



FernUniversität
Gesamthochschule in Hagen

FACHBEREICH MATHEMATIK
LEHRGEBIET NUMERISCHE MATHEMATIK
Prof. Dr. F. Locher

Mathematisches Praktikum 1074
Gitterpunkt-Integrationsformeln

Ulrich Telle
Matrikel-Nr. 1471341

Köln, den 16. Februar 2005

Inhaltsverzeichnis

1	Mathematische Grundlagen	2
1.1	Einleitung	2
1.2	Numerische multivariate Integration	2
1.3	Gitterpunkt-Formeln	4
1.4	Fehlerabschätzung für Gitterpunkt-Formeln	7
1.5	Kriterien für die Güte von Integrationsgittern	8
1.6	„Gute“ Gitterpunkt-Formeln vom Rang 1	10
1.7	Nicht-periodische Integranden	11
2	Programmbeschreibung	13
2.1	Integrationsframework	13
2.2	Implementierte Verfahren	14
2.3	Schätzung des Integrationsfehlers	15
2.4	Package- und Klassenstruktur	16
2.4.1	Klasse Integrator	17
2.4.2	Klasse LatticeRule	17
2.4.3	Klasse Optimizer	17
2.4.4	Klasse Periodizer	18
2.5	Anwendungsbeispiel	18
3	Testergebnisse	20
3.1	Erster Funktionstest	20
3.2	Einfluss von Dimension und Ordnung	22
3.3	Einfluss der Wahl des Generierungsvektors \mathbf{z}	23
3.4	Suche optimaler Gitterpunkte, Einfluss des Parameters α	23
3.5	Spezialfall: Fibonacci-Gitterpunkt-Formeln	24
3.6	Einfluss der Periodisierung	25
3.7	Test mit den Genz-Integranden-Familien	27
4	Zusammenfassung	30
5	Literaturverzeichnis	30

1 Mathematische Grundlagen

1.1 Einleitung

Für die numerische Integration in einer Dimension (univariate Integration) sind schon vor mehreren Jahrhunderten Verfahren wie etwa die Rechteck-, Simpson- oder Gauss-Regel entwickelt worden. Ein kleiner Überblick findet sich in [Tel02]. Multivariate Integrale treten in vielen Anwendungsgebieten wie beispielsweise in der Wahrscheinlichkeitstheorie, in der statistischen Mechanik, in der Quantenphysik oder der Finanzmathematik auf. In der Regel können diese Integrale nicht analytisch gelöst werden, sondern müssen mit Hilfe numerischer Methoden berechnet werden. In der Praxis sind der Berechnung hochdimensionaler Integrale mit Hilfe klassischer Integrationsverfahren oft Grenzen gesetzt, da der Rechenaufwand exponentiell mit der Dimension des Problems ansteigt. Daher spricht man in diesem Zusammenhang auch vom *Fluch der Dimension*. Auch klassische adaptive numerische Integrationsverfahren können den Fluch der Dimension nicht prinzipiell brechen.

Erst nach der Entwicklung der Computer konnten nennenswerte Fortschritte erzielt werden. Es wurden Verfahren entwickelt, deren Aufwand zur Lösung hochdimensionaler Integrationsprobleme, d.h., deren Konvergenzordnung, von der Dimension nahezu unabhängig ist. Allerdings ist bei manchen dieser Verfahren wie etwa bei Monte-Carlo-Methoden die Konvergenzrate gering (kleiner als 1), so dass zur Erreichung hoher Genauigkeiten wiederum sehr hoher Rechenaufwand anfällt.

Einen möglichen Ausweg aus diesem Dilemma zeigen geschickt konstruierte Gitterpunkt-Integrationsformeln, mit denen sich die Mathematiker bereits vor rund 50 Jahren beschäftigt haben. In diesem Beitrag zum Mathematischen Praktikum sollen Gitterpunkt-Integrationsformeln zur numerischen Behandlung mehrdimensionaler Integrale näher untersucht werden. Die Ausarbeitung basiert auf dem Text von Überhuber [Übe01] und stützt sich darüberhinaus wesentlich auf das Buch von Sloan und Joe [SJ94].

1.2 Numerische multivariate Integration

Ohne Beschränkung der Allgemeinheit soll im folgenden von Problemen ausgegangen werden, die als s -dimensionales Integral über dem Einheitswürfel $C^s = [0, 1]^s$ ausgedrückt werden können:

$$I f = \int_{C^s} f(\mathbf{x}) d\mathbf{x} = \int_0^1 \cdots \int_0^1 f(x_1, \dots, x_s) dx_1 \cdots dx_s \quad (1)$$

In der Praxis kann diese Bedingung durch vergleichsweise einfache lineare Transformationen der Integrationsvariablen erfüllt werden.

In der Regel ist der Integrand f stetig bis auf mögliche Singularitäten an isolierten Punkten oder auf niedrig-dimensionalen Flächen. Numerische Integrationsformeln, die auf der Auswertung des Integranden an bestimmten Punkten basieren, sind meist ziemlich ineffektiv in der Behandlung von Singularitäten, die im Inneren des Integrationsgebiets liegen. Daher sollte, wenn möglich, die Lage und Art der Singularitäten vorab bestimmt werden, um sie entweder durch geeignete Transformationen zu beseitigen oder auf den Rand des Integrationsgebiets zu verlegen oder das Integrationsgebiet in geeignete Teilgebiete zu zerlegen.

Für eindimensionale Integrale

$$I f = \int_0^1 f(x) dx$$

gibt es eine Vielzahl von Integrationsformeln der Form

$$Q f = \sum_{j=0}^{N-1} \omega_j f(x_j),$$

die sich durch die Wahl der Stützstellen x_0, \dots, x_{N-1} und der Gewichte $\omega_0, \dots, \omega_{N-1}$ unterscheiden. In ähnlicher Weise kann ein s -dimensionales Integral durch eine Integrationsformel der Form

$$Q f = \sum_{j=0}^{N-1} \omega_j f(\mathbf{x}_j) \quad (2)$$

angenähert werden, wobei $\mathbf{x}_0, \dots, \mathbf{x}_{N-1}$ aus C^s sind. Die Anzahl N wird *Ordnung* der Integrationsregel genannt.

Der naheliegendste Ansatz für die Annäherung des s -dimensionalen Integrals (1) ist, wiederholt für jede Dimension eine eindimensionale Integrationsregel anzuwenden. Daraus ergibt sich die *Produktregel*:

$$Q f = \sum_{j_s=0}^{n-1} \cdots \sum_{j_1=0}^{n-1} \omega_{j_1} \cdots \omega_{j_s} f(x_{j_1}, \dots, x_{j_s}). \quad (3)$$

Produktregeln sind im Prinzip einfach anzuwenden und der Integrationsfehler lässt sich auf der Basis der Fehleranalyse für die zugrundeliegende eindimensionale Integrationsregel recht gut abschätzen. Für hohe Dimensionen s ist die sich ergebende Ordnung $N = n^s$ allerdings nicht mehr mit vertretbarem Aufwand handhabbar. Für eine eindimensionale Regel mit $n = 10$ und eine Dimension $s = 20$ folgt beispielsweise schon $N = 10^{20}$.

Vielfach werden numerische Integrationsregeln derart konstruiert, dass sie bestimmte Klassen von Funktionen exakt integrieren. So integriert z.B. die eindimensionale Gauss-Regel mit n Stützstellen algebraische Polynome vom Grad $2n - 1$ exakt. Entsprechend gibt es eindimensionale Integrationsregeln, die es erlauben, trigonometrische Polynome bestimmten Grades exakt zu integrieren. Durch mehrfache Anwendung entsprechend der Produktregel (3) können dadurch multivariate Integrale berechnet werden. Wie bei allen Produktregeln steigt jedoch der Rechenaufwand exponentiell mit der Dimension s .

Um den Fluch der Dimension zu brechen, wurde die klassische Monte-Carlo-Methode entwickelt [MU49, HH64], in der die Punkte $\mathbf{x}_1, \dots, \mathbf{x}_N$ zufällig aus C^s gewählt werden und dann das Integral (1) durch

$$Q_N f = \frac{1}{N} \sum_{j=1}^N f(\mathbf{x}_j)$$

angenähert wird. Die - von der Dimension unabhängige - Konvergenz ist durch den zentralen Grenzwertsatz mehr oder weniger garantiert und der Integrationsfehler kann durch

$$\frac{\sigma(f)}{\sqrt{N}} \quad (4)$$

abgeschätzt werden, wobei eine Näherung für σ als Wurzel aus der Varianz leicht mit Hilfe der Formel

$$\sigma(f)^2 \approx Q_N f^2 - (Q_N f)^2$$

gewonnen werden kann. Nachteilig bei dieser Methode ist insbesondere, dass die Güte des Verfahrens sehr stark von der Güte der verwendeten Zufallszahlen abhängt. In der Regel ist die Konvergenzrate sehr schlecht: wie Formel (4) zeigt, muss N um den Faktor 100 erhöht werden, um die Genauigkeit um eine Dezimale zu erhöhen. Darüberhinaus machen statistische Schwankungen die Methode ungeeignet für die Untersuchung von Integralen mit kontinuierlich variierenden Parametern.

Bei den Quasi-Monte-Carlo-Methoden, ausführlich untersucht in [Nie92], versucht man, mit Hilfe deterministisch konstruierter Zahlenfolgen die Stützstellen derart zu bestimmen, dass sich eine bessere Konvergenz als bei der klassischen Monte-Carlo-Methode ergibt. Auch die Gitterpunkt-Formeln gehören zur Klasse der Quasi-Monte-Carlo-Methoden.

Schließlich sollen hier noch die adaptiven Methoden erwähnt werden, die in der Regel auf Integrationsformeln zur Abschätzung des Integrals auf Teilregionen des Integrationsgebiets, einer Methode zur Abschätzung des Integrationsfehlers, einem Fehlertoleranzkriterium und einer Verfeinerungsstrategie basieren. Optimierte für bestimmte Problemstellungen entstehen laufend neue Verfahren in dieser Methodenkatgorie, wie etwa die von Gerstner und Griebel [GG98, GG03] entwickelten Dünngitterverfahren belegen.

Zur Veranschaulichung werden für die genannten Methodenkatgorien in Abbildung 1 exemplarische Stützstellenmengen dargestellt.

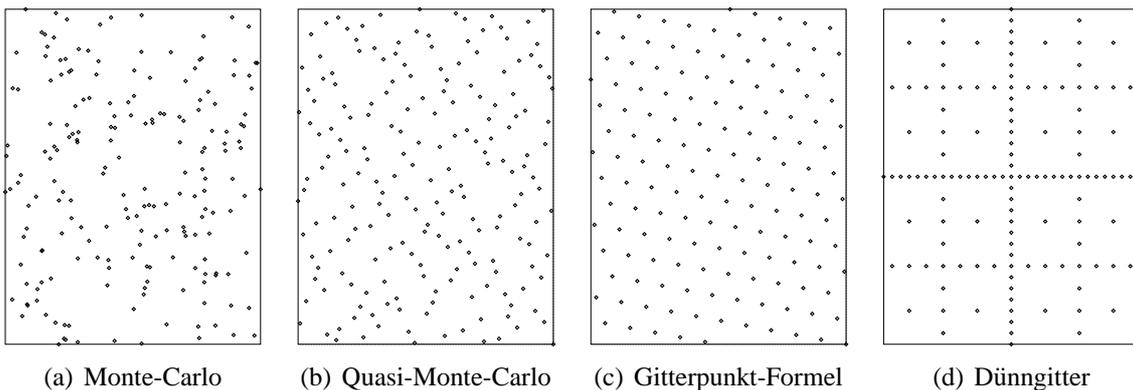


Abbildung 1: Beispiele für Stützstellenmengen verschiedener Integrationsmethoden

1.3 Gitterpunkt-Formeln

Univariate Integrationsformeln können nicht ohne weiteres auf den multivariaten Fall verallgemeinert werden. Es zeigt sich jedoch, dass für die Klasse der in jeder Komponente einperiodischen Funktionen eine Verallgemeinerung der eindimensionalen Rechteckregel konstruiert werden kann. Für den Integranden f muss in diesem Fall

$$f(\mathbf{x}) = f(\mathbf{x} + \mathbf{z}) \quad \forall \mathbf{z} \in \mathbb{Z}^s \text{ und } \forall \mathbf{x} \in \mathbb{R}^s$$

gelten bzw. f muss sich außerhalb des Integrationsgebiets C^s zu einem einperiodischen Integranden fortsetzen lassen.

Bekanntlich lautet die eindimensionale Rechteck-Regel

$$R_n f = \frac{1}{n} \sum_{j=0}^{n-1} f\left(\frac{j}{n}\right)$$

Da der Integrand f als eins-periodisch vorausgesetzt wird, handelt es sich genau genommen sogar um die Trapez-Regel

$$T_n f = \frac{1}{2n} f(0) + \frac{1}{n} \sum_{j=1}^{n-1} f\left(\frac{j}{n}\right) + \frac{1}{2n} f(1)$$

Die Rechteck-Regel lässt sich auf unterschiedliche Weise auf höhere Dimensionen verallgemeinern. Naiv könnte man sie etwa schreiben als

$$R_{n^s} f = \frac{1}{n^s} \sum_{j_s=0}^{n-1} \cdots \sum_{j_1=0}^{n-1} f\left(\frac{j_1}{n}, \dots, \frac{j_s}{n}\right) \quad (5)$$

Allerdings hat diese Formel die Ordnung $N = n^s$ und ist damit für höhere Dimensionen ineffizient.

Eine interessantere Verallgemeinerung ist die *Methode der guten Gitterpunkte*, die von den Zahlentheoretikern Korobov [Kor59] und Hlawka [Hla62] entwickelt wurde. Die Integrationsformel lautet in diesem Fall

$$Qf = \frac{1}{N} \sum_{j=0}^{N-1} f\left(\frac{j}{N} \mathbf{z}\right) \quad (6)$$

wobei N die Anzahl der Stützstellen und \mathbf{z} ein sorgfältig gewählter Vektor mit ganzzahligen Komponenten ist. Es wird dabei vorausgesetzt, dass die Komponenten von \mathbf{z} und N keinen gemeinsamen Teiler > 1 besitzen. Dadurch ist gewährleistet, dass alle N Abszissen verschieden sind. Wegen der vorausgesetzten Periodizität von f spielt es keine Rolle, dass einige Punkte $j\mathbf{z}/N$ außerhalb des Einheitswürfels C^s liegen können, da der ganzzahlige Anteil einfach abgezogen werden kann. Da die Ordnung N nicht von der Dimension s abhängt, ergibt sich eine wesentlich effizientere Formel.

Ganz allgemein versteht man unter einer Gitterpunkt-Formel eine multivariate Integrationsformel (2) mit gleichen Gewichten $\omega_j = \frac{1}{N}$ für $0 \leq j < N$ und \mathbf{x}_j aus einem *Integrationsgitter* L für $0 \leq j < N$:

$$Qf = \frac{1}{N} \sum_{j=0}^{N-1} f(\mathbf{x}_j) \quad (7)$$

Unter einem Integrationsgitter L versteht man eine diskrete Teilmenge des \mathbb{R}^s mit folgenden Eigenschaften:

1. Wenn x_1 und x_2 zu L gehören, dann gehören auch $x_1 + x_2$ und $x_1 - x_2$ zu L ,
2. L enthält s linear unabhängige Punkte,
3. 0 ist ein isolierter Punkt von L , d.h., es gibt eine Umgebung von 0 , deren Durchschnitt mit L nur 0 enthält und
4. es gilt $Z^s \subseteq L$.

Aus diesen Eigenschaften folgt, dass jedes Gitter L eine diskrete additive Untergruppe des \mathbb{R}^s bildet.

Für die Abszissenmenge $A(L) := \{\mathbf{x}_0, \dots, \mathbf{x}_{N-1}\}$ einer Gitterpunkt-Formel gilt $A(L) = L \cap C^s$. Wegen $0 \in L$ ist $A(L) \neq \emptyset$.

Abbildung 2 zeigt ein Integrationsgitter im \mathbb{R}^2 , das man für $N = 18$ und $\mathbf{z} = (1, 5)^T$ aus der Methode der guten Gitterpunkte erhält. Die schwarzen Punkte markieren die Abszissenmenge $A(L)$.

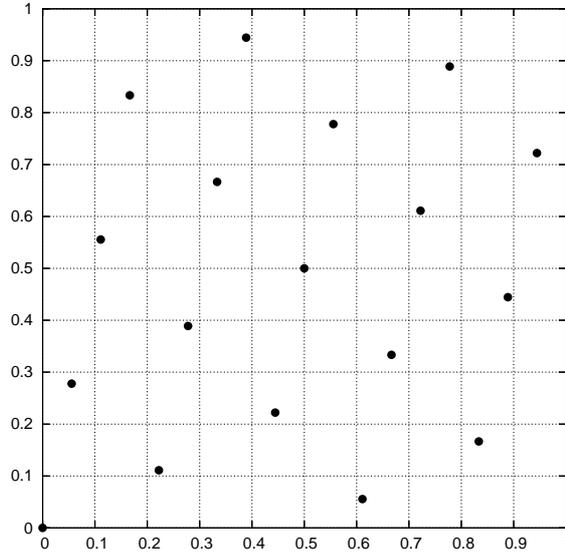


Abbildung 2: Integrationsgitter im \mathbb{R}^2

Für die praktische Anwendung von Gitterpunkt-Formeln in Integrationsprogrammen ist die Darstellung in t -Zyklus-Form nützlich. Jede Gitterpunkt-Formel lässt sich als mehrfache Summe der Form

$$Qf = \frac{1}{n_1, n_2 \dots n_t} \sum_{j_1=0}^{n_1-1} \dots \sum_{j_t=0}^{n_t-1} f\left(\frac{j_1}{n_1} \mathbf{z}_1 + \dots + \frac{j_t}{n_t} \mathbf{z}_t\right) \quad (8)$$

wobei $\mathbf{z}_1, \dots, \mathbf{z}_t \in \mathbb{Z}^s$ ganzzahlige Vektoren sind. Wenn man sich auf die Speicherung der t ganzen Zahlen $n_1, \dots, n_t \in \mathbb{N}$ und der t ganzzahligen Vektoren $\mathbf{z}_1, \dots, \mathbf{z}_t \in \mathbb{Z}^s$ beschränkt, lassen sich Gitterpunkt-Formeln sehr speichereffizient darstellen.

Gitterpunkt-Formeln lassen sich nun danach klassifizieren, welcher minimale Wert von t benötigt wird, um die Gitterpunkt-Formel in der Form (8) darzustellen. Der Wert von t wird *Rang* der Gitterpunkt-Formel genannt. Es ist leicht einzusehen, dass t alle Werte zwischen 1 und s annehmen kann. Die Produkt-Rechteck-Regel (5) ist ein Beispiel für eine Gitterpunkt-Formel mit maximalem Rang s , während die Methode der guten Gitterpunkte (6) offensichtlich den Rang 1 hat.

In der Praxis kann es gelegentlich nützlich sein, die Integrationspunkte einer Gitterpunkt-Formel zu verschieben, z.B., um zu gewährleisten, dass keine Integrationspunkte auf dem Rand des Integrationsgebietes liegen. Für jedes $\mathbf{c} \in \mathbb{R}^s$ erhält man so eine *versetzte Gitterpunkt-Formel*

$$Q_c f = \frac{1}{N} \sum_{j=0}^{N-1} f(\{\mathbf{x}_j + \mathbf{c}\}) \quad (9)$$

Hierbei bedeutet $\{\mathbf{x}\} = \mathbf{x} - \mathbf{z} \in C^s$, wobei \mathbf{z} ein geeigneter ganzzahliger Vektor ist.

1.4 Fehlerabschätzung für Gitterpunkt-Formeln

Um beurteilen zu können, welches Integrationsgitter L , also welche Abszissenmenge $A(L)$, in einer Gitterpunkt-Formel sinnvollerweise verwendet werden sollte, ist es erforderlich, den Fehler der Gitterpunkt-Formel abzuschätzen. Nur so lässt sich feststellen, welches Integrationsgitter besser als ein anderes ist.

Wenn der Integrand f hinreichend glatt ist, kann man ihn in seine Fourier-Reihe entwickeln:

$$f(\mathbf{x}) = \sum_{\mathbf{h} \in \mathbb{Z}^s} \hat{f}(\mathbf{h}) e^{2\pi i \mathbf{h} \cdot \mathbf{x}}$$

wobei $\mathbf{h} \cdot \mathbf{x}$ das Skalarprodukt im \mathbb{R}^s bezeichnet und

$$\hat{f}(\mathbf{h}) = \int_{C^s} e^{-2\pi i \mathbf{h} \cdot \mathbf{x}} f(\mathbf{x}) d\mathbf{x}, \quad \mathbf{h} \in \mathbb{Z}^s$$

Wendet man die Gitterpunkt-Formel (7) Term für Term auf die Fourier-Reihe von f an, so erhält man

$$Qf = \sum_{\mathbf{h} \in \mathbb{Z}^s} \hat{f}(\mathbf{h}) Q e^{2\pi i \mathbf{h} \cdot \mathbf{x}}$$

Es bleibt nun noch zu klären, welchen Effekt Q auf $e^{2\pi i \mathbf{h} \cdot \mathbf{x}}$ für alle $\mathbf{h} \in \mathbb{Z}^s$ hat. Mit Hilfe des zu Q dualen Gitters gibt es darauf eine einfache Antwort. Das duale Gitter L^\perp zu einem Integrationsgitter L ist definiert als die Menge

$$L^\perp = \{\mathbf{h} \in \mathbb{R}^s \mid \mathbf{h} \cdot \mathbf{x} \in \mathbb{Z} \quad \forall \mathbf{x} \in L\}$$

Ein Beispiel für ein duales Gitter wird in Abbildung 3 dargestellt.

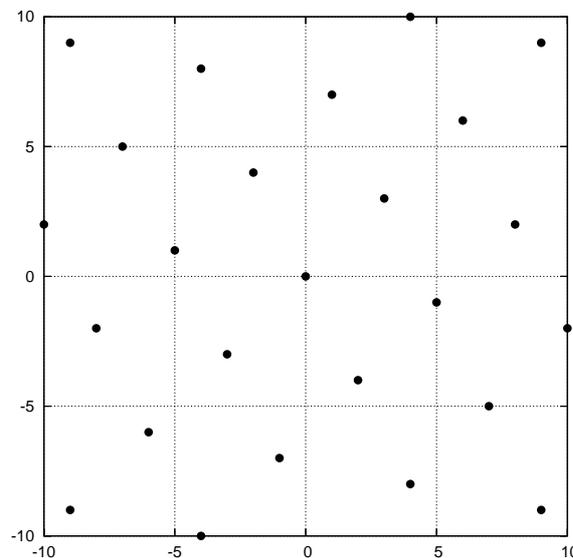


Abbildung 3: Duales Gitter L^\perp zum Integrationsgitter L in Abbildung 2

Es gilt nun folgende Aussage für alle $\mathbf{h} \in \mathbb{Z}^s$:

$$Q e^{2\pi i \mathbf{h} \cdot \mathbf{x}} = \begin{cases} 1 & \text{falls } \mathbf{h} \in L^\perp \\ 0 & \text{sonst} \end{cases}$$

Ein Beweis findet sich u.a. in auf Seite 26 in [SJ94].

Der Integrationsfehler einer Gitterpunkt-Formel (7) lässt sich wegen $If = \hat{f}(\mathbf{0})$ nun recht einfach beschreiben:

$$Qf - If = \sum_{\mathbf{h} \in L^\perp \setminus \mathbf{0}} \hat{f}(\mathbf{h}) \quad (10)$$

Für versetzte Gitterpunkt-Formeln (9) ist der Integrationsfehler

$$Q_c f - If = \sum_{\mathbf{h} \in L^\perp \setminus \mathbf{0}} e^{2\pi i \mathbf{h} \cdot \mathbf{c}} \hat{f}(\mathbf{h})$$

Damit der Verfahrensfehler $Qf - If$ möglichst klein wird, muss Q so konstruiert werden, dass möglichst viele Basisfunktionen $e^{2\pi i \mathbf{h} \cdot \mathbf{x}}$, deren Fourier-Koeffizient $\hat{f}(\mathbf{h})$ „groß“ ist, auf Null abgebildet werden. Während es im eindimensionalen Fall möglich ist, in dieser Hinsicht optimale Integrationsformeln zu bestimmen, stößt man im mehrdimensionalen Fall auf Schwierigkeiten, da das Abklingen mehrdimensionaler Fourier-Koeffizienten ein wesentlich komplizierter zu charakterisierendes Verhalten aufweist. Bis heute ist dieses Problem noch nicht allgemeingültig gelöst worden.

1.5 Kriterien für die Güte von Integrationsgittern

Im folgenden soll für Gitterpunkt-Formeln, die nach der Methode der guten Gitterpunkte konstruiert wurden, die also die Form (6) aufweisen, untersucht werden, auf welche Weise zur Bestimmung der Abszissenmenge $A(L)$ ein Vektor \mathbf{z} , der in gewissem Sinne „gut“ ist, gewählt werden kann. Das Gitter muss so konstruiert werden, dass der Integrationsfehler (10) „möglichst klein“ ist. Damit ein Gitter für eine ganze Klasse von Integranden geeignet ist, müssen die Fourier-Koeffizienten des Integranden ein - zumindest der Größenordnung nach - einheitliches Abklingverhalten zeigen.

Für $c > 0$ und ein festes $\alpha > 1$ sei $E_s^\alpha(c)$ die Klasse der Funktionen f , deren Fourier-Koeffizienten der Ungleichung

$$|\hat{f}(\mathbf{h})| \leq \frac{c}{(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s)^\alpha} \quad (11)$$

genügen, wobei $\bar{h} = \max(1, |h|)$ ist. Die Funktionenräume $E_s^\alpha(c)$ bezeichnet man als *Korobov-Räume*.

Das Abklingverhalten der Fourier-Koeffizienten einer Funktion hängt mit der Glattheit der Funktion zusammen. $E_s^\alpha(c)$ ist also eine Klasse von Funktionen mit einer gewissen Glattheit. Das folgende Ergebnis, das von Zaremba bewiesen wurde, liefert eine hinreichende Bedingung dafür, dass eine Funktion Element von $E_s^\alpha(c)$ ist, wenn $\alpha > 1$ eine ganze Zahl ist:

Falls $\alpha > 1$ eine ganze Zahl ist und f eine eins-periodische Funktion auf \mathbb{R}^s , deren partielle Ableitungen

$$\frac{\partial^{q_1 + \cdots + q_s} f}{\partial x_1^{q_1} \cdots \partial x_s^{q_s}}, \quad 0 \leq q_k \leq \alpha - 1, \quad 1 \leq k \leq s$$

existieren und stetig auf C^s sind, dann existiert ein $c > 0$, so dass $f \in E_s^\alpha(c)$ gilt.

Für Funktionen $f \in E_s^\alpha(c)$ gibt es nun eine einfache Schranke für den Integrationsfehler. Es gilt

$$|Q(\mathbf{z}, N)f - If| \leq c \sum_{\mathbf{h} \in L^\perp \setminus \mathbf{0}} \frac{1}{(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s)^\alpha} \quad (12)$$

Sei nun f_α mit $\alpha > 1$ definiert als

$$f_\alpha(\mathbf{x}) := \sum_{\mathbf{h} \in \mathbb{Z}^s} \frac{1}{(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s)^\alpha} e^{2\pi i \mathbf{h} \cdot \mathbf{x}}$$

dann gilt $f_\alpha \in E_s^\alpha(1)$ und $If_\alpha = 1$. Die Funktion ist die schlimmstmögliche Funktion der Klasse $E_s^\alpha(1)$, denn für diese Funktion tritt in der Ungleichung (12) Gleichheit ein. Nun definiert man

$$P_\alpha(\mathbf{z}, N) = P_\alpha(Q(\mathbf{z}, N)) := \sum_{\mathbf{h} \in L^\perp \setminus \mathbf{0}} \frac{1}{(\bar{h}_1 \bar{h}_2 \cdots \bar{h}_s)^\alpha}$$

Damit kann die Ungleichung (12) für Funktionen $f \in E_s^\alpha(c)$ geschrieben werden als

$$|Q(\mathbf{z}, N)f - If| \leq cP_\alpha(\mathbf{z}, N) \quad (13)$$

wobei Gleichheit für $f = cf_\alpha$ gilt.

$P_\alpha(\mathbf{z}, N)$ kann als Kriterium für die Güte eines Gitters herangezogen werden. In der praktischen Anwendung wird $P_\alpha(\mathbf{z}, N)$ als Integrationsfehler der Funktion f_α berechnet. Nach obigen Ausführungen gilt:

$$P_\alpha(\mathbf{z}, N) = Q(\mathbf{z}, N)f_\alpha - 1,$$

wobei sich die Funktion f_α als Produkt von Funktionen einer einzigen Variablen ausdrücken lässt:

$$f_\alpha(\mathbf{x}) = \prod_{k=1}^s F_\alpha(x_k)$$

mit

$$F_\alpha(x) = \sum_{-\infty}^{\infty} \frac{e^{2\pi i hx}}{\bar{h}^\alpha} = 1 + \sum_{h \neq 0} \frac{e^{2\pi i hx}}{|h|^\alpha}$$

Normalerweise wird α als gerade ganze Zahl gewählt, da in diesem Fall F_α mit Hilfe von Bernoulli-Polynomen ausgedrückt werden kann:

$$F_\alpha(x) = 1 - (-1)^{\frac{\alpha}{2}} \frac{(2\pi)^\alpha B_\alpha(x)}{\alpha!}$$

Für die ersten drei geraden ganzzahligen Werte von α ergibt sich explizit

$$\begin{aligned} F_2(x) &= 1 + 2\pi^2(x^2 - x + \frac{1}{6}) \\ F_4(x) &= 1 + \frac{\pi^4}{45}(1 - 30x^2(1-x)^2) \\ F_6(x) &= 1 + \frac{2\pi^6}{945}(1 - 21x^2 + 105x^4 - 126x^5 + 42x^6) \end{aligned}$$

Es kann gezeigt werden, dass sich $F_\alpha(x)$ für $\alpha \rightarrow \infty$ der Funktion $F_\alpha() \rightarrow 1 + 2 \cos 2\pi x$ annähert. In Abbildung 4 werden die Graphen von $F_2(x), F_4(x), F_6(x)$ und $F_{\alpha=\infty}(x)$ dargestellt. Offensichtlich ist bereits $F_6(x)$ eine sehr gute Näherung für die Grenzfunktion.

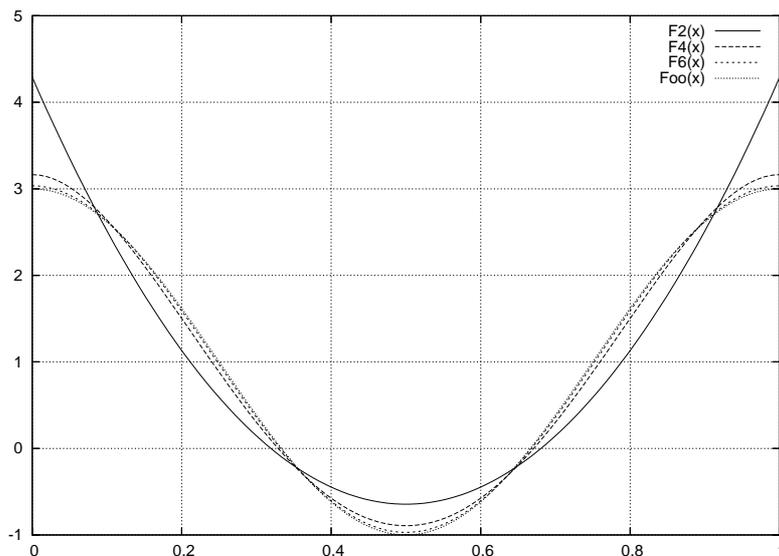


Abbildung 4: $F_2(x)$, $F_4(x)$, $F_6(x)$ und $F_\infty(x)$ auf $[0, 1]$

In Satz 12.4.6 in [Übe01] wird ein weiteres fundamentales, von Niederreiter bewiesenes Konvergenzresultat angegeben:

Zu beliebig vorgegebenen $\alpha > 1$ und Dimension $s \geq 2$ existiert für jede Zahl N eine N -punktige Gitterpunkt-Formel Q_N , so dass

$$P_\alpha(Q_N) \leq d(s, \alpha) \frac{(\log N)^{c(s, \alpha)}}{N^\alpha} \quad (14)$$

gilt, wobei c und d nur von s und α , nicht aber von N abhängen.

Daraus lässt sich ableiten, dass es für jede Dimension s eine Folge von Gitterpunkt-Formeln $\{Q_N\}$ gibt, für die das asymptotische Verhalten des Integrationsfehlers durch

$$|Q_N f - I f| = \mathcal{O}\left(\frac{(\log N)^c}{N^\alpha}\right) \quad \text{für } N \rightarrow \infty$$

Wegen $(\log N)^c = \mathcal{O}(N^\alpha)$ für $N \rightarrow \infty$ ist die Konvergenzordnung der Gitterpunkt-Formeln daher – zumindest theoretisch – nicht vom *Fluch der Dimension* betroffen.

1.6 ‚Gute‘ Gitterpunkt-Formeln vom Rang 1

Für Gitterpunkt-Formeln vom Rang 1 nannte Korobov die Komponenten eines Vektors \mathbf{z} *optimale Koeffizienten*, wenn für Funktionen $f \in E_\alpha^s(c)$ die Ungleichung (13) erfüllt wird, wobei für $P_\alpha(\mathbf{z}, N)$ die Ungleichung (14) gilt. Eine Gitterpunkt-Formel solchen Typs bezeichnet man mit *Methode der guten Gitterpunkte*. Der erste Beweis für die Existenz guter Gitterpunkte wurde von Korobov für den Fall, dass N eine Primzahl ist, geführt.

Für einen festen Wert von $\alpha > 1$ und gegebene Werte von N und s kann man per Computer einen Vektor \mathbf{z} finden, der $P_\alpha(\mathbf{z}, N)$ minimiert. Aufgrund der Form einer Gitterpunkt-Formel genügt es, wenn dabei jede Komponente von \mathbf{z} die Werte von 1 bis N durchläuft. Insgesamt

müssen also höchstens N^s Möglichkeiten für die Wahl von z auf Optimalität überprüft werden. Für große Werte von N und s ist allerdings der Aufwand dafür unvertretbar hoch. Aus diesem Grund wird die Suche nach dem Optimum auf Vektoren \mathbf{z} einer bestimmten Form beschränkt. Die heute gebräuchlichste Form ist die *Korobov-Form*

$$\mathbf{z}(\ell) = (1, \ell, \ell^2 \bmod N, \dots, \ell^{s-1} \bmod N), \quad 1 \leq \ell < N \quad (15)$$

Bei dieser Wahl von \mathbf{z} müssen zur Bestimmung des Optimums nur $N - 1$ verschiedene Möglichkeiten untersucht werden. Unter Berücksichtigung der Symmetrie $F_\alpha(x) = F_\alpha(1 - x)$ für $x \in [0, 1]$ kann der Aufwand sogar noch halbiert werden, indem die Suche des optimalen $\mathbf{z}(\ell)$ auf $1 \leq \ell \leq \lfloor \frac{N}{2} \rfloor$ eingeschränkt wird. In gleicher Weise kann eine vollständige Suche um etwa den Faktor 2^s reduziert werden, indem jede Komponente aus dem Intervall $0 \leq z_k \leq \lfloor \frac{N}{2} \rfloor$ gewählt wird.

Die *Korobov-Form* ist sowohl aus theoretischen als auch aus praktischen Erwägungen eine gute Wahl. Auch wenn die theoretischen Ergebnisse nur für den Fall, dass N eine Primzahl ist, gelten, haben Praxistests gezeigt, dass diese Form auch für beliebige N gut funktioniert.

Da selbst bei Verwendung von Generierungsvektoren der *Korobov-Form* der Aufwand zur Bestimmung optimaler Generierungsvektoren von der Ordnung $\mathcal{O}(N^2)$ ist, ist es recht ineffizient, die Generierungsvektoren stets aufs Neue zu berechnen. Daher wurden im Laufe der Jahre für die unterschiedlichsten Konstellationen Tabellenwerke mit guten Gitterpunkt-Formeln erstellt. So finden sich etwa in [CL01] und [CG03] gute Gitterpunkt-Formeln in 3 bis 6 Dimensionen. Ein Konstruktionsverfahren für Gitterpunkt-Formeln mit Millionen von Punkten findet sich in [DK03]. Auf verschiedenen Wegen wird versucht, die Bestimmung guter Gitterpunkt-Formeln zu parallelisieren, wie etwa durch den Einsatz der Software *PVM* (Parallel Virtual Machine) [LZF94] oder sogar durch verteilte Berechnung im Internet [SM00].

1.7 Nicht-periodische Integranden

In den vorangehenden Ausführungen wurde der Integrand f stets als stetig, möglichst glatt und eins-periodisch in jeder Komponente vorausgesetzt. Im wahren Leben erfüllen die Integrandenfunktionen in den seltensten Fällen alle diese Eigenschaften. Daher ist in der Regel eine vorbereitende Transformation erforderlich, um den Integranden periodisch zu machen.

Zunächst soll der eindimensionale Fall betrachtet werden. Wendet man eine nichtlineare Transformation der Form $x = \phi(t)$ an, wobei ϕ das Intervall $[0, 1]$ auf sich selbst abbildet, so ergibt sich für das Integral einer in $[0, 1]$ stetigen Funktion f :

$$If = \int_0^1 f(x) dx = \int_0^1 f(\phi(t)) \phi'(t) dt$$

Im s -dimensionalen Fall wird die Transformation ϕ einfach auf jede Komponente angewendet. Damit ergibt sich mit $\mathbf{x} = (x_1, \dots, x_s)$ und $\mathbf{t} = (t_1, \dots, t_s)$

$$If = \int_0^1 f(x_1, \dots, x_s) d\mathbf{x} = \int_0^1 f(\phi(t_1), \dots, \phi(t_s)) \phi'(t_1) \cdots \phi'(t_s) d\mathbf{t}$$

Es stellt sich nun die Frage, wie die Transformation ϕ gewählt werden sollte. Fordert man z.B. $\phi'(0) = \phi'(1) = 0$, so verschwindet der Integrand für 0 bzw. 1 in jeder Komponente und hat damit eine stetige periodische Fortsetzung. Das einfachste Polynom mit dieser Eigenschaft für $0 \leq t \leq 1$ ist

$$\phi(t) = 3t^2 - 2t^3 \quad (16)$$

mit der Ableitung

$$\phi'(t) = 6t(1 - t)$$

Möchte man eine glattere periodische Fortsetzung erreichen, so muss man zusätzlich $\phi''(0) = \phi''(1) = 0$ fordern. Das einfachste Polynom mit dieser Eigenschaft für $0 \leq t \leq 1$ ist

$$\phi(t) = t^3(10 - 15t + 6t^2) \quad (17)$$

mit der Ableitung

$$\phi'(t) = 30t^2(1 - t)^2$$

Für hinreichend glatte Funktionen f kann man ohne weiteres Transformationen ϕ höheren Grades finden, bei denen weitere Ableitungen an den Rändern die Bedingung $\phi^{(k)}(0) = \phi^{(k)}(1) = 0$ erfüllen.

Eine weitere Klasse von Transformationen hat statt polynomialen trigonometrischen Charakter. Die einfachste dieser Art hat die Gestalt

$$\phi(t) = t - \frac{1}{2\pi} \sin 2\pi t \quad (18)$$

mit der Ableitung

$$\phi'(t) = 1 - \cos 2\pi t$$

Transformationen dieser Art sind deshalb besonders interessant, weil ihre Ableitungen $\phi'(t)$, die ja einen wesentlichen Teil des Integranden ausmachen, trigonometrische Polynome sind, die von Gitterpunkt-Formeln besonders gut integriert werden.

2 Programmbeschreibung

Zur Untersuchung von Gitterpunkt-Integrationsformeln für die numerische Integration multivariater Integrale wurden Module in der Programmiersprache *Java* unter Verwendung des *Sun Java Developer Kits Version 1.4.2* entwickelt. Besonderer Wert wurde dabei darauf gelegt, dass einerseits die numerischen Eigenschaften analysiert werden konnten, andererseits das Programmpaket aber auch für den praktischen Einsatz einfach anzuwenden ist. In den nachfolgenden Abschnitten werden die allgemeinen Eigenschaften des Integrationsframeworks sowie die implementierten Verfahren und die Struktur der Java-Packages und -Klassen beschrieben.

2.1 Integrationsframework

Für die numerische Integration ein- oder mehrdimensionaler Integranden wird ein Rahmen bereitgestellt, in den unterschiedlichste Integrationsverfahren eingebunden werden können. Es werden einheitliche Schnittstellen für die Steuerung der Integrationsverfahren sowie die Bereitstellung von Integrandenfunktionen definiert. Ein Integrationsverfahren berechnet dabei sukzessive Näherungswerte für das gesuchte Integral. Als Basisfunktionalität wird die Vorgabe und Überwachung verschiedener Terminierungskriterien bereitgestellt. Folgende Kriterien sind derzeit verfügbar:

Absolute Toleranz Unterschreitet die absolute Differenz zweier aufeinanderfolgender Integralwerte die vorgegebene absolute Toleranz, wird das Verfahren beendet und der zuletzt berechnete Integralwert als Ergebnis bereitgestellt. Dieses Kriterium ist standardmäßig deaktiviert. Durch Setzen eines Wertes > 0 wird das Kriterium aktiviert.

Relative Toleranz Sobald die relative Abweichung des zuletzt berechneten Integralwerts vom vorherigen geringer als die vorgegebene relative Toleranz ist, wird das Verfahren beendet und der zuletzt berechnete Integralwert als Ergebnis bereitgestellt. Dieses Kriterium ist standardmäßig aktiviert, wobei die relative Toleranz mit 10^{-6} vorgegeben ist. Durch Setzen eines Wertes ≤ 0 kann das Kriterium deaktiviert werden.

Anzahl Integrandenauswertungen Mit Hilfe dieses Kriteriums kann die Gesamtanzahl der Integrandenauswertungen begrenzt werden. Das Verfahren wird beendet, sobald die vorgegebene Anzahl überschritten wurde. Standardmäßig ist dieses Kriterium deaktiviert. Durch Setzen eines Wertes > 0 wird das Kriterium aktiviert.

Rechenzeit Mit Hilfe dieses Kriteriums kann die Rechenzeit des Verfahrens begrenzt werden. Das Verfahren wird beendet, sobald die Rechenzeit die vorgegebene Zeitdauer überschritten hat.

Mindestens eine der Terminierungsbedingungen muss spezifiziert werden, aber es ist auch möglich, mehrere vorzugeben. Das Verfahren wird beendet, sobald mindestens eines der Kriterien erfüllt wird.

Als Ergebnis liefert ein Integrationsverfahren die Näherung des Integralwertes, eine Abschätzung des Integrationsfehlers (soweit möglich), die Anzahl benötigter Integrandenauswertungen, die benötigte Rechenzeit sowie eine Information darüber, welches Terminierungskriterium zur Beendigung des Verfahrens geführt hat.

2.2 Implementierte Verfahren

Zwar lag der Schwerpunkt der vorliegenden Arbeit auf Gitterpunkt-Formeln nach der *Methode der guten Gitterpunkte*, jedoch wurden zur besseren Einschätzung der Ergebnisse zwei weitere Verfahren implementiert. Folgende Verfahren stehen damit im erstellten Programmpaket zur Verfügung:

Methode der guten Gitterpunkte Mit Hilfe eines vorgegebenen Gittergenerierungsvektors \mathbf{z} wird eine Näherung des Integrals nach der Formel

$$Qf = \frac{1}{N} \sum_{j=0}^{N-1} f\left(\frac{j}{N}\mathbf{z}\right)$$

berechnet. Das Verfahren kann mit einer Periodisierungsfunktion parametrisiert werden, die auf den Integranden angewendet wird. Für nicht-periodische Integranden lässt sich dadurch vielfach ein besseres Integrationsergebnis erzielen.

Eingebettete Gitterpunkt-Formel Unter einer eingebetteten Gitterpunkt-Formel versteht man eine Folge von Gitterpunkt-Formeln der Form

$$Q_r f = \frac{1}{n^r m} \sum_{k_r=0}^{n-1} \cdots \sum_{k_1=0}^{n-1} \sum_{j=0}^{m-1} f\left(\frac{j}{m}\mathbf{z} + \frac{(k_1, \dots, k_r, 0, \dots, 0)}{n}\right)$$

für $0 \leq r \leq s$ wobei n und m relativ prim sind und \mathbf{z} ein ganzzahliger Vektor, dessen Komponenten relativ prim zu m sind. Dabei ist Q_0 eine Gitterpunkt-Formel vom Rang 1 mit m Punkten und Q_r ist für $1 \leq r \leq s$ eine Gitterpunkt-Formel vom Rang r mit $n^r m$ Punkten. Das Besondere an dieser Form ist, dass jede Gitterpunkt-Formel Q_r die Punkte der vorangegangenen beibehält und eine gleiche Anzahl neuer Punkte hinzufügt. Einer der Vorteile dieser Form ist, dass nur für Q_0 ein optimaler Vektor \mathbf{z} bestimmt werden muss. Da Q_0 nur m Punkte hat, ist der Rechenaufwand gegenüber einer Formel Q_s mit $n^s m$ Punkten deutlich geringer. Ein weiterer Vorteil ist, dass der Integrationsfehler gut abgeschätzt werden kann. Implementiert wurde die in [SJ94] vorgeschlagene Version mit $n = 2$.

Monte-Carlo-Methode Bei dieser Methode wird eine Näherung des Integralwerts bestimmt, indem der Mittelwert des Integrals für N Zufallsvektoren \mathbf{x} gebildet wird:

$$Qf = \frac{1}{N} \sum_{j=0}^{N-1} f(\mathbf{x}_j)$$

Alle Verfahren berechnen das Integral der Integrandenfunktion über dem s -dimensionalen Einheitswürfel C^s .

Für die *Methode der guten Gitterpunkte* und die *eingebettete Gitterpunkt-Formel* sollte der Generierungsvektor \mathbf{z} möglichst optimal gewählt werden. In den Auswertungsmodulen wird dies nicht erzwungen, um die Wirkung ungünstiger Generierungsvektoren untersuchen zu können. Im Integrationsframework wird jedoch stets eine Optimierung nach der Korobov-Form vorschaltet, wobei der Parameter α aus der Menge $\{2, 4, 6\}$ gewählt werden kann.

2.3 Schätzung des Integrationsfehlers

In Abschnitt 1.4 auf Seite 7 wurden einige theoretische Betrachtungen über die bei Gitterpunkt-Formeln auftretenden Integrationsfehler gemacht. Allerdings taugen die dort erzielten Ergebnisse kaum für eine Fehlerabschätzung in der Praxis. Daher soll hier für die implementierten Verfahren kurz skizziert werden, wie der Integrationsfehler in diesen Fällen realistisch abgeschätzt werden kann.

Im allgemeinen ist es für Gitterpunkt-Formeln vom Rang 1 nicht möglich, brauchbare Schätzungen für den Integrationsfehler zu erhalten, indem man die Berechnung mit einer wachsenden Zahl von Gitterpunkten wiederholt, da der wahre Fehler in hohem Maße schwanken kann. Der Grund dafür ist, dass bei jedem Berechnungsschritt andere Gitterpunkte gewählt werden, die mit denen, die im vorangegangenen Schritt verwendet wurden, nichts gemein haben. Eine mögliche Strategie, doch zu einer vernünftigen Fehlerabschätzung zu gelangen, basiert auf der Anwendung der versetzten Gitterpunkt-Formel (9), wobei als Verschiebevektor \mathbf{c} Zufallsvektoren benutzt werden.

Für eine beliebige natürliche Zahl q und unabhängigen Zufallsvektoren c_1, \dots, c_q , die auf C^s gleichverteilt sind, lässt sich ein Schätzwert für das Integral berechnen als

$$\bar{Q}(\mathbf{z}, N)f = \frac{1}{q} \sum_{k=1}^q Q(\mathbf{z}, N, \mathbf{c}_k)f$$

Und damit kann auch eine Schätzung des Integrationsfehlers berechnet werden, indem man eine Näherung für die Standardabweichung bestimmt als

$$\sigma = \sqrt{\frac{\sum_{k=1}^q (Q(\mathbf{z}, N, \mathbf{c}_k)f - \bar{Q}(\mathbf{z}, N)f)^2}{q(q-1)}}$$

Für die Wahl von q werden Werte zwischen 3 und 5 empfohlen. Der Nachteil dieser Methode besteht darin, dass der Aufwand das q -fache der normalen Auswertung der Gitterpunkt-Formel beträgt. Für das implementierte Verfahren nach der *Methode der guten Gitterpunkte* kann die Berechnung der Fehlerschätzung optional aktiviert werden.

Der Algorithmus zur Methode der *eingebetteten Gitterpunkt-Formel* enthält eine Berechnung der Standardabweichung nach der Formel

$$\sigma = \sqrt{\frac{\sum_{i=1}^s (Q_s f - Q^{(i)} f)^2}{s}}$$

wobei $Q_s f$ der abschließende Näherungswert für das Integral und $Q^{(i)} f$ die genäherten Integralwerte für die Folge der eingebetteten Gitterpunkt-Formeln sind. Die Berechnung benötigt nur minimalen zusätzlichen Aufwand.

Bei der Monte-Carlo-Methode wird nach üblichen statistischen Methoden die Standardabweichung

$$\sigma = \sqrt{\frac{\sum_{j=0}^{N-1} (Qf(\mathbf{x}_j) - \bar{Q}f)^2}{N(N-1)}}$$

berechnet, wobei $\bar{Q}f$ den Mittelwert der Näherungswerte darstellt.

2.4 Package- und Klassenstruktur

Das Integrationsframework wurde als Java-Package realisiert. Die Package-Struktur ist in Abbildung 5 dargestellt.

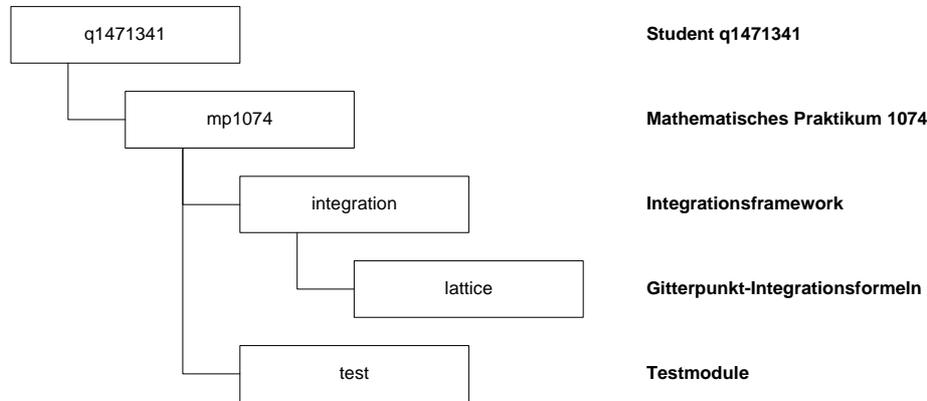


Abbildung 5: Package-Struktur

Das Package `q1471341.mp1074.integration` enthält alle derzeit vorhandenen Klassen zur numerischen Berechnung s -dimensionaler Integrale mit $s \geq 1$. Neben den bereits implementierten Verfahren *Gitterpunkt-Formel-Integration* und *Monte-Carlo-Integration* können hier zukünftig weitere Integrationsverfahren bereitgestellt werden, die die vorhandene Basisfunktionalität nutzen können. Im Unter-Package `lattice` sind diejenigen Klassen angesiedelt, die zur Auswertung von Gitterpunkt-Formeln benötigt werden.

Das Package `q1471341.mp1074.test` enthält einen kleinen Testrahmen, mit dem die Anwendung der Integrationsmodule demonstriert wird bzw. der zur Analyse der numerischen Eigenschaften dient.

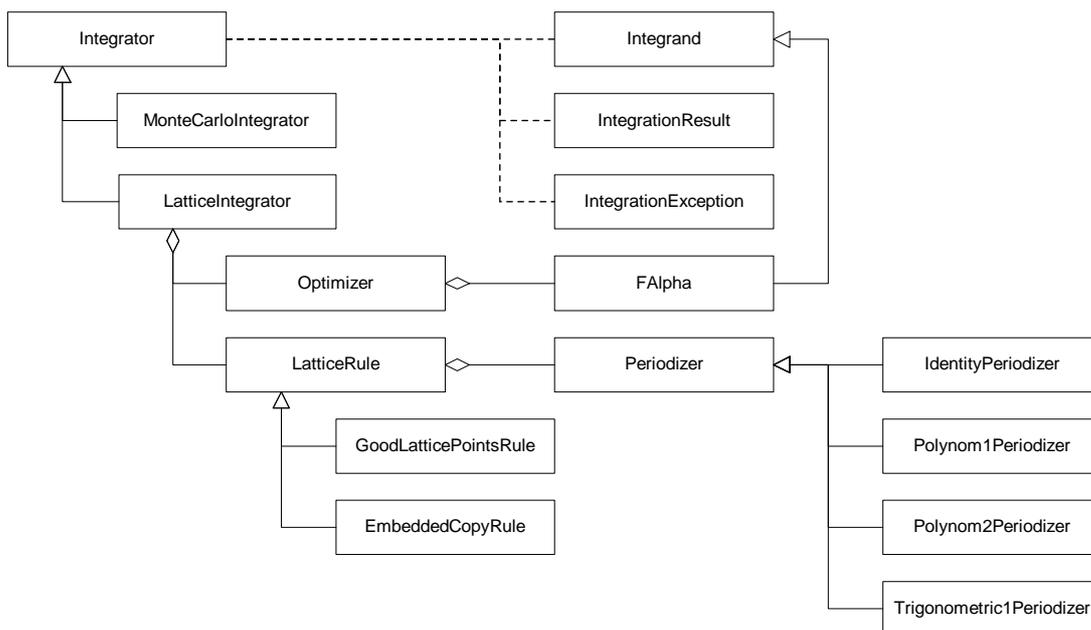


Abbildung 6: Klassendiagramm

Die Klassenstruktur wird in Abbildung 6 auf der vorherigen Seite detaillierter dargestellt. Linien mit Dreiecksspitzen symbolisieren dabei Vererbungsbeziehungen, Linien mit Rautenspitzen stehen für Aggregationsbeziehungen während gestrichelte Linien Assoziationsbeziehungen ausdrücken.

In den nachfolgenden Abschnitten werden die wichtigsten Klassen kurz dargestellt. Eine ausführliche Beschreibung der Schnittstellen findet sich in der JavaDoc-Dokumentation.

2.4.1 Klasse `Integrator`

Die Klasse `Integrator` ist die abstrakte Basisklasse des Integrationsframework, die insbesondere Methoden für die Vorgabe und Überwachung verschiedener Terminierungskriterien bereitstellt. Integrandenfunktionen müssen von der abstrakten Klasse `Integrand` abgeleitet werden. Das Ergebnis einer Integration wird als Instanz der Klasse `IntegrationResult` geliefert, damit neben dem Näherungswert für das Integral weitere Informationen bereitgestellt werden können. Falls die Integration nicht durchgeführt werden kann, weil beispielsweise kein Terminierungskriterium spezifiziert wurde, so wird eine `IntegrationException` geworfen.

Im Rahmen dieser Ausarbeitung wurden folgende beiden von der Klasse `Integrator` abgeleiteten Integrationsverfahren implementiert:

Klasse `LatticeIntegrator` steuert Integrationsverfahren, die auf Gitterpunkt-Formeln basieren. Die zu verwendende Gitterpunkt-Formel wird im Konstruktor spezifiziert.

Klasse `MonteCarloIntegrator` implementiert ein einfaches Monte-Carlo-Integrationsverfahren.

2.4.2 Klasse `LatticeRule`

Die Klasse `LatticeRule` stellt die abstrakte Basisklasse aller Gitterpunkt-Formeln dar. Insbesondere werden Schnittstellenmethoden für die Parametrisierung mit einem Periodisierer sowie für die Abfrage des geschätzten Integrationsfehlers sowie der benötigten Anzahl von Integrandenauswertungen bereitgestellt.

Folgende konkrete Implementierungen stehen zur Verfügung.

Klasse `GoodLatticePointsRule` realisiert die *Methode der guten Gitterpunkte*.

Klasse `EmbeddedCopyRule` realisiert die *Methode der eingebetteten Gitterpunkt-Formel*.

2.4.3 Klasse `Optimizer`

Für die Optimierung der Generierungsvektoren \mathbf{z} für Gitterpunkt-Formeln steht die Klasse `Optimizer` zur Verfügung. Sie dient dazu, denjenigen Generierungsvektor der Korobov-Form zu bestimmen, der die Funktion $P_\alpha(\mathbf{z}, N)$ (siehe Abschnitt 1.5 auf Seite 8) minimiert. Die zur Berechnung von $P_\alpha(\mathbf{z}, N)$ benötigte Funktion f_α wird durch die Klasse `FAlpha`, die von der Klasse `Integrand` abgeleitet ist, bereitgestellt. Für α können dabei Werte aus $\{2, 4, 6\}$ gewählt werden.

2.4.4 Klasse `Periodizer`

Die Klasse `Periodizer` definiert ein Interface, das jede Klasse für die Periodisierung eines nicht-periodischen Integranden implementieren muss. Jede Gitterpunkt-Formel lässt sich mit einem `Periodizer` parametrisieren.

Folgende konkrete Implementierungen stehen derzeit zur Verfügung:

Klasse `IdentityPeriodizer` realisiert die identische Abbildung

$$\phi(t) = t$$

und lässt damit die Integrandenfunktion ungeändert. Mit Hilfe dieser Implementierung kann eine Sonderbehandlung des Falles vermieden werden, wenn die Integrandenfunktion bereits periodisch ist.

Klasse `Polynom1Periodizer` realisiert die polynomische Periodisierungsfunktion

$$\phi(t) = 3t^2 - 2t^3$$

Klasse `Polynom2Periodizer` realisiert die polynomische Periodisierungsfunktion

$$\phi(t) = t^3(10 - 15t + 6t^2)$$

Klasse `Trigonometric1Periodizer` realisiert die trigonometrische Periodisierungsfunktion

$$\phi(t) = t - \frac{1}{2\pi} \sin 2\pi t$$

2.5 Anwendungsbeispiel

Im nachfolgenden Programmausschnitt wird gezeigt, wie das Integrationsframework angewendet werden kann. Zunächst ist der gewünschte Integrand von der Klasse `Integrand` abzuleiten. An der Stelle, an der das Integrationsresultat benötigt wird, ist der Integrand und der Integrator zu instanzieren. Im Falle der Verwendung einer Gitterpunkt-Formel ist diese ggf. in geeigneter Weise zu initialisieren. Bei Bedarf kann der Integrator parametrisiert werden.

Weitere Einzelheiten können der JavaDoc-Dokumentation entnommen werden.

Listing 1: Anwendungsbeispiel

```

import q1471341.mp1074.integration.Integrand;
import q1471341.mp1074.integration.IntegrationResult;
import q1471341.mp1074.integration.LatticeIntegrator;
import q1471341.mp1074.integration.lattice.GoodLatticePointsRule;
import q1471341.mp1074.integration.lattice.Optimizer;
import q1471341.mp1074.integration.lattice.Trigonometric1Periodizer;

public class MyTestClass {

    public class MyIntegrand extends Integrand {
        private static int s;
        public MyIntegrand(int dim) {
            s = dim;
        }
        public int dimension() {
            return s;
        }
        public double evaluate(double[] x) {
            checkArgument(x);
            double result = /* ... */;
            return result;
        }
        public String toString() {
            return "Function_description";
        }
    }

    public void myMethod() {
        // ...
        int s = /* ... */;
        Integrand myIntegrand = new MyIntegrand(s);
        Optimizer opt = new Optimizer();
        LatticeRule rule =
            new GoodLatticePointsRule(new Trigonometric1Periodizer());
        Integrator integrator = new LatticeIntegrator(rule, opt);
        IntegrationResult result = integrator.integrate(myIntegrand);
        double value = result.getValue();
        double error = result.getEstimatedError();
        // ...
    }
}

```

3 Testergebnisse

In den nachfolgenden Abschnitten sollen die Ergebnisse zu den Untersuchungen der verschiedenen Einflussgrößen dargestellt werden.

- Überprüfung der korrekten Implementierung der Algorithmen
- Einfluss der Dimension des Integranden und der Ordnung der Gitterpunkt-Formel
- Einfluss der Wahl des Generierungsvektors \mathbf{z}
- Suche optimaler Gitterpunkte, Einfluss des Parameters α
- Spezialfall: Fibonacci-Gitterpunkt-Formeln
- Einfluss der Periodisierung bei nicht-periodischen Integranden
- Test mit den Genz-Integranden-Familien

3.1 Erster Funktionstest

Zur Überprüfung der korrekten Implementierung wurde das 6-dimensionale Integral mit dem exakten Wert 1 über die Integrandenfunktion

$$f_2(\mathbf{x}) = \prod_{i=1}^6 F_2(x_i)$$

wie in [JS93] mit der Methode der eingebetteten Gitterpunkt-Formel approximiert. Die Ergebnisse sind in Tabelle 1 dargestellt. Die Resultate aus [JS93] konnten qualitativ reproduziert werden, wobei generell kleinere Fehler zu verzeichnen waren. Dadurch, dass einerseits alle Berechnungen mit `double`-Variablen durchgeführt wurden und andererseits die Optimierung Symmetrieeigenschaften ausnutzt, stimmen die Generierungsvektoren nicht in allen Fällen überein. In der ersten Spalte der Tabelle 1 ist die Anzahl m der Gitterpunkte angegeben, für die der Generierungsvektor \mathbf{z} optimiert wurde; die zweite Spalte entspricht der Ordnung des Verfahrens – also der Anzahl Integrandenauswertungen –; die dritte Spalte enthält den wahren Fehler und die vierte Spalte den geschätzten Fehler; in der fünften Spalte ist die benötigte Rechenzeit (inklusive der Optimierung des Generierungsvektors) in Sekunden angegeben; die sechste und letzte Spalte zeigt den verwendeten Generierungsvektor.

Als nächstes wurde für dasselbe Integral die Methode der guten Gitterpunkte getestet, wobei die Fehlerschätzung mittels Randomisierung aktiviert war. Das Ergebnis ist in Tabelle 2 dargestellt. Die Bedeutung der Spalten ist – bis auf die zweite Spalte – identisch mit der in Tabelle 1. Die Ordnung m wurde so gewählt, dass sich in jedem Fall die etwa gleiche Anzahl von Integrandenauswertungen (Spalte 2) ergab wie für die Methode der eingebetteten Gitterpunkt-Formel. Die wesentlich längeren Rechenzeiten werden in erster Linie durch die Optimierung des Generierungsvektors verursacht. Während der tatsächliche Integrationsfehler mit den in Tabelle 1 bei etwa gleicher Ordnung auftretenden durchaus vergleichbar ist, fällt auf, dass die Vorhersage des tatsächlichen Fehler häufig zu niedrig ausfällt. Um vertrauenswürdigeren Schätzungen zu erhalten, wurde daher in späteren Tests die Fehlerschätzung mit dem Faktor 2 versehen.

m	$2^6 m$	Wahrer Fehler	Geschätzter Fehler	Rechenzeit [s]	\mathbf{Z}
79	5056	0,004824	0,010419	0,000	(1,27,18,12,8,58)
157	10048	0,001968	0,004366	0,016	(1,18,10,23,100,73)
313	20032	0,000590	0,001588	0,031	(1,80,140,245,194,183)
619	39616	0,000269	0,000742	0,125	(1,102,500,242,543,295)
1249	79936	0,000094	0,000287	0,468	(1,364,102,907,412,88)
2503	160192	0,000033	0,000094	1,750	(1,253,1434,2370,1393,2009)
5003	320192	0,000011	0,000029	6,828	(1,162,1229,3981,4538,4718)

Tabelle 1: Eingebettete Gitterpunkt-Formel mit Fehlerschätzung

m	$5 \cdot m$	Wahrer Fehler	Geschätzter Fehler	Rechenzeit [s]	\mathbf{Z}
1011	5055	0,002243	0,002165	0,344	(1,504,255,123,321,24)
2009	10045	0,000743	0,000521	1,172	(1,351,652,1835,1205,1065)
4006	20030	0,000252	0,000203	4,594	(1,1236,1410,150,1124,3188)
7923	39615	0,000233	0,000155	18,125	(1,2373,5799,6699,3189,1032)
15987	79935	0,000025	0,000011	73,360	(1,3526,10777,14590,14161,4285)

Tabelle 2: Methode der guten Gitterpunkte mit Fehlerschätzung

In Tabelle 3 sind die Ergebnisse aufgelistet, die ebenfalls mittels der Methode der guten Gitterpunkte – jedoch ohne Fehlerschätzung – erzielt wurden. Es zeigt sich, dass der tatsächliche Fehler gegenüber der Variante mit Fehlerschätzung deutlich größer ist.

Schließlich wurde das Integral noch mittels einer Monte-Carlo-Methode approximiert. Die dabei erreichten Ergebnisse sind in Tabelle 4 aufgeführt. Gegenüber den anderen Integrationsmethoden fällt die deutlich geringere Konvergenzrate in Form entsprechend größerer tatsächlicher Fehler auf. Nach den Betrachtungen in Abschnitt 1.2 ist dies allerdings wenig verwunderlich. Der Integrationsfehler wird darüberhinaus häufig unterschätzt.

Insgesamt belegt dieser Test, dass die Verfahren funktionsfähig implementiert wurden.

m	Wahrer Fehler	Rechenzeit [s]	\mathbf{Z}
1011	0,037826	0,296	(1,504,255,123,321,24)
2009	0,009689	1,156	(1,351,652,1835,1205,1065)
4006	0,003028	4,609	(1,1236,1410,150,1124,3188)
7923	0,002314	18,031	(1,2373,5799,6699,3189,1032)
15987	0,000305	73,875	(1,3526,10777,14590,14161,4285)

Tabelle 3: Methode der guten Gitterpunkte ohne Fehlerschätzung

m	Wahrer Fehler	Geschätzter Fehler	Rechenzeit [s]
5056	0,003351	0,014976	0,015
10048	0,016612	0,010996	0,016
20032	0,012778	0,007694	0,063
39616	0,012070	0,005229	0,093
79936	0,000772	0,003765	0,172
160192	0,004873	0,002638	0,360
320192	0,001109	0,001894	0,703

Tabelle 4: Monte-Carlo-Methode mit Fehlerschätzung

3.2 Einfluss von Dimension und Ordnung

Zur Untersuchung des Einflusses der Dimension des Integranden und der Ordnung der Gitterpunkt-Formel in Hinsicht auf den Integrationsfehler, wurde die Funktion f_α für verschiedene Dimensionen mit Gitterpunkt-Formeln unterschiedlicher Ordnung integriert.

In Abbildung 7 ist für Integranden der Dimensionen 2 bis 12 der Zusammenhang zwischen der Genauigkeit und der Ordnung dargestellt. Mit kleinen Kreisen sind die Stellen markiert, für die konkrete Rechnungen durchgeführt wurden.

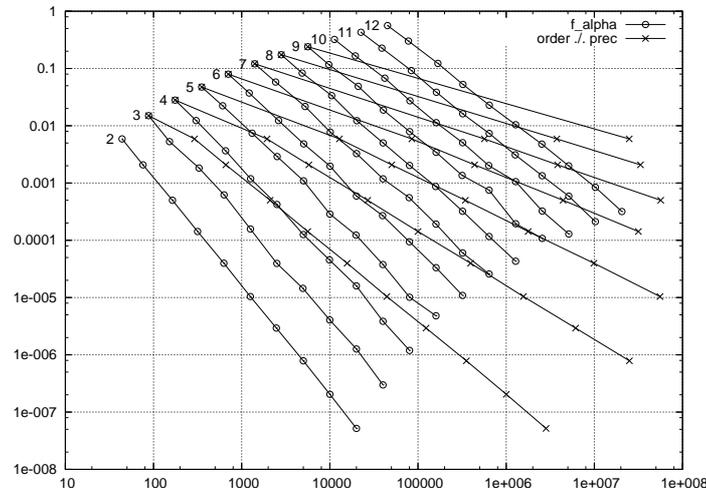


Abbildung 7: Genauigkeit abhängig von der Ordnung für verschiedene Dimensionen

Zum Vergleich sind für jede Dimension Linien mit Kreuzen eingezeichnet, die für eine vorgegebene Genauigkeit die erforderliche Ordnung markieren, wenn die Ordnung n exponentiell mit der Dimension – also n^s – ansteigen würde. Es wird deutlich, dass die Gitterpunkt-Formeln für höhere Dimensionen zwar auch eine stark anwachsende Ordnung für eine gleichbleibende Genauigkeit des Ergebnisses benötigen, aber der *Fluch der Dimension* schlägt zumindest nicht mit voller Macht zu.

Da die Funktion f_α in ihrer Klasse ja den *schlimmsten Fall* repräsentiert, ist davon auszugehen, dass dieses Verhalten erst recht für andere Integranden $f \in E_\alpha^s(c)$ zutrifft.

3.3 Einfluss der Wahl des Generierungsvektors \mathbf{z}

Um den Einfluss der Wahl des Generierungsvektors auf den Integrationsfehler zu überprüfen, wurde die zweidimensionale Funktion $f(x, y) = \frac{ye^{xy}}{e^{-2}}$, für die $If = 1$ gilt, mit einer Gitterpunkt-Formel der Ordnung $N = 53$ integriert, wobei als Generierungsvektor in Korobov-Form $\mathbf{z} = (1, \ell)$ mit $\ell \in \{1, 2, \dots, 52\}$ verwendet wurde. Da es sich bei dem Integranden f um eine nicht-periodische Funktion handelt, wurde die implementierte trigonometrische Periodisierung eingesetzt.

In Abbildung 8(a) werden die bestimmten Näherungen sowohl für den oben genannten Integranden f als auch für den Integranden f_α (mit $\alpha = 2$) in Abhängigkeit von ℓ dargestellt. Der Wert $\ell = 23$, für den der Integrationsfehler für f_α minimal wird, ist mit einem Pfeil markiert.

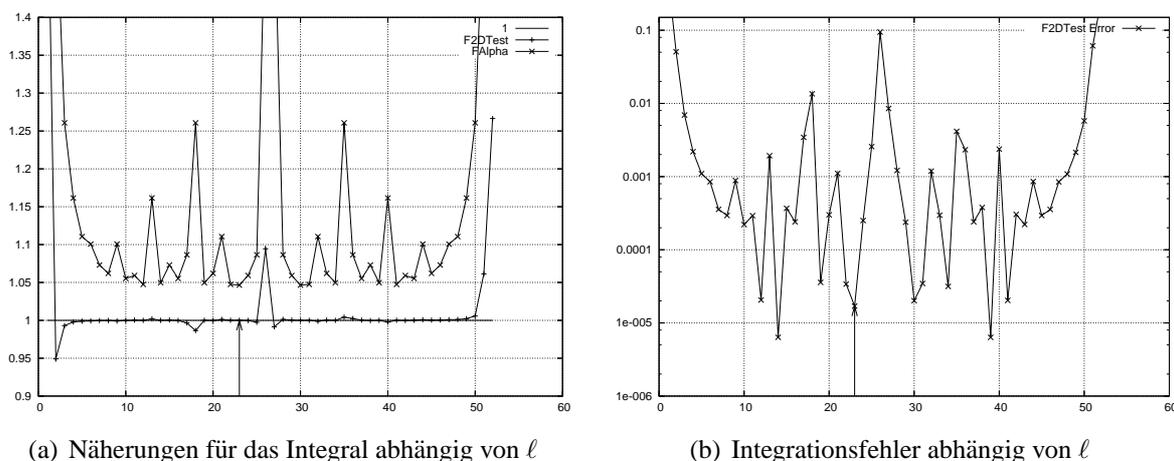


Abbildung 8: Einfluss der Wahl des Generierungsvektors $\mathbf{z} = (1, \ell)$

Abbildung 8(b) zeigt den tatsächlichen Integrationsfehler abhängig von ℓ . Für den ‚optimalen‘ Generierungsvektor $\mathbf{z} = (1, 23)$ ergibt sich im konkreten Fall zwar nicht das absolute Fehlerminimum, aber die Abweichung ist mit weniger als einer Dezimalen äußerst gering.

Die Einschränkung auf Generierungsvektoren in Korobov-Form liefert also nicht zwingend den bestmöglichen Generierungsvektor, stellt aber hinsichtlich des Rechenaufwands zur Bestimmung guter Generierungsvektoren einen sinnvollen Kompromiss dar.

3.4 Suche optimaler Gitterpunkte, Einfluss des Parameters α

Um für eine Gitterpunkt-Formel der Ordnung N zur Integration einer s -dimensionalen Funktion den optimalen Generierungsvektor \mathbf{z} zu bestimmen, muss das Minimum von $P_\alpha(\mathbf{z}, N)$ bestimmt werden. Um den Rechenaufwand in Grenzen zu halten, wurde im Testprogramm die Ordnung $N = 2503$ gewählt. Für die Dimensionen $s \in \{5, 10, 15, 20, 25, 50, 75\}$ werden optimale Generierungsvektoren in Korobov-Form erzeugt, wobei der Parameter α über die implementierten Werte $\{2, 4, 6\}$ variiert.

In Tabelle 5 zeigt sich, dass der Einfluss von α gering zu sein scheint. Daher wird in der Praxis in der Regel mit $\alpha = 2$ gearbeitet.

Der Rechenaufwand für die Bestimmung optimaler Gitterpunkte ist ganz erheblich. Für die praktische Anwendung ist es daher anzuraten, dass für die im geplanten Anwendungsgebiet

relevanten Dimensionen und für die angestrebten Genauigkeiten erforderliche Ordnung der Gitterpunkt-Formeln vorab Generierungsvektoren einmalig bestimmt werden.

s	α	\mathbf{z}
5	2,4,6	(1, 705, 1431, 146, 307)
10	2,4,6	(1, 540, 1252, 270, 626, 135, 313, 1319, 1408, 1911)
15	2,4	(1, 842, 615, 2212, 272, 1251, 2082, 944, 1397, 2367, 626, 1462, 2031, 553, 68)
15	6	(1, 433, 2267, 435, 630, 2466, 1500, 1223, 1426, 1720, 1369, 2069, 2306, 2304, 1438)
20	2	(1, 485, 2446, 2391, 746, 1378, 29, 1550, 850, 1758 1610, 2417, 841, 2399, 2123, 922 1636, 9, 1862, 1990)
20	4	(1, 501, 701, 781 813, 1827, 1732, 1694, 177, 1072 1430, 572, 1230, 492, 1198, 1981, 1293, 2019, 307, 1124)
20	6	(1, 1249, 632, 923, 1447, 137, 909, 1482, 1301, 502 1248, 1886, 291, 524, 1193, 772, 573, 2322, 1704, 746)
25	2	(1, 261, 540, 772, 1252, 1382, 270, 386, 626, 691, 135, 193, 313, 1597, 1319, 1348, 1408, 2050, 1911, 674, ...)
25	4,6	(1, 517, 1971, 286, 185, 531, 1700, 347, 1686, 618, 1625, 1620, 1538, 1695, 265, 1843, 1691, 700, 1468, ...)
50	2	(1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 1593, 683, 1366, 229, 458, 916, 1832, 1161, 2322, 2141, ...)
50	4,6	(1, 905, 544, 1732, 582, 1080, 1230, 1818, 819, 307, 2, 1810, 1088, 961, 1164, 2160, 2460, 1133, 1638, 614, ...)
75	2,4,6	(1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 1593, 683, 1366, 229, 458, 916, 1832, 1161, 2322, 2141, ...)

Tabelle 5: Einfluss von α

3.5 Spezialfall: Fibonacci-Gitterpunkt-Formeln

Auch wenn Gitterpunkt-Formeln für die Berechnung zweidimensionaler Integranden nicht unbedingt die erste Wahl sind, so gibt es in diesem speziellen Fall doch eine sehr einfache Möglichkeit, zu optimalen Gitterpunkt-Formeln zu gelangen:

Bekanntlich sind die Fibonacci-Zahlen durch die Rekursionsformel

$$F_1 = F_2 = 1, \quad F_k = F_{k-1} + F_{k-2} \quad \text{für } k \geq 3$$

definiert. Für jede Fibonacci-Zahl F_k mit $k \geq 3$ gibt es nun eine optimale Gitterpunkt-Formel der Ordnung F_k

$$Q_k f = \frac{1}{F_k} \sum_{j=0}^{F_k-1} f \left(\left\{ \frac{j}{F_k} (1, F_{k-1}) \right\} \right)$$

In Tabelle 6 wird für die Fibonacci-Zahlen von F_3 bis F_{19} aufgelistet, welcher Fehler sich bei der Integration der Funktion f_α mit den Fibonacci-Gitterpunkt-Formeln und im Vergleich dazu mit der jeweils ‚optimalen‘ Gitterpunkt-Formel ergibt. Die Unterschiede zwischen den beiden Formeln liegen fast ausnahmslos im Bereich der Rechengenauigkeit, sind also vernachlässigbar.

Dass sich nicht exakt die gleichen Generierungsvektoren ergeben, liegt daran, dass bei der Bestimmung der optimalen Generierungsvektoren die Symmetrie ausgenutzt wird, so dass nur die erste Hälfte der möglichen Werte für ℓ untersucht wird. Tatsächlich handelt es sich bei den Generierungsvektoren der Fibonacci-Gitterpunkt-Formeln jeweils um den zu dem optimalen Generierungsvektor symmetrischen Generierungsvektor.

k	Fib_k	$\mathbf{z} = (1, Fib_{k-1})$	Fehler	$\mathbf{z} = (1, z_{opt})$	Fehler	Differenz
3	2	(1, 1)	8,3402e-01	(1, 1)	8,3402e-01	0,0000e+00
4	3	(1, 2)	4,5836e-01	(1, 1)	4,5836e-01	0,0000e+00
5	5	(1, 3)	1,9156e-01	(1, 2)	1,9156e-01	-2,2204e-16
6	8	(1, 5)	8,6807e-02	(1, 3)	8,6807e-02	0,0000e+00
7	13	(1, 8)	3,7042e-02	(1, 5)	3,7042e-02	-2,2204e-16
8	21	(1, 13)	1,5866e-02	(1, 8)	1,5866e-02	2,2204e-16
9	34	(1, 21)	6,6763e-03	(1, 10)	8,2245e-03	-1,5483e-03
10	55	(1, 34)	2,7921e-03	(1, 21)	2,7921e-03	-4,4409e-16
11	89	(1, 55)	1,1578e-03	(1, 34)	1,1578e-03	-2,2204e-16
12	144	(1, 89)	4,7732e-04	(1, 55)	4,7732e-04	0,0000e+00
13	233	(1, 144)	1,9568e-04	(1, 89)	1,9568e-04	-1,5543e-15
14	377	(1, 233)	7,9854e-05	(1, 144)	7,9854e-05	-1,7764e-15
15	610	(1, 377)	3,2453e-05	(1, 233)	3,2453e-05	6,6613e-16
16	987	(1, 610)	1,3141e-05	(1, 377)	1,3141e-05	2,6645e-15
17	1597	(1, 987)	5,3041e-06	(1, 610)	5,3041e-06	-4,4409e-16
18	2584	(1,1597)	2,1347e-06	(1, 987)	2,1347e-06	1,1102e-15
19	4181	(1,2584)	8,5693e-07	(1,1597)	8,5693e-07	2,6645e-15

Tabelle 6: Fibonacci-Gitterpunkt-Formel

3.6 Einfluss der Periodisierung

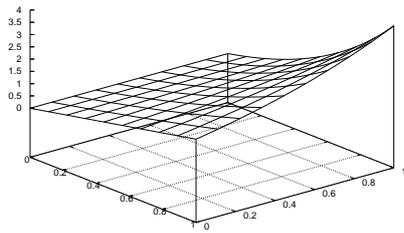
Um den Einfluss der Periodisierung eines nicht-periodischen Integranden bei der Verwendung von Gitterpunkt-Formeln zu untersuchen, wurde der Integrand $f(x, y) = (e - 2)^{-1}ye^{xy}$ auf 4 verschiedene Weisen integriert: einmal ohne Periodisierung und jeweils einmal unter Verwendung zweier polynomialer und einer trigonometrischen Periodisierung. Zum Vergleich wurde diese Vorgehensweise auch auf den bereits periodischen Integranden $f(x, y) = 1$ angewandt. Desweiteren wurde die Integration sowohl nach der Methode der guten Gitterpunkte als auch der Methode der eingebetteten Gitterpunkt-Formel durchgeführt.

In Abbildung 9 und 10 ist zunächst anschaulich dargestellt, wie der Integrand durch die Periodisierung verändert wird.

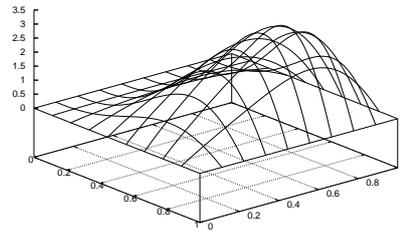
In Abbildung 11 sind die Integrationsfehler in Abhängigkeit von der Ordnung der Gitterpunkt-Formeln für jede verwendete Periodisierung dargestellt. Die jeweils weiter links beginnenden Linien der Linienpaare zu jeder Periodisierung gehören zu den Gitterpunkt-Formeln nach der Methode der guten Gitterpunkte.

Für die Funktion $f(x, y) = (e - 2)^{-1}ye^{xy}$ zeigt sich in Abbildung 11(a), dass die Periodisierung entscheidenden Einfluss auf die erzielte Genauigkeit der Integrationsergebnisse hat. Schon eine einfache polynomiale Periodisierung bringt bei steigender Ordnung einen um mehrere Zehnerpotenzen niedrigeren Integrationsfehler. Die Periodisierung mit einem Polynom höheren Grades liefert noch einmal deutlich bessere Resultate, die durch die trigonometrische Periodisierung nur geringfügig übertroffen werden.

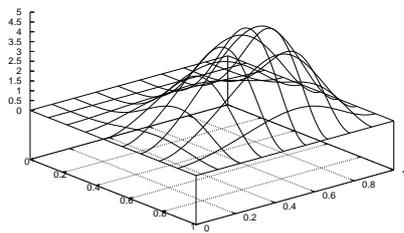
Die Funktion $f(x, y) = 1$ wurde ohne Periodisierung mit beiden Gitterpunkt-Formeln – erwartungsgemäß – exakt integriert. Bei Verwendung polynomialer Periodisierungen ergeben sich



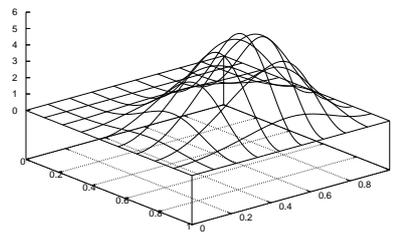
(a) $\phi(t) = t$



(b) $\phi(t) = 3t^2 - 2t^3$

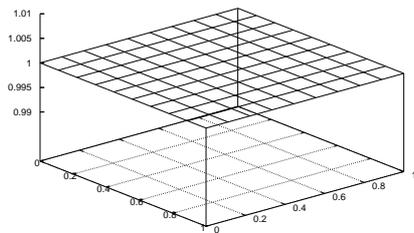


(c) $\phi(t) = t^3(10 - 15t + 6t^2)$

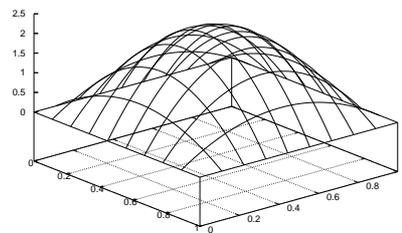


(d) $\phi(t) = t - \frac{1}{2\pi} \sin 2\pi t$

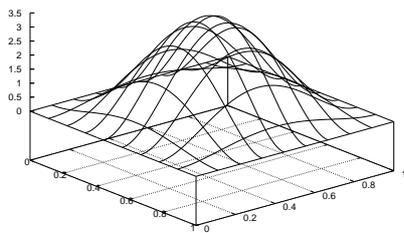
Abbildung 9: Periodisierung von $f(x, y) = (e - 2)^{-1} y e^{xy}$



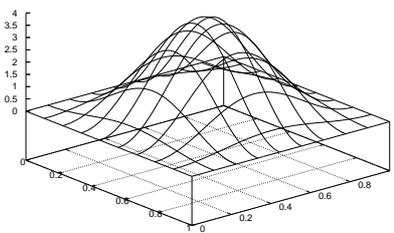
(a) $\phi(t) = t$



(b) $\phi(t) = 3t^2 - 2t^3$



(c) $\phi(t) = t^3(10 - 15t + 6t^2)$



(d) $\phi(t) = t$

Abbildung 10: Periodisierung von $f(x, y) = 1$

bei Formeln kleiner Ordnung relativ große Fehler, während bei der trigonometrischen Periodisierung die Integrationsfehler durchgängig im Bereich der Rechengenauigkeit liegen.

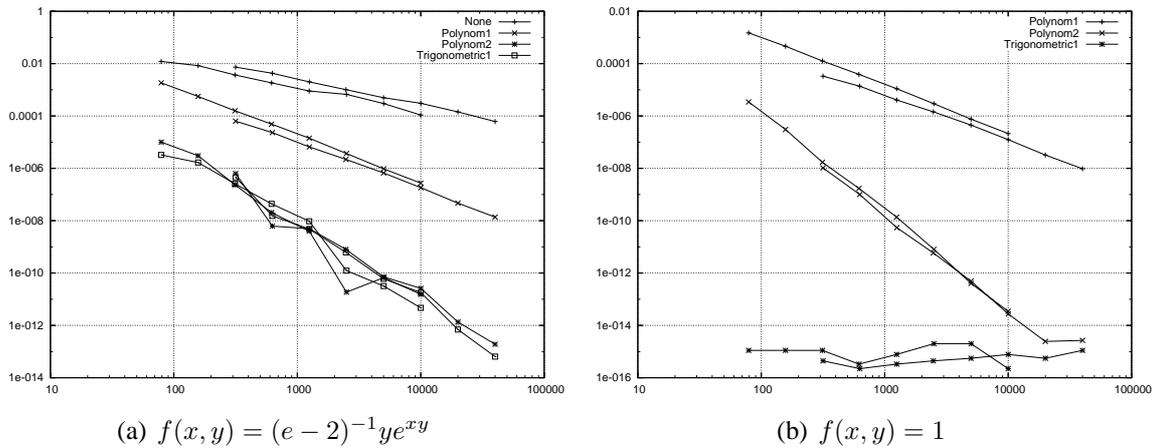


Abbildung 11: Einfluss der Periodisierung auf den Integrationsfehler

Die Ergebnisse machen deutlich, dass es sehr zu empfehlen ist nicht-periodische Integranden zu periodisieren. Trigonometrischen Periodisierungen sollte dabei in der Regel der Vorzug gegeben werden, da die Gitterpunkt-Integrationsformeln ja gerade für solche Funktionsklassen ausgelegt sind.

3.7 Test mit den Genz-Integranden-Familien

Ende der 80er Jahre wurde für den Untersuchung mehrdimensionaler Integrationsverfahren von Genz [Gen84, Gen87] ein Testpaket entwickelt, auf das in vielen Literaturquellen verwiesen wird. Daher wurden auch die hier implementierten Methoden anhand der dort beschriebenen Integranden getestet. Das Testpaket von Genz basiert auf einer Gruppe von sechs Familien von Integranden-Funktionen. Jede dieser Familien wird mit einem eigenen Namen bezeichnet:

1. OSZILLATORISCH $f^{(1)} = \cos(2\pi w_1 + \sum_{i=1}^s c_i x_i)$
2. PRODUKT-HÖCHSTWERT $f^{(2)} = \prod_{i=1}^s (c_i^{-2} + (x_i - w_i)^2)^{-1}$
3. ECKEN-HÖCHSTWERT $f^{(3)} = \left(1 + \sum_{i=1}^s c_i x_i\right)^{-(s+1)}$
4. NORMALVERTEILT $f^{(4)} = \exp\left(-\sum_{i=1}^s c_i^2 (x_i - w_i)^2\right)$
5. KONTINUIERLICH $f^{(5)} = \exp\left(-\sum_{i=1}^s c_i |x_i - w_i|\right)$
6. DISKONTINUIERLICH $f^{(6)} = \begin{cases} 0 & \text{für } x_1 > w_1 \text{ oder } x_2 > w_2 \\ \exp\left(\sum_{i=1}^s c_i x_i\right) & \text{sonst} \end{cases}$

In Abbildung 12 wird für jede Familie exemplarisch ein Beispiel in zwei Dimensionen dargestellt. Zur Vereinfachung des Testverfahrens werden alle Integranden in der Weise normiert, dass das jeweilige Integral über den Einheitswürfel C^s den Wert 1 hat. Unterschiedliche Testintegrale werden gewonnen, indem die Parameter $\mathbf{c} = (c_1, \dots, c_s)$ und $\mathbf{w} = (w_1, \dots, w_s)$ variiert

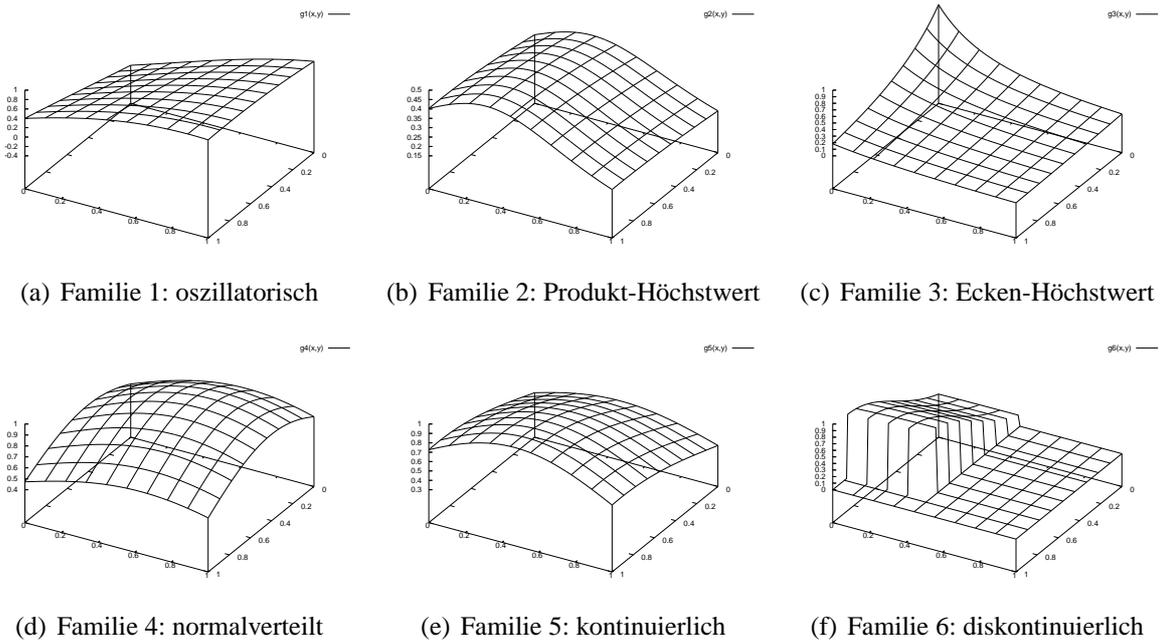
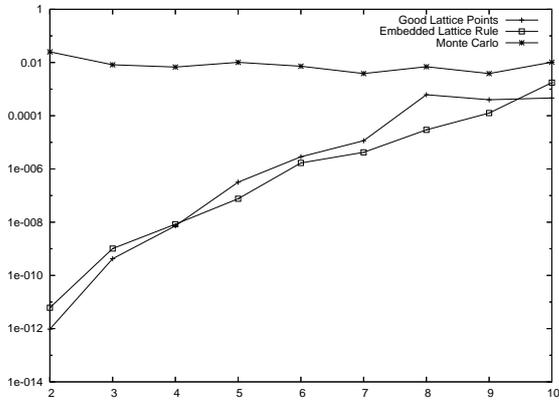


Abbildung 12: Beispiele für die Genz-Integranden-Familien

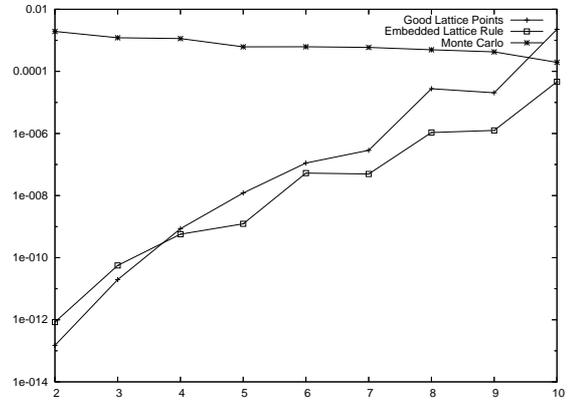
werden, wobei die Komponenten von \mathbf{w} gleichverteilt aus dem Intervall $[0, 1]$ gewählt werden. Die *Schwierigkeit* des Problems wird durch den Vektor \mathbf{c} beeinflusst: je größer die euklidische Norm $\|\mathbf{c}\|$ ist, desto schwieriger ist das Integral. Wenn h die vorgegebene Schwierigkeit des Integrals ist, wird der Vektor \mathbf{c} bestimmt als $\mathbf{c} = \gamma \mathbf{c}'$, wobei die Komponenten von \mathbf{c}' gleichverteilt aus dem Intervall $[0, 1]$ gewählt werden und $\gamma = h / \sum_{i=1}^s c'_i$ gesetzt wird.

Da Gitterpunkt-Integrationsformeln üblicherweise nur dann gut funktionieren, wenn der Integrand periodisch ist, wird in den Tests stets die vorgestellte trigonometrische Periodisierung des Integranden durchgeführt. Um den Rechenaufwand zu begrenzen, wurden die optimalen Generierungsvektoren für die Gitterpunkt-Formeln etwa der Ordnung 5.000 bzw. 10.000 vorab bestimmt. Das Programm erlaubt es, höhere Ordnungen bis zu etwa 80.000 zu spezifizieren, verwendet diese allerdings nur für die Methode der eingebetteten Gitterpunkt-Formeln. Für Integranden-Dimensionen von 2 bis 10 werden jeweils 20 Integrale mit Hilfe der Methode der guten Gitterpunkte, der Methode der eingebetteten Gitterpunkt-Formeln sowie der Monte-Carlo-Methode berechnet und der durchschnittliche tatsächliche Integrationsfehler bestimmt.

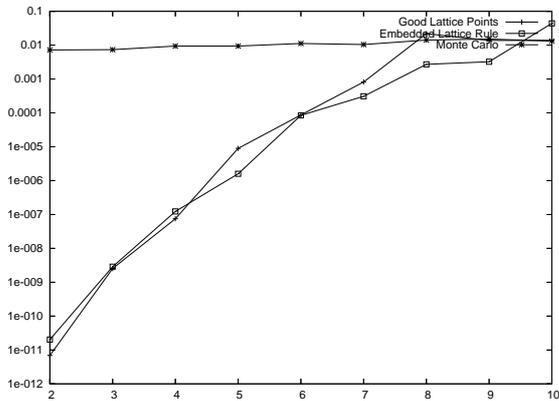
In Abbildung 13 werden die Ergebnisse grafisch dargestellt. Erwartungsgemäß schneidet das Monte-Carlo-Verfahren durchgängig für alle Integrandenfamilien recht schlecht ab. Interessant ist allerdings, dass die erzielte Genauigkeit von der Dimension des Integranden faktisch unabhängig ist. Mit Ausnahme der diskontinuierlichen Integrandenfamilie liefern die Gitterpunkt-Formeln insbesondere bei niedrigen Dimensionen der Integranden beachtlich gute Ergebnisse. Für höhere Dimensionen verschlechtern sich die Ergebnisse zwar deutlich, fallen aber – zumindest bei den untersuchten Dimensionen – in der Regel immer noch etwas günstiger aus als die der Monte-Carlo-Methode. Das Verhalten der Gitterpunkt-Formeln nach der Methode der guten Gitterpunkte und der nach der Methode der eingebetteten Gitterpunkt-Formel ist im Wesentlichen vergleichbar. Für höhere Dimensionen scheint die Methode der eingebetteten Gitterpunkt-Formel etwas günstiger zu sein. Wenig verwunderlich ist, dass faktisch alle Verfahren für die diskontinuierliche Integrandenfamilie sehr schlechte Resultate liefern.



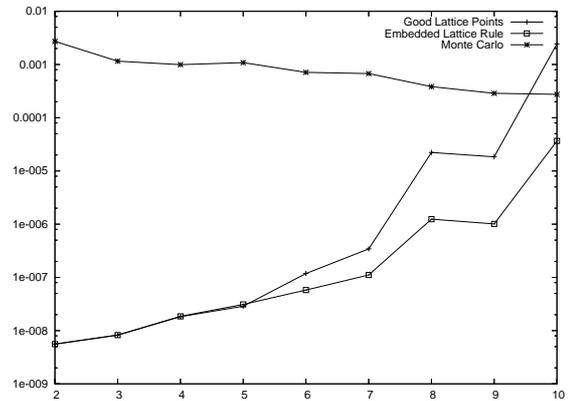
(a) Familie 1: oszillatorisch



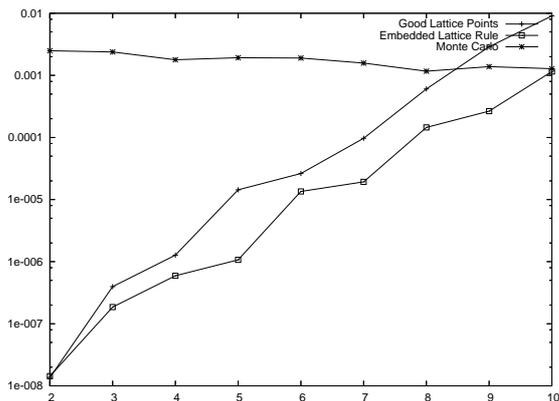
(b) Familie 2: Produkt-Höchstwert



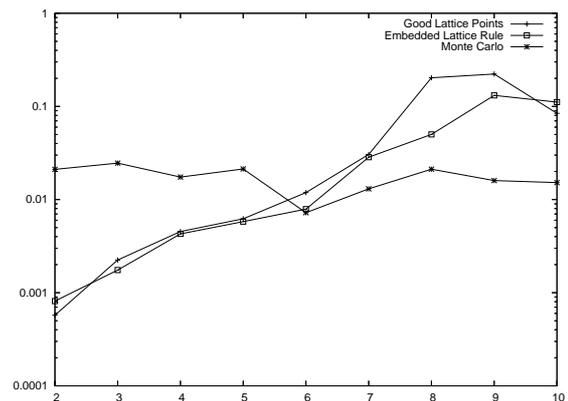
(c) Familie 3: Ecken-Höchstwert



(d) Familie 4: normalverteilt



(e) Familie 5: kontinuierlich



(f) Familie 6: diskontinuierlich

Abbildung 13: Integrationsfehler in Abhängigkeit von der Dimension des Integranden

4 Zusammenfassung

In dieser Ausarbeitung wurden die mathematischen Grundlagen von Gitterpunkt-Integrationsformeln dargelegt sowie praktische Tests mit Hilfe von in der Programmiersprache Java entwickelter Algorithmen durchgeführt. Es hat sich gezeigt, dass die Gitterpunkt-Formeln für die numerische multivariate Integration ein geeignetes Instrument darstellen, da sie gegenüber klassischen Verfahren eine akzeptable Konvergenz aufweisen. Für nicht-periodische Integranden ist der Einsatz eines geeigneten Periodisierers allerdings nahezu unerlässlich.

Für den praktischen Einsatz zur Integration hochdimensionaler Integranden ist vor allem die Methode der eingebetteten Gitterpunkt-Formel prädestiniert, da die Ordnung des Verfahrens $N = 2^s m$ ist, aber nur ein günstiger Generierungsvektors zur Ordnung m bestimmt werden muss. Verfahren nach der Methode der guten Gitterpunkte erfordern für die gleiche Verfahrensordnung N einen erheblich höheren Aufwand für die Bestimmung des Generierungsvektors. Zumindest für die getesteten Integranden hat die Methode der guten Gitterpunkte keine entscheidenden Vorteile gegenüber der Methode der eingebetteten Gitterpunkt-Formel für sich verbuchen können. Um festzustellen, ob dies nicht nur ein zufälliges Resultat ist, müssten allerdings weitergehende Untersuchungen angestellt werden.

In speziellen Einsatzgebieten, in denen etwa tiefergehende Informationen über die zu integrierende Funktion vorhanden sind, dürften allerdings Verfahren wie etwa die Dünngitter-Methoden vorteilhafter sein, da sie sich der Problemstellung besser anpassen lassen.

5 Literaturverzeichnis

- [CG03] R. Cools and H. Govaert. Five- and six-dimensional lattice rules generated by structured matrices. *Journal of Complexity*, 19(6):715–729, 2003.
- [CL01] R. Cools and J.N. Lyness. Three- and Four-dimensional K -optimal Lattice Rules of Moderate Triogonometric Degree. *Mathematics of Computation*, 70:1549–1567, 2001.
- [DK03] J. Dick and F.Y. Kuo. Constructing good lattice rules with millions of points. Technical Report AMR03/16, School of Mathematics, University of New South Wales, Sydney, Australia, April 2003.
- [Gen84] A.C. Genz. Testing multidimensional integration routines. In B. Ford, J.C. Rault, and F. Thomasset, editors, *Tools, methods, and languages for scientific and engineering computation*, pages 81–94. North-Holland, Amsterdam, 1984.
- [Gen87] A.C. Genz. A package for testing multiple integration subroutines. In P. Keast and G. Fairweather, editors, *Numerical integration: recent developments, software and applications*, pages 337–340. D. Reidel Publishing, Dordrecht, 1987.
- [GG98] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18:209–232, 1998.
- [GG03] T. Gerstner and M. Griebel. Dimension-adaptive tensor-product quadrature. *Computing (special issue on sparse grids)*, 2003.

- [HH64] J.M. Hammersley and D.C. Handscomb. *Monte Carlo Methods*. Methuen, London, 1964.
- [Hla62] E. Hlawka. Zur angenäherten Berechnung mehrfacher Integrale. *Monatshefte für Mathematik*, 66:140–151, 1962.
- [JS93] S. Joe and I.H. Sloan. Implementation of a Lattice Method for Numerical Multiple Integration. *ACM Transactions on Mathematical Software*, 19(4):523–545, December 1993.
- [Kor59] N.M. Korobov. The approximate computation of multiple integrals. *Doklady Akademii Nauk, SSSR*, 124:1207–1210, 1959. (in Russisch).
- [LZF94] Andrea Lauß, Peter Zinterhof, and Monika Feldbacher. Parallel implementation of fast algorithms for good lattice points. Technical Report D5Z-1a, Research Institute for Software Technology, University of Salzburg, April 1994.
- [MU49] N.C. Metropolis and S.M. Ulam. The Monte Carlo Method. *Journal of the American Statistical Association*, 44:335–341, 1949.
- [Nie92] H. Niederreiter. *Random number generation and quasi-Monte Carlo methods*. Society for Industrial and Applied Mathematics, 1992.
- [SJ94] I.H. Sloan and S. Joe. *Lattice Methods for Multiple Integration*. Clarendon Press, Oxford, 1994.
- [SM00] Tor Sørøvik and Jan Frode Myklebust. Grisk: An internet based search for k-optimal lattice rules. In *PARA*, pages 196–205. Springer, 2000.
- [Tel02] U. Telle. *Proseminar 1092 - Analysis: Beitrag Numerische Integration*. Fernuniversität Hagen, Dezember 2002.
- [Übe01] Christoph W. Überhuber. *Computer Numerik 2*, chapter 12.4.5, pages 144–155. Springer, Berlin, 2001.