# Gitterbasenreduktion für beliebige Normen

•

Dissertation
zur Erlangung des Doktorgrades
der Naturwissenschaften

•

vorgelegt beim Fachbereich Mathematik
der Johann Wolfgang Goethe–Universität

in Frankfurt am Main

von

Michael Kaib[1]

aus Frankfurt am Main

Frankfurt am Main 1994

[1]e-mail: kaib@cs.uni-frankfurt.de

vom Fachbereich Mathematik der
Johann Wolfgang Goethe–Universität
als Dissertation angenommen

| | |
|---|---|
| Dekan | Prof. Dr. J. Bliedtner |
| Gutachter | Prof. Dr. C.P. Schnorr<br>Prof. Dr. B. Vallée |
| Datum der Disputation | 24. Februar 1995 |

**Abstract**

We generalize the reduction theory of lattice bases to arbitrary norms and show new properties of reduced bases for the generalized reduction concepts. We generalize the Gaussian Algorithm for the reduction of two–dimensional lattice bases to arbitrary norms and obtain an universally sharp upper bound on the number of its iterations for all norms. We develop a new variant of the Gaussian Algorithm with low bit complexity for special $l_p$-norms. Therefore we use the ideas of Schoenhage's fast reduction algorithm for quadratic forms to obtain the first asymptotically fast centered reduction algorithm for two–dimensional lattice bases.

**Zusammenfassung**

Wir verallgemeinern die Reduktionstheorie von Gitterbasen für beliebige Normen. Dabei zeigen wir neue Eigenschaften reduzierter Basen für die verallgemeinerten Reduktionsbegriffe. Wir verallgemeinern den Gauß–Algorithmus zur Reduktion zweidimensionaler Gitterbasen für alle Normen und erhalten eine universelle scharfe obere Schranke für die Zahl seiner Iterationen. Wir entwickeln für spezielle $l_p$-Normen eine Variante des Gauß–Algorithmus mit niedriger Bit–Komplexität. Hierzu wird Schönhages schneller Reduktionsalgorithmus für quadratische Formen auf die Reduktion von Gitterbasen im klassischen zentrierten Fall übertragen.

# Inhaltsverzeichnis

**Einleitung** **3**

**1 Reduzierte Gitterbasen für beliebige Normen** **9**

1.1 Grundlagen . . . . . 9

1.2 Die Höhenfunktionen . . . . . 11

1.3 Blockreduzierte Basen . . . . . 17

1.4 Gaußreduzierte Basen . . . . . 25

**2 Der verallgemeinerte Gauß–Algorithmus** **31**

2.1 Der Reduktionsschritt . . . . . 31

2.2 Vorgänger einer wohlgeordneten Basis . . . . . 32

2.3 Vorgänger einer reduzierten Basis . . . . . 35

2.4 Norm aufeinanderfolgender Basisvektoren . . . . . 37

2.5 Abschätzung der Anzahl der Iterationen . . . . . 40

**3 Zur Schrittzahl des verallgemeinerten Gauß–Algorithmus 49**

3.1 Effiziente Reduktion . . . . . 50

3.2 Algorithmen für die $l_1$- und die $l_\infty$-Norm . . . . . 52

3.2.1 Ein gewichteter Median–Algorithmus für die $l_1-$Norm . . . . 52

3.2.2 Sortierung der Komponenten für die $l_\infty-$Norm . . . . . 54

**4 Verbesserte Bitkomplexität des Gauß–Algorithmus 57**

4.1 Vorüberlegungen zu Bitkomplexität und Stabilität . . . . . 58

4.2 Der schnelle Algorithmus . . . . . 63

4.3 Beweis der Schrittzahl . . . . . 65

**Literaturverzeichnis 71**

**Symbolverzeichnis 75**

**Index 76**

# Einleitung

Der historische Zugang zu Gittern war das zahlentheoretische Studium der ganzzahligen Gitterpunkte im mehrdimensionalen reellen Raum. Die geometrische Betrachtungsweise führte zum Begriff „Geometrie der Zahlen“. Allgemeiner versteht man unter einem Gitter jede (durch eine Gitterbasis erzeugte) diskrete additive Untergruppe des $\mathbb{R}^n$ . Die Reduktionstheorie von Gitterbasen hat das Ziel, eine kanonische Basis aus der Menge der Basen eines Gitters auszuwählen. Sie wurde zunächst in der äquivalenten Darstellung quadratischer Formen von Lagrange [La1773], Gauß [Ga1801] , Dirichlet [Di1850], Hermite [He1850], Korkine und Zolotarev [KZ1873] und Minkowski [Mi1891] entwickelt.

Die algorithmischen Gesichtspunkte der Gitterbasenreduktion sind in letzter Zeit wieder in das Blickfeld der angewandten Mathematik gerückt. Eingeleitet wurde dies durch H.W. Lenstra's neue Methode zur ganzzahligen Optimierung (1982), die im wesentlichen die Berechnung reduzierter Basen von geeigneten Gittern erfordert, wie auch durch den effizienten Reduktionsalgorithmus von Lovasz, den sogenannten $L^3$-Algorithmus [LLL82]. Inzwischen wurden diese Algorithmen in vielen Bereichen verbessert [S89, SE91]. Verwandte Algorithmen zur Berechnung reduzierter Basen für neue Reduktionsbegriffe wurden entwickelt [S87]. Daneben wurden neue Anwendungen der Gitterbasenreduktion, z.B. in der Kryptographie und algorithmischen Zahlentheorie, gefunden [Co&a92, S92].

Ein Beispiel für eine wichtige Anwendung sind die kryptographischen Schemen von Chor, Rivest [CR88] und Damgard, deren Sicherheit auf der Schwierigkeit des Subset–Sum–Problems basiert. Mit Subset–Sum–Problem bezeichnet man das NP–vollständige Entscheidungsproblem, ob zu gegebenen $a_1, \ldots, a_m, s \in \mathbb{N}_0$ gewisse $x_1, \ldots, x_m \in \{0, 1\}$ existieren, so daß $\sum_i x_i a_i = s$ . Zum Lösen dieser Aufgabe genügt die Berechnung eines kürzesten Vektors bezüglich der $l_\infty$-Norm in dem Gitter $L \subset \mathbb{Z}^{m+2}$, das von den Vektoren $b_i = 2e_i + 2a_i e_{m+1}$ für $i = 1, \ldots, m$ und $b_{m+1} = \sum_1^m e_i + 2se_{m+1} + e_{m+2}$ erzeugt wird ($e_i$ bezeichnet den $i$-ten Einheitsvektor). Für die Vektoren dieses Gitters gilt nämlich $\| \sum_{i=1}^{m+1} x_i b_i \|_\infty = 1$ dann und nur dann, wenn $\varepsilon = \pm 1,\ x_1, \ldots, x_m \in \{0, \varepsilon\},\ x_{m+1} = -\varepsilon,\ \sum_i x_i a_i = \varepsilon s$ . Andere

Beispiele sind das 3-SAT–Problem und das Aussieben von Kongruenzgleichungen in Faktorisierungsalgorithmen. Diese Probleme lassen sich durch Berechnung eines kürzesten Vektors geeigneter Gitter bezüglich der $l_\infty$- resp. $l_1$-Norm lösen.

Kürzlich stellten Lovasz und Scarf eine Variante des $L^3$-Algorithmus für beliebige Normen vor [LS92]. Bis dahin war der algorithmische Teil der Gitterbasenreduktion nur für die euklidische Norm entwickelt. Man versuchte gegebenenfalls, die allgemeine Norm durch die euklidische zu approximieren. In diesem Zusammenhang zeigte Lenstra, daß für jede Norm ein geeigneter Isomorphismus $A$ von $\mathbb{R}^n$ berechnet werden kann, so daß $\| x \| \leq \| Ax \|_2 \leq 2n^{1.5} \| x \|$ für alle $x \in \mathbb{R}^n$ gilt [Len83]. Die teilwiese sehr aufwendigen Approximationstechniken sind jedoch in hohen Dimensionen oft nicht ausreichend, um eine Lösung für ein gegebenes Problem zu berechnen. So kann man für das Subset–Sum–Problem zwar eine große Klasse der Eingaben durch $l_2$-Norm-kürzeste Vektoren des Gitters lösen, jedoch bleibt ein schwieriger Teil unzugänglich [Co&a92]. Daher erscheint die Verallgemeinerung der Gitterbasenreduktion und ihrer Algorithmen für beliebige Normen sinnvoll.

Die bekanntesten Reduktionsbegriffe stammen von Gauß, Hermite, Minkowski, Lovasz und Schnorr. Gauß' Reduktionsbegriff beschränkt sich auf den zweidimensionalen Fall. Die Begriffe von Hermite und Lovasz entstanden durch Verallgemeinerung des Gaußschen Begriffes auf beliebige Dimensionen, indem die Dimension durch Projektion verringert wird. Dieses Vorgehen ist algorithmisch motiviert. Allerdings sind die Algorithmen zur Berechnung Hermite–reduzierter Basen für höhere Dimensionen nicht praktikabel. Daher erklärte Schnorr eine Hierarchie von Reduktionsbegriffen, die die Begriffe von Hermite und Lovasz als Extremalfälle umfaßt. Diese Begriffe wurden bisher fast nur für die euklidische Norm betrachtet. Beliebige Normen hatte nur Minkowski betrachtet. Sein Reduktionsbegriff ist aber leider nicht algorithmisch motiviert.

Diese Arbeit hat das Ziel, die algorithmischen Aspekte der Gitterbasenreduktion von der euklidischen Norm auf beliebige Normen zu erweitern. Damit wird die von Lovasz und Scarf begonnene Forschungsarbeit fortgesetzt. Wir übertragen im ersten Kapitel die oben genannten Reduktionsbegriffe für allgemeine Normen. Unser Augenmerk gilt dabei insbesondere den algorithmisch motivierten Begriffen. Wir zeigen erstmals die Eigenschaften Schnorr–reduzierter Basen für beliebige Normen. Wesentlich sind hierfür die von Lovasz und Scarf eingeführten Höhenfunktionen, die die orthogonalen Projektionen des euklidischen Falles ersetzen. Ist $b_1, \ldots, b_m$ eine feste Gitterbasis, so mißt die Höhenfunktion $F_i$ den Abstand eines Punktes (in der gegebenen Norm) zu dem von $b_1, \ldots, b_{i-1}$ aufgespannten Unterraum. Das Höhenprodukt $\prod F_i(b_i)$ verallgemeinert den euklidischen Begriff der Gitterdeterminante. Das Höhenprodukt ist im Gegensatz zur Determinante von der speziellen

Wahl der Basis abhängig. Die Verallgemeinerung der bekannten Theorie wird jedoch ermöglicht durch die gitterunabhängigen oberen und unteren Schranken für den Quotienten aus Höhenprodukt und Determinante, die wir in Lemma 5 zeigen. Ein häufig verwendetes Maß für die Reduziertheit einer Gitterbasis sind die sukzessiven Minima $\lambda_i$ . Für die euklidische Norm ist der Quotient $\lambda_1^2 \left(\det L\right)^{-2/m}$ nach oben durch die Hermite–Konstante $\gamma_m$ beschränkt. Wir zeigen in Satz 4 die wichtige Ungleichung

$$\lambda_1 \left( \prod_{i=1}^{m} F_i \left(b_i\right) \right)^{-1/m} \leq \quad m!^{1/m}$$

für alle Normen. Wir nennen $\kappa_m$ das Supremum der linken Seite für alle Normen und Gitterbasen. Es folgt $\gamma_m \leq \kappa_m^2$ . Die von der euklidischen Norm bekannten Ungleichungen $\frac{2}{\sqrt{i+3}} \gamma_\beta^{-\frac{i-1}{\beta-1}} \leq \| b_i \|_2 / \lambda_i \leq \frac{\sqrt{i+3}}{2} \gamma_\beta^{\frac{m-1}{\beta-1}}$ für mit Blockweite $\beta$ Schnorr–reduzierte Basen erscheinen in Satz 12 in der Form

$$\frac{4}{i+3} \kappa_\beta^{-2\frac{i-1}{\beta-1}} \leq \| b_i \| / \lambda_i \leq \frac{i+1}{4} \kappa_\beta^{2\frac{m-1}{\beta-1}} .$$

Zum Abschluß des ersten Kapitels werden schärfere Resultate für den zweidimensionalen Fall gezeigt. Gauß–reduzierte Basen werden definiert. Ihre Eigenschaften werden in den Sätzen 18 und 19 gezeigt.

Wir analysieren Algorithmen für den zweidimensionalen Fall. Der Gauß–Algorithmus erhält als Eingabe eine Gitterbasis $(a, b)$ und findet eine reduzierte Basis desselben Gitters aus Vektoren, deren Normen die sukzessiven Minima sind. Er führt hierzu Reduktionsschritte der Art

$$(a, b) \mapsto ((b - \mu a), a)$$

aus, wobei $\mu \in \mathbb{Z}$ die Norm $\| b - \mu a \|$ minimiert. Das zweite Kapitel verallgemeinert den Gauß–Algorithmus von der euklidischen auf beliebige Normen. Wir zeigen für die Anzahl seiner Reduktionsschritte die obere Schranke

$$\log_{1+\sqrt{2}} (2\sqrt{2}\, B) + o(1) ,$$

für $B \to \infty$ , wobei $B$ der Quotient aus der Norm des längsten Eingabevektors und dem zweiten sukzessiven Minimum des Gitters ist (Satz 31). Für jede Norm und

jedes Gitter gibt es eine Folge von Eingabebasen, die diese Schranke bis auf maximal 1.393 Iterationen erreicht (Satz 32). Die worst–case–Eingaben erfüllen dieselbe Rekursion wie sie schon Dupré in seiner Analyse des zentrierten euklidischen Algorithmus entdeckte [Du1846]. Somit werden die scharfen Schranken von Vallée [Va91] für beliebige Normen gezeigt. Sehr hilfreich ist die von Vallée eingeführte Definition wohlgeordneter Basen, die wir auf den Fall beliebiger Normen übertragen. Wir charakterisieren in Lemma 24 die Vorgängerbasen einer wohlgeordneten Gitterbasis unabhängig von der Norm.

Im dritten Kapitel analysieren wir die Schrittzahl des verallgemeinerten Gauß–Algorithmus für das RAM–Modell. Wir zeigen in Satz 33 die obere Schranke von

$$O(n \log (n + \lambda_2/\lambda_1) + \log B)$$

arithmetischen Schritten und $O(\log (n + \lambda_2/\lambda_1))$ Normberechnungen ($n$ ist die Dimension der Eingabevektoren). Für die $l_1$- und $l_\infty$-Norm werden effiziente Algorithmen zur Durchführung des Reduktionsschrittes angegeben. Dies ergibt eine Verbesserung der soeben genannten Schranke für diese beiden speziellen Normen auf $O(n \log n + \log B)$ arithmetische Schritte.

Im letzten Kapitel stellen wir zum ersten Mal eine Variante des Gauß–Algorithmus für die $l_1$-, $l_2$- und $l_\infty$-Norm mit niedriger Bitkomplexität vor. Es bezeichne $\mathcal{M}(B)$ eine obere Schranke für die Bitoperationen zur Multiplikation von $B$-Bit–Zahlen. Der Algorithmus benötigt höchstens $O((n+\log B)\mathcal{M}(B))$ Bitoperationen für die $l_2$-Norm und höchstens $O(n\mathcal{M}(B) \log B)$ bzw. $O(n \log n\, \mathcal{M}(B) \log B)$ Bitoperationen für die $l_\infty$- bzw. $l_1$-Norm bei Eingabe von Vektoren $a, b \in \mathbb{Z}$ mit Norm höchstens $2^B$. Dadurch werden die schnellen Algorithmen von Schönhage [Sh71, Sh91] auf den zentrierten Fall und auf verschiedene Normen erweitert.

Wie bei Lehmer [Leh38] und Schönhage wird der größte Teil der arithmetischen Operationen nur auf den führenden Stellen der ganzen Zahlen ausgeführt. Wir übertragen diese Ideen erstmals auf die Gauß–Reduktion im zentrierten Fall. Unsere neue Überlegung ist, daß die Schritte des Gauß–Algorithmus stabil bleiben, solange der Approximationsfehler $\frac{1}{12}$ der Norm des bislang kürzeren Basisvektors nicht übersteigt. Dieses Ergebnis gilt für alle Normen. Weiter verwenden wir die genauen Analysen der Reduktionsschritte aus dem zweiten Kapitel. Der Approximationsfehler wird im Algorithmus durch den Abstieg, die Differenz aus Eingabe- und Ausgabegröße, kontrolliert. Der Algorithmus vollzieht nach Schönhages Vorbild zwei rekursive Aufrufe von etwa halber Genauigkeit.

Ich möchte mich insbesondere bei meinem Lehrer, Professor Claus Schnorr, für viele fachliche Hinweise und fruchtbare Diskussionen und für die umfassende Ausbildung

bedanken, an deren Abschluß diese Arbeit steht. Weiter möchte ich mich für hilfreiche Diskussionen besonders bei Brigitte Vallée (Caen), Hervé Daudé (Marseille) und Arnold Schönhage (Bonn) bedanken.

# Kapitel 1

# Reduzierte Gitterbasen für beliebige Normen

Wir entwickeln in diesem Kapitel Reduktionskonzepte für verschiedene Normen. Die Begriffe blockreduzierter und Gauß–reduzierter Basen werden von der euklidischen auf beliebige Normen übertragen. Die für die euklidische Norm bekannten Ergebnisse entstehen dabei für beliebige Normen in veränderter Form und in neuem Licht.

## 1.1 Grundlagen

*Gitter* sind diskrete additive Untergruppen $L \subset \mathbb{R}^n$ . Ein linear unabhängiges Erzeugendensystem eines Gitters $L$ heißt *Basis* von $L$. Jedes Gitter besitzt eine Basis. Umgekehrt ist jede von endlich vielen linear unabhängigen Vektoren erzeugte Gruppe diskret, also ein Gitter. Der *Rang* eines Gitters ist die Anzahl der Vektoren einer Basis. Ein System von $k$ Gittervektoren $b_1, \ldots, b_k$ heißt *primitives System* für das Gitter $L$, wenn sich $b_1, \ldots, b_k$ zu einer Basis von $L$ ergänzen lassen. Ein System $b_1, \ldots, b_k$ ist genau dann primitiv, wenn $\text{span}(b_1, \ldots, b_k) \cap L = \mathbb{Z}b_1 + \ldots + \mathbb{Z}b_k$ .

Sei eine beliebige Norm $\| \, . \, \|$ auf $\mathbb{R}^n$ gegeben. Wir bezeichnen mit

$$B_{\rho,c} \;=\; \{x \in \mathbb{R}^n \mid \| \, x - c \, \| \;\leq\; \rho\}$$

die *Kugel* mit *Radius* $\rho$ um $c$. Auf $\mathbb{R}^n$ ist die Diskretheit einer Menge unabhängig von der speziellen Wahl der Norm. In jeder Kugel können nur endlich viele Gittervektoren liegen. Wir definieren

**Definition 1.** *Das* $i$–te sukzessive Minimum $\lambda_i$ *des Gitters* $L$ *bezüglich der Norm* $\| \, . \, \|$ *ist der minimale Radius einer Kugel um* 0, *die* $i$ *linear unabhängige Gittervektoren enthält:*

$$\lambda_i \;=\; \inf\{\rho \,|\, \dim \operatorname{span} L \cap B_{\rho,0} \;\geq\; i\} \;.$$

Insbesondere ist $\lambda = \lambda_1(L)$ die kürzeste positive Länge eines Vektors im Gitter $L$. Ist $x \in L$ mit $\| \, x \, \| \;=\; \lambda$, so heißt $x$ ein *kürzester Gittervektor*.

Ziel der *Gitterbasenreduktion* ist es, mit effizienten Algorithmen eine gegebene Gitterbasis in eine Basis desselben Gitters zu transformieren, deren Vektoren möglichst gut die sukzessiven Minima approximieren. Solche Basen nennt man *reduzierte Basen*. Sie wurden zuerst für die Dimensionen 2 und 3 von Lagrange [La1773], Gauß [Ga1801] und Dirichlet [Di1850] studiert. Die grundlegenden Arbeiten für beliebige Dimensionen stammen von Hermite [He1850], Korkine und Zolotarev [KZ1873], Minkowski [Mi1891], Lovasz e.a. [LLL82, LS92] und von Schnorr [S87] . Der naheliegendste, aber nicht algorithmisch motivierte Begriff stammt von Minkowski:

**Definition 2.** *Eine Gitterbasis* $b_1, \ldots, b_m$ *heißt* reduziert *im Sinne von Minkowski, wenn jeweils* $b_k$ *der kürzeste unter allen Gittervektoren ist, die mit* $b_1, \ldots, b_{k-1}$ *ein primitives System bilden* $(k = 1, \ldots, m)$ .

Die algorithmischen Aspekte der Gitterbasenreduktion in den oben genannten Arbeiten beschränkten sich (bis auf [LS92]) auf die euklidische Norm $\| \, x \, \| = \sqrt{\sum_{i=i}^{n} x_i^2}$ . Diese ist dafür besonders geeignet, u.a. wegen ihrer Rotationssymmetrie sowie einfachen Berechnungen von Winkeln und Projektionen mithilfe von Skalarprodukten. Wir werden die wichtigsten Reduktionsbegriffe für beliebige Normen verallgemeinern. Wir können die speziellen Eigenschaften der euklidischen Norm nicht mehr nutzen, sondern alleine die Definition

1. $\| \, x \, \| \;>\; 0$ für alle $x \in \mathbb{R}^n - \{0\}$ (Positivität)
2. $\| \, \xi x \, \| \;=\; |\xi| \;\| \, x \, \|$ für alle $\xi \in \mathbb{R}$ und alle $x \in \mathbb{R}^n$ (Symmetrie, Linearität)
3. $\| \, x + y \, \| \;\leq\; \| \, x \, \| \;+\; \| \, y \, \|$ für alle $x, y \in \mathbb{R}^n$ (Konvexität)

Die Kugel $B = B_{1,0}$ ist eine kompakte konvexe nullsymmetrische Menge von positivem Volumen. Umgekehrt definiert jede solche Menge $B$ eine Norm durch die Setzung $\| \, x \, \| \;=\; \inf\{\rho \,|\, x \;\in\; \rho B\}$ . Von besonderer Bedeutung bleiben die $l_p$–Normen, die durch $\| \, x \, \|_p = (\sum_{i=1}^{n} |x_i|^p)^{1/p}$ für $1 \leq p < \infty$ und durch $\| \, x \, \|_\infty = \max_{i=1}^{n} |x_i|$ definiert sind.

Nicht jedes Gitter besitzt eine Basis aus Vektoren, deren Normen die sukzessiven Minima sind. Es kann in hinreichend großen Dimensionen vorkommen, daß ein primitives System von Gittervektoren durch Hinzunahme eines linear unabhängigen Vektors von minimaler Norm seine Primitivität verliert. Ein Beispiel hierfür ist das von den Einheitsvektoren $e_1, \ldots, e_n \in \mathbb{R}^n$ und $x = \frac{1}{2}\sum_{i=1}^{n} e_i$ erzeugte Gitter. Ist $n > 2^p$, so gilt in der $l_p$–Norm $\lambda_1 = \ldots = \lambda_n = 1$. Die $n$ Einheitsvektoren sind die einzigen Gittervektoren mit Norm 1, erzeugen das Gitter aber nicht. Von Mahler [Ma38] und H. Weyl [We42] stammt folgendes Resultat über die Approximation der sukzessiven Minima durch Minkowski–reduzierte Basen für beliebige Normen:

**Satz 3.** *Für jede Minkowski–reduzierte Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$1 \;\leq\; \| b_i \| / \lambda_i(L) \;\leq\; \left(\frac{3}{2}\right)^{i-2} \quad .$$

Eine schöne Darstellung der nicht algorithmischen Gittertheorie für beliebige Normen findet man zum Beispiel in dem Buch von Gruber und Lekkerkerker [GL87]. Wir suchen jetzt neue Reduktionsbegriffe, für die

- die Normen der Basisvektoren möglichst gute Approximationen der sukzessiven Minima sind und
- sich eine reduzierte Basis „effizient“ berechnen läßt.

Das zweite Ziel ist Gegenstand der nächsten Kapitel, wobei wir uns auf Gitter vom Rang 2 beschränken werden. Im weiteren Verlauf dieses Kapitels verallgemeinern wir zunächst die Definitionen reduzierter Basen im Sinne von Korkine, Zolotarev und Hermite [He1850, KZ1873] und von Schnorr [S87] für beliebige Normen. Wir beweisen, daß diese Basen die sukzessiven Minima besser approximieren als dies von im Sinne von Minkowski reduzierten Basen bekannt ist.

## 1.2 Die Höhenfunktionen

Sei im folgenden $b_1, \ldots, b_m \in \mathbb{R}^n$ eine feste, geordnete Gitterbasis. Wir definieren zu dieser Basis die *Höhenfunktionen* $F_i : \mathbb{R}^n \longrightarrow \mathbb{R}$ bezüglich der gegebenen Norm und Basis durch $F_1(x) = \| x \|$ und für $i \leq m$ durch

$$F_i(x) \;=\; \min_{\xi_1,\ldots,\xi_{i-1}\in\mathbb{R}} \| x - (\xi_1 b_1 + \ldots + \xi_{i-1} b_{i-1}) \| \;=\; \min_{\xi\in\mathbb{R}} F_{i-1}(x - \xi b_{i-1}) \; .$$

Die Höhe $F_i$ eines Vektors ist sein Abstand zu dem von $b_1, \ldots, b_{i-1}$ erzeugten Unterraum. Man rechnet leicht nach, daß jede Höhenfunktion $F_i$ eine Norm auf $\operatorname{span}\{b_i, \ldots, b_m\}$ ist. Die Höhen der Basisvektoren approximieren die Norm des kürzesten Gittervektors:

**Satz 4.** *Für jede Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$\min_{i=1,\ldots,m} F_i(b_i) \;\leq\; \lambda_1(L) \;\leq\; \left(m! \prod_{i=1}^{m} F_i(b_i)\right)^{1/m} .$$

Bevor wir Satz 4 beweisen, veranschaulichen wir seine Bedeutung am Beispiel der euklidischen Norm. Dort ist das Produkt der Höhen $\prod F_i(b_i)$ das Volumen des von den Basisvektoren $b_1, \ldots, b_m$ aufgespannten Parallelepipeds, also nach Definition die *Determinante* $\det L$ des Gitters $L = \mathbb{Z}b_1 + \cdots + \mathbb{Z}b_m$. Die *Hermite–Konstante* $\gamma_m$ wird definiert als Supremum des Ausdruckes $\lambda_1^2 (\det L)^{-2/m}$ für alle Gitter vom Rang $m$. Man kennt die expliziten Werte von $\gamma_m$ für $1 \leq m \leq 8$ und für $m \to \infty$ die asymptotischen Schranken

$$\frac{m}{2\pi e}\,(1+o(1)) \;\leq\; \gamma_m \;\leq\; \frac{m}{1.14\pi e}\,(1+o(1)) .$$

Satz 4 erlaubt, in Analogie für beliebige Normen $\|\cdot\|$ auf $\mathbb{R}^m$ die Konstante $\kappa_{m,\|\cdot\|}$ zu definieren als Supremum

$$\kappa_{m,\|\cdot\|} \;=\; \sup_{\substack{b_1,\ldots,b_m \\ \text{Basis von } \mathbb{R}^m}} \lambda_1(\mathbb{Z}b_1 + \cdots + \mathbb{Z}b_m) \left(\prod_{i=1}^{m} F_i(b_i)\right)^{-1/m} , \tag{1.1}$$

und $\kappa_m = \sup \kappa_{m,\|\cdot\|}$ als Supremum von $\kappa_{m,\|\cdot\|}$ über alle Normen $\|\cdot\|$ auf $\mathbb{R}^m$. Dann ist $\gamma_m = \kappa^2_{m,\|\cdot\|_2}$, $\kappa_{m,\|\cdot\|} \leq \kappa_m$ und, wegen Satz 4, $\kappa_m \leq m!^{1/m}$, also gilt

$$\sqrt{\gamma_m} \;\leq\; \kappa_m \;\leq\; m!^{1/m} .$$

Dadurch wird den Höhen, die wesentlich für die algorithmische Theorie sind, für alle Normen eine ähnlich zentrale Rolle wie bei der euklidischen Norm zuerkannt. Das wirkliche asymptotische Verhalten der $\kappa$-Konstanten für $m \to \infty$ bleibt ein offenes Problem. Unsere obere Schranke ist etwa das Quadrat der unteren:

$$\sqrt{\frac{m}{2\pi e}}\,(1+o(1)) \;\leq\; \kappa_m \;\leq\; \frac{m}{e}\,(1+o(1)) .$$

Wir werden im weiteren nur die Konstanten $\kappa_m$ verwenden, weil wir im nächsten Abschnitt beim Beweis von Satz 12 zu einer „Blockweite“ $\beta \leq m$ eine simultane Schranke für $\kappa_{\beta,F_1}, \ldots, \kappa_{\beta,F_{m-\beta}}$ benötigen. Im allgemeinen Fall sind die Normen $F_1, \ldots, F_{m-\beta}$ auf $\beta$-dimensionalen Unterräumen verschieden.

Wir betrachten zunächst ein Beispiel für die $l_\infty$-Norm. Sei $b_1, \ldots, b_m \in \mathbb{R}^m$ die durch $b_i = e_i + \ldots + e_m$ konstruierte Basis von $L = \mathbb{Z}^m$, wobei $e_j \in \mathbb{R}^m$ der $j$-te Einheitsvektor ist.

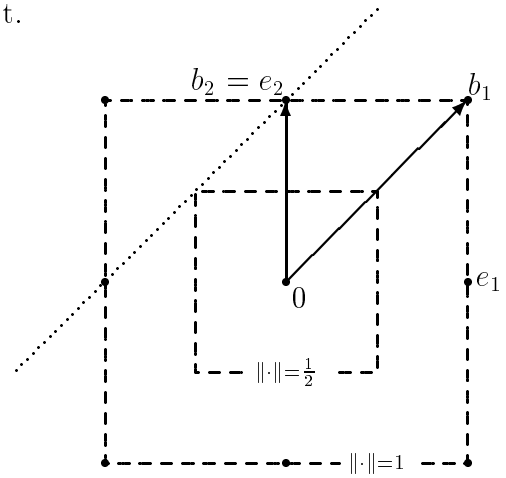


Abbildung 1.1: Beispiel für m=2

Daraus folgt $F_i(b_i) = \frac{1}{2}$ für $i \geq 2$, und mithin ist $\kappa_m \geq 2^{\frac{m-1}{m}}$. Aus dem Satz folgt somit

$$\kappa_2 \;=\; \sqrt{2}\,. \tag{1.2}$$

Weiter folgt, daß $\kappa_m$ für $2 \leq m \leq 8$ größer ist als die untere Schranke $\sqrt{\gamma}_m$. Durch Erweiterung dieser Konstruktion erhalten wir auf Seite 16 die allgemeinere Ungleichung $\kappa_{m+1}^{m+1} \geq 2\,\kappa_m^m$. Man sieht an diesem Beispiel auch, daß das *Höhenprodukt* im Gegensatz zur Determinante *von der Wahl der Basis abhängen kann*, denn die Basis $e_1, \ldots, e_m$ von $L = \mathbb{Z}^m$ hat Höhenprodukt 1. Es gilt aber die zentrale Ungleichung:

**Lemma 5.** *Für jede Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ gilt*

$$\frac{2^m}{m!\;V_m} \;\leq\; \frac{\prod_i F_i\,(b_i)}{\det L} \;\leq\; \frac{2^m}{V_m}\,,$$

*wobei* $V_m = \mathrm{vol}_m\,(B_{1,0} \cap \mathrm{span}\,L) \;=\; \mathrm{vol}_m\,\{x \in \mathrm{span}\,L \mid \|\,x\,\| \leq 1\}\,.$

Aus dem Lemma folgt für zwei verschiedene Basen $b_1, \ldots, b_m$ und $\tilde{b}_1, \ldots, \tilde{b}_m$ desselben Gitters mit den korrespondierenden Höhenfunktionen $F_i, \tilde{F}_i$ die Ungleichung

$$\prod_{i=1}^{m} F_i(b_i) \;/\; \prod_{i=1}^{m} \tilde{F}_i(\tilde{b}_i) \;\leq\; m! \;. \tag{1.3}$$

**Beweis des Lemmas.** Bezeichne $V_i$ für $i = 1, \ldots, m$ das $i$-dimensionale Volumen von $B_{1,0} \cap \mathrm{span}\{b_1, \ldots, b_i\}$ und $\hat{b}_1, \ldots, \hat{b}_m$ die zu $b_1, \ldots, b_m$ korrespondierende orthogonale Basis bezüglich des Standardskalarproduktes, so daß $\det L = \| \hat{b}_1 \|_2 \cdots \| \hat{b}_m \|_2$ . Wir beweisen die Ungleichungen durch Induktion über die Dimension $m$. Wir schreiben zunächst $V_m$ als Integration in Richtung $\hat{b}_m$:

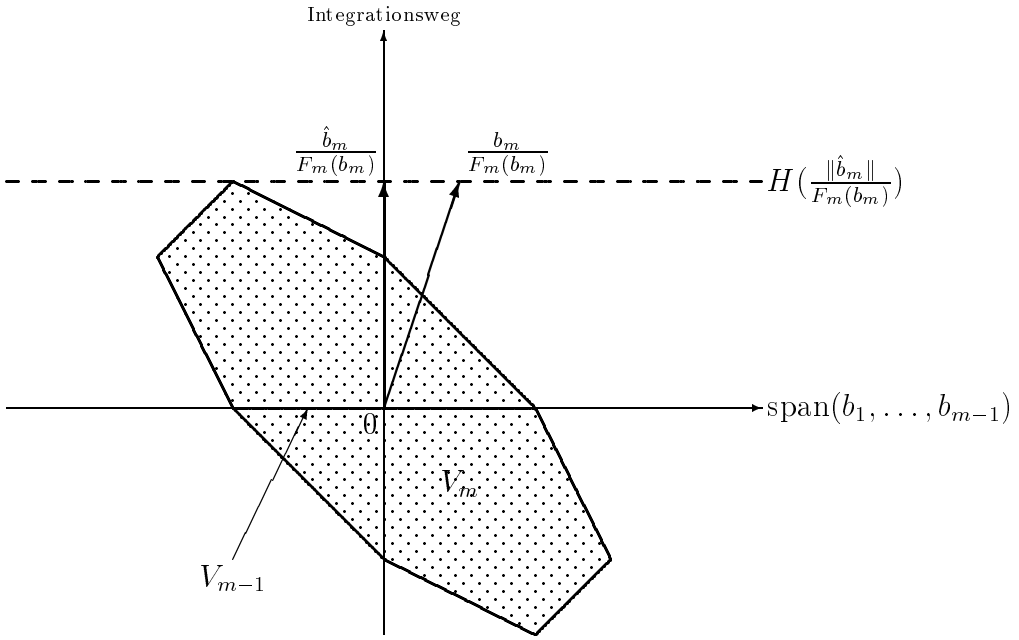


Abbildung 1.2: Abschätzung des Kugelvolumens $V_m$

$$V_m \;=\; \int_{-\infty}^{\infty} v(r)\, dr \;,$$

wobei dann $v(r)$ das Volumen des Schnittes von $B_{1,0}$ mit der affinen Hyperebene

$$H(r) \;=\; \{\, x \in \mathrm{span} L \mid < x, \hat{b}_m > \;=\; r \,\| \hat{b}_m \|_2 \}$$

ist (vgl. Abb. 1.2). Es gilt $v(0) = V_{m-1}$ . Weiter gilt für alle $x \in H(r)$, daß

$$\| x \| \;\geq\; F_m(x) \;=\; r \,\| \hat{b}_m \|_2^{-1}\, F_m(\hat{b}_m) \;.$$

Da $F_m(\hat{b}_m) = F_m(b_m)$ folgt für $|r| > \hat{r} := \| \hat{b}_m \|_2 / F_m(b_m)$, daß $\| x \| > 1$ und mithin $v(r) = 0$. Wegen Konvexität und Nullsymmetrie enthält $B_{1,0}$ eine $m$-dimensionale Pyramide mit Grundfläche $B_{1,0} \cap \mathrm{span}(b_1, \ldots, b_{m-1})$ und Spitzen in $H(\pm\hat{r})$. D.h., für $|r| \le \hat{r}$ gelten die Ungleichungen

$$\left(1 - \frac{|r|}{\hat{r}}\right)^{m-1} V_{m-1} \;\le\; v(r) \;\le\; V_{m-1} \,.$$

Die Auswertung des Integrales ergibt

$$\frac{2}{m} \frac{\| \hat{b}_m \|}{F_m(b_m)} V_{m-1} \;\le\; V_m \;\le\; 2 \frac{\| \hat{b}_m \|}{F_m(b_m)} V_{m-1} \,,$$

woraus induktiv die Behauptung folgt. □

Man findet die rechte Ungleichung des Lemmas für das Gitter $L = \mathbb{Z}^n$ bereits bei Lovasz und Scarf [LS92] (Seite 755, Gleichung 5).

**Beweis von Satz 4.** Zum Beweis der linken Ungleichung sei $b = r_1 b_1 + \ldots + r_k b_k$ eine Darstellung eines kürzesten Gittervektors $b$ mit $r_1, \ldots, r_k \in \mathbb{Z}$ wobei $r_k \neq 0$. Dann ist

$$\lambda_1 \;=\; \| b \| \;\ge\; F_k(b) \;=\; |r_k| \, F_k(b_k) \;\ge\; F_k(b_k) \,.$$

Zum Beweis der rechten Ungleichung zeigen wir zunächst:

$$\left(\frac{\lambda_1}{2}\right)^m V_m \;\le\; \det L \,. \tag{1.4}$$

Die linke Seite ist das Volumen einer Kugel mit Radius $\lambda_1/2$, die rechte Seite das Volumen des von den Basisvektoren aufgespannten Parallelepipeds. Zentriert man in jedem Gitterpunkt eine Kugel vom Radius $\lambda_1/2$, so hat der Schnitt von jeweils zweien das $m$-dimensionale Volumen 0. Hängt man an jeden Gitterpunkt ein von den Basisvektoren aufgespanntes Parallelepiped, so hat der Schnitt von jeweils zweien auch das Volumen 0. Ihre Vereinigung ist jedoch ganz span$L$, also gilt Ungleichung 1.4. Unter Verwendung der linken Ungleichung aus dem Lemma folgt jetzt die Behauptung:

$$\lambda_1^m \;\le\; \frac{2^m \det L}{V_m} \;\le\; m! \prod_{i=1}^{m} F_i(b_i) \;. \qquad \square$$

Ungleichung 1.4 wird auch als „erster Satz von Minkowski“ bezeichnet. Es gilt sogar der noch schärfere „zweite Satz von Minkowski“:

$$\frac{\det L}{m!} \;\le\; \frac{\lambda_1 \cdots \lambda_m}{2^m}\, V_m \;\le\; \det L \;. \tag{1.5}$$

Daraus folgt zusammen mit dem Lemma eine Verschärfung der oberen Schranke von Satz 4:

**Satz 6.** *Für jede Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$\frac{1}{m!} \prod_{i=1}^{m} F_i(b_i) \;\le\; \lambda_1 \cdots \lambda_m \;\le\; m! \prod_{i=1}^{m} F_i(b_i) \;.$$

Wir benötigen zum Beweis von Satz 12 im nächsten Abschnitt noch die folgende schwache Monotonieeigenschaft der $\kappa$-Konstanten:

**Lemma 7.** *Für $m \ge 1$ ist $\kappa_{m+1}^{m+1} \;\ge\; 2\,\kappa_m^m$ .*

**Beweis**. Zu jedem $\delta < 1$ gibt es eine geordnete Basis $b_1, \ldots, b_m$ eines Gitters $L$ und eine Norm $\|\cdot\|$, so daß $\lambda_1(L)^m \,/\, \prod_i F_i(b_i) \;\ge\; \delta\kappa_m^m$ . Wir können ohne Einschränkung der Allgemeinheit annehmen, daß $L \subset \operatorname{span}\{e_1, \ldots, e_m\} \subset \mathbb{R}^{m+1}$ . Sei $b \in L$ ein kürzester Gittervektor mit Norm $\lambda_1$ . Wir konstruieren eine Basis $\tilde{b}_1, \ldots, \tilde{b}_{m+1}$ eines Gitters $\tilde{L} \subset \mathbb{R}^{m+1}$ durch $\tilde{b}_i = b_i$ für $i = 1, \ldots, m$ und $\tilde{b}_{m+1} = \frac{\lambda_1}{2} e_{m+1} + \frac{1}{2} b$ und definieren auf $\mathbb{R}^{m+1} = \operatorname{span}\{e_1, \ldots, e_{m+1}\}$ eine Norm $\|\cdot\|_\sim$ durch

$$\| \sum_{i=1}^{m+1} x_i e_i \|_\sim := \| \sum_{i=1}^{m} x_i e_i \| \;+\; |x_{m+1}| \;.$$

Dann gilt einerseits $\tilde{F}_{m+1}(\tilde{b}_{m+1}) = \lambda_1/2$ . Andererseits gilt für jeden Gittervektor $\tilde{x} = \sum_{i=1}^{m+1} x_i \tilde{b}_i \in \tilde{L} - 0$ , daß $\| \tilde{x} \|_\sim \ge \lambda_1$ . Dies ist offensichtlich für $|x_{m+1}| \ne 1$ .

Für $|x_{m+1}| = 1$ ist $\| \tilde{x} \|_\sim = \| x \pm \frac{1}{2} b \| + \frac{\lambda_1}{2} \geq \lambda_1$ , da $x = \sum_{i=1}^m x_i b_i \in L - 0$ . Folglich ist $\lambda_1(\tilde{L}) = \lambda_1(L) = \lambda_1$, also gilt

$$\kappa_{m+1}^{m+1} \;\geq\; \frac{\lambda_1^{m+1}}{\tilde{F}_1(\tilde{b}_1) \cdots \tilde{F}_{m+1}(\tilde{b}_{m+1})} \;\geq\; \delta\, \kappa_m^m \cdot 2 \; .$$

Daraus folgt das Lemma mit $\delta \to 1$ . □

Die Höhenfunktionen führen zu einem stärkeren Reduktionsbegriff, der auf Hermite, Korkine und Zolotarev zurückgeht:

**Definition 8.** *Eine Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ heißt* Hermite–reduziert, *wenn für $i = 1, \ldots, m$ gilt*

- $F_i(b_i) \;=\; \min\{F_i(b) \mid b \in \mathbb{Z}b_i + \ldots + \mathbb{Z}b_m - 0\}$
- $F_j(b_i) \;\leq\; F_j(b_i \pm b_j)$ *für alle $j < i$ .*

Die folgende Überlegung zeigt, daß jedes Gitter eine Hermite–reduzierte Basis besitzt. Sei eine beliebige Gitterbasis $\tilde{b}_1, \ldots, \tilde{b}_m$ gegeben. Da die $i$-te Höhe eine Norm auf $\mathrm{span}\{\tilde{b}_i, \ldots, \tilde{b}_m\}$ ist, gibt es einen minimalen Gittervektor $b_i$ in $\mathbb{Z}\tilde{b}_i + \ldots + \mathbb{Z}\tilde{b}_m$ . Die Vektoren $b_1, \ldots, b_m$ bilden eine Basis des Gitters. Andernfalls gäbe es nämlich einen Gittervektor $b = \alpha_1 b_1 + \ldots + \alpha_i b_i$ mit $0 < \alpha_i < 1$, für den $F_i(b) = \alpha_i F_i(b_i) < F_i(b_i)$ gelte, im Widerspruch zur Minimalität von $F_i(b_i)$ . Lovasz und Scarf haben für allgemeine Normen gezeigt [LS92], daß die Normen der Basisvektoren einer solchen Basis die sukzessiven Minima in folgender Güte approximieren:

**Satz 9.** *Für jede Hermite–reduzierte Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$\frac{2}{i+1} \;\leq\; \| b_i \| \,/\, \lambda_i \;\leq\; \frac{i+1}{2} \; .$$

## 1.3 Blockreduzierte Basen

Man kennt selbst für die euklidische Norm keine Algorithmen, die eine gegebene Basis in eine Hermite–reduzierte Basis desselben Gitters in polynomial in der Dimension $m$ vielen Schritten transformieren. Wir haben in der Einleitung gesehen,

daß für die $l_\infty$-Norm das Entscheidungsproblem, ob ein durch eine Basis gegebenes Gitter einen Vektor von Norm 1 enthält, sogar NP–vollständig ist. Somit ist nicht zu hoffen, daß es Polynomialzeitalgorithmen zur Hermite–Reduktion gibt. Wir suchen Reduktionsbegriffe, für die sich reduzierte Basen effizient berechnen lassen. Von Schnorr stammt der Begriff blockreduzierter Basen [S87] , der sich in der Praxis bewährt [SE91, Co&a92]. Sei $\beta \geq 2$ eine ganze Zahl, die .

**Definition 10.** *Eine Basis* $b_1, \ldots, b_m \in \mathbb{R}^n$ *heißt* $\beta$-blockreduziert, *wenn für* $i = 1, \ldots, m$ *gilt*

- $F_i(b_i) \;=\; \min\{F_i(b) \mid b \in \mathbb{Z}b_i + \ldots + \mathbb{Z}b_{\min(i+\beta-1,m)} - 0\}$
- $F_j(b_i) \;\leq\; F_j(b_i \pm b_j)$ *für alle* $j < i$ .

Für $\beta = 1$ ist die erste Bedingung leer. Wir nennen eine solche Basis, die die zweite Bedingung erfüllt, *längenreduziert* . Eine Basis $b_1, \ldots, b_m$ ist genau dann $\beta$-blockreduziert, wenn sie längenreduziert ist und alle Blöcke $b_i, \ldots, b_{i+j}$ von jeweils $j+1$ aufeinanderfolgenden Vektoren Hermite–reduziert sind bezüglich der Norm $F_i$ für $j < \beta$, $i+j \leq m$ . Im Falle $\beta = m$ sind blockreduzierte Basen Hermite–reduziert. Im Falle $\beta = 2$ nennen wir sie *Lovász–reduziert* , weil solche Basen erstmals von Lenstra, Lenstra und Lovsaz eingeführt wurden [LLL82] und von Lovasz und Scarf für allgemeine Normen untersucht wurden [LS92].

Für den klassischen Fall der euklidischen Norm kennt man für $\beta$-blockreduzierte Basen $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ die Ungleichungen von Schnorr [S87, S94]

$$\| b_1 \|_2 \;\leq\; \alpha_{\beta,l_2}^{\frac{m-1}{\beta-1}} \lambda_1(L) \tag{1.6}$$

für $\beta - 1 \,|\, m - 1$ sowie

$$\frac{2}{\sqrt{i+3}}\, \gamma_\beta^{-\frac{i-1}{\beta-1}} \;\leq\; \| b_i \|_2 \,/\, \lambda_i \;\leq\; \frac{\sqrt{i+3}}{2}\, \gamma_\beta^{\frac{m-1}{\beta-1}} \tag{1.7}$$

Dabei bezeichnen wir mit $\alpha_{k,l_2}$ das Supremum des Quotienten $\| b_1 \| / F_k(b_k)$ über alle Hermite–reduzierten Gitterbasen $b_1, \ldots, b_k$ für die $l_2$-Norm. Sei $\alpha_k$ das Supremum desselben Quotienten über alle Hermite–reduzierten Gitterbasen $b_1, \ldots, b_k$ und alle Normen. Es gilt stets $\alpha_k \leq \alpha_{k+1}$ . Ist nämlich die Basis $b_1, \ldots, b_k \in \mathrm{span}\{e_1, \ldots, e_k\} \subset \mathbb{R}^{k+1}$ Hermite–reduziert bezüglich der Norm $\| \cdot \|$, so ist die Basis $b_0, \ldots, b_k \in \mathbb{R}^{k+1}$ mit $b_0 = \lambda_1 e_{k+1}$ Hermite–reduziert bezüglich der Norm $\| \sum_{i=0}^{m} x_i b_i \|_\sim := \max(\| \sum_{i=1}^{m} x_i b_i \|, |x_0| \lambda_1)$ . Wir verallgemeinern zunächst Ungleichung 1.6 auf den Fall beliebiger Normen:

**Satz 11.** *Für jede $\beta$-blockreduzierte Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$\| b_1 \| \;\leq\; \alpha_\beta^{\lceil \frac{m-1}{\beta-1} \rceil} \lambda_1(L) \; .$$

**Beweis**. Bezeichne $h_i := F_i(b_i)$ für $i = 1, \ldots, m$, und sei $\mu \in \{1, \ldots, m\}$ so gewählt, daß $h_\mu = \min h_i$ . Aus Satz 4 folgt dann $h_\mu \leq \lambda_1$ . Die Basen $b_i, \ldots, b_{i+j}$ sind Hermite–reduziert bezüglich der Norm $F_i$ für $0 \leq j < \beta,\ i + j \leq m$ . Wegen der Monotonie der $\alpha_k$ gelten die Ungleichungen

$$h_i \;\leq\; \alpha_\beta \, h_{i+j}$$

für $0 \leq j < \beta,\ i + j \leq m$ . Somit ist

$$h_1 \;\leq\; \alpha_\beta \, h_{1+(\beta-1)} \;\leq\; \ldots \;\leq\; \alpha_\beta^{\lfloor \frac{\mu-1}{\beta-1} \rfloor} \, h_{1+\lfloor \frac{\mu-1}{\beta-1} \rfloor (\beta-1)} \;\leq\; \alpha_\beta^{\lceil \frac{\mu-1}{\beta-1} \rceil} \, h_\mu \; . \qquad \square$$

Die Güte des Satzes hängt von der unbekannten Größe $\alpha_\beta$ ab. Aus Satz 19 dieses Kapitels folgt $\alpha_2 = 2$. Für $k \geq 2$ gilt die folgende Abschätzung von Schnorr [S87, S94p]:

$$\alpha_{k,l_2} \leq \; k^{\frac{1+\log k}{2}} \; , \qquad \alpha_k \;\leq\; k(k-1)^{\log(k-1)} \; . \tag{1.8}$$

Einsetzen dieser Schranken in Ungleichung 1.6 bzw. Satz 11 ergibt die Schranken $\| b_1 \| \leq m^{O(\frac{m \log m}{\beta})} \lambda_1$ , die schwächer sind als Ungleichung 1.7. Wir verallgemeinern Ungleichung 1.7 für beliebige Normen:

**Satz 12.** *Für jede $\beta$-blockreduzierte Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$\left.\begin{array}{ll} 2 \leq i \leq m : & \frac{4}{i+1} \; \kappa_\beta^{-2 \, \frac{i-1}{\beta-1}} \\ 1 \leq i \leq \beta \; : & \frac{2}{i+1} \end{array}\right\} \;\leq\; \frac{\| b_i \|}{\lambda_i} \;\leq\; \left\{\begin{array}{ll} \frac{i+1}{4} \; \kappa_\beta^{2 \, \frac{m-1}{\beta-1}} \; : & 1 \leq i \leq m \\ \frac{i+1}{2} & : m - \beta + 1 \leq i \leq m \end{array}\right. .$$

**Beweis**. Der Beweis gliedert sich zunächst in zwei Lemmata, aus denen die rechte Ungleichung des Satzes für $i = 1$ folgt:

**Lemma 13.** *Für jede $\beta$-blockreduzierte Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$\| b_1 \| \leq \left( \prod_{i=1}^{\beta-1} \kappa_i^i \right)^{\frac{2}{\beta(\beta-1)}} \kappa_\beta^{2\frac{m-\beta}{\beta-1}} \max_{i=m-\beta+1}^{m-1} F_i(b_i) .$$

**Beweis**. Bezeichne $h_i = F_i(b_i)$ die $i$-te Höhe. Nach Definition gelten die Ungleichungen

$$\begin{aligned} h_1^i &\leq \kappa_i^i \, h_1 \cdots h_i , \quad \text{für } i = 1, ..., \beta - 1 \text{ und} \\ h_i^\beta &\leq \kappa_\beta^\beta \, h_i \cdots h_{i+\beta-1} , \quad \text{für } i = 1, ..., m - \beta . \end{aligned}$$

Multiplikation dieser Gleichungen ergibt

$$h_1^{\binom{\beta+1}{2}} h_2^\beta \cdots h_{m-\beta}^\beta \leq \kappa_1^1 \, \kappa_2^2 \cdots \kappa_{\beta-1}^{\beta-1} \, \kappa_\beta^{\beta(m-\beta)} \, h_1^\beta h_2^\beta \cdots h_{m-\beta}^\beta h_{m-\beta+1}^{\beta-1} \cdots h_{m-1}^1 ,$$

woraus durch Kürzen folgt

$$\begin{aligned} h_1^{\binom{\beta}{2}} &\leq \kappa_1^1 \, \kappa_2^2 \cdots \kappa_{\beta-1}^{\beta-1} \, \kappa_\beta^{\beta(m-\beta)} \quad h_{m-\beta+1}^{\beta-1} \cdots h_{m-1}^1 \\ &\leq \kappa_1^1 \, \kappa_2^2 \cdots \kappa_{\beta-1}^{\beta-1} \, \kappa_\beta^{\beta(m-\beta)} \left( \max_{i=m-\beta+1}^{m-1} F_i(b_i) \right)^{\binom{\beta}{2}} . \end{aligned}$$

□

**Lemma 14.** *Für jede $\beta$-blockreduzierte Basis $b_1, \ldots, b_m \in \mathbb{R}^n$ eines Gitters $L$ gilt*

$$\| b_1 \| \leq \left( \prod_{i=1}^{\beta-1} \kappa_i^i \right)^{\frac{2}{\beta(\beta-1)}} \kappa_\beta^{2\frac{m-\beta}{\beta-1}} \lambda_1(L) .$$

**Beweis**. Die Behauptung folgt mittels Induktion über $m$ aus Lemma 13. Für $m = \beta$ gilt die Behauptung nach Definition 10. Sei jetzt $b = r_1 b_1 + \ldots + r_m b_m$ ein kürzester Gittervektor. Für $r_m = 0$, folgt die Behauptung aus der Induktionsannahme. Andernfalls gilt für $m - \beta + 1 \leq i \leq m$ die Ungleichung

$$\lambda_1(L) = \| b \| \geq F_i(b) \geq F_i(b_i) ,$$

und die Behauptung folgt aus Lemma 13. □

Durch sukzessive Anwendung von Lemma 7 folgt aus Lemma 14 die 1. Ungleichung des Satzes:

$$\| b_1 \| \;\leq\; \frac{1}{2}\, \kappa_\beta^{\,2\,\frac{m-1}{\beta-1}}\; \lambda_1(L)\,. \tag{1.9}$$

**Beweis der rechten Ungleichung von Satz 12.** Für jedes $j \leq m$ ist die Basis $b_j, \ldots, b_m$ $\beta$-blockreduziert bezüglich der Norm $F_j$. Also gilt $F_j(b_j) = \lambda_{1,F_j}(\mathbb{Z}b_j + \ldots + \mathbb{Z}b_m)$ für $m - \beta + 1 \leq j \leq m$ und, wegen Ungleichung 1.9:

$$F_j(b_j) \;\leq\; \tfrac{1}{2}\, \kappa_\beta^{\,2\,\frac{m-1}{\beta-1}}\; \lambda_{1,F_j}(\mathbb{Z}b_j + \ldots + \mathbb{Z}b_m)\,.$$

für $1 \leq i \leq m$. Weiterhin gilt $\lambda_{1,F_j}(\mathbb{Z}b_j + \ldots + \mathbb{Z}b_m) \leq \lambda_j(L) \leq \lambda_i(L)$ für $j \leq i$. Die obere Schranke folgt jetzt aus der Ungleichung

$$\| b_i \| \;\leq\; F_i(b_i) \;+\; \frac{1}{2}\sum_{j=1}^{i-1} F_j(b_j)\;. \tag{1.10}$$

Wir zeigen die Gültigkeit von Ungleichung 1.10 für jede längenreduzierte Basis: Sei $F_j(b_i + \xi_0 b_j) = \min_{\xi\in\mathbb{R}} F_j(b_i + \xi b_j) = F_{j+1}(b_i)$. Aus $F_j(b_i) \leq F_j(b_i \pm b_j)$ für $j < i$ folgt

$$\begin{aligned} F_j(b_i) &= \min_{\mu\in\mathbb{Z}} F_j(b_i + \mu b_j) \\ &\leq F_j(b_i + \lfloor \xi_0 \rceil b_j) \\ &= F_j\left(b_i + \xi_0 b_j \;+\; (\lfloor \xi_0 \rceil - \xi_0)\; b_j\right) \\ &\leq F_{j+1}(b_i) + \frac{1}{2} F_j(b_j)\;, \end{aligned}$$

da für $F_j$ die Dreieckungleichung gilt. Sukzessive Anwendung dieser Ungleichung für $j = 1, \ldots, i-1$ beweist Ungleichung 1.10.

**Beweis der linken Ungleichung von Satz 12.** Nach Definition der sukzessiven Minima und Ungleichung 1.10 gilt

$$\lambda_i \;\leq\; \max_{j=1}^{i} \| b_j \| \;\leq\; \frac{i+1}{2}\, \max_{j=1}^{i} F_j(b_j)\;.$$

Die Behauptung folgt jetzt aus den Ungleichungen $F_j(b_j) \le \| b_i \|$ für $i - \beta + 1 \le j \le i$ und

$$F_j(b_j) \;\le\; \tfrac{1}{2}\, \kappa_\beta^{\,2\frac{i-j}{\beta-1}} \;\| b_i \| \qquad \text{für } 1 \le j < i \;. \tag{1.11}$$

Die Ungleichungen 1.11 folgen unmittelbar: Jede Basis $b_j, \ldots, b_i$ ist nämlich $\beta$-blockreduziert bezüglich der Norm $F_j$. Also beschränkt Lemma 13 die vorderen Höhen für $1 \le j \le i - \beta + 1$ in folgender Weise durch die hinteren:

$$F_j(b_j) \;\le\; \tfrac{1}{2}\, \kappa_\beta^{\,2\frac{i-j}{\beta-1}} \max_{h=i-\beta+1}^{i-1} F_h(b_h) \;.$$

Die hinteren Höhen erfüllen für $i - \beta + 1 \le j \le i$ nach Definition 10 die Ungleichungen

$$F_j(b_j) \;\le\; F_j(b_i) \;\le\; \| b_i \| \;. \qquad \square$$

**Bemerkungen. A.** Für $\beta = 2$ folgt aus Satz 12 mit $\kappa_2 = \sqrt{2}$ (Ungl. 1.2) die Schranke $\| b_1 \| \le 2^{m-2}\lambda_1$ . Das verschärft sogar leicht die Schranke $\| b_1 \| \le 2^{m-1}\lambda_1$ von Lovasz und Scarf [LS92], Th. 2.

**B.** Durch Anwendung von Ungleichung 1.14 auf das von $b_i, \ldots, b_m$ erzeugte Gitter bezüglich der Norm $F_i$ folgen für die Höhen einer $\beta$-blockreduzierten Basis die Ungleichungen

$$F_i(b_i)/\lambda_i \;\le\; \begin{cases} \tfrac{1}{2}\, \kappa_\beta^{\,2\frac{m-i}{\beta-1}} & \text{für } i \le m - \beta \,, \\ 1, & \text{für } i > m - \beta \,. \end{cases} \tag{1.12}$$

Diese Ungleichungen gelten unabhängig von den Längenreduktionsbedingungen $F_j(b_j) \;\le\; F_j(b_i \pm b_j)$ .

**C.** Es gibt für jede Norm $\| \cdot \|$ ein Ellipsoid $E$, so daß in der korrespondierenden Norm $\| \cdot \|_E$ für alle $x \in \mathbb{R}^n$ gilt [Jo48]

$$\| x \|_E \;\le\; \| x \| \;\le\; \sqrt{n} \| x \|_E \;.$$

Daraus folgt für für eine bezüglich $\|\cdot\|_E$ $\beta$-blockreduzierte Basis $b_1, \ldots, b_m$

$$\frac{1}{n}\ \frac{2}{\sqrt{i+3}}\ \gamma_\beta^{-\frac{i-1}{\beta-1}}\ \leq\ \|\,b_i\,\|\,/\,\lambda_{i,\|\cdot\|}\ \leq\ n\ \frac{\sqrt{i+3}}{2}\ \gamma_\beta^{\frac{m-1}{\beta-1}}\ .$$

**D.** Unsere oberen Schranken für die $\kappa$-Konstanten aus Satz 4 liefern im Vergleich etwas schwächere absolute Schranken als die absoluten oberen Schranken für die Hermite-Konstanten. Verwendet man im Beweis von Satz 12 zur Abschätzung des Produktes der $\kappa$-Konstanten aus Lemma 14 die obere Schranke aus Satz 4, so erhält man für alle $\beta$-blockreduzierten Basen $b_1, \ldots, b_m$ die Ungleichungen

$$\tfrac{2}{\sqrt{e}}\ \frac{\beta}{i+1}\ \left(\beta!^{\frac{2}{\beta}}\right)^{-\frac{i-1}{\beta-1}}\ <\ \|\,b_i\,\|\,/\,\lambda_i\ <\ \tfrac{\sqrt{e}}{2}\ \frac{i+1}{\beta}\ \left(\beta!^{\frac{2}{\beta}}\right)^{\frac{m-1}{\beta-1}}\ . \tag{1.13}$$

**Beweis der Ungleichungen 1.13.** Sei $P(\beta) = \left(\frac{1!\cdots(\beta-1)!}{\beta!^{\beta-1}}\right)^{\frac{2}{\beta(\beta-1)}}$. Dann ist

$$\begin{aligned}
-\log P(\beta) &= \tfrac{2}{\beta(\beta-1)}\ \sum_{i=2}^{\beta}(i-1)\ \log i \\
&\geq \tfrac{2}{\beta(\beta-1)}\ \left(\int_1^{\beta}(x-1)\ \log\ x\ dx\ +\ \sum_{i=2}^{\beta}\ \tfrac{(i-1)\ \log\ i\ -\ (i-2)\ \log(i-1)}{2}\right) \\
&= \log\ \beta\ -\ \frac{\log\ \beta}{\beta(\beta-1)}\ -\ \frac{1}{2}\ +\ \frac{3}{2\beta}\ >\ \log\ \beta\ -\ \frac{1}{2}\ ,
\end{aligned}$$

und somit ist $P(\beta) < \sqrt{e}/\beta$. Also folgt aus Lemma 14 unter Verwendung von $\kappa_k^k \leq k!$ (Satz 4):

$$\|\,b_1\,\|\ \leq\ P(\beta)\ \left(\beta!^{\frac{2}{\beta}}\right)^{\frac{m-1}{\beta-1}}\ \lambda_1(L)\ <\ \frac{\sqrt{e}}{\beta}\ \left(\beta!^{\frac{2}{\beta}}\right)^{\frac{m-1}{\beta-1}}\ \lambda_1(L)\ . \tag{1.14}$$

Die Ungleichungen 1.13 folgen jetzt durch Verwendung von Ungleichung 1.14 anstelle von Ungleichung 1.9 im Beweis von Satz 12. □

**E.** Für $i > e^{-\frac{1}{2}}\,\beta\,/\,\log\beta$ lassen sich die Schranken 1.13 folgendermaßen verschärfen: Wir setzen Ungleichung 1.12 in Ungleichung 1.10 ein, werten das Resultat mit der geometrischen Summenformel aus und erhalten

$$\begin{aligned}
\|\,b_i\,\| &\leq P(\beta)\ B^m\ \frac{1+(B-2)/B^i}{2\,(B-1)}\ \lambda_i \\
&\leq \frac{P(\beta)}{2\,(B-1)}\ B^m\ \lambda_i\,,
\end{aligned}$$

wobei $B = \beta!^{\frac{2}{\beta(\beta-1)}}$ . Wir wenden die geometrische Summenformel in gleicher Weise im Beweis der linken Unlgeichung an. Mit etwas Mühe erhalten wir die Abschätzung $\frac{P(\beta)}{B-1} < \frac{1}{\log \beta}$ , woraus dann folgt:

$$2 \log \beta \ \left(\beta!^{\frac{2}{\beta}}\right)^{-\frac{i}{\beta-1}} \ < \ \| \, b_i \, \| \, / \, \lambda_i \ < \ \frac{1}{2 \log \beta} \ \left(\beta!^{\frac{2}{\beta}}\right)^{\frac{m}{\beta-1}} \ . \tag{1.15}$$

**F.** Für $\beta > 2\,(m-\beta)\,\log\,(m-\beta) + 1$ lassen sich die Ungleichungen 1.9 bzw. 1.14, und damit die bisher genannten Schranken folgendermaßen verschärfen: Für jedes $k \leq \beta$ gelten die Ungleichungen

$$h_1^i \ \leq \ \kappa_k^k \ h_1 \cdots h_i \, , \quad \text{für } i = 1, ..., k-1 \quad \text{und}$$

$$h_i^k \ \leq \ \kappa_k^k \ h_i \cdots h_{i+k-1} \, , \quad \text{für } i = 1, ..., m-\beta \, .$$

Daraus erhalten wir wie im Beweis von Lemma 13 und Ungleichung 1.14 die Ungleichung

$$\| \, b_1 \, \| \ < \ \frac{\sqrt{e}}{k} \ \left(k!^{\frac{2}{k}}\right)^{\frac{m-\beta+k-1}{k-1}} \ \lambda_1(L) \ \leq \ (m-\beta+1)^{1+o(1)} \ \lambda_1(L) \ , \tag{1.16}$$

wobei im letzten Schritt $k = \lceil 2\,(m-\beta)\,\log\,(m-\beta)+1 \rceil$ gesetzt wurde.

**G.** Ist $\beta$ ein konstanter Bruchteil von $m$, so ist die Schranke von Satz 12 polynomial in $m$.

**Algorithmen zur Blockreduktion.** Für die euklidische Norm gibt es einen polynomialzeit–Algorithmus zur Berechnung einer *semi*–blockreduzierten Basis (für feste Blockgröße $\beta$) von Schnorr [S87]. An gleicher Stelle findet man einen effizienten Algorithmus zur Blockreduktion, der von Schnorr und Euchner praktisch verbessert und zum Lösen von Subset–Sum–Problemen erfolgreich eingesetzt wurde [SE91]. Ritter entwickelte und analysierte Varianten des Algorithmus zur Berechnung eines kürzesten Gittervektors in anderen Normen, die besonders effizient für die $l_\infty$-Norm sind [R94], und die zu einem Blockreduktions–Algorithmus für beliebige Normen führen [KR94].

## 1.4 Gaußreduzierte Basen

Wir verstehen die Geometrie in niedrigen Dimensionen wesentlich besser als in höheren. Vermutlich studierten deshalb Lagrange, Gauß und Dirichlet bei der Entwicklung der Reduktionsbegriffe für die euklidische Norm nur die Dimensionen zwei und drei. Wie wir sehen werden, überblicken wir in Dimension zwei auch für beliebige Normen den Gaußschen Algorithmus und die in ihm auftretenden Begriffe.

Wir nutzen für beliebige Normen $\| \cdot \|$ häufig die Konvexität der Funktion $\| F(\xi) \|$ für jede Gerade $F : \mathbb{R} \to \mathbb{R}^n : \xi \mapsto \xi a + b$ in folgender Schlußweise:

**Lemma 15.** *Sei $F : \mathbb{R} \to \mathbb{R}^n$ eine Gerade in $\mathbb{R}^n$ und $\xi_1, \xi_2, \eta_1, \eta_2 \in \mathbb{R}$ mit*

$$\xi_1 < \xi_2,\ \eta_1 < \eta_2, \quad \xi_1 \le \eta_1,\ \xi_2 \le \eta_2\ .$$

*Dann folgt aus $\| F(\xi_1) \| \le \| F(\xi_2) \|$ die Ungleichung $\| F(\eta_1) \| \le \| F(\eta_2) \|$ und aus $\| F(\xi_1) \| < \| F(\xi_2) \|$ die Ungleichung $\| F(\eta_1) \| < \| F(\eta_2) \|$ .*

Meistens verwenden wir Lemma 15 im Fall $\xi_1 = 0$ und $\xi_2 = 1$. Außerdem benötigen wir die folgende elementare Schlußweise:

**Lemma 16.** *Sei $M$ eine abgeschlossene Menge in $\mathbb{R}^n$ und $0 \notin M$. Dann nimmt die Norm ihr Minimum auf $M$ auf dem Rand von $M$ an.*

Sei im folgenden $L = \mathbb{Z}a + \mathbb{Z}b \subset \mathbb{R}^n$ ein Gitter vom Rang 2 mit Basis $a, b$. Wir definieren reduzierte und wohlgeordnete Basen, die für die Analyse des verallgemeinerten Gauß–Algorithmus im nächsten Kapitel wesentlich sind. Damit werden diese auf Gauß [Ga1801] und Vallee [Va91] zurückgehenden Begriffe für beliebige Normen verallgemeinert.

**Definition 17.** *Eine Gitterbasis $(a, b)$ heißt* Gauß–reduziert *, falls*

$$\| a \|,\ \| b \| \le \| a - b \| \le \| a + b \|$$

*und , falls*

$$\| a \| \le \| a - b \| < \| b \| \ .$$

Mittels Lemma 15 können wir von $\| a - b \| < \| b \|$ auf

$$\| b \| < \| \eta a + b \| \quad \forall \eta > 0 \tag{1.17}$$

schließen. Folglich ist $(a, b)$ genau dann wohlgeordnet, wenn

$$\| a \| \leq \| a - b \| < \| b \| < \| a + b \| .$$

Proposition 20 gibt einen algorithmischen Beweis dafür, daß jedes Gitter eine reduzierte Basis besitzt. Wir beweisen zunächst, daß die Norm der Vektoren einer reduzierten Basis die sukzessiven Minima in der gegebenen Norm sind.

**Satz 18.** *Sei $(a, b)$ eine Gauß–reduzierte Basis. Dann sind die Normen der Basisvektoren die sukzessiven Minima des Gitters $L = \mathbb{Z}a + \mathbb{Z}b$ .*

**Beweis**. Sei o.B.d.A. $\| a \| \leq \| b \|$ . Die Aussage des Satzes ist:

$$\| a \| \leq \| ra + sb \| \quad \text{für alle } (r, s) \in \mathbb{Z}^2 - \{(0, 0)\} ,$$

$$\| b \| \leq \| ra + sb \| \quad \text{für alle } r \in \mathbb{Z}, s \in \mathbb{Z} - \{0\} .$$

Diese Ungleichungen folgen aus den Ungleichungen:

$$\begin{aligned} \| a \| &\leq \| b \| , \\ \| a \| &\leq \| ra \| \quad \text{für alle } r \in \mathbb{Z} - \{0\} , \\ \| b \| &\leq \| \xi a + \eta b \| \quad \text{für alle } \xi, \eta \in \mathbb{R} \text{ mit } |\xi|, |\eta| \geq 1 . \end{aligned} \tag{1.18}$$

Es genügt also, Ungleichung 1.18 zu zeigen. Hierfür beweisen wir folgende

**Behauptung.** *In jeder der vier schattierten Flächen in Abbildung 1.3 nimmt die Norm ihr Minimum in den Punkten $\pm a \pm b$ an.*

Ungleichung 1.18 folgt direkt aus der Behauptung und den Reduktionsbedingungen

$$\| b \| \leq \| a \pm b \| .$$

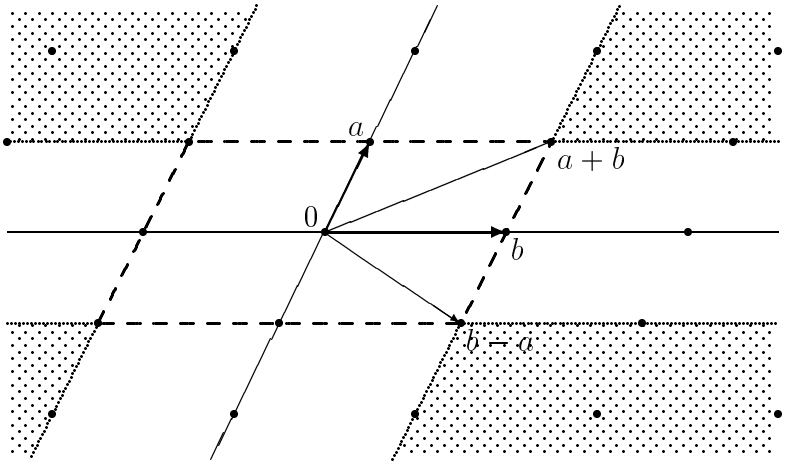


Abbildung 1.3: Reduzierte Basis $(a, b)$

**Beweis der Behauptung.** Jede gestrichelte Strecke in der Abbildung enthält drei Gitterpunkte, von denen der jeweils mittlere minimale Norm hat:

$$\begin{aligned} \| \pm a - b \| &\geq \| \pm a \| \leq \| \pm a + b \| \\ \| -a \pm b \| &\geq \| \pm b \| \leq \| a \pm b \| \end{aligned}$$

Mit Lemma 15 folgt für alle $\xi \geq 1$:

$$\begin{aligned} \| \pm a \pm \xi b \| &\geq \| \pm a \pm b \| \geq \| \pm a \| \,, \\ \| \pm \xi a \pm b \| &\geq \| \pm a \pm b \| \geq \| \pm b \| \,. \end{aligned}$$

Folglich minimieren die Punkte $\pm a \pm b$ die Norm auf den punktierten Linien, also auf dem Rand der schattierten Flächen. Nach Lemma 16 nimmt die Norm ihr Minimum in jeder schattierten Fläche auf dem Rand an. Das beweist die Behauptung. □

Aus Satz 18 folgt, daß Gauß–reduzierte Basen (Definition 17) bis auf die Reihenfolge der Vektoren auch im Sinne von Minkowski, von Hermite, Korkine, Zolotarev, von Schnorr und von Lovasz (Definitionen 2, 8 und 10) reduziert sind. Wir sprechen deshalb in Dimension 2 lediglich von *reduzierten* Basen. Der folgende Satz beschreibt die Normen, für die eine gegebene Gitterbasis vom Rang zwei reduziert ist, indem eine die Kreisscheibe $B_{\lambda_1,0}$ umfassende Kurve beschrieben wird.

**Satz 19.** *Sei* $(a, b)$ *eine reduzierte Basis mit* $\lambda_1 = \| a \| \leq \| b \| = \lambda_2$ . *Dann gilt* $\| x \| \geq \| a \|$ *für alle Punkte* $x$ *auf der Kurve*

$$\begin{aligned} K_{a,b} &= \pm\{b + t(b-a),\, (a-b) - tb,\, (a-b) + ta \mid 0 \leq t \leq 1\} \\ &\cup \pm\{2a - tb,\, 2a + t(b-a) \mid 0 < t \leq 1\} \\ &\cup \pm\{sa + tb \mid t \geq 1 \text{ und } (s - \frac{1}{2})^2 + (t - \frac{1}{2})^2 = \frac{1}{2}\} \,. \end{aligned}$$

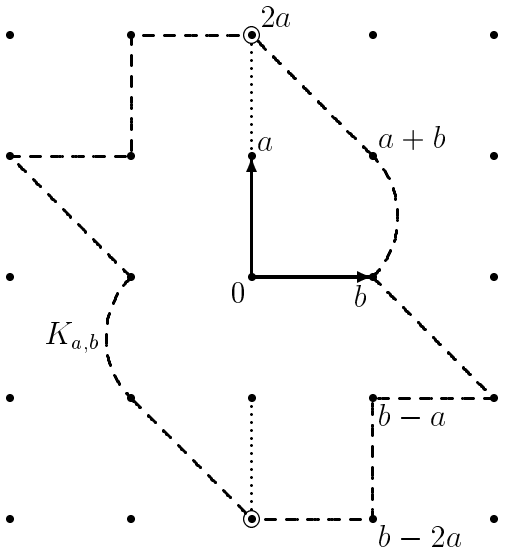


*Abbildung 1.4: Grenze für $B_{\lambda_1,0}$ bei reduzierter Basis*

Der Satz ist scharf in dem Sinne, daß es zu jeder geordneten Basis $(a, b)$ für jeden Punkt $x \in K_{a,b}$ eine Norm gibt, so daß $(a, b)$ reduziert ist mit $\| a \| \leq \| b \|$ und $\| x \| = \| a \|$. Man sieht dies an den Beispielen in Abbildung 1.5. Mithin ist Kurve in Abbildung 1.4 der Rand der Vereinigung aller Kugeln $B_{\lambda_1,0}$ für alle Normen, für die $(a, b)$ reduziert ist mit $\| a \| \leq \| b \|$.

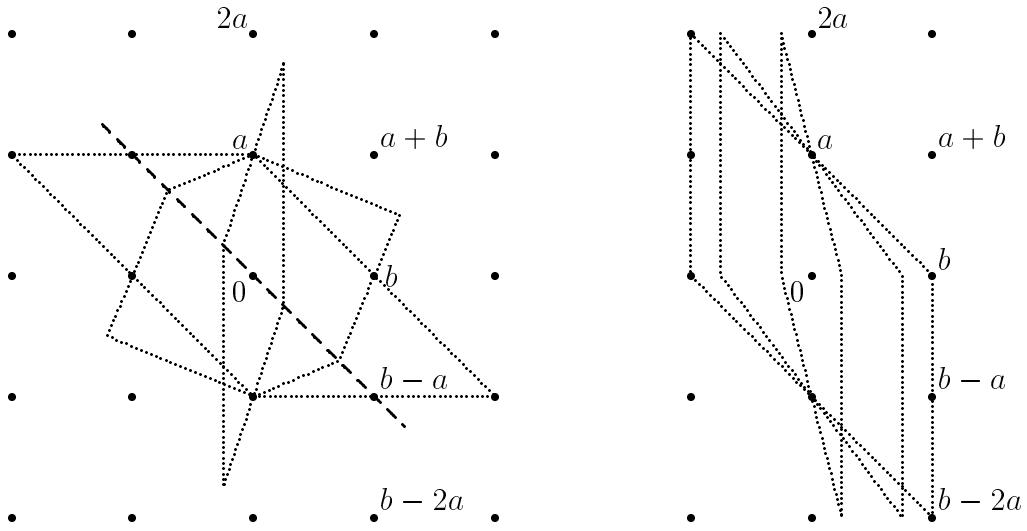


Abbildung 1.5: Extreme Beispiele für $B_{\lambda_1,0}$

Von den Punkten $sa + tb \in K_{a,b}$ mit $s + t > 1$ hat stets höchstens einer Norm $\lambda_1$ .

**Beweis**. Der Beweis gliedert sich in Fallunterscheidungen für die einzelnen Streckenabschnitte. Wir führen hier nur den schwierigsten Fall der beiden Kreissektoren aus (vgl. Abb. 1.6). Nehmen wir an, $x = sa + tb$, wobei $t > 1$, $(s - \frac{1}{2})^2 + (t - \frac{1}{2})^2 = \frac{1}{2}$, aber, im Widerspruch zum Satz: $\| x \| < \| a \|$ . Sei

$$\eta = \frac{t - s}{t - s + 1}, \quad \xi = \frac{t}{t - s + 1}, \quad x^+ = \xi(b + a), \quad x^- = \xi(b - a) \ .$$

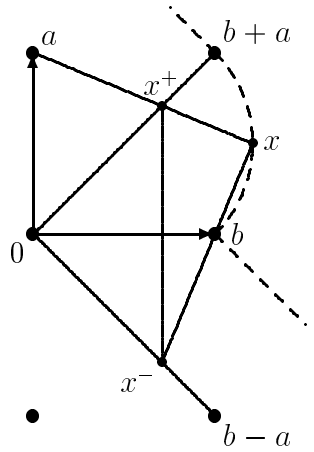


Abbildung 1.6: Abschätzung der Norm auf dem Kreisabschnitt

Aus der Reduktionsbedingung $\| b - a \| \leq \| b + a \|$ folgt $\| x^- \| \leq \| x^+ \|$ . Aus Lemma 15 folgt wegen $x^+ = (1-\eta)x + \eta a$ die Ungleichung $\| x^+ \| < \| a \|$, woraus $\| x^- \| < \| a \|$ folgt. Wir zeigen, daß $b$ auf der Gerade zwischen $x$ und $x^-$ liegt: Die Kreisgleichung ist gleichbedeutend mit $t^2 + s^2 = t + s$ . Nach Setzung von $\zeta := \frac{t+s-1}{t+s}$ verifiziert man leicht

$$(1 - \zeta)\, s - \zeta\, \xi = 0\,, \quad (1 - \zeta)\, t + \zeta\, \xi = 1\,,$$

was gleichbedeutend ist mit $(1 - \zeta)\, x + \zeta\, x^- = b$ . Aus Lemma 15 folgt nun $\| b \| \leq \max\{\| x \|, \| x^- \|\} < \| a \|$, im Widerspruch zur Voraussetzung des Satzes. □

**Einfache Folgerungen aus Satz 19.**

- Die Norm jedes Puktes $y \in \operatorname{span} L$ ist mindestens $\lambda_1 \sup\{\rho \,|\, y \in \rho K_{a,b}\}$ .
- Es gilt $\min_{\xi \in \mathbb{R}} \| b - \xi a \| \geq \frac{1}{2} \| a \|$ . Somit ist die in Satz 11 definierte Konstante $\alpha_2 = 2$ .

- Falls andere Gitterpunkte als $\{\pm a,\ \pm b,\ \pm a \pm b\}$ existieren mit Norm $\lambda_1$, so sind dies entweder $\pm(2b - a)$ oder $\pm(2a - b)$. Für beide Fälle ist die Kugel $B_{\lambda_1,0}$ in Abbildung 1.5 dargestellt.

- In jedem Gitter gibt es höchstens 8 Vektoren mit Norm $\lambda_1$. (In der Regel sind es 2. Hier sehen wir, daß das Besispiel $(\mathbb{Z}^2, \| \ . \ \|_\infty)$ bereits der „worst case“ ist.)

- Jedes Gitter hat eine reduzierte Basis $(a, b)$, so daß die Norm aller Gittervektoren außerhalb der konvexen Hülle von $\pm a \pm b$ mindestens $\frac{3}{2}\lambda_1$ ist.

# Kapitel 2

# Der verallgemeinerte Gauß–Algorithmus

Wir verallgemeinern den Gauß–Algorithmus von der euklidischen Norm auf eine beliebige Norm. Dabei wird die Gültigkeit und Schärfe der worst–case–Schranken für die Zahl der Iterationen aus Vallées Analyse [Va91] für beliebige Normen gezeigt. Die Kapitel 2 und 3 sind eine vertiefende Darstellung der neuen Arbeiten von Kaib und Schnorr. [Ka91, KS93]

## 2.1 Der Reduktionsschritt

Gauß entwickelte seinen Algorithmus aus dem Studium reduzierter quadratischer Formen [Ga1801] . Der Algotrithmus führt *Reduktionsschritte* der Art

$$(a, b) \;\mapsto\; (\varepsilon(b - \mu a), a)$$

aus, wobei der ganzzahlige *Reduktionskoeffizient* $\mu = \mu(a, b)$ so gewählt wird, daß die Norm $\| b - \mu a \|$ minimiert wird und $\varepsilon = \pm 1$ . Man kann den Algorithmus als natürliche Verallgemeinerung des zentrierten Euklidischen Algorithmus auf Vektoren auffassen. Bei Gauß galt stets $\varepsilon = 1$ . Er betrachtete nur die euklidische Norn, für die $\mu(a, b)$ eine nächste ganze Zahl zu $\frac{<a,b>}{<a,a>}$ ist. Wir wählen das Vorzeichen $\varepsilon = \varepsilon(a, b) = \pm 1$ einer Idee von Vallée [Va91] folgend so, daß die neue Basis wohlgeordnet oder reduziert ist. Außerdem betrachten wir beliebige Normen. Es kann vorkommen, daß $\varepsilon, \mu$ durch die eben genannten Bedingungen nicht eindeutig bestimmt sind. Es bezeichne $\mathrm{succ} : \mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{R}^n$ eine feste Funktion mit den Eigenschaften

1. $\text{succ}\,(a,b) \;=\; \varepsilon(b-\mu a)$ für ein $\mu \in \mathbb{Z},\ \varepsilon = \pm 1$,
2. $\| \,\text{succ}(a,b)\,\| \;=\; \min_{\mu\in\mathbb{Z}} \| \,b-\mu a\,\|$,
3. $\| \,\text{succ}(a,b)-a\,\| \;\leq\; \| \,\text{succ}(a,b)+a\,\|$ .

Wir nennen succ die *Nachfolgerfunktion*. Eine der beiden Möglichkeiten $\varepsilon(a,b) = \pm 1$ erfüllt stets Bedingung 3. Die Berechnung des Reduktionskoeffizienten $\mu(a,b)$ für andere Normen als die euklidische behandeln wir in Kapitel 3.

Wir beschreiben den (verallgemeinerten) Gauß–Algorithmus als Iteration der Nachfolgerfunktion auf den Basisvektoren:

**Der Gauß–Algorithmus.**

**INPUT:** $a,b \in \mathbb{R}^n$
**REPEAT**
$(a,b) := (\text{succ}\,(a,b)\,,a)$
**UNTIL** $\| \,b\,\| \;\leq\; \| \,a-b\,\|$
**OUTPUT:** $(a,b)$

**Proposition 20.**

1. *Jede Basis* $(a,b) = (\text{succ}\,(\bar a,\bar b),\,\bar a)$ *ist wohlgeordnet oder reduziert und*
2. *genau dann reduziert, wenn* $\| \,b\,\| \;\leq\; \| \,a-b\,\|$ .
3. *Der Algorithmus endet nach endlich vielen Iterationen.*

**Beweis**. Ad 1 und 2. Sei $(a,b) = (\varepsilon(\bar a-\mu\bar b),\,\bar a)$ . Aus Bedingung 2 folgt $\| \,a\,\| \;=\; \| \,\bar b-\mu\bar a\,\| \;\leq\; \| \,\bar b-(\mu-\varepsilon)\bar a\,\| \;=\; \| \,a-b\,\|$ . Wegen Bedingung 3 gilt $\| \,a-b\,\| \;\leq\; \| \,a+b\,\|$ . Ist jetzt $\| \,a-b\,\| \;<\; \| \,b\,\|$ , so ist $(a,b)$ wohlgeordnet. Andernfalls ist $(a,b)$ reduziert und der Algorithmus bricht ab.

Ad 3. In jeder beschränkten Menge gibt es nur endlich viele Gittervektoren. Wegen Aussagen 1 und 2 gilt entweder $\| \,\text{succ}\,(a,b)\,\| \;<\; \| \,a\,\|$ oder $(\text{succ}\,(a,b),\,a)$ ist reduziert und der Algorithmus bricht ab. □

## 2.2 Vorgänger einer wohlgeordneten Basis

Wir assoziieren mit der Eingabebasis die von den Reduktionsschritten $(a,b) := (\text{succ}\,(a,b),\,a)$ erzeugte Folge von Gitterbasen. Wir sehen an Proposition 20, daß

die letzte Basis dieser Folge reduziert ist und alle vorherigen wohlgeordnet sind. Wir nennen $(a,b) = (\mathrm{succ}\,(b,c),\, b)$ die *Nachfolgerbasis* von $(b,c)$ und $(b,c)$ eine *Vorgängerbasis* von $(a,b)$. Eine wohlgeordnete Basis hat genau eine Nachfolgerbasis, kann jedoch unendlich viele Vorgängerbasen haben. Das Ziel der nächsten Definition und des folgenden Lemma 22 ist, die Menge aller Vorgängerbasen einer wohlgeordneten Gitterbasis zu charakterisieren.

**Definition 21.** *Ein Vektor $c$ heißt* Vorgänger *der Basis $(a,b)$, wenn die Basis $(b,c)$ wohlgeordnet ist und $a = \mathrm{succ}\,(b,c)$.*

**Fakt.** Es sei $(a,b)$ eine wohlgeordnete Gitterbasis. Dann ist ein Vektor $c$ genau dann Vorgänger von $(a,b)$ wenn $(b,c)$ wohlgeordnet ist und $c = \varepsilon a + \mu b$ mit $\varepsilon = \pm 1,\ \mu \in \mathbb{Z}$.

Das folgende Lemma ist eine Verallgemeinerung von Lemma 1 aus Vallées Arbeit [Va91] für beliebige Normen.

**Lemma 22.** *Sei $(a,b)$ eine wohlgeordnete Gitterbasis. Ein Vektor $c$ ist Vorgänger von $(a,b)$ dann und nur dann, wenn $c = \varepsilon a + \mu b$ gilt, wobei entweder $\varepsilon = 1$, $\mu \geq 2$ oder $\varepsilon = -1$, $\mu \geq 3$ ist.*

**Bemerkung.** Ein interessanter Aspekt von Lemma 22 ist, daß die Menge der Vorgängerbasen einer wohlgeordneten Gitterbasis *unabhängig von der Norm* ist. Ist eine Gitterbasis $(a,b)$ wohlgeordnet für zwei verschiedene Normen, so sind die Mengen ihrer Vorgängerbasen für beide Normen gleich.

**Beweis**. Die Wohlordnung der Basen $(a,b)$ und $(b,c)$ besagt

$$\| a \| \ \leq\ \| a - b \| \ <\ \| b \| \ \leq\ \| b - c \| \ <\ \| c \| \,. \tag{2.1}$$

Wir betrachten die Geraden

$$\begin{aligned}
F(\xi) &= (1-\xi)\ (b-a) + \xi\ \ b\,,\\
G(\xi) &= (1-\xi)\ \ a\ \ \ + \xi\ \ b\,,\\
H^+(\xi) &= (1-\xi)\ \ a\ \ \ + \xi\ (a+b)\,,\\
H^-(\xi) &= (1-\xi)\ -a\ \ \ + \xi\ (b-a)\,.
\end{aligned}$$

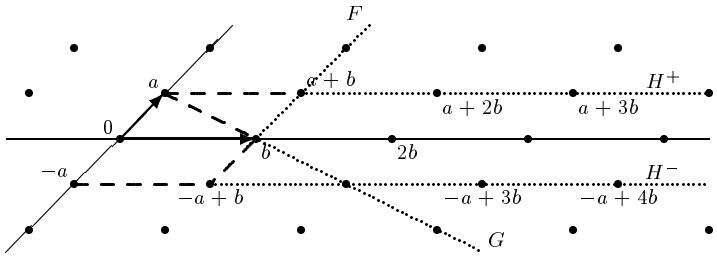


Abbildung 2.1: Wohlgeordnete Basis $(a, b)$

Wegen Ungleichung 2.1 gilt:

$$\begin{aligned}
\| F(0) \| &= \| b - a \| < \| b \| = \| F(1) \| , \\
\| G(0) \| &= \| a \| < \| b \| = \| G(1) \| , \\
\| H^-(0) \| &= \| a \| \leq \| b - a \| = \| H^-(1) \| .
\end{aligned}$$

Nach Lemma 15 ist nun $\| F(\xi) \|$ und $\| G(\xi) \|$ eine streng monoton wachsende Funktion und $\| H^-(\xi) \|$ eine monoton wachsende (d.h. nicht–fallende) Funktion für $\xi \geq 1$. Hieraus folgt die entsprechende Ungleichung für $H^+$:

$$\begin{aligned}
\| H^+(0) \| &= \| G(0) \| \\
&< \| G(1) \| = \| F(1) \| \\
&< \| F(2) \| = \| H^+(1) \|
\end{aligned}$$

Wir entscheiden für alle möglichen Fälle von $\mu$ und $\varepsilon = \pm 1$, ob $(b, c)$ wohlgeordnet ist oder nicht:

$\underline{\varepsilon = 1, \mu \leq -1}$
Dann ist $\| b - c \| = \| (1 - \mu)b - a \| = \| H^-(1 - \mu) \| \geq \| H^-(2) \| = \| 2b - a \| = \| G(2) \| > \| G(1) \| = \| b \|$ und folglich ist $(b, c)$ nicht wohlgeordnet.

$\underline{\varepsilon = -1,\ \mu \leq 0}$
Dann ist $\| b - c \| = \| (1 - \mu)b + a \| = \| H^+(1 - \mu) \| \geq \| H^+(1) \| = \| F(2) \| > \| F(1) \| = \| b \|$ und folglich ist $(b, c)$ nicht wohlgeordnet.

$\underline{\mu = 0}$
Dann ist $\| c \| = \| a \| < \| b \|$ und mithin $(b, c)$ nicht wohlgeordnet.

$\underline{\mu = 1}$
Dann ist $\| b - c \| = \| a \| < \| b \|$ und deshalb $(b, c)$ nicht wohlgeordnet.

$\underline{\varepsilon = -1,\ \mu = 2}$
Dann ist $\| b - c \| = \| a - b \| < \| b \|$, also ist $(b, c)$ nicht wohlgeordnet.

$\underline{\varepsilon = 1,\ \mu \geq 2}$
Dann ist $\| c \| = \| a + \mu b \| = \| H^+(\mu) \| > \| H^+(\mu - 1) \| = \| a + (\mu - 1)b \| = \| b - c \| \geq \| H^+(1) \| = \| F(2) \| > \| F(1) \| = \| b \|$ und folglich ist $(b, c)$ wohlgeordnet.

$\underline{\varepsilon = -1,\ \mu \geq 3}$
Dann ist $\| c \| = \| -a + \mu b \| = \| H^-(\mu) \| \geq \| H^-(\mu - 1) \| = \| b - c \| \geq \| H^-(2) \| = \| G(2) \| > \| G(1) \| = \| b \|$ und folglich ist $(b, c)$ wohlgeordnet. □

## 2.3 Vorgänger einer reduzierten Basis

**Lemma 23.** *Sei $(a, b)$ reduziert und $c = \varepsilon a + \mu b$ wobei $\varepsilon = \pm 1$ und $\mu \in \mathbb{Z}$. Dann gilt:*

1. *Ist $\| a \| \leq \| b \|$ und $(\varepsilon, \mu) \neq (-1, 2)$, so ist $(b, c)$ wohlgeordnet g.d.w. $\mu \geq 2$.*
2. *Ist $\| a \| \geq \| b \|$, so ist $(b, c)$ nicht wohlgeordnet für $\mu \leq 0$ und wohlgeordnet für $\varepsilon = 1,\ \mu > 2\lambda_2/\lambda_1 - 1$ und für $\varepsilon = -1,\ \mu > 2\lambda_2/\lambda_1$ wobei $\lambda_i = \lambda_i(\mathbb{Z}a + \mathbb{Z}b)$.*

**Beweis**. Die Basis $(b, c)$ ist wohlgeordnet g.d.w.

$$\| b \| \leq \| \varepsilon a + (\mu - 1)b \| < \| \varepsilon a + \mu b \| . \tag{2.2}$$

Die *linke Ungleichung* ist immer erfüllt, falls $\| b \| \leq \| a \|$ ist. Aus Satz 18 folgt, daß die linke Ungleichung äquivalent zu $\mu \neq 1$ ist, falls $\| a \| \leq \| b \|$ ist.

Für die *rechte Ungleichung* betrachten wir die Gerade $H(\xi) = \varepsilon a + \xi b$. In beiden Fällen des Lemmas gilt die Ungleichung $\| H(-1) \| = \| \varepsilon a - b \| \geq \| a \| = \| H(0) \| \leq \| \varepsilon a + b \| = \| H(1) \|$. Für $\mu \leq 0$ folgt aus Lemma 15

$$\| \varepsilon a + (\mu - 1)b \| = H(\mu - 1) \geq H(\mu) = \| \varepsilon a + \mu b \| .$$

Also ist $(b, c)$ nicht wohlgeordnet. Das beweist alle Behauptungen für $\mu \leq 0$.

Sei also $\mu \geq 1$. Es gilt

$$\| \varepsilon a + (\mu - 1)b \| \;=\; H(\mu - 1) \;\leq\; H(\mu) \;=\; \| \varepsilon a + \mu b \| \quad . \tag{2.3}$$

Falls $\varepsilon = -1$ gilt $\| a \| \;\leq\; \lambda_2 \;\leq\; \| c \|$ , also

$$\| b \| \;=\; \frac{1}{\mu} \| a + c \| \;\leq\; \frac{2}{\mu} \| c \| \; . \tag{2.4}$$

Falls $\varepsilon = 1$ folgt aus $\| b - a \| \;\leq\; \| a + b \| \;=\; H(1) \;\leq\; \| c \|$ :

$$\| b \| \;=\; \frac{1}{\mu + 1} \| (b - a) + c \| \;\leq\; \frac{2}{\mu + 1} \| c \| \; . \tag{2.5}$$

Angenommen, die linke Ungleichung gilt, aber $(b, c)$ ist nicht wohlgeordnet. Dann gilt in Ungleichung 2.3 Gleichheit. Also ist $(b, c)$ reduziert und folglich $\| b \| \;=\; \lambda_1, \quad \| c \| \;=\; \lambda_2$. Weiterhin ist $(a, b)$ reduziert mit $\| a \| \;\leq\; \| b \|$ , also gilt die rechte Ungleichung 2.2 nicht. Es gilt $\| b \| \;=\; \lambda_2 \;=\; \| c \|$ , also $\mu \leq 1$ für $\varepsilon = 1$ und $\mu \leq 2$ für $\varepsilon = -1$. Falls $\| b \| \;\leq\; \| a \|$ , folgt aus Ungleichung 2.4, daß $\mu \leq 2\frac{\lambda_2}{\lambda_1}$ und aus Ungleichung 2.5, daß $\mu \leq 2\frac{\lambda_2}{\lambda_1} - 1$ .

Zusammfassend haben wir also im Fall $\mu \geq 1$ für jene $\varepsilon, \mu$, für die Wohlordnung von $(b, c)$ behauptet ist, die Annahme, daß $(b, c)$ nicht wohlgeordnet ist, zum Widerspruch geführt. Weiter haben wir gezeigt, daß für $\| a \| \leq \| b \|$ und $\mu = 1$ die Basis $(b, c)$ nicht wohlgeordnet sein kann. □

**Bemerkung.** In den unbestimmten Fällen von Lemma 23, d.h.,

- $\| a \| \;\leq\; \| b \|$ , $\varepsilon = -1,\ \mu = 2$
- $\| b \| \;\leq\; \| a \|$ , $\varepsilon = -1,\ 1 \leq \mu \leq 2\lambda_2/\lambda_1$
- $\| b \| \;\leq\; \| a \|$ , $\varepsilon = \quad 1,\ 1 \leq \mu \leq 2\lambda_2/\lambda_1 - 1$ ,

zeigt der Beweis, daß $(b, c)$ *entweder* reduziert *oder* wohlgeordnet ist. Beide Fälle treten für gewisse Normen bzw. Gitter auf. Für die euklidische Norm (und für alle „glatten“ Normen) gibt es allerdings nur einen unbestimmten Fall, nämlich $\| b \| \;\leq\; \| a \|$ und $\varepsilon = -1,\ \mu = 1$ .

## 2.4 Norm aufeinanderfolgender Basisvektoren

Jede Nachfolgerbasis im Gauß–Algorithmus ist entweder wohlgeordnet, oder der Algorithmus bricht ab. Aus der Charakterisierung der Vorgänger einer wohlgeordneten Basis in Lemma 22 folgt, daß die Reduktionskoeffizienten $\varepsilon = \pm 1,\ \mu \in \mathbb{Z}$ in der Transformation $\text{succ}\,(a,b) = \varepsilon(b - \mu a)$ der Bedingung $\varepsilon + \mu \geq 3$ genügen. In diesem Abschnitt folgern wir daraus scharfe Schranken für die Norm der Vektoren in aufeinanderfolgenden Iterationen des Gauß–Algorithmus. Die Ergebnisse dieses Abschnittes werden für die scharfe Abschätzung der Anzahl der Iterationen im nächsten Abschnitt nicht benötigt. Sie werden aber für Abschätzungen zur Schrittzahl und Stabilität des Gauß–Algorithmus in den Kapiteln 3 und 4 verwendet.

**Lemma 24.** *Es seien $(a,b),\ (b,c)$ wohlgeordnete Basen mit $a = \text{succ}\,(b,c) = \varepsilon(c - \mu b)$ . Dann gilt*

1. $(\mu + 1) \parallel b \parallel \ \geq \ \parallel c \parallel \ \geq \ \mu \parallel b \parallel$ *für* $\varepsilon = 1$
2. $\mu \parallel b \parallel \ \geq \ \parallel c \parallel \ \geq \ (\mu - 1) \parallel b \parallel$ *für* $\varepsilon = -1$ .

**Beweis**. Mit Hilfe der Dreiecksungleichung und der Wohlordnungsbedingung sieht man leicht zunächst die linken Ungleichungen:

$$\begin{aligned} \parallel a + \mu b \parallel &\leq \parallel a \parallel \ + \ \parallel \mu b \parallel \\ &\leq (\mu + 1) \parallel b \parallel \\ \parallel -a + \mu b \parallel &\leq \parallel -a + b \parallel \ + \ \parallel (\mu - 1) b \parallel \\ &\leq \parallel b \parallel \ + (\mu - 1) \parallel b \parallel \\ &= \mu \parallel b \parallel . \end{aligned}$$

Mit den rechten Ungleichungen verfahren wir ebenso:

$$\begin{aligned} \parallel a + \mu b \parallel &= \parallel a - b + (\mu + 1) b \parallel \\ &\geq (\mu + 1) \parallel b \parallel \ - \ \parallel a - b \parallel \\ &\geq \mu \parallel b \parallel \\ \parallel -a + \mu b \parallel &\geq \parallel \mu b \parallel \ - \ \parallel a \parallel \\ &\geq (\mu - 1) \parallel b \parallel . \end{aligned}$$

□

Wir folgern aus Lemma 24, daß in den inneren Basen des Gauß–Algorithmus die Norm des kürzeren Vektors höchsten halb so groß ist wie die Norm des längeren. Somit folgt unmittelbar aus der Charakterisierung der Vorgänger in Lemma 22, daß der Gauß–Algorithmus höchstens $\log_2 B + 1$ Iterationen ausführt, wobei $B$ der

Quotient aus der Norm des längsten Eingabevektors und dem zweiten sukzessiven Minimum des Gitters ist.

**Satz 25.** *Hat die wohlgeordnete Basis $(b, c)$ eine wohlgeordnete Nachfolgerbasis, so gilt* $\| c \| \geq 2 \| b \|$ .

**Beweis**. Sei $(a, b)$ die wohlgeordnete Nachfolgerbasis von $(b, c)$ wobei $a = \text{succ}\,(b, c) = \varepsilon(c - \mu b)$ . Aus Lemma 22 folgen die Ungleichungen $\mu \geq 2$ für $\varepsilon = 1$, und $\mu \geq 3$ für $\varepsilon = -1$. Also folgt aus den rechten Ungleichungen von Lemma 24 die Ungleichung $\| c \| \geq 2 \| b \|$ . □

Das folgende Lemma gibt Schranken für die Norm der Vektoren in den drei Basen nach zwei aufeinanderfolgenden Iterationen des Gauß–Algorithmus.

**Lemma 26.** *Sei $(c, d)$ wohlgeordnet, $b = \text{succ}\,(c, d)$ und $a = \text{succ}\,(b, c) = \varepsilon(c - \mu b)$ mit Reduktionskoeffizienten $\varepsilon = \pm 1$, $\mu \in \mathbb{Z}$.*

- *Sei $\gamma = \| c \| / \| a \|$ . Dann ist*
  1. $\| d \| \geq \frac{(2\mu+1)\gamma - 1}{\mu\gamma} \| c \| = (2 + \frac{1}{\mu}) \| c \| - \frac{1}{\mu} \| a \|$ ,
  2. $\| c \| \leq \frac{\mu\gamma}{\gamma - 1} \| b \|$ .
- *Ist $(a, b)$ wohlgeordnet, so gilt*
  3. $\| d \| \geq (2\mu + 1) \| b \|$ *für* $\varepsilon = 1$
  4. $\| d \| \geq 2\mu \| b \|$ *für* $\varepsilon = -1$

Aussage 1 verschärft Satz 25. Aus den Aussagen 3 und 4 folgt, daß in allen außer den beiden letzten Iterationen des Gauß–Algorithmus gilt

$$\| b \| \geq 5 \| \text{succ}\,(a, b) \| \quad . \tag{2.6}$$

Daraus folgt für den Gauß–Algorithmus eine elementare obere Schranke von $\log_{\sqrt{5}} B + O(1)$ Iterationen.

**Beweis des Lemmas.** Sei $m = b + 2c$ und $n = -b + 3c$. Da $\| b \| \leq \| c \pm b \|$ ist auch $\min(\| m \|, \| n \|) \leq \| d \|$ . Sei $F$ die Gerade durch $\gamma\varepsilon a$ und $b$:

$$F(\xi) = (1 - \xi)\, \gamma\varepsilon a + \xi\, c$$

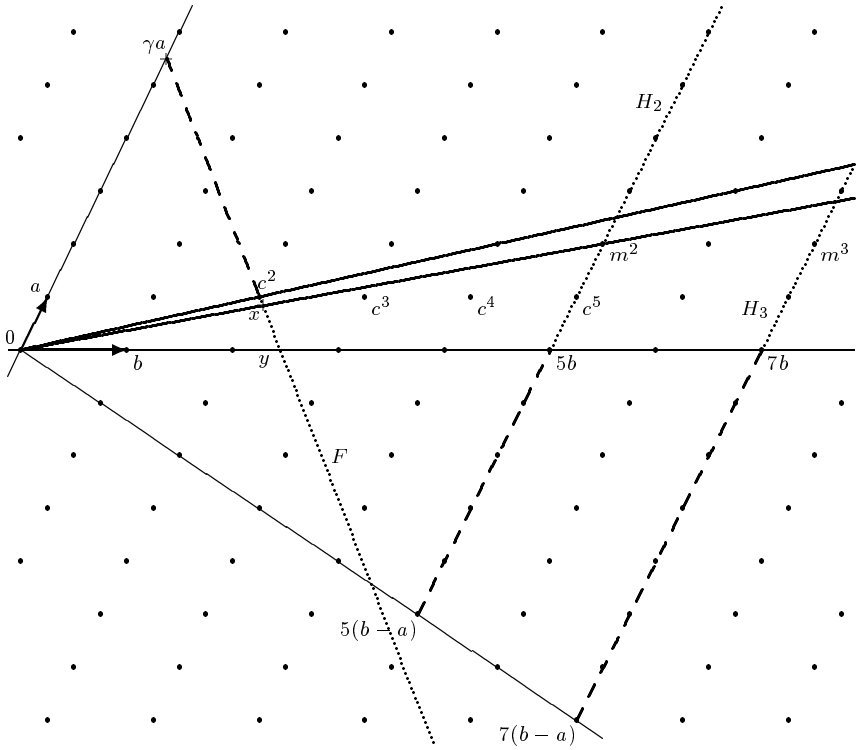


Abbildung 2.2: Beweis von Lemma 26 ($\varepsilon = 1$)

Seien $x, y, z$ die Schnittpunkte von $F$ mit den Geraden von 0 durch $m, b, n$. Es gilt

$$F(\frac{(2\mu+1)\gamma}{(2\mu+1)\gamma-1}) \;=\; x \;=\; \frac{\mu\gamma}{(2\mu+1)\gamma-1}\, m \;, \tag{2.7}$$

$$F(\frac{\gamma}{\gamma-1}) \;=\; y \;=\; \frac{\mu\gamma}{\gamma-1}\, b \;, \tag{2.8}$$

$$F(\frac{(3\mu-1)\gamma}{(3\mu-1)\gamma+1}) \;=\; z \;=\; \frac{\mu\gamma}{(3\mu-1)\gamma+1}\, n \;. \tag{2.9}$$

Nach Voraussetzung ist $\| F(0) \| \;=\; \| \gamma a \| \;\leq\; \| c \| \;=\; \| F(1) \|$ . Also ist wegen Lemma 15

$$\| c \| \;=\; \| F(1) \| \;\leq\; \| F(\tfrac{(2\mu+1)\gamma}{(2\mu+1)\gamma-1}) \| \;\leq\; \| F(\tfrac{\gamma}{\gamma-1}) \| \;\leq\; \| F(\tfrac{(3\mu-1)\gamma}{(3\mu-1)\gamma+1}) \| \;.$$

Daraus folgt direkt Aussage 2. Für Aussage 1 bleibt zu zeigen: $(3\mu-1)\gamma+1 \geq (2\mu+1)\gamma-1$, was äquivalent zu $\mu \geq 2 \vee \gamma \leq 2$ ist. Für $\mu = 1$ folgt aus Lemma 23, daß $\| a \| \;\geq\; \| b \|$ , also $\gamma \leq \| c \| / \| a \| = \| \varepsilon a + b \| / \| a \| \leq 2$.

Zum Beweis der Aussagen 3 und 4 betrachten wir die Geraden

$$H_\mu(\xi) \;=\; (1-\xi)\,(2\mu+\varepsilon)(b-a) \;+\; \xi\,(2\mu+1)b$$

für $\mu = 2, 3, \ldots$. Da $(a, b)$ wohlgeordnet ist, gilt einerseits $\| n \| \;\geq\; \| m \|$ und andererseits $\| b-a \| \;\leq\; \| b \|$ und folglich $\| (2\mu+1)(b-a) \| \;\leq\; \| (2\mu+1)b \|$. Anwendung von Lemma 15 ergibt $\| m \| \;=\; \| 2b_0 + (2\mu+1)b \| \;\geq\; \| (2\mu+1)b \|$. Zum Beweis von Aussage 4 schließen wir genauso mit

$$H_\mu(\xi) \;=\; (1-\xi)\,(2\mu-1)a \;+\; \xi\,(2\mu-1)b\;. \qquad \square$$

## 2.5 Abschätzung der Anzahl der Iterationen

Wir numerieren die Vektoren in einem Lauf des Algorithmus. Sei $(b_1, b_0) \;:=\; (a, b)$ die Eingabebasis. Wir nehmen an, die Eingabe sei wohlgeordnet — andernfalls ist dies nach der ersten Iteration der Fall. Der in der $i$-ten Iteration berechnete Vektor $a$ sei mit

$$b_{i+1} \;:=\; \mathrm{succ}\,(b_i, b_{i-1}) \;=:\; \varepsilon_i(b_{i-1} - \mu_i b_i)$$

bezeichnet, wobei $\varepsilon_i, \mu_i$ die dabei auftretenden Reduktionskoeffizienten sind. Die $i$-te Iteration transformiert also die Basis $(b_i, b_{i-1})$ in die Basis $(b_{i+1}, b_i)$ mittels

$$(b_{i+1}, b_i) \;=\; (b_i, b_{i-1}) \begin{pmatrix} -\varepsilon_i\mu_i & 1 \\ \varepsilon_i & 0 \end{pmatrix}\;. \tag{2.10}$$

Sei $k+1$ die Anzahl der Iterationen. Die ersten $k$ Iterationen erzeugen die Basen $(b_2, b_1), \ldots, (b_{k+1}, b_k)$, die wegen Proposition 20 wohlgeordnet sind. Die reduzierte Ausgabebasis wird von der $k+1$-ten Iteration erzeugt und ist $(b_{k+2}, b_{k+1})$ oder $(b_{k+1}, b_{k+2})$. Inversion der Matrix in Gleichung 2.10 ergibt

$$(b_i, b_{i-1}) \;=\; (b_{i+1}, b_i) \begin{pmatrix} 0 & \varepsilon_i \\ 1 & \mu_i \end{pmatrix}\;,$$

woraus wir folgende Darstellung der Eingabe als Linearkombination der letzten wohlgeordneten Basis in dem Lauf des Algorithmus erhalten:

$$(b_1, b_0) = (b_{k+1}, b_k) \begin{pmatrix} 0 & \varepsilon_k \\ 1 & \mu_k \end{pmatrix} \begin{pmatrix} 0 & \varepsilon_{k-1} \\ 1 & \mu_{k-1} \end{pmatrix} \cdots \begin{pmatrix} 0 & \varepsilon_1 \\ 1 & \mu_1 \end{pmatrix} . \tag{2.11}$$

Unser Ziel ist es nun, eine scharfe untere Schranke für die Größe von $b_0$ zu finden. Dazu werten wir das Matrixprodukt in Gleichung 2.11 explizit aus. Zunächst führen wir ein hilfreiches Werkzeug hierfür ein. Die *verallgemeinerten Kontinuanten* sind Polynome

$$\left[ \begin{matrix} x_1 \dots x_{n-1} \\ y_1 \ \dots \ y_n \end{matrix} \right]_n \in \mathbb{Z}[x_1, \dots, x_{n-1}, y_1, \dots, y_n] ,$$

die rekursiv definiert sind durch

$$\left[ \ \right]_{-1} = 0, \quad \left[ \ \right]_0 = 1, \quad \left[ y_1 \right]_1 = y_1$$

$$\left[ \begin{matrix} x_1 \dots x_{n-1} \\ y_1 \ \dots \ y_n \end{matrix} \right]_n = y_n \left[ \begin{matrix} x_1 \dots x_{n-2} \\ y_1 \ \dots \ y_{n-1} \end{matrix} \right]_{n-1} + x_{n-1} \left[ \begin{matrix} x_1 \dots x_{n-3} \\ y_1 \ \dots \ y_{n-2} \end{matrix} \right]_{n-2} . \tag{2.12}$$

Es gilt

$$\begin{aligned} \left[ \begin{matrix} x_1 \dots x_{n-1} \\ y_1 \ \dots \ y_n \end{matrix} \right]_n &= \left[ \begin{matrix} x_{n-1} \dots x_1 \\ y_n \ \dots \ y_1 \end{matrix} \right]_n \\ &= y_1 \left[ \begin{matrix} x_2 \dots x_{n-1} \\ y_2 \ \dots \ y_n \end{matrix} \right]_{n-1} + x_1 \left[ \begin{matrix} x_3 \dots x_{n-1} \\ y_3 \ \dots \ y_n \end{matrix} \right]_{n-2} . \end{aligned} \tag{2.13}$$

**Bemerkung.** Die verallgemeinerten Kontinuanten wurden von Rieger zur Analyse des zentrierten Euklidischen Algorithmus eingeführt [R78]. Sie verallgemeinern die gewöhnlichen Kontinuanten $Q_n(y_1, \dots, y_n) = \left[ \begin{smallmatrix} 1 \dots 1 \\ y_1 \ \dots \ y_n \end{smallmatrix} \right]_n$, die von der Analyse des gewöhnlichen (monotonen) Euklidischen Algorithmus wohlbekannt sind. Für $x_i = \varepsilon_i \in \{\pm 1\}$ lassen sich die verallgemeinerten Kontinuanten auch durch die gewöhnlichen ausdrücken:

$$\left[ \begin{matrix} \varepsilon_1 \dots \varepsilon_{n-1} \\ \mu_1 \ \dots \ \mu_n \end{matrix} \right]_n = \varepsilon_{n-1}\varepsilon_{n-3} \cdots Q(\mu_1, \, \varepsilon_1\mu_2, \, \varepsilon_1\varepsilon_2\mu_3, \, \dots, \, \varepsilon_1 \cdots \varepsilon_{n-1}\mu_n) .$$

Obwohl die Forderung $\varepsilon_i \in \{\pm 1\}$ beim zentrierten Reduktionsalgorithmus erfüllt ist, scheint mir der Gebrauch der verallgemeinerten Kontinuanten übersichtlicher. Wir erhalten folgende Resultate:

**Lemma 27.** *Für* $k \geq 1$ *gilt*

$$\begin{pmatrix} 0 & \varepsilon_k \\ 1 & \mu_k \end{pmatrix} \cdots \begin{pmatrix} 0 & \varepsilon_1 \\ 1 & \mu_1 \end{pmatrix} = \begin{pmatrix} \varepsilon_k \left[\begin{smallmatrix} \varepsilon_2 \ldots \varepsilon_{k-2} \\ \mu_2 \ \ldots \ \mu_{k-1} \end{smallmatrix}\right]_{k-2} & \varepsilon_k \left[\begin{smallmatrix} \varepsilon_1 \ldots \varepsilon_{k-2} \\ \mu_1 \ \ldots \ \mu_{k-1} \end{smallmatrix}\right]_{k-1} \\ \left[\begin{smallmatrix} \varepsilon_2 \ldots \varepsilon_{k-1} \\ \mu_2 \ \ldots \ \mu_k \end{smallmatrix}\right]_{k-1} & \left[\begin{smallmatrix} \varepsilon_1 \ldots \varepsilon_{k-1} \\ \mu_1 \ \ldots \ \mu_k \end{smallmatrix}\right]_k \end{pmatrix}$$

**Beweis**. Wir beweisen die Gleichung mittels Induktion über $k$. Die Gleichung gilt für $k = 1$. Einsetzen der rekursiven Definition (2.12) in die Gleichung für $k$ ergibt direkt

$$\begin{aligned} &\begin{pmatrix} 0 & \varepsilon_{k+1} \\ 1 & \mu_{k+1} \end{pmatrix} \begin{pmatrix} 0 & \varepsilon_k \\ 1 & \mu_k \end{pmatrix} \cdots \begin{pmatrix} 0 & \varepsilon_1 \\ 1 & \mu_1 \end{pmatrix} \\ &= \begin{pmatrix} 0 & \varepsilon_{k+1} \\ 1 & \mu_{k+1} \end{pmatrix} \begin{pmatrix} \varepsilon_k \left[\begin{smallmatrix} \varepsilon_2 \ldots \varepsilon_{k-2} \\ \mu_2 \ \ldots \ \mu_{k-1} \end{smallmatrix}\right]_{k-2} & \varepsilon_k \left[\begin{smallmatrix} \varepsilon_1 \ldots \varepsilon_{k-2} \\ \mu_1 \ \ldots \ \mu_{k-1} \end{smallmatrix}\right]_{k-1} \\ \left[\begin{smallmatrix} \varepsilon_2 \ldots \varepsilon_{k-1} \\ \mu_2 \ \ldots \ \mu_k \end{smallmatrix}\right]_{k-1} & \left[\begin{smallmatrix} \varepsilon_1 \ldots \varepsilon_{k-1} \\ \mu_1 \ \ldots \ \mu_k \end{smallmatrix}\right]_k \end{pmatrix} \\ &= \begin{pmatrix} \varepsilon_{k+1} \left[\begin{smallmatrix} \varepsilon_2 \ldots \varepsilon_{k-1} \\ \mu_2 \ \ldots \ \mu_k \end{smallmatrix}\right]_{k-1} & \varepsilon_{k+1} \left[\begin{smallmatrix} \varepsilon_1 \ldots \varepsilon_{k-1} \\ \mu_1 \ \ldots \ \mu_k \end{smallmatrix}\right]_k \\ \left[\begin{smallmatrix} \varepsilon_2 \ldots \varepsilon_k \\ \mu_2 \ \ldots \ \mu_{k+1} \end{smallmatrix}\right]_k & \left[\begin{smallmatrix} \varepsilon_1 \ldots \varepsilon_k \\ \mu_1 \ \ldots \ \mu_{k+1} \end{smallmatrix}\right]_{k+1} \end{pmatrix}, \end{aligned}$$

was die Induktionsbehauptung für $k + 1$ ist. □

Wir studieren die Produktmatrix aus Lemma 27 für den Fall $\mu_1 = \mu_2 = \ldots = \mu_k = 2$. Wegen Lemma 22 sind dann alle $\varepsilon_i$ positiv. Sei $T_i = \left[\begin{smallmatrix} 1 \ldots 1 \\ 2 \ \ldots \ 2 \end{smallmatrix}\right]_i$. Wir beweisen zunächst folgende einfache Formel für $T_i$:

**Lemma 28.** *Für* $i \geq 0$ *gilt*

$$T_i \ = \ \frac{(1+\sqrt{2})^{i+1} - (1-\sqrt{2})^{i+1}}{2\sqrt{2}} \ .$$

**Beweis**. Nach Rekursionsvorschift gilt $T_i = 2T_{i-1} + T_{i-2}$ , als Rekurrenz gelesen:

$$(T_{i-1}, T_i) \ = \ (T_{i-2}, T_{i-1}) \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix} \ = \ (T_{-1}, T_0) \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}^i \ = \ (0, 1) \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}^i \ .$$

Wir transformieren die Matrix $\left(\begin{smallmatrix} 0 & 1 \\ 1 & 2 \end{smallmatrix}\right)$ auf Diagonalgestalt. Seien $\eta_{1/2} = 1 \pm \sqrt{2}$ ihre Eigenwerte. Dann gilt

$$\begin{aligned}
\begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}^i &= \frac{1}{\eta_2 - \eta_1} \begin{pmatrix} \eta_2 & -\eta_1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \eta_1^i & 0 \\ 0 & \eta_2^i \end{pmatrix} \begin{pmatrix} 1 & \eta_1 \\ 1 & \eta_2 \end{pmatrix} \\
&= \frac{1}{\eta_2 - \eta_1} \begin{pmatrix} \eta_2 & -\eta_1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \dots & \eta_1^{i+1} \\ \dots & \eta_2^{i+1} \end{pmatrix} \\
&= \frac{1}{\eta_2 - \eta_1} \begin{pmatrix} \dots & \dots \\ \dots & -\eta_1^{i+1} + \eta_2^{i+1} \end{pmatrix},
\end{aligned}$$

also

$$T_i = \frac{1}{\eta_2 - \eta_1} (-\eta_1^{i+1} + \eta_2^{i+1}) = \frac{(1+\sqrt{2})^{i+1} - (1-\sqrt{2})^{i+1}}{2\sqrt{2}} . \qquad \square$$

Lemma 27 bestimmt mit Gleichung 2.11 die Koeffizienten der Darstellung des Eingabevektors $b_0$ bezüglich der Ausgabebasis

$$b_0 = \varepsilon_k \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-2} \\ \mu_1 \ \dots \ \mu_{k-1} \end{bmatrix}_{k-1} b_{k+1} + \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-1} \\ \mu_1 \ \dots \ \mu_k \end{bmatrix}_k b_k . \tag{2.14}$$

Für die Kontinuanten in dieser Gleichung gilt:

**Lemma 29.** *Sei* $i \geq 0$, $\mu_j \geq 2$ *für* $1 \leq j \leq i$ *und* $\mu_j \geq 3$ *für* $\varepsilon_j = -1$. *Dann ist*

$$\begin{bmatrix} \varepsilon_1 \dots \varepsilon_{i-1} \\ \mu_1 \ \dots \ \mu_i \end{bmatrix}_i \geq T_i \tag{2.15}$$

$$\begin{bmatrix} \varepsilon_1 \dots \varepsilon_{i-1} \\ \mu_1 \ \dots \ \mu_i \end{bmatrix}_i \geq 2 \begin{bmatrix} \varepsilon_2 \dots \varepsilon_{i-1} \\ \mu_2 \ \dots \ \mu_i \end{bmatrix}_{i-1} . \tag{2.16}$$

**Beweis**. Wir beweisen die Ungleichungen 2.15 und 2.16 durch Induktion über $i$. Man sieht die Ungleichungen direkt für $i = 0$ und $i = 1$. Nehmen wir an, sie gelten für alle $i \leq l$. Aus der rekursiven Definition 2.13 folgt

$$\begin{bmatrix} \varepsilon_1 \dots \varepsilon_l \\ \mu_1 \ \dots \ \mu_{l+1} \end{bmatrix}_{l+1} = \mu_1 \begin{bmatrix} \varepsilon_2 \dots \varepsilon_l \\ \mu_2 \ \dots \ \mu_{l+1} \end{bmatrix}_l + \varepsilon_1 \begin{bmatrix} \varepsilon_3 \dots \varepsilon_l \\ \mu_3 \ \dots \ \mu_{l+1} \end{bmatrix}_{l-1} .$$

Wegen Induktionsvoraussetzung 2.16 für $i = l$ und 2.15 für $i = l - 1$ gilt

$$\begin{bmatrix} \varepsilon_2 \dots \varepsilon_l \\ \mu_2 \ \dots \ \mu_{l+1} \end{bmatrix}_l \ \geq \ 2 \ \begin{bmatrix} \varepsilon_3 \dots \varepsilon_l \\ \mu_3 \ \dots \ \mu_{l+1} \end{bmatrix}_{l-1} \ \geq 0 \ .$$

Somit folgt in den beiden Fällen $\varepsilon_1 = 1, \mu_1 \geq 2$ und $\varepsilon_1 = -1, \mu_1 \geq 3$ die Ungleichung

$$\begin{bmatrix} \varepsilon_1 \dots \varepsilon_l \\ \mu_1 \ \dots \ \mu_{l+1} \end{bmatrix}_{l+1} \ \geq \ 2 \ \begin{bmatrix} \varepsilon_2 \dots \varepsilon_l \\ \mu_2 \ \dots \ \mu_{l+1} \end{bmatrix}_l + \ \begin{bmatrix} \varepsilon_3 \dots \varepsilon_l \\ \mu_3 \ \dots \ \mu_{l+1} \end{bmatrix}_{l-1} \ ,$$

womit Induktionsbehauptung 2.16 für $i = l + 1$ gezeigt ist. Mit Induktionsvoraussetzung 2.15 für $i = l, \ l - 1$ folgt jetzt:

$$\begin{bmatrix} \varepsilon_1 \dots \varepsilon_l \\ \mu_1 \ \dots \ \mu_{l+1} \end{bmatrix}_{l+1} \ \geq \ 2\,T_l \ + T_{l-1} \ = \ T_{l+1} \ . \qquad \square$$

**Lemma 30.** *Sei* $(b_1, b_0), \dots, (b_{k+1}, b_k)$ *eine Folge von wohlgeordneten Basen mit* $b_{i+1} = \mathrm{succ}\,(b_i, b_{i-1}) = \varepsilon_i(b_{i-1} - \mu_i b_i)$. *Dann ist* $\| b_0 \| \ \geq \ T_k \, \| b_k \|$ .

**Beweis**. Wegen Lemma 29 ist der Koeffizient von $b_k$ in Gleichung 2.14 positiv und der Koeffizient von $b_{k+1}$ hat Vorzeichen $\varepsilon_k$. Wir unterscheiden zwei Fälle:

**Fall 1.** $\varepsilon_k = 1$. Die Behauptung folgt aus Gleichung 2.14 unter Verwendung der Wohlordnung vom $(b_{k+1}, b_k)$ und Ungleichung 2.15:

$$\begin{aligned} \| b_0 \| \ &\geq \ \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-1} \\ \mu_1 \ \dots \ \mu_k \end{bmatrix}_k \ \| b_k \| \\ &\geq \ T_k \, \| b_k \| \end{aligned}$$

**Fall 2.** $\varepsilon_k = -1$. Da $(b_{k+1}, b_k)$ wohlgeordnet ist, gilt $\| b_{k+1} \| \ < \ \| b_k \|$ . Aus Gleichung 2.14 folgt mit der Dreiecksungleichung

$$\| b_0 \| \ \geq \ \left( \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-1} \\ \mu_1 \ \dots \ \mu_k \end{bmatrix}_k \ - \ \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-2} \\ \mu_1 \ \dots \ \mu_{k-1} \end{bmatrix}_{k-1} \right) \ \| b_k \| \ .$$

Die Auswertung der Kontinuantendifferenz ergibt nach Definition:

$$\begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-1} \\ \mu_1 \ \dots \ \mu_k \end{bmatrix}_k - \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-2} \\ \mu_1 \ \dots \ \mu_{k-1} \end{bmatrix}_{k-1} =$$

$$\mu_k \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-2} \\ \mu_1 \ \dots \ \mu_{k-1} \end{bmatrix}_{k-1} + \varepsilon_{k-1} \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-3} \\ \mu_1 \ \dots \ \mu_{k-2} \end{bmatrix}_{k-2} - \begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-2} \\ \mu_1 \ \dots \ \mu_{k-1} \end{bmatrix}_{k-1} =$$

$$\begin{bmatrix} \varepsilon_1 \dots \varepsilon_{k-1} \\ \mu_1 \ \dots \ (\mu_k - 1) \end{bmatrix}_k .$$

Da $\varepsilon_k = -1$ gilt $\mu_k \geq 3$. Also ist der letzte Kontinuant wegen Ungleichung 2.15 mindestens $T_i$. □

**Satz 31.** *Bei Eingabe einer wohlgeordneten Gitterbasis $(a, b)$ bricht der verallgemeinerte Gauß–Algorithmus nach höchstens*

$$\log_{1+\sqrt{2}}(2\sqrt{2}\,B) \ + \ o(1)$$

*Iterationen ab für $B \to \infty$, wobei $B = \max(\| a \|, \| b \|)/\lambda_2$ ist.*

**Bemerkung.** Ist die Eingabebasis nicht wohlgeordnet, kann eine zusätzliche Iteration ausgeführt werden. Der Betrag des $o$-Termes ist beschränkt durch $2 - \log_{1+\sqrt{2}}(4\sqrt{2}) \approx 0.0339$ wobei das Maximum bei einer einzigen Iteration angenommen wird.

**Beweis**. Wegen Lemma 30 gilt

$$T_k \ \leq \ \frac{\| b_0 \|}{\| b_k \|} \ \leq \ B \ .$$

Der Satz folgt mittels Einsetzen der Schranke aus Lemma 28 und anschließendem Logarithmieren der Ungleichung. □

**Satz 32.** *Sei $(b_{m+1}, b_m)$ eine reduzierte Gitterbasis und die Folge $b_m, \dots, b_0$ von Vorgängern definiert durch $b_{i-1} = b_{i+1} + 2b_i$ für $i = m, \dots, 1$. Dann durchläuft der*

*verallgemeinerte Gauß–Algorithmus auf Eingabe* $(b_1, b_0)$ *genau* $m$ *Iterationen, wobei die Folge* $b_2, \ldots, b_{m+1}$ *durchlaufen wird. Es gilt*

$$m \;\geq\; \log_{1+\sqrt{2}}(2\,B) \;-\; 1 \;+\; o(1)\;,$$

*wobei* $B = \| b_0 \| / \lambda_2$.

**Bemerkung.** Die Differenz der Schranken in den beiden Sätzen ist

$$\log_{1+\sqrt{2}}(2\sqrt{2}) - \log_{1+\sqrt{2}}(\tfrac{2}{1+\sqrt{2}}) + o(1)$$

$$= \log_{1+\sqrt{2}}(\sqrt{2}) + 1 + o(1) \;\approx\; 1.393\;.$$

**Beweis**. Wegen Lemma 23 ist $(b_m, b_{m-1})$ wohlgeordnet und wegen Lemma 24 ist auch jede weitere Basis $(b_i, b_{i-1})$ für $i = m, \ldots, 1$ wohlgeordnet. Folglich ist stets $b_{i+1} = \mathrm{succ}\,(b_i, b_{i-1})$ und $m$ die Anzahl der Iterationen, wobei alle Reduktionskoeffizienten gleich 2 sind. Wie in Gleichung 2.14 gilt

$$b_0 = T_{m-2} b_m + T_{m-1} b_{m-1}\;.$$

Wir verwenden $\| b_1 \| = \lambda_2(L)$, $\| b_2 \| \leq 3\lambda_2(L)$ und Lemma 28 für die Abschätzung

$$\begin{aligned} \| b_{m+1} \| &\leq (T_{m-2} + 3T_{m-1})\,\lambda_2 \\ &= (1+\sqrt{2})^{m-1}\,\frac{1+3(1+\sqrt{2})}{2\sqrt{2}}\;\lambda_2\;(1+o(1))\;. \end{aligned}$$

Also gilt

$$m \;\geq\; \log_{1+\sqrt{2}}(\tfrac{2}{1+\sqrt{2}}\,B) \;+\; o(1)\;. \qquad \square$$

**Bemerkungen.**

- Die obere Schranke von Satz 31 wird für gewisse Normen und Gitter angenommen. Als Beispiel nehme man bei der Konstruktion aus Satz 32 das Gitter $\mathbb{Z}^2$ bezüglich der $l_\infty$-Norm mit der reduzierten Basis $b_{m+1} = (1,0),\ b_m = (0,1)$. Dann gilt tatsächlich die Gleichheit $\| b_0 \| = T_m$.

- Wählt man die ganzen Zahlen $b_{m+1} = 0,\ b_m = 1$ anstelle der reduzierten Ausgabebasis in Satz 32, so liefert die Rekursion $b_{i-1} = b_{i+1} + 2b_i$, wie bereits Dupré erkannte [Du1846], die minimalen ganzen Zahlen, für die der euklidische Algorithmus genau $m$ Divisionen durchführt. Vallée hat diese Minimalität für das Gitter $\mathbb{Z}^2$ und die euklidische Norm gezeigt [Va91]. Die neue Erkenntnis aus Satz 32 ist die Gültigkeit dieses Sachverhaltes für alle Gitter, alle Normen und alle reduzierten Ausgabebasen.

# Kapitel 3

# Zur Schrittzahl des verallgemeinerten Gauß–Algorithmus

Wir geben in diesem Kapitel obere Schranken für die Schrittzahl des Gauß–Algorithmus im RAM–Modell an. Zur Ausführung des im letzten Kapitel vorgestellten verallgemeinerten Gauß–Algorithmus muß in jeder Iteration der ganzzahlige Reduktionskoeffizient $\mu = \mu(a,b) \in \mathbb{Z}$ berechnet werden, der $\| b - \mu a \|$ für $\mu \in \mathbb{Z}$ minimiert. Da außerdem Normen von Vektoren verglichen werden müssen, setzen wir voraus, daß die RAM

- Zugriff auf ein Norm–Orakel hat und
- neben den arithmetischen Operationen Addition, Subtraktion, Multiplikation und Division auch Größenvergleich und Berechnung der nächsten ganzen Zahl durchführen kann.

Wir zählen die arithmetischen Operationen im uniformen Zeitmodell, d.h. zu einheitlichen Kosten. Als *Schritte* zählen wir arithmetische Schritte und Orakelaufrufe. Wir beweisen mit Satz 33 eine Schrittzahlschranke für beliebige Normen. Im zweiten Abschnitt dieses Kapitels geben wir Algorithmen für die $l_1$- und $l_\infty$-Norm an, die dann keine Orakelaufrufe verwenden und eine schärfere Schranke für diese Normen ergeben.

**Satz 33.** *Für jede Norm gibt es einen Algorithmus, der unter Verwendung eines Orakels für die Norm jede gegebene wohlgeordnete Basis $(a,b)$ in $O(n \log(n + \lambda_2/\lambda_1) + \log B)$ arithmetischen Schritten und $O(\log(n + \lambda_2/\lambda_1))$ Orakelaufrufen reduziert, wobei $B = \| b \|/\lambda_2$ .*

## 3.1 Effiziente Reduktion

Zur effizienten Reduktion einer Basis $a, b \in \mathbb{R}^n$ in einer beliebigen Norm $\| \, . \, \|$ führen wir zunächst eine Vorreduktion von $a, b$ in der durch ein geeignetes Skalarprodukt $<,>$ definierten sphärischen Norm durch. Anschließend reduzieren wir die $<,>$-reduzierte Basis in der ursprünglichen Norm. Die sphärische Reduktion ist sinnvoll, weil hier eine Iteration in nur $O(1)$ arithmetischen Schritten durchgeführt werden kann.

Das Skalarprodukt $<,>$ wird so gewählt, daß die Kugel $\{x \in \mathbb{R}^n | \, \| \, x \, \| \leq 1\}$ sphärisch in dem Sinne ist, daß $\max_{x,y \in \mathbb{R}^n} \frac{\|x\| <y,y>^{1/2}}{\|y\| <x,x>^{1/2}} = O(n^{1.5})$ . Die Existenz eines solchen Skalarprodukts folgt aus [Len83], siehe Konstruktion von $\tau$ in Kapitel 2, pp. 542 ff. Wir nehmen an, daß das Skalarprodukt gegeben ist und zählen nicht die Schritte für seine Konstruktion. Es gilt

$$B_{<,>} = \max\left(< a, a >^{1/2}, \; < b, b >^{1/2}\right) / \lambda_{2,<,>} \; = \; O(n^{1.5} \, B) \; .$$

**Vorreduktion in $O(n+\log B)$ arithmetischen Schritten.** In jeder Iteration wird nur die Gram–Matrix

$$G = \begin{pmatrix} < a, a > & < a, b > \\ < b, a > & < b, b > \end{pmatrix}$$

transformiert und die Transformation der Vektoren in den Matrizen $H \in GL_n(\mathbb{Z})$ mit $(a_{current}, b_{current}) = (a_{input}, b_{input})H$ protokolliert als $G := S^\top G S, H := HS$ wobei

$$S = \begin{pmatrix} -\varepsilon\mu & 1 \\ \varepsilon & 0 \end{pmatrix}$$

ist und $\mu$ die nächste ganze Zahl zu $\frac{<a,b>}{<a,a>}$ ist. Jede Iteration kostet 6 Multiplikationen, 6 Subtraktionen, 1 Division und 1 Berechnung einer nächsten ganzen Zahl. Die Transformationen von $(a, b)$ in $G$ und zurück von $H$ in die Ausgabebasis $(a, b)$ kosten $7n$ Multiplikationen und $5n - 3$ Additionen. Nach Satz 31 führt der Gauß–Algorithmus zur $<,>$-Reduktion von $(a, b)$ höchstens $O\left(\log \, B_{<,>}\right) = O\left(\log\left(B+n\right)\right)$ Iterationen durch.

**Berechnung von $\mu$ in $O(1)$ bzw. $O(\log(\lambda_2/\lambda_1))$ Orakelschritten.** Bezeichne $\mu : \mathbb{R}^n \mapsto \mathbb{Z}$ die in Abschnitt 2.1 eingeführte Funktion, die Norm $\| b - \mu a \|$ minimiert. Ist $a, b \in \mathbb{R}^n$ gegeben, so gilt es $\mu = \| b \pm x \| / \| a \|$ zu berechnen, wobei $x = \mathrm{succ}(a, b)$. Folglich ist $|\mu - \frac{\|b\|}{\|a\|}| \leq \frac{\|x\|}{\|a\|}$. Für den letzten Quotienten gelten die folgenden Schranken:

1. $\frac{\|x\|}{\|a\|} \leq \frac{\lambda_2}{\lambda_1}$,
2. $\frac{\|x\|}{\|a\|} < 1$, in allen außer der letzten Iteration,
3. $\frac{\|x\|}{\|a\|} \leq \frac{1}{2}$, in allen außer den letzten beiden Iterationen,
4. $\frac{\|x\|}{\|a\|} < \frac{1}{2}$, in allen außer den letzten drei Iterationen.

Die Behauptungen sind direkt einsehbar: Ist $(x, a)$ reduziert, so gilt wegen Satz 18 die Identität $\{\| x \|, \| a \|\} = \{\lambda_1, \lambda_2\}$. Andernfalls ist nach Proposition 20 die Basis $(x, a)$ wohlgeordnet, und somit ist $\| x \| < \| a \|$. Weiter gilt nach Satz 25 in allen außer den letzten beiden Iterationen $2 \| x \| \leq \| a \|$ und nach Behauptung 1 aus Lemma 26 in allen außer den letzten beiden Iterationen sogar $2 \| x \| < \| a \|$.

Folglich ist $\mu(a, b)$ in allen Iterationen außer der letzten eine der beiden nächsten ganzen Zahlen zu $\frac{\|b\|}{\|a\|}$ und kann mithin durch Auswertung des Normorakels an den beiden Werten $b - \lceil \frac{\|b\|}{\|a\|} \rceil a$, $b - \lfloor \frac{\|b\|}{\|a\|} \rfloor a$ und Vergleich beider Werte ermittelt werden. (In allen außer den letzten 3 Iterationen ist $\mu$ sogar die nächste ganze Zahl zu $\frac{\|b\|}{\|a\|}$.)

In der letzten Iteration ist das gesuchte $\mu$ eine ganze Zahl in einem Intervall der Länge $2\lambda_2/\lambda_1$ mit Zentrum $\frac{\|b\|}{\|a\|}$. Der Quotient $\lambda_2/\lambda_1$, der a priori i.A. nicht bekannt ist, läßt sich durch Auswertung der Norm an den benachbarten ganzen Zahlen und sukzessiver Verdoppelung der Testintervallgröße in $O(\log \lambda_2/\lambda_1)$ Orakelaufrufen und $O(n \log \lambda_2/\lambda_1)$ arithmetischen Schritten abschätzen. Mittels Bisektion läßt sich $\mu$ dann in einer ebenso beschränkten Anzahl von Schritten berechnen.

**Endgültige $\| \cdot \|$-Reduktion in $O(n \log(n + \lambda_2/\lambda_1))$ Schritten.** Aus Satz 31 folgt, daß der Gauß–Algorithmus zur endgültigen $\| \cdot \|$-Reduktion der vorreduzierten Basis höchstens $\log_{1+\sqrt{2}}(2\sqrt{2} \max_{x,y \in \mathbb{R}^n} \frac{\|x\| < y, y >^{1/2}}{\|y\| < x, x >^{1/2}}) + 1 + o(1) = O(\log n)$ Iterationen durchführt. Jede Iteration außer der letzten kostet $O(1)$ Normberechnungen und $O(n)$ arithmetische Schritte. Die letzte Iteration kostet $O(\log \lambda_2/\lambda_1)$ Normberechnungen und $O(n \log \lambda_2/\lambda_1)$ arithmetische Schritte.

## 3.2 Algorithmen für die $l_1$- und die $l_\infty$-Norm

Wir geben in diesem Abschnitt einfache effiziente Algorithmen zur Berechnung des Reduktionskoeffizienten $\mu(a,b)$ für die $l_\infty$-Norm $\| \ . \ \|_\infty$ und die $l_1$–Norm $\| \ . \ \|_1$ an. Wir beweisen folgende Laufzeitschranken:

**Lemma 34.** *Für die $l_\infty$- und die $l_1$-Norm kann der ganzzahlige Reduktionskoeffizient $\mu = \mu(a,b)$, der $\| \ b - \mu a \ \|$ minimiert, in $O(n)$ arithmetischen Operationen für die $l_1$-Norm und in $O(n \log n)$ arithmetischen Operationen für die $l_\infty$-Norm berechnet werden.*

Zusammen mit den zu Beginn des Kapitels beschriebenen Algorithmen ergibt sich also für die beiden Normen die angekündigte Verschärfung von Satz 33:

**Satz 35.** *Für die $l_1$- und $l_\infty$-Norm gibt es einen Algorithmus, der jede gegebene wohlgeordnete Basis $(a,b)$ in $O(n \log n + \log B)$ arithmetischen Schritten reduziert, wobei $B = \| \ b \ \| / \lambda_2$.*

In den beiden Fällen $\| \ (a_1, \ldots, a_n) \ \|_1 = \sum_{i=1}^n |a_i|$ und $\| \ (a_1, \ldots, a_n) \ \|_\infty = \max_i |a_i|$ ist die Funktion $f(\xi) = \| \ b - \xi a \ \|$ stückweise linear und konvex mit höchstens $n$ *Ecken*. Die Funktion $f$ nimmt ihr Minimum an einer dieser Ecken an und der Reduktionskoeffizient ist folglich eine der beiden nächsten ganzen Zahlen zu der Ecke $\xi_M$, die $f(\xi_M)$ minimiert. Um $\xi_M$ zu bestimmen betrachten wir die Komponenten $f_i(\xi) = |b_i - \xi a_i|$ für $i = 1, \ldots, n$. Es gilt $f(\xi) = \sum_{i=1}^n f_i(\xi)$ für die $l_1-$Norm und $f(\xi) = \max\{ \ f_i(\xi) \mid 1 \le i \le n\}$ für die Maximumnorm. Wir betrachten nur Koordinaten $i$ mit $a_i \neq 0$ da $f_i$ sonst konstant ist.

### 3.2.1 Ein gewichteter Median–Algorithmus für die $l_1-$Norm

Für die $l_1$–Norm sind die Ecken die Nullstellen $\xi_i = b_i/a_i$ der Komponentenfunktionen. Es gilt

$$\begin{aligned} f(\xi) &= \sum_{i=1}^n |b_i - \xi a_i| \\ &= \sum_{i=1}^n |a_i| \, |\xi_i - \xi| \end{aligned}$$

$$= \sum_{\substack{i=1\\ \xi_i>\xi}}^{n} |a_i|\,(\xi_i - \xi) \;-\; \sum_{\substack{i=1\\ \xi_i\le\xi}}^{n} |a_i|\,(\xi_i - \xi)$$
$$= \sum_{\xi_i>\xi} |a_i|\xi_i - \sum_{\xi_i\le\xi} |a_i|\xi_i \;-\; \xi \underbrace{\Big(\sum_{\xi_i>\xi} |a_i| - \sum_{\xi_i\le\xi} |a_i|\Big)}_{A(\xi)}$$

wobei die Steigung $-A(\xi)$ konstant ist zwischen zwei Ecken. $f$ ist ein einseitig differenzierbares konvexes Polygon mit $\lim_{\xi\to\pm\infty} f(\xi) = +\infty$. Also ist eine Nullstelle $\xi_M$ Minimalstelle von $f$ genau dann, wenn die einseitigen Ableitungen $f'_l(\xi_M) \le 0 \le f'_r(\xi_M)$ erfüllen. Da die Ableitung $f'(\xi)$ zwischen zwei Nullstellen gleich $-A(\xi)$ ist, ist dies gleichbedeutend mit

$$\sum_{\xi_i<\xi_M} |a_i| \;-\; \sum_{\xi_i\ge\xi_M} |a_i| \;\le\; 0 \;\le\; \sum_{\xi_i\le\xi_M} |a_i| \;-\; \sum_{\xi_i>\xi_M} |a_i|\,. \tag{3.1}$$

Für $|a_i| = 1$ nennt man ein solches $\xi_M$ mit $|\sum_{\xi_i\le\xi_M} 1 - \sum_{\xi_i\ge\xi_M} 1| \le 1$ *Median* von $\xi_1,\ldots,\xi_n$. Die gesuchte Minimalstelle $\xi_M$ ist also ein gewichteter Median der Nullstellen $\xi_i$ mit Gewichten $|a_i|$. Der folgende Algorithmus WM berechnet dieses $\xi_M$ in $O(n)$ RAM–Schritten:

**Algorithmus WM für den $\|\,.\,\|_1$–Reduktionskoeffizienten**

***Input*** $(\xi_1,a_1),\ldots,(\xi_n,a_n) \in \mathbb{R}\times(\mathbb{R}-\{0\})$ .
*(Wobei $\xi_i = b_i/a_i$ für $b=(b_1,\ldots,b_n)$ und $a=(a_1,\ldots,a_n)$ .)*

1. **IF** $n \le 2$ **THEN** ***Output*** $\xi_M$ für das $|a_M|$ maximal ist, **STOP.**

2. Berechne den Median $\xi_S$ von $\{\xi_1,\ldots,\xi_n\}$ in linearer Zeit
(z.B. mit dem Schönhage–Paterson–Pippenger–Algorithmus [ShPP76]).

3. $A_- := \sum_{\xi_i<\xi_S} |a_i|\,,\; A_+ := \sum_{\xi_i>\xi_S} |a_i|$ .

4. $\xi_M := \begin{cases} \mathrm{WM}(\{(\xi_S, |a_S|+A_-),\ (\xi_i,a_i) \mid \xi_i>\xi_S\})\,, & \text{falls } A_+ > |a_S|+A_- \\ \mathrm{WM}(\{(\xi_S, |a_S|+A_+),\ (\xi_i,a_i) \mid \xi_i<\xi_S\})\,, & \text{falls } A_- > |a_S|+A_+ \\ \xi_S\,, & \text{sonst.} \end{cases}$

***Output*** $\xi_M$ . *($\mu = \lceil\xi_M\rceil$ oder $\mu = \lfloor\xi_M\rfloor$ minimiert $\|\,b-\mu a\,\|_1$ .)*

Die Ausgabe $\xi_M$ in Schritt 1 minimiert $f$, da sie die Ungleichungen 3.1 erfüllt, die $\xi_M$ charakterisieren. Die Korrektheit von Schritt 4 folgt rekursiv, da er die Ungleichungen 3.1 erhält. Die Schritte 1-3 benötigen $O(n)$ RAM–Schritte. Der rekursive Aufruf in Schritt 4 ist von etwa halber Eingabegröße, woraus eine Laufzeitschranke von $O(n)$ RAM–Schritten folgt.

### 3.2.2 Sortierung der Komponenten für die $l_\infty$–Norm

Für die $l_\infty$-Norm ist der Graph von $f(x) = \max_{i\le n} |b_i - xa_i|$ das Maximalpolygon der steigenden Geraden $f_i^+(x) := |a_i|(x - b_i/a_i)$ und der fallenden Geraden $f_i^-(x) := |a_i|(b_i/a_i - x)$, jeweils für $i = 1, \ldots, n$, wobei die Komponenten mit $a_i = 0$ bereits entfernt seien. Die Komponenten werden nach dem Betrag ihrer Steigung sortiert. Sei also o.B.d.A. $|a_1| \ge |a_2| \ge \ldots \ge |a_n| > 0$. Wir bezeichnen mit $f^{(k)}(x) = \max_{i\le k} |b_i - xa_i|$ das Maximalpolygon der $k$ steilsten Komponenten. Wir bestimmen sukzessive für $k = 1, \ldots, n$ die Darstellung von $f^{(k)}$ als geordnete Teilmenge von Geraden, deren Abschnitte das Maximalpolygon bilden. Mit diesen Geraden werden ihre aufeinanderfolgenden Schnittstellen, darunter die Minimalstelle $x^{(k)}$ von $f^{(k)}$, bestimmt.

Ist $f_{k+1}(x^{(k)}) \le f^{(k)}(x^{(k)})$, so gilt $f^{(k)} = f^{(k+1)}$ und die Komponente $f_{k+1}$ braucht nicht berücksichtigt zu werden. Andernfalls wird sie hinzugenommen und die Darstellung von $f^{(k+1)}$ wird neu berechnet. Das beschreiben wir im Folgenden:

Von der Komponente $f_{k+1} = \max\{f_{k+1}^+, f_{k+1}^-\}$ braucht nur die Gerade $f_{k+1}^\pm$ mit größerem Wert $f_{k+1}^\pm(x^{(k)})$ berücksichtigt werden. Treten mehrere Komponenten mit gleicher Steigung $|a_i| = \ldots = |a_j|$ auf, so wird von ihnen nur die steigende Gerade mit kleinster Nullstelle $b_{\min}/a_{\min}$ und die fallende Gerade mit größter Nullstelle $b_{\max}/a_{\max}$ berücksichtigt, da stets $f_{\min}^+ \ge f_r^+$ und $f_{\max}^- \ge f_r^-$ für $i \le r \le j$ gilt.

Wir verwalten Geradenabschnitte von $f^{(k)}$ in zwei Stacks $R, L$ wobei $R$ die steigenden und $L$ die fallenden Geraden enthält. Der Betrag der Steigung der obersten Geraden ist jeweils minimal für den Stack. Der Wert $x = x^{(k)}$ bezeichnet stets die Minimalstelle von $f^{(k)}$, nämlich die Schnittstelle der obersten Geraden der beiden Stacks. Auf jedem Stack legen wir neben einer Geraden auch ihre Schnittstelle mit der unter ihr liegenden Geraden ab. Wir bezeichnen das oberste Stackelement mit $(g_L, x_L)$ bzw. $(g_R, x_R)$ . Wir bezeichnen für zwei Geraden $g_i(\xi) = b_i - \xi a_i$ mit $g_1 \cap g_2 = \frac{b_1 - b_2}{a_1 - a_2}$ ihre Schnittstelle.

Wir initialisieren $R$ mit $(g^+, +\infty)$ und $L$ mit $(g^-, -\infty)$ wobei $g^+$ die steilste verbliebene steigende bzw. $g^-$ die steilste fallende Gerade ist und $x := g^+ \cap g^-$. Jetzt

werden alle verbliebenen Geraden in Reihenfolge von abnehmender Steigung durchlaufen. Wir unterscheiden dabei, ob die neue Gerade steigend oder fallend ist. Die beiden Fälle sind symmetrisch. Wir beschreiben die Hinzunahme einer steigenden Geraden:

**Füge steigende Gerade $g$ zu.**

1. $\tilde{x} := g \cap g_L$
2. Falls $\tilde{x} \geq x$, gehe zur nächsten Geraden.
3. Solange $\tilde{x} \leq x_L$: [POP$(L)$, $\tilde{x} := g \cap g_L$].
4. $x := \tilde{x}$.
5. $\hat{x} := g \cap g_R$
6. Solange $\hat{x} \geq x_R$: [POP$(R)$, $\hat{x} := g \cap g_R$].
7. PUSH$(R, (g, \hat{x}))$.

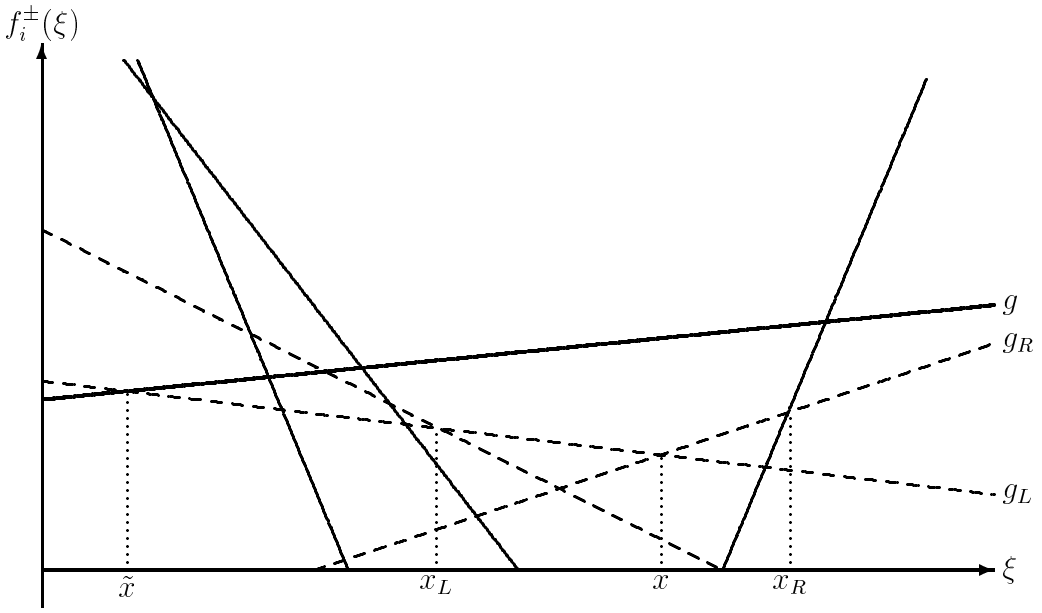


Abbildung 3.1: Hinzufügung der steigenden Gerade $g$.
(Die gestrichelten Geraden werden durch POP-Schritte entfernt.)

Für das Sortieren der Komponenten nach Steigung werden i.a. $O(n \log n)$ arithmetische Schritte benötigt. Wie in Abschnitt 3.1 gezeigt, kann der Reduktionskoeffizient in $O(n)$ Schritten berechnet werden, wenn $a$ nicht bereits ein kürzester Gittervektor

ist. In diesem Fall können die Komponenten durch Verwendung von verallgemeinertem Bucket–Sort in $O(n \log \| a \|_\infty) = O(n \log \lambda_1)$ arithmetischen Schritten sortiert werden.

Der Durchlauf der sortierten Geraden kostet höchstens $O(n)$ arithmetische Schritte. Für jede Komponente wird nämlich höchstens ein PUSH–Schritt (d.h. ein Stackelement hinzufügen) durchgeführt. Die Schleifen können also bei allen Hinzufügungen von steigenden (fallenden) Geraden insgesamt nicht mehr als $n - 1$ POP–Schritte (d.h. ein Stackelement entfernen) durchführen. Die Stacks werden nie leer, da für das unterste Stackelement jeweils $x = \pm\infty$ gilt.

# Kapitel 4

# Verbesserte Bitkomplexität des Gauß–Algorithmus

Der Gauß–Algorithmus ist eine natürliche Verallgemeinerung des zentrierten Euklidischen Algorithmus auf den Fall von zwei linear unabhängigen Eingabevektoren. Lehmer [Leh38] verwendete zuerst die Idee, im Euklidischen Algorithmus den größten Teil der arithmetischen Operationen nur auf den führenden Stellen der Zahlen durchzuführen. Schönhage [Sh71] verfeinerte diese Technike zu einem Algorithmus von niedriger Bitkomplexität. Kürzlich hat Schönhage [Sh91] diese Methoden zu einem schnellen Reduktionsalgorithmus für binäre quadratische Formen weiterentwickelt. Yap [Ya92] veröffentlichte eine Skizze eines ähnlichen Algorithmus in der Sprache der Gitterbasenreduktion. Alle diese Arbeiten betrachteten nur *monotone* Reduktionsalgorithmen, bei denen keine negativen Reduktionskoeffizienten — in unserer Notation keine negativen Vorzeichen $\varepsilon$ — auftreten. Es ist aber bekannt, daß der Gauß–Algorithmus bei *zentrierten* Reduktionsschritten effizienter ist [Va91, Da93] . Wir stellen in diesem Kapitel einen zentrierten Reduktionsalgorithmus mit niedriger Bitkomplexität vor (siehe auch [Ka94]). Eine wesentliche Erkenntnis hierfür ist, daß die Schritte des Gauß–Algorithmus stabil sind solange der Approximationsfehler $\frac{1}{12}$ der Norm des kürzesten Basisvektors nicht übersteigt. Dieses Resultat gilt für beliebige Normen. Die genaue Kontrolle des Approximationsfehlers wird ermöglicht durch die expliziten Formeln für die Transformationsmatrizen sowie die scharfen Abschätzungen der Norm von Vektoren in aufeinanderfolgenden wohlgeordneten Basen aus Kapitel 2.

## 4.1 Vorüberlegungen zu Bitkomplexität und Stabilität

Sei $\mathcal{M}(B)$ eine Schranke für die Bitkomplexität der Multiplikation von $B$-stelligen Binärzahlen. Die beste bekannte asymptotische Schranke hierfür ist $\mathcal{M}(B) = O(B \log B \log\log B)$ und stammt von Schönhage und Strassen [ShS70].

Für das Ziel einer niedrigen Bitkomplexität bei einer Variante des Gauß–Algorithmus muß jeder Reduktionsschritt an sich „effizient“ berechenbar sein. Genauer fordern wir, daß für $a, b \in \mathbb{Z}^n$ mit $\max(\| a \|, \| b \|) \leq 2^B$ der Nachfolger $\operatorname{succ}(a, b)$ und die Norm (oder eine monotone Funktion der Norm) in $O(n\mathcal{M}(B))$ Bitoperationen berechnet werden kann. Wir haben in Kapitel 3 gesehen, daß dies für die $l_1$-, $l_2$- und $l_\infty$-Norm der Fall ist und beschränken uns deshalb in diesem Kapitel auf jene drei $l_p$-Normen. Die Resultate lassen sich leicht an beliebige andere Normen anpassen.

Nach der Lektüre von Kapitel 2 wissen wir, daß die Transformationsmatrizen der Nachfolgerrelation $(\operatorname{succ}(a, b), a) = (a, b)M^{-1}$ im Gauß–Algorithmus von der Gestalt

$$M = \begin{pmatrix} 0 & \varepsilon \\ 1 & \mu \end{pmatrix} \in SL_2(\mathbb{Z}) \quad \text{wobei } \varepsilon = 1,\ \mu \geq 2 \text{ oder } \varepsilon = -1,\ \mu \geq 3$$

sind. Der Gaußsche Reduktionsschritt $(\alpha, \beta) := (\operatorname{succ}(a, b), a) = (a, b)M^{-1}$ wird von dem *Inversen* von $M$ beschrieben. Wir verwenden wesentlich die Charakterisierung der Vorgänger einer wohlgeordneten Basis aus Lemma 22. Daraus folgt für eine wohlgeordnete Basis $(\alpha, \beta)$, daß das Produkt $(a, b) = (\alpha, \beta)\, M$ wohlgeordnet ist und $(\alpha, \beta) = (\operatorname{succ}(a, b), a)$. Wir nennen ein Produkt solcher Matrizen $M_t \cdots M_1$, das die Transformation $(a, b) = (\alpha, \beta)\, M_t \cdots M_1$ von $t$ aufeinanderfolgenden Reduktionsschritten beschreibt, kurz *Reduktionsmatrix*.

Ziel dieses Kapitels ist es, durch Verwendung von approximativer Arithmetik eine asymptotisch kleine Bitkomplexität zu erreichen. Dies ist möglich, da die Schritte des Gauß–Algorithmus stabil sind solange der Approximationsfehler unterhalb einer impliziten Schwelle liegt. Um dies zu zeigen definieren wir eine Eigenschaft, die alle bis auf die letzten drei Basen in jedem Lauf des Gauß–Algorithmus erfüllen. Daraus folgt eine Schwelle für den Approximationsfehler, unterhalb derer die Wohlordnung der Basen erhalten bleibt.

**Definition 36.** *Eine Gitterbasis $(a, b)$ heißt*

1. strikt wohlgeordnet (swo) *falls*
   - $\frac{5}{4} \| a \| \leq \| a - b \| < \| b \| - \frac{1}{2} \| a \|$ *und*
   - $2 \| a \| \leq \| b \|$ ,
2. $\sigma$–minimal *für eine Schwelle $\sigma \in \mathbb{R}$ falls $(a, b)$ strikt wohlgeordnet ist mit $\| a \| \geq \sigma$ und entweder*
   - $\| \operatorname{succ}(a, b) \| < \sigma$ *oder*
   - $(\operatorname{succ}(a, b), a)$ *ist nicht strikt wohlgeordnet.*

Diese Begriffe verallgemeinern das Konzept der wohlgeordneten und reduzierten Basen des Gauß–Algorithmus bei exakter Arithmetik. Im schnellen Algorithmus erzeugt jeder Reduktionsschritt entweder eine strikt wohlgeordnete oder eine $\sigma$-minimale Basis. Ist die erzeugte Basis $\sigma$-minimal, so würde der Algorithmus nach spätestens drei weiteren Schritten eine reduzierte Basis erzeugen. Sind die Vektoren nur Approximationen der eigentlichen Basis, so ist in diesem Falle die Norm der Vektoren zu klein in Relation zum Approximationsfehler und der schnelle Algorithmus ruft sich mit angepaßter Genauigkeit erneut auf. Wir werden sehen, daß der Approximationsfehler auf diese Weise genau kontrolliert werden kann.

Eine strikt wohlgeordnete Basis bleibt auch bei geringen Störungen wohlgeordnet:

**Lemma 37.** *Es sei $(a, b)$ eine strikt wohlgeordnete Basis und* $\max\{\| \Delta_a \|, \| \Delta_b \|\} \leq \frac{1}{12} \| a \|$ . *Dann ist $(a + \Delta_a,\ b + \Delta_b)$ wohlgeordnet.*

**Beweis**. Man sieht unmittelbar:

$$\| b + \Delta_b \| - \| a + \Delta_a - (b + \Delta_b) \| \geq \tfrac{1}{2} \| a \| - 2 \| \Delta_a \| - \| \Delta_b \| > 0$$

$$\| a + \Delta_a - (b + \Delta_b) \| - \| a + \Delta_a \| \geq \tfrac{1}{4} \| a \| - 2 \| \Delta_a \| - \| \Delta_b \| \geq 0 \ .$$

□

Das folgende Lemma sagt aus, daß in einem Lauf des Gauß–Algorithmus die Norm der Differenz der Basisvektoren sich nicht beliebig der Norm der Vektoren selbst nähern kann:

**Lemma 38.** *Seien* $(a,b)$, $(b,c)$ *wohlgeordnete Gitterbasen mit* $a = \mathrm{succ}\,(b,c) = \varepsilon(c-\mu b)$ *und* $\| \mathrm{succ}\,(a,b) \| \leq \Delta \| b \|$, *wobei* $\Delta < 1$. *Dann gilt*

1. $\| c \| - \| c-b \| \geq \| b \| - \| a \|$
2. $\| c-b \| - \| b \| \geq \frac{1-\Delta}{2} \| b \|$ .

Daraus folgt, daß einem Lauf des Gauß–Algorithmus höchstens die letzten drei Basen nicht strikt wohlgeordnet sind:

**Korollar 39.** *Seien* $(b_0,b_1)$, $(b_1,b_2)$, $(b_2,b_3)$ *wohlgeordnete Gitterbasen mit* $b_i = \mathrm{succ}\,(b_{i+1},b_{i+2})$. *Dann ist* $(b_2,b_3)$ *strikt wohlgeordnet.*

Zur Reduktion einer gegebenen Gitterbasis $(a,b) \in \mathbb{Z}^n$ genügt es also, eine 0-minimale Basis desselben Gitters zu erzeugen. Diese ist dann swo, ihre Nachfolgerbasis aber nicht. Also führt der Gauß–Algorithmus auf ihr höchstens drei Reduktionsschritte durch. Eine genauere, hier nicht ausgeführte Fallunterscheidung zeigt sogar, daß im dritten Reduktionsschritt $\mu = 1$ gelten muß und mithin für eine 0-minimale Basis $a,b$ gilt: $\| \mathrm{succ}\,(a,b) \| \leq 2\lambda_2(L_{a,b})$ .

**Beweis von Korollar 39.** Wir wenden Lemma 38 an mit $b_1 = a$, $b_2 = b$, $b_3 = c$. Da $(b_0,b_1)$ wohlgeordnet ist, gilt $\| b_1 \| \leq \frac{1}{2} \| b_2 \|$ und folglich ist

$$\| b_3 \| - \| b_3 - b_2 \| \geq \| b_2 \| - \| b_1 \| \geq \frac{1}{2} \| b_2 \| .$$

Weiterhin ist $\| b_0 \| < \| b_1 \| \leq \frac{1}{2} \| b_2 \|$ , also folgt mit $\Delta = \frac{1}{2}$:

$$\| b_3 - b_2 \| - \| b_2 \| \geq \tfrac{1}{4} \| b_2 \| . \qquad \square$$

**Beweis von Lemma 38.** Ad 1: Sei $\rho := \| b \| / \| a \|$. Wir unterscheiden zwei Fälle:

$\underline{\varepsilon = 1}$: Es gilt

$$\begin{aligned} \| c-b \| &= \| (\mu-1)b + a \| \\ &\leq (\mu-1) \| b \| + \frac{1}{\rho} \| b \| \\ &\leq (\mu - 1 + \frac{1}{\rho}) \frac{1}{\mu} \| c \| \\ &= (1 - \frac{\rho-1}{\rho\mu}) \| c \| . \end{aligned}$$

Folglich ist

$$\| c \| - \| c - b \| \geq \frac{\rho - 1}{\rho\mu} \| c \| \geq \frac{\rho - 1}{\rho\mu}\mu \| b \| = \| b \| - \| a \| .$$

$\underline{\varepsilon = -1}$: Sei $\eta = \mu - 1 - \frac{1}{\rho}$ und $G$ die Gerade $G(\xi) = (1-\xi)\eta\rho a + \xi\eta b$. Es gilt $\| G(0) \| = \eta\rho \| a \| = \eta \| b \| = \| G(1) \|$ und somit nach Lemma 15 auch $\eta\rho \| a \| \leq \| G(\frac{\mu-1}{\eta}) \| = \| c - b \|$ . Bezeichne $H$ die Gerade $H(\xi) = (1-\xi)(-\eta\rho a) + \xi(c-b)$. Wegen $\| H(0) \| = \eta\rho \| a \| \leq \| c - b \| = \| H(1) \|$ gilt dann wie oben $\| c - b \| \leq \| H(\frac{\mu\rho}{\mu\rho-1}) \| = (1 - \frac{\rho-1}{\mu\rho-1}) \| c \|$ . Zusammen ergibt sich also

$$\| c \| - \| c - b \| \geq \frac{\rho - 1}{\mu\rho - 1} \| c \| \geq \frac{\rho - 1}{\mu\rho - 1}(\mu - 1) \| b \| \geq \| b \| - \| a \| .$$

Ad 2: Bezeichne $x := \mathrm{succ}\,(a,b) = \tilde{\varepsilon}(b - \tilde{\mu}a)$ den Nachfolger von $(a,b)$ und $G$ die Gerade $G(\xi) = (1-\xi)\lambda b + \xi c$ wobei $\lambda = \frac{(2\tilde{\mu}+1)/\Delta - 1}{\tilde{\mu}/\Delta}$. Aus Lemma 26 folgt $G(1) = \| c \| \geq \lambda \| b \| = G(0)$ , also, wieder unter Verwendung von Lemma 15:

$$\frac{\lambda}{\lambda - 1} \| c - b \| = \| G(\tfrac{\lambda}{\lambda-1}) \| \geq \| G(0) \| = \lambda \| b \| .$$

Somit folgt Behauptung 2:

$$\| c - b \| - \| b \| \geq (\lambda - 2) \| b \| \geq \tfrac{1-\Delta}{2} \| b \| . \qquad \square$$

Die Schranken für die Differenz der Vektoren in dem vorangehenden Lemma und Korollar lassen sich noch etwas verschärfen. Wir stellen diese Eigenschaften hier kurz vor, verzichten jedoch auf vollständige Beweise, da die Aussagen im weiteren nicht benötigt werden.

**Bemerkungen.** Unter den Voraussetzungen von Lemma 38 gilt

1. $\| c \| - \| c - b \| \geq \frac{2}{5} \| b \|$ ,
2. $\| c - b \| - \| b \| \geq \| b \| - \| a \|$ falls $\mu \geq 3$ .

**Beweis von Aussage 1.** Wir unterscheiden zwei Fälle:

$\underline{\varepsilon = 1}$:

$$\begin{aligned} \| c - b \| &= \| (\mu - 1)b + a \| \\ &\leq (\mu - 1) \| b \| + \frac{1}{2} \| b \| \\ &\leq (\mu - \frac{1}{2})\frac{1}{\mu} \| c \| \\ &= (1 - \frac{1}{2\mu}) \| c \| . \end{aligned}$$

Also ist

$$\| c \| - \| c - b \| \geq \frac{1}{2\mu} \| c \| \geq \frac{1}{2\mu}\mu \| b \| = \frac{1}{2} \| b \| .$$

$\underline{\varepsilon = -1}$: Wir verwenden die Gerade $G(\xi) = (1 - \xi)(-2a) + \xi(c - b)$. Es gilt $\| G(0) \| = 2 \| a \| \leq \| b \| \leq \| c - b \| = \| G(1) \|$ . Also ist $\| c - b \| \leq \| G(\frac{2\mu}{2\mu-1}) \| = \| (1 - \frac{1}{2\mu-1}) c \|$ und somit

$$\begin{aligned} \| c \| - \| c - b \| &\geq \tfrac{1}{2\mu-1} \| c \| \\ &\geq \tfrac{1}{2\mu-1}(\mu - 1) \| b \| \\ &\geq \tfrac{2}{5} \| b \| \qquad \text{(weil } \mu \geq 3 \text{ ).} \end{aligned}$$

□

Die vorangehenden Überlegungen beweisen folgende *zentrale Stabilitätsbetrachtung:*

**Korollar 40.** *Seien* $b_0, b_1, b_2, \ldots$ *mit* $b_i := \operatorname{succ}(b_{i+1}, b_{i+2})$ *die letzten in einem Lauf des Gauß–Algorithmus erzeugten Vektoren. Dabei sei* $(b_0, b_1)$ *die reduzierte Ausgabebasis,* $(b_i, b_{i+1})$ *swo für* $i \geq t$ *aber* $(b_{t-1}, b_t)$ *nicht swo. Wegen Korollar 39 gilt* $t \leq 3$. *Seien die beiden Vektoren* $\bar{b}_t$, $\bar{b}_{t+1}$ *Approximationen von* $b_t$, $b_{t+1}$ *so daß* $\| \bar{b}_i - b_i \| \leq \frac{1}{12} \| b_i \|$ *für* $i = t,\ t+1$ . *Bezeichne* $\varphi$ *die durch* $b_i \mapsto \bar{b}_i$ $(i = t,\ t+1)$ *definierte lineare Abbildung und* $\bar{b}_i := \varphi(b_i)$ *für* $i > t+1$ .

*Dann ist* $(\bar{b}_i, \bar{b}_{i+1})$ *wohlgeordnet für* $i \geq t$ *und wegen Korollar 39 sogar swo für* $i \geq t+2$ . *Der Gauß–Algorithmus bricht auf Eingabe* $(\bar{b}_t, \bar{b}_{t+1})$ *nach höchstens 5 Gaußschritten ab.*

## 4.2 Der schnelle Algorithmus

Die vorangehenden Ergebnisse geben uns das Muster für das Vorgehen im schnellen Algorithmus:

- Wird die Approximationsgenauigkeit erhöht oder erniedrigt, beschrieben durch Addition von Fehlertermen $\Delta$ in $(a, b) := (a + \Delta_a,\ b + \Delta_b)$, so erfüllt deren Norm stets $\max(\| \Delta_a \|, \| \Delta_b \|) \ \leq \ \frac{1}{12} \| a \|$.

- Tritt durch Wechsel der Approximationsgüte eine wohlgeordnete aber nicht strikt wohlgeordnete Basis auf, so werden die Transformationen der (höchstens 2) letzten Gaußschritte aufgehoben, bis man wieder bei einer strikt wohlgeordneten Basis angelangt ist.

Zur Rekonstruktion der vollen Genauigkeit nach Abschluß eines rekursiven Aufrufes muß die Reduktionsmatrix $M$, die die vom Algorithmus auf den Basisvektoren durchgeführte Transformation beschreibt, mitgeführt werden. Für das Aufheben einzelner Schritte ist es sogar nötig, die Faktorisierung von $M = M_t \cdot \ldots \cdot M_1$ in die Transformationsmatrizen aller Gaußschritte zu speichern. In unserer Notation lesen sich die Prozeduren für die Durchführung bzw. Aufhebung von Gaußschritten folgendermaßen:

**Gaußschritt:** $t := t + 1$, ($t$ zählt die Gaußschritte)
$(\alpha, \beta) := \mathrm{succ}\,(\alpha, \beta) =: (\alpha, \beta) M_t^{-1}, \quad M := M_t M.$

**Rückschritt:** $(\alpha, \beta) := (\alpha, \beta) M_t\,, \quad M := M_t^{-1} M, \quad t := t - 1.$

Um Approximation von Vektoren im allgemeinen Fall betrachten zu können, verallgemeinern wir die „Nächste–Ganze–Zahl“–Funktion auf $n$-dimensionale Vektoren. Sei $\lfloor . \rceil : \mathbb{R}^n \longrightarrow \mathbb{Z}^n$ irgendeine Funktion, die $\| x - \lfloor x \rceil \|$ minimiert und die in $O(n\, \mathcal{M}(\log \| x \|))$ Bitoperationen zu berechnen ist. Wir bezeichnen die schlechteste Approximation mit

$$\Gamma \ := \ \sup_{x \in \mathbb{R}^n} \ \inf_{\omega \in \mathbb{Z}^n} \ \| x - \omega \| \ .$$

Für $l_p$-Normen gilt $\Gamma = \frac{1}{2} n^{1/p}$, denn hier ist lediglich eine nächste ganze Zahl in jeder Komponente zu wählen. Im schnellen Algorithmus verwenden wir $\Gamma$ in Form der technischen Konstante $\theta \ = \ \lceil \frac{1}{2} \log_2 \Gamma / \tau - 1 \rceil \ \approx \ \lceil \frac{1}{2p} \log_2 n + 1.31 \rceil$ wobei $\tau = \sqrt{\frac{13}{12}} - 1$.

**Algorithmus** $FG(a,b,m)$.

***Eingabe:*** Wohlgeordnete Basis $(a,b)$, Schwelle $m \in \mathbb{N}$.

1. $M := I$, $(\alpha,\beta) := (a,b)$
   **IF** $\| a \| \leq 2^{m+\theta+1}$ **THEN GOTO** 8.

2. Wähle $d$ minimal mit $\| b \| \leq 2^{m+d}$,
   $m' := \min(m,d)$, $h := m' + \lfloor \frac{d+\theta}{2} \rfloor$, $h' := \lceil m' + \theta \rceil$,
   **IF** $m \leq d$ **THEN GOTO** 4.

3. $k := m - d + 1$, $(\alpha,\beta) := (\lfloor 2^{-k} a \rceil, \lfloor 2^{-k} b \rceil)$,
   **IF** $(\alpha,\beta)$ nicht swo **THEN** $[(\alpha,\beta) := (a,b)$, **GOTO** 8 $]$.

4. **IF** $\| \alpha \| \geq 2^h$ **THEN** $(\alpha,\beta,M) := FG(\alpha,\beta,h)$.

5. **WHILE** $\| \beta \| > 2^h$ und $\| \alpha \| \geq 2^{h'}$ und $(\alpha,\beta)$ swo **DO** Gaußschritt
   **IF** $\| \alpha \| < 2^{h'}$ oder $(\alpha,\beta)$ nicht swo **THEN** [Rückschritt, **GOTO** 7].

6. $(\alpha,\beta,M') := FG(\alpha,\beta,h')$, $M := M' M$.

7. **IF** $m > d$ **THEN** $(\alpha,\beta) := (a,b)\, M^{-1}$.

8. **WHILE** $(\alpha,\beta)$ swo und $\| a \| \geq 2^m$ **DO** Gaußschritt,
   **WHILE** $(\alpha,\beta)$ nicht swo **DO** Rückschritt.

***Ausgabe:*** $(\alpha,\beta,M)$ wobei

- $(\alpha,\beta)$ eine $2^m$–minimale Basis ist und
- $M$ die Reduktionsmatrix für $(\alpha,\beta)\, M = (a,b)$ mit Faktorisierung in die Matrizen der einzelnen Gaußschritte ist.

Aus der rekursiven Konstruktion folgt, daß die Ausgabe die geforderten Bedingungen erfüllt. Die grundsätzliche Idee für die Abschätzung der Schrittzahl ist, daß der in Schritt 2 eingeführte Parameter $d$ ein Maß für den *Abstieg* des Algorithmus ist. Wir werden sehen, daß die beiden rekursiven Aufrufe des Algorithmus in Schritt 4 und 6 höchstens Abstieg $d/2$ haben. Wir erhalten folgende Schranke für die Zahl der Bitoperationen:

**Satz 41.** *Bei Eingabe von $a,b \in \mathbb{Z}^n$ mit $\| a \|, \| b \| \leq 2^B$ bricht Algorithmus FG nach höchstens $O(n\,(1+\log n^{1/p})\, \mathcal{M}(B) \log B)$ Bitoperationen ab.*

**Bemerkung.** Für die drei Eingangs erwähnten Normen ergeben sich folgende Schranken für die Bitkomplexität:

$$\begin{aligned} l_2 - \text{norm} \quad & O\left(\mathcal{M}\left(B\right)\left(n + \log B\right)\right) \\ l_\infty - \text{norm} \quad & O\left(\mathcal{M}\left(B\right)\, n\, \log B\right) \\ l_1 - \text{norm} \quad & O\left(\mathcal{M}\left(B\right)\, n\, \log B\, \log n\right)\ . \end{aligned}$$

Dabei folgt die Schranke für die $l_2$-Norm durch Reduktion auf den Fall $n = 2$. Hier lassen sich nämlich, wie im letzten Kapitel bereits erwähnt, sämtliche Operationen unter Verwendung der Gram–Matrix $(a,b)^\top(a,b)$ anstelle der ganzen Vektoren $a, b \in \mathbb{Z}^n$ durchführen. Die anfängliche Berechnung der Gram–Matrix und die Berechnung der Ausgabebasis aus der Gram–Matrix und der Eingabebasis erfordern $O(n\,\mathcal{M}(B))$ Bitoperationen.

## 4.3 Beweis der Schrittzahl

**Notation.** Bezeichne $(\alpha_3, \beta_3) = (\lfloor 2^{-k}a \rceil, \lfloor 2^{-k}b \rceil) = (2^{-k}a - \Delta_a, 2^{-k}b - \Delta_b)$ die in Schritt 3 erhaltene Approximation der Basis aus den führenden Bits der Vektoren. Bezeichne weiter $(\alpha_6, \beta_6)$ die Ausgabebasis des rekursiven Aufrufs in Schritt 6 und $(\alpha, \beta)$ die in Schritt 7 erhaltene Basis in voller Genauigkeit. Sei $M$ die Reduktionsmatrix für die Berechnungen auf Approximationen der Basis so daß $(\alpha_6, \beta_6) = (\alpha_3, \beta_3)M^{-1}$. Somit liest sich die Wiederherstellung der vollen Genauigkeit in Schritt 7 als

$$\begin{aligned} (\alpha, \beta) &= (a,b)\,M^{-1} \\ &= 2^k\left[(\alpha_3, \beta_3) + (\Delta_a, \Delta_b)\right] M^{-1} \\ &= 2^k\left[(\alpha_6, \beta_6) + (\Delta_a, \Delta_b)M^{-1}\right] \\ &=: 2^k\left[(\alpha_6, \beta_6) + (\Delta_\alpha, \Delta_\beta)\right]\ . \end{aligned}$$

**Schranken für die Reduktionsmatrix $M$.** Zur Abschätzung der Norm der Fehlervektoren $\Delta_\alpha, \Delta_\beta$ verwenden wir die in Lemma 27 explizit angegebene Form der Reduktionsmatrix $M$. Danach hat $M$ die Gestalt

$$M = \begin{pmatrix} \varepsilon p & \varepsilon q \\ r & s \end{pmatrix}$$

wobei $\varepsilon = \pm 1$ das Vorzeichen des letzten Reduktionskoeffizienten ist und $p, q, r, s$ die positiven, ganzzahligen Werte der Kontinuanten sind, die nach Lemma 29 die Ungleichungen $s \geq 2r, 2q \geq 4p \geq 0$ und $\varepsilon(ps - qr) = \det M = \pm 1$ erfüllen. Also gilt

$$\| \beta_3 \| \;=\; \| \varepsilon q \alpha_6 + s\beta_6 \| \;\geq\; (s - \frac{q}{2}) \, \| \beta_6 \| \;\geq\; \frac{4}{3} \, s \, \| \beta_6 \| \, .$$

**Andere elementare Abschätzungen.** Es gilt

$$(\Delta_\alpha, \Delta_\beta) \;=\; (\Delta_a, \Delta_b) \, M^{-1} \;=\; \pm \left( s\Delta_a - r\Delta_b, \varepsilon \left( q\Delta_a - p\Delta_b \right) \right) \, ,$$

wobei aus den letzten Ungleichungen $r + s \leq 2\frac{\|\beta_3\|}{\|\beta_6\|}$ und $p + q \leq \frac{\|\beta_3\|}{\|\beta_6\|}$ folgt. Nach Definition von $\Gamma$ gilt $\| \Delta_a \|, \| \Delta_b \| \leq \Gamma$, und somit ist $\| \Delta_\alpha \| \leq 2\frac{\|\beta_3\|}{\|\beta_6\|}\Gamma$ sowie $\| \Delta_\beta \| \leq \frac{\|\beta_3\|}{\|\beta_6\|}\Gamma$ . Nach Schritt 1 gilt $\theta < d - 2$ und nach Schritt 2 $\Gamma \leq \tau 2^{2\theta+2}$ wobei $\tau = (\sqrt{\frac{13}{12}} - 1)$. Somit gilt auch

$$\| \beta_3 \| \;\leq\; 2^{-k} \, \| b \| \;+\; \Gamma \;\leq\; 2^{2d-1} \;+\; \Gamma \;\leq\; (1 + \frac{\tau}{2}) \, 2^{2d-1}.$$

Aus der $2^{h'}$-Minimalität von $(\alpha_6, \beta_6)$ folgt $2^{h'} \leq \| \alpha_6 \| \leq \frac{1}{2} \| \beta_6 \|$ . Unter Verwendung der Vorausgehenden Ungleichungen folgen leicht die engültigen Schranken für die Approximationsfehler:

$$\| \Delta_\alpha \| \;\leq\; \frac{1}{12} \, \| \alpha_6 \| \quad \text{und} \quad \| \Delta_\beta \| \;\leq\; \frac{1}{24} \, \| \alpha_6 \| \, .$$

**Lemma 42.** *Algorithmus FG führt höchstens*

- 2 *Gaußschritte in Schritt 5*
- $\theta + 3$ *Gaußschritte in Schritt 8*
- 2 *Rückschritte in Schritt 8*

*aus.*

**Beweis für Schritt 5.** Wie in Satz 25 gezeigt verkleinert jeder Gaußschritt die Norm der Vektoren mindestens um einen Faktor 2. Sei die Basis bei Eintritt in Schritt 5 mit $(\alpha_5, \beta_5)$ bezeichnet und $\omega_5 = \mathrm{succ}\,(\alpha_5, \beta_5)$ ihr Nachfolger. Wegen der $2^h$–Minimalität von $(\alpha_5, \beta_5)$ gilt $\| \omega_5 \| < 2^h$. Also ist $\| \beta \| \leq 2^h$ nach höchstens zwei Iterationen. □

**Beweis für Schritt 8.** Der Algorithmus erreicht Schritt 8 nach der Ausführung von Schritt 7 oder nach einem Sprung von Schritt 1 oder Schritt 3. Wir untersuchen diese Fälle separat. Wir können dabei $m > d$ annehmen, denn andernfalls werden die Schritte 3 und 7 übersprungen und somit ist die Ausgabe der rekursiven Aufrufs in Schritt 6 bereits $2^{m+\theta}$–minimal.

Sprung von Schritt 1: Es gilt $\| \alpha \| < 2^{m+\theta+1}$. Nach Satz 25 fällt die Norm bei jedem Gaußschritt mindestens um den Faktor 2. Daher ist $(\alpha, \beta)$ nach spätestens $\theta + 1$ Gaußschritten $2^m$–minimal.

Sprung von Schritt 3: In diesem Fall ist $(\alpha_3, \beta_3)$ nicht swo. Wir verifizieren zunächst die Ungleichung $\max\{\| \Delta_a \|, \| \Delta_b \|\} \leq \frac{1}{12} \| a \|$ . Dies sieht man leicht unter Verwendung von $d < m$:

$$\frac{12\Gamma}{\| a \|} \leq 12\,\tau 2^{2\theta+2-m-\theta} < 12\,\tau 2^{\theta+d-m-\theta-1} \leq 3\tau < 1 .$$

Folglich ist $(\alpha_3, \beta_3)$ wohlgeordnet. Aus der Korollar 40 folgt, daß der Algorithmus in diesem Fall nach höchstens 5 Gaußschritten auf $(a, b)$ abbricht.

Sprung von Schritt 7: Es sei der Nachfolger von $(\alpha_6, \beta_6)$ mit $\omega_6 = \mathrm{succ}\,(\alpha_6, \beta_6) = \varepsilon(\beta_6 - \mu\alpha_6)$ bezeichnet. Wir können annehmen, daß $(\alpha_6, \beta_6)$ strikt wohlgeordnet ist, denn andernfalls werden keine Gaußsschritte in Schritt 8 ausgeführt. Also gilt wegen der $2^{h'}$-Minimalität von $(\alpha_6, \beta_6)$ die Ungleichung $\| \omega_6 \| < 2^{h'}$. Weiter erfüllen wegen $\| \Delta_\alpha \| \leq \frac{1}{12} \| \alpha_6 \|$ die gleichen Koeffizienten $\mu, \varepsilon$ die Transformationsgleichung

$$\omega = \mathrm{succ}\,(\alpha, \beta) = \varepsilon(\beta - \mu\alpha) = 2^k[\omega_6 + \varepsilon(\Delta_\beta - \mu\Delta_\alpha)] .$$

Wir verwenden

$$\mu = \frac{\| \beta_6 - \varepsilon\omega_6 \|}{\| \alpha_6 \|} \leq \frac{3}{2}\frac{\| \beta_6 \|}{\| \alpha_6 \|}$$

zur Abschätzung der Approximationsfehler:

$$\begin{aligned} \| \Delta_\beta \| + \mu \| \Delta_\alpha \| &\leq \frac{\| \beta_3 \|}{\| \beta_6 \|} \Gamma + \frac{3 \| \beta_6 \|}{2 \| \alpha_6 \|} 2 \frac{\| \beta_3 \|}{\| \beta_6 \|} \Gamma \\ &= \left(\frac{\| \beta_3 \|}{\| \beta_6 \|} + \frac{3 \| \beta_3 \|}{\| \alpha_6 \|}\right) \Gamma \\ &\leq \frac{7}{2} \Gamma \frac{\| \beta_3 \|}{\| \alpha_6 \|} \\ &\leq \frac{7}{2} \tau \, 2^{2\theta+2} \left(1 + \frac{\tau}{2}\right) 2^{2d-1} \cdot 2^{-d-\theta} \\ &= \frac{7}{24} \, 2^{d+\theta} \, . \end{aligned}$$

Somit gilt

$$\| \omega \| \leq 2^k [2^{d+\theta} + \frac{7}{24} 2^{d+\theta}] = \frac{31}{12} 2^{m+\theta} \, .$$

Also bricht der Algorithmus in Schritt 8 nach höchstens $\theta + 1 + \log_2 \frac{31}{12}$ Gaußschritten ab. □

**Abschätzung der Schrittzahl.** Der wesentliche Parameter für die Rechenzeitanalyse ist der in Schritt 2 bestimmte Abstiegswert $d$. Wir bezeichnen mit $d_6$ den Abstiegswert des rekursiven Aufrufs in Schritt 6 und mit $d_4$ den Abstiegswert des rekursiven Aufrufs in Schritt 4. Dann ist

$$d_6 = h - h' = \lfloor \frac{d - \theta}{2} \rfloor$$

sowie

$$d_4 = \lceil \log_2 \| \beta_3 \| \rceil - h \leq 2d - 1 + \log_2(1 + \frac{\tau}{2}) - d - \lfloor \frac{d+\theta}{2} \rfloor < \lceil \frac{d-\theta}{2} \rceil \, ,$$

und für $m \leq d$

$$d_4 = m + d - (m + \lfloor \frac{d+\theta}{2} \rfloor) = \lfloor \frac{d-\theta}{2} \rfloor \, .$$

Die beiden rekursiven Aufrufe haben also höchstens den halben Abstiegswert. Alle anderen Schritte können bei geschickter Implementierung in $O(n \ (1 + \log n^{1/p}) \ \mathcal{M}(B))$ Bitoperationen durchgeführt werden. Dazu beachte man, daß der

Algorithmus außerhalb der beiden rekursiven Aufrufe nach Lemma 42 höchstens $\lceil \frac{1}{2p} \log_2 n + 1.307 \rceil + 3$ Gaußschritte ausführt. Jeder Gaußschritt benötigt $O(n\ \mathcal{M}(B))$ Bitoperationen. Die Fehlerkorrektur in Schritt 7 kann sehr effizient durch die Berechnung

$$(\alpha, \beta) \ := \ 2^k\,(\alpha, \beta) + (a - 2^k\,\lfloor 2^{-k}\,a \rceil, b - 2^k\,\lfloor 2^{-k}\,b \rceil)\,M^{-1}\,.$$

ausgeführt werden. Bezeichne $T(d)$ die Zahl der Bitoperationen, die der Algorithmus auf einer Eingabe von Abstieg $d$ und Norm $O(2^d)$ benötigt. Im nichttrivialen Fall gilt $m > d$ und folglich auch $d < B < 2d$ holds. Folglich liefert die Rekursion

$$\begin{aligned} T(d) &\leq 2T(\frac{d}{2}) + O(\theta\ n\ \mathcal{M}(d)) \\ &= O(n\theta\ \mathcal{M}(d)\ \log\ d) \end{aligned}$$

□

# Literaturverzeichnis


[CR88] B. Chor, R. Rivest: A knapsack–type public key cryptosystem based on arithmetc in finite fields. IEEE Trans. Information Theory IT-34 (1988), pp. 578-593.

[Co&a92] M. Coster, A. Joux, B. LaMacchia, A.M. Odlyzko, C.P. Schnorr, J. Stern: Improved Low-Density Subset Sum Algorithms.

[Da93] H. Daudé: Des fractions continues a la réduction des réseaux: Analyse en moyenne. Thèse de doctorat, Université de Caen 1993.

[Di1850] G.L. Dirichlet: Über die Reduction der positiven quadratischen Formen mit drei unbestimmten ganzen Zahlen. J. reine angew. Math. 40 (1850), pp. 228-232.

[Du1846] A. Dupré: Sur le nombre de divisions à effectuer pour obtenir le plus grand commun diviseur entre deux nombres entiers. J. de Math., vol. 11 (1846), pp. 41-64.

[Ga1801] C.F. Gauss: Disquisitiones Arithmeticae. Leipzig 1801. German translation: Untersuchungen über die höhere Arithmetik. Springer, Berlin 1889. (reprint: Chelsea, New York, 1981.)

[GL87] P.M. Gruber, C.G. Lekkerkerker: Geometry of Numbers (second edition). North Holland Mathematical Library 37 (1987).

[He1850] C. Hermite: Extraits de lettres de M. Ch. Hermite à M. Jacobi sur differents objets de la théorie des nombres, Deuxième lettre. J. Reine Angew. Math, vol. 40 (1850), pp. 279-290.

[Jo48] F. John: Studies and essays presented to R. Courant. (1948), pp. 187-204.

[JG94] A. Joux, L. Granboulan: A practical attack against knapsack based hash functions. Proc. Eurocrypt '94, pp. 61-70.

[Ka91] M. Kaib: The Gauß Lattice Basis Reduction Algorithm Succeeds With Any Norm. Proceedings of the FCT'91, Springer Lecture Notes on Computer Science, vol. 529 (1991), pp. 275-286.

[Ka94] M. Kaib: A Fast Variant of the Gausssian Lattice Basis Reduction Algorithm. Technical Report, Universität Frankfurt (1994). Proceedings of the ANTS'94, Springer Lecture Notes on Computer Science.

[KS93] M. Kaib, C.P. Schnorr: The Generalized Gauß Reduction Algorithm. Technical Report, Universität Frankfurt (1993). Eingereicht für J. Algorithms.

[KR94] M. Kaib, H. Ritter: Block Reduction for Arbitrary Norms. Preprint, Universität Frankfurt 1994.

[KZ1873] A. Korkine, G. Zolotarev: Sur les formes quadratiques. Math. Ann., vol. 6 (1873), pp. 336-389.

[La1773] L. Lagrange: Recherches d'arithmetique. Nouv. Mém. Acad. Berlin (1773), pp. 265-312.

[Leh38] D.H. Lehmer: AMM, vol. 45, (1938), pp. 227-233.

[Len83] H.W. Lenstra, Jr: Integer programming with a fixed number of variables. Mathematics of Operations Research, Vol. 8, No. 4, (1983), pp. 538-548.

[LLL82] A.K. Lenstra, H.W. Lenstra, Jr., L. Lovász: Factoring polynomials with rational coefficients. Math. Annalen 261 (1982), pp. 515-534.

[LS92] L. Lovász, H. Scarf: The Generalized Basis Reduction Algorithm. Mathematics of Operations Research, vol. 17, No. 3 (1992), pp. 754-764.

[Ma38] K. Mahler: On Minkowski's theory of reduction of positive definite quadratic forms. Quarterly J. Math. 9, (1938) pp. 259-262.

[Mi1891] H. Minkowski: Über die positiven quadratischen Formen und über kettenbruchähnliche Algorithmen. Journal f. reine u. angew. Math. 107 (1891), pp. 278-297.

[R78] G.J. Rieger: Über die mittlere Schrittzahl bei Divisionsalgorithmen. Math. Nachr. 82 (1978), pp. 157-180.

[R94] H. Ritter: Aufzählung von kurzen Gittervektoren bezüglich beliebiger Norm. Preprint, Universität Frankfurt 1994.

[S87] C.P. SCHNORR: A Hierarchy of Polynomial Time Lattice Basis Reduction Algorithms. Theoretical Computer Science 53 (1987), pp. 201-224.

[S89] C.P. SCHNORR: A more efficient algorithm for lattice basis reduction. J. Algorithms 9 (1988), pp. 47-62.

[S92] C.P. SCHNORR: Factoring Integers and Computing Discrete Logarithms via Diophantine Approximation. In: Advances in Computational Complexity, Ed. Jim–Yi Cai, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, AMS (1993).

[S94] C.P. SCHNORR: Block Reduced Lattice Bases and Successive Minima. Technical Report, ICSI Berkeley (1992), 18 Seiten. Erscheint in Combinatorics, Probability and Computing.

[S94p] C.P. SCHNORR: Private Korrespondenz. (1994), 2 Seiten.

[SE91] C.P. SCHNORR, M. EUCHNER: Lattice basis reduction: improved algorithms and solving subset sum problems. Proceedings of the FCT'91, Springer Lecture Notes on Computer Science, vol. 529 (1991), pp. 68-85. Erscheint in: Mathematical Programming Studies.

[Sh71] A. SCHÖNHAGE: Schnelle Berechnung von Kettenbruchentwicklungen. Acta Informatica 1, (1971), pp. 139-144.

[Sh91] A. SCHÖNHAGE: Fast Reduktion and Composition of Binary Quadratic Forms. In: Proc. ISSAC 1991, Ed. S.M. Watt, ACM 1991, pp. 128-133.

[ShPP76] A. SCHÖNHAGE, M. PATERSON, M. PIPPENGER: Finding the Median, JCSS 13 (1976), pp. 184-199.

[ShS70] A. SCHÖNHAGE, V. STRASSEN: Schnelle Multiplikation großer Zahlen. J. Computing.

[Va91] B. VALLÉE: Gauss' Algorithm Revisited. J. Algorithms 12 (1991), pp. 556-572.

[VF90] B. VALLÉE, PH. FLAJOLET: The Lattice Reduction Algorithm of Gauss: An Average Case Analysis. Proc. 31st IEEE Symposium on Foundations of Computer Science, 1990, pp. 830-842.

[Wa68] B.L. VAN DER WAERDEN: Die Reduktionstheorie der positiven quadratischen Formen. In: B.L. VAN DER WAERDEN, H. GROSS: Studien zur Theorie der Quadratischen Formen. Birkhäuser, Basel 1968, pp. 17-44.

[We42] H. Weyl: On geometry of numbers. Proc. London Math. Soc. (2), 47 (1942), pp. 268-289.

[Ya92] C.K. Yap: Fast Unimodular Reduction: Planar Integer Lattices. 33rd IEEE Symposium on Foundations of Computer Science (1992), pp. 437-446.

# Symbolverzeichnis

| | |
|---|---|
| $\|\cdot\|$ | Norm, **10** |
| $< a, b >$ | Skalarprodukt der Vektoren $a, b$ |
| $\subset$ | Teilmenge, Identität nicht ausschließend |
| $(\cdots)$ | Matrix oder geordnetes Paar |
| $[\cdots]$ | verallgemeinerter Kontinuant, **41** |
| $\lceil\cdot\rfloor$ | nächste ganze Zahl, nächster ganzzahliger Vektor, **63** |
| $\lceil x \rceil$ | $\inf\{n \in \mathbb{Z} \,\vert\, n \geq x\}$ |
| $\lfloor x \rfloor$ | $\sup\{n \in \mathbb{Z} \,\vert\, n \leq x\}$ |
| $\alpha_k,\ \alpha_{k,l_2}$ | $\alpha$-Konstanten, **18** |
| $B_{\rho,c}$ | Kugel $\{x \in \mathbb{R}^n \mid \| x - c \| \leq \rho\}$ mit Radius $\rho$ |
| $e_i$ | $i$-ter Einheitsvektor |
| $F_i$ | $i$-te Höhenfunktion zur Norm $\|\cdot\|$ und Basis $b_1, \ldots, b_i$, **11** |
| $\kappa_m,\ \kappa_{m,\|\cdot\|}$ | $\kappa$-Konstanten, **12** |
| $\lambda_i,\ \lambda_i(L)$ | $i$-tes sukzessives Minimum des Gitters $L$, **10** |
| $\log(x)$ | natürlicher Logarithmus |
| $\log_y(x)$ | $\log(x)/\log(y)$ |
| $\mathcal{M}(B)$ | Bitkomplexität der Multiplikation $B$-stelliger Zahlen, **58** |

| | |
|---|---|
| $f(x) = O(g(x))$ | $\exists\, c > 0 \;\; \forall x : \; \lvert f(x)\rvert \leq g(x)$ |
| $f(x) = o(g(x))$ | $f(x)/g(x) \longrightarrow 0$ |
| $\mathbb{R}$ | reelle Zahlen |
| span $M$ | lineare Hülle: der kleinste Vektorraum, der $M$ enthält |
| succ | Nachfolgerfunktion, **31** |
| swo | strikt wohlgeordnet, **59** |
| $\mathbb{Z}$ | (rationale) ganze Zahlen |

# Index


$\alpha$-Konstanten, 18
$\kappa$-Konstanten, 12
$\sigma$–minimal, 59

Abstieg, 64
Algorithmen
  Block–Reduktion, 24
  FG (Bitkomplexität), 64
  Gauß, **32**, 31–49
  LLL, 3
  Reduktionsschritt
    in allgemeiner Norm, 50–51
    in der $l_1$-Norm, 52–54
    in der $l_2$-Norm, 50
    in der $l_\infty$-Norm, 54–56
  Schönhage
    –Paterson–Pippenger (Median), 53
    –Strassen (Multiplikation), 58
    GGT, 6, 57
    quadratische Formen, 6, 57
  WM, *siehe* $l_1$-Norm

Basis, 9
  $\sigma$–minimale, 59
  reduzierte
    block–, **18**, 19–24
    Gauß-, **25**, 25–30
    Hermite–, 17
    KZ–, *siehe* Hermite
    längen–, 18
    Lovász–, 18
    Minkowski–, 10
  strikt wohlgeordnete, 59
  wohlgeordnete, 6, **25**, 32–35, 37–40, 60–62
blockreduziert, 18
Blockweite, 18

Daudé, Hervé, 7, 57
Determinante, 12
Dirichlet, G.L., 3, 10
Dupré, A., 6, 47

Gauß, Carl Friedrich, 3, 10, 25, 31
  –Algorithmus, 31
  –Schritt, *siehe* Reduktionsschritt
  –reduziert, *siehe* Basis
Gitter, 9

Hermite, C., 3, 10, 11
  –Konstante, 12
  –reduziert, 17
Höhenfunktionen, 4, **11**, 11–17
Höhenprodukt, 13

Joux, A., 4

Kontinuanten, 41
Korkine, A., 3, 17, 27
  –Zolotarev–reduziert *siehe* Hermite 17
kürzester Gittervektor, 10
Kugel, 9

längenreduziert, 18
Lagrange, L., 3
Lenstra, H.W., 3, 4, 18
LLL–Algorithmus, *siehe* Lovász
Lovász, Laszlo, 3

–Algorithmus, 3, 10
reduzierte Basis, 18

Median, 53
Minkowski, Hermann, 3, 10
–reduziert, *siehe* Basis
Sätze von, 16

Nachfolger, 33
–basis, 33
–funktion, 32
Norm, 10

Odlyzko, Andrew, 4

primitives System, 9

Radius, 9
Rang, 9
Reduktion, 10
Reduktionskoeffizient, 31
Reduktionsschritt, 31, *siehe auch* Algorithmen
reduziert, *siehe* Basis
Rieger, G.J., 41
Ritter, Harald, 24

Schnorr, Claus Peter, 3, 4, 10, 11, 18, 19, 24, 31
–reduziert, *siehe* blockreduziert
Schritt
arithmetischer, 49
Gauß–, *siehe* Reduktionsschritt
Orakel-, 49
Reduktions–, 31
Schönhage, Arnold, 6, 7, 57
–e.a.–Algorithmen, *siehe* Algorithmen
Stern, Jacques, 4
strikt wohlgeordnet (swo), 59
sukzessive Minima, 10

Vallée, Brigitte, 6, 7, 25, 31, 33, 47, 57
Vorgänger, 33
–basis, 33

wohlgeordnet, *siehe* Basis

Zhang, Jiping, 79
Zolotarev, G., 3, 17, 27

# Lebenslauf

| | |
|---|---|
| Name | Michael Andreas Kaib, |
| geboren am | 13. März 1965 in Frankfurt am Main, |
| verheiratet mit | Jiping Zhang seit dem 10. Februar 1994, |
| Eltern | Friedrich und Ruth Kaib (geb. Pietratus). |
| | |
| 1971 – 1975 | Werner–von–Siemens–Grundschule in Dörnigheim, |
| 1975 – 1981 | Dietrich–Bonhoeffer–Gesamtschule in Maintal, |
| 1981 – 1984 | Albert–Einstein–Gymnasium in Maintal. |
| 10/1984 – 6/1986 | Mathematik–Grundstudium mit Nebenfach Informatik an der J.W. Goethe–Universität Frankfurt, |
| 6/1986 | Diplom–Vorprüfung Mathematik, |
| 7/1986 – 6/1989 | Mathematik–Hauptstudium mit Nebenfach Informatik an der J.W. Goethe–Universität Frankfurt, |
| 6/1989 | Diplom–Hauptprüfung Mathematik,<br>Thema der Diplomarbeit: *Ein neuer Zugang zur Kettenbruchentwickung reell–quadratischer Irrationalzahlen*,<br>Betreuer: Prof. Dr. J. Wolfart. |
| seit 7/1989 | Mathematik–Promotionsstudium an der J.W. Goethe–Universität Frankfurt. |
| | |
| 7/1989 – 6/1994 | Wissenschaftlicher Mitarbeiter am FB Mathematik der J.W. Goethe–Universität Frankfurt in der Arbeitsgruppe 7.2 (Angewandte Mathematik, theoretische Informatik). |
| seit 8/1994 | Senior–Softwareingenieur bei software, design & management. |