $a$  der Wert in dem die Funktion  $T_{L^*, \tau(L)}$  in  $S(r)$  ihr Maximum und  $b$  der Punkt in dem sie ihr Minimum annimmt. Nach dem Mittelwertsatz (Satz 3.23) gilt die erste Ungleichung.

$$\begin{aligned} |T_{L^*, \tau(L)}(b) - T_{L^*, \tau(L)}(a)| &\leq \text{diam}(S(r)) \cdot \max_{x \in \mathcal{P}(L^*)} \frac{d}{dx} \left( \frac{d(L)}{\lambda_1(L)} \sum_{k \in \mathbb{Z}} e^{-\pi \left( \frac{k + \langle \tau(L), x \rangle}{\lambda_1(L)} \right)^2} \right) \\ &\leq c \cdot \frac{1}{K} \sum_{i=1}^{n-1} \|v_i^*\| \cdot \frac{d(L)}{\lambda_1(L)} \end{aligned}$$

Die letzte Ungleichung folgt aus Lemma 7.3. Lemma 7.3 lässt sich anwenden, da  $\frac{1}{\lambda_1(L)} \geq 2$  aus der Annahme  $\lambda_1(L) \leq \frac{2\sqrt{n}}{g(n)} \leq \frac{1}{2}$  folgt.

Es gilt für alle  $r \in [0, 1)$ ,

$$\begin{aligned}
|T_1(r) - T_{|\langle \tau(L), w \rangle|, \lambda_1(L)^2}(r)| &= \frac{d(L^*)}{\text{vol}(S(r))} \left| \int_{S(r)} T_{L^*, \tau(L)}(x) - T_{L^*, \tau(L)}(rw) dx \right| \\
&\leq \frac{d(L^*)}{\text{vol}(S(r))} \int_{S(r)} |T_{L^*, \tau(L)}(x) - T_{L^*, \tau(L)}(rw)| dx \\
&\leq d(L^*) \cdot c \cdot \frac{1}{K} \sum_{i=1}^{n-1} \|v_i^*\| \cdot \frac{d(L)}{\lambda_1(L)} \\
&= c \cdot \frac{1}{K} \sum_{i=1}^{n-1} \|v_i^*\| \cdot \frac{1}{\lambda_1(L)}.
\end{aligned}$$

Die Ungleichung gilt, da wir nachgerechnet haben, dass  $rw$  in  $S(r)$  liegt und wegen der obigen Anwendung des MWS.

Da nach Annahme die Basis von  $L$  LLL-reduziert ist, können wir Lemma 7.15(iii) anwenden. Also gilt

$$\begin{aligned}
|T_1(r) - T_{|\langle \tau(L), w \rangle|, \lambda_1(L)^2}(r)| &\leq c \cdot \frac{1}{K} \sum_{i=1}^{n-1} \|v_i^*\| \cdot \frac{1}{\lambda_1(L)} \\
&\leq c \cdot \frac{1}{K} \cdot n \cdot \frac{\sqrt{n}}{\lambda_1(L)} \cdot 2^{2n} \cdot \frac{1}{\lambda_1(L)} \\
&\leq c \cdot \sqrt{n} \cdot g(n)^2 \frac{1}{K} \cdot 2^{2n}.
\end{aligned}$$

Wenn wir die Konstante  $K = 2^{3n}$  setzen, ist der Abstand zwischen diesen beiden Verteilungen exponentiell klein. Damit sind die Verteilungen für Polynomialzeit-Turing-Maschinen ununterscheidbar.

Es ist noch zu klären, wie groß wir die Konstante  $c_h$  in Definition 6.8 wählen müssen, damit die Funktion  $T_{|\langle \tau(L), w \rangle|, \lambda_1(L)^2}(r)$  in  $\Upsilon_{g(n)}$  liegt. Dazu müssen wir klären, wie groß das Skalarprodukt  $|\langle \tau(L), w \rangle|$  mit  $w = v_1^* + K v_2^* + \dots + K^{n-1} v_n^*$  und  $\tau(L) = a_i v_i$  werden kann. Nach Lemma 7.15 (i) können wir den Betrag der  $a_i$  jeweils mit  $2^{2n}$  abschätzen. Wegen  $\langle v_i, v_j^* \rangle = \delta_{ij}$ , muss das Skalarprodukt  $z = \langle \tau(L), w \rangle$  eine ganze Zahl sein. Für  $c_h = 4$  gilt

$$\begin{aligned}
|z| &\leq 2^{2n} \sum_{i=1}^n K^{i-1} \leq 2^{2n} \frac{K^n - 1}{K - 1} \\
&= 2^{2n} \frac{2^{3n^2} - 1}{2^{3n} - 1} \\
&\leq 2^{c_h n^2}.
\end{aligned}$$

Dabei wurde Satz 11.5 angewandt. Wegen  $\lambda_1(L) \in [\frac{\sqrt{n}}{g(n)}, \frac{2\sqrt{n}}{g(n)})$  ist die Behauptung bewiesen.

Durch die letzte Rechnung erhalten wir folgendes Ergebnis.

**Lemma 8.16** *Es gilt  $c_h \geq 4$ .*

## 9 Die Analyse des Verschlüsselungsverfahrens

In diesem Kapitel wollen wir die Korrektheit und Sicherheit des Kryptosystems beweisen. Dafür ist Folgendes zu zeigen. Sei  $n$  ein Sicherheitsparameter.

- Ein Entschlüsselungsfehler tritt mit zu vernachlässigender Wahrscheinlichkeit in Abhängigkeit des Sicherheitsparameters  $n$  auf (Lemma 9.1).
- Die Sicherheit des Systems basiert auf einem  $SV P$  in einem  $O(n^{1.5})$ -eindeutigem Gitter (Lemma 9.13).

Zunächst aber noch einige Vorbemerkungen. Angenommen es ist eine Dichtefunktion  $\rho$  auf den reellen Zahlen gegeben, so erhält man auf einem Intervall  $[0, b)$  mit  $b > 0$  eine Dichtefunktion  $\tilde{\rho}$  auf folgende Weise. Man definiert den Funktionswert der Dichtefunktion  $\tilde{\rho}$  an der Stelle  $x \in [0, b)$  durch die Summe aller Funktionswerte  $\rho(y)$  mit  $y \bmod b$  gleich  $x$ . Die resultierende Dichtefunktion werden wir mit  $\rho \bmod b$  bezeichnen. Der Beweis, dass die resultierenden Funktionen Dichtefunktionen sind, ist analog zu dem Beweis, dass  $T_{h,\beta}$  eine Dichtefunktion ist (siehe Lemma 6.7).

Die Konstanten,  $N, d, h, m, i_0, \beta$  sind ebenso wie die Funktion  $\gamma(n)$  und die Menge von Funktionen  $\Upsilon_{g(n)}$  wie in Kapitel 6 definiert. Die Konstruktion der Elemente des öffentlichen Schlüssels  $a_i$  mit  $i \in [m]$  erfolgt ebenfalls wie in Kapitel 6 beschrieben. Desweiteren bezeichnet  $U$  die Gleichverteilung auf der Menge  $[N]$ . Nun zum Beweis der Korrektheit des Kryptosystems.

**Lemma 9.1 (Korrektheit)** *Die Wahrscheinlichkeit für einen Entschlüsselungsfehler ist höchstens  $2^{-\Omega(\frac{\gamma(n)^2}{m})}$  zuzüglich einiger in  $n$  exponentiell kleiner Terme.*

Beweis: Sei  $\omega$  der Schlüsseltext. Ein Schlüsseltext wird als Null entschlüsselt, falls  $\text{frc}(\frac{\omega}{d}) < \frac{1}{4}$  ist und als Eins sonst.

1. Fall:  $\omega$  ist eine Verschlüsselung der 0:

Dann ist  $\omega$  von der Form  $\sum_{i \in S} a_i \bmod N$ , wobei  $S$  eine zufällig gewählte Teilmenge von  $[m]$  ist. Zu zeigen ist, dass  $\text{frc}(\frac{\omega}{d})$  mit exponentiell nah bei Eins liegender Wahrscheinlichkeit kleiner als  $\frac{1}{4}$  ist. Bei folgenden Gleichungen ist zu beachten, dass  $N = dh$  gilt.

$$\begin{aligned} \left| \omega - \sum_{i \in S} a_i \bmod d[h] \right| &= \left| \sum_{i \in S} a_i \bmod N - \sum_{i \in S} a_i \bmod d[h] \right| \\ &= \left| \sum_{i \in S} a_i + k_1 \cdot dh - \left( \sum_{i \in S} a_i + k_2 \cdot d[h] \right) \right| \\ &= \left| k_1 \cdot N - k_2 \cdot d[h] \right| \end{aligned}$$

Nach Lemma 7.6 sind die Konstanten  $k_1, k_2$  mit zu vernachlässigender Wahrscheinlichkeit ungleich. Also können wir diesen Fall vernachlässigen. Da  $S$  höchst-

tens  $m$  Summanden hat, ist  $\sum_{i \in S} a_i \leq m \cdot N$  und es gilt

$$\begin{aligned} \left| \sum_{i \in S} a_i + k_1 \cdot N - \left( \sum_{i \in S} a_i + k_2 \cdot d[h] \right) \right| &\leq m \cdot |dh - d[h]| \\ &= m \cdot d \cdot \text{frc}(h) < \frac{1}{16}d. \end{aligned}$$

Dabei folgt die letzte Ungleichung aus der Konstruktion des geheimen Schlüssels. Nun benutzen wir diese Abschätzung, um eine Schranke für  $\text{frc}\left(\frac{w}{d}\right)$  zu finden. Nach Lemma 1.5 gilt  $\text{frc}(a+b) \leq \text{frc}(a) + \text{frc}(b)$ . Damit gilt folgende Abschätzung

$$\begin{aligned} \text{frc}\left(\frac{w}{d}\right) &= \text{frc}\left(\frac{w - \sum_{i \in S} a_i \bmod d[h] + \sum_{i \in S} a_i \bmod d[h]}{d}\right) \\ &\leq \text{frc}\left(\frac{w - \sum_{i \in S} a_i \bmod d[h]}{d}\right) + \text{frc}\left(\frac{\sum_{i \in S} a_i \bmod d[h]}{d}\right) \\ &< \frac{1}{16} + \text{frc}\left(\frac{\sum_{i \in S} a_i \bmod d[h]}{d}\right). \end{aligned}$$

Diesen Trick haben wir angewandt, um nicht mehr modulo rechnen zu müssen. Es gibt nämlich ein  $k \in \mathbb{Z}$ , so dass

$$\text{frc}\left(\frac{\sum_{i \in S} a_i \bmod d[h]}{d}\right) = \text{frc}\left(\frac{\sum_{i \in S} a_i + k[h]d}{d}\right) = \text{frc}\left(\frac{\sum_{i \in S} a_i}{d}\right)$$

gilt. Bis jetzt haben wir

$$\text{frc}\left(\frac{w}{d}\right) < \frac{1}{16} + \text{frc}\left(\frac{\sum_{i \in S} a_i}{d}\right).$$

Nun müssen wir noch den rechten Teil untersuchen. Aus der Konstruktion der  $a_i$  wissen wir, dass  $a_i = \lfloor N \cdot z_i \rfloor$  gilt. Die  $z_i$  sind dabei von der Form  $\frac{x_i + y_i}{h}$ , wobei  $x_i$  gleichverteilt aus der Menge  $\{1, \dots, [h] - 1\}$  und  $y_i$  ein  $Q_\beta$ -verteilter Wert ist. Also folgt  $|N \cdot z_i - a_i| < 1$ . Mit der Dreiecksungleichung gilt

$$\text{frc}\left(\frac{\sum_{i \in S} a_i}{d}\right) < \frac{m}{d} + \text{frc}\left(\frac{\sum_{i \in S} N \cdot z_i}{d}\right) < \frac{1}{16} + \text{frc}\left(\frac{\sum_{i \in S} N \cdot z_i}{d}\right). \quad (8)$$

Man beachte, dass nach Konstruktion  $\frac{N}{d} z_i = h \cdot \frac{(x_i + y_i)}{h} = x_i + y_i$  gilt. Also folgt

$$\text{frc}\left(\frac{\sum_{i \in S} N \cdot z_i}{d}\right) = \text{frc}\left(\sum_{i \in S} y_i\right).$$

Die  $y_i$  sind  $Q_\beta$ -verteilt. Dies hat zur Folge, dass die Summe  $\sum_{i \in S} y_i \bmod 1$   $Q_{|S|\beta}$ -verteilt ist (siehe dazu Lemma 9.3). Also ist die Wahrscheinlichkeit für das

Ereignis  $\text{frc}(\sum_{i \in S} y_i) > \frac{1}{16}$  höchstens genauso groß wie die Wahrscheinlichkeit, dass Werte einer normalverteilten Zufallsvariable mit Erwartungswert 0 und Varianz  $|S|\beta \leq m \cdot \beta = O(\frac{m}{(\gamma(n))^2})$  im Intervall  $[-\frac{1}{16}, \frac{1}{16}]$  liegen (siehe Definition 6.4). Diese Wahrscheinlichkeit ist nach Lemma 7.1 höchstens

$$\sqrt{\frac{2}{\pi}} \cdot \frac{\sigma}{t} e^{-\frac{t^2}{2\sigma^2}}.$$

Setzen wir obige Abschätzung für die Varianz ein, erhalten wir, dass die Wahrscheinlichkeit

$$P(\text{frc}(\sum_{i \in S} y_i) > \frac{1}{16}) \leq 2^{-\Omega(\frac{(\gamma(n))^2}{m})}$$

ist. Wenn wir dieses Ergebnis in die Ungleichung (8) einsetzen, gilt mit einer Wahrscheinlichkeit von  $1 - 2^{-\Omega(\frac{(\gamma(n))^2}{m})}$

$$\text{frc}\left(\frac{w}{d}\right) < \frac{1}{8} + \frac{1}{16}. \quad (9)$$

Damit ist die Aussage für den ersten Fall bewiesen.

2.Fall:  $\omega$  war eine Verschlüsselung der 1.

Dann hat  $\omega$  die Form  $\sum_{i \in S} a_i \bmod N + \lfloor \frac{a_{i_0}}{2} \rfloor$ , wobei  $S$  eine zufällig gewählte

Teilmenge von  $[m]$  ist. Wir wissen mit exponentiell nahe bei Eins liegender Wahrscheinlichkeit, dass  $\text{frc}(y_{i_0})$  kleiner als  $\frac{1}{16}$  ist (Lemma 7.1). In diesem Fall gilt

$$\begin{aligned} \text{frc}\left(\frac{\lfloor a_{i_0}/2 \rfloor}{d}\right) &= \text{frc}\left(\frac{\lfloor \lfloor N \cdot \frac{1}{h}(x_{i_0} + y_{i_0}) \rfloor / 2 \rfloor}{d}\right) \\ &> \text{frc}\left(\frac{N \cdot \frac{1}{h}(x_{i_0} + y_{i_0})}{2d}\right) - \frac{1}{d} \\ &\geq \text{frc}\left(\frac{N \cdot x_{i_0}}{2dh}\right) - \text{frc}\left(\frac{y_{i_0}}{2dh}\right) - \frac{1}{d} \\ &> \frac{1}{2} - \frac{1}{32} - \frac{1}{d}. \end{aligned}$$

Kombiniert mit (9) und wegen  $\text{frc}(a - b) \geq \text{frc}(a) - \text{frc}(b)$  (siehe Lemma 1.5) erhält man

$$\text{frc}\left(\frac{w}{d}\right) \geq \text{frc}\left(\frac{\lfloor a_{i_0}/2 \rfloor}{d}\right) - \text{frc}\left(\frac{\sum_{i \in S} a_i}{d}\right) > \frac{1}{2} - \frac{1}{32} - \frac{1}{d} - \frac{1}{8} - \frac{1}{16} > \frac{1}{4}.$$

Damit ist der Beweis vollständig.

Bevor wir zu dem Beweis der Sicherheit des Systems kommen können, müssen wir noch einige Eigenschaften der Funktionen  $T_{h,\beta}$  und  $Q_\delta$  herleiten.

**Lemma 9.2** *Sei  $h \in \mathbb{N}$ ,  $\beta \in \mathbb{R}$  und  $X, Y$  zwei unabhängige Zufallsvariablen.  $X$  sei gleichverteilt auf der Menge  $\{0, \frac{1}{h}, \dots, \frac{h-1}{h}\}$  und  $Y$  eine Normalverteilung mit Eigenwert 0 und Varianz  $\frac{\beta}{2\pi h^2}$ . Dann gilt*

$$T_{h,\beta} = X + Y \bmod 1.$$

Beweis: Es gilt

$$T_{h,\beta}(r) = Q_\beta(hr \bmod 1) = \sum_{k=-\infty}^{\infty} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(hr-k)^2}.$$

Jede natürliche Zahl  $k$  hat eine eindeutige Darstellung  $k = hm + l$  mit  $0 \leq l < h$ . Außerdem konvergiert die Reihe absolut. Ihr Grenzwert ist also nach Satz 11.5 invariant unter Umordnung und es folgt

$$\begin{aligned} \sum_{k=-\infty}^{\infty} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(hr-k)^2} &= \sum_{l=0}^{h-1} \sum_{m=-\infty}^{\infty} \frac{1}{\sqrt{\beta}} e^{-\frac{\pi}{\beta}(hr-hm-l)^2} \\ &= \sum_{l=0}^{h-1} \frac{1}{h} \sum_{m=-\infty}^{\infty} \frac{h}{\sqrt{\beta}} e^{-\frac{\pi h^2}{\beta}(r-m-\frac{l}{h})^2}. \end{aligned}$$

Jetzt müssen wir nur noch überlegen, dass das die Dichtefunktion  $\rho$  von  $X+Y$  reduziert modulo 1 ist. Sei dazu  $r \in [0, 1[$ . Für jeden Wert  $\frac{l}{h}$ ,  $l \in \{0, 1, \dots, h-1\}$  den  $X$  annehmen kann, gibt es eine Zahl  $t \in ]-1, 1[$  mit  $r = \frac{l}{h} + t$ . Wenn man eine ganze Zahl zu  $t$  addiert kommt modulo 1 ebenfalls  $r$  raus. Also folgt

$$\rho(r) = \sum_{l=0}^{h-1} \frac{1}{h} \sum_{m=-\infty}^{\infty} \frac{h}{\sqrt{\beta}} e^{-\frac{\pi h^2}{\beta}(r-\frac{l}{h}+m)^2}.$$

Als nächstes wollen wir zeigen, dass Lemma 6.3 auch für die modulo 1 reduzierten normalverteilten Zufallsvariablen  $Q_\delta$  gilt.

**Lemma 9.3** *Es gilt  $Q_\delta + Q_\gamma \bmod 1 = Q_{\delta+\gamma}$ .*

Beweis: Seien  $X, Y$  unabhängige normalverteilte Zufallsvariablen mit Eigenwert 0 und  $\text{Var}(X) = \delta$  und  $\text{Var}(Y) = \gamma$ . Dann ist die Zufallsvariable  $Z = X + Y$  nach Lemma 6.3 normalverteilt mit Eigenwert 0 und Varianz  $\delta + \gamma$ . Wenn man diese Zufallsvariable modulo 1 reduziert, erhält man die Verteilung  $Q_{\delta+\gamma}$  auf dem Intervall  $[0, 1)$ . Es gilt

$$(x_1 \bmod 1 + x_2 \bmod 1) \bmod 1 = (x_1 + x_2) \bmod 1.$$

Es ist also unerheblich, ob wir erst normalverteilte Zufallsvariablen addieren und dann reduzieren oder sie erst reduzieren und dann modulo 1 addieren. Somit gilt

$$(Q_\delta + Q_\gamma) \bmod 1 = Z \bmod 1 = Q_{\delta+\gamma}.$$

**Lemma 9.4** *Für jedes  $h \in \mathbb{N}$  gilt*

$$(T_{h,\beta} + Q_\delta) \bmod 1 = T_{h,\beta+\delta h^2}.$$

Beweis: Nach Lemma 9.2 ist  $T_{h,\beta} = X + Y \bmod 1$ , wobei  $Y$  normalverteilt mit Eigenwert 0 und Varianz  $\frac{\beta}{2\pi h^2}$  ist. Wie wir bereits in Lemma 9.3 gesehen haben, gilt

$$((X + Y) \bmod 1 + Q_\delta) \bmod 1 = (X + (Y \bmod 1 + Q_\delta) \bmod 1) \bmod 1.$$

Es gilt  $Y \bmod 1 = Q_{\frac{\delta}{h^2}}$  (siehe Definition 6.4). Also folgt aus Lemma 9.3

$$(X + (Y \bmod 1 + Q_{\delta}) \bmod 1) \bmod 1 = (X + Q_{\frac{\delta}{h^2} + \delta}) \bmod 1.$$

Nun wenden wir wieder Lemma 9.2 an, und der Beweis ist komplett.

**Definition 9.5** Sei  $X$  eine Dichtefunktion auf  $[0, 1)$ . Eine Stauchung von  $X$  auf  $[0, 1)$  mit dem Faktor  $\delta \geq 1$  ist durch die Dichtefunktion

$$C_{\delta}(r) := \frac{1}{\int_0^1 X(\delta x \bmod 1) dx} X(\delta r \bmod 1)$$

definiert.

**Lemma 9.6** Sei  $h \in \mathbb{N}$  und  $\delta \geq 1$ . Die Stauchung von  $T_{h,\beta}$  um den Faktor  $\delta$  ist  $T_{\delta h,\beta}$ .

Beweis: Der Beweis folgt direkt aus der Definition von  $T_{h,\beta}$  (siehe Definition 6.6).

Aus der Definition der Funktion  $T_{h,\beta}$  folgt, dass sie periodisch mit Periode  $\frac{1}{h}$  ist. Das Kryptosystem ist eine Diskretisierung dieser Verteilung. Gipfel der Dichtefunktion werden als Verschlüsselungen der Null, Täler als Verschlüsselung der Eins interpretiert. Die Periodizität der Verteilung wird also auf das Kryptosystem übertragen. In Kapitel 10 werden wir sehen, dass dies eine Schwachstelle des Verfahrens ist.

In Abschnitt 6.2 wird beschrieben, wie man  $T_{h,\beta}$ -verteilte Werte generiert. Hierzu wählt man gleichverteilt ein  $h \in \{0, 1, \dots, \lceil h \rceil - 1\}$  aus und addiert einen  $Q_{\beta}$ -verteilten Wert hinzu. Damit man keine Informationen aus den beobachteten Schlüsseltexten ziehen kann, darf es keine Perioden geben, in denen ein Schlüsseltext mit wesentlich höherer Wahrscheinlichkeit liegt. Um dies zu erreichen, müssen wir die Anzahl der Elemente des öffentlichen Schlüssels so hoch wählen, dass die Verteilung, die durch die "zufällige" Summation von gleichverteilten Elementen entsteht, nicht von der Gleichverteilung zu unterscheiden ist. Dadurch wird die Periode, in der ein Schlüsseltext liegt, zufällig. Aus dieser Zahl resultiert die Größe des öffentlichen Schlüssels. Um sie zu bestimmen, müssen wir uns zunächst eine Zufallsvariable definieren, die die Summation von zufällig gewählten Teilmengen der Menge  $\{1, 2, \dots, 2^l - 1\}$  simuliert.

**Definition 9.7** Seien  $c, l \in \mathbb{N}$ . Die Zufallsvariable  $U_{c,l}$  sei durch folgendes Zufallsexperiment definiert: Wähle Zahlen  $a_1, \dots, a_{c,l}$  gleichmäßig aus der Menge  $\{0, \dots, 2^l - 1\}$  aus und addiere diese zusammen.

**Lemma 9.8** Die Wahrscheinlichkeit, dass der statistische Abstand zwischen der Gleichverteilung auf der Menge  $\{0, \dots, 2^l - 1\}$  und der Zufallsvariable  $U_{c,l}$  größer als  $2^{-l}$  ist, ist für ausreichend großes  $c$  höchstens  $2^{-l}$ .

Um dieses Lemma zu beweisen, benötigen wir die folgenden zwei Zufallsvariablen. Die Zufallsvariable  $X_{t,b}$  soll simulieren, mit welcher Wahrscheinlichkeit

eine zufällig gewählte Teilmenge den Wert  $t$  annimmt. Die Zufallsvariable  $Y_t$  zählt die Teilmengen, deren Summe  $t$  ist. Die Wahrscheinlichkeit hängt von der Wahl des öffentlichen Schlüssels  $(a_1, \dots, a_{c,l})$  ab.

**Definition 9.9** Sei  $t \in \{0, \dots, 2^l - 1\}$  und  $b \in \{0, 1\}^{c,l} \setminus 0^{c,l}$ . Die diskrete Zufallsvariable  $X_{t,b}$  ist definiert durch:

$$X_{t,b} : (\mathbb{Z}/2^l)^{c,l} \rightarrow \{0, 1\},$$

$$(a_1, \dots, a_{c,l}) \mapsto \begin{cases} 1 & , \text{ falls } \sum_{i=0}^{c,l} b_i a_i \equiv t \pmod{2^l} \\ 0 & , \text{ sonst} \end{cases}$$

**Bemerkung 9.10** Der Erwartungswert von  $X_{t,b}$  ist  $2^{-l}$ , da eine zufällig gewählte Teilmenge jeden Wert  $t \in \{0, \dots, 2^l - 1\}$  mit gleicher Wahrscheinlichkeit annehmen kann. Also ist

$$P(X_{t,b} = 1) = 2^{-l}.$$

Da die Zufallsvariable sonst nur den Wert 0 annimmt, entspricht dies dem Erwartungswert der Zufallsvariable.

**Lemma 9.11** Seien  $b, b' \in \{0, 1\}^{c,l} \setminus 0^{c,l}$  mit  $b \neq b'$ . Dann sind die Zufallsvariablen  $X_{t,b}$  und  $X_{t,b'}$  unabhängig.

Beweis: Unabhängigkeit von Zufallsvariablen bedeutet (siehe Definition 4.10), dass die Verteilungsfunktion der zweidimensionalen Zufallsvariable  $(X_{t,b}, X_{t,b'})$  die Bedingung  $F(X_{t,b}, X_{t,b'}) = F(X_{t,b}) \cdot F(X_{t,b'})$  erfüllt. Das heißt also, dass wir überprüfen müssen, ob  $P(X_{t,b} = i, X_{t,b'} = j) = P(X_{t,b} = i) \cdot P(X_{t,b'} = j)$  für alle  $(i, j)$  aus der Menge  $\{0, 1\}^2$  erfüllt ist.

Angenommen  $X_{t,b}$  nimmt den Wert 1 an. Das bedeutet, dass es eine zu  $b$  korrespondierende Menge  $B$  gibt, so dass die Summe über alle Elemente aus  $B$  gleich  $t$  ist. Da  $b$  ungleich  $b'$  ist, gibt es ein Element, das in der einen Summe, jedoch nicht in der anderen enthalten sind. Da die Elemente zufällig gewählt wurden, ist die Wahrscheinlichkeit, dass  $X_{t,b'}$  den Wert Eins annimmt, genauso hoch wie vorher. Das Gleiche gilt, wenn  $X_{t,b}$  Null war. Also sind die Zufallsvariablen  $X_{t,b}$  und  $X_{t,b'}$  unabhängig.

**Definition 9.12** Die Zufallsvariable  $Y_t$  ist definiert durch

$$Y_t := \sum_{b \in \{0,1\}^{c,l} \setminus 0^{c,l}} X_{t,b}.$$

Beweis von Lemma 9.8: Seien  $X_{t,b}$  und  $Y_t$  wie oben definiert. Für die Varianz von  $X_{t,b}$  gilt

$$\begin{aligned} V(X_{t,b}) &= (1 - 2^{-l})^2 \cdot 2^{-l} + (2^{-l})^2(1 - 2^{-l}) \\ &= 2^{-l} - 2^{-(2l-1)} + 2^{-3l} + 2^{-2l} - 2^{-3l} \\ &= 2^{-l} - 2^{-2l} < 2^{-l} \end{aligned}$$

Nun untersuchen wir die Zufallsvariable  $Y_t$ . Wegen der Linearität des Erwartungswertes (Satz 4.11) muss  $E(Y_t) = (2^{c \cdot l} - 1) \cdot E(X_{t,b}) = \frac{2^{c \cdot l} - 1}{2^l} = 2^{(c-1) \cdot l} - 2^{-l}$  gelten. Da nach Lemma 9.11 die Zufallsvariablen  $X_{t,b}$  und  $X_{t,b'}$  für  $b \neq b'$  unabhängig sind, sind sie nach Satz 4.15 auch unkorreliert. Damit gilt nach Bemerkung 4.14

$$V(Y_t) = \sum_{b \in \{0,1\}^{c \cdot l} \setminus 0^{c \cdot l}} V(X_{t,b}) = \frac{2^{c \cdot l} - 1}{2^l} < 2^{(c-1) \cdot l}.$$

Wenn man diese Abschätzung für die Varianz in die Tschebyscheffsche Ungleichung (Satz 4.8) einsetzt, erhält man

$$P\left(\left|Y_t - (2^{(c-1) \cdot l} - 2^{-l})\right| \geq 2^{(\frac{c-1}{2}+1) \cdot l}\right) \leq 2^{-2l}.$$

Nun können wir die Dreiecksungleichung anwenden und erhalten

$$P\left(\left|Y_t - 2^{(c-1) \cdot l}\right| \geq 2^{(\frac{c-1}{2}+1) \cdot l} + 2^{-l}\right) \leq 2^{-2l}.$$

Die Wahrscheinlichkeit dafür, dass mindestens ein  $t$  existiert für das die Bedingung  $\left|Y_t - 2^{(c-1) \cdot l}\right| \geq 2^{(\frac{c-1}{2}+1) \cdot l} + 2^{-l}$  erfüllt ist, ist also höchstens  $2^l \cdot 2^{-2l} = 2^{-l}$ .

Das bedeutet, dass, bei festem Schlüssel  $(a_1, a_2, \dots, a_{c \cdot l})$ , die Anzahl der Teilmengen, die auf  $t$  abgebildet werden, mit mindestens der Wahrscheinlichkeit  $1 - 2^{-l}$  nicht mehr als  $2^{(\frac{c-1}{2}+1) \cdot l} + 2^{-l} + 1$  von  $2^{(c-1) \cdot l}$  entfernt ist.

Die Wahrscheinlichkeit, dass eine zufällig gewählte Teilmenge den Wert  $t$  annimmt, ist

$$\frac{Y_t}{2^{c \cdot l} - 1}.$$

Mit einer Wahrscheinlichkeit von mindestens  $1 - 2^{-l}$  gilt folgende Abschätzung für den statistischen Abstand zwischen der Gleichverteilung  $U$  und der Verteilung, die durch die Summation von zufällig gewählten Teilmengen von  $\{0, \dots, 2^l - 1\}$  entsteht,

$$\begin{aligned} \frac{1}{2} \sum_{t \in \{0, 1, \dots, 2^l - 1\}} \left| \frac{Y_t}{2^{c \cdot l} - 1} - 2^{-l} \right| &\leq 2^l \max\left\{ \left| \frac{Y_t}{2^{c \cdot l} - 1} - 2^{-l} \right| : t \in [2^l - 1] \right\} \\ &= 2^l \max\left\{ \left| \frac{Y_t}{2^{c \cdot l} - 1} - \frac{2^{(c-1) \cdot l} - 2^{-l}}{2^{c \cdot l} - 1} \right| : t \in [2^l - 1] \right\} \\ &\leq 2^l (2^{(\frac{c-1}{2}+1) \cdot l} + 2^{-l} + 1 + 2^{-l}) \cdot (2^{-c \cdot l} - 1). \end{aligned}$$

Wenn wir  $c$  hinreichend groß wählen, folgt

$$2^l (2^{(\frac{c-1}{2}+1) \cdot l} + 2^{-l+1} + 1) \cdot (2^{-c \cdot l} - 1) < 2^{-l}. \quad (10)$$

Die Größe von  $c$  werden wir später bestimmen.

Bevor zum Beweis der Sicherheit des Verfahrens kommen wollen wir noch mal

kurz zusammenfassen, was es bedeutet, dass ein Algorithmus zwischen einer Verteilung und einer Menge von Verteilungen unterscheiden kann. Ein probabilistischer Polynomialzeitalgorithmus  $\mathcal{B}$  kann zwischen einer Verteilung  $U$  und einer Menge von Verteilungen  $\Upsilon$  unterscheiden, wenn er jede Verteilung  $X \in \Upsilon$  mit nicht zu vernachlässigender Wahrscheinlichkeit von  $U$  unterscheiden kann.

**Lemma 9.13** *Für  $c_N \geq 2c_h$  und ausreichend großes  $c_m$  gilt: Angenommen es existiert ein Polynomialzeitalgorithmus  $\mathcal{A}$ , der Verschlüsselungen der 0 und 1 unterscheiden kann, dann existiert ein Algorithmus  $\mathcal{B}$ , der zwischen  $U$  und  $\Upsilon_{\sqrt{n}\gamma(n)}$  unterscheiden kann.*

Beweis: Der Algorithmus  $\mathcal{A}$  gibt nach Übergabe des öffentlichen Schlüssels  $(a_1, a_2, \dots, a_n)$  und eines Wertes  $\omega \in \{0, 1, 2, 3, \dots, N-1\}$  Null oder Eins aus. Sei  $p_0$  die Wahrscheinlichkeit, dass der Algorithmus 1 ausgibt unter der Voraussetzung das  $\omega$  eine Verschlüsselung der 0 war.  $p_0$  hängt von der Wahl des öffentlichen und des privaten Schlüssels sowie vom Verschlüsselungsalgorithmus ab. Sei  $p_1$  analog für den Fall definiert, dass  $\omega$  eine Verschlüsselung der Eins ist und  $p_u$  falls  $\omega$  zufällig aus der Menge  $\{0, 1, 2, 3, \dots, N-1\}$  gewählt wurde. Da  $\mathcal{A}$  zwischen Verschlüsselungen der 0 und der 1 unterscheiden kann, muss  $|p_0 - p_1|$  größer als  $\frac{1}{n^c}$  für eine Konstante  $c \geq 0$  sein (siehe Bemerkung 5.9). Als erstes wollen wir einen Algorithmus  $\tilde{\mathcal{A}}$  konstruieren, der unterscheiden kann, ob  $\omega$  eine Verschlüsselung der 0 ist oder zufällig gewählt wurde.

Die Dreiecksungleichung ist der Grund für die Korrektheit folgender Ungleichung:

$$|p_0 - p_u| + |p_1 - p_u| \geq |p_0 - p_1| \geq \frac{1}{n^c}.$$

Dies bedeutet, dass  $|p_0 - p_u| \geq \frac{2}{n^c}$  oder  $|p_1 - p_u| \geq \frac{2}{n^c}$  gelten muss. Im ersten Fall ist  $\mathcal{A}$  der gesuchte Algorithmus. Im zweiten Fall konstruieren wir  $\tilde{\mathcal{A}}$  folgendermaßen: Die Eingabe von  $\tilde{\mathcal{A}}$  ist  $((a_1, a_2, \dots, a_m, i_0), \omega)$ .  $\tilde{\mathcal{A}}$  ruft  $\mathcal{A}$  mit  $((a_1, a_2, \dots, a_m, i_0), \omega + \lfloor \frac{a_{i_0}}{2} \rfloor)$  auf. Dies macht Verschlüsselungen der 0 zu Verschlüsselungen der 1 und umgekehrt. Sind die Werte gleichverteilt, so sind sie es auch nach der Addition. Also ist  $\tilde{\mathcal{A}}$  der gesuchte Algorithmus.

Sei  $p_0(a_1, \dots, a_m, i_0)$  die Wahrscheinlichkeit, dass  $\tilde{\mathcal{A}}$  Eins ausgibt, wobei der Schlüssel fix und  $\omega$  eine Verschlüsselung der 0 ist. Analog sei  $p_u(a_1, \dots, a_m, i_0)$  für zufällig gewähltes  $\omega \in \{0, 1, 2, \dots, N-1\}$  definiert. Sei  $Y$  die Menge aller Schlüssel  $(a_1, \dots, a_m, i_0)$ , für die  $|p_0(a_1, \dots, a_m, i_0) - p_u(a_1, \dots, a_m, i_0)| \geq \frac{1}{4n^c}$  gilt.

Als nächstes wollen wir untersuchen, mit welcher Wahrscheinlichkeit ein zufällig gewählter Schlüssel in  $Y$  ist. Im folgenden soll  $k$  den öffentlichen Schlüssel

$(a_1, \dots, a_m, i_0)$  bezeichnen. Es gilt

$$\begin{aligned} \frac{1}{2n^c} &\leq |p_0 - p_u| \\ &= |P(k \in Y) \cdot (p_0(k) - p_u(k)) + P(k \notin Y) \cdot (p_0(k) - p_u(k))| \\ &\leq |P(k \in Y)(p_0(k) - p_u(k))| + \frac{1}{4n^c} P(k \notin Y) \\ &\leq P(k \in Y) + \frac{1}{4n^c}. \end{aligned}$$

Also ist die Wahrscheinlichkeit, dass ein zufällig gewählter Schlüssel in  $Y$  liegt, mindestens  $\frac{1}{4n^c}$ .

Nun beschreiben wir den Algorithmus  $\mathcal{B}$ . Es ist eine Verteilung  $R$  gegeben, die entweder  $U$  oder ein  $T_{h,\beta} \in \mathcal{T}_{\sqrt{n}\gamma(n)}$  ist.  $\mathcal{B}$  soll mit nicht zu vernachlässigender Wahrscheinlichkeit die Verteilungen  $U$  und  $T_{h,\beta}$  unterscheiden können. Als erstes wählen wir ein  $\tilde{h}$  gleichverteilt aus der Menge  $\{1, 2, 4, 8, \dots, \sqrt{N}\}$  aus. Darüber hinaus wählen wir  $\delta$  gleichverteilt aus der Menge  $[\frac{\sqrt{N}}{\tilde{h}}, 4\frac{\sqrt{N}}{\tilde{h}})$  und  $s$  ebenfalls gleichverteilt aus  $[0, 7\frac{1}{(\gamma(n))^2})$ . Nun betrachten wir die Verteilung  $R' = C_\delta(R + Q_{\delta^2 s/N} \bmod 1)$ . Wir bilden  $m$   $[N \cdot R']$ -verteilte Werte  $a_1, \dots, a_m$  und wählen zufällig ein  $i_0 \in [m]$ . Jetzt werden die beiden Wahrscheinlichkeiten  $p_0(a_1, \dots, a_m, i_0)$  und  $p_1(a_1, \dots, a_m, i_0)$  durch Berechnen ausreichend vieler Werte Verschlüsselungen  $\omega$  angenähert. Durch polynomiell viele Schritte lassen sich  $p_0$  und  $p_u$  bis auf einen Fehler, der höchstens  $\frac{1}{32n^c}$  groß ist, annähern. Unterscheiden sich die angenäherten Wahrscheinlichkeiten für  $p_0$  und  $p_u$  um mehr als  $\frac{1}{4n^c}$ , so gibt  $\mathcal{B}$  Eins aus, ansonsten Null.

Als erstes wollen wir zeigen, dass  $\mathcal{B}$  Null ausgibt, falls  $R = U$  war. In diesem Fall ist  $R'$  ebenfalls die Gleichverteilung auf  $[0, 1]$ . Also sind die Werte  $a_1, a_2, \dots, a_m$  ebenfalls gleichverteilt aus der Menge  $\{0, 1, 2, \dots, N-1\}$ . Wählt man die Anzahl  $m = c \cdot l$  der Elemente wie in Lemma 9.8, so ist sichergestellt, dass die Verteilung der Werte, die wir durch Verschlüsselungen der Null erhalten, exponentiell nah bei der Gleichverteilung ist. Damit ist  $|p_0(a_1, \dots, a_m, i_0) - p_1(a_1, \dots, a_m, i_0)|$  exponentiell klein und  $\mathcal{B}$  gibt Null aus.

Angenommen  $R = T_{h,\beta}$  für ein  $h \leq 2^{c_h n^2}$  und eine reelle Zahl  $\beta \in \frac{1}{(\gamma(n))^2} [1, 4)$ . Es ist zu zeigen, dass  $\mathcal{B}$  mit nicht zu vernachlässigender Wahrscheinlichkeit Eins ausgibt. Das heißt, wir müssen zeigen, dass die Werte  $a_1, a_2, \dots, a_m$  für ein  $i_0$  mit einer Wahrscheinlichkeit in der Größenordnung  $\frac{1}{poly(n)}$  ein öffentlicher Schlüssel sind.

Hierfür müssen wir untersuchen, wann  $R'$  einen korrekten Schlüssel erzeugt. Nach Lemma 9.4 ist  $T_{h,\beta} + Q_{\delta^2 s/N} \bmod 1 = T_{h,\beta+(\delta h)^2 s/N}$ . Da  $R'$  die Stauchung dieser Funktion ist, gilt nach Lemma 9.6  $R' = T_{\delta h, \beta+(\delta h)^2 s/N}$ .

Damit diese Funktion einen korrekten Schlüssel erzeugt müssen folgende Bedingungen erfüllt sein:

1.  $\delta h$  muss im Intervall  $[\sqrt{N}, 2\sqrt{N})$  sein
2.  $\text{frc}(\delta h) < \frac{1}{16m}$
3.  $\beta + (\delta h)^2 s/N \in [\frac{1}{(\gamma(n))^2}, 4\frac{1}{(\gamma(n))^2})$
4. Damit die  $a_i = \lfloor z_i \rfloor = \frac{x_i + y_i}{h}$  einen korrekten öffentlichen Schlüssel bilden,

muss der Wert  $x_{i_0} \in \{0, 1, \dots, h-1\}$  ungerade Periode liegt. Der Index  $i_0$  muss so gewählt sein, dass der Wert  $a_{i_0}$  diese Bedingung erfüllt.

Wir müssen also zeigen, dass diese 4 Fälle gleichzeitig mit einer Wahrscheinlichkeit  $\frac{1}{\text{poly}(n)}$  auftreten.

(1)  $\tilde{h}$  ist aus der Menge  $\{1, 2, 4, \dots, \sqrt{N}\}$  und  $h$  ist nicht größer als  $\sqrt{N}$ . Also gilt

$$P(h \leq \tilde{h} \leq 2h) = \frac{1}{\log(\sqrt{N})} = \frac{2}{c_N n^2}.$$

In diesem Fall ist  $\delta h$  gleichmäßig im Intervall  $[\frac{h}{h}\sqrt{N}, 4\frac{h}{h}\sqrt{N})$  verteilt. Dieses Intervall enthält die Menge  $[\sqrt{N}, 2\sqrt{N})$ . Es gilt

$$\begin{aligned} P(\delta h \in [\sqrt{N}, 2\sqrt{N})) &= \frac{\sqrt{N}}{4\frac{h}{h}\sqrt{N} - \frac{h}{h}\sqrt{N}} \\ &= \frac{1}{3\frac{h}{h}} \\ &\geq \frac{1}{3}. \end{aligned}$$

Also tritt mit einer Wahrscheinlichkeit von mindestens  $\frac{1}{3} \frac{2}{c_N}$  die Bedingung (1) ein.

(2) Alle Werte für  $\delta h \in [\sqrt{N}, 2\sqrt{N})$  sind gleich wahrscheinlich. Also ist die Wahrscheinlichkeit für  $\text{frc}(\delta h) < \frac{1}{16m}$  unter der Bedingung, dass (1) erfüllt ist,  $\frac{1}{8m}$ .

(3) Angenommen (1) und (2) sind erfüllt. Dann gilt:

- Die Zahl  $\beta + (\delta h)^2 s/N$  ist für einen festen Faktor  $\delta h$  gleichmäßig im Intervall  $[\beta, \beta + (\delta h)^2/N \cdot \frac{7}{(\gamma(n))^2})$  verteilt.
- Aus  $\delta h \in [\sqrt{N}, 2\sqrt{N})$  folgt, dass  $(\delta h)^2/N$  im Intervall  $[1, 4)$  liegen muss.

Der Bereich in dem  $\beta + (\delta h)^2 s/N$  liegt ist also höchstens  $4 \cdot \frac{7}{(\gamma(n))^2}$  lang. Wegen  $\beta \in [\frac{1}{(\gamma(n))^2}, 4\frac{1}{(\gamma(n))^2})$  enthält er immer das Intervall  $[4\frac{1}{(\gamma(n))^2}, 8\frac{1}{(\gamma(n))^2})$ . Also ist die Wahrscheinlichkeit, für das Ereignis  $\beta + (\delta h)^2 s/N \in [4\frac{1}{(\gamma(n))^2}, 8\frac{1}{(\gamma(n))^2})$  unter der Voraussetzung, dass (1) und (2) erfüllt sind mindestens  $\frac{4}{28}$ . Insgesamt ergibt sich eine Wahrscheinlichkeit von  $\frac{1}{3} \cdot \frac{2}{c_N n^2} \cdot \frac{1}{7} \cdot \frac{1}{8m} = \frac{1}{\text{poly}(n)}$  dafür, dass alle drei Ereignisse auf einmal auftreten.

(4) Sei  $i_0$  zufällig gewählt. Man beachte, dass  $a_{i_0} = \frac{x_{i_0} + y_{i_0}}{2}$ . Die Wahrscheinlichkeit dafür, dass  $x_{i_0}$  ungerade ist, ist  $\frac{1}{2}$ .

Also ist  $\mathcal{B}$  ein Algorithmus, der mit nicht zu vernachlässigender Wahrscheinlichkeit zwischen  $U$  und  $\Upsilon_{\sqrt{n}\gamma(n)}$  unterscheiden kann.

## Die Größe der Konstante $c_m$

Die Konstante  $c_m$  wird in Lemma 9.13 spezifiziert. Sie bestimmt die Anzahl der zur Erzeugung des öffentlichen Schlüssels zu generierenden Werte  $a_i$ , d.h. die

Länge des öffentlichen Schlüssels. In dem Beweis des Lemmas wird ein Algorithmus  $\mathcal{B}$  beschrieben, der mit Hilfe eines Algorithmus  $\mathcal{A}$ , der Verschlüsselungen der Null und der Eins unterscheiden kann, die Gleichverteilung  $U$  von der Verteilung  $T_{h,\beta}$  unterscheiden kann. Dazu generiert er Werte  $(a_1, a_2, \dots, a_m)$ , die entweder gleichverteilt oder mit nicht zu vernachlässigender Wahrscheinlichkeit zusammen mit einem Index  $i_0$  einen korrekten öffentlichen Schlüssel bilden.

Der Beweis erfolgt in mehreren Schritten.

- Der Algorithmus  $\mathcal{A}$  wird so modifiziert, dass er Verschlüsselungen der Null von gleichverteilten Werten unterscheiden kann.
- Aus dem Index  $i_0$  und den Werten  $(a_1, a_2, \dots, a_m)$ , die entweder gleichverteilt sind oder einen öffentlichen Schlüssel bilden, werden Verschlüsselungen der Null erzeugt.
- Mit Hilfe von  $\mathcal{A}$  kann die Gleichverteilung von der Verteilung  $T_{h,\beta}$  unterschieden werden.

Der Algorithmus  $\mathcal{A}$  kann Verschlüsselungen der Null von gleichmäßig verteilten Werten unterscheiden. Damit man dies ausnutzen kann, muss der öffentliche Schlüssel so groß gewählt werden, dass die Verteilung, die durch Summation über zufällig gewählte Teilmengen von gleichverteilten Werten  $(a_1, a_2, \dots, a_m)$  entsteht, nicht von der Gleichverteilung zu unterscheiden ist. Nach Lemma 9.8 gibt es eine Konstante  $c$  so, dass der statistische Abstand zwischen der Gleichverteilung auf der Menge  $M = \{1, \dots, 2^l - 1\}$  und der Summe modulo  $2^l$  von zufälligen Teilmengen von  $c \cdot l$  gleichverteilten Werten  $a_i \in M$  exponentiell klein ist. Beachte, dass  $N = 2^{c_N n^2}$ . Also ist die Anzahl der Elemente im öffentlichen Schlüssel gerade  $c_N \cdot n^2 \cdot c$ . Die Konstante  $c_N$  haben wir bereits vorher bestimmt. Wir haben in diesem Kapitel gezeigt, dass die Sicherheit des Kryptosystems auf der Schwierigkeit eines *uSVP* reduziert werden kann. Der Sicherheitsparameter  $n$  ist die Dimension des Gitters auf dem dieses Problem gestellt wird.

**Lemma 9.14** *Für die Konstante  $c_m$  gilt*

$$c_m \geq 6.$$

Beweis: Für den Beweis muss nochmal auf den Beweis von Lemma 9.8 eingegangen werden. Es wird die Tschebyschevsche Ungleichung benutzt. Dies führt zu dem Ergebnis, dass die Konstante  $c_m$  so groß gewählt werden muss, dass

$$2^l (2^{(\frac{c_m-1}{2}+1) \cdot l} + 2^{-l+1} + 1) \cdot (2^{-c_m \cdot l} - 1) \leq 2^{(\frac{-c_m}{2}+1.5) \cdot l} + 2^{-c_m l+1} + 2^{-(c_m-1)l} < 2^{-l}$$

gilt.

Setzt man  $c_m = 5$  erhält man:

$$2^{-l} + 2^{-5l+1} + 2^{-4l} > 2^{-l}$$

Für  $c_m = 6$  erhält man:

$$2^{-1.5l} + 2^{-6l+1} + 2^{-5l} < 2^{-l}.$$

Also muss  $c_m$  mindestens 6 sein.

**Theorem 9.15** *Für  $c_N = 2c_h = 8$  und  $c_m = 6$  gilt. Das Kryptosystem aus Teil 6.2 macht Entschlüsselungsfehler mit zu vernachlässigender Wahrscheinlichkeit und seine Sicherheit basiert auf dem  $\sqrt{n} \cdot \gamma(n)$ -uSVP.*

Beweis folgt direkt aus Lemma 9.1, 9.13 und 9.14.

## 10 Chosen ciphertext Attacken auf das Regev-Kryptosystem

Ein großer Nachteil des Regev Kryptosystems ist seine aus der Symmetrie der Funktion  $T_{h,\beta}$  resultierende Symmetrie. Diese ist periodisch mit Periode  $\frac{1}{h}$ . Das die Symmetrie von  $T_{h,\beta}$  erhalten bleibt, liegt an der Konstruktion des öffentlichen Schlüssel. Bei dessen Generierung werden  $T_{h,\beta}$ -verteilte Werte mit einer großen Zahl multipliziert. Die Symmetrie der Verteilung ändert sich dabei nicht. „Täler“ der Verteilung werden bei der Entschlüsselung als eine Verschlüsselung der Eins interpretiert, Buckel als Verschlüsselungen der Null. Für einen chosen ciphertext Angriff kann man sich diese Symmetrie zunutze machen, indem man versucht die Grenze zu finden an der Verschlüsselungen der Eins zu Verschlüsselungen der Null werden. Aus Laufzeitgründen ist es am besten, hierbei von der Null ausgehend die erste Grenze zu finden, da dann eine Grenze genügt, um die Periode bestimmen zu können. Wenn man diese kennt, kann man Verschlüsselungen der Eins von Verschlüsselungen der Null unterscheiden.

Ein von Izmerley in [Izm] vorgestellter adaptiver chosen ciphertext Angriff macht sich diese Tatsache folgendermaßen zunutze. Ausgehend von einem Wert  $\sigma(1)$  in einer Periode sucht man einen Wert  $\sigma(2)$  in einer benachbarten Periode. Hat man diesen gefunden, wendet man ein Intervallhalbierungsverfahren an. Dafür wird der Mittelwert  $m$  der beiden Werte gebildet. Ist der Klartext zu  $m$  gleich dem Klartext von  $\sigma(1)$ , so wird diese Grenze verändert, sonst die andere. Hat man Werte in zwei beieinander liegenden Perioden in Zeit  $O(\log N)$  gefunden, so hat dieses Verfahren offensichtlich Laufzeit  $O(\log N)$ .

Werte aus benachbarten Perioden findet man auf folgende Weise in  $O(\log N)$  arithmetischen Schritten. Gegeben sei ein Wert  $a$  in der Periode  $\sigma(1)$ . Auf  $a$  wird eine kleine Zahl  $b$  addiert. Ist die Entschlüsselung von  $a + b$  gleich der Entschlüsselung von  $a$ , so wird die Zahl  $b$  verdoppelt und dieser Schritt wiederholt. Das wird so lange wiederholt bis man einen Wert mit anderem Klartext gefunden hat. Da die Perioden, in denen Schlüsseltexte als eins entschlüsselt werden genauso lang sind wie Perioden in denen Schlüsseltexte als Null entschlüsselt werden, findet man auf diese Weise einen Wert  $c = a + b$  in einer benachbarten Periode  $\sigma(2)$ . Bei praktischen Untersuchungen hat sich eine Laufzeit von ungefähr  $2 \log N$  ergeben.

Wir haben einen nicht adaptiven chosen ciphertext Angriff mit konstanter Laufzeit auf den geheimen Schlüssel des Oded Regev Kryptosystems gefunden. Der Schlüssel liegt im Intervall  $[\sqrt{N}, 2\sqrt{N})$  mit  $N = 2^{c_N n^2}$ . Bei dem Verfahren wird in jedem Schritt ein Bit des geheimen Schlüssels gefunden. Da man das erste Bit bereits kennt ist die Laufzeit

$$\log(\sqrt{N}) = \frac{\log(N)}{2}.$$

Wir werden den Algorithmus für gerade ganzzahlige Exponenten  $c_N n^2$  vorstellen.  $\sqrt{N}$  ist dann eine gerade Zahl. Der Algorithmus bestimmt eine Zahl  $k$  mit  $h = k\sqrt{N}$ . Die Zahl  $k$  liegt im Intervall  $[1, 2)$ . Um eine Idee für die Wirkungsweise des Algorithmus zu bekommen, zunächst ein kleines Beispiel.

**Beispiel 10.1** Sei  $w = \frac{\sqrt{N}}{2}$ . Dann ist  $\text{frc}(\frac{w}{d}) = \text{frc}(\frac{\sqrt{N}}{2} \frac{k \cdot \sqrt{N}}{N}) = \text{frc}(\frac{k}{2})$ . Ist  $k \in [1, \frac{3}{2}]$ , dann ist  $\text{frc}(\frac{k}{2}) \geq \frac{1}{4}$  und der Klartext von  $w$  ist Eins. Ansonsten ist  $\text{frc}(\frac{k}{2}) < \frac{1}{4}$  und der Klartext ist 0. Wenn man den zu  $w$  gehörigen Klartext kennt, kann man entscheiden in welcher Hälfte des Intervalls  $k$  liegt. Wie wir später sehen werden, können wir mit dieser Information entscheiden welchen Wert das erste Bit von  $k$  und damit  $h$  hat.

Da  $\sqrt{N}$  eine Zweierpotenz ist, ist die Division durch  $\sqrt{N}$  einfach ein Shift um  $\log(\sqrt{N})$  viele Stellen nach rechts. Im Intervall von  $[\sqrt{N}, 2\sqrt{N})$  liegen  $\sqrt{N}$  viele Zahlen. Also hat  $k$  die Darstellung

$$k = 1 + \sum_{i=1}^{\log(\sqrt{N})} k_i 2^{-i}.$$

Nun wollen wir den Algorithmus zur Berechnung von  $k$  vorstellen. Hierfür benötigen wir zunächst ein Entschlüsselungsrakel.

**Definition 10.2** Sei  $r \in \mathbb{N}$  mit  $r < N - 1$ . Dann gibt das Orakel  $\mathcal{O}(r)$  Eins aus, wenn der zu  $r$  gehörige Klartext Eins ist und Null sonst.

### Algorithmus 10.3

**Eingabe:** Werte des Entschlüsselungsrakel  $\mathcal{O}$  und die Zahl  $\sqrt{N}$ .

1: Setze  $k_0 = 1$ . /  $k$  ist eine Zahl im Intervall  $[1, 2)$

2: **for**  $i = 1, 2, \dots, \log(\sqrt{N})$  **do**

3: **if**  $(\mathcal{O}(2^{i-2}\sqrt{N}) = 1$  **and**  $k_{i-1} = 1)$

4: Setze  $k_i = 0$ .

5: **end if**

6: **if**  $(\mathcal{O}(2^{i-2}\sqrt{N}) = 0$  **and**  $k_{i-1} = 1)$

7: Setze  $k_i = 1$ .

8: **end if**

9: **if**  $(\mathcal{O}(2^{i-2}\sqrt{N}) = 1$  **and**  $k_{i-1} = 0)$

10: Setze  $k_i = 1$ .

11: **end if**

12: **if**  $(\mathcal{O}(2^{i-2}\sqrt{N}) = 0$  **and**  $k_{i-1} = 0)$

13: Setze  $k_i = 0$ .

14: **end if**

15: **end for**

**Ausgabe:** Eine Bitfolge  $k_0, k_1, \dots, k_{\log(\sqrt{N})}$  mit  $h = (\sum_{i=0}^{\log(\sqrt{N})} k_i 2^{-i})\sqrt{N}$  oder

$h = (\sum_{i=0}^{\log(\sqrt{N})} k_i 2^{-i})\sqrt{N} + 1$ .

Beweis der Wohldefiniertheit des Algorithmus:

Wegen  $2^{i-2}\sqrt{N} < 2^{\log(\sqrt{N})}\sqrt{N} = N$  wird das Orakel nur mit zulässigen Werten aufgerufen. Also ist der Algorithmus wohldefiniert.

Beweis der Korrektheit mittels vollständiger Induktion.

(IV): Sei  $n = 1$ .

Dann ist  $k_{n-1} = 1$ .

1.Fall:  $\mathcal{O}(\frac{\sqrt{N}}{2}) = 0$ .

Also gilt

$$\text{frc}(\frac{\sqrt{N}}{2} \frac{k\sqrt{N}}{N}) = \text{frc}(\frac{k}{2}) < \frac{1}{4}.$$

Da  $k$  eine Zahl im Intervall zwischen 1 und 2 ist, muss  $\frac{k}{2}$  im Intervall  $[\frac{1}{2}, 1)$  liegen. Wenn der Abstand von  $\frac{k}{2}$  zur nächsten ganzen Zahl kleiner als  $\frac{1}{4}$  ist, muss  $\frac{k}{2}$  mindestens  $\frac{3}{4}$  sein. Damit gilt

$$\frac{k}{2} = \frac{1}{2} + \frac{k_1}{4} + \frac{k_2}{8} + \dots > \frac{3}{4}.$$

Die Summe  $\frac{1}{4} \sum_{i=1}^n \frac{1}{2^i}$  ist nach Satz 11.9 kleiner als  $\frac{1}{4}$ . Der Koeffizient  $k_1$  muss also Eins sein. Der Algorithmus setzt  $k_1$  also korrekt.

2.Fall:  $\mathcal{O}(\frac{\sqrt{N}}{2}) = 1$ .

In diesem Fall ist  $\text{frc}(\frac{k}{2}) \geq \frac{1}{4}$ . Mit der gleichen Begründung wie eben, muss

$$\frac{1}{2} \leq \frac{k}{2} \leq \frac{3}{4}$$

gelten. Wir müssen zwei Fälle betrachten.

Fall 2.1:  $\frac{k}{2} = \frac{1}{2} + \frac{k_1}{4} + \frac{k_2}{8} + \dots < \frac{3}{4}$

In diesem Fall ist nach Satz 11.9  $k_1 = 0$ . Der erste Koeffizient wird also richtig bestimmt.

Fall 2.2:  $\frac{k}{2} = \frac{3}{4}$ .

Also ist  $k = 1 + \frac{1}{2}$  und das Bit  $k_1$  wurde fälschlicherweise Null gesetzt. Im nächsten Schritt wird dem Orakel der Wert  $\sqrt{N}$  übergeben. Es gilt

$$\text{frc}(\sqrt{N} \frac{k\sqrt{N}}{N}) = \frac{1}{2}.$$

Also gibt das Orakel im nächsten Schritt Eins zurück.

Das Bit  $k_2$  wird Eins gesetzt. In den weiteren Schleifen gibt  $\mathcal{O}$  immer Null zurück, da  $\text{frc}(k \cdot 2^i)$  für  $i > 1$  Null ist. Das bedeutet, dass alle weiteren Bits Eins gesetzt werden. Da die Multiplikation mit  $\sqrt{N}$  nur ein Shift nach links ist, weicht  $h = k\sqrt{N}$  nur um Eins von  $h$  ab.

Falls ein Bit falsch gesetzt wurde, haben wir bereits gezeigt, dass die Ausgabe trotzdem nur um Eins von dem geheimen Schlüssel abweicht. Wir können also annehmen, dass alle bisher bestimmten Bits korrekt sind. Wird allerdings im  $n + 1$ -ten Schritt ein Bit falsch gesetzt, müssen wir zeigen, dass die Ausgabe trotzdem nur um Eins von dem geheimen Schlüssel abweicht.

(IA): Der Algorithmus hat die ersten  $n$  Bits richtig bestimmt.

(IS): Zunächst eine kleine Vorüberlegung. Durch die Multiplikation mit  $2^{n-1}$  werden die Zahlen  $k_i 2^{-i}$  für  $i < n$  ganzzahlig. Also ist

$$\text{frc}(\frac{2^{n-1}\sqrt{N}}{d}) = \text{frc}(2^{n-1}k) = \text{frc}(k_n 2^{-1} + k_{n+1} 2^{-2} + \dots).$$

Es sind vier Fälle zu betrachten.

1.Fall:  $k_n = 0$  und  $\mathcal{O}(\sqrt{N} 2^{n-1}) = 0$ .

Dann ist

$$\text{frc}(2^{n-1}k) < \frac{1}{4}.$$

Also ist

$$\text{frc}(k_n 2^{-1} + k_{n+1} 2^{-2} + \dots) < \frac{1}{4}.$$

Da der  $n$ -te Koeffizient Null war, muss diese Zahl kleiner als  $\frac{1}{4}$  sein. Somit ist der Koeffizient  $k_{n+1}$  null. Der Algorithmus bestimmt das nächste Bit korrekt.

2.Fall:  $k_n = 0$  und  $\mathcal{O}(\sqrt{N} 2^{n-1}) = 1$ .

Dann gilt

$$\text{frc}(2^{n-1}k) \geq \frac{1}{4}$$

Also ist

$$\text{frc}(k_n 2^{-1} + k_{n+1} 2^{-2} + \dots) \geq \frac{1}{4}.$$

Damit der Rest dieser Zahl größer gleich  $\frac{1}{4}$  ist, muss der Koeffizient  $k_{n+1}$  Eins sein, was zu zeigen war.

3.Fall:  $k_n = 1$  und  $\mathcal{O}(\sqrt{N} 2^{n-1}) = 0$ .

Dann ist

$$\text{frc}(k_n 2^{-1} + k_{n+1} 2^{-2} + \dots) < \frac{1}{4}.$$

Wegen  $k_n = 1$  muss diese Zahl größer  $\frac{3}{4}$  sein. Also ist Nächste Bit von  $k$  Eins. Auch in diesem Bit handelt der Algorithmus korrekt.

4.Fall:  $k_n = 1$  und  $\mathcal{O}(\sqrt{N} 2^{n-1}) = 1$ .

Dann ist

$$\text{frc}(k_n 2^{-1} + k_{n+1} 2^{-2} + \dots) \geq \frac{1}{4}.$$

Es müssen wieder zwei Fälle unterschieden werden.

Fall 4.1:  $\text{frc}(k_n 2^{-1} + k_{n+1} 2^{-2} + \dots) > \frac{1}{4}$ .

Also muss die Zahl  $k_n 2^{-1} + k_{n+1} 2^{-2} + \dots$  kleiner als  $\frac{3}{4}$  sein. Damit dies der Fall ist, muss das Bit  $k_{n+1}$  null sein. Der Algorithmus bestimmt das Bit also korrekt.

Fall 4.2:  $\text{frc}(k_n 2^{-1} + k_{n+1} 2^{-2} + \dots) = \frac{1}{4}$ .

$$\Rightarrow k_n 2^{-1} + k_{n+1} 2^{-2} + \dots = \frac{3}{4}$$

Also muss  $k_{n+1}$  Eins sein. Alle folgenden Bits sind Null. Der Algorithmus setzt in diesem Fall das  $n + 1$ -te Bit falsch. Für  $i = n + 2$  gilt,

$$\text{frc}(2^n k) = \text{frc}\left(\frac{k_{n+1}}{2}\right) = \text{frc}\left(\frac{1}{2}\right) = \frac{1}{2}.$$

Also gibt das Orakel im nächsten Schritt eins aus. Das nächste Bit  $k_{n+2}$  wird somit Eins gesetzt. Ab jetzt gibt das Orakel in jedem Schritt Null zurück. Das bedeutet, dass alle weiteren Bits 1 sind. Damit ist die Aussage bewiesen.

**Lemma 10.4** *Der Algorithmus 10.3 benötigt die Klartexte zu  $\log(\sqrt{N})$  vielen Werten, um den geheimen Schlüssel zu bestimmen. Die Laufzeit ist  $\log(\sqrt{N})$ .*

Neben den Vorteilen der verbesserten Laufzeit und der Nichtadaptivität des eben vorgestellten Algorithmus, ist ein weiterer Vorteil gegenüber dem Angriff von Izmerley, dass man sofort für die Größe des geheimen Schlüssels signifikante Bits erhält. Dies liegt daran, dass die Bits des geheimen Schlüssels von links nach rechts bestimmt werden.

Um diesen Angriff auch auf Schlüssel durchzuführen deren Wurzel keine ganze Zahl ist, braucht man eine gute Näherung für  $\sqrt{N}$ . Das Orakel wird dann mit nah an diesem Wert liegenden Werten aufgerufen. Ist der Abstand zur nächsten ganzen Zahl zu „groß“, dann lässt man sich für zwei Zahlen den Klartext bestimmen. Sind diese gleich, so kann man den Algorithmus fortsetzen, ansonsten hat man bereits eine gute Näherung für  $h$  gefunden.

## 11 Grundlagen

Im ersten Teil werden wir einige grundlegende Ergebnisse aus verschiedenen mathematischen Gebieten und der Kryptographie zusammenfassen.

**Theorem 11.1 (Satz von Euklid)** *Seien  $a, b$  ganze Zahlen, dann kann man mit dem euklidischen Algorithmus zwei ganze Zahlen  $x, y$  mit,*

$$\text{ggT}(a, b) = ax + by$$

*finden.*

Beweis siehe [Buch 1999]

### 11.1 Reihen

Wir benötigen noch einige Grundlagen aus der Analysis über Reihen und deren Grenzwerte. Wir setzen voraus, dass bekannt ist, was eine Reihe ist was Konvergenz bedeutet und wie der Grenzwert einer Reihe definiert ist.

**Definition 11.2 (Reihe)** *Sei  $(a_n)_{n \in \mathbb{N}}$  eine Folge reeller Zahlen. Die Folge*

$$s_n := \sum_{k=0}^n a_k, \quad n \in \mathbb{N}$$

*heißt Reihe und wird mit  $\sum_{k=0}^{\infty} a_k$  bezeichnet.*

**Definition 11.3 (Umordnung von Reihen)** *Sei  $\sigma : \mathbb{N}_0 \rightarrow \mathbb{N}_0$  bijektiv und  $b_k = a_{\sigma(k)}$ .*

*Dann heißt die Reihe  $\sum_{i=1}^{\infty} b_i$  Umordnung der Reihe  $\sum_{i=1}^{\infty} a_i$ .*

Es stellt sich die Frage, ob sich der Grenzwert einer Reihe durch Umordnung verändert. Das verblüffende Ergebnis ist, dass eine Reihe, die konvergiert, aber nicht absolut konvergiert durch geschickte Umordnung gegen jede beliebige reelle Zahl konvergiert. Die absolut konvergenten Reihen sind die Reihen deren Grenzwert invariant unter Umordnung ist.

**Definition 11.4 (absolut konvergent)** *Eine Reihe  $\sum_{i=1}^{\infty} a_i$  heißt absolut konvergent, wenn die Reihe  $\sum_{i=1}^{\infty} |a_i|$  konvergiert.*

**Satz 11.5** *Sei  $\sum_{i=1}^{\infty} a_i$  eine absolut konvergente Reihe. Dann ist auch jede Umordnung  $\sum_{i=1}^{\infty} a_{\sigma(i)}$  absolut konvergent und es gilt*

$$\sum_{i=1}^{\infty} a_i = \sum_{i=1}^{\infty} a_{\sigma(i)}.$$

Beweis siehe [Bar 1987] Seite 152.

## 11.2 Funktionenfolgen

**Definition 11.6** Sei  $K$  eine Menge und seien  $f_n : K \rightarrow \mathbb{C}, n \in \mathbb{N}$ , Funktionen.

(i) Die Folge  $(f_n)$  konvergiert punktweise gegen eine Funktion  $f : K \rightarrow \mathbb{C}$ , wenn für jedes  $x \in K$  die Folge  $(f_n(x))_n \in \mathbb{N}$  gegen  $f(x)$  konvergiert, d.h.

$$(\forall x \in K)(\forall \epsilon > 0)(\exists N)(\forall n \geq N) |f_n(x) - f(x)| < \epsilon.$$

(ii) Die Folge  $(f_n)$  konvergiert gleichmäßig gegen eine Funktion  $f : K \rightarrow \mathbb{C}$ , wenn

$$(\forall \epsilon > 0)(\exists N)(\forall x \in K)(\forall n \geq N) |f_n(x) - f(x)| < \epsilon.$$

**Satz 11.7** Seien  $f_n : [a, b] \rightarrow \mathbb{R}, n \in \mathbb{N}$  stetig differenzierbare Funktionen, die punktweise gegen die Funktionen  $f : [a, b] \rightarrow \mathbb{R}$  konvergieren. Die Folge der Ableitungen  $f'_n : [a, b] \rightarrow \mathbb{R}$  konvergiere gleichmäßig. Dann ist  $f$  differenzierbar und es gilt für alle  $x \in [a, b]$

$$f'(x) = \lim_{n \rightarrow \infty} f'_n(x).$$

Beweis:siehe [For 1983].

## 11.3 g-adische Entwicklung

**Satz 11.8 (Geometrische Summenformel)** Sei  $q \neq 1$ . Dann gilt für alle natürliche Zahlen:

$$\sum_{i=0}^n q^i = \frac{1 - q^{n+1}}{1 - q}.$$

Beweis siehe [Bar 1987] Seite 27.

**Satz 11.9 (g-adische Darstellung)** Sei  $g \geq 2$  eine natürliche Zahl. Dann besitzt jede reelle Zahl  $x$  eine g-adische Entwicklung.

$$x = \lfloor x \rfloor + \sum_{n=1}^{\infty} \frac{x_n}{g^n},$$

mit  $x_n \in \{0, 1, 2, \dots, g-1\}$ .

siehe[Bar 1987]

**Folgerung 11.10** Sei  $x \in [0, 1)$  eine reelle Zahl und  $g \geq 2$ . Dann existiert für jede natürliche Zahl  $n$  eine Folge  $a_1, \dots, a_{n-1}$  mit  $a_i \in \{0, 1, 2, \dots, g-1\}$  und eine Zahl  $a \in [0, g)$ , so dass

$$x = \sum_{i=1}^{n-1} \frac{a_i}{g^i} + \frac{a}{g^n}.$$

Beweis: Aus Satz 11.9 folgt, dass man jede reelle Zahl  $x$  aus dem Intervall  $[0, 1)$  als Reihe

$$x = \sum_{i=1}^{\infty} \frac{a_i}{g^i}$$

mit  $a_i \in \{0, 1, 2, \dots, g-1\}$  darstellen kann. Also lässt sich  $x$  auch durch die endliche Summe

$$x = \sum_{i=1}^{n-1} \frac{a_i}{g^i} + d$$

mit  $d = \sum_{i=n}^{\infty} \frac{a_i}{g^i}$  darstellen. Setzt man  $a = g^n d$  so erhält man die Aussage.

## 11.4 Public-Key Kryptosysteme und Sicherheitsbegriffe

Der Unterschied Public-Key Kryptosystems zu symmetrischen Verschlüsselungsverfahren ist, dass zur Kommunikation kein gemeinsamer Schlüssel ausgetauscht werden muss.

**Definition 11.11** Ein Public-Key Kryptosystem (PKC) ist ein 5-Tupel von Mengen  $\{N, K_p, K_s, M, C\}$  und ein 3-Tupel  $\{K, E_{k_p}, D_{k_s}\}$  von Algorithmen, mit

- $K$  ist ein probabilistischer Schlüsselerzeugungsalgorithmus, der mit Eingabe eines Sicherheitsparameters  $n \in N$  einen öffentlichen Schlüssel  $k_p \in K_p$  und einen privaten Schlüssel  $k_s \in K_s$  erzeugt.
- $E_{k_p}$  ist ein deterministischer Verschlüsselungsalgorithmus, der mit der Eingabe eines öffentlichen Schlüssel  $k_p$  und einer Nachricht aus dem Raum der Klartexte  $M$  einen Schlüsseltext  $c$  aus dem Schlüsseltextraum  $C$  ausgibt.
- $D_k$  ist ein deterministischer Entschlüsselungsalgorithmus, der mit der Eingabe von  $s_k, c$  den zu  $c$  gehörigen Klartext  $m$  zurückgibt.

**Definition 11.12** Ein probabilistisches PKC ist ein 6-Tupel von Mengen  $\{N, K_p, K_s, M, R, C\}$  und ein 3-Tupel von Algorithmen  $\{K, E_{k_p}, D_{k_s}\}$  mit

- $K$  ist ein probabilistischer Schlüsselerzeugungsalgorithmus, der mit Eingabe eines Sicherheitsparameters  $n \in N$  einen öffentlichen Schlüssel  $k_p \in K_p$  und einen privaten Schlüssel  $k_s \in K_s$  erzeugt.
- $E_{k_p}$  ist ein deterministischer Verschlüsselungsalgorithmus, der mit der Eingabe eines öffentlichen Schlüssel  $k_p$  und einer Nachricht aus dem Raum der Klartexte  $M$  einen Schlüsseltext  $c$  aus dem Schlüsseltextraum  $C$  ausgibt.
- $D_k$  ist ein deterministischer Entschlüsselungsalgorithmus, der mit der Eingabe von  $s_k, c$  mit nicht zu vernachlässigender Wahrscheinlichkeit den zu  $c$  gehörigen Klartext  $m$  zurückgibt.

Es wurden verschiedene Sicherheitsbegriffe für Kryptosysteme entwickelt, von denen einige speziell für probabilistische PKC definiert wurden. Wichtig ist dabei die Rolle eines Angreifer. Dieser könnte versuchen, sich neben der Beschreibung des PKC und der Erzeugung von Schlüsseltexten („Ciphertext only Attack“) an weitere Informationen zu gelangen. Ein realistischer Angriff ist der Versuch sich zu bestimmten Schlüsseltexten ohne Kenntnis des Schlüssel

Klartexte zu beschaffen („chosen ciphertext Attacke“). Mit Hilfe dieser Informationen versucht er sich Informationen über den Schlüssel zu beschaffen.

**Definition 11.13** (*Non-adaptive chosen-ciphertext-Attacke*)

Ein Angreifer hat Zugriff auf ein Entschlüsselunsorakel. Dieses gibt ihm zu vorher gewählten Schlüsseltexten, die dazugehörigen Klartexte.

**Definition 11.14** (*Adaptive chosen-ciphertext-Attacke*)

Ein Angreifer hat Zugriff auf ein Entschlüsselunsorakel. Er darf die übergebenen Schlüsseltexte in Abhängigkeit der erhaltenen Klartexte wählen.

Ein Kryptosystem gilt als unsicher, wenn es einem Angreifer in angemessener Zeit den privaten Schlüssel teilweise oder vollständig zu ermitteln oder ohne Kenntnis des Schlüssel Informationen über einen Klartext zu erlangen.

**Definition 11.15** Ein Kryptosystem heißt perfekt sicher, wenn die Wahrscheinlichkeit, dass ein bestimmter Schlüsseltext  $c$  und ein Klartext  $m$  auftritt unabhängig sind.

**Definition 11.16** Ein Kryptosystem heißt beweisbar sicher, wenn gezeigt werden kann, dass ein Angreifer der zu Schlüsseltexten die zugehörigen Klartexte bestimmen kann, auch ein schwieriges mathematisches Problem lösen kann.

Für die probabilistischen Kryptosysteme wurden die zueinander äquivalenten Sicherheitsbegriffe der semantischen und der polynomiellen Sicherheit entwickelt. Diesem Sicherheitsbegriff liegt zugrunde, dass die Ressourcen eines Angreifer polynomiell beschränkt sind. Ob dies sinnvoll ist, wurde bereits in Kapitel 5 diskutiert. Er wird deshalb mit einem probabilistischen Polynomzeitalgorithmus beschrieben.

**Definition 11.17** Ein PKC heißt polynomiell sicher, wenn es keinem passiven Angreifer möglich ist die zu den Klartexten  $m_1$  und  $m_2$  Schlüsseltexte mit einer nicht zu vernachlässigend größeren Wahrscheinlichkeit als  $\frac{1}{2}$  zu unterscheiden.

Wenn ein Angreifer einen Text verschlüsselt kennt er dessen Schlüsseltext. Da der öffentliche Schlüssel bekannt ist, kann er dies immer tun. Ein PKC ist also erstmal nicht sicher. Um es sicher zu machen benutzt man sogenannte Paddings. Ein Padding könnte zum Beispiel eine Menge von zufälligen Bits, die man an bestimmten Stellen des Klartextes einfügt. Es gibt allerdings auch noch andere Anwendungsgebiete für Paddings, wie das Erkennen von Übertragungsfehlern.

**Definition 11.18** Ein Padding für ein PKC ist eine probabilistische Funktion  $P : M' \rightarrow M$  mit  $M' \subset M$ .  $m' \in M'$  bezeichnen wir als Nachricht und  $m$  als Klartext.

Als letztes wichtiges Ergebnis wollen wir Chernoff's Grenze vorstellen.

**Theorem 11.19** Sei  $p \leq \frac{1}{2}$  und  $X_1, X_2, \dots, X_n$  unabhängige Zufallsvariablen, die nur die Werte 0,1 annehmen und  $Y = \sum_{i=1}^n X_i$ . Sei weiterhin  $P[X_i = 1] = p$  für alle  $i$ . Dann gilt für alle  $\delta$  mit  $0 < \delta \leq p(1-p)$

$$P\left[\left|\frac{Y}{n} - p\right| > \delta\right] < 2 \cdot e^{-\frac{\delta^2}{2p(1-p)} \cdot n}$$

## 12 Symbolverzeichnis

| Zeichen              | Beschreibung   |
|----------------------|--|
| $\tau(L)$            | kürzeste Vektor eines eindeutigen Gitters (s. Definition 2.31)                                     |
| $\lambda_1(L)$       | Länge des kürzesten Vektors (s. Definition 2.30)   |
| $[n]$                | Das ist die Menge $\{1, 2, \dots, n\}$ .   |
| $\text{diam}(M)$     | Durchmesser der Menge $M$ (s. Definition 2.8)  |
| $\text{frc}(x)$      | Die am nächsten zu $x$ liegende ganze Zahl (s. Definition 1.4).                                    |
| $\mathcal{P}(L)$     | Die Grundmasche des Gitters $L$ (s. Definition 2.19).  |
| $L^*$                | Das zu $L$ duale Gitter (s. Definition 2.26).  |
| $\chi_A$             | Die charakteristische Funktion (s. Definition 2.8).  |
| $\Omega(n)$          | Ein Polynom in $n$   |
| $\log$               | Der Logarithmus zur Basis 2.   |
| $\gamma(n)$          | Eine Funktion mit $\frac{\gamma(n)}{n\sqrt{\log n}} \rightarrow \infty$ für $n \rightarrow \infty$ |
| $U$                  | Die Dichtefunktion der Gleichverteilung.   |
| $\lfloor h \rfloor$  | Die ganze Zahl mit geringstem Abstand zu $h$   |
| $\delta_{i,j}$       | Kroneckerdelta (Definition 2.23)   |
| $L^*$                | Das duale Gitter (s. Definition 2.24)  |
| $B^*$                | Die duale Basis (s. Definition 2.26)   |
| $uSVP$               | Das unique Shortest Vector Problem (s. Def. 2.33)  |
| $dSVP_p$             | Decision SVP mit Parameter $p$ (s. Def. 2.34)  |
| $SVP$                | Das Shortest Vector Problem (s. Abschnitt 2.5)   |
| $CVP$                | Das Closest Vector Problem (s. Abschnitt 2.5)  |
| $SBP$                | Das Shortest Basis Problem (s. Abschnitt 2.5)  |
| $\lambda$            | Das Lebesguemaß (s. Satz 3.5)  |
| $\mu$                | Der Erwartungswert einer Zufallsvariable.  |
| $\text{grad}(f(x))$  | Die Ableitung von $f$ im Punkt $a$   |
| $\Delta(X; Y)$       | Stat. Abstand der Zufallsvariablen $X$ und $Y$ (s. Def. 4.17)                                      |
| $\sigma^2$           | Varianz einer Zufallsvariable (s. Def. 3.1).   |
| $\mathbb{N}$         | Die Menge $\{1, 2, \dots\}$  |
| $\text{ggt}(a, b)$   | Die größte natürliche Zahl, die $a$ und $b$ teilt.   |
| $b \equiv a \pmod p$ | bedeutet, dass $a$ und $b$ in der selben Kongruenzklasse liegen.                                   |
| $B^\top$             | die transponierte Matrix   |

## Literatur

- [Adl 1983] L.M. Adleman, *On breaking generalized knapsack public key cryptosystems*, Proc. of 15th STOC, S. 402-412, ACM, 1983.
- [Ajt 1998] M. Ajtai, *The Shortest Vector Problem in  $L_2$  is NP-hard for Randomized Reductions*, In: Stoc '98, Proceedings of 30th Annual ACM Symp. on Theory of Computing, ACM Press, S. 10-19, 1998.
- [Ajt 1996] M. Ajtai, *Generating hard instances of lattice problems*, In Proc. 28th ACM Symp. on Theory of Computing, Seiten 284-293, 1996.
- [Ajt 1997] M. Ajtai und C.Dwork, *A public-key cryptosystem with worst-case/average-case equivalence*, In Proc. 29th ACM Symp. on theory of Computing, Seiten 284-293, 1997.
- [Ban 1993] W. Banaszczyk, *New bounds in some transference theorems in the geometry of numbers*, Mathematische Annalen, 296(4):625-635, 1993.
- [Bar 1987] Martin Barner und Friedrich Flohr, *Analysis I*, Berlin-New York: W. de Gruyter & Co. 1987.
- [Bar 1996] Martin Barner und Friedrich Flohr, *Analysis II*, Berlin-New York: W. de Gruyter & Co. 1996.
- [Beu 1998] Albrecht Beutelspacher, *Lineare Algebra*, Friedrich Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig/Wiesbaden, 1998.
- [Buch 1999] Johannes Buchmann, *Einführung in die Kryptographie*, Berlin-Heidelberg-New York: Springer 1999.
- [Cai 1999] J.Y. Cai, *Some Recent Progress on the Complexity of Lattice Problems*, In: Proceedings of FCRC, 1999.
- [Els 2002] Jürgen Elstrodt, *Maß- und Integrationstheorie*, Berlin-Heidelberg-New York: Springer 2002.
- [For 1983] O. Forster, *Analysis I*, Friedrich Vieweg & Sohn Verlagsgesellschaft mbH, 1983.
- [Gol 1999] Oded Goldreich, *Modern Cryptography, probabilistic proofs and pseudorandomness*, Berlin-Heidelberg-New York: Springer 1999.
- [Gol 1997] O. Goldreich, S. Goldwasser, S. Halevi, *Eliminating Decryption errors in the Ajtai-Dwork cryptosystem*, Lecture Notes in Computer Science, 1294:105, 1997.
- [Has] J. Hastad, R. Impagliazzo, L.A. Levin, M. Luby, *Construction of Pseudorandom Generator on any One-Way Function*, In: SIAM Journal on Computing.

- [Hop 1979] John Hopcroft und Jeffrey Ullman, *Introduction to Automata Theory, Languages and Computation*, Addison Wesley, 1979.
- [Izm] O. Izmerley, Vortrag „Does Public Key Encryption Exist?“, Aufzeichnungen unter <http://www.cs.technion.ac.il/~talmo/Qubitconf/QUBIT-2003/program/izmerly.pdf>
- [Leh 2000] J. Lehn, H. Wegmann, *Einführung in die Statistik*, Stuttgart-Leipzig:Teubner, 2000.
- [Len 1982] A.K. Lenstra, H.W. Lenstra, L. Lovász, *Factoring Polynomials with Rational Coefficients*, *Mathematische Annalen* 261 (1982), S. 513-534.
- [Men] Alfred Menezes, Paul van Oorschot, Scott Vanstone, *Handbook of applied cryptography*, CRC Press LLC 2001.
- [Odl 1990] A.M. Odlyzko, *The rise and fall of knapsack cryptosystems*, *Cryptology and Computational Number Theory*, volume 42 of Proc. of Symposia in Applied Mathematics, S. 75-88, A.M.S. 1990.
- [Regev 2003] O. Regev, *New Lattice Based Cryptographic Constructions*, erhältlich unter <http://www.cs.tau.ac.il/~odedr/>
- [Rei 1990] Karl Rüdiger Reischuk, *Einführung in die Komplexitätstheorie*, Reischuk-Stuttgart: Teubner 1990.
- [Schn] C.P. Schnorr, *Vorlesungen Gittertheorie und algorithmische Geometrie an der Johann Wolfgang von Goethe-Universität Frankfurt a.M. im Wintersemester 2001/02*, Mitschrift unter [http://www.mi.informatik.uni-frankfurt.de/teaching/lecture\\_notes/schnorr.gitter.ps](http://www.mi.informatik.uni-frankfurt.de/teaching/lecture_notes/schnorr.gitter.ps).
- [Sho 1997] Peter Shor, *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*, In: *SIAM Journal on Computing*, 26, 1997.
- [Tur 1936] Alan Turing, *On computable numbers with an application to the Entscheidungsproblem*, In: *Proceedings of the London Mathematical Society*, Band 42, Seiten 230-265, 1936.
- [Weg 1993] Ingo Wegener, *Theoretische Informatik*, B.G. Teubner, Stuttgart, 1993.
- [Wer 2000] Dirk Werner, *Funktionalanalysis*, Berlin-Heidelberg-New York: Springer 2000.
- [Yao 1982] A.C. Yao, *Theory and Applications of Trapdoor Functions*, In 23rd IEEE Symposium on Foundations of Computer Science, Seiten 80-91, 1982.