

Bachelorthesis,
Lattice-Regeln zur Bewertung pfadabhängiger
Optionen

Oliver Horch, Matrikelnummer: 3820410

Sommersemester 2011

Inhaltsverzeichnis

1	Klassische Lattice-Regeln	4
1.1	Einleitung	4
1.2	(Integrations-)Gitter und Lattice Regeln	6
1.3	Rang-1-Regeln (Method of ‘good lattice points’)	8
1.4	Fibonacci-Regeln (2-dim)	10
1.5	Praktische Implementierung	12
2	Korobov-Konstruktion vs. Komponentenweise Konstruktion (CBC)	13
2.1	Korobov-Konstruktion (Algorithmus)	14
2.2	Komponentenweise Konstruktion	15
2.3	Reproduzierende Kerne in verschiebungsinvarianten Hilbert-Räumen und gewichtete Funktionenräume	16
2.3.1	Gewichteter Korobov-Raum	18
2.4	Schnelle komponentenweise Konstruktion (Algorithmus)	19
3	Die Wahl der Gewichte des Funktionenraums	21
3.1	Varianzanalyse (ANOVA) und Sensitivitätsindizes des Integranden	21
3.2	Anpassungsstrategie der Gewichte mit Finanzbezug	22
4	Optionsbewertung	23
4.1	Black-Scholes-Modell	23
4.2	Martingal-Ansatz	24
4.3	Pfaddiskretisierung	25
4.4	Transformation	26
4.5	Beispiele	28
5	Numerische Ergebnisse	30
5.1	Asiatische Option (geometrisches Mittel)	31
5.2	Weitere Ergebnisse	34
6	Fazit	39
7	Codes	40
7.1	<i>asianopt_lattice_bb.m</i>	40
7.2	<i>fastrank1pt.m</i>	41
A	Literaturverzeichnis	42

1 Klassische Lattice-Regeln

1.1 Einleitung

Lattice-Regeln sind Integrationsregeln, die für das Ausgangsproblem $I f = \int_{C^s} f(x) dx$ entwickelt wurden. Sie nutzen die Periodizität und Glattheit des Integranden und sind demzufolge für Funktionen mit diesen Eigenschaften sehr effizient. Im Folgenden sei stets unterstellt, dass der Integrand f in allen Komponenten einperiodisch sei, d.h.

$$f(x) = f(x + z) \quad \forall z \in \mathbb{Z}^s, \quad x \in \mathbb{R}^s.$$

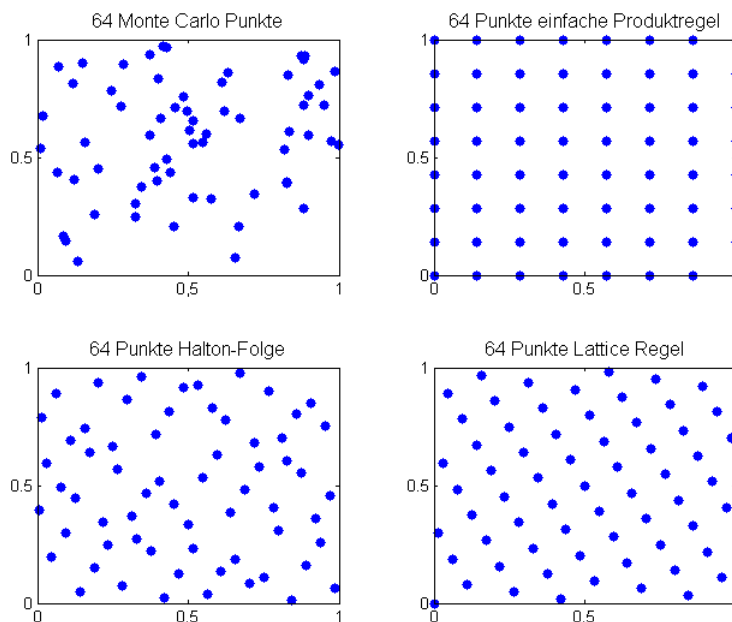
Da die Stützstellen von Lattice-Regeln keine Zufallsvariablen sind, sondern wie auch bei Niederdiskrepanzfolgen deterministisch bestimmt werden, fallen sie in die Kategorie der Quasi-Monte Carlo Verfahren. Die Grundlage aller Lattice-Regeln bildet die eindimensionale Rechteckregel

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

welche aufgrund des einperiodischen Integranden mit der Trapezregel übereinstimmt. Ihre erste mehrdimensionale Verallgemeinerung auf die Dimension s ist die Produkt-Rechteckregel,

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

Zum Vergleich der verschiedenen Monte Carlo und Quasi-Monte Carlo Verfahren zeigt die folgende Grafik jeweils 64 Punkte der unterschiedlichen Konstruktionen.



Die Produktregel weist eine sehr gute Gleichverteilung der Punkte auf, ist aber für höhere Dimensionen nicht kosteneffektiv. Beispielsweise bei einer Funktion, die nur von einer Variablen x abhängig ist, nutzt sie von ihren $N = n^s$ Stützstellen nur jede n -te für eine Verbesserung des Integrationsergebnisses. Das Ziel bei der Entwicklung von Lattice-Regeln ist es also, unter Ausnutzung bestimmter Eigenschaften eine möglichst kosteneffektive Verallgemeinerung der Rechteckregel für hohe Dimensionen zu finden. Dazu befassen wir uns zunächst mit den Eigenschaften der eindimensionalen Regel. Es ist bekannt, dass für alle Riemann-integrierbaren Funktionen f gilt:

$$R_n f \longrightarrow I f \quad \text{wie } n \longrightarrow \infty$$

Trotzdem lässt sich für Funktionen von verschiedener Glattheit und Periodizität ein großer Unterschied bei der Fehlerbestimmung beobachten.

Beispiel:

$$f(x) = x \quad \text{und} \quad g(x) = e^{\sin(2\pi x)} \quad (\text{glatt und periodisch})$$

Für den Fehler von f gilt $|R_n f - I f| = \frac{1}{2n}$, wobei der Fehler von g schon für den Wert $n = 4$ nur noch rund 0,005 beträgt. Eine Begründung hierfür lässt sich in der Fourier-Reihen-Darstellung des Integranden finden. Also treffen wir die Annahme: Nach periodischer Erweiterung habe der Integrand f eine absolut konvergente Fourierreihe,

$$f(x) = \sum_{h=-\infty}^{\infty} c_h e^{2\pi i h x}, \quad -\infty < x < \infty$$

mit $c_h = \int_0^1 e^{-2\pi i h x} f(x) dx$, $h \in \mathbb{Z}$ (absolut konvergent, wenn $\sum_{h=-\infty}^{\infty} |c_h| < \infty$).

Wendet man jetzt die Rechteckregel auf f an, so erhält man $R_n f = \sum_{h=-\infty}^{\infty} c_h R_n(e^{2\pi i h x})$. Die Auswirkung von R_n unter Abhängigkeit von h und n ist nun gegeben durch:

$$R_n(e^{2\pi i h x}) = \frac{1}{n} \sum_{j=0}^{n-1} \left(e^{\frac{2\pi i h}{n}} \right)^j = \begin{cases} 1, & \text{falls } h \text{ Vielfaches von } n, \\ 0, & \text{sonst.} \end{cases}$$

Die Regel liefert also genau die Summe derer Fourierkoeffizienten von f , welche Vielfache der Anzahl an Stützstellen sind $R_n f = \sum_{k=-\infty}^{\infty} c_{kn}$. Weil $I f = c_0$ gilt, ergibt sich für den Fehler

$$R_n f - I f = \sum_{k=-\infty}^{\infty} c_{kn} = c_n + c_{-n} + c_{2n} + c_{-2n} + \dots,$$

und man sieht, dass die führenden Koeffizienten bei periodischen glatten Funktionen für wachsendes n sehr schnell klein werden.

1.2 (Integrations-)Gitter und Lattice Regeln

Um später Lattice-Regeln definieren zu können, benötigen wir einige Grundlagen.

Def.: Ein **Gitter** im \mathbb{R}^s ist eine abzählbare Teilmenge des \mathbb{R}^s , die unter Addition und Subtraktion abgeschlossen ist.

Wegen der Abgeschlossenheit unter Subtraktion ist der Punkt $0 = (0, 0, \dots)$ in jedem Gitter enthalten ($x \in L \Rightarrow x - x \in L$). Außerdem ist durch die Additionseigenschaft jedes Gitter (außer dem trivialen, welches nur den 0-Punkt enthält) unbeschränkt.

Def.: Eine Teilmenge $\{g_1, \dots, g_t\}$ heißt **Basis** des Gitters, wenn jeder darin enthaltene Punkt eine ganzzahlige Linearkombination der Erzeugenden g_1, \dots, g_t ist.

Mit jedem Erzeugendensystem ist eine **Einheitszelle** und eine **Generatormatrix** A verbunden:

$$A = \begin{pmatrix} g_1^1 & \cdots & g_1^s \\ g_2^1 & \cdots & g_2^s \\ \vdots & \ddots & \vdots \\ g_s^1 & \cdots & g_s^s \end{pmatrix}$$

Das naheliegendste Maß für die Dichte eines Gitters L ist seine Determinante, welche das Volumen der Einheitszelle angibt und wie folgt definiert ist,

$$\det L := |\det A| .$$

Def.: Ein **Integrationsgitter** im \mathbb{R}^s ist eine abzählbare Teilmenge des \mathbb{R}^s , die unter Addition und Subtraktion abgeschlossen ist und die \mathbb{Z} als Teilmenge enthält.

Def.: Eine **Lattice-Regel** für $I f = \int_{C^s} f(x) dx$ ist eine Regel der Form

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

wobei $\{x_0, \dots, x_{N-1}\}$ alle Punkte eines Integrationsgitters $L \subset \mathbb{R}^s$ sind, die im Inneren des halboffenen Einheitswürfels $U^s := [0, 1)^s$ liegen.

Um später entscheiden zu können, ob eine Regel besser ist als eine andere, betrachten wir zunächst wieder den Fehler der allgemeinen Darstellung unter der Annahme, dass f eine absolut konvergente Fourier-Reihen-Darstellung besitzt, also

$$f(x) = \sum_{h \in \mathbb{Z}^s} c_h e^{2\pi i h x} \quad \text{mit} \quad c_h = \int_{C^s} e^{-2\pi i h x} f(x) dx, \quad h \in \mathbb{Z}^s.$$

Wenn man die allgemeine Lattice-Regel auf diesen Integranden angewendet, liefert sie $Qf = \sum_{h \in \mathbb{Z}^s} c_h Q(e^{2\pi i h x})$, ähnlich wie zuvor bei der Rechteckregel. Allerdings benötigen wir jetzt noch die Definition, des im nächsten Kapitel genauer besprochenen Dualgitters, um die Auswirkung der Regel auf $e^{2\pi i h x}$ für höhere Dimensionen bestimmen zu können.

Def.: Zu einem gegebenen Gitter L , ist das **Dualgitter** L^\perp die Menge:

$$\{h \in \mathbb{R}^s : h * x \in \mathbb{Z}, \forall x \in L\}$$

Lemma: Sei Q eine s -dimensionale Lattice-Regel und L das dazugehörige Integrationsgitter, dann gilt für $h \in \mathbb{Z}^s$:

$$Qe^{2\pi i h x} = \begin{cases} 1, & \text{falls } h \in L^\perp \\ 0, & \text{sonst} \end{cases}$$

Satz: Sei Q eine s -dimensionale Lattice-Regel und L das dazugehörige Integrationsgitter. Außerdem habe f eine absolut konvergente Fourier Reihe, dann gilt:

$$Qf - If = \sum'_{h \in L^\perp} c_h$$

Das Apostroph über der Summe symbolisiert hier, wie auch im Folgenden, dass der Nullpunkt nicht in der Summe enthalten ist. Wie nun bereits klar gemacht wurde, ist das Dualgitter sehr wichtig zur Fehlerberechnung höherdimensionaler Lattice-Regeln und somit der Schlüssel zur Unterscheidung zwischen guten und schlechten Regeln. Geometrisch liefert es Informationen über $(s-1)$ -dimensionale Hyperebenen, die auf dem Gitter L liegen. Jede Hyperebenenschar wird durch einen Punkt $h \in L^\perp$ (Normalenvektor) des Dualgitters erzeugt, so dass gilt

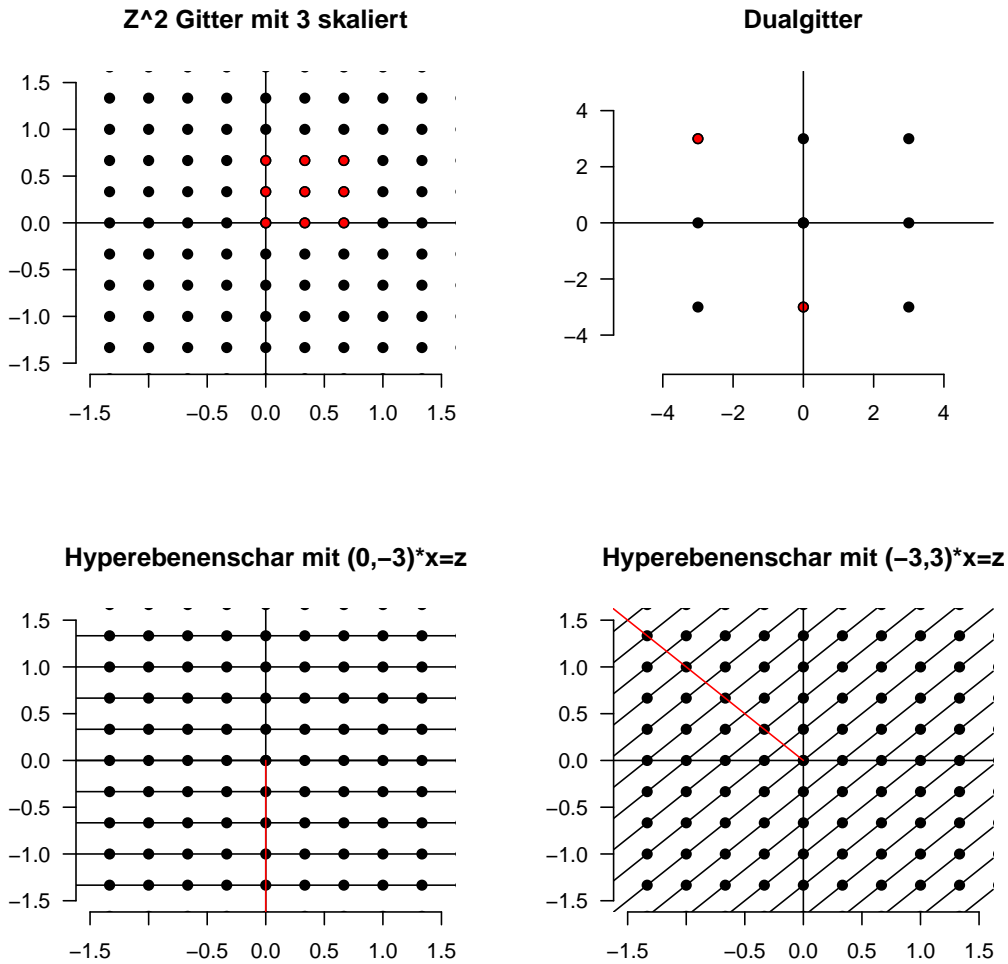
$$h * x = \delta, \quad \text{für } \delta = 0, \pm 1, \pm 2, \dots$$

Der Abstand zwischen den Hyperebenen einer Schar ist dann gegeben durch $1/||h||$, wobei

$$||h|| = (h_1^2 + \dots + h_s^2)^{1/2}$$

die gewöhnliche Euklidische Norm im \mathbb{R}^s bezeichne. Interessant sind hierbei hauptsächlich die Hyperebenen mit dem größten Abstand, da diese die höchste Punktdichte haben und

somit den größten Integrationsfehler beinhalten. Für Regeln der Ordnung N liegen durchschnittlich $\frac{1}{N}$ Punkte im Einheitswürfel des Dualgitters (Fehlerpunkte). Das folgende Beispiel zeigt das Gitter, Dualgitter und zwei durch Punkte aus dem Dualgitter erzeugte Hyperebenenscharen des mit dem Faktor $1/3$ skalierten Gitters \mathbb{Z}^2 .



1.3 Rang-1-Regeln (Method of ‘good lattice points’)

In diesem Abschnitt befassen wir uns mit Rang-1-Regeln, welche auch als ‘Method of good lattice points’ bekannt sind. Dazu benötigen wir aber zuerst die Definition des Rangs einer Regel. Jede Regel der Form $Qf = \frac{1}{N} \sum_{j=0}^{N-1} f(x_j)$ kann geschrieben werden als:

$$Qf = \frac{1}{n_1 n_2 \dots n_t} \sum_{j_t=0}^{n_t-1} \dots \sum_{j_1=0}^{n_1-1} f \left(\left\{ \frac{j_1}{n_1} z_1 + \dots + \frac{j_t}{n_t} z_t \right\} \right),$$

wobei die geschweifte Klammer $\{x\}$ hier, wie auch im Folgenden, den Bruchteil von x modulo 1 bezeichne. Der **Rang** t einer Regel Q ist nun definiert als die minimale Anzahl der für obige Darstellung benötigten Vektoren. Demzufolge sind die im Weiteren betrachteten Rang-1-Regeln von der Form:

$$Q(z, N)f = \frac{1}{N} \sum_{j=0}^{N-1} f\left(\left\{\frac{j}{N}z\right\}\right).$$

Für diesen speziellen Fall ist leicht zu erkennen, dass das Dualgitter aussieht wie folgt:

$$L^\perp = \{h \in \mathbb{Z}^s : h * z \equiv 0 \pmod{N}\}.$$

Der mit der Regel verbundene Integrationsfehler ist dann also genau

$$Q(z, N)f - If = \sum_{h * z \equiv 0 \pmod{N}} ' c_h.$$

Um gute Integrationsergebnisse zu erhalten, sollte dieser Fehler möglichst klein sein. Wie zu Beginn bereits erwähnt, wurden Lattice-Regeln für periodische und vorzugsweise glatte Integranden entwickelt. Im Folgenden wird gezeigt, wie sich der Fehler in Abhängigkeit von der Glattheitseigenschaft ändert und umschreiben lässt.

Def: Für $c > 0$ und festes $\alpha > 1$ definiere

$$E_\alpha(c) := \{f : |c_h| \leq \frac{c}{(\bar{h}_1 \bar{h}_2 \dots \bar{h}_s)^\alpha}, \bar{h} = \max(1, |h|)\}$$

die Klassen der Funktionen von bestimmter Glattheit.

Mit dieser Definition ergibt sich für den Fehler der Rang 1-Regel $Q(z, N)$ und $f \in E_\alpha(c)$,

$$|Q(z, N)f - If| \leq c \sum_{h * z \equiv 0 \pmod{N}} ' \frac{1}{(\bar{h}_1 \bar{h}_2 \dots \bar{h}_s)^\alpha}$$

Definiert man nun für $\alpha > 1$ eine Funktion f_α mit den Eigenschaften $f_\alpha \in E_\alpha(1)$ und $If_\alpha = 1$ durch

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

dann hat f_α genau den Integrationsfehler $P_\alpha(z, N)$ (Multiplizität),

$$P_\alpha(z, N) := Q(z, N)f_\alpha - 1 = \sum_{h * z \equiv 0 \pmod{N}} ' \frac{1}{(\bar{h}_1 \bar{h}_2 \dots \bar{h}_s)^\alpha}.$$

Der Integrationsfehler einer Funktion $f \in E_\alpha(c)$ lässt sich also schreiben als:

$$|\mathbf{Q}(\mathbf{z}, \mathbf{N})\mathbf{f} - \mathbf{I}\mathbf{f}| \leq \mathbf{c} * \mathbf{P}_\alpha(\mathbf{z}, \mathbf{N}).$$

(Gleichheit gilt für $f = cf_\alpha$)

Die größten Summanden in $|Q(z, N)f - 1|$ sind von der Form $\frac{1}{\rho(z, N)^\alpha}$, wobei

$$\rho(z, N) = \rho(Q(z, N)) := \min_{\substack{h \neq 0 \\ hz \equiv 0 \pmod{N}}} (\bar{h}_1 \bar{h}_2 \dots \bar{h}_s).$$

Außerdem lässt sich zeigen, dass für eine von N unabhängige Konstante $d(s, \alpha)$ gilt:

$$P_\alpha(z, N) \leq d(s, \alpha) \frac{(\log N)^{s-1}}{\rho(z, N)^\alpha}.$$

Demnach ist es zweckmäßig, entweder Verfahren zur Minimierung der Multiplizität $P_\alpha(z, N)$ oder Verfahren zur Maximierung von $\rho(z, N)$ einzusetzen, um den Integrationsfehler möglichst klein zu halten. Regeln, welche wir nun wirklich für $f \in E_\alpha(c)$, $\alpha > 1$ und $d(s, \alpha)$, $\beta(s, \alpha)$ unabhängig von N als **‘Gute’ Rang-1-Regeln (Method of ‘good lattice points’)** bezeichnen, sind diejenigen, die die Fehlerschranke

$$P_\alpha(z(N), N) \leq d(s, \alpha) \frac{(\log N)^{\beta(s, \alpha)}}{N^\alpha}$$

einhalten. Im Jahre 1978 bewies Niederreiter, dass für alle $N > 0$ ein $z \in \mathbb{R}^s$ existiert, sodass die damit verbundene Regel diese Schranke einhält. Das heißt, wenn man $\alpha > 1$ festgelegt hat und N und s gewählt sind, kann man mit einer Computeroptimierung $P_\alpha(z, N)$ minimieren, um ein z zu finden, in der Gewissheit, dass es ein ‘good lattice point’ ist. Allerdings kann dieser Suchlauf durch N^s Möglichkeiten für die Wahl von z für große N und s auch sehr zeitaufwendig werden. Für diese Suche existieren aber effektive Restriktionen und Einschränkungen, die die Anzahl der Vektoren reduzieren können. Die bekannteste Einschränkung ist die sog. **Korobov-Konstruktion**, bei der nur alle $N - 1$ Vektoren der Form $(1, a, \dots, a^{s-1}) \pmod{N}$ durchsucht werden.

1.4 Fibonacci-Regeln (2-dim)

In diesem Abschnitt wird ein Beispiel der besten bekannten 2-dimensionalen Rang-1-Regeln behandelt. Diese sind die Fibonacci-Regeln, welche jeweils von der Ordnung der zugehörigen Fibonacci-Zahl sind. Die Fibonacci-Zahlen $1, 1, 2, 3, 5, 8, 13, \dots$ sind definiert als

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

Die damit verbundenen Fibonacci-Regeln der Ordnung F_k sind dann gegeben durch

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

Zaremba zeigte 1966, dass für diese Regeln gilt:

$$\rho_k = \rho((1, F_{k-1}), F_k) = F_{k-2}, \quad k \geq 3.$$

Es lässt sich leicht nachprüfen, dass ρ_k in F_{k-2} wirklich sein Maximum annimmt, da aus den Eigenschaften der Fibonacci-Zahlen folgt, dass $\pm(F_{k-2}, 1)$ die am nächsten an 0 gelegenen Punkte des Dualgitters sind:

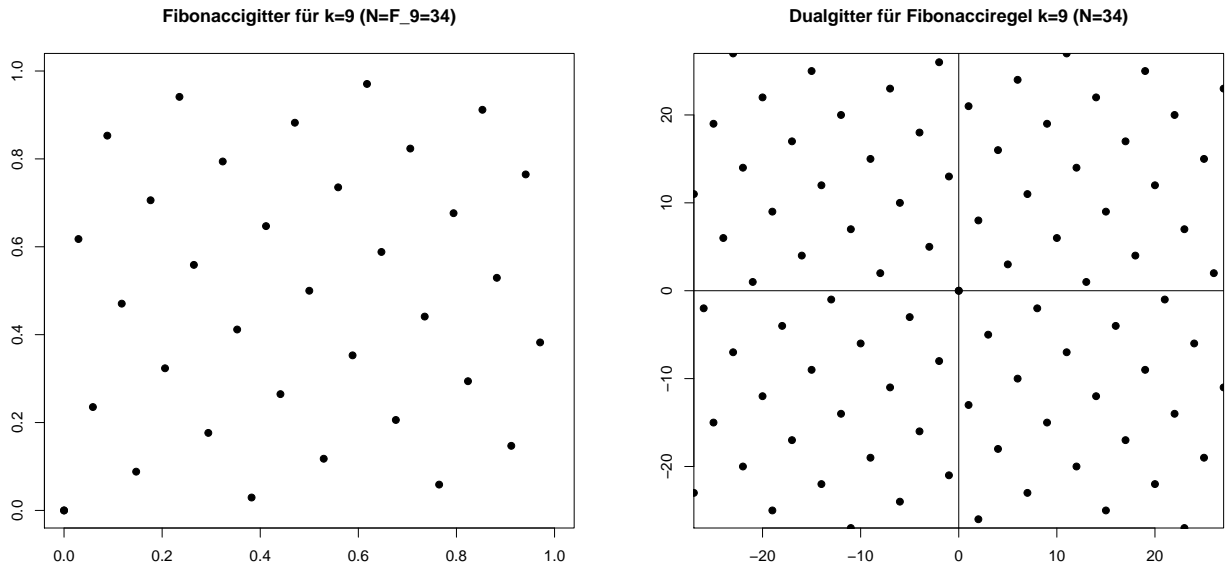
$$(F_{k-2}, 1) * (1, F_{k-1}) = F_{k-2} + F_{k-1} = F_k \equiv 0 \pmod{F_k}.$$

Schließlich kommt man zu dem Ergebnis, dass $\rho_k = ((1, F_{k-1}), F_k)$ von der Ordnung $O(N)$ ist. Damit erhält man für die Multiplizität folgende Fehlerschranke:

$$P_\alpha(z, N) \leq d(\alpha) \frac{\log N}{N^\alpha}.$$

Das folgende Beispiel für $k = 9$ zeigt das Gitter und das Dualgitter der Fibonacci-Regel der Ordnung $N = F_9 = 34$.

$$Q_9 f = \frac{1}{34} \sum_{j=0}^{33} f \left(\left\{ \frac{j}{34} (1, 21) \right\} \right)$$



1.5 Praktische Implementierung

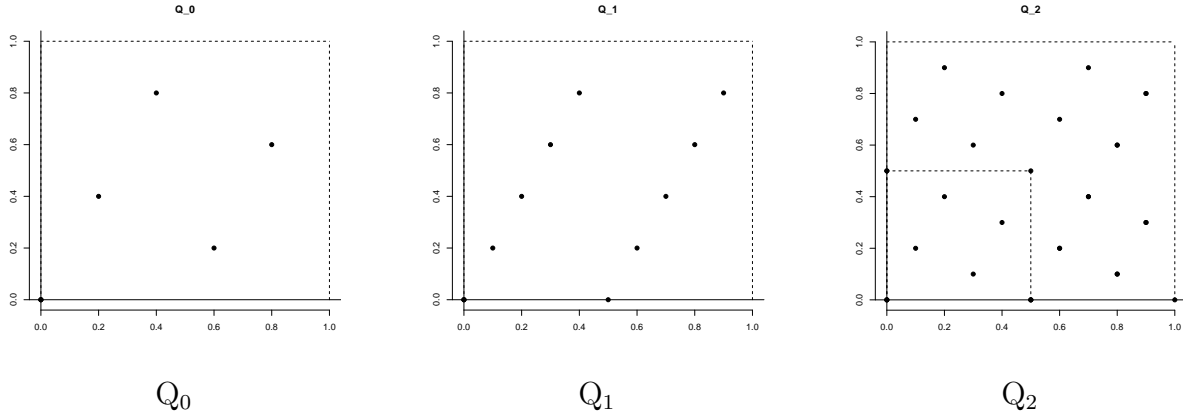
Eine eingebettete Sequenz von Regeln, die wir betrachten, sei gegeben durch:

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

für $0 \leq r \leq s$, wobei n, m teilerfremd seien und m mit $z \in \mathbb{Z}^s$ keinen nicht-trivialen Faktor gemeinsam habe.

Durch diese Darstellung wird direkt klar, warum man von einer eingebetteten Sequenz spricht, denn offensichtlich sind für $0 \leq r \leq s - 1$ alle Punkte aus Q_r auch in Q_{r+1} enthalten. Außerdem ist bekannt, dass die Regel Q_r genau $n^r m$ Punkte nutzt. Insbesondere ist Q_0 eine Rang-1-Regel und für $1 \leq r \leq s$ hat Q_r den Rang r .

Beispiel: Die folgende Grafik zeigt die Quadraturpunkte einer zwei-dimensionalen eingebetteten Sequenz mit den Parametern: $s = 2$, $n = 2$, $m = 5$, $z = (1, 2)$



Bei genauerem Hinsehen lässt sich erkennen, dass Q_2 die vierfache Kopie einer skalierten Version von Q_0 ist. Diese Beobachtung lässt sich mit folgendem Satz quantifizieren.

Satz: Wenn n und m teilerfremd sind, lässt sich die Regel Q_s schreiben als:

$$Q_s f = \frac{1}{n^s m} \sum_{k_s=0}^{n-1} \dots \sum_{k_1=0}^{n-1} \sum_{j=0}^{m-1} f \left(\left\{ \frac{j}{nm} z + \frac{(k_1, \dots, k_s)}{n} \right\} \right),$$

d.h. Q_s ist die n^s -fache Kopie der Regel Q_0 .

Der Vorteil solch eingebetteter Sequenzen liegt darin, dass sich ein Fehlerschätzer bestimmen lässt, der nur geringen zusätzlichen Rechenaufwand fordert.

2 Korobov-Konstruktion vs. Komponentenweise Konstruktion (CBC)

In diesem Kapitel wird die komponentenweise Konstruktion (component-by-component) 'guter Rang-1-Regeln' eingeführt. Da der Generatorvektor z jeder Lattice-Regel stark an die Anzahl N der Stützstellen und an die Dimension s gebunden ist, erscheint es intuitiv als sehr unwahrscheinlich, dass man für gegebenes N alle Komponenten von z nacheinander berechnen kann und dabei die bereits ermittelten Komponenten beibehält. Aus diesem Grund war die Erkenntnis umso erstaunlicher, dass solch eine Konstruktion existiert, für die die resultierenden Regeln die bekannte Fehlerschranke $P_{\alpha, N}^{(s)}(z) \leq d(s, \alpha) \frac{(\log N)^{\beta(s, \alpha)}}{N^\alpha}$ einhalten. Der Vorteil einer solchen Konstruktion liegt darin, dass sich auch die $d \leq s$ ersten Komponenten von z für ein d -dimensionales Integrationsproblem nutzen lassen, ohne diese neu berechnen zu müssen. Außerdem kann ein bereits berechneter s -dimensionaler Generatorvektor z als Input genutzt werden, um ihn für ein höherdimensionales Problem zu

vervollständigen. Um einen direkten Vergleich zwischen den beiden Verfahren ziehen zu können, wird vor der Einführung des CBC-Algorithmus zuerst der Algorithmus der klassischen Korobov-Konstruktion explizit wiederholt.

2.1 Korobov-Konstruktion (Algorithmus)

INPUT: N =Anzahl Stützstellen, s =Dimension, α =Glattheitsparameter

FOR $a = 1 : N - 1$,

BERECHNE $P_\alpha^{(s)}(z(a), N)$ mit $z(a) = (1, a, \dots, a^{s-1}) \pmod{N}$;

END FOR

SETZE (als OUTPUT)

$$z = \arg \min_{z(a)} P_\alpha^{(s)}(z(a), N)$$

Die Konstruktionskosten dieses Algorithmus sind also von der Ordnung $\mathcal{O}(sN^2)$ für die s -dimensionale Regel und für alle Regeln der Dimensionen $d \leq s$ insgesamt $\mathcal{O}(s^2N^2)$.

2.2 Komponentenweise Konstruktion

Um zu zeigen, dass die folgende CBC wirklich die traditionelle Fehlerschranke einhält, gehen wir in zwei Schritten vor. Dies ist der Fall da der erste Satz, der auch die Konstruktion enthält, mit einer anderen Darstellung als Fehlerschranke arbeitet. Diese wird dann im zweiten Satz auf die uns bereits bekannte Form gebracht. Auf die Beweise der beiden Sätze habe ich verzichtet, siehe dazu Quelle 4. Zunächst sei die Zetafunktion wie gewöhnlich definiert, durch

$$\zeta(\beta) = \sum_{k=1}^{\infty} \frac{1}{k^{\beta}}.$$

SATZ 1:

i) Für beliebiges $\beta > 1$ und N prim mit $N \geq 2\zeta(\beta) + 1$, existiert eine Folge $(g_j)_{j=1}^{\infty}$ mit $g_j \in \mathbb{Z}_N$, sodass für alle $s \geq 1$ und alle $\alpha \geq \beta$ gilt:

$$P_{\alpha, N}^{(s)}(g_1, \dots, g_s) \leq \frac{(1 + 2\zeta(\beta))^{\frac{s\alpha}{\beta}}}{N^{\frac{\alpha}{\beta}}} \quad (*)$$

ii) Die einzelnen Folgenglieder g_j , die (*) erfüllen, können rekursiv bestimmt werden, indem man $g_1 = 1$ setzt und die Komponente $g_{s+1} \in \mathbb{Z}_N$ der Wert ist, welcher $P_{\beta, N}^{(s+1)}(g_1, \dots, g_s, g)$ für $g \in \mathbb{Z}_N$ minimiert.

SATZ 2 (traditionelle Fehlerschranke):

i) Sei $\alpha > 1$, $s_{max} \geq 2$ fest und N prim mit

$$N > \exp s_{max} \frac{\alpha}{\alpha - 1}.$$

Dann existiert eine endliche Folge $(g_j)_{j=1}^{s_{max}}$, sodass für alle s mit $1 \leq s \leq s_{max}$ gilt:

$$P_{\alpha, N}^{(s)}(g_1, \dots, g_s) \leq d(s, \alpha) \frac{(\log N)^{s\alpha}}{N},$$

wobei

$$d(s, \alpha) := \left(\frac{3}{s_{max}} \right)^{s\alpha} \exp s_{max} \alpha.$$

ii) Die Folge $(g_j)_{j=1}^{s_{max}}$ lässt sich durch ii) aus Satz 1 konstruieren mit $\beta := \frac{\log N}{\log N - s_{max}}$.

Der resultierende Algorithmus der komponentenweisen Konstruktion mit $\alpha = \beta$ sieht wie folgt aus:

INPUT: N =Anzahl Stützstellen, s_{max} =Dimension, α =Glattheitsparameter

SETZE $z_1 = 1$;

FOR $s = 1 : s_{max}$,

FOR $z = 1 : N - 1$,

BERECHNE $P_{\alpha,N}^{(s)}(z_1, \dots, z_{s-1}, z)$;

END FOR

SETZE $z_s = \arg \min_{z \in \mathbb{Z}_N} P_{\alpha,N}^{(s)}(z_1, \dots, z_{s-1}, z)$;

END FOR

OUTPUT $z = (z_1, \dots, z_{s_{max}})$

Die Konstruktionskosten für diesen Algorithmus sind also auch von der Ordnung $\mathcal{O}(s_{max}^2 N^2)$, welche mit denen der Korobov-Konstruktion übereinstimmen, wenn man z für jede Dimension bis s_{max} berechnet. Das heißt, dass der Rechenaufwand für eine große Anzahl von Stützstellen in hohen Dimensionen sehr groß werden kann. Im nächsten Abschnitt dieses Kapitels wird nach der Einführung eines allgemeineren Fehlermaßes $e_{\alpha,N}^{(s)}$ an Stelle von $P_{\alpha,N}^{(s)}$, eine schnellere Version des CBC-Algorithmus präsentiert, die die Konstruktionskosten erheblich senkt.

2.3 Reproduzierende Kerne in verschiebungsinvarianten Hilbert-Räumen und gewichtete Funktionenräume

Im nächsten Abschnitt wird die schnelle CBC von Dirk Nuyens und Cools vorgestellt. Dazu führen wir in diesem Abschnitt den verschiebungsinvarianten Hilbertraum mit reproduzierendem Kern ein, aus welchem der Integrand f später stammen soll. Außerdem betrachten wir nun auch Integranden aus gewichteten Funktionenräumen, was später bei der Anwendung auf bestimmte Finanzprodukte zu einer Verbesserung der Konvergenzrate beitragen kann. Die drei zusätzlichen Annahmen bezüglich des Hilbertraumes H , die wir treffen, um eine allgemeinere Form des Fehlermaßes zu erhalten sind folgende:

- H besitze einen reproduzierenden Kern K

- K sei verschiebungsinvariant
 - H sei ein s -dimensionales Tensorprodukt, d.h. K ist das Produkt s eindimensionaler Kerne
1. Es wird die Annahme getroffen, dass in H ein reproduzierender Kern $K(x, y)$ existiert mit folgender Eigenschaft:

$$f(x) = \langle f, K(\cdot, x) \rangle_H \quad \forall f \in H, \forall x \in [0, 1]^s$$

In einem solchen Funktionenraum lässt sich durch den Kern für jeden Operator ein Repräsentant definieren. Zum Beispiel für Integration, Quadratur und insbesondere auch für den Quadraturfehler. Dieser sei von nun an bezeichnet durch:

$$E_{K, P_N}(x) := I((K(\cdot, x)) - Q_{P_N}(K(\cdot, x))).$$

Für den Integrationsfehler der Quadraturregel Q_{P_N} lässt sich mittels der Koksma-Hlawka Ungleichung folgende Fehlerschranke ableiten:

$$|I(f) - Q_{P_N}(f)| = |\langle E_{K, P_N}, f \rangle_H| \leq \|E_{K, P_N}\| * \|f\|.$$

Dadurch lässt sich der 'worst-case-error', der später minimiert werden soll, explizit darstellen durch:

$$e_{K, P_N} = \sup_{\substack{f \in H \\ \|f\| \leq 1}} |I(f) - Q_{P_N}(f)| = |I(E_{K, P_N}) - Q_{P_N}(E_{K, P_N})|^{1/2}$$

$$= \left(\int_{[0,1]^s} \int_{[0,1]^s} K(x, y) dx dy - \frac{2}{N} \sum_{x^{(k)} \in P_N} \int_{[0,1]^s} K(x^{(k)}, y) dy + \frac{1}{N^2} \sum_{x^{(k)} \in P_N} \sum_{x^{(l)} \in P_N} K(x^{(k)}, x^{(l)}) \right)^{1/2} \quad (*)$$

2. Die Verschiebungsinvarianz des Kerns bedeutet, dass gilt:

$$K(x, y) = K(\{x + \Delta\}, \{y + \Delta\}) \quad \forall \Delta \in [0, 1]^s$$

$$\Rightarrow K(x, y) = K(\{x - y\}, 0) =: K'(\{x - y\})$$

Durch die Gruppenstruktur eines Integrationsgitters (d.h falls $x, y \in L \Rightarrow \{x - y\}$) lässt sich die Doppelsumme in (*) also durch eine einzelne Summe ersetzen.

3. Die dritte Annahme, dass H ein Tensorprodukt ist, dass also K das Produkt eindimensionaler Kerne ist, lässt sich darstellen durch:

$$K_s(x, y) = \prod_{j=1}^s K_{1,j}(x_j, y_j) = \prod_{j=1}^s K'_{1,j}(\{x_j - y_j\}).$$

Durch diese beiden weiteren Annahmen lässt sich der 'worst-case-error' aus (*) wie folgt umformen.

$$\begin{aligned}
e_{K,P_N}^2 &= \int_{[0,1]^s} \int_{[0,1]^s} K(x,y) dx dy - \frac{2}{N} \sum_{x^{(k)} \in P_N} \int_{[0,1]^s} K(x^{(k)}, y) dy + \frac{1}{N^2} \sum_{x^{(k)} \in P_N} \sum_{x^{(l)} \in P_N} K(x^{(k)}, x^{(l)}) \\
&\stackrel{2.)}{=} \int_{[0,1]^s} K'(\{x-y\}) dx - \frac{2}{N} \sum_{x^{(k)} \in P_N} \int_{[0,1]^s} K'(\{y-x^{(k)}\}) dy + \frac{1}{N^2} \sum_{x^{(k)} \in P_N, x^{(l)} \in P_N} K'(\{x^{(k)}-x^{(l)}\}) \\
&\stackrel{2.)}{=} \int_{[0,1]^s} K'(x) dx - \frac{2}{N} N \int_{[0,1]^s} K'(y) dy + \frac{1}{N^2} N \sum_{k=0}^{N-1} K'(x^{(k)}) \\
&= - \int_{[0,1]^s} K'(x) dx + \frac{1}{N} \sum_{k=0}^{N-1} K'(x^{(k)}) \\
&\stackrel{3.)}{=} - \prod_{j=1}^s \int_0^1 K'_{1,j}(x) dx + \frac{1}{N} \sum_{k=0}^{N-1} \prod_{j=1}^s K'_{1,j}(x^{(k)}) \\
&= -1 + \frac{1}{N} \sum_{k=0}^{N-1} \prod_{j=1}^s K'_{1,j}(x^{(k)})
\end{aligned}$$

Unter Beachtung des Erzeugendenvektors z der Lattice-Regel Q hat unser s -dimensionales Fehlermaß also schließlich folgende Darstellung:

$$e_N^{(s)}(z) = \left(-1 + \frac{1}{N} \sum_{k=0}^{N-1} \prod_{j=1}^s K'_{1,j} \left(\left\{ \frac{kz_j}{N} \right\} \right) \right)^{1/2}$$

2.3.1 Gewichteter Korobov-Raum

Um explizite Berechnungen durchführen zu können, betrachten wir im Weiteren den gewichteten Korobov-Raum $H_{s,\alpha,\gamma}$ der auf $[0,1]^s$ definierten Funktionen. Also den Hilbertraum der periodischen, integrierbaren Funktionen mit absolut konvergenter Fourier-Reihe und Koeffizientenabfall α . Der dazugehörige reproduzierende Kern sei definiert wie folgt:

$$K_{s,\alpha,\gamma}(x,y) := \prod_{j=1}^s \left(1 + \gamma_j \sum_{h=-\infty}^{\infty} \frac{e^{2\pi i h(x_j - y_j)}}{|h|^\alpha} \right),$$

wobei $\{\gamma_j\}$ nun die Gewichtungen der einzelnen Dimensionen sind und als Parameter für die 'Relevanz' der einzelnen Variablen interpretiert werden können.

Durch diese Wahl sieht der betrachtete quadratische 'worst-case-error' für eine Lattice-Regel in unserem Funktionenraum wie folgt aus:

$$e_{\alpha,N}^2(z) = \left(-1 + \frac{1}{N} \sum_{k=0}^{N-1} \prod_{j=1}^s \left(1 + \gamma_j \sum_{h=-\infty}^{\infty} \frac{e^{2\pi i h \frac{kz_j}{N}}}{|h|^\alpha} \right) \right).$$

(Für den Fall, dass $\gamma_j = 1$ für alle j gilt, stimmt der Raum mit dem bereits aus Kapitel 1 bekannten Korobov-Raum überein und auch für das Fehlermaß gilt $e_{\alpha,N}(z) = P_{\alpha,N}(z)$.) Für den nun folgenden Algorithmus wird die Summe in der inneren Klammer der Gleichung nur noch als $\omega(\{kz_j/N\})$ bezeichnet.

2.4 Schnelle komponentenweise Konstruktion (Algorithmus)

Mit dieser neuen Darstellung des Fehlermaßes kann nun der verbesserte Algorithmus von Nuyens und Cools aus der gewöhnlichen CBC abgeleitet werden. Die Gewichte γ_j werden als gegeben vorausgesetzt und kommen somit noch als weitere Inputparameter hinzu. Die Verbesserung besteht im Wesentlichen aus zwei einfachen Schritten. Die normale CBC aus 2.2 sieht mit dem jetzigen Fehlermaß so aus:

FOR $s = 1 : s_{max}$,

FOR $z = 1 : N - 1$,

BERECHNE $e_{\alpha,N,s}^2(z) = -1 + \frac{1}{N} \sum_{k=0}^{N-1} \prod_{j=1}^s \left[1 + \gamma_j * \omega \left(\left\{ \frac{kz_j}{N} \right\} \right) \right];$

END FOR

SETZE $z_s = \arg \min_{z \in \mathbb{Z}_N} e_{\alpha,N,s}^2(z);$

END FOR

Da der Algorithmus rekursiv abläuft und somit in jedem Schleifendurchlauf bereits berechnete Werte wiederverwendet werden, kann man das Produkt in zwei Faktoren aufteilen, den bereits berechneten $j = 1, \dots, (s - 1)$ und den noch nicht berechneten $j = s$. Dadurch erhält man für die innere Schleife:

FOR $z = 1 : N - 1$,

BERECHNE $e_{\alpha,N,s}^2(z) = -1 + \frac{1}{N} \sum_{k=0}^{N-1} \left[1 + \gamma_s * \omega \left(\left\{ \frac{kz_j}{N} \right\} \right) \right] * p_{s-1};$

END FOR

Durch diese Änderung wird die Ordnung der Konstruktionskosten von $\mathcal{O}(s_{max}^2 N^2)$ auf $\mathcal{O}(s_{max} N^2)$ und $\mathcal{O}(N)$ Speicher gesenkt.

Der zweite Trick besteht darin, die innere Schleife jetzt zu einem Matrix-Vektor Produkt

umzuformen, d.h.

$$e_{\alpha,N,s}^2 = -1 + N^{-1}(\mathbb{I}_{(N-1) \times N} + \gamma \Omega_N) * p_{s-1} ,$$

wobei

$$p_s = \text{diag}(\mathbb{I}_{N \times 1} + \gamma_s \Omega_N(z_s)) * p_{s-1} ,$$

mit $p_0 := \mathbb{I}_{N \times 1}$. Die Matrix ist definiert durch

$$\Omega_N := \left[\omega \left(\left\{ \frac{kz}{N} \right\} \right) \right]_{z,k \in \mathbb{Z}_N} = \left[\omega \left(\frac{kz \pmod{N}}{N} \right) \right]_{z,k \in \mathbb{Z}_N}.$$

Da immernoch die Annahme besteht, dass N prim ist, lässt sich ein Generatorelement (Primitivwurzel) g aus \mathbb{Z}_N finden, das $\mathbb{Z}_N \setminus \{0\}$ zyklisch erzeugt. Durch Umsortieren der zyklischen Gruppe $\langle g \rangle$ lässt sich eine zyklische Matrix gewinnen, für die sich das resultierende Problem mittels der schnellen Fourier-Transformation effizient berechnen lässt. Die mittlere Schleife des CBC Algorithmus wird also erneut umgeschrieben und man erhält als Ergebnis für den gesamten Algorithmus:

INPUT: N =Anzahl Stützstellen, s_{max} =Dimension, α =Glattheitsparameter, γ =Gewichte

SETZE $z_1 = 1$; $p_0 := \mathbb{I}_{N \times 1}$;

FOR $s = 1 : s_{max}$,

BERECHNE $e_{\alpha,N,s}^2 = -1 + N^{-1}(\mathbb{I}_{(N-1) \times N} + \gamma \Omega_N) * p_{s-1}$; (1)

SETZE $z_s = \arg \min_{z \in \mathbb{Z}_N} e_{\alpha,N,s}^2(z)$; (2)

SETZE $p_s = \text{diag}(\mathbb{I}_{N \times 1} + \gamma_s \Omega_N(z_s)) * p_{s-1}$; (3)

END FOR

OUTPUT $z = (z_1, \dots, z_{s_{max}})$

1. Berechne Matrix-Vektor Produkt (schnelle Fourier-Transformation)
2. Wähle den minimalen 'worst-case-error' aus
3. Update p durch Multiplikation mit der berechneten Matrix

Dadurch dass sich das Matrix-Vektor Prudukt aus Schritt 1.) mittels schneller Fourier Transformation in $\mathcal{O}(N \log(N))$ berechnen lässt, sind die gesamten Konstruktionskosten nun bis zur Ordnung $\mathcal{O}(N \log(N) s_{max})$ mit $\mathcal{O}(N)$ Speicheraufwand gesenkt worden.

3 Die Wahl der Gewichte des Funktionenraums

Wie bereits in der Einleitung von Kapitel 2 angedeutet, besteht durch die Einführung gewichteter Funktionenräume die Möglichkeit, eine Verbesserung der Konvergenzrate des Quadraturverfahrens zu erzielen. Dies ist der Fall, da die Gewichte γ_j im Güte-/Fehlermaß $e_{\alpha, N, s}^2$ enthalten sind, dadurch den Erzeugendenvektor z und somit auch die resultierende Lattice-Regel Q entscheidend beeinflussen.

Wie bereits erwähnt, ist der Korobov-Raum, für den die klassischen Lattice-Regeln entworfen wurden, nicht gewichtet, d.h. $\gamma_j = 1$ für alle j . Aber warum sollten die Gewichte überhaupt von 1 abweichen? Eine Antwort auf diese Frage lässt sich mit Hilfe der ANOVA-Darstellung (analysis of variance) des Integranden finden.

3.1 Varianzanalyse (ANOVA) und Sensitivitätsindizes des Integranden

Jede quadratisch integrierbare Funktion $f(x)$ lässt sich als Summe ihrer ANOVA-Terme darstellen:

$$f(x) = f_\emptyset + \underbrace{\sum_{j=1}^s f_{\{j\}}(x_j)}_{1.\text{Ordnung}} + \underbrace{\sum_{1 \leq i < j \leq s} f_{\{i, j\}}(x_i, x_j)}_{2.\text{Ordnung}} + \dots + \underbrace{f_{\{1, \dots, s\}}(x_1, \dots, x_s)}_{s\text{-te Ordnung}}$$

Dabei sei $f_\emptyset = I_s(f)$ und die Terme f_u seien rekursiv definiert durch

$$f_u(x) = \int_{[0,1]^{d-|u|}} f(x) dx_{-u} - \sum_{v \subset u} f_v(x) ,$$

wobei $|u|$ die Kardinalität von u bezeichne. Außerdem definiere $-u$ die Komplementärmenge aus $\{1, \dots, s\}$ zu u und in der Summe werden nur echte Teilmengen betrachtet. Die Bezeichnung ANOVA-Darstellung kommt daher, dass die Varianzen

$$\sigma_u^2(f) = \int_{[0,1]^s} [f_u(x)]^2 dx \quad (\text{für } |u| > 0)$$

der Terme aller Ordnungen zur Gesamtvarianz von f summieren, d.h.

$$\sigma^2(f) = \sum_{\emptyset \neq u \subseteq \{1, \dots, s\}} \sigma_u^2(f) .$$

Die von Sobol eingeführten Sensitivitätsindizes sind nun definiert als:

$$S_u(f) = \frac{\sigma_u^2(f)}{\sigma^2(f)} \quad \text{für } \emptyset \neq u \subseteq \{1, \dots, s\},$$

d.h. sie geben den Beitrag jedes ANOVA-Terms relativ zur Gesamtvarianz von f an. Demnach sind sie ein Indikator für die 'Relevanz' aufeinander folgender Variablen bzw. Variablengruppen im Bezug auf die Funktion f . Um nun zu messen, wie wichtig alle Terme der Ordnung l bezüglich der Gesamtvarianz sind, summiert man einfach über alle Sensitivitätsindizes der Kardinalität $|u| = l$, also

$$R_{(l)}(f) := \sum_{|u|=l} \frac{\sigma_u^2(f)}{\sigma^2(f)} = \sum_{|u|=l} S_u(f) .$$

Mit diesen Hilfsmitteln werden wir im nächsten Abschnitt versuchen zu verstehen, warum die historischen Gewichte $\gamma_j = 1$ speziell für den Zweck der Bewertung von pfadabhängigen Optionen und anderen Finanzprodukten nicht wirklich geeignet sind und eher unbefriedigende Ergebnisse liefern.

3.2 Anpassungsstrategie der Gewichte mit Finanzbezug

Die Grundidee, die hinter der Änderung der Gewichte steckt ist die, dass der Integrand des untersuchten Problems möglichst ähnliche Eigenschaften aufweisen soll, wie 'typische' Funktionen aus dem Funktionenraum, in dem das Fehlermaß minimiert wird. Unter dem Gesichtspunkt der Sensitivitätsindizes heißt das also, dass man versucht die Gewichte des Funktionenraums so zu wählen, dass sich die Sensitivitätsindizes typischer Funktionen dieses Raumes so nah wie möglich an die des Integranden annähern.

Nun weiß man von einigen Finanzprodukten mit einer bestimmten Form (z.B. pfadabhängige Optionen), dass sie sehr einflussreiche ANOVA-Terme 1. und 2.Ordnung besitzen, was mit anderen Worten heißt, $R_{(1)}(f)$ und $R_{(2)}(f)$ tragen einen Großteil der Gesamtvarianz. Außerdem ist bekannt, dass die Bedeutung einzelner aufeinanderfolgender Variablen auch abfallend ist. Zum Vergleich untersuche man eine typische Funktion des ungewichteten Korobov-Raums, die gewisse Charakteristika aufweist. Eine solche Funktion ist der Kern $K_{s,\alpha,\gamma}(x, 0) = K'_{s,\alpha,\gamma}(x)$. Bei seiner Varianzanalyse kommt man zu dem Ergebnis, dass er speziell für die relativen Varianzen $R_{(l)}(K)$ für $l > 2$ einflussreiche Werte aufweist und seine Sensitivitätsindizes demnach stark von denen der untersuchten Finanzprodukte abweichen. Um bessere Approximationsergebnisse und eine höhere Konvergenzrate des Verfahrens zu erreichen, kann man nun mittels einer Anpassungsstrategie versuchen, die Gewichte γ_j so zu wählen, dass $K_{s,\alpha,\gamma}$ ähnliche Sensitivitätsindizes bzw. relative Varianzen $R_{(l)}(K)$ aufweist, wie die zu integrierende Funktion selbst.

Zusammengefasst besteht das Problem der konservativen Gewichte $\gamma_j = 1$ bei der Bewertung vieler Finanzprodukte darin, dass ANOVA-Terme höherer Ordnung zu stark gewichtet werden. Speziell die Terme 1. und 2.Ordnung erhalten eine viel zu schwache Gewichtung im Verhältnis zu den Termen höherer Ordnung.

Ein gern genutzter Ansatz zur Modellierung der abfallenden Relevanz aufeinanderfolgender Variablen und Variablengruppen ist die zwei Parameter Form bei der die Gewichte

folgende Darstellung haben:

$$\gamma_j = a * \gamma^j \quad , j = 1, \dots, s$$

1. γ soll hierbei den annähernd exponentiellen Verfall der Wichtigkeit aufeinanderfolgender Variablen charakterisieren, wie er bei einigen uns interessierenden Finanzprodukten zu beobachten ist.
2. a soll die Bedeutung der ANOVA-Terme von verschiedenen Ordnungen charakterisieren. Wählt man a klein, so signalisiert dies, dass hauptsächlich die ANOVA-Terme von niedriger Ordnung eine wichtige Rolle spielen.

Da die Strategie durch die Nutzung der Sensitivitätsindizes relativ zur Gesamtvarianz abläuft, liegen a und γ also beide im Einheitsintervall. Der Vorteil dieses Modells ist, dass ausschließlich zwei Parameter ermittelt werden müssen und sich dadurch gut mit Gewichten dieser Form arbeiten lässt. Man sollte zur Kenntnis nehmen, dass ein Schätzer, der sich auf alle Einzelvarianzen bzw. Sensitivitätsindizes der Teilmengen u aus $\{1, \dots, s\}$ mit Kardinalität $|u| = 1, 2$ stützt, um einiges robuster wäre, ein solches Modell aber auch einen enorm hohen Rechenaufwand benötigen würde. Außerdem ist anzumerken, dass sich die benötigten Sensitivitätsindizes der Terme 1. und 2.Ordnung für einfache Problemstellungen noch relativ gut numerisch berechnen lassen. Sobald der Integrand aber komplexer wird, kann eine Anpassungsstrategie der Gewichte schnell komplizierter zu lösen sein, als das eigentliche Integrationsproblem selbst, denn um die Varianzen $\sigma_u^2(f)$ zu ermitteln müssen ebenfalls hochdimensionale Integrale berechnet werden.

4 Optionsbewertung

In diesem Kapitel wird das bekannte Modell von Black, Scholes und Merton zur Modellierung von Aktienkursen vorgestellt und erläutert, wie sich Optionen unter den gegebenen Modellannahmen bewerten lassen. Außerdem wird vor dem Hintergrund der numerischen Berechnung auf die Diskretisierung des im Modell als kontinuierlich angenommenen Kursverlaufs eingegangen. Abschließend werden einige Optionstypen vorgestellt, die später in Kapitel 4 auch mittels Lattice-Regeln berechnet werden.

4.1 Black-Scholes-Modell

Für die Entwicklung des folgenden Modells erhielten Myron Scholes und Robert Merton im Jahre 1997 den Nobelpreis für Wirtschaftswissenschaften, Fischer Black war zu diesem Zeitpunkt bereits verstorben. Da Merton aber nicht an der Veröffentlichung des Modells beteiligt war, trägt es auch nicht seinen Namen. In ihrem Modell genügt der Zuwachs des Aktienkurses folgender stochastischen Differentialgleichung:

$$dS(t) = \mu S(t)dt + \sigma S(t)dW(t)$$

Alternativ lässt sich der Zuwachs durch eine Integralgleichung darstellen:

$$S(t+s) - S(t) = \int_t^{t+s} \mu S(u) du + \int_t^{t+s} \sigma S(u) dW(u)$$

Die Kursänderungsrate hängt somit von den Konstanten μ, σ und dem stochastischen Teil $W(t)$ ab. μ ist hierbei ein Driftparameter, der später durch den angenommenen Marktzins r ersetzt wird. σ bezeichnet die Volatilität des zugrunde liegenden Wertpapiers. $W(t)$ steht für einen Wiener-Prozess, ein zeitstetiger stochastischer Prozess mit unabhängigen normalverteilten Zuwächsen, der auch als geometrische Brownsche Bewegung bekannt ist. Die analytische Lösung dieser stochastischen Differentialgleichung ist gegeben durch:

$$S(t) = S(0)e^{(\mu - \frac{1}{2}\sigma^2)t + \sigma W(t)}$$

Aus dem Beweis der Richtigkeit dieser Lösung mittel des *Itô*-Lemmas lassen sich zwei wichtige Bemerkungen ableiten.

1. Der Kursverlauf $S(t)$ eines Wertpapiers ist unter den gegebenen Modellannahmen log-normalverteilt.
2. Für den Erwartungswert des modellierten Wertpapiers gilt:

$$E[S(t)] = S(0)e^{\mu t}$$

Die zweite Anmerkung wird im folgenden Martingal-Ansatz, nach Wechsel des Wahrscheinlichkeitsmaßes, noch von Bedeutung sein.

4.2 Martingal-Ansatz

Def.: Ein stochastischer Prozess $S(t)$ heißt Martingal, falls für seinen Erwartungswert gilt:

$$E[S(s) \mid S(t) = x] = x \quad \forall s > t$$

Def.: Das äquivalente Martingalmaß P^* zur Wahrscheinlichkeitsverteilung P von $S(t)$ ist dasjenige Wahrscheinlichkeitsmaß, unter dem der diskontierte Prozess $e^{-r\Delta t}S(t)$ die Martingaleigenschaft erfüllt, d.h.

$$e^{-r\Delta t}E^*[S(t + \Delta t)] = S(t) .$$

Die Drift μ des Black-Scholes-Modells wird also unter dem äquivalenten Martingalmaß durch den risikofreien Marktzinssatz r ersetzt, denn durch Multiplikation mit $e^{r\Delta t}$ erhält man für den Erwartungswert unter dem neuen Maß:

$$E^*[S(t)] = e^{-r\Delta t}S(0) \quad (\text{Vergleich Bem. 2.) in 4.1})$$

Das bedeutet, unter dem Wahrscheinlichkeitsmaß P^* erwartet man von einer Investition in ein riskantes Wertpapier denselben Zinsertrag, wie bei der Investition in den risikofreien Kapitalmarkt. Aus diesem Grund wird P^* auch als 'risikoneutrales' Wahrscheinlichkeitsmaß bezeichnet. Da P^* aber nur ein anderes Maß ist, das zu Bewertungszwecken dient, ist diese Bezeichnung nicht mit der Annahme zu verwechseln, dass Investoren tatsächlich risikoneutral handeln.

SATZ:(Martingal-Anastz für europäische Optionen)

Der faire Wert V einer Option ohne vorzeitigem Ausübungsrecht lässt sich als der diskontierte Erwartungswert der Auszahlung unter dem äquivalenten Martingalmaß P^* darstellen:

$$V(S, 0) = e^{-rT} E^*[V(S, T)]$$

bzw. im Falle pfadabhängiger Optionen:

$$V(S, 0) = e^{-rT} E^*[V(S_{t_1}, \dots, S_T)]$$

Dieser Satz bildet die entscheidende Grundlage zur Optionsbewertung mittels Monte Carlo oder Quasi-Monte-Carlo-Verfahren, denn durch diese Darstellung des Optionspreises kann der Aktienkurs oft simuliert und somit der Erwartungswert der Auszahlung numerisch berechnet werden. Zu Simulationszwecken werden im nächsten Abschnitt noch zwei verschiedene Arten der Pfaddiskretisierung erläutert. Schließlich werden in Kapitel 5 einige Optionen nach dem Prinzip des Martingal-Ansatzes simuliert.

4.3 Pfaddiskretisierung

Wie in Abschnitt 4.1 bereits erwähnt, folgt der Aktienkurs im Black-Scholes-Modell einem zeitstetigen Prozess. Um diesen simulieren zu können, muss er diskretisiert werden. Durch diese Diskretisierung wächst die Dimension des entstehenden Integrals, da jeder einzelne Zeitschritt eine weitere Variable mit sich bringt. Wenn wir die analytische Lösung

$$S(t) = S(0)e^{(\mu - \frac{1}{2}\sigma^2)t + \sigma W(t)}$$

des Black-Scholes-Modells betrachten, sehen wir dass der Aktienkurs bis auf den Teil $W(t)$ ausschließlich von Konstanten abhängt. Der einzige Teil der simuliert werden muss, um den Kursverlauf zu diskretisieren, ist also der Wiener-Prozess. Dazu werden wir nun zwei verschiedene Möglichkeiten betrachten. Die Standardvariante und auch die einfachste Möglichkeit ist der so genannte **Random Walk** (Zufallspfad). Hierbei wird der Prozess zu einem bestimmten Zeitpunkt jeweils durch den Stand des vorherigen Zeitpunktes und einer normalverteilten Zufallsvariablen mit Varianz Δt erzeugt. Angenommen man teilt den Prozess in s verschiedene Zeitpunkte $t_1 < t_2 < \dots < t_s = T$ mit äquidistanten Abständen ein, dann gilt folgende Beziehung:

$$W(t_j) = W(t_{j-1}) + \sqrt{\Delta t} z_j \quad \text{für } j = 1, \dots, s ,$$

wobei $W(0) := 0$ und z_j $N(0, 1)$ -normalverteilte Zufallsvariablen sind. Durch Einsetzen in die Black-Scholes Lösung ergibt sich für den Aktienkurs zu jedem dieser Zeitpunkte:

$$S(t_i) = S(0)e^{(\mu - \frac{1}{2}\sigma^2)t_i + \sigma\sqrt{\Delta t}\sum_{j=1}^i z_j} \quad \text{für } i = 1, \dots, s$$

Die zweite Möglichkeit um den Wiener-Prozess diskret zu simulieren ist die **Brownsche Brücke**. Der Unterschied zum Random Walk ist, dass die Werte nicht inkrementell, sondern hierarchisch erzeugt werden. Das heißt die Bedeutung der einzelnen Zufallsvariablen z_j ist abfallend gestaffelt. Jeder Wert wird aus einem vorherigen und einem zukünftigen Zeitpunkt gewonnen, wobei gilt $W(0) := 0$ und $W(T) := \sqrt{T}z$. Zur Vereinfachung nehmen wir an, dass s nun eine Potenz von 2 sei, dann werden alle Werte mittels folgender Gleichung ermittelt:

$$W(t_j + \Delta t) = \frac{W(t_j) + W(t_j + s\Delta t)}{2} + \sqrt{\frac{\Delta t}{2}} z_j$$

Da Δt in der Konstruktion der Brownschen Brücke nicht konstant ist, sondern sich immer halbiert, haben die Zufallsvariablen $\sqrt{\frac{\Delta t}{2}} z_j$ unterschiedliche Varianzen. Nach diesem Schema ergibt sich folgende Staffelung der Relevanz der einzelnen Werte $\{W(T)\}$, $\{W(T/2)\}$, $\{W(T/4), W(3T/4)\}$, $\{W(T/8), W(3T/8), W(5T/8), W(7T/8)\}$ und so weiter. Den Aktienkurs zu einem bestimmten Zeitpunkt erhält man einfach wieder durch Einsetzen der ermittelten Werte in die Gleichung

$$S(t_i) = S(0)e^{(\mu - \frac{1}{2}\sigma^2)t_i + \sigma W(t_i)}.$$

Der Vorteil dieser Konstruktion im Bezug auf die Bewertung pfadabhängiger Optionen ist der, dass die in Kapitel 3 angesprochene abfallende Relevanz aufeinander folgender Variablen besser synthetisiert wird als bei der Standardkonstruktion mittels Random Walk.

4.4 Transformation

Wir betrachten zu Beginn die Darstellung einer einfachen europäischen Option als Erwartungswert und wie sich die Integralgrenzen auf das Einheitsintervall transformieren lassen. Danach wird der Vorgang für pfadabhängige Optionen wiederholt, um Lattice-Regeln auf das entstehende hochdimensionale Problem anwenden zu können. In diesem und dem nächsten Abschnitt bezeichne $\Phi(x)$ stets die kumulative Standardnormalverteilung.

- Eine einfache europäische Call Option hat die Auszahlungsfunktion $V(S, T) = \max(S(T) - K, 0)$. Also lässt sich der faire Wert zum Zeitpunkt 0 nach dem Martingal-Ansatz unter P^* darstellen als:

$$V(S, 0) = e^{-rT} E^*[\max(S(T) - K, 0)]$$

Nutzt man nun die Lösung der Black-Scholes SDG zur Modellierung des Aktienkurses, so sieht dieser Erwartungswert wie folgt aus:

$$V(S, =) = e^{-rT} \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} [S(0)e^{(r-\frac{1}{2}\sigma^2)T+\sigma\sqrt{T}x} - K]^+ dx$$

Die Dichtefunktion $\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ der Normalverteilung steht im Integral, da der Zuwachs des Wiener-Prozesses normalverteilt ist und dieser durch die Variable x beschrieben wird. Durch die Substitution $x = \Phi^{-1}(y)$ kürzt sich die Dichtefunktion weg und man erhält dem folgende Integral über das Einheitsintervall:

$$\int_0^1 [S(0)e^{(r-\frac{1}{2}\sigma^2)T+\sigma\sqrt{T}x} - K]^+ dx$$

- Für pfadabhängige Optionen ohne vorzeitigem Ausübungsrecht nutzen wir eine allgemeiner Schreibweise für die Auszahlungsfunktion $V(S_{t_1}, \dots, S_{t_s})$. Das heißt, dass in die Auszahlung die Aktienkurse s diskreter Zeitpunkte einfließen. Zur Vereinfachung seien die Zeitabstände äquidistant gewählt, d.h. $t_i = \Delta t * i$ mit $\Delta t = T/s$. Der Zufallsvektor $x = (x_1, \dots, x_s)$, der den Wiener-Prozess darstellt, genügt dann einer multivariaten Normalverteilung mit Erwartungswert 0 und Kovarianzmatrix

$$\Sigma = \left(\min(t_i, t_j) \right)_{i,j=1}^s = \begin{pmatrix} \Delta t & \Delta t & \Delta t & \dots & \Delta t \\ \Delta t & 2\Delta t & 2\Delta t & \dots & 2\Delta t \\ \Delta t & 2\Delta t & 3\Delta t & \dots & 3\Delta t \\ \vdots & & & \ddots & \vdots \\ \Delta t & & & & s\Delta t \end{pmatrix}.$$

Um die einzelnen Aktienkurse nicht in der Form $S(t_i) = S(0)e^{(r-\frac{1}{2}\sigma^2)T+\sigma W(t_i)}$ aus-schreiben zu müssen, kann man $\mu_j := \log(S_0) + (r - \sigma^2/2)t_j$ definieren. Dadurch erhält man $S(t_i) = \exp(\mu_j + \sigma W(t_j))$ und somit für die diskretisierte Auszahlungsfunktion:

$$V(S_{t_1}, \dots, S_{t_s}) = V(e^{\mu_1+\sigma x_1}, \dots, e^{\mu_s+\sigma x_s}) =: V'(x)$$

Nach Anwenden des Martingal-Ansatzes ist der Optionspreis zum Zeitpunkt $t = 0$ also ein s -dimensionales Integral der Form:

$$V_0 = E^*[e^{-rT}V(S_{t_1}, \dots, S_{t_s})] = \frac{e^{-rT}}{(2\pi)^{s/2}\sqrt{\det \Sigma}} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} V'(x) e^{-\frac{1}{2}x^T \Sigma^{-1}x} dx$$

Durch eine $s \times s$ Matrix A die $AA^T = \Sigma$ erfüllt, lässt sich $(x_1, \dots, x_s)^T = A(y_1, \dots, y_s)^T$ linear transformieren, sodass die Kovarianzmatrix aus dem Integral verschwindet:

$$V_0 = \frac{e^{-rT}}{(2\pi)^{s/2}} \int_{\mathbb{R}^s} V'(Ay) e^{-\frac{(y_1^2+\dots+y_s^2)}{2}} dy$$

Um das Problem nun auf die Standardform zu bringen, d.h. man integriert über den Hypereinheitswürfen $[0, 1]^s$, muss nur noch jede einzelne Variable durch $y_i = \Phi^{-1}(x_i)$ substituiert werden, wie im univariaten Fall. Dadurch erhält man letztendlich eine Problemstellung, auf die sich unsere Lattice-Regeln anwenden lassen.

$$V_0 = \int_{[0,1]^s} V'(A\Phi^{-1}(x))dx \approx \frac{1}{N} \sum_{k=1}^N V' \left(\left\{ \frac{k * z}{N} \right\} \right)$$

4.5 Beispiele

In diesem Abschnitt werden wir ausschließlich Optionen betrachten, für die geschlossene Lösungsformeln existieren. Das hat den Hintergrund, dass man durch die Existenz der Lösungsformel im nächsten Kapitel die Abweichung der Approximation vom exakten Ergebnis genau berechnen kann.

- Eine einfache europäische Call Option hat an ihrem Laufzeitende, d.h. zum Ausübungszeitpunkt T , den Wert

$$V(S, T) = \max(S(T) - K, 0) =: [S(T) - K]^+ .$$

Die analytische Lösung einer Option mit dieser Auszahlungsfunktion ist gegeben durch die bekannte Black-Scholes-Formel

$$V_{BS} = S(0)\Phi(d_1) - e^{-rT}K\Phi(d_2)$$

mit

$$d_1 = \frac{\log\left(\frac{S(0)}{K}\right) + (r + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}$$

und

$$d_2 = d_1 - \sigma\sqrt{T} .$$

- Nun werden drei verschiedene Typen pfadabhängiger Optionen vorgestellt, deren exakter Wert sich jeweils über eine verallgemeinerte Form der obigen Black-Scholes-Formel berechnen lässt.

1. **Asiatischen Optionen** liegt die Überlegung zugrunde, dass die Kursentwicklung kurz vor Laufzeitende bei einfachen Optionen einen zu großen Einfluss auf den Wert der Option hat. Deshalb wählt man statt $S(T)$ eine Mittelung des Aktienkurses über die gesamte Laufzeit der Option für die Auszahlungsfunktion. Wir betrachten nur die geometrische Mittelbildung, da hierfür im Gegensatz

zum arithmetischen Mittel eine geschlossene Lösungsformel existiert. Die Auszahlungsfunktion einer asiatischen Option mit geometrischer Mittelung über s diskrete Zeitpunkte lautet

$$V_T(S_{t_1}, \dots, S_{t_s}) = \left[\left(\prod_{i=1}^s S(t_i) \right)^{1/s} - K \right]^+ .$$

Die zugehörige Black-Scholes-Formel zur exakten Bestimmung des Optionspreises sieht aus, wie folgt:

$$V_{geo}^{dis} = S_0 A \Phi(d + d\sigma\sqrt{T_1}) - e^{-rT} K \Phi(d)$$

mit

$$\begin{aligned} A &= e^{-r(T-T_2) - \sigma^2(T_2-T_1)/2} , \\ d &= \frac{\log\left(\frac{S_0}{K}\right) + (r - \frac{1}{2}\sigma^2)T_2}{\sigma\sqrt{T_1}} , \\ T_1 &= T - \frac{M(M-1)(4M+1)}{6M^2}\Delta t , \\ T_2 &= T - \frac{(M-1)}{2}\Delta t . \end{aligned}$$

Durch immer kleiner werdende Zeitabstände und der damit verbundenen wachsenden Anzahl an berücksichtigten Aktienkursen konvergiert die diskrete Auszahlungsfunktion gegen die kontinuierliche

$$V_T(S(t)) = \left[e^{1/T \int_0^T \log(S(t))dt} - K \right]^+ .$$

Die Black-Scholes-Formel für eine asiatische Option mit kontinuierlicher geometrischer Mittelbildung ist dann gegeben durch:

$$V_{geo} = S_0 e^{-(r + \frac{1}{6}\sigma^2)\frac{T}{2}} \Phi(d + \sigma\sqrt{T/3}) - e^{-rT} K \Phi(d)$$

mit

$$d = \frac{\log\left(\frac{S_0}{K}\right) + (r - \frac{1}{2}\sigma^2)\frac{T}{2}}{\sigma\sqrt{T/3}}$$

2. **Lookback Optionen** liegt dieselbe Überlegung zugrunde wie auch bei asiatischen Optionen. Der Unterschied ist, dass entweder $S(T)$ oder K in der Auszahlungsfunktion nun nicht durch eine Mittelwertbildung, sondern den höchsten bzw. niedrigsten Aktienkurs während der Laufzeit der Option ersetzt wird. Die Auszahlungsfunktion eines Lookback Calls mit variablem Strikepreis lautet also

$$V_T(S(t)) = \left[S(T) - \min_{t \leq T} S(t) \right]^+ .$$

Da dieser Optionstyp immer im Geld ist (außer $S(T)$ ist wirklich der niedrigste Kurs seit Emission der Option), ist der Preis im Vergleich zu einfachen europäischen Optionen immer höher. Die Black-Scholes-Formel einer Lookback Option mit variablem Strike ist gegeben durch:

$$V_{Lb} = V_{BS}(S_0, S_0) + \frac{S_0\sigma^2}{2r} \left(e^{-rT}\Phi(d) - \Phi(-d + \sigma\sqrt{T}) \right)$$

mit

$$d = \frac{(r + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}$$

3. **Barrier Optionen** sind im Vergleich zu einfachen europäischen Optionen billiger, da diese Optionen erst an Wert gewinnen bzw. wertlos werden (in/out), sobald sie eine gewisse Schranke über- bzw. unterschreiten (up/down). Betrachtet man zusätzlich noch Call und Put Optionen, dann gibt es acht verschiedene Arten von Barrier Optionen. Als Beispiel betrachten wir eine Call Option die beim Durchbrechen einer unteren Schranke H wertlos wird, also einen Down-Out-Call mit der Auszahlungsfunktion

$$V_T(S(t)) = \begin{cases} 0, & \text{falls } \exists t < T \text{ mit } S(t) < H \\ [S(T) - K]^+, & \text{sonst} \end{cases}$$

Die exakte Lösung für den fairen Preis dieser Option zum Zeitpunkt $t = 0$ ist durch folgende Black-Scholes-Formel gegeben:

$$V_{Bar} = V_{BS}(S_0, \bar{H}) - ZV_{BS}\left(\frac{H^2}{S_0}, \bar{H}\right) + e^{-rT}(\bar{H} - K) \left[\Phi(d(S_0, \bar{H})) - Z\Phi\left(d\left(\frac{H^2}{S_0}, \bar{H}\right)\right) \right]$$

mit

$$Z = \left(\frac{H}{S_0}\right)^{\frac{2r}{\sigma^2} - 1}, \quad \bar{H} = \max\{H, K\}$$

und $d(S, K)$ wie d_1 in der einfachen Black-Scholes-Formel

$$d(S, K) = \frac{\log\left(\frac{S}{K}\right) + (r + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}$$

5 Numerische Ergebnisse

In diesem Kapitel werden numerische Ergebnisse der Bewertung pfadabhängiger Optionen mittels Lattice-Regeln präsentiert. Zur Berechnung des Erzeugendenvektors z wurde das Programm *fastrank1pt.m* (siehe Anhang) von Dirk Nuyens mit verschiedenen Inputparametern genutzt. Außerdem wurden die Pfade aller Simulationen jeweils einmal mittels Random Walk und einmal mittels Brownscher Brücke diskretisiert. Als Funktionenraum wurde der in 2.3.1 eingeführte Korobov-Raum mit Glattheitsparameter $\alpha = 2$ gewählt. Das

hat den Hintergrund, dass sich die unendliche Summe im zu minimierenden worst case error mit Hilfe des Bernoulli Polynoms vom Grad α umschreiben lässt und das Bernoulli-Polynom $B_2(x) = x^2 - x + 1/6$ bekannt ist. Für $\alpha = 2$ gilt dementsprechend:

$$\sum_{h=-\infty}^{\infty} \frac{e^{2\pi i h x}}{|h|^\alpha} = \frac{(2\pi)^\alpha}{(-1)^{\alpha/2+1} + \alpha!} B_\alpha(x) = 2\pi^2 \left(x^2 - x + \frac{1}{6}\right)$$

Variiert wurden hauptsächlich die Dimension, d.h. die Anzahl der Zeitschritte, mit welchen diskretisiert wurde und die Gewichte des Funktionenraums. Aber auch mit verschiedenen Ausübungspreisen und Optionslaufzeiten wurde experimentiert. In allen Abschnitten sind Tabellen mit den Ergebnissen für $N = 101, 1009, 10007, 100003, 1000003$ Punkte und die entsprechenden Fehlerplots abgebildet. Punkte und Fehler wurden dann auf eine logarithmische Skala aufgetragen und es wurde eine lineare Regressionsanalyse durchgeführt, um eine grobe Approximation der entstehenden Konvergenzrate zu erhalten.

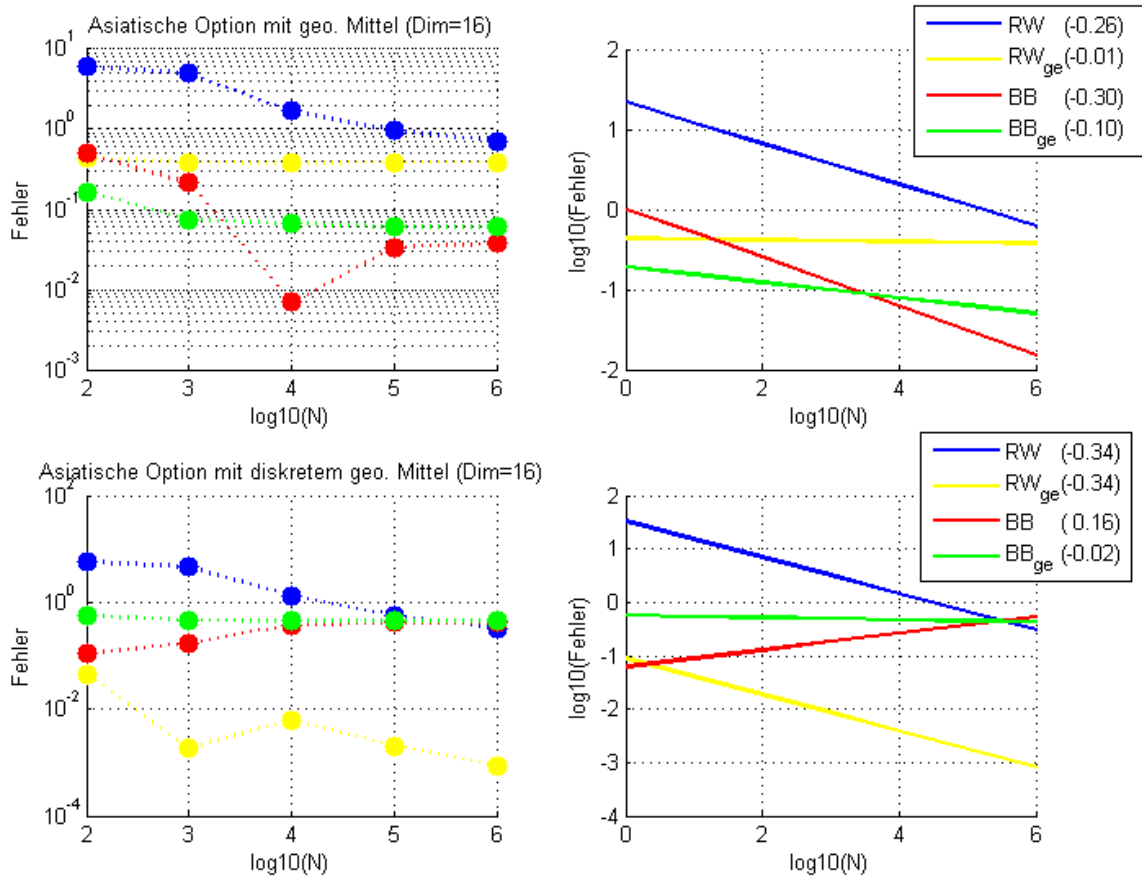
5.1 Asiatische Option (geometrisches Mittel)

In diesem Abschnitt wird versucht anhand numerischer Berechnungen die Wichtigkeit der richtigen Wahl der Gewichte des Funktionenraums darzustellen. Dazu wurden die (durch die in Abschnitt 3.2 angesprochene Vergleichsstrategie ermittelten) Gewichte aus Quelle 15 übernommen und die resultierenden Ergebnisse mit denen des ungewichteten Funktionenraumes verglichen. Die Modellparameter der bewerteten Option waren $S_0 = 100$, Strike 100, Laufzeit 1 Jahr, Volatilität 0,2 und ein risikofreier Zins von 0,1. Als exakte Ergebnisse betrachten wir für 16, bzw. 64 Zeitschritte jeweils die Lösung der stetigen und der diskreten Black-Scholes-Formel.

Die Gewichte für $D = 16$ sind gegeben durch $\gamma_j = 0,2 * 0.93^j$.

D=16	N	101	1009	10007	100003	1000003	Konvr.
RW	ungew.	6.05518E+00	4.95075E+00	1.71037E+00	9.67733E-01	7.08465E-01	-0.2576
	gew.	4.31745E-01	3.87767E-01	3.79757E-01	3.83914E-01	3.86810E-01	-0.0100
BB	ungew.	4.95727E-01	2.15153E-01	6.98435E-03	3.39298E-02	3.80477E-02	-0.3034
	gew.	1.67436E-01	7.42240E-02	6.61322E-02	5.99559E-02	6.07280E-02	-0.0975

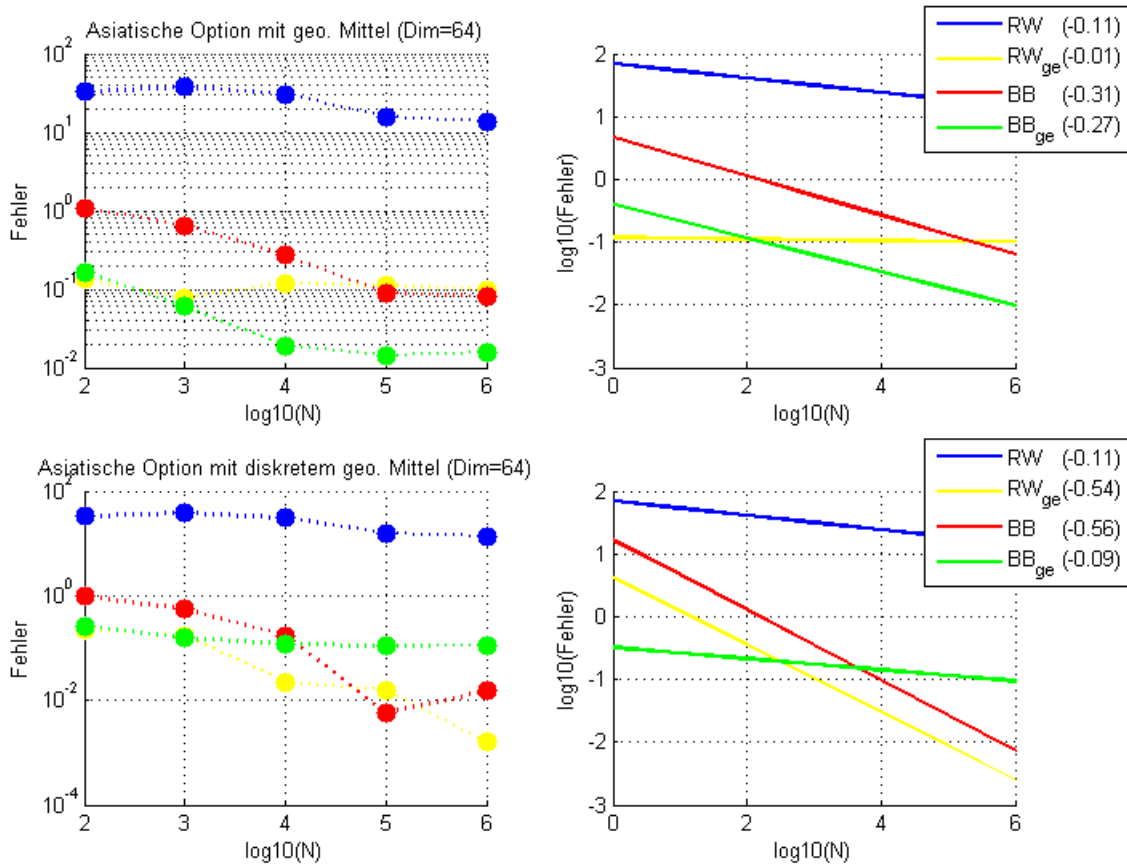
D=16	N	101	1009	10007	100003	1000003	Konvr.
RW	ungew.	5,66923E+00	4,56479E+00	1,32443E+00	5,81782E-01	3,22514E-01	-0,3389
	gew.	4,57941E-02	1,81625E-03	6,19458E-03	2,03720E-03	8,58918E-04	-0,3409
BB	ungew.	1,09776E-01	1,70798E-01	3,78967E-01	4,19881E-01	4,23999E-01	0,1566
	gew.	5,53387E-01	4,60175E-01	4,52083E-01	4,45907E-01	4,46679E-01	-0,0200



Die Gewichte für $D = 64$ sind gegeben durch $\gamma_j = 0,098 * 0.98^j$.

D=64	N	101	1009	10007	100003	1000003	Konvr.
RW	ungew.	3,30757E+01	3,74965E+01	3,05067E+01	1,58073E+01	1,36426E+01	-0,1146
	gew.	1,35088E-01	7,79217E-02	1,18866E-01	1,12016E-01	9,78536E-02	-0,0123
BB	ungew.	1,08742E+00	6,50346E-01	2,69712E-01	9,06042E-02	8,08926E-02	-0,3117
	gew.	1,67394E-01	6,06878E-02	1,91780E-02	1,45281E-02	1,56770E-02	-0,2681

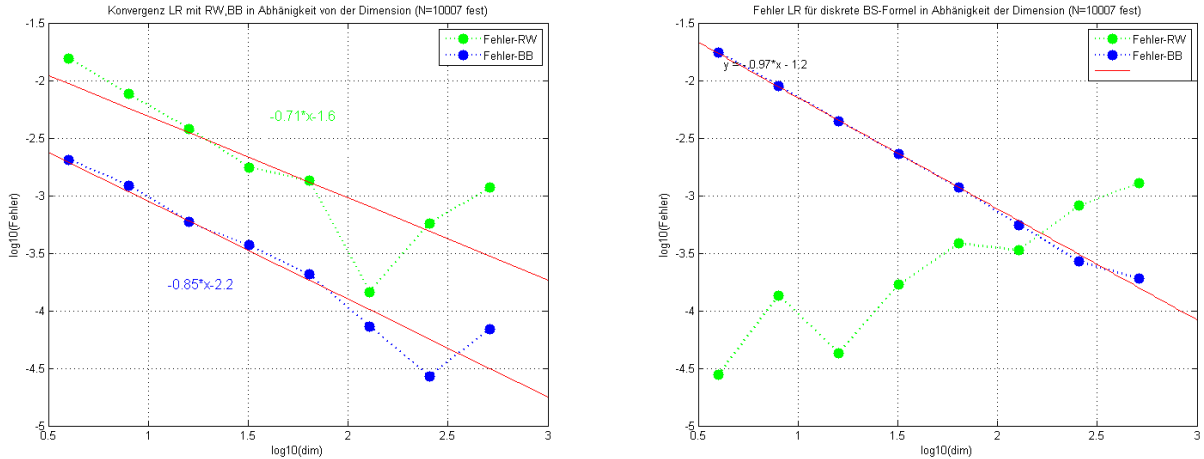
D=64	N	101	1009	10007	100003	1000003	Konvr.
RW	ungew.	3,29794E+01	3,74002E+01	3,04104E+01	1,57110E+01	1,35464E+01	-0,1150
	gew.	2,31369E-01	1,74202E-01	2,25854E-02	1,57353E-02	1,57284E-03	-0,5379
BB	ungew.	9,91139E-01	5,54065E-01	1,73431E-01	5,67654E-03	1,53882E-02	-0,5607
	gew.	2,63674E-01	1,56969E-01	1,15459E-01	1,10809E-01	1,11958E-01	-0,0895



Betrachtet man nur die Konvergenzen gegen das stetige Ergebnis, so fällt auf, dass der Fehler der ungewichteten RW-Konstruktion für beide Dimensionen immer größer ist als bei der gewichteten RW-Konstruktion, diese aber keinerlei Konvergenzverhalten aufweist. Außerdem ist zu sehen, dass die Konstruktionen mittels BB bessere Konvergenzeigenschaften aufweisen als die RW-Konstruktionen. Speziell für den Fall mit Dimension 64 ist gut zu erkennen, dass die gewichtete BB Konstruktion eindeutig die besten Ergebnisse liefert. Interessant ist nun der Vergleich mit den diskreten Ergebnissen. Wie bereits bei den stetigen Beobachtungen angemerkt, liegt der Fehler der gewichteten RW-Konstruktion signifikant unter dem der ungewichteten, mit dem Unterschied, dass die gewichtete Konstruktion nun auch wirklich gegen das diskrete Ergebnis konvergiert. Für die Dimension 64 liegt die Konvergenzrate sogar über $1/2$. Für die BB Konstruktionen ist anzumerken, dass sie für die Dimension 16 noch gar nicht konvergieren, was sich dann aber nach Erhöhung der Zeitschritte ändert.

Insgesamt macht sich also auch bemerkbar, dass der Diskretisierungsfehler der verschiedenen Pfaddiskretisierungen stark voneinander abweicht. Um zu sehen, wie gut die Lattice-Regeln unter beiden Konstruktionsarten die lineare Konvergenz der diskreten gegen die stetige Black-Scholes-Formel nachbilden, wurde nun die Anzahl der Punkte bei $N = 10007$ festgehalten und die Dimension in jedem Schritt von 4 bis 512 verdoppelt. Die Gewichtung

gen des Funktionenraums sind gegeben durch $\gamma_j = 1/dim$. Außerdem wurden Start- und Strikepreis von 100 auf 1 heruntergesetzt, um Überlauf des Rechners zu vermeiden, also sind die Fehler jetzt auch genau 100 mal so klein, wie im vorangegangenen Beispiel.



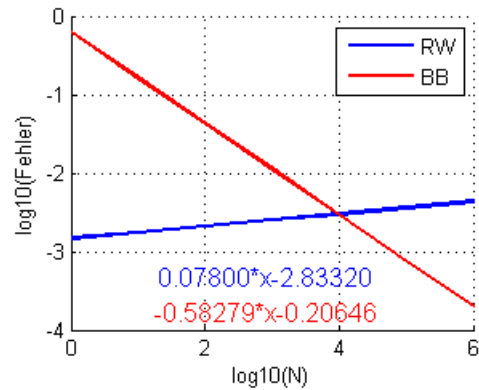
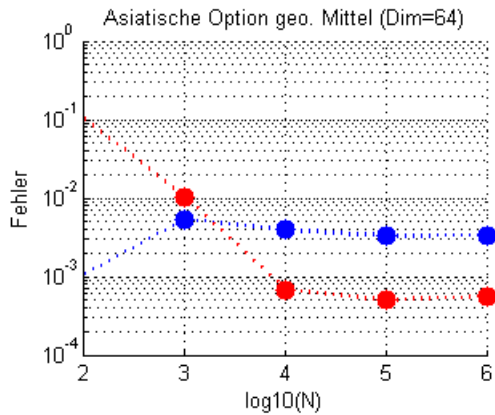
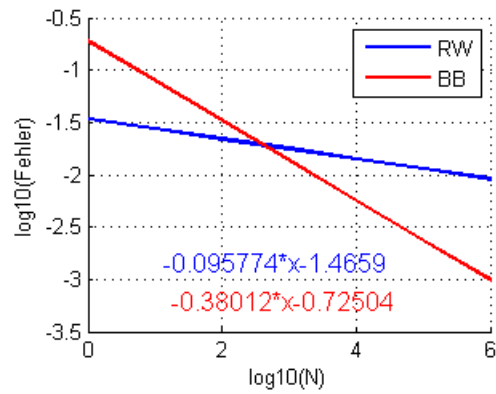
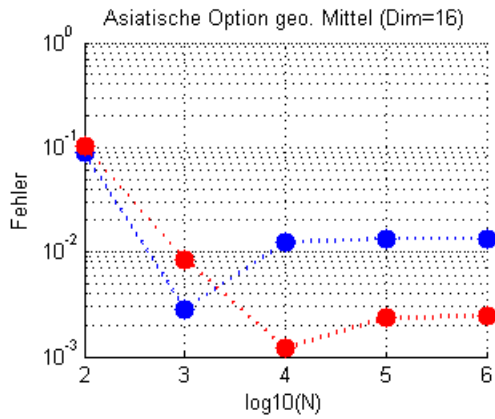
Wie auch in obigem Beispiel bereits festgestellt, konvergiert die BB-Konstruktion mit der Raten 0,85 besser als die Konstruktion durch den RW (0,71) mit dem Zusatz, dass der Fehler bei der BB-Konstruktion zu Beginn bereits kleiner ist. Außerdem sieht man im rechten Bild wie die Lattice-Regel mit BB die lineare Konvergenz fast eins zu eins (0,97) nachbildet, während die RW-Konstruktion für wenige Zeitschritte sehr gute Ergebnisse liefert, die mit zunehmender Dimension aber schlechter werden.

5.2 Weitere Ergebnisse

Um das im vorangegangenen Abschnitt angesprochene Problem der Zeitdiskretisierung und der damit verbundenen extrem hohen Dimension einzuschränken wurde die Optionslaufzeit für die folgenden Berechnungen von einem Jahr auf einen Monat verkürzt. Außerdem wurde der Strikepreis verringert, um den 'glatten Anteil' des Integranden zu vergrößern. Die Gewichte des Korobov-Raums sind wieder gegeben durch $\gamma_j = 1/dim$.

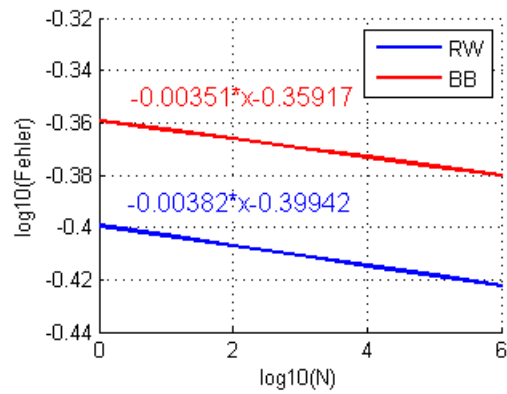
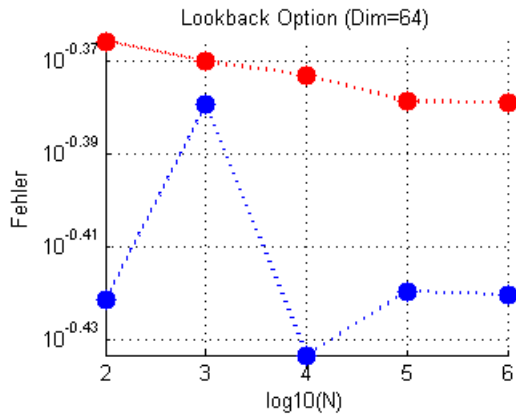
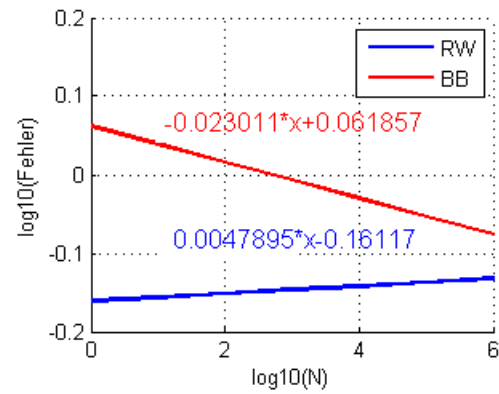
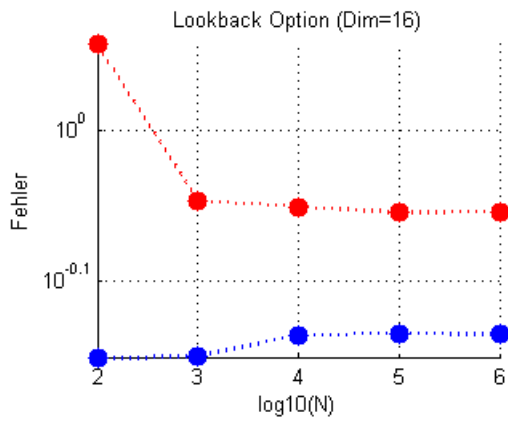
- Asiatische (geo.) Option : $S_0 = 100$, $K = 90$, $T = 1/12$, $\sigma = 0.2$, $r = 0,05$

N		101	1009	10007	100003	1000003	Konvr.
D=16	RW	8,79144E-02	2,87494E-03	1,23311E-02	1,34859E-02	1,35021E-02	-0,0958
	BB	1,01781E-01	8,43285E-03	1,19570E-03	2,35555E-03	2,43132E-03	-0,3801
D=64	RW	1,06427E-03	5,39713E-03	3,91875E-03	3,30789E-03	3,33310E-03	0,0780
	BB	1,01376E-01	1,03536E-02	6,80217E-04	5,02924E-04	5,64625E-04	-0,5828



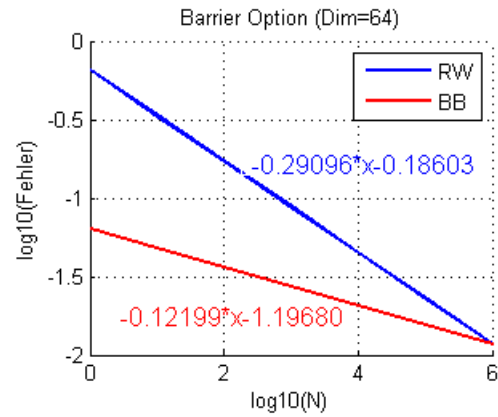
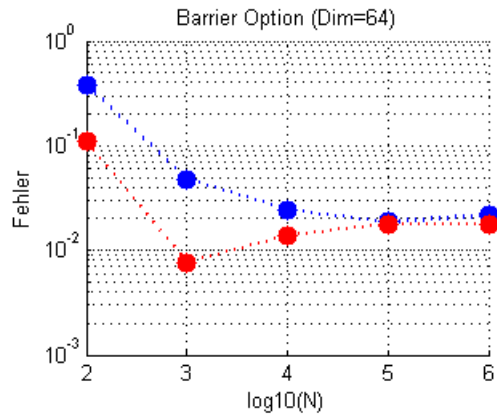
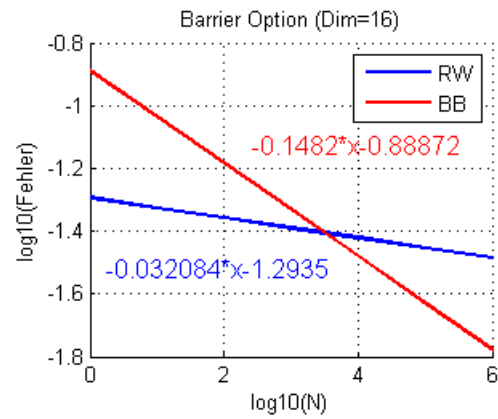
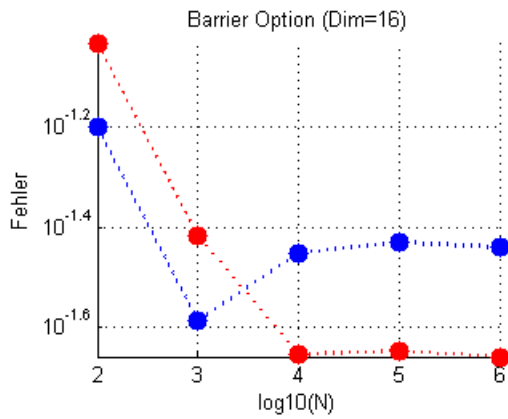
- *Lookback Option (var. Strike)* : $S_0 = 100$, $T = 1/12$, $\sigma = 0.2$, $r = 0,05$

N		101	1009	10007	100003	1000003	Konvr.
D=16	RW	7,04854E-01	7,06639E-01	7,30613E-01	7,32326E-01	7,31595E-01	0,0048
	BB	1,13882E+00	8,97478E-01	8,88654E-01	8,81482E-01	8,81944E-01	-0,0230
D=64	RW	3,79050E-01	4,17636E-01	3,68775E-01	3,80643E-01	3,79961E-01	-0,0038
	BB	4,30940E-01	4,26533E-01	4,23736E-01	4,18180E-01	4,18029E-01	-0,0035



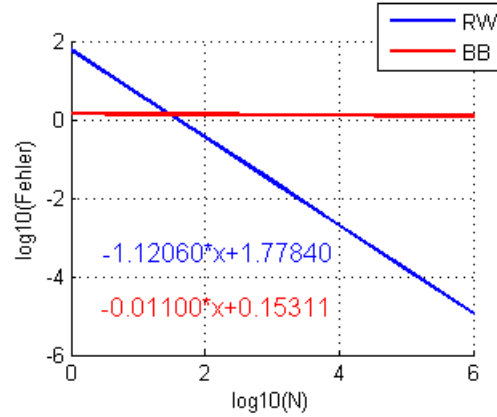
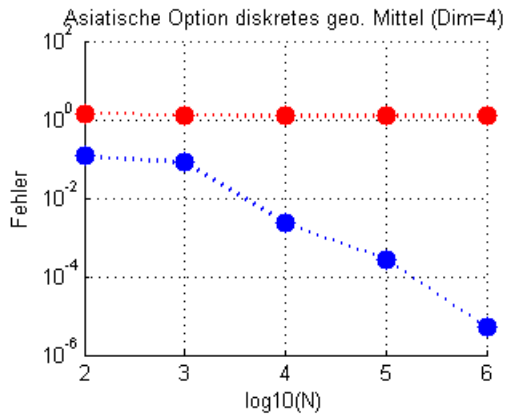
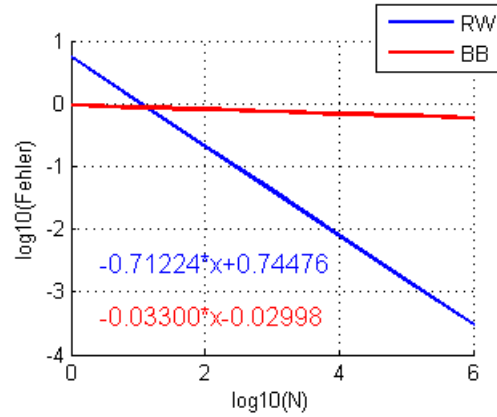
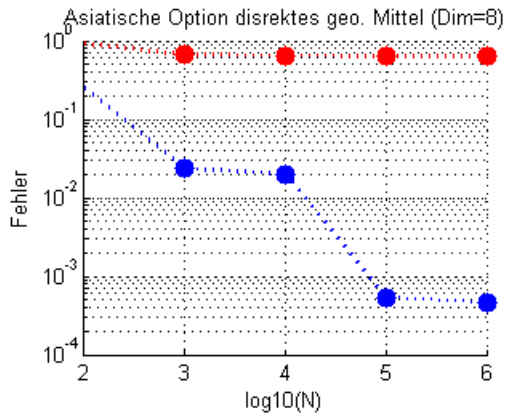
- *Barrier Option* : $S_0 = 100$, $K = 90$, $B = 90$, $T = 1/12$, $\sigma = 0.2$, $r = 0,05$

N		101	1009	10007	100003	1000003	Konvr.
D=16	RW	6,29665E-02	2,58688E-02	3,53246E-02	3,71597E-02	3,63320E-02	-0,0321
	BB	9,29087E-02	3,81766E-02	2,21456E-02	2,26117E-02	2,19558E-02	-0,1482
D=64	RW	3,82981E-01	4,74693E-02	2,42754E-02	1,86403E-02	2,15265E-02	-0,2910
	BB	1,10325E-01	7,71995E-03	1,39165E-02	1,77133E-02	1,79137E-02	-0,1220



- Asiatische diskrete (geo.) Option : $S_0 = 100$, $K = 90$, $T = 1$, $\sigma = 0.2$, $r = 0,05$

N		101	1009	10007	100003	1000003	Konvr.
D=8	RW	2,51162E-01	2,36616E-02	2,01452E-02	5,35382E-04	4,63584E-04	-0,7122
	BB	9,13767E-01	6,55468E-01	6,40523E-01	6,35480E-01	6,34956E-01	-0,0330
D=4	RW	1,15340E-01	8,06735E-02	2,28867E-03	2,79173E-04	4,97183E-06	-1,1206
	BB	1,40186E+00	1,27992E+00	1,25229E+00	1,25012E+00	1,24990E+00	-0,0110



Anmerkungen

Das mit Abstand beste Resultat ist die Bewertung der diskreten asiatischen Option mit wenigen (8, 4) Zeitschritten. Hier liefern die Lattice-Regeln mit der RW-Konstruktion sehr genaue Ergebnisse und die Konvergenz gegen das exakte Ergebnis ist wirklich annähernd linear. Die Konvergenz der Regeln gegen die stetige asiatische Option verbessert sich (wie im vorherigen Abschnitt bereits gesehen) für die BB-Konstruktion mit steigender Dimension enorm. Für $D = 64$ kann die Konvergenz dieser Konstruktion auch mit der Rate $1/2$ der gewöhnlichen Monte Carlo Simulation mithalten. Bei den Barrier und Lookback Optionen fällt wahrscheinlich die Zeitdiskretisierung noch stärker ins Gewicht, da Extremwerte der gesamten Optionslaufzeit in die Bewertung miteinbezogen werden. Für Barrier Optionen liefern die Lattice-Regeln zwar einigermaßen akzeptable Ergebnisse, die Rate der Konvergenz ist aber für alle Settings eher unbefriedigend. Im Falle der Lookback Option versagen alle Konstruktionen. Für keins der unterschiedlichen Settings kommt eine signifikante Konvergenz zustande und der Fehler ist nie kleiner als E-01.

6 Fazit

Als Fazit meiner Bachelorarbeit lässt sich festhalten, dass Lattice-Regeln durchaus eine effektive Methode zur Bewertung pfadabhängiger Finanzderivate darstellen. Allerdings erfordert speziell die Wahl der richtigen Gewichte des Funktionenraumes eine detaillierte Analyse des Integranden, die die Komplexität des Ausgangsproblems schnell überschreitet und im Umfang dieser Arbeit nur kurz angesprochen wurde, aber nicht tiefergehend umgesetzt werden konnte. Trotzdem hat sich gezeigt, dass beispielsweise asiatische Optionen gut zu bewerten sind. Für passende Ausgangsdaten lassen sich speziell bei diskreter Mittelbildung auch mit relativ einfachen Mitteln gute Konvergenzraten erreichen, die mit gewöhnlichen Monte Carlo-Verfahren mithalten können oder diese sogar übertreffen. Das größte Problem besteht in der Zeitdiskretisierung und dem resultierenden Fehler, denn die Ergebnisse zeigen auch, dass mit zunehmender Dimension der logarithmische Term $(\log N)^{\beta(s,\alpha)}$ der Fehlerschranke in der Praxis nicht mehr vernachlässigt werden kann. Als zusätzliche Verbesserungsmöglichkeiten kann statt RW und BB die PCA (principal component analysis) als Diskretisierungsverfahren genutzt werden, da diese die abfallende Relevanz der Variablen noch besser nachbildet. Außerdem können verschiedene Varianzreduktionstechniken angewendet werden, um die Anzahl der wichtigen (und somit für die Diskretisierung benötigten) Variablen möglichst niedrig zu halten. Zusammengefasst ist das beschriebene Quasi-Monte-Carlo-Verfahren auch für die Praxis ein Ansatz mit Aussicht auf eine Verbesserung der Konvergenzrate des gewöhnlichen Monte Carlo Ansatzes. Allerdings ist noch weitere Forschung von Nöten, um Lattice-Regeln robuster zu machen, damit sie effektiv auf eine breitere Masse von Optionstypen oder anderen Finanzderivaten angewendet werden können.

7 Codes

7.1 *asianopt_lattice_bb.m*

```
%Approximation des fairen Preises einer asiatischen Option mit geometrischem
%Mittel via QMC-Verfahren und 'guten Lattice Punkten' als Stützstellen erzeugt
%durch die schnelle komponentenweise Konstruktion 'fastrankipt.m' von Dirk Nuyens
%Pfaddirktretisierung durch die Brownsche Brücke und Dimensionsgewichtung
%durch die 2 Parameter-Form

function[call]=asianopt_lattice_bb(n,s_max,T,K,S_0,sigma,r,a,gamma)

    h=T/s_max; %relative Schrittweite

    omega = inline('2*pi^2*(x.^2-x+1/6)'); %Input für fastrankipt.m
    beta=ones(s_max,1);
    gamma = ones(s_max, 1);
    for i=1:s_max, %2-Parameterform der Gewichte
        gamma(i,1)=a*gamma^i;
    end

    z = fastrankipt(n, s_max, omega, gamma, beta); %Erzeugt z-Vektor für Dim s_max und n Punkte

    lattice=zeros(n,s_max);
    for i=1:n,
        for j=1:s_max, %Erzeugt die Gitterpunkte der Lattice Regel
            lattice(i,j)=mod((i/n)*z(j,1),1);
        end
    end

    call=zeros(1,n); %Vektor zum Speichern jedes Simulationsdurchlaufs
    for i=1:n,
        St=zeros(1,s_max); %Vektor zum Speichern der Aktienkurse
        W=zeros(1,s_max+1); %Diskretisierung der Wiener Prozesses mit BB
        W(1,s_max+1)=sqrt(T)*norminv(lattice(i,1));
        k=2; m=2;
        while k<=s_max,
            dt=s_max/k;
            for l=0:(k/2-1),
                W(1,dt+l*2*dt+1)=(W(1,l*2*dt+1)+W(1,l*2*dt+1+2*dt))/2+...
                    sqrt(dt*h/2)*norminv(lattice(i,m));
                m=m+1;
            end
            k=k*2;
        end
        for j=1:s_max, %Simulation der Aktienkurse in t_j
            St(1,j)=S_0*exp((r-((sigma)^2)/2)*(j*h) + sigma*W(1,j));
        end
        call(1,i)=max(prod(St)^(1/s_max)-K,0); %Optionswert für den i-ten Gitterpunkt
    end
    call=exp(-r*T)*(1/n)*sum(call); %Mittelung der einzelnen Iterationen
end
```


7.2 *fastrank1pt.m*

```

% Eingabeparameter
% n      Anzahl der Punkte der Lattice Regel (ganzzahlig, prim)
% s_max  Anzahl der Dimensionen (ganzzahlig)
% omega  Eine Funktion die den variablen Teil des verschiebungsinvarianten
%        Kerns nachbildet (als symmetrisch angenommen)
% gamma  Gewichte der Dimensionen des Funktionenraums (Vektor [s_max 1])
% beta   Betaparameter einer noch allgemeineren (im der Arbeit nicht vorgestellten)
%        Darstellung des worst case errors (Vektor [s_max x 1])
%        (für den eingeführten Korobovraum gilt beta=1)
% Ausgabeparameter
% z      Erzeugendenvektor der Lattice Regel (Vektor [s_max x 1])
% e2     Optimaler quadratischer Fehler für jede Dimension (Vektor [s_max x 1])
%
% z.B.
% Konstruktion einer Lattice Regel im Korobovraum mit alpha=2, gamma=1/s_max, beta=1
% n=... ; s_max=... ;
% omega = inline('2*pi^2*(x.^2-x+1/6)');
% gamma = ones(s_max, 1) / s_max;
% beta = ones(s_max, 1);
% (C) 2004, <dirk.nuyens@cs.kuleuven.ac.be>
function [z, e2] = fastrank1pt(n, s_max, omega, gamma, beta)

if ~isprime(n), error('n must be prime'); end;
z = zeros(s_max, 1);
e2 = zeros(s_max, 1);

m = (n-1)/2; % Angenommene Symmetrie der Kernfunktion um 1/2
E2 = zeros(m, 1); % Vektor der die Werte des Fehlermaßes enthält
cumbeta = cumprod(beta);

g = generatorp(n); % Berechnung der Primitivwurtel von Z_n
perm = zeros(m, 1); % Permutation von Z_n erzeugt durch Potenzen von g
perm(1) = 1; for j=1:m-1, perm(j+1) = mod(perm(j)*g, n); end;
perm = min(n - perm, perm); % Zurückabbilden des Vektors auf das Intervall [1, n/2)
psi = omega(perm/n); % Der Vektor psi
psi0 = omega(0); % Nullindex von psi
fft_psi = fft(psi); % Schnelle Fourier-Transformation auf psi angewendet

q = ones(m, 1); % Permutierter Produktvektor q
q0 = 1; % Nullindex des permutierten Produktvektors

for s = 1:s_max
    % Schritt 2a: zyklische Matrix-Vektor Multiplikation
    E2 = ifft(fft_psi .* fft(q));
    E2 = real(E2); % Entfernen des imaginären Rundungsfehlers
    % Schritt 2b: Wähle w_s und z_s die den kleinsten Wert ergeben
    [min_E2, w] = min(E2); % Wähle den Index mit minimalem Wert aus
    if s == 1, w = 1; noise = abs(E2(1) - min_E2); end;
    z(s) = perm(w);
    % Zusatz: Angabe des exakten worst-case errors
    e2(s) = -cumbeta(s) + (beta(s) * (q0 + 2*sum(q)) + ...
        gamma(s) * (psi0*q0 + 2*min_E2)) / n;
    % Schritt 2c: Update des Vektors q
    q = (beta(s) + gamma(s) * psi([w:-1:1 m:-1:w+1])) .* q;
    q0 = (beta(s) + gamma(s) * psi0) * q0;
    fprintf('s=%4d, z=%6d, w=%6d, e2=%4e, e=%4e\n', ...
        s, z(s), w, e2(s), sqrt(e2(s)));
end;

```

A Literaturverzeichnis

1. Christiane Lemieux and Pierre L'Ecuyer: A Comparison of Monte Carlo, Lattice Rules and Other Low-Discrepancy Point Sets, 1999.
2. D. Nuyens and R. Cools: Fast algorithms for component-by-component construction of rank-1 lattice rules in shift invariant reproducing kernel Hilbert spaces, Report TW392, Dept. of Computer Science, K.U.Leuven, 2004.
3. D. Nuyens: Ph.D.Thesis, Fast construction of good lattice rules, Dept. of Computer Science, K.U.Leuven, 2007.
4. I.H. Sloan and A.V. Reztsov: Component-by-component construction of good lattice rules, Appl. Math. Rep. AMR00/8, 2000.
5. I. H. Sloan: Lattice Methods for Multiple Integration, Oxford University Press, 1994.
6. I. H. Sloan and H. Woźniakowski. When are quasi-Monte Carlo algorithms efficient for high dimensional integrals, J. Complexity, 1998.
7. I. M. Sobol: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates, Math. Comp. Simulation, 2001,
8. J. Dick: On the convergence rate of the component-by-component construction of good lattice rules, Appl. Math. Rep. AMR03/21, 2003.
9. J. Dick, I. H. Sloan, X. Wang and H. Woźniakowski: Good lattice rules in weighted Korobov spaces with general weights, Appl. Math. Rep. AMR03/24, 2005.
10. J. C. Hull: Options, Futures, and Other Derivatives, 5th ed. Prentice-Hall, NJ, 2003.
11. T. Gerstner: Vorlesungsskript Computational Finance, Goethe-Universität Frankfurt am Main, Sommersemester 2010.
12. T. Gerstner: Vorlesungsskript Monte Carlo Verfahren, Goethe-Universität Frankfurt am Main, Sommersemester 2010.
13. T. Gerstner: Vorlesungsskript Numerische Mathematik, Goethe-Universität Frankfurt am Main, Wintersemester 2009/2010.
14. Xiaoqun Wang and Ian H. Sloan: Why are high-dimensional finance problems often of low effective dimension?, SIAM J. Sic. Comput., 2005.
15. Xiaoqun Wang and Ian H. Sloan: Efficient Weighted Lattice Rules with Applications to Finance, 2006.