



TECHNISCHE
UNIVERSITÄT
WIEN

Vienna University of Technology

Diplomarbeit

Analyse einer Least Squares Monte Carlo Methode zur Bewertung amerikanischer Optionen

ausgeführt am
Institut für Wirtschaftsmathematik
der Technischen Universität Wien

betreut von
Privatdoz. Dipl.-Ing. Dr.techn. Stefan Gerhold

Verfasser:
Paul Brandstätter, BSc
Matr. Nr. 0725686
Wien, am 22. November 2012

Zusammenfassung

Diese Arbeit untersucht den von Longstaff und Schwartz vorgeschlagenen Monte-Carlo Algorithmus (LSM Algorithmus) zur Bewertung von amerikanischen Optionen. Nach der Vorstellung des Algorithmus und Konvergenzüberlegungen dazu wird dieser in Maple 15 implementiert. Damit kann die Auswirkung unterschiedlicher Parameter und Ansatzfunktionen auf den berechneten Preis untersucht werden. Anschließend wird das Konzept des kritischen Preises vorgestellt. Nach einer Analyse des dazugehörigen Konzepts des Continuation Values wird eine Verbesserung der Berechnung des kritischen Preises und des Optionspreises für Polynome 2. Ordnung vorgeschlagen. Abschließend werden vom LSM Algorithmus errechnete Preise in einem Varianz-Gamma Modell mit Marktpreisen verglichen.

Abstract

This thesis takes a closer look at the least squares Monte Carlo algorithm (LSM algorithm) proposed by Longstaff and Schwartz for pricing American options. After introducing the algorithm and some considerations regarding convergence, the algorithm is implemented in Maple 15. Using this, the influence of different parameters and basis functions to the calculated price is examined. The related concept of the critical price is then introduced. The analysis of the concept of the continuation value is followed by a proposition to enhance the calculation of an option's critical price. Finally a variance gamma framework is set up and used to compare calculated prices with market prices.

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbstständig und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Wien, am 22. November 2012 _____
Paul Brandstätter

Inhaltsverzeichnis

Inhaltsverzeichnis	1
1 Einleitung	2
2 Der Least Squares Monte Carlo Algorithmus	3
2.1 Ein kleines Rechenbeispiel	3
2.2 Mathematische Grundlagen	7
2.3 Konvergenz nach Clément	10
2.4 Konvergenzordnung	15
3 Implementierung des LSM Algorithmus in Maple	16
3.1 Vorbereitende Prozeduren	16
3.2 Der LSM Algorithmus	20
3.3 Zurück zum einführenden Rechenbeispiel	22
3.4 Vergleich verschiedener Ansatzfunktionen	27
3.5 Early Exercise Premium	30
3.6 Merton's Jump Diffusion als Modell für den Aktienpreis	32
4 Kritischer Stockpreis einer amerikanischen Put-Option	36
4.1 Theoretische Grundlagen	36
4.2 Approximationen in geschlossener Form	38
4.2.1 Approximation nach Barles	39
4.2.2 Approximation nach Kuske und Keller	39
4.3 Numerische Berechnung	41
4.4 Adaptionen am LSM Algorithmus	52
5 Kalibrierung an Marktdaten in einem Varianz-Gamma Aktienpreismodell	56
5.1 Das Varianz-Gamma Modell	56
5.2 Vergleich von errechneten Preisen mit Marktpreisen	61
Literaturverzeichnis	66
Abbildungsverzeichnis	67
Tabellenverzeichnis	69

Kapitel 1

Einleitung

Eines der zentralen Probleme in der Finanzmathematik ist die Bewertung von Optionen. Für europäische Optionen veröffentlichten Fischer Black, Myron Samuel Scholes und Robert C. Merton 1973 ihre bekannte Formel, die unter sehr rigiden Annahmen eine explizite Lösung der partiellen Differentialgleichung des Optionspreises liefert. Für amerikanische Optionen existieren keine expliziten Lösungen in geschlossener Form, sodass die Optionspreise numerisch berechnet werden müssen. Eine Möglichkeit, dies zu tun, ist der Least Squares Monte Carlo (LSM) Algorithmus, der von Francis A. Longstaff und Eduardo S. Schwartz 2001 vorgestellt wurde.

Least Squares Monte Carlo Methoden werden dazu verwendet, eine unbekannte Funktion, abhängig von einer oder mehreren Variablen, durch Simulation und Regression zu schätzen. In unserem Fall soll damit der Continuation Value, also der Barwert des Nichtausübens der amerikanischen Option zu einem Zeitpunkt t , geschätzt werden. Wir simulieren dazu eine Anzahl N an möglichen Verläufen des zugrundeliegenden Basistitels und werden eine polynomiale Regression verwenden, um den Continuation Value als Funktion vom Preis des Basistitels dazustellen. Ein Vergleich dieses Barwertes mit dem Wert des sofortigen Ausübens lässt uns die Option bewerten.

In Kapitel 2 wird zuerst der LSM Algorithmus anhand eines übersichtlichen Zahlenbeispiels vorgestellt. Anschließend werden das mathematische Rahmenwerk und Konvergenzüberlegungen ein solides Fundament für den Einsatz des Algorithmus bilden. Kapitel 3 widmet sich der Implementierung des LSM Algorithmus in Maple. Der Algorithmus und seine zugrundeliegende Idee können so besser verstanden werden. Wir wollen mit verschiedenen Ansatzfunktionen und Modellen für den Aktienpreis experimentieren und die Ergebnisse vergleichen und analysieren. Die gewonnenen Einsichten werden unter anderem in Kapitel 4 weiter verarbeitet und dienen dem Verständnis des dort vorgestellten Konzepts des kritischen Preises, mittels dessen wir eine Strategie zum optimalen Ausüben einer amerikanischen Option erhalten. In Kapitel 5 widmen wir uns dem praktischen Einsatz des LSM Algorithmus zur Bepreisung von tatsächlich gehandelten Optionen in einem Varianz-Gamma Modell für den zugrundeliegenden Basistitel. Die errechneten Preise werden mit den Kursen verglichen um ein Gefühl für die Qualität des Algorithmus und seine Alltagstauglichkeit zu erhalten.

Kapitel 2

Der Least Squares Monte Carlo Algorithmus

Eine amerikanische Put Option gibt dem Käufer das Recht, aber nicht die Pflicht, bis zur Maturity T der Option den zugrunde liegenden Titel S_t zu einem im Vorhinein festgelegten Preis K zu verkaufen. Im Unterschied zur europäischen Option hat der Käufer also theoretisch unendlich viele Möglichkeiten, die Option auszuüben. Um mittels numerischer Methoden zu einem Preis für die Option zu gelangen, muss dieses Zeitintervall $[0, T]$ diskretisiert werden, es gibt also nur endlich viele mögliche Ausübungszeitpunkte.

Für das gegebene Zeitintervall werden mögliche Pfade des zugrundeliegenden Titels simuliert. Die grundlegende Idee hinter dem LSM Algorithmus ist nun, zu jedem Zeitpunkt den Continuation Value, also den Wert des Beibehaltens der Option, als Regressionsfunktion zu schätzen und mit dem Payoff des sofortigen Ausübens zu vergleichen. Dafür wird in der Zeit rückwärts gegangen und zu jedem Zeitpunkt eine Regression berechnet. Dadurch fließt in jeden Pfad auch die Information aus den übrigen Pfaden mit ein. Der Algorithmus ist anhand eines kleinen Rechenbeispiels leicht zu verstehen:

2.1 Ein kleines Rechenbeispiel¹

Zur Maturity T wird eine Option optimalerweise dann ausgeübt, wenn sie “in the money” ist, also ihr Payoff positiv ist. Zu einem früheren Zeitpunkt hingegen muss der Wert des sofortigen Ausübens mit dem Wert des Beibehaltens (Continuation Value) verglichen werden. Die Berechnung dieses Continuation Values ist also die zentrale Aufgabe des LSM Algorithmus. Dabei wird der bedingte Erwartungswert, der den Barwert des Beibehaltens der Option angibt, durch ein Regressionspolynom approximiert und somit die Information aus allen Pfaden genutzt.

In diesem Beispiel wird eine amerikanische Put Option mit folgenden Parametern betrachtet: Der Strikepreis beträgt 1.10, mögliche Ausübungszeitpunkte sind $\{1, 2, 3 = T\}$ und die risikolose Zinsrate beträgt 6%. Für die Berechnung

¹vgl. [LS], Kapitel 1

werden die folgenden acht möglichen Pfade einer Aktie mit $S_0 = 1$ herangezogen, wobei diese unter dem risikoneutralen Maß erzeugt wurden.

Pfad	t=0	t=1	t=2	t=3
1	1.00	1.09	1.08	1.34
2	1.00	1.16	1.26	1.54
3	1.00	1.22	1.07	1.03
4	1.00	0.93	0.97	0.92
5	1.00	1.11	1.56	1.52
6	1.00	0.76	0.77	0.90
7	1.00	0.92	0.84	1.01
8	1.00	0.88	1.22	1.34

Tabelle 2.1: zugrunde liegende Pfade

Der Algorithmus verfährt nun rekursiv. Bedingt darauf, dass die Option nicht zuvor schon ausgeübt wurde, ergibt sich für den Zeitpunkt $t = 3$ folgende Auszahlungsmatrix mithilfe der Payoff Funktion $\max(1.10 - S_t, 0)$:

Pfad	t=0	t=1	t=2	t=3
1	-	-	-	0.00
2	-	-	-	0.00
3	-	-	-	0.07
4	-	-	-	0.18
5	-	-	-	0.00
6	-	-	-	0.20
7	-	-	-	0.09
8	-	-	-	0.00

Tabelle 2.2: Auszahlungsmatrix zum Zeitpunkt $t = 3$

Zum Zeitpunkt $t = 2$ muss der Halter der Option nun entscheiden, ob er die Option weiter behält oder ob er sie ausübt. Zu $t = 2$ gibt es nur fünf Pfade, bei denen die Option in the money ist. Mit X sei nun der Aktienpreis für diese fünf Pfade bezeichnet, Y sei der zugehörige diskontierte Cashflow zu $t = 3$ im Falle des Nichtausübens. Für den LSM Algorithmus werden nur die Pfade verwendet, für die die Option in the money ist. Die Vektoren X und Y ergeben sich somit wie folgt:

Um nun den erwarteten Cashflow des Beibehaltens der Option zu schätzen, wird Y auf eine Konstante, X und X^2 regressiert. Damit ergibt sich als bedingte Erwartungswertfunktion $E[Y|X] = -1.070 + 2.983 \cdot X - 1.813 \cdot X^2$. Es sei hier angemerkt, dass natürlich auch andere Basisfunktionen für die Regression herangezogen werden können. Darauf wird später in dieser Arbeit noch eingegangen.

Mit der eben berechneten Funktion kann nun der Wert des Beibehaltens mit dem Wert des Ausübens verglichen werden. Dazu wird die eben berechnete Regressionsfunktion auf die Aktienpreise zu $t = 2$, die in the money sind, angewendet. Der Wert des Ausübens ergibt sich als $\max(1.10 - S_2, 0)$. (Tabelle 2.4)

Pfad	Y	X
1	$0.00 \cdot \exp(-0.06)$	1.08
2	-	-
3	$0.07 \cdot \exp(-0.06)$	1.07
4	$0.18 \cdot \exp(-0.06)$	0.97
5	-	-
6	$0.20 \cdot \exp(-0.06)$	0.77
7	$0.09 \cdot \exp(-0.06)$	0.84
8	-	-

Tabelle 2.3: Regression zum Zeitpunkt $t = 2$

Pfad	Ausüben	Beibehalten
1	0.02	0.0369
2	-	-
3	0.03	0.0461
4	0.13	0.1176
5	-	-
6	0.33	0.1520
7	0.26	0.1565
8	-	-

Tabelle 2.4: Optimales Ausüben zu $t = 2$

Die Option wird zum Zeitpunkt $t = 2$ also im Pfad vier, sechs und sieben ausgeübt. Damit ergibt sich wieder eine Cashflow Matrix für den Zeitpunkt $t = 2$ bedingt darauf, dass bis $t = 2$ nicht ausgeübt wurde. (Tabelle 2.5)

Pfad	t=0	t=1	t=2	t=3
1	-	-	0.00	0.00
2	-	-	0.00	0.00
3	-	-	0.00	0.07
4	-	-	0.13	0.00
5	-	-	0.00	0.00
6	-	-	0.33	0.00
7	-	-	0.26	0.00
8	-	-	0.00	0.00

Tabelle 2.5: Auszahlungsmatrix zum Zeitpunkt $t = 2$

In den Fällen, wo zu $t = 2$ ausgeübt wurde, wird der Cashflow zu $t = 3$ natürlich zu 0. Im Pfad 1 liefert die optimale Ausübungsstrategie aus Tabelle 2.4, dass die Option beibehalten werden soll. Da aber der Pfad für den Optionshalter negativ verläuft, ergibt sich trotzdem kein Cashflow zu $t = 3$.

Wir befinden uns nun in einer ähnlichen Situation wie zu Beginn des Algorithmus und verfahren rekursiv weiter. Aus der Aktienpreismatrix in Tabelle 2.1 ergeben sich wieder fünf in the money Pfade für $t = 1$. Der Vektor Y beinhal-

tet wieder die diskontierten Cashflows der Option. Zu beachten ist dabei, dass hierfür die tatsächlichen Cashflows für jeden Pfad verwendet werden und nicht die geschätzten Cashflows aus der vorangegangenen Regression.

Da die Option nur einmal ausgeübt werden kann, können zukünftige Cashflows nur entweder zu $t = 2$ oder $t = 3$ auftreten. In jedem Fall müssen sie aber korrekt diskontiert werden. Zu $t = 3$ ergibt sich nur für den Pfad drei ein Cashflow. Da aber der Pfad drei zu $t = 1$ nicht in the money ist, wird in diesem Beispiel keine Auszahlung über zwei Perioden diskontiert. Für die Regressionsvektoren X und Y ergeben die Werte aus Tabelle 2.6.

Pfad	Y	X
1	$0.00 \cdot \exp(-0.06)$	1.09
2	-	-
3	-	-
4	$0.13 \cdot \exp(-0.06)$	0.93
5	-	-
6	$0.33 \cdot \exp(-0.06)$	0.76
7	$0.26 \cdot \exp(-0.06)$	0.92
8	$0.00 \cdot \exp(-0.06)$	0.88

Tabelle 2.6: Regression zum Zeitpunkt $t = 1$

Eine Regression mit den gleichen Basisfunktionen wie zu $t = 2$ liefert die bedingte $\mathbb{E}[Y|X] = 2.038 - 3.335 \cdot X + 1.356 \cdot X^2$. Angewandt auf die relevanten Aktienpreise zu $t = 1$ ergibt sich die optimale Ausübungsstrategie zu $t = 1$ aus Tabelle 2.7.

Pfad	Ausüben	Beibehalten
1	0.01	0.0139
2	-	-
3	-	-
4	0.17	0.1092
5	-	-
6	0.34	0.2866
7	0.18	0.1175
8	0.22	0.1533

Tabelle 2.7: Optimales Ausüben zu $t = 1$

Zu $t = 1$ werden somit die Pfade vier, sechs, sieben und acht ausgeübt.

Für alle möglichen Ausübungszeitpunkte wurde also eine Entscheidungsregel gefunden, die in Tabelle 2.8 dargestellt wird.

Wird die Option nach diesen Stopregeln ausgeübt, ergeben sich die Cashflows in Tabelle 2.9

Um nun die Option zum Zeitpunkt $t = 0$ zu bewerten, müssen nur noch alle Cashflows auf den Zeitpunkt 0 diskontiert und anschließend gemittelt werden. Der Wert dieser amerikanischen Put Option beträgt hier 0.1144. Der Wert des zugehörigen europäischen Puts ergibt sich durch Diskontieren und Mitteln der Cashflows zu $t = 3$, in diesem Beispiel 0.0564. Die Prämie, die für das Recht des

Pfad	t=1	t=2	t=3
1	0	0	0
2	0	0	0
3	0	0	1
4	1	0	0
5	0	0	0
6	1	0	0
7	1	0	0
8	1	0	0

Tabelle 2.8: Stoppzeiten

Pfad	t=1	t=2	t=3
1	0.00	0.00	0.00
2	0.00	0.00	0.00
3	0.00	0.00	0.07
4	0.17	0.00	0.00
5	0.00	0.00	0.00
6	0.34	0.00	0.00
7	0.18	0.00	0.00
8	0.22	0.00	0.00

Tabelle 2.9: Cashflows der Option

vorzeitigen Ausübens bezahlt werden muss (early exercise premium) ist somit 0.4496.

2.2 Mathematische Grundlagen²

Wir betrachten einen Maßraum $(\Omega, \mathcal{F}, \mathbb{P})$ und einen endlichen Zeithorizont $[0, T]$. Ω ist dabei die Grundmenge und $\omega \in \Omega$ bestimmt einen zufälligen Pfad. \mathcal{F} ist eine σ -Algebra auf Ω zur Zeit T und \mathbb{P} ein Wahrscheinlichkeitsmaß auf \mathcal{F} . Weiters bezeichne $\mathbb{F} = \{\mathcal{F}_t : t \in [0, T]\}$ die um Nullmengen erweiterte natürliche Filtration bezüglich des Preisprozesses der Aktie, wobei $\mathcal{F}_T = \mathcal{F}$ ist. Weiters nehmen wir die Existenz eines äquivalenten Martingalmaßes \mathbb{Q} an.

Ganz allgemein können amerikanische Optionen betrachtet werden, die eine zufällige Auszahlung im Zeitintervall $[0, T]$ liefern. Für die Payoff Funktion $v : (t, x) \mapsto v(t, x)$ wird quadratische Integrierbarkeit gefordert, also $v \in L^2(\Omega, \mathcal{F}, \mathbb{Q})$. Im Falle eines Puts ist die Auszahlungsfunktion $v(t, S_t) = \max(K - S_t, 0)$, wobei S_t der Preis des Basistitels zur Zeit t ist.

Das Standardresultat zur Bepreisung von Amerikanischen Optionen stellt das Konzept der Snell Einhüllenden dar³. Der Wert einer amerikanischen Option wird dabei als Supremum der diskontierten Zahlungsströme dargestellt, wobei das Supremum über alle möglichen Stoppzeiten gebildet wird. In der Praxis ist es nicht möglich, all diese Stoppzeiten in Betracht zu ziehen, eine Approxima-

²vgl. [LS], Kapitel 2 sowie [S], Kapitel 2

³vgl. [K]

tion ist nötig. Wir beschränken uns daher auf L diskrete Ausübungszeitpunkte $0 < t_1 \leq t_2 \leq t_3 \leq \dots \leq t_L = T$. Optionen, die nur zu im Vorhinein bestimmten Zeitpunkten ausgeübt werden können, heißen auch Bermuda Optionen. Hier werden diese benutzt, um amerikanische Optionen zu approximieren.

Die Cashflowtabellen im vorangegangenen Beispiel finden ihre Verallgemeinerung in der Funktion $C(\omega, s; t, T)$. Diese beschreibt den Cashflow für den Pfad ω zur Zeit s , bedingt darauf, dass die Option nicht zu oder vor $t < s \leq T$ ausgeübt wurde und dass sich der Halter der Option für alle s optimal verhält.

Am Ende der Laufzeit der Option übt der Halter die Option aus, wenn sie in the money ist, ansonsten verfällt sie. Zu jedem Zeitpunkt t_k davor muss er entscheiden, ob er ausüben will oder nicht. Für diese beiden Möglichkeiten sollen nun die Cashflows verglichen werden. Der Cashflow des Ausübens ist dabei sofort berechenbar. Der Cashflow des Beibehaltens der Option kann als bedingter Erwartungswert ausgedrückt werden:

$$F(\omega; t_k) = \mathbb{E}_Q \left[\sum_{j=k+1}^L \exp \left(- \int_{t_k}^{t_j} r(\omega, s) ds \right) C(\omega, t_j; t_k, T) | \mathcal{F}_{t_k} \right] \quad (2.1)$$

wobei $r(\omega, t)$ die risikolose, möglicherweise stochastische Zinsrate ist. Die Option wird somit zu t_k vorzeitig ausgeübt, wenn der Wert des sofortigen Ausübens den Wert des Beibehaltens, ausgedrückt in Formel (2.1) übersteigt.

Sobald also der Wert dieses bedingten Erwartungswerts berechnet oder zumindest hinreichend genau approximiert werden kann, ist die Bewertung der Option möglich. Hier kommt nun die polynomiale Regression ins Spiel. Wenn angenommen werden darf, dass der bedingte Erwartungswert ein Element des Hilbertraums L^2 ist, kann er als gewichtete Summe von orthogonalen Basiselementen des Raums dargestellt werden⁴, also

$$F(\omega; t_{L-1}) = \sum_{j=0}^{\infty} \alpha_j e_j(X_{t_k}(\omega)). \quad (2.2)$$

wobei die α_j konstante Koeffizienten und die Familie $(e_j(x))_{j=0}^{\infty}$ eine Basis des L^2 bildet. Eine mögliche Wahl für orthogonale Basisfunktionen sind gewichtete Laguerre Polynome:

2.2.1 Definition. *Die nach folgendem Gesetz gebildeten Polynome heißen Laguerre Polynome.*

$$\begin{aligned} e_0(x) &= 1 \\ e_1(x) &= 1 - x \\ e_2(x) &= 1 - 2x + \frac{x^2}{2} \\ e_n(x) &= \frac{\exp(x)}{n!} \frac{d^n}{dx^n} (x^n \exp(-x)) \end{aligned} \quad (2.3)$$

2.2.2 Satz. *Das System der Laguerre Polynome aus Definition (2.2.1) zusam-*

⁴vgl. [W], Kapitel 3

men mit der Gewichtsfunktion $\exp(-x)$ bildet ein Orthogonalsystem, d.h.

$$\begin{aligned} & \left(\exp\left(-\frac{x}{2}\right) L_n(x), \exp\left(-\frac{x}{2}\right) L_m(x) \right) = \\ & = \int_0^\infty \exp(-x) e_n(x) e_m(x) dx = 0 \quad \forall m, n \in \mathbb{N}, m \neq n. \end{aligned} \quad (2.4)$$

Andere mögliche Basisfunktionen sind etwa Hermite, Legendre, Chebyshev, Gegenbauer oder Jacobi Polynome. Das Rechenbeispiel aus Kapitel 2.1 verwendet klassische Polynome 2. Ordnung.

Mithilfe dieser Konstruktion kann der bedingte Erwartungswert (2.1) durch die ersten $M < \infty$ Summanden aus (2.2) approximiert werden. Wir erhalten die Näherung

$$F_M(\omega; t_{L-1}) = \sum_{j=0}^{M-1} \alpha_j e_j(X_{t_k}(\omega)). \quad (2.5)$$

Nun werden die diskontierten Werte von $C(\omega, s; t_{L-1}, T)$ auf die Basisfunktionen projiziert. Da wir die exakten Koeffizienten α_j für die Projektion nicht kennen, werden wir diese mithilfe von N simulierten Pfaden schätzen, wobei nur jene Pfade in Betracht gezogen werden, die zum Zeitpunkt t_{L-1} in the money sind. Die Bedingung an den gesuchten Koeffizientenschätzer $\hat{\alpha}^{(M,N)}$ ist, dass er

$$\begin{aligned} & \min_{a \in \mathbb{R}^M} \sum_{n=1}^N \left[a_0 e_0(X_{t_k}(\omega)) + \dots + a_{M-1} e_{M-1}(X_{t_k}(\omega)) \right. \\ & \left. - \exp\left(-\int_{t_k}^{t_{k+1}} r(\omega_n, s) ds\right) C(\omega_n, t_{k+1}; t_k, T) \right]^2 \end{aligned} \quad (2.6)$$

löst.

Mithilfe der Matrix

$$V_M := \begin{pmatrix} e_0(X_{t_k}(\omega_1)) & \dots & e_M(X_{t_k}(\omega_1)) \\ \vdots & & \vdots \\ e_0(X_{t_k}(\omega_N)) & \dots & e_M(X_{t_k}(\omega_N)) \end{pmatrix} \quad (2.7)$$

die die Basisfunktionen angewandt auf die N Pfade enthält, erhalten wir eine Darstellung der Lösung

$$\hat{\alpha}^{(M,N)} = (V_M^T V_M)^{-1} V_M^T y, \quad y = (y_1, \dots, y_N)^T. \quad (2.8)$$

Mathematisch gesehen entspricht das Schätzen der $\hat{\alpha}^{(M,N)}$ einer Projektion auf den von $\{e_0(\cdot), \dots, e_M(\cdot)\}$ aufgespannten Unterraum. Die Bedingte Erwartung aus Formel (2.2) ist dabei wie vorausgesetzt ein Element des L^2 , welcher bekanntlich ein Hilbertraum ist. Dieser hat die Orthogonalbasis $\{e_j\}_{j=0}^\infty$ und jedes Element besitzt also eine Darstellung bezüglich dieser Basis. Wir betrachten nun den von endlich vielen Basiselementen aufgespannten Unterraum $\{e_0(\cdot), \dots, e_M(\cdot)\}$ und projizieren auf diesen. Es existiert eine eindeutige Lösung des Minimierungsproblems, falls der aufgespannte Unterraum nichtleer, konvex und abgeschlossen ist⁵. Klarerweise ist jede Linearkombination von Basisvektoren wieder Element des Unterraums und damit ist dieser nichtleer, konvex und abgeschlossen.

⁵vgl. [W]

Mit den Schätzern $\hat{\alpha}_j^{(M,N)}$ erhalten wir einen Schätzer für $F_M(\omega; t_{L-1})$

$$\hat{F}_M^N(\omega; t_{L-1}) = \sum_{j=0}^M \hat{\alpha}_j^{(M,N)} e_j(X_{t_{L-1}}(\omega)) \quad (2.9)$$

Die Überlegung, ob vorzeitig ausgeübt wird, macht nur dann Sinn, wenn die Option zu diesem Zeitpunkt in the money ist. Dadurch kann die Beschränkung auf die in the money Pfade gerechtfertigt werden. Longstaff und Schwartz weisen auch auf numerische Experimente, die eine bessere Performance und weniger nötige Basisfunktionen für gleiche Genauigkeit implizieren.

Die simulierten Pfade werden unabhängig und identisch verteilt generiert. Also sind auch die Werte der Basisfunktionen zu jedem Zeitpunkt unabhängig und identisch verteilt. Wir erhalten die Konvergenz von $\hat{F}_M^N(\omega; t_{L-1}) \rightarrow F(\omega; t_{L-1})$ in Wahrscheinlichkeit und auch in L^2 . Weiters ist $\hat{F}_M^N(\omega; t_{L-1})$ bezüglich der L^2 -Norm der beste lineare unverzerrte Schätzer für $F(\omega; t_{L-1})$. Genauere Konvergenzüberlegungen werden im folgenden Kapitel angestellt.

Sobald der Schätzer für $F(\omega; t_{L-1})$ berechnet wurde, kann entschieden werden, ob die Option zu t_{L-1} ausgeübt wird, bedingt darauf, dass sie bis dahin noch nicht ausgeübt wurde. Falls ja, bedeutet das, dass die Option spätestens zu t_{L-1} ausgeübt wird. Der Algorithmus läuft nun rückwärts in der Zeit weiter, es kann also sein, dass die Option noch früher ausgeübt wird. Für jedes ω werden dazu wieder analog zu obigem Algorithmus die Cashflows $C(\omega, s; t_{L-2}, T)$ approximiert. Nach dem letzten Schritt des Algorithmus erhält man für jeden Pfad die optimale Ausübungsstrategie. Um nun den Preis der gesamten Option zu bestimmen, wird für jeden Pfad der resultierende Cashflow auf den Zeitpunkt $t = 0$ diskontiert und über alle Pfade gemittelt.

Es sei noch am Rande erwähnt, dass die hier beschriebene Vorgehensweise auch angewandt werden kann, wenn der Payoff der Option auf mehr als einer Aktie basiert. Die Basisfunktionen enthalten in diesem Fall Terme aller Aktien.

2.3 Konvergenz nach Clément et al.⁶

Die Konvergenz des Algorithmus basiert also auf der Konvergenz des Schätzers $\hat{F}_M^N(\omega; t_k)$ gegen den Continuation Value $F(\omega; t_k)$ zu jedem Zeitpunkt t_k . Diese Konvergenz wird nun näher betrachtet.

Wir betrachten wieder den Wahrscheinlichkeitsraum $(\Omega, \mathcal{F}, \mathbb{P})$ mit der Filtration $(\mathcal{F}_{t_j})_{j=0, \dots, L}$ und einen dazu adaptierten Prozess der Auszahlungen $(Z_{t_j})_{j=0, \dots, L}$ mit quadratisch integrierbaren Zufallsvariablen Z_{t_j} . Im Falle einer Put Option gilt $Z_{t_j} = v(t_j, S_{t_j}) = (K - S_{t_j})^+$. Der Wert dieser Option ist dann

$$\sup_{\tau \in \mathcal{T}_{0,L}} \mathbb{E} Z_\tau$$

wobei $\mathcal{T}_{j,L}$ die Menge aller Stoppzeiten mit Werten in $\{t_j, \dots, t_L\}$ bezeichnet. Die Snell Einhüllende $(U_{t_j})_{j=0, \dots, L}$ ist dann definiert als

$$U_{t_j} = \text{ess sup}_{\tau \in \mathcal{T}_{0,L}} \mathbb{E} (Z_\tau | \mathcal{F}_{t_j}), \quad j = 0 \dots, L$$

⁶vgl. [C], ab Kapitel 2

Dem Prinzip der dynamischen Programmierung folgend kann diese geschrieben werden als

$$\begin{cases} U_{t_L} = Z_{t_L} \\ U_{t_j} = \max \{ Z_{t_j}, \mathbb{E}(U_{t_{j+1}} | \mathcal{F}_{t_j}) \}, \quad j = 0, \dots, L-1 \end{cases} \quad (2.10)$$

Weiters gilt $U_{t_j} = \mathbb{E}(Z_{\tau_j} | \mathcal{F}_{t_j})$ mit

$$\tau_j = \min\{k \geq j | U_{t_k} = Z_{t_k}\}$$

Mit dieser optimalen Stoppzeit lässt sich der Wert der Option zum Zeitpunkt 0 angeben als $\mathbb{E}U_0 = \sup_{\tau \in \mathcal{T}_{0,L}} \mathbb{E}Z_\tau = \mathbb{E}Z_{\tau_0}$. Nun gilt es, diese optimale Stoppzeit zu finden, wobei das Problem schrittweise angegangen werden kann:

$$\begin{cases} \tau_L = T \\ \tau_j = t_j \mathbb{1}_{\{Z_{t_j} \geq \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})\}} + \tau_{j+1} \mathbb{1}_{\{Z_{t_j} < \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})\}}, \quad j \in \{0, \dots, L-1\} \end{cases} \quad (2.11)$$

Das zugrunde liegende Modell für den Aktienpreis soll eine Markov Kette sein. Dazu nehmen wir an, dass es eine (\mathcal{F}_{t_j}) -Markov Kette (X_{t_j}) im Zustandsraum (E, \mathcal{E}) gibt, sodass für $j \in \{0, \dots, L\}$ gilt

$$Z_{t_j} = f(t_j, X_{t_j})$$

mit einer Borel-messbaren Funktion $f(t_j, \cdot)$. Somit ist auch U_{t_j} mit einer Borel-messbaren Funktion $V(t_j, \cdot)$ als $U_{t_j} = V(t_j, X_{t_j})$ darstellbar. Der Anfangszustand $X_0 = x$ wird deterministisch festgesetzt und somit ist auch U_0 deterministisch. Das zugrundeliegende Modell für den Stock kann ohne Weiteres eine Markov Kette sein. Der zukünftige Wert des Stocks ist damit nur vom aktuellen Wert abhängig und nicht von der ganzen Kurshistorie. Die später in dieser Arbeit verwendeten Modelle erfüllen alle diese Bedingung.

Die erste Approximation besteht nun darin, die bedingte Erwartung in Gleichung (2.11) durch eine endliche Linearkombination aus Basisfunktionen des zugrunde liegenden Zustandsraumes (E, \mathcal{E}) zu ersetzen. Dazu betrachten wir eine Folge $(e_k(X_{t_j}))_{k \geq 1} \in L^2(\sigma(X_{t_j}))$ von linear unabhängigen, messbaren, reellwertigen Funktionen auf E . Für $j = 1, \dots, L-1$ bezeichne P_j^M die orthogonale Projektion von $L^2(\Omega)$ auf den von M Basisfunktionen aufgespannten Vektorraum $\{e_1(X_{t_j}), \dots, e_M(X_{t_j})\}$. Damit lässt sich (2.11) schreiben als

$$\begin{cases} \tau_L^M = T \\ \tau_j^M = t_j \mathbb{1}_{\{Z_{t_j} \geq P_j^M(Z_{\tau_{j+1}^M})\}} + \tau_{j+1}^M \mathbb{1}_{\{Z_{t_j} < P_j^M(Z_{\tau_{j+1}^M})\}}, \quad j \in \{1, \dots, L-1\} \end{cases} \quad (2.12)$$

Und wir erhalten eine Näherung für die Wertefunktion:

$$U_0^M = \max \left(Z_0, \mathbb{E}Z_{\tau_1^M} \right) \quad (2.13)$$

Wie oben verlangt ist dabei $Z_0 = f(0, x)$ deterministisch. Die zweite Approximation besteht nun darin, den Erwartungswert $\mathbb{E}Z_{\tau_1^M}$ durch eine

Monte Carlo Prozedur auszuwerten. Hierzu werden N unabhängige Pfade $(X_{t_j}^{(1)}, \dots, X_{t_j}^{(n)}, \dots, X_{t_j}^{(N)})$, beziehungsweise die dazugehörigen Auszahlungsfunktionen $(Z_{t_j}^{(1)}, \dots, Z_{t_j}^{(n)}, \dots, Z_{t_j}^{(N)})$ für $j \in \{0, t_1, \dots, t_L = T\}$ simuliert.

Für jeden Pfad n schätzen wir dann rekursiv die Stoppzeiten τ_j^M durch

$$\begin{cases} \tau_L^{n,M,N} = T \\ \tau_j^{n,M,N} = t_j \mathbb{1}_{\{Z_{t_j}^{(n)} \geq \alpha_j^{(M,N)} \cdot e^M(X_{t_j}^{(n)})\}} + \tau_{j+1}^{n,M,N} \mathbb{1}_{\{Z_{t_j}^{(n)} < \alpha_j^{(M,N)} \cdot e^M(X_{t_j}^{(n)})\}} \\ j \in \{1, \dots, L-1\} \end{cases} \quad (2.14)$$

Dabei bezeichne $x \cdot y$ das innere Produkt auf \mathbb{R}^M , e^M die vektorwertige Funktion (e_1, \dots, e_M) und $\alpha_j^{(M,N)}$ den Kleinstquadrateschätzer

$$\alpha_j^{(M,N)} = \operatorname{argmin}_{a \in \mathbb{R}^M} \sum_{n=1}^N \left(Z_{\tau_{j+1}^{n,M,N}}^{(n)} - a \cdot e^M(X_{t_j}^{(n)}) \right)^2 \quad (2.15)$$

Mit den so errechneten Stoppzeiten $\tau_j^{n,M,N}$ erhalten wir einen Schätzer für U_0^M :

$$U_0^{M,N} = \max \left(Z_0, \frac{1}{N} \sum_{n=1}^N Z_{\tau_{j+1}^{n,M,N}}^{(n)} \right) \quad (2.16)$$

Nun kann gezeigt werden, dass für ein festes M der Schätzer $U_0^{M,N}$ fast sicher gegen U_0^M konvergiert für $N \rightarrow \infty$ und dass U_0^M gegen U_0 konvergiert, wenn $M \rightarrow \infty$.

Longstaff und Schwarz berücksichtigen in ihrem Algorithmus zur Optionsbewertung nur die in the money Pfade. Um den Konvergenzbeweis anwenden zu können, sind nur ein paar kleine Modifikationen nötig. Statt τ_j^M verwenden wir für $1 \leq j \leq L-1$

$$\hat{\tau}_j^M = t_j \mathbb{1}_{\{Z_{t_j} \geq \hat{\alpha}_j^M \cdot e(X_{t_j})\} \cap \{Z_{t_j} > 0\}} + \hat{\tau}_{j+1}^M \mathbb{1}_{\{Z_{t_j} < \hat{\alpha}_j^M \cdot e(X_{t_j})\} \cup \{Z_{t_j} = 0\}} \quad (2.17)$$

Analog können wir $\hat{\tau}_j^{n,M,N}$ und $\hat{\alpha}_j^{(M,N)}$ definieren, wobei

$$\hat{\alpha}_j^{(M,N)} = \arg \min_{a \in \mathbb{R}^M} \mathbb{E} \mathbb{1}_{\{Z_{t_j} > 0\}} \left(Z_{\hat{\tau}_{j+1}^M} - a \cdot e(X_{t_j}) \right)^2 \quad (2.18)$$

Weiters muss $(\mathbb{1}_{\{Z_{t_j} > 0\}} e(X_{t_j}))_{j=1, \dots, L-1}$ linear unabhängig und Element von $L^2(\sigma(X_{t_j}), \mathbb{1}_{\{Z_{t_j} > 0\}} d\mathbb{P})$ sein.

Die folgenden beiden Sätze behandeln nun die Konvergenz der Approximation des Continuation Values:

2.3.1 Satz. *Angenommen die Folge $(e_k(X_{t_j}))_{k \geq 1}$ liegt für alle $j = 1, \dots, L$ in $L^2(\sigma(X_{t_j}))$. Dann gilt:*

$$\lim_{M \rightarrow +\infty} \mathbb{E} \left(Z_{\tau_j^M} | \mathcal{F}_{t_j} \right) = \mathbb{E} \left(Z_{\tau_j} | \mathcal{F}_{t_j} \right) \quad (2.19)$$

in L^2 .

Beweis. Wir verwenden vollständige Induktion nach j .

Induktionsanfang: Die Aussage gilt für $j = L$.

Induktionsschritt: Wir zeigen, dass aus der Gültigkeit für $j + 1$ die für $j \leq L - 1$ folgt. Es gilt

$$Z_{\tau_j^M} = Z_{t_j} \mathbb{1}_{\{Z_{t_j} \geq \alpha_j^{(M,N)} \cdot e^M(X_{t_j})\}} + Z_{\tau_{j+1}^M} \mathbb{1}_{\{Z_{t_j} < \alpha_j^{(M,N)} \cdot e^M(X_{t_j})\}}. \quad (2.20)$$

Für $j \leq L - 1$ gilt

$$\begin{aligned} & \mathbb{E}(Z_{\tau_j^M} - Z_{\tau_j} | \mathcal{F}_{t_j}) \\ = & \underbrace{(Z_{t_j} - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})) \left(\mathbb{1}_{\{Z_{t_j} \geq \alpha_j^{(M,N)} \cdot e^M(X_{t_j})\}} - \mathbb{1}_{\{Z_{t_j} \geq \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})\}} \right)}_{B_j^M} \\ & + \underbrace{\mathbb{E}(Z_{\tau_{j+1}^M} - Z_{\tau_{j+1}} | \mathcal{F}_{t_j}) \mathbb{1}_{\{Z_{t_j} < \alpha_j^{(M,N)} \cdot e^M(X_{t_j})\}}}_{(\Delta)} \end{aligned} \quad (2.21)$$

(Δ) konvergiert der Induktionsannahme nach gegen 0. Zu zeigen bleibt, dass B_j^M auch gegen 0 in L^2 konvergiert. Dazu betrachten wir zuerst

$$\begin{aligned} |B_j^M| &= |(Z_{t_j} - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j}))| \cdot \\ & \left| \mathbb{1}_{\{\mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j}) > Z_{t_j} \geq \alpha_j^{(M,N)} \cdot e^M(X_{t_j})\}} \mathbb{1}_{\{\alpha_j^{(M,N)} \cdot e^M(X_{t_j}) \geq Z_{t_j} \geq \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})\}} \right| \\ & \leq |Z_{t_j} - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})| \cdot \\ & \quad \mathbb{1}_{\{|Z_{t_j} - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})| \leq |\alpha_j^{(M,N)} \cdot e^M(X_{t_j}) - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})|\}} \\ & \leq |\alpha_j^{(M,N)} \cdot e^M(X_{t_j}) - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})| \\ & \leq |\alpha_j^{(M,N)} \cdot e^M(X_{t_j}) - P_j^M(\mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j}))| + \\ & \quad |P_j^M(\mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j}))| - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})| \end{aligned} \quad (2.22)$$

Nun gilt aber nach Definition der Projektion P_j^M

$$\alpha_j^{(M,N)} \cdot e^M(X_{t_j}) = P_j^M \left(Z_{\tau_{j+1}}^M \right) = P_j^M \left(\mathbb{E}(Z_{\tau_{j+1}^M} | \mathcal{F}_{t_j}) \right) \quad (2.23)$$

da P_j^M die Orthogonalprojektion auf einen Unterraum von \mathcal{F}_{t_j} -messbaren Zufallsvariablen ist. Somit folgt

$$\begin{aligned} \|B_j^M\|_2 &\leq \left\| \mathbb{E}(Z_{\tau_{j+1}^M} | \mathcal{F}_{t_j}) - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j}) \right\|_2 + \\ & \quad \left\| P_j^M(\mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j})) - \mathbb{E}(Z_{\tau_{j+1}} | \mathcal{F}_{t_j}) \right\|_2 \end{aligned} \quad (2.24)$$

Dabei geht der erste Term wieder nach Induktionsvoraussetzung gegen 0 und der zweite Term nach der Voraussetzung des Satzes gegen 0. \square

Wir haben somit eine L^2 -Konvergenz des auf M Ansatzfunktionen basierenden Continuation Values gegen den echten Continuation Value. Im nächsten Schritt wird die Anzahl M der Basisfunktionen fixiert und $U_0^{M,N}$ betrachtet wenn die Anzahl der Pfade N , auf Basis derer die Koeffizienten $\alpha^{(M,N)}$ geschätzt werden, gegen Unendlich geht:

2.3.2 Satz. *Angenommen für $j = 1, \dots, L - 1$ gilt $\mathbb{P}(\alpha_j \cdot e(X_{t_j}) = Z_{t_j}) = 0$. Dann konvergiert $U_0^{M,N}$ fast sicher gegen U_0^M mit $N \rightarrow \infty$. Außerdem erhalten wir fast sichere Konvergenz von $\frac{1}{N} \sum_{n=1}^N Z_{\tau_j^{n,M,N}}^{(n)}$ gegen $\mathbb{E}Z_{\tau_j}^M$ falls $N \rightarrow \infty$ für $j = 1, \dots, L$.*

Beweis. Auf den Beweis dieses Satzes wird hier verzichtet. Der interessierte Leser findet ihn in [C]. \square

Die Approximation des Continuation Values konvergiert somit gegen den echten Continuation Value und alle nötigen Werkzeuge für die Bewertung der Option stehen zur Verfügung.

Während der LSM Algorithmus zwar gegen den echten Preis der amerikanischen Option konvergiert, wird für jede endliche, numerische Implementierung der echte Preis systematisch unterschätzt, wie leicht zu zeigen ist. [LS] liefern folgenden Beweis für diese Tatsache:

2.3.3 Proposition. *Für jede endliche Wahl von M Ansatzfunktionen, L Ausübungszeitpunkten und N simulierten Pfaden sei $LSM(\omega, M, L)$ der resultierende Cashflow des Pfades ω , wobei nach der vom LSM Algorithmus berechneten Strategie ausgeübt wird. Sei $V(X)$ der echte Preis der Amerikanischen Option. Dann gilt folgende Ungleichung fast sicher:*

$$V(X) \geq \lim_{N \rightarrow \infty} \sum_{i=1}^N LSM(\omega_i, M, L) \quad (2.25)$$

Beweis. Definitionsgemäß ist der Wert X_{t_k} der zugrunde liegenden Aktie zu t_k \mathcal{F}_{t_k} -messbar. Außerdem ist der Wert des sofortigen Ausübens \mathcal{F}_{t_k} -messbar. $F_M(\omega; t_k)$ ist als Linearkombination von \mathcal{F}_{t_k} -messbaren Funktionen von X_{t_k} ebenfalls messbar. Daher ist das Ereignis, dass der Wert des vorzeitigen Ausübens größer null und größer als $F_M(\omega; t_k)$ ist, in \mathcal{F}_{t_k} und die vom LSM Algorithmus identifizierte Strategie eine Stoppzeit. Der Barwert, den diese Stoppzeit liefert, sei mit E_θ bezeichnet. Da $V(X)$ das Supremum der Barwert gebildet über alle Stoppzeiten ist, gilt

$$V(X) \geq E_\theta. \quad (2.26)$$

Da die funktionale Form der $F_M(\omega; t_k)$ für alle Pfade gleich ist und die Pfade unabhängig und identisch verteilt sind, sind auch die diskontierten Cashflows $LSM(\omega_i, M, L)$ unabhängig und identisch verteilt. Nach dem Gesetz der großen Zahlen gilt nun

$$\mathbb{P} \left[\lim_{N \rightarrow \infty} \sum_{i=1}^N LSM(\omega_i, M, L) = E_\theta \right] = 1 \quad (2.27)$$

Zusammen mit (2.26) folgt die Aussage. \square

Die Intuition hinter diesem Satz ist recht einfach zusammenzufassen: Der Preis einer amerikanischen Option ist der Payoff unter der optimalen Stoppzeit. Diese optimale Stoppzeit kennen wir aber nicht und können sie nur approximieren. Somit verwenden wir eine suboptimale Stoppzeit. Das bedeutet aber, dass auch der Payoff unter dieser approximierten Stoppzeit kleiner gleich dem Payoff mit optimaler Stoppzeit ist.

2.4 Konvergenzordnung⁷

Stentoft [S] liefert neben einem Konvergenzbeweis auch Betrachtungen der Konvergenzordnung des LSM Algorithmus, wobei er dafür hauptsächlich die allgemeinen Resultate für Series Estimators von [N] verwendet. [S] passt dafür Theorem 1 in [N] an die gegebene Situation an. Er führt dazu selbst drei Annahmen ein. Sein Beweis beruht darauf zu zeigen, dass aus den Annahmen 1-3 die Annahmen I-III von [N] folgen und daher Theorem 1 in [N] anwendbar ist.

[S] verwendet als Ansatzfunktionen verschobene Legendre Polynome, welche eine Orthonormalbasis bilden. Zusätzlich verlangt er die Gültigkeit folgender Annahmen:

Annahme 1. (i) Die simulierten Pfade $X(\omega_n), 1 \leq n \leq N$ sind unabhängig und (ii) $\mathbb{P}(Z_{t_k}(\omega) = F(\omega, t_k)) = 0, 0 \leq k \leq L$.

Während (i) genau den Gedanken von Monte Carlo Methoden verdeutlicht, ist (ii) weniger intuitiv. [S] liefert hierfür folgende Erklärung: angenommen, es existiert ein $\bar{\omega}$ mit $F(\bar{\omega}, t) = Z_t(\bar{\omega})$. Obwohl $\hat{F}_M^N(\omega, t)$ gegen $F(\omega, t)$ konvergiert, gilt immer $\hat{F}_M^N(\bar{\omega}, t) > F(\bar{\omega}, t)$. Es ist also suboptimal die Option vorzeitig auszuüben, aber optimal im Grenzwert. Die Näherung der bedingten Erwartung konvergiert also, aber die optimale Stoppzeit wird nie gefunden. Annahme 1 stellt sicher, dass dieses Problem mit Wahrscheinlichkeit Null auftritt und die optimale Stoppzeit gefunden werden kann.

Annahme 2. Der Träger \mathcal{H} von X ist ein kartesisches Produkt von zusammenhängenden, kompakten Intervallen, auf denen X eine Dichte $f(x) \geq \varepsilon$ für ein $\varepsilon > 0$ besitzt.

Annahme 3. Die bedingte Erwartungswertfunktion ist s -mal stetig differenzierbar auf \mathcal{H} .

Nun können wir eine Konvergenzaussage für die Näherung der bedingten Erwartung formulieren (Theorem 1 in [S]):

2.4.1 Satz. *Es gelten die Annahmen 1-3 und $M = M(N)$ wächst mit N so, dass $\frac{M^3}{N} \xrightarrow{M \rightarrow \infty} 0$. Dann konvergiert der Schätzer $\hat{F}_M^N(\omega; t_k)$ in L^2 und*

$$\int \left(F(\omega; t_k) - \hat{F}_M^N(\omega; t_k) \right)^2 dF_0 = O_p \left(\frac{M}{N} + M^{-2s} \right) \quad (2.28)$$

wobei $F_0(x)$ die Verteilungsfunktion der Pfade bezeichnet.

Es ist nicht weiter verwunderlich, dass die Konvergenzordnung hauptsächlich von der Anzahl N der Monte Carlo Pfade und der Anzahl der Basisfunktionen M abhängt. Es ist aber auch wichtig, dass diese beiden Parameter in einem günstigen Verhältnis zueinander stehen. Dieser Umstand findet sich auch schon in den Voraussetzungen des Satzes.

⁷vgl. [S], Kapitel 2 und [N]

Kapitel 3

Implementierung des LSM Algorithmus in Maple

In diesem Kapitel beschäftigen wir uns mit der expliziten Implementierung des Least Squares Monte Carlo Algorithmus in Maple 15. Dabei wurde kein Fokus auf Performance gelegt, sondern es geht primär darum, den Algorithmus besser zu verstehen, auszuprobieren und damit zu experimentieren. Der Aufbau des gesamten Programms ist dabei modular in Form von mehreren Prozeduren. Damit erhöht sich die Lesbarkeit und Verständlichkeit des Codes wesentlich, der Performance sind sie nicht unbedingt zuträglich.

3.1 Vorbereitende Prozeduren

Begonnen wird mit dem Einbinden der benötigten Packages:

```
with(LinearAlgebra);  
with(plots);  
with(Statistics);  
with(PolynomialTools)
```

Das Aktienpreismodell ist ein Black-Scholes Modell. Die treibende stochastische Komponente dahinter ist eine Brownsche Bewegung. Die folgende Prozedur erzeugt eine diskrete Standard Brownsche Bewegung für das Intervall $[\frac{b}{n}, b]$ zu n äquidistanten Zeitpunkten.

```
makeBB := proc (n, b)  
    local X, L, B;  
    X := Sample(RandomVariable(Normal(0, (b/n)^(1/2))), n);  
    B := t->add(X[i], i = 1 .. t);  
    L := [seq([j*b/n, B(j)], j = 1 .. n)];  
    return L;  
end proc
```

Ausgegeben wird eine Liste mit den Zeitpunkten und zufälligen Werten der Brownschen Bewegung:

```
L := evalf[3](makeBB(8, 1));
[[0.125, 0.430], [0.250, 0.082], [0.375, 0.765], [0.500, 0.808],
 [0.625, 0.961], [0.750, 0.776], [0.875, 0.855], [1., 0.709]]
```

Mit dieser diskreten Brownschen Bewegung kann nun eine Matrix mit N zufälligen Pfaden erzeugt werden. Jede Zeile entspricht einem Pfad. Die Zeit bis zum Endzeitpunkt T wird dabei in n äquidistante Intervalle aufgeteilt. r ist die Zinsrate, σ die Volatilität und S_0 der Anfangspreis.

```
SimStockMatrix := proc (n, T, r, sigma, S0, N)
  local t, S, W, k;
  S := Matrix(N, n+1);
  t := <0|seq(j*T/n, j = 1 .. n)>;
  for k from 1 to N do ;
    W := makeBB(n, T);
    S[k, 1 .. n+1] := <S0, seq(S0*exp((r-(1/2)*sigma^2)*
      W[i, 1]+sigma*W[i, 2]), i = 1 .. n)>;
  end do;
  return t, S;
end proc;
```

Ausgegeben wird ein Vektor der Länge $n+1$ und eine $N \times n+1$ -Matrix mit den Pfaden:

```
Time, Stock := evalf[3](SimStockMatrix(7, 1, 0.06, .2, 1, 8))
Time:= [0., 0.14, 0.29, 0.43, 0.57, 0.71, 0.86, 1.]
Stock:=[[1., 1.14, 1.09, 1.04, 1.14, 1.10, 1.04, 0.91],
 [1., 1.09, 1.05, 1.14, 1.08, 1.07, 1.08, 1.06],
 [1., 1.08, 1.10, 1.02, 1.19, 1.03, .913, 1.04],
 [1., 1.12, 1.21, 1.22, 1.37, 1.34, 1.39, 1.39],
 [1., 1.13, 1.14, 1.22, 1.27, 1.30, 1.37, 1.58],
 [1., 1.08, 1.06, 0.89, 0.94, 0.88, 0.84, 0.83],
 [1., 0.97, 1.09, 1.27, 1.10, 1.17, 1.19, 1.22],
 [1., 1.07, .976, 1.05, 1.10, 1.12, 1.04, 0.99]]
```

Der Vorteil des separaten Erstellens der Pfadmatrix liegt neben der Lesbarkeit des Programmcodes darin, dass nun für eine feste Menge von Szenarien der LSM Algorithmus ausprobiert werden kann, also etwa mit unterschiedlichen Ansatzfunktionen experimentiert werden kann. Mit einer festen Pfadmatrix kann der Einfluss von unterschiedlichen Monte Carlo Szenarien eliminiert werden, da immer auf die gleichen Pfade zugegriffen wird und nicht für jede Ansatzfunktion eine neue Pfadmatrix generiert wird.

Die nächste Prozedur plottet bis zu z zufällig ausgewählte Pfade aus der Matrix M mit dem Zeitvektor T . Diese Prozedur dient dazu, die zufällig generierten Pfade zu visualisieren.

```
PlotSimMatrix:=proc(M,T,z,plotcol:=grey,plotlegend:="")
  local i, n, L, A, m, Z, roll, q, k;
  n := ColumnDimension(M);
  m := RowDimension(M);
  roll := rand(1 .. m);
```

```

Z := {seq(roll(), i = 1 .. z)};
q := nops(Z);
k := 1;
for i in Z do;
  L := [seq([T[j], M[i, j]], j = 1 .. n)];
  A[k] := pointplot(L, style = 'line',
    color = plotcol, legend = plotlegend)
  k := k+1
end do;
display({seq(A[i], i = 1 .. q)})
end proc;

```

Die Prozedur gibt den Plot in Abbildung 3.1 zurück.

```

Time, Stock := SimStockMatrix(500, 1, 0.06, 0.2, 1, 6);
PlotSimMatrix(Stock, Time, 6, 'blue');

```

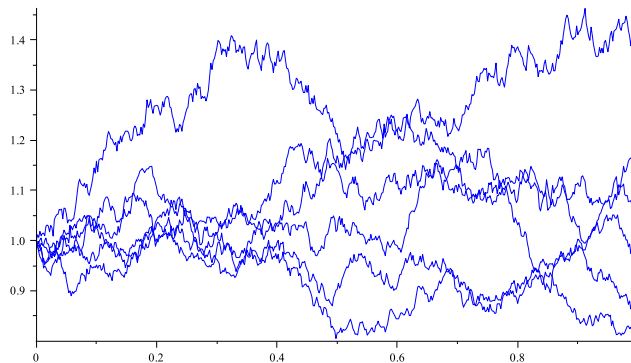


Abbildung 3.1: Pfade einer simulierten Aktie

Die nächste Prozedur dient der Verifikation der erzeugten Pfadmatrix M anhand des durchschnittlichen Returns der Pfade:

```

MeanReturn := proc (M)
  local m, n, R1;
  n := ColumnDimension(M);
  m := RowDimension(M);
  R1 := add(M[i, n], i = 1 .. m)/m;
  return (R1-M[1, 1])/M[1, 1];
end proc

```

Zurückgegeben wird der durchschnittliche Return, gemittelt über alle Pfade. Die risikolose Zinsrate beträgt hier 0.06, das heißt, wir erwarten als durchschnittlichen Return $\exp(0.06) - 1 \approx 0.06$, was bei den folgenden 2000 Pfaden auch ungefähr stimmt.

```

Time, Stock := SimStockMatrix(5, 1, 0.06, 0.2, 100, 2000)
MeanReturn(Stock);
0.06039548802612899

```

Mit den simulierten Pfaden kann natürlich auch eine europäische Option bewertet werden. Dazu werden anstatt des gesamten Pfadverlaufs nur die Schlusskurse verwendet. Folgende simple Prozedur führt diese Bewertung durch. Benötigt werden die Pfadmatrix M , der Zeitvektor T , die Auszahlungsfunktion v sowie der risikolose Zinssatz r .

```
ValueEuropean := proc (M, T, v, r)
  local m, n, Value;
  n := ColumnDimension(M);
  m := RowDimension(M);
  Value := add(v(M[t, n])*exp(-r*(T[n]-T[1])), t = 1 .. m)/m;
  return Value;
end proc;
```

Zum Vergleich kann die Black-Scholes Formel für europäische Put-Optionen herangezogen werden. St ist der Kurs der Aktie zum Bewertungszeitpunkt t . Die Fälligkeit der Option ist T , die risikolose Zinsrate ist r , der Strikepreis K und die Volatilität σ .

```
BSPut := proc (St, t, T, K, r, sigma)
  local d1, d2, C, Phi;
  d1 := (s,u)->(ln(s/K)+(r+(1/2)*sigma^2)*u)/(sigma*sqrt(u));
  d2 := (s,u)->(ln(s/K)+(r-(1/2)*sigma^2)*u)/(sigma*sqrt(u));
  Phi := t->int(exp(-(1/2)*s^2)/sqrt(2*Pi), s = -infinity .. t);
  C := evalf(-St*Phi(-d1(St, T-t))+
    exp(-r*(T-t))*K*Phi(-d2(St, T-t)))
  return C
end proc
```

Folgende Bewertung basiert auf 2000 simulierten Pfaden mit 10 Zeitschritten (die aber hier nicht benötigt werden). Die Volatilität der Aktie beträgt 0.3 und die risikolose Zinsrate 0.06. Schon bei dieser geringen Anzahl an Pfaden beträgt die Abweichung weniger als 3%, eine größere Anzahl an Pfaden würde die Genauigkeit weiter erhöhen.

```
Time, Stock := SimStockMatrix(10, 3, 0.06, 0.3, 100, 2000);
BSCall(100, 0, 3, 100, 0.06, 0.3);
28.13577559
ValueEuropean(Stock, Time, x->max(x-100, 0), 0.06);
28.92657742769853
```

Es macht natürlich nicht viel Sinn, bei Existenz einer expliziten Formel des Optionspreises eine Monte Carlo Simulation zu verwenden. Die explizite Formel gilt allerdings nur für das Black-Scholes Aktienpreismodell, die Monte Carlo Simulation kann mit jeder Pfadmatrix verwendet werden, also auch mit komplizierteren, möglicherweise realistischeren Aktienpreismodellen. Weiters können verschiedene Auszahlungsfunktionen mit nur kleinen Modifikationen implementiert werden und somit etwa neben der Put Option auch Barrier Optionen, also Optionen, bei denen eine vorher festgelegte Schranke über- oder unterschritten werden muss, bewertet werden. Bei solchen Optionen ist die Kurshistorie aber wieder nötig. Wenn die gesamte Pfadmatrix vorliegt, stellt das jedoch kein Problem dar.

3.2 Der LSM Algorithmus

Die bisher betrachteten Prozeduren waren als Vorarbeit für den LSM Algorithmus nötig, sie haben die wichtigsten Eingabedaten erstellt. Nun kann der Algorithmus - hier `ValueAmerican()` genannt - selbst aufgerufen werden. Der Übersichtlichkeit und Verständlichkeit halber werden wir die Prozedur aufgeteilt auf mehrere Teile behandeln. Die Prozedur verlangt als Eingabewerte eine Matrix `M` mit den Pfaden. Dabei ist es für den Algorithmus nicht relevant, nach welchem Modell diese Pfadmatrix erstellt wurde. Für die anschließenden numerischen Experimente wird dafür - sofern nicht anders erwähnt - ein Black-Scholes Aktienpreismodell verwendet. Der Parameter `T` soll ein Zeitvektor sein, der die vorzeitigen Ausübungszeitpunkte angibt. Die Dimension des Vektors muss mit der Spaltendimension von `M` übereinstimmen. `v` ist die Auszahlungsfunktion, für einen amerikanischen Put also $v : x \mapsto (K - x)^+$. Die Funktion `f` ist die Ansatzfunktion, die für die Regression verwendet wird, also etwa $f : x \mapsto a_0 + a_1x + a_2x^2$ für ein Polynom zweiten Grades. Und `r` ist der risikolose Zins, mit dem die Auszahlungen diskontiert werden. Zuerst werden einige Dimensionen ausgelesen und Variablen initialisiert.

```
ValueAmerican := proc (M, T, v, f, r)
  local m, n, C, S, U, z, i, j, Xtemp, Ytemp, X, Y, E, Xcont,
    Xex, k, Value, Strategy, RegCoeff;
  n := ColumnDimension(M);
  m := RowDimension(M);
```

Die Matrix `C`, in der die schon berechneten Cashflows gespeichert werden, wird nun initialisiert und ihre letzte Spalte mit dem Payoff befüllt, der entsteht, wenn die Option für jeden Pfad erst im letzten Schritt ausgeübt wird.

```
C := Matrix(m, n);
C[1 .. m, n] := <map(x-> v(x) , M[1 .. m, n])>;
```

Die Matrix `S` speichert die Ausübungsstrategie. In der letzten Spalte werden jene Zeilen mit 1 befüllt, bei denen der zugehörige Pfad eine Auszahlung erzeugt, sprich die Option in the money ist.

```
S := Matrix(m, n);
S[1 .. m, n] := <seq(piecewise(0 < v(M[j, n]), 1),
  j = 1 .. m)>;
RegCoeff := [];
```

Der Algorithmus läuft nun rückwärts in der Zeit mit der Laufvariable `j`. Die untere Grenze des Algorithmus ist der Schritt `j=3`, da zum Zeitpunkt 0 (entspricht `j=1`) nicht ausgeübt werden kann. Die Auszahlungen zu `j` sind im aktuellen Schritt immer bekannt, jene für `j-1` werden im aktuellen Schritt berechnet.

Der Vektor `U` ist eine Hilfsvariable, in der gespeichert wird, welche Pfade in the money sind. Nur jene Pfade werden später in der Regression berücksichtigt.

```
for j from n by -1 to 3 do;
  U := Vector(m);
  z := 1;
  Ytemp := [];
  Xtemp := [];
```


Es werden nun die in the money Pfade gesucht und daraus die Regressionsvektoren erstellt. `Xtemp` enthält die in the money Kurse aus der (j-1)ten Spalte der Pfadmatrix, `Ytemp` enthält für jeden in the money Pfad den zukünftigen, auf den zu j korrespondierenden Zeitpunkt diskontierten Payoff. Die Variable `z` zählt, wie viele Pfade in the money sind.

```

for i from 1 to m do;
  if 0 < v(M[i, j-1]) then;
    U[i] := 1;
    Ytemp := [op(Ytemp), add(C[i, k]*
      exp(-r*(T[k]-T[k-1])), k = j .. n)];
    Xtemp := [op(Xtemp), M[i, j-1]];
    z := z+1;
  end if
end do;
X := convert(Xtemp, Vector);
Y := convert(Ytemp, Vector);

```

Nun kann die polynomiale Regression berechnet werden. Dafür verwenden wir die in Maple vorhandene Funktion `Fit()`. Rückgabewert ist das anfangs übergebene Polynom mit geschätzten Koeffizienten. Dieses Polynom ist die Approximation des Continuation Values. Auf jeden Preis in `X` werden jetzt zwei Funktionen angewandt: einerseits die Continuation Value Funktion, gespeichert in `Xcont` und andererseits die Auszahlungsfunktion, gespeichert in `Xex`. Weiters wird die Regressionsfunktion in `RegCoeff` gespeichert.

```

E := unapply(Fit(f, X, Y, x), x);
RegCoeff := [op(RegCoeff), E(x)];
Xcont := map(x->E(x), X);
Xex := map(x-> v(x), X);
z := 1;

```

Jetzt können die jeweiligen Cashflows der in the money Pfade zeilenweise verglichen und dadurch eine Strategie erstellt werden.

```

for i from 1 to m do;
  if U[i] = 1 then;

```

Wenn der Cashflow des Beibehaltens der Option größer als der Cashflow des sofortigen Ausübens ist, wird dieser Pfad nicht zu j-1 ausgeübt und die vorher schon berechnete Strategie für diesen Pfad beibehalten.

```

  if Xex[z] < Xcont[z] then;
    for k from j to n do;
      C[i, k] := S[i, k]*v(M[i, k]);
    end do;

```

Ist aber der Cashflow des sofortigen Ausübens größer, so wird diese Entscheidung in die Strategiematrix `S` gespeichert, der Payoff in die Cashflowmatrix `C` und alle zukünftigen Cashflows und Strategien auf 0 gesetzt (die Option kann nur einmal ausgeübt werden).

```

else;
  S[i, j-1] := 1;
  S[i, j .. n] := 0;
  C[i, j-1] := v(M[i, j-1]);
  C[i, j .. n] := 0;
end if;
z := z+1;
end if;
end do;
end do;

```

Wenn alle Zeitpunkte behandelt wurden, erhalten wir die Cashflowmatrix **C** und die Strategiematrix **S**. In jeder Zeile kann maximal ein Eintrag ungleich 0 sein.

Die einzelnen Cashflows werden - um den Preis der Option zu Beginn zu erhalten - auf den Zeitpunkt 0 diskontiert und anschließend über alle Szenarien gemittelt. Gespeichert wird der Preis in **Value**. Daneben hat die Prozedur noch andere Rückgabewerte: **Strategy** mittelt spaltenweise über die Strategiematrix **S** und gibt damit für jeden Zeitpunkt die relative Anzahl der ausgeübten Pfade an. Neben den Matrizen **S** und **C** wird außerdem die Liste **RegCoeff** zurückgegeben, in der die Regressionsfunktionen für jeden Zeitpunkt enthalten sind.

```

Value := add(add(C[t, k]*exp(-r*(T[k]-T[1])),
  k = 2 .. n), t = 1 .. m)/m;
Strategy := [seq([T[j], add(S[i, j]/m, i = 1 .. m)],
  j = 1 .. n)];
return Value, Strategy, S, C, RegCoeff;
end proc

```

3.3 Zurück zum einführenden Rechenbeispiel

Als Beispiel für die Funktionsweise des Algorithmus kann nun das einführende Rechenbeispiel aus Kapitel 2.1 verwendet werden. Anstatt eine Pfadmatrix mit der Funktion **SimStockMatrix()** zu erstellen, verwenden wir natürlich die im Beispiel gegebene Pfadmatrix mit acht Pfaden sowie die angegebenen Parameter: Strike Preis 1.1, drei mögliche Ausübungszeitpunkte $\{1, 2, 3 = T\}$ und die risikolose Zinsrate 6%.

```

Time:= [0, 1 , 2 , 3 ]
Stock:=[[1, 1.09, 1.08, 1.34],
  [1, 1.16, 1.26, 1.54],
  [1, 1.22, 1.07, 1.03],
  [1, 0.93, 0.97, 0.92],
  [1, 1.11, 1.56, 1.52],
  [1, 0.76, 0.77, 0.90],
  [1, 0.92, 0.84, 1.01],
  [1, 0.88, 1.22, 1.34]]

```

Die Prozedur **ValueAmerican()** wird aufgerufen:

```

P, E, S, C, RegCoeff := ValueAmerican(Stock, Time,
  x->max(1.1-x,0), a+b*x+c*x^2, 0.06)

```

und gibt folgende Werte zurück:

```
P:=0.1144343300
```

Der Preis stimmt exakt mit dem Preis aus Kapitel 2.1 überein. Der nächste Rückgabewert ist das Verhältnis der ausgeübten Pfade zu den nicht ausgeübten Pfaden:

```
E:=[[0, 0], [1, 1/2], [2, 0], [3, 1/8]]
```

Zum Zeitpunkt 0 wird klarerweise in keinem Szenario vorzeitig ausgeübt. Zu $t = 1$ werden 50% aller Szenarien, hier also vier, ausgeübt, zu $t = 2$ wird für keinen Pfad ausgeübt und zu $t = 3$ wird in einem Fall ausgeübt.

Die Matrix S enthält die berechnete Strategie. Es ist wieder ersichtlich, dass zu $t = 1$ in vier Szenarien und zu $t = 3$ in einem Szenario ausgeübt wird.

```
S:=[[0, 0, 0, 0],
     [0, 0, 0, 0],
     [0, 0, 0, 1],
     [0, 1, 0, 0],
     [0, 0, 0, 0],
     [0, 1, 0, 0],
     [0, 1, 0, 0],
     [0, 1, 0, 0]]
```

Die Matrix C enthält die berechneten Cashflows, die für die verwendeten Pfade von der Option generiert werden. Aus diesen errechnet sich der Optionspreis durch Diskontieren auf $t = 0$ und Mitteln über alle Zeilen.

```
C:=[[0, 0, 0, 0],
     [0, 0, 0, 0],
     [0, 0, 0, 0.07],
     [0, 0.17, 0, 0],
     [0, 0, 0, 0],
     [0, 0.34, 0, 0],
     [0, 0.18, 0, 0],
     [0, 0.22, 0, 0]]
```

Zuletzt wird noch eine Liste mit den verwendeten Regressionsfunktionen ausgegeben. Man beachte, dass die erste Funktion in dieser Liste jene ist, die zuerst berechnet wurde, sprich jene, die zu $t = 2$ verwendet wurde. Die geschätzten Koeffizienten stimmen wieder mit denen aus Kapitel 2.1 überein.

```
RegCoeff:=[-1.06998765437038+2.98341062378606*x-
            1.81357618181514*x^2, 2.03751234269075-
            3.33544340377013*x+1.35645658842110*x^2]
```

Abbildung 3.2 zeigt die approximierten Continuation Values für die Zeitpunkte $t = 2$ und $t = 1$.

Zu $t = 2$ wird der relevante Preisbereich $[0, 1.1]$ durch den Schnittpunkt der Regressionsfunktion mit der Auszahlungsfunktion bei 1.0004 in genau zwei Bereiche aufgeteilt: Falls der Preis S_t kleiner als 1.0004 ist, wird ausgeübt, darüber nicht. Für $t = 1$ erhalten wir die beiden Schnittpunkte $\{0.637, 1.084\}$, zwischen denen ausgeübt wird.

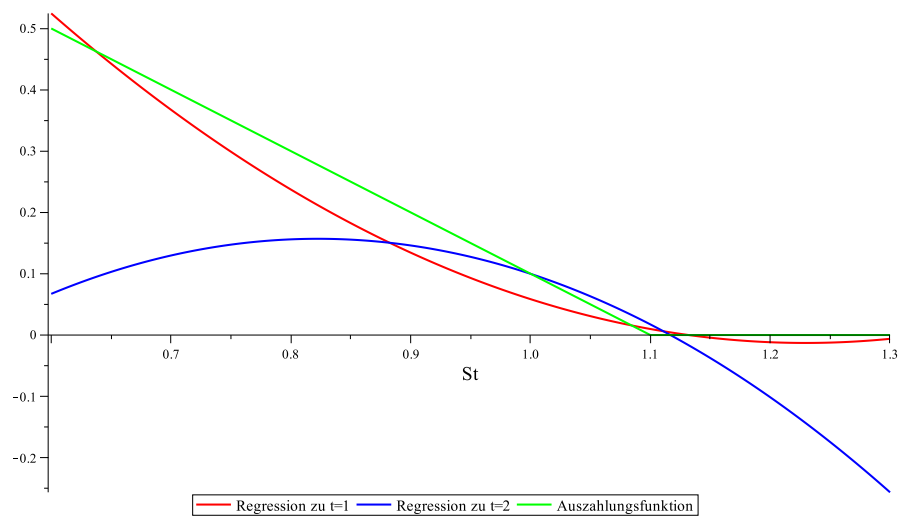


Abbildung 3.2: Regressionsfunktionen und Auszahlungsfunktion für ein Polynom 2. Grades als Ansatzfunktion

Mit der exakt selben Pfadmatrix kann nun der Algorithmus auch mit einer anderen Ansatzfunktion aufgerufen werden, beispielsweise einem Polynom 3. Grades:

```
P, E, S, C, RegCoeff := ValueAmerican(Stock, Time,
  x->max(1.1-x,0), a+b*x+c*x^2+d*x^3, 0.06)
```

Der Preis der Option beträgt nun

```
P:=0.1154327146
```

was einem um nur 0.8% höheren Preis entspricht. Die Strategie und die Cashflow Matrix hingegen weisen große Unterschiede im Vergleich zur Berechnung mit einem Polynom 2. Grades auf: Es wird nun zu jedem möglichen Zeitpunkt in mindestens einem Szenario ausgeübt, wie sowohl der Anteil der ausgeübten Pfade als auch die Strategiematrix zeigen:

```
E:=[[0, 0], [1, 3/8], [2, 1/8], [3, 1/4]]
```

```
S:=[[0, 0, 1, 0],
     [0, 0, 0, 0],
     [0, 0, 0, 1],
     [0, 0, 0, 1],
     [0, 0, 0, 0],
     [0, 1, 0, 0],
     [0, 1, 0, 0],
     [0, 1, 0, 0]]
```

Folgende Cashflows werden dadurch generiert:

```
C:=[[0, 0, 0.02, 0],
     [0, 0, 0, 0]]
```

```

[0, 0, 0, 0.07],
[0, 0, 0, 0.18],
[0, 0, 0, 0 ],
[0, 0.34, 0, 0 ],
[0, 0.18, 0, 0 ],
[0, 0.22, 0, 0 ]]
```

Für die Pfade {3, 6, 7, 8} stimmen die Cashflows mit denen von oben überein. Die veränderte Ansatzfunktion bewirkt, dass nun auch zusätzlich der Pfad 1 ausgeübt wird und für den Pfad 4 der Ausübungszeitpunkt (und damit natürlich auch die Auszahlungssumme) von $t = 1$ auf $t = 3$ verschoben wurde.

Als Regressionsfunktionen werden folgende Polynome geschätzt:

```

[49.1205341785303-162.255315488354*x+
 178.013868478294*x^2-64.7033646719150*x^3,
150.718936272216-498.351038181398*x+
 544.928535569530*x^2-196.850763514396*x^3]
```

Abbildung 3.3 zeigt zwei Approximationen des Continuation Values für den Zeitpunkt $t = 1$, einerseits durch ein Polynom 2. Grades, andererseits durch ein Polynom 3. Grades. Damit werden auch die so unterschiedlichen Ergebnisse

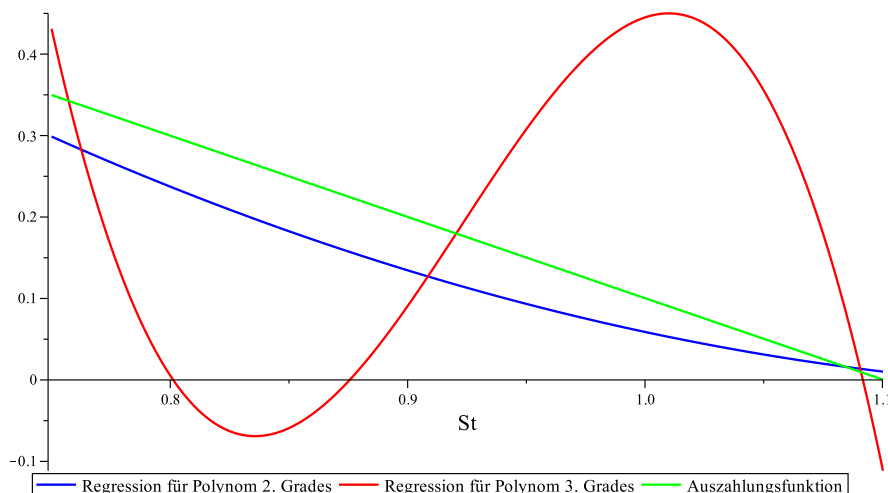


Abbildung 3.3: Regressionsfunktionen und Auszahlungsfunktion zu $t = 1$

bei der Ausübungsstrategie klar. Das Polynom 2. Grades schneidet die Auszahlungsfunktion im relevanten Bereich $[0, 1.1]$ bei $\{0.637, 1.084\}$, das Polynom 3. Grades hingegen schneidet bei $\{0.757, 0.920, 1.091\}$. Das Polynom 2. Grades schlägt also vor, dass zu $t = 1$ jene Pfade ausgeübt werden, bei denen der Aktienpreis kleiner als 1.084 liegt, also die Pfade {4, 6, 7, 8}. Außerdem sollte der Approximation nach für Preise kleiner als 0.637 die Option nicht ausgeübt werden, was rein logisch keinen Sinn macht. In diesem Beispiel ist jedoch kein so niedriger Aktienpreis enthalten.

Das Polynom 3. Grades unterteilt das Intervall $[0, 1.1]$ in insgesamt 4 Bereiche:

S_t	Entscheidung
$[0, 0.757)$	beibehalten
$[0.757, 0.920)$	ausüben
$[0.920, 1.091)$	beibehalten
$[1.091, 1.1]$	ausüben

Das theoretische Problem sind also die Bereiche $[0, 0.757)$ und $[1.091, 1.1]$. Einerseits ist es nicht sinnvoll, im niedrigen Preisbereich unter 0.757 nicht auszuüben, sondern erst ab dem Schnittpunkt. Andererseits ist es sinnlos, nach dem Beibehaltbereich $[0.920, 1.091)$ noch den kleinen Ausübungsbereich $[1.091, 1.1]$ einzufügen. In diesem Beispiel entsteht dadurch jedoch kein Fehler, da die Pfadmatrix für $t = 1$ keine Aktienpreise im problematischen Bereich enthält.

Im relevanten Preisbereich $[0, 1.1]$ ist die Ausübungsfunktion, im Falle eines Amerikanischen Puts $\max(1.1 - S_t, 0)$ eine Gerade. Ein Polynom n -ten Grades kann somit maximal n Schnittpunkte mit der Geraden haben. Damit ergeben sich bis zu $n + 1$ Bereiche mit alternierenden Ausübungsentscheidungen. Mathematisch ist das auch korrekt, in der Praxis jedoch nicht erwünscht. Wir hätten eigentlich gerne eine eindeutige Abgrenzung zwischen dem Ausübungs- und dem Beibehaltbereich. Naheliegender ist daher ein Versuch mit einem Polynom 1. Grades - es hat maximal einen Schnittpunkt mit der Auszahlungsfunktion:

```
P, E, S, C, RegCoeff1 := ValueAmerican(Stock, Time,
x->max(1.1-x,0), a+b*x, 0.06)
```

Der Preis liegt mit $P:=0.1156115357$ nicht weit weg von den bisher berechneten Preisen. Folgende Cashflow Matrix wird berechnet:

```
C:=[[0, 0.01, 0, 0 ],
[0, 0 , 0, 0 ],
[0, 0 , 0, 0.07],
[0, 0.17, 0, 0 ],
[0, 0 , 0, 0 ],
[0, 0.34, 0, 0 ],
[0, 0.18, 0, 0 ],
[0, 0.22, 0, 0 ]]
```

Die Pfade $\{3, 6, 7, 8\}$ werden wieder wie für die ersten beiden verwendeten Ansatzfunktionen zu $t = 1$ ausgeübt, zusätzlich aber auch noch die Pfade $\{1, 4\}$. Insgesamt werden also fünf Pfade schon bereits zum Zeitpunkt $t = 1$ ausgeübt. Ein Blick auf die Regressionsfunktionen in Abbildung 3.4 zeigt wieso.

Während für $t = 2$ noch der Schnittpunkt 1.032 im relevanten Bereich $[0, 1.1]$ existiert, schneidet die Regressionsfunktion zu $t = 1$ die Auszahlungsfunktion nie und liegt immer unter dieser. Die Regression schlägt also vor, dass zu $t = 1$ immer vorzeitig ausgeübt wird (sofern $S_t \leq 1.1$ ist). Die Problematik der Schnittpunkte tritt also bei allen bisher verwendeten Polynomen auf.

Der echte Wert des Continuation Values ist für die Bewertung der Option eher unerheblich, nur zwei Eigenschaften zählen: Der - im Idealfall einzige - Schnittpunkt mit der Auszahlungsfunktion sowie seine Lage im Vergleich zur Auszahlungsfunktion.

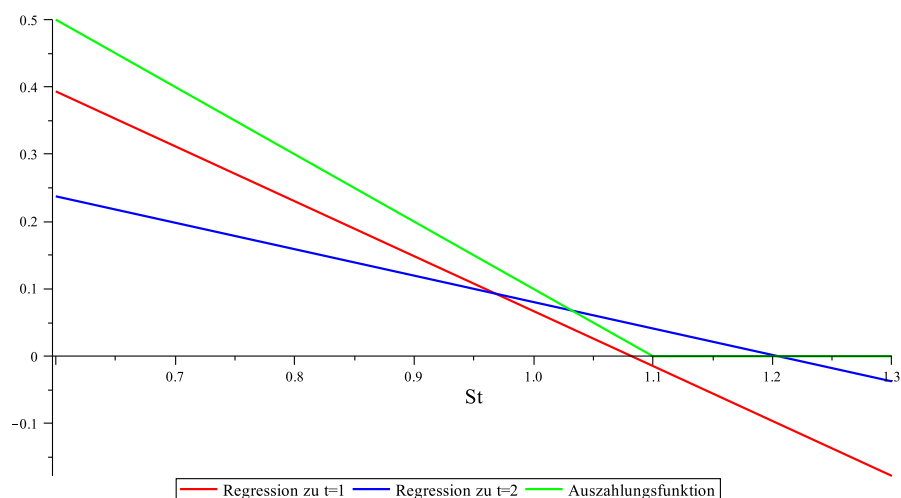


Abbildung 3.4: Regressionsfunktionen und Auszahlungsfunktion für ein Polynom 1. Grades als Ansatzfunktion

Die konstruierte Abbildung 3.5 verdeutlicht diese Problematik. Angenommen der echte Continuation Value ist bekannt (blau). Dieser teilt den relevanten Preisbereich (hier $[0, 1]$) in zwei Teile, den Ausübungsbereich $[0, 0.8]$ und den Beibehaltungsbereich $[0.8, 1]$. Approximation 1 (grün) nähert den echten Continuation Value (blau) nur sehr schlecht an, Approximation 2 (rot) hingegen verläuft viel näher am Continuation Value, schneidet dabei aber auch die Auszahlungsfunktion mehrmals. Approximation 1 schneidet die Auszahlungsfunktion einmal und das auch noch in der Nähe des Schnittpunktes zwischen Continuation Value und Auszahlungsfunktion. Vor dem Schnittpunkt ist Approximation 1 kleiner als die Auszahlungsfunktion, danach größer - genau wie der echte Continuation Value. Sie unterteilt somit den relevanten Preisbereich (hier $[0, 1]$) in nur zwei Intervalle, die Ausübungsstrategie kann eindeutig bestimmt werden. Approximation 2 liefert drei Intervalle und diese decken sich kaum mit den vom echten Continuation Value erhaltenen Intervallen. Somit eignet sich in diesem Fall Approximation 1 besser für die Bepreisung und Berechnung der optimalen Strategie der Option zu diesem Zeitpunkt.

3.4 Vergleich verschiedener Ansatzfunktionen

Wir können nun verschiedene Ansatzfunktionen für die Regression miteinander vergleichen. Dazu wird eine Amerikanische Put Option bepreist. Folgende Funktionen werden dafür verwendet:

Neben den klassischen Polynomen ersten bis vierten Grades betrachten wir hier auch Laguerre Polynome zweiten Grades sowie Hermitesche Polynome dritten Grades. Für die Berechnung wird zuerst eine Pfadmatrix mit der Funktion `SimStockMatrix()` erstellt und basierend auf dieser die Funktion `ValueAmerican()` mit den angeführten Ansatzfunktionen aufgerufen. Das heißt, die Preise werden auf den gleichen Pfaden berechnet und damit der Einfluss

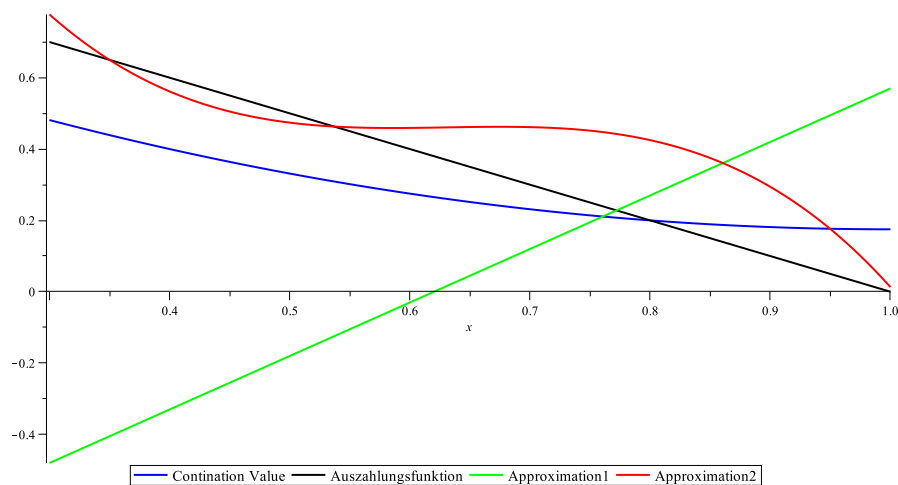


Abbildung 3.5: Zwei Approximationen an den Continuation Value

Abkürzung	Gestalt
Poly1	$a_0 + a_1x$
Poly2	$a_0 + a_1x + a_2x^2$
Poly3	$a_0 + a_1x + a_2x^2 + a_3x^3$
Poly4	$a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4$
Lagu2	$a_0 + a_1(1 - x) + a_2 \left(1 - 2x + \frac{x^2}{2}\right)$
Herm3	$a_0 + a_12x + a_2(4x^2 - 2) + a_3(8x^3 - 12x)$

Tabelle 3.1: Verwendete Ansatzfunktionen für numerische Tests des LSM Algorithmus

der Pfade auf den Preis reduziert.

Als Strike Preis wurde $K = 40$ gewählt und der risikolose Zins beträgt $r = 0.06$. Es ist möglich, die Option zu 50 Zeitpunkten pro Zeiteinheit auszuüben. Die Simulation basiert auf $N = 1000$ Pfaden. Neben den Preisen für die diversen Ansatzfunktionen wird auch der mithilfe der Black-Scholes Formel berechnete Preis (BS) einer europäischen Put Option angegeben.

Als Referenzpreis findet sich in der letzten Spalte der in [LS], Table 1 angegebene Preis (LS), der auf 100000 Pfaden und - vermutlich - Laguerre Polynomen 2. Grades als Ansatzfunktionen basiert. Wobei aus [LS] nicht ganz klar hervorgeht, welche Polynome tatsächlich zur Berechnung der Preise verwendet wurden. Wenn, wie eigentlich beschrieben, gewichtete Laguerre Polynome 2. Ordnung verwendet wurden, dann erscheinen die Ergebnisse nicht plausibel. Mit Laguerre Polynomen 2. Ordnung ohne die Gewichtsfunktion hingegen sind die präsentierten Ergebnisse sehr einfach verifizierbar.

Die Preise für das Laguerre Polynom 2. Grades und das klassische Polynom 2. Grades stimmen klarerweise überein, ebenso für das Hermitesche Polynom 3. Grades und das klassische Polynom 3. Grades. Die Koeffizienten können ineinander übergeführt werden. Da das Laguerre Polynom 2. Grades dazu verwendet

Parameter			LSM mit Ansatzfunktion						BS	LS
S_0	T	σ	Poly1	Poly2	Poly3	Poly4	Lagu2	Herm3		
36	1	0.2	4.513	4.545	4.558	4.579	4.545	4.558	3.844	4.472
		0.4	7.171	7.181	7.266	7.297	7.181	7.266	6.711	7.091
	2	0.2	4.681	4.803	4.773	4.754	4.803	4.773	3.763	4.821
		0.4	8.390	8.463	8.532	8.477	8.463	8.532	7.700	8.488
38	1	0.2	3.286	3.328	3.346	3.296	3.328	3.346	2.852	3.244
		0.4	6.213	6.230	6.260	6.317	6.230	6.260	5.834	6.139
	2	0.2	3.567	3.706	3.736	3.719	3.706	3.736	2.991	3.735
		0.4	7.553	7.704	7.687	7.774	7.704	7.687	6.979	7.669
40	1	0.2	2.279	2.357	2.360	2.388	2.357	2.360	2.066	2.313
		0.4	4.991	5.063	5.264	5.271	5.063	5.264	5.060	5.308
	2	0.2	2.814	2.982	3.012	3.009	2.982	3.012	2.356	2.879
		0.4	7.021	7.218	7.209	7.182	7.218	7.209	6.326	6.921
42	1	0.2	1.627	1.605	1.623	1.640	1.605	1.623	1.465	1.617
		0.4	4.441	4.529	4.584	4.571	4.529	4.584	4.379	4.588
	2	0.2	2.267	2.263	2.287	2.308	2.263	2.287	1.841	2.206
		0.4	6.467	6.412	6.498	6.492	6.412	6.498	5.736	6.243
44	1	0.2	1.097	x	x	x	x	x	1.017	1.118
		0.4	3.802	3.893	3.885	3.932	3.893	3.885	3.783	3.957
	2	0.2	x	x	x	x	x	x	1.429	1.675
		0.4	5.767	5.841	5.907	5.902	5.841	5.907	5.202	5.622

Tabelle 3.2: Ergebnisse des LSM Algorithmus mit verschiedenen Ansatzfunktionen; Strike $K = 40$, risikoloser Zins $r = 0.06$ und 1000 Pfade

wurde, die Werte in [LS] zu berechnen, wird es hier auch angeführt.

Es zeigt sich, dass bereits mit dieser kleinen Anzahl an simulierten Pfaden die Optionspreise relativ gut approximiert werden können, wenn der von [LS] übernommene Preis als Referenz herangezogen wird. Hervorzuheben ist weiters, dass auch mit Polynomen 1. Grades der berechnete Preis nahe an der Referenz liegt. Eine höhere Ordnung der Basisfunktion bringt also nicht immer einen Vorteil. Ein möglicher Grund könnte sein, dass Ansatzfunktionen höherer Ordnung die Auszahlungsfunktion öfters schneiden und damit keine klare Trennung des Preisbereichs in Beibehaltbereich und Ausübungsbereich zustande bringen (vgl. dazu auch Abbildung 3.5).

Liegt der Ausgangspreis S_0 nahe oder über dem Strikepreis K , verringert sich die Anzahl der in the money Pfade, da durch den positiven Zins ein mittlerer Wertzuwachs erwartet wird und sich die Pfade entsprechend verhalten. Es kann dadurch vorkommen, dass zu wenige Beobachtungen für die einzelnen Regressionen vorhanden sind. Der LSM Algorithmus kann dann keinen Preis berechnen (in Tabelle 3.2 mit x gekennzeichnet). Für den Startpreis $S_0 = 44$ tritt dieses Problem bei den Durchläufen mit $\sigma = 0.2$ auf. Bei höherer Volatilität ist das Spektrum der Preise größer und daher werden auch Preise mit höherer Abweichung vom Mittelwert nach unten simuliert. Die Regression findet somit ausreichend verwendbare Beobachtungen, um den Continuation Value zu approximieren. Bei [LS] tritt das Problem vermutlich aufgrund einer viel höheren Anzahl an verwendeten Pfaden nicht auf.

Tabelle 3.3 gibt die relativen Abweichungen der Preise zum Referenzpreis aus [LS] an. Zusätzlich wird zeilenweise und spaltenweise die mittlere relative

Parameter			LSM mit Ansatzfunktion				
S_0	T	σ	Poly1	Poly2	Poly3	Poly4	\varnothing_1
36	1	0.2	0.92	1.63	1.92	2.39	1.72
		0.4	1.13	1.27	2.47	2.91	1.94
	2	0.2	-2.90	-0.37	-1.00	-1.39	-1.42
		0.4	-1.15	-0.29	0.52	-0.13	-0.27
38	1	0.2	1.29	2.59	3.14	1.60	2.16
		0.4	1.21	1.48	1.97	2.90	1.89
	2	0.2	-4.50	-0.78	0.03	-0.43	-1.42
		0.4	-1.51	0.46	0.23	1.37	0.14
40	1	0.2	-1.47	1.90	2.03	3.24	1.43
		0.4	-5.97	-4.62	-0.83	-0.70	-3.03
	2	0.2	-2.26	3.58	4.62	4.52	2.61
		0.4	1.44	4.29	4.16	3.77	3.42
42	1	0.2	0.62	-0.74	0.37	1.42	0.42
		0.4	-3.20	-1.29	-0.09	-0.37	-1.24
	2	0.2	2.77	2.58	3.67	4.62	3.41
		0.4	3.59	2.71	4.08	3.99	3.59
44	1	0.2	-1.88	x	x	x	-1.88
		0.4	-3.92	-1.62	-1.82	-0.63	-2.00
	2	0.2	x	x	x	x	x
		0.4	2.58	3.90	5.07	4.98	4.123
\varnothing_2			-0.70	0.93	1.70	1.89	

Tabelle 3.3: Relative Abweichungen zu den Ergebnissen aus [LS] in % mit verschiedenen Ansatzfunktionen; Strike $K = 40$, risikoloser Zins $r = 0.06$ und 1000 Pfade

Abweichung angegeben. Es zeigt sich, dass die relative Abweichung mit der Ordnung der Basisfunktion steigt. Das Polynom 1. Grades hat die geringste Abweichung von -0.7%, das Polynom 4. Grades bereits 1.89%. Zeilenweise ist das Bild nicht ganz so klar. Die Abweichungen sind sowohl positiv als auch negativ und weisen keine eindeutige Tendenz auf. Tabelle 3.4 gibt die Korrelationskoeffizienten der durchschnittlichen zeilenweisen Abweichung zu den drei Parametern S_0 , T und σ an, womit sich erkennen lässt, dass die Abweichung am stärksten von der Fälligkeit T abhängt.

	S_0	T	σ
Korrelation zu \varnothing_1	0.028	0.318	0.017

Tabelle 3.4: Korrelationskoeffizienten der spaltenweisen Abweichung zu den Parametern

3.5 Early Exercise Premium

Die Differenz zwischen dem Preis der Amerikanischen Option und der dazugehörigen Europäischen Option kann als Early Exercise Premium interpretiert

werden, also als Preis des Rechts, auch vor dem Ablaufdatum die Option bereits ausüben zu dürfen. Tabelle 3.5 enthält die Early Exercise Premium für die oben bereits berechneten Preise, wobei sie die Differenz des Preises der Amerikanischen Option berechnet mit einem Polynom 2. Grades und dem Black-Scholes Put Preis ist. Sie ist einerseits relativ zum Preis der Amerikanischen Option (EEP rel) und andererseits absolut (EEP abs) angegeben.

Klarerweise ist sie von der Laufzeit der Option abhängig. Je größer diese ist, desto höher ist auch die Early Exercise Premium, da das Recht auf vorzeitige Ausübung mehr Möglichkeiten beinhaltet die Option gewinnbringend auszuüben. Dies ist auch sofort offensichtlich in Tabelle 3.5. Tabelle 3.6 bestätigt dies, wobei die Korrelation der relativen Early Exercise Premium mit der Laufzeit sogar noch stärker ist als jene der absoluten.

Weiters zeigt sich, dass der Preis des Underlyings negativ mit der Early Exercise Premium korreliert ist. Je höher also der Startpreis ist, desto kleiner ist die Early Exercise Premium. Das kann dadurch erklärt werden, dass die Wahrscheinlichkeit, dass die Option in the money ist, für höhere Startpreise kleiner ist als bei niedrigeren Startpreisen. Dadurch ist auch die Wahrscheinlichkeit, dass vorzeitig ausgeübt werden kann, kleiner und das Recht auf vorzeitige Ausübung weniger wert.

Parameter						
S_0	T	σ	Poly2	BS	EEP rel	EEP abs
36	1	0.2	4.545	3.844	15.42%	0.701
		0.4	7.181	6.711	6.55%	0.47
	2	0.2	4.803	3.763	21.65%	1.04
		0.4	8.463	7.7	9.02%	0.763
38	1	0.2	3.328	2.852	14.30%	0.476
		0.4	6.23	5.834	6.36%	0.396
	2	0.2	3.706	2.991	19.29%	0.715
		0.4	7.704	6.979	9.41%	0.725
40	1	0.2	2.357	2.066	12.35%	0.291
		0.4	5.063	5.06	0.06%	0.003
	2	0.2	2.982	2.356	20.99%	0.626
		0.4	7.218	6.326	12.36%	0.892
42	1	0.2	1.605	1.465	8.72%	0.14
		0.4	4.529	4.379	3.31%	0.15
	2	0.2	2.263	1.841	18.65%	0.422
		0.4	6.412	5.736	10.54%	0.676
44	1	0.2	x	1.017	x	x
		0.4	3.893	3.783	2.83%	0.11
	2	0.2	x	1.429	x	x
		0.4	5.841	5.202	10.94%	0.639

Tabelle 3.5: Early Exercise Premium der amerikanischen Option

	S_0	T	σ
Korrelation zu EEB abs	-0.471	0.701	-0.115
Korrelation zu EEB rel	-0.267	0.541	-0.713

Tabelle 3.6: Korrelationskoeffizienten der spaltenweisen Abweichung zu den Parametern

3.6 Merton's Jump Diffusion als Modell für den Aktienpreis¹

Ein großer Vorteil der LSM Methode ist es, dass für den Aktienpreis unterschiedlichste Modelle zugrunde gelegt werden können. Exemplarisch betrachten wir nun ein Modell mit Sprüngen. Dabei geht es weniger um möglichst große Realitätstreue, als um das Veranschaulichen des LSM Algorithmus.

Das verwendete Modell ist unter dem Namen Merton's Jump Diffusion bekannt und erweitert das klassische Black-Scholes Aktienpreismodell um Sprünge aus einem zusammengesetzten Poissonprozess.

Die folgende stochastische Differentialgleichung beschreibt den Aktienpreis:

$$dS_t = \mu S_t dt + \sigma S_t dW_t + S_t dJ_t \quad (3.1)$$

W_t ist eine Brownsche Bewegung, μ die Drift und σ die Volatilität. Der Prozess J_t ist ein zusammengesetzter Poissonprozess der Form

$$J_t = \sum_{i=1}^{N_t} Y_i \quad (3.2)$$

N_t ist die Anzahl der Sprünge bis t , deren Ankunftszeit τ_i exponentialverteilt mit Parameter λ ist. $T_n = \sum_{i=1}^n \tau_i$ und somit $N_t = \sum_{n=1}^{\infty} \mathbb{1}_{t \geq T_n}$. Die Sprunghöhen Y_i sind unabhängig und identisch verteilt mit $Y_i \sim N(\xi, \rho)$.

Die Gleichung (3.1) besitzt die eindeutige Lösung

$$S_t = S_0 \exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t + J_t\right) \quad (3.3)$$

Um ein arbitragefreies Modell für den Aktienpreis zu erhalten müssen wir die Drift noch um die erwarteten Sprünge korrigieren. Wir nehmen an, die Drift μ setzt sich zusammen als $\mu = \hat{\mu} + \mu_J$, wobei μ_J vom Sprungprozess kommt. $\hat{\mu}$ ist bekanntlich die risikolose Drift r , also der risikolose Zinssatz. Pro Zeiteinheit erwarten wir $\frac{1}{\lambda}$ Sprünge und $\mathbb{E}[\exp(Y_i) - 1]$ ist die zufällige prozentuelle Änderung des Aktienpreises, falls ein Sprung auftritt. Da die Y_i identisch und unabhängig normalverteilt sind, ist $\exp(Y_i)$ lognormalverteilt und $\mathbb{E}[\exp(Y_i) - 1] = \exp\left(\xi + \frac{\rho^2}{2}\right) - 1$.

Somit erhalten wir für den Aktienpreis das arbitragefreie Modell

$$S_t = S_0 \exp\left(\left(r - \frac{\sigma^2}{2} - \frac{1}{\lambda} \left(\exp\left(\xi + \frac{\rho^2}{2}\right) - 1\right)\right)t + \sigma W_t + J_t\right) \quad (3.4)$$

¹vgl. [M], [TV] und [EQ]

Die Prozedur `SimStockJMatrix()` erzeugt eine Pfadmatrix, mit Aktienkursen nach einer Jump Diffusion. Die Eingabeparameter sind groÙtenteils analog zum klassischen Black-Scholes Modell: `n` ist die Anzahl der Zeitschritte, `T` der Zeithorizont, `r` der risikolose Zins, `S0` der Startpreis und `N` die Anzahl der zu simulierenden Pfade. AuÙerdem werden der Intensitatsparameter `lambda` fur den Poissonprozess sowie Erwartungswert `xi` und Standardabweichung `rho` fur die Sprunghohenverteilung verlangt.

```
SimStockJMatrix := proc (n, T, r, sigma, S0, lambda, xi, rho, N)
  local t, S, W, k, Y, Nt, Jt, Tn;
  S := Matrix(N, n+1);
  t := <0, seq(j*T/n, j = 1 .. n)>;
  for k from 1 to N do;
    W := makeBB(n, T);
```

Nun wird der zusammengesetzte Poissonprozess erzeugt:

```
    Tn := [Sample(RandomVariable(Exponential(lambda)), 1)[1]];
    while Tn[nops(Tn)] < T do;
      Tn := [op(Tn), Tn[nops(Tn)]+
            Sample(RandomVariable(Exponential(lambda)), 1)[1]];
    end do;
    Nt := t->add(piecewise(Tn[i] < t, 1), i = 1 .. nops(Tn));
    Y := Sample(RandomVariable(Normal(xi, rho)), Nt(T));
    Jt := [seq(add(Y[j], j = 1 .. Nt(i*T/n)), i = 1 .. n)];
```

Aus Brownscher Bewegung und dem zusammengesetzten Poissonprozess kann nun der gesamte Pfad errechnet werden:

```
    S[k, 1 .. n+1] := <S0, seq(S0*exp((r-(1/2)*sigma^2-
      (exp(xi+(1/2)*rho^2)-1)/lambda)*W[i, 1]+
      sigma*W[i, 2]+Jt[i]), i = 1 .. n)>;
  end do;
  return t, S;
end proc;
```

Abbildung 3.6 zeigt Pfade einer Jump Diffusion im Vergleich zu Pfaden ohne Sprunge. Beide Male wurde ein risikoloser Zins von $r=0.06$ und Volatilitat $\sigma=0.2$ verwendet. Die Sprungparameter sind $\lambda=0.1$, $\xi=-0.2$ und $\rho=0.07$. Wir erwarten somit pro Zeiteinheit $\frac{1}{\lambda} = 10$ Sprunge und die Sprunghohe hat einen kleinen negativen Erwartungswert. Die Sprunge sollten somit eher nach unten gehen.

In beiden Fallen ist der mittlere Return aller Pfade etwa der risikolose Zins. Die mittlere totale Variation der Pfade betragt jedoch im Black-Scholes Fall 32.3, im Jump Diffusion Fall 101.2. Die gesamte Volatilitat ist im Fall von Sprungen viel hoher und wir erwarten somit auch hohere Optionspreise.

Auf diesen simulierten Pfaden wollen wir nun Optionspreise berechnen. Als Parameter wahlen wir die schon in Kapitel 3.4 verwendeten Werte, um so auch einen Vergleich zum Modell ohne Sprunge ziehen zu konnen. Den Parameter `lambda` fixieren wir mit 0.1 und `xi` mit -0.1. Als Ansatzfunktion wahlen wir ein Polynom 2. Grades. Die restlichen fixen Parameter nehmen die schon oben

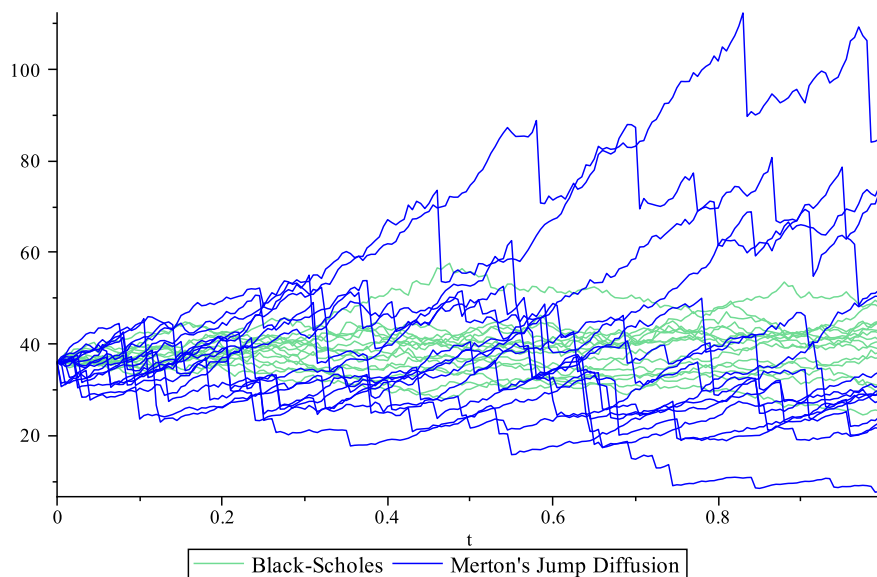


Abbildung 3.6: Pfade einer Jump Diffusion im Vergleich zu klassischen Black-Scholes Pfaden

verwendeten Werte an: $K=40$, $r=0.06$, $N=1000$ und wir verwenden $n=50$ vorzeitige Ausübungszeitpunkte pro Zeiteinheit.

Tabelle 3.8 enthält die für die bereits bekannten Konfigurationen errechneten Preise. Neben den Preisen für zwei unterschiedliche Standardabweichungen $\rho \in \{0.03, 0.09\}$ für die Sprunghöhe ist zum Vergleich wieder der in [LS] angegebene Preis enthalten. Durch die Sprungkomponente erhalten wir mehr Pfade, die in the money sind und damit ergeben sich auch keine Berechnungsprobleme für höhere Startpreise, wie es der Fall in Tabelle 3.2 war.

Die verwendete Jump Diffusion setzt sich wie oben erklärt aus einer Brownschen Bewegung und einem zusammengesetzten Poissonprozess für die Sprünge zusammen. Im Vergleich mit den Preisen aus [LS], welche auf einer reinen geometrischen Brownschen Bewegung basieren, können wir einen Preis für die Sprungkomponente, sozusagen eine Sprungprämie, berechnen. Diese erhöht den Wert der Option, da sie zusätzliche Volatilität und damit mehr potentielle Ausübungsmöglichkeiten hinzufügt.

Betrachten wir nun wieder die Korrelationen der Eingabeparameter zu den Preisen (Tabelle 3.7). Die Abhängigkeit von der Volatilität der zugrunde liegenden geometrischen Brownschen Bewegung ist wie erwartet positiv, wobei auffällt, dass sie im Falle einer Jump Diffusion etwas weniger ausgeprägt ist wie im Falle einer reinen geometrischen Brownschen Bewegung. Das ist dadurch zu erklären, dass σ nun nicht mehr der einzige Parameter ist, der die Gesamtvolatilität steuert, auch Parameter der Sprungkomponente tragen dazu bei.

Der Optionspreis ist in allen drei Fällen negativ mit dem Startpreis S_0 korreliert, was auch rein logisch Sinn macht, da eine Aktie mit geringerem Startpreis (bei gleichem Strike Preis) tiefer in the money liegt und auch eher in the money

bleibt. Auch die Korrelation mit der Laufzeit ist erwartungsgemäß positiv.

	S_0	T	σ
Korrelation zu $\rho = 0.03$	-0.553	0.478	0.581
Korrelation zu $\rho = 0.09$	-0.516	0.605	0.502
Korrelation zu LS	-0.495	0.234	0.764

Tabelle 3.7: Korrelationen der Preise zu den Eingabeparametern

Parameter			Preise			Sprungprämie	
S_0	T	σ	$\rho = 0.03$	$\rho = 0.09$	LS	$\rho = 0.03$	$\rho = 0.09$
36	1	0.2	7.306	8.028	4.472	2.834	3.556
		0.4	8.603	9.774	7.091	1.512	2.683
	2	0.2	8.236	9.773	4.821	3.415	4.952
		0.4	10.908	12.233	8.488	2.420	3.745
38	1	0.2	6.431	7.157	3.244	3.187	3.913
		0.4	8.452	9.045	6.139	2.313	2.906
	2	0.2	7.435	9.058	3.735	3.700	5.323
		0.4	10.133	11.230	7.669	2.464	3.561
40	1	0.2	5.626	6.366	2.313	3.313	4.053
		0.4	7.195	7.800	5.308	1.887	2.492
	2	0.2	7.097	8.446	2.879	4.218	5.567
		0.4	9.352	11.014	6.921	2.431	4.093
42	1	0.2	4.958	5.678	1.617	3.341	4.061
		0.4	6.402	7.058	4.588	1.814	2.470
	2	0.2	6.134	7.758	2.206	3.928	5.552
		0.4	9.183	10.183	6.243	2.940	3.940
44	1	0.2	4.222	5.073	1.118	3.104	3.955
		0.4	5.741	6.484	3.957	1.784	2.527
	2	0.2	5.634	7.154	1.675	3.959	5.479
		0.4	8.165	9.557	5.622	2.543	3.935

Tabelle 3.8: Preise für Optionen mit einer Jump Diffusion als Aktienpreismodell

Kapitel 4

Kritischer Stockpreis einer amerikanischen Put-Option

4.1 Theoretische Grundlagen¹

Wir betrachten wieder einen Finanzmarkt mit dem üblichen Rahmenkonzept: Einen Maßraum $(\Omega, \mathcal{F}, \mathbb{P})$, ein risikoloses Bankkonto B_t mit Zinsrate r sowie eine risikobehaftete Aktie S_t mit endlichem Zeithorizont $0 \leq t \leq T$. Es existiert ein risikoneutrales, äquivalentes Martingalmaß, mit dem Preise risikoneutral berechnet werden können. Die Aktie dient als Basistitel für eine amerikanische Put Option F^P mit Strikepreis K und Auszahlungsfunktion $g(x) = (K - x)^+$.

4.1.1 Proposition. *Der Preisprozess der Put Option erfüllt folgende Eigenschaften:*

(i) $F^P(t, \cdot)$ ist fallend und konvex auf \mathbb{R}^+ für alle $t \in [0, T]$

(ii) $F^P(\cdot, x)$ ist wachsend auf $[0, T]$ für alle $x \in \mathbb{R}^+$

(iii) $F^P(t, x) > 0$ für alle $(t, x) \in (0, T] \times \mathbb{R}^+$

Beweis. Die Eigenschaft (i) folgt direkt aus der Definition der Auszahlungsfunktion: zu einem festen Zeitpunkt t betrachtet ist die Put Option mehr wert, je kleiner x , also der Wert des Basistitels ist, also gilt (i).

Nun betrachten wir einen festen Preis x des Basistitels. Bei der Bewertung einer amerikanischen Option wird das Supremum über alle möglichen Stoppzeiten gebildet, also $\sup_{\tau \in \mathcal{T}_{0,T}} \mathbb{E}Z_\tau$. Nun gilt, dass aus $\tau \in \mathcal{T}_{0,t}$ folgt, dass $\tau \in \mathcal{T}_{0,s}$, falls $t \leq s$. Somit wird das Supremum über eine größere Menge gebildet und der Optionspreis ist wachsend in der Zeit, also (ii).

Für den etwas technischeren Beweis von (iii) siehe [P], Proposition 2.1. □

Da $F^P(t, 0) = K$ wegen $r \geq 0$ gilt, muss es für jedes $t > 0$ einen kritischen Aktienpreis $b^P(t)$ geben, unter dem die amerikanische Put Option vorzeitig ausgeübt werden sollte:

¹vgl. [P]

- falls $0 \leq x \leq b^P(t)$ ist, dann ist $F^P(t, x) = g(x)$
- falls $x > b^P(t)$ ist, dann ist $F^P(t, x) > g(x)$

Man beachte dabei die Ähnlichkeit zum LSM Algorithmus, auch bei diesem lautet die Entscheidungsregel, dass ausgeübt werden muss, falls die sofortige Auszahlung größer als der Continuation Value ist. Diese Ähnlichkeit wird später dazu genutzt, eine Approximation für den kritischen Preis herzuleiten.

Der kritische Preis $\{(t, b^P(t)), t \in (0, T]\}$ teilt somit den Definitionsbereich $(0, T] \times \mathbb{R}^+$ der Put Option F^P in zwei Bereiche:

- Die Continuation Region:

$$\begin{aligned} \mathcal{C}^P &:= \{(t, x) \in (0, T] \times \mathbb{R}^+, F^P(t, x) > g(x)\} \\ &= \{(t, x) \in (0, T] \times \mathbb{R}^+, x > b^P(t)\} \end{aligned} \quad (4.1)$$

- Die Exercise Region:

$$\begin{aligned} \mathcal{S}^P &:= \{(t, x) \in (0, T] \times \mathbb{R}^+, F^P(t, x) = g(x)\} \\ &= \{(t, x) \in (0, T] \times \mathbb{R}^+, x \leq b^P(t)\} \end{aligned} \quad (4.2)$$

Da F^P stetig ist, folgt, dass die Menge \mathcal{C}^P offen ist. Aus Proposition 4.1.1 folgt folgende Eigenschaft des kritischen Preises:

4.1.2 Proposition. *Der kritische Preis b^P ist wachsend in t und nach oben durch K beschränkt.*

Beweis. Da $F^P(\cdot, x)$ eine wachsende Funktion in der Zeit t ist, muss b^P auch eine wachsende Funktion in t sein. Weiters ist $F^P(t, x) > 0 = g(x)$ für $t > 0$ und $x \geq K$, daraus folgt, dass b^P nach oben durch K beschränkt ist: $b^P < K$. \square

Abbildung 4.1 zeigt die beiden Regionen für eine amerikanische Option mit Fälligkeit $T = 5$ und Strikepreis $K = 40$, getrennt durch den kritischen Preis. Oberhalb des kritischen Preises ist es optimal, die Option nicht auszuüben, sobald der Preis aber unter den kritischen Preis fällt, ist es optimal, die Option vor Laufzeitende auszuüben. Die Intuition hinter dem Verhalten des kritischen Preises ist folgende: Je weiter die Put Option von ihrem Laufzeitende entfernt ist, desto eher kann sich der Preis noch weiter zum Vorteil des Optionshalters entwickeln. Die Option muss also schon tief in the money sein, um vorzeitig ausgeübt zu werden. Nähert sich die Option ihrem Laufzeitende, so wird auch bei schon näher beim Strikepreis vorzeitig ausgeübt. Es ist kaum noch Zeit, dass sich die Option weiter zum Vorteil des Halters entwickelt und der Stock Preis könnte sich auch noch über den Strike Preis steigen, wodurch der Payoff für den Halter der Option 0 wäre.

Das Problem besteht nun darin, diesen kritischen Preis zu bestimmen. Sobald er bestimmt ist, lässt sich die Option bewerten oder der optimalen Strategie gemäß handeln. Die Option kann bei Kenntnis des kritischen Preises für jeden Zeitpunkt etwa mit Monte Carlo Methoden bewertet werden. Und hierin besteht auch der enge Zusammenhang mit dem LSM Algorithmus. Für jeden der N Pfade wird jener Zeitpunkt gesucht, zu dem zum ersten Mal der kritische Preis unterschritten wird, dann die Option ausgeübt und über alle diskontierten

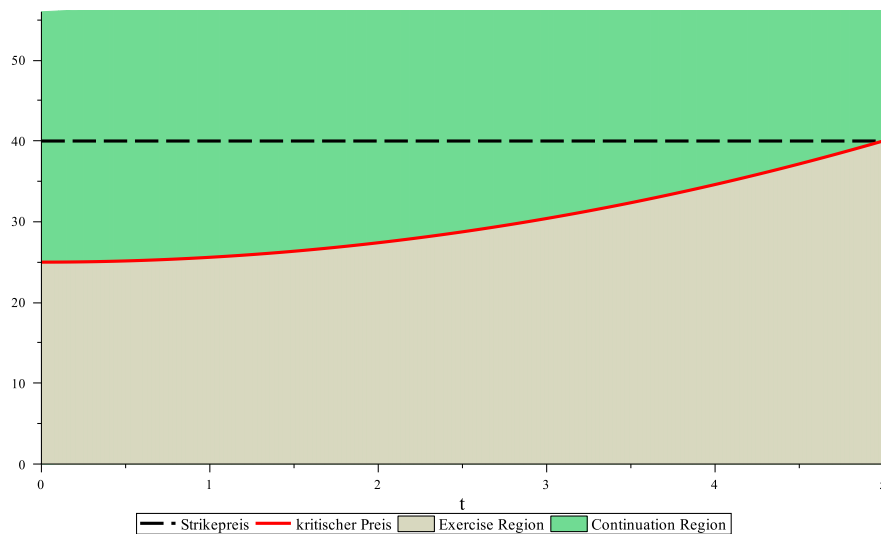


Abbildung 4.1: Continuation Region und Exercise Region einer amerikanischen Put Option

Cashflows gemittelt. Der kritische Preis ist somit nichts anderes als der Schnittpunkt des Continuation Values mit der Auszahlungsfunktion. Ersterer wird im LSM Algorithmus durch polynomiale Regression geschätzt und die Auszahlungsfunktion ist bekannt. Wir erhalten aus dem LSM Algorithmus also auch eine numerische Approximation des kritischen Preises zu jedem Diskretisierungszeitpunkt t_k .

Eine solcher kritischer Preis existiert also, aber die Berechnung dieses Preises ist nur eine andere Ausprägung des Problems, die Option selbst zu bepreisen. Es gibt keine geschlossene, analytische Form dieses kritischen Preises, ebenso wie es nicht möglich ist, eine einfache Formel für den Preis der amerikanischen Option anzugeben (im Gegensatz zu europäischen Optionen, die ja bekanntlich mittels der Black-Scholes Formel bepreist werden können). Es gibt aber mehrere einfache Approximationen des kritischen Preises nahe dem Laufzeitende.

4.2 Approximationen in geschlossener Form

Barles [B] schreibt, dass zwei Arten von Ergebnissen des kritischen Preises existieren: einerseits implizite Integrodifferentialgleichungen und andererseits analytische Approximationen. Zwei solche analytische Approximationen werden hier untersucht.

4.2.1 Approximation nach Barles²

Für $t < T$, aber nahe bei T gibt [B] eine erste Approximation an:

$$b^P(t) \sim K \left(1 - \sigma \sqrt{(T-t)|\ln(T-t)|} \right) \quad (4.3)$$

wobei K der Strike Preis der Put Option und σ die Volatilität im Black-Scholes Modell ist. Diese Approximation wird aus der partiellen Differentialgleichung des Optionspreises hergeleitet. Nahe bei T werden eine obere und eine untere Schranke der Lösung konstruiert, die in $(T-t)$ gegen die echte Lösung konvergieren.

4.2.2 Approximation nach Kuske und Keller³

[KK] gehen von der Approximation nach [B] aus und verbessern das Argument im Logarithmus. Für $t < T$, aber nahe bei T geben sie folgende Approximation an:

$$b^P(t) \sim K \left(1 - \sigma \sqrt{2(T-t) \ln \left(\frac{\sigma^2}{6\sqrt{\pi r^2 \sigma^2 (T-t)/2}} \right)} \right) \quad (4.4)$$

Ausgegangen wird vom Randwertproblem für den Preis einer amerikanischen Put Option:

$$-P_t = \frac{\sigma^2 S^2}{2} P_{SS} + rSP_S - rP, \quad 0 < t < T, \quad S \geq b^P(t) \quad (4.5)$$

$$P(S, t) = K - S, \quad S \leq b^P(t) \quad (4.6)$$

$$b^P(T) = K \quad (4.7)$$

$$P(S, T) = 0, \quad S \geq K \quad (4.8)$$

$$P(b^P(t), t) = K - b^P(t), \quad (4.9)$$

$$P_S(b^P(t), t) = -1, \quad 0 \leq t \leq T \quad (4.10)$$

Gleichung (4.6) besagt dabei wieder, dass die Option ausgeübt wird, sobald der Preis unter den kritischen Preis fällt, (4.7) ist die Endbedingung für den kritischen Preis analog zu oben. (4.8) gibt die Auszahlungsfunktion der Put Option wieder. Laut (4.9) wird genau beim kritischen Preis $b^P(t)$ ausgeübt, was den Payoff $K - b^P(t)$ ergibt. Und (4.10) besagt, dass eine Preisänderung der Aktie, ausgehend vom Preis $b^P(t)$ eine gleichgroße Änderung des Optionspreises in entgegengesetzte Richtung bewirkt. Das Randwertproblem beschreibt also genau die Bedingungen, die wir an den kritischen Preis stellen. [KK] lösen dieses Randwertproblem mithilfe der Greenschen Funktion asymptotisch und erhalten dadurch eine Approximation für $b^P(t)$ für kleine $(T-t)$.

Diese Approximation des kritischen Preises ist nur für gewisse Parameterwerte brauchbar, wie schon aus ihrer Definition in Gleichung (4.4) ersichtlich ist. Das Argument der innersten Wurzel ist immer positiv und somit problemlos. Ist jedoch

$$\frac{\sigma^2}{6\sqrt{\pi r^2 \sigma^2 (T-t)/2}} < 1 \quad (4.11)$$

²vgl. [B]

³vgl. [KK]

so ist der Logarithmus darüber negativ und damit die äußere Wurzel komplex. Daraus lassen sich Bedingungen für r und σ herleiten:

$$\begin{aligned}
 & \frac{\sigma^2}{6\sqrt{\pi r^2 \sigma^2 (T-t)}/2} > 1 \\
 \Leftrightarrow & \sigma^2 > 6\sqrt{\pi r^2 \sigma^2 (T-t)}/2 \\
 \Leftrightarrow & \sigma^4 > 36\pi r^2 \sigma^2 \frac{T-t}{2} \\
 \Leftrightarrow & \sigma^2 > 18\pi r^2 (T-t) \\
 \Leftrightarrow & \sigma > r \cdot \sqrt{18\pi(T-t)} \tag{4.12}
 \end{aligned}$$

$(T-t)$ soll voraussetzungsgemäß klein sein. Angenommen der kritische Preis soll im letzten Zehntel der Laufzeit approximiert werden, also $(T-t) = 0.1T$. Bei einer Option mit Laufzeit $T = 1$ ergibt das die Bedingung

$$\sigma > r \cdot \sqrt{18 \cdot 0.1 \cdot \pi} \approx 2.38 \cdot r \tag{4.13}$$

Um beispielsweise einen kritischen Preis im letzten Zehntel der Laufzeit zu erhalten, muss bei einer Zinsrate von $r = 0.06$ die Volatilität der Aktie mindestens 0.143 betragen, bei $r = 0.1$ muss $\sigma > 0.238$ sein. Diese Kriterien sind hinreichend für die Existenz der Approximation. Das bedeutet aber noch nicht, dass sie für diese Werte auch sinnvoll ist, wie ein Blick auf Abbildung 4.2 zeigt:

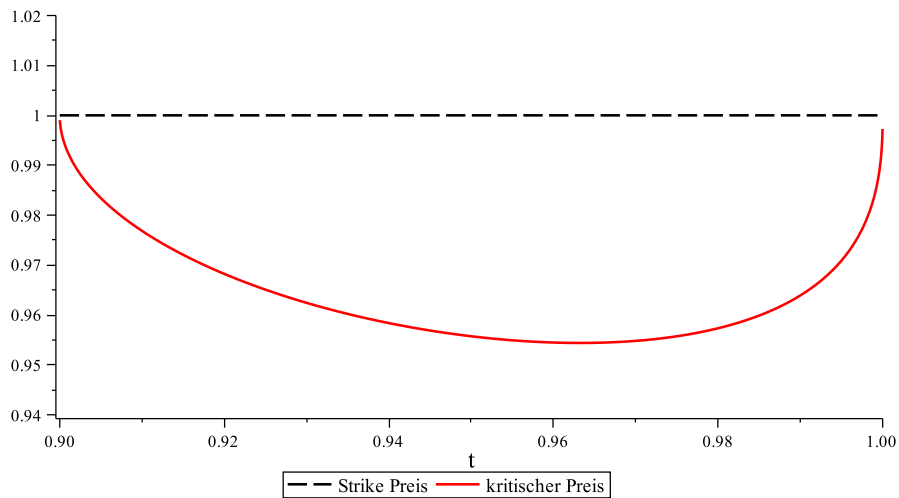


Abbildung 4.2: Kritischer Preis für $T = 1, K = 1, r = 0.1, \sigma = 0.238$

Die Volatilität wurde hier so klein gewählt, dass der approximative kritische Preis gerade noch für das letzte Zehntel der Laufzeit existiert. In der Praxis ist diese Approximation aber nicht brauchbar, da sie zuerst fällt und erst ab etwa $t = 0.97$ wieder das gewünschte, wachsende Verhalten aufweist. Bei höherer Volatilität verbessert sich das Verhalten der Approximation, wie Abbildung 4.3 zeigt, hier ist sie zumindest wachsend auf dem gesamten letzten Zehntel der Laufzeit.

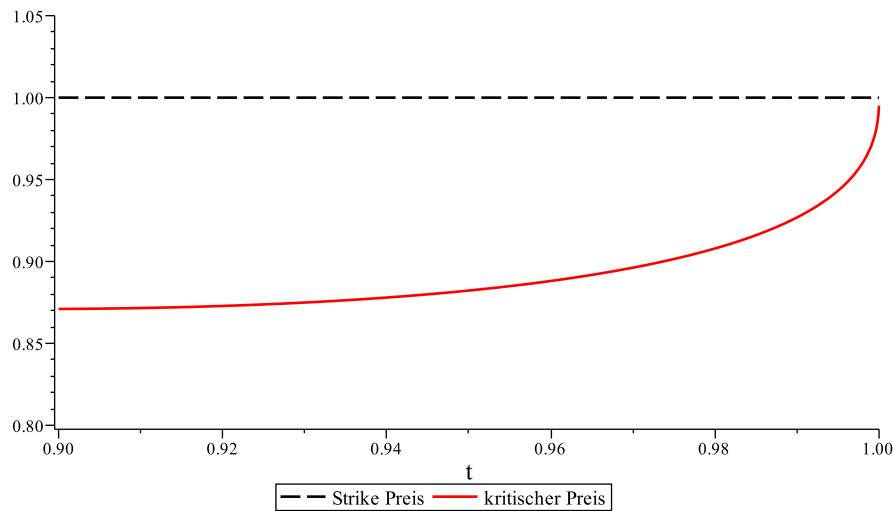


Abbildung 4.3: Kritischer Preis für $T = 1, K = 1, r = 0.1, \sigma = 0.4$

Wir können nun das Verhalten des kritischen Preises in Abhängigkeit der Volatilität anhand der Approximation studieren (Abbildung 4.4). Dabei fällt auf, dass der kritische Preis höher liegt, je kleiner die Volatilität ist. Die heuristische Interpretation davon ist wieder in Analogie zu oben die, dass eine größere Volatilität die Möglichkeit beinhaltet, dass sich der Basistitel weiter vorteilhaft für den Optionshalter entwickelt. Ein vorzeitiges Ausüben und damit ein Verzicht auf eine weitere Partizipation an der möglichen Kurssteigerung der Option sollte also tendenziell nur dann erfolgen, wenn der sofortige Payoff bereits groß ist. Gegen Ende der Laufzeit konvergiert der kritische Preis für alle möglichen Volatilitäten σ gegen den Strike Preis K . Bei größerer Volatilität muss daher die Steigung größer sein.

4.3 Numerische Berechnung

Nachdem in Kapitel 3.3 das Thema Continuation Value und seine Schnitte mit der Auszahlungsfunktion bereits kurz behandelt wurden, möchten wir diese Analyse zusammen mit den obigen Resultaten vertiefen.

Wir gehen beim Studium der numerischen Berechnung des kritischen Preises in diesem Kapitel folgendermaßen vor: Zuerst werden einige Methoden vorgestellt, von denen dann eine ausgewählt werden soll. Zusätzlich arbeiten wir zuerst mit Ansatzfunktionen verschiedener Ordnung und entscheiden uns auch hier für eine spezielle.

Wie bereits oben beschrieben ist der kritische Preis zu jedem Zeitpunkt jener Preis, ab dem es für den Halter der Option vorteilhaft ist, diese auch ausüben. Die theoretischen Resultate implizieren, dass es einen solchen Preis gibt und dass dieser auch eindeutig ist. Wir können ihn aber nicht direkt berechnen und müssen wieder auf numerische Resultate zurückgreifen. Nicht nur theoretisch ist diese Schwelle interessant, sondern in der Praxis lässt sich sehr einfach

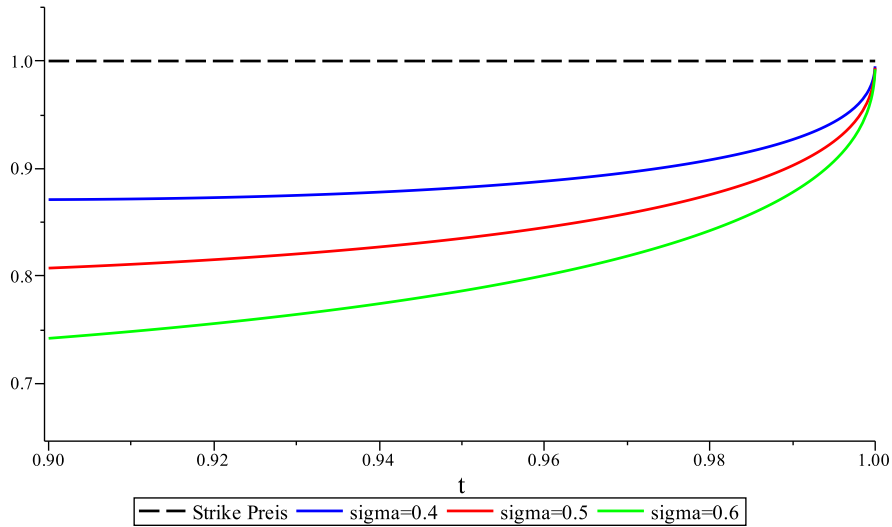


Abbildung 4.4: Kritischer Preis für $T = 1, K = 1, r = 0.1$, verschiedene σ

danach handeln: Sobald der Preis des Basistitels unter diese Schwelle fällt, soll die Option ausgeübt werden.

[LS] erwähnen den kritischen Preis unter dem Schlagwort *early exercise boundary*, behandeln ihn aber nur am Rande. Sie berechnen ihn als Lösung der Gleichung

$$\max(K - x, 0) = p_t(x). \quad (4.14)$$

Hier bezeichnet K den Strike Preis und $p_t(\cdot)$ die Regressionsfunktion zum Zeitpunkt t , sprich jene Funktion, die zu t zur Approximation des Continuation Values verwendet wird. Es wird aber kein genaueres Vorgehen erwähnt oder wie mit Problemen bei der Lösbarkeit von Gleichung (4.14) vorgegangen wird.

Wir interessieren uns hierbei nur für Lösungen im Bereich $[0, K]$. Preise kleiner 0 sind nicht möglich und für Preise größer als der Strike Preis ist der sofortige Payoff der Option 0 und somit stellt sich die Frage des vorzeitigen Ausübens nicht.

Ist die Ordnung des als Ansatzfunktion verwendeten Polynoms z , kann Gleichung (4.14) maximal z Lösungen im relevanten Intervall $[0, K]$ besitzen. Der kritische Preis ist jedoch eindeutig. Das heißt, da wir nur eine Approximation des Continuation Values kennen, können wir auch den kritischen Preis nicht eindeutig berechnen.

Die Eindeutigkeit des kritischen Preises impliziert nun, dass der Continuation Value nur einen Schnittpunkt mit der Ausübungsfunktion haben kann, Gleichung (4.14) also im relevanten Intervall $[0, K]$ eindeutig lösbar ist. Wie wir in Kapitel 2.3 gesehen haben, konvergiert die Approximation des Continuation Values gegen den echten Continuation Value. Daraus folgt, dass auch der Schnittpunkt des approximierten Continuation Values mit der Ausübungsfunktion gegen den echten Schnittpunkt konvergiert. Wir erhalten als Lösung von Gleichung (4.14) also einen konsistenten Schätzer für den kritischen Preis $b^P(t)$.

In der Praxis tritt aber trotzdem das Problem auf, dass wir oft mehr als einen Schnittpunkt erhalten. Es gibt mehrere Möglichkeiten, wie damit umgegangen werden kann. Die einfachste Möglichkeit ist, als Ansatzfunktion ein Polynom 1. Grades zu verwenden. Dieses hat maximal einen Schnittpunkt mit der Ausübungsfunktion, aber es leidet die Qualität der Approximation des Continuation Values darunter. Eine andere simple Möglichkeit besteht darin, den größten Schnittpunkt zu wählen oder über alle Schnittpunkte zu mitteln.

Tabelle 4.1 enthält für einen fixen Ausübungszeitpunkt t vier Approximationen an den Continuation Value sowie die Auszahlungsfunktion der Put Option. Zusätzlich werden die Schnittpunkte angegeben. Für das Polynom 1. Grades (Poly1) ergibt sich ein Schnittpunkt bei 62.74 und das Intervall $[0, 100]$ wird dadurch eindeutig in Exercise Region und Continuation Region geteilt. Das Polynom 2. Grades (Poly2) schneidet die Auszahlungsfunktion nicht. Das bedeutet, bei einer Optionspreisberechnung mit dieser Ansatzfunktion würde für diesen Zeitpunkt die Option nicht vorzeitig ausgeübt. Das Polynom 3. Grades (Poly3) schneidet die Auszahlungsfunktion bei 47.48 und teilt eindeutig in Continuation- und Exercise Region.

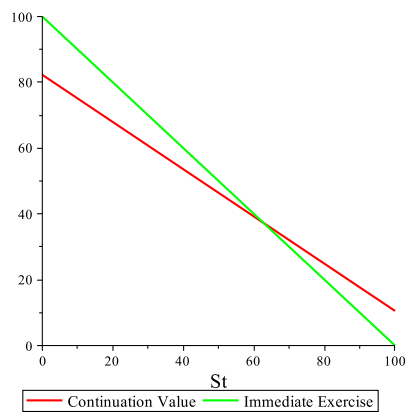
Für das Polynom 4. Grades (Poly4) erhalten wir die zwei Schnittpunkte $\{24.68, 48.58\}$. Für sehr niedrige Aktienpreise würde damit nicht ausgeübt, für Preise zwischen den Schnittpunkten wird ausgeübt und über 48.58 wird nicht ausgeübt. Wünschenswert wäre wieder eine konsistente Strategie. Wir müssen uns also für einen der beiden Schnittpunkte entscheiden und können beispielsweise den größeren Schnittpunkt 48.58 als den plausibleren bezeichnen.

Im Falle einer Put Option nehmen wir an, dass für niedrige Preise eher ausgeübt wird, für höhere Preise eher nicht. Weiters nehmen wir an, dass die Teilung des Intervalls $[0, K]$ eindeutig ist, also dass es nur eine Continuation- und eine Exercise Region gibt. Auf Basis dieser Annahmen liegt es nahe, jenen Schnittpunkt auszuwählen, bei dem der geschätzte Continuation Value die Auszahlungsfunktion von unten schneidet. Das gewährleistet, dass links vom Schnittpunkt ausgeübt und rechts davon beibehalten wird. In unserem Beispiel würden wir für Poly4 somit wieder 48.58 als den plausibleren Schnittpunkt auswählen. Auch bei Poly1 und Poly3 schneidet der Continuation Value die Ausübungsfunktion von unten und ist somit konsistent mit dieser Entscheidungsregel. Der Nachteil dieser Methode ist, dass das Problem nur für eine weitere Klasse von Ansatzfunktionen - Polynome 2. Grades - gelöst wurde und nicht allgemein. Ist die Ordnung des Ansatzpolynoms größer als zwei, können Situationen auftreten, bei denen danach keine eindeutige Entscheidung für einen Schnittpunkt möglich ist, da die Auszahlungsfunktion zweimal von unten geschnitten wird.

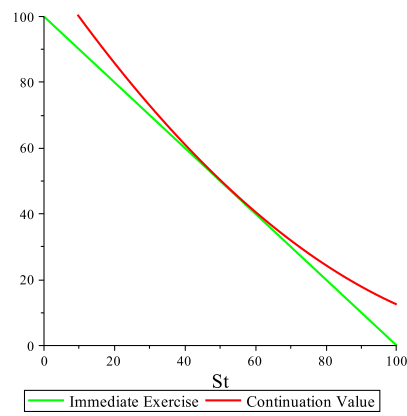
Berechnen wir nun den Schnittpunkt des approximierten Continuation Values mit der Auszahlungsfunktion für jeden Zeitpunkt t_k , so erhalten wir eine Approximation für den kritischen Preis während der gesamten Laufzeit. Mit dieser deterministischen Schranke können wir die Option optimal handeln, indem wir die Option ausüben, sobald der Preis der Aktie unter den kritischen Preis fällt.

Die folgende Maple Prozedur löst nach dem kritischen Preis und verwendet dabei die oben beschriebene Entscheidungsregel für die Auswahl des Schnittpunkts. Ist auch damit keine eindeutige Auswahl des kritischen Preises möglich, wird der größte Schnittpunkt mit obiger Eigenschaft ausgewählt.

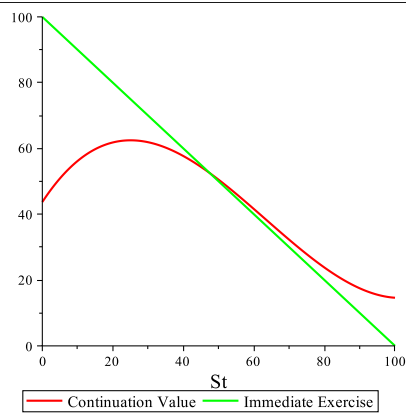
Technisch umgesetzt wird die Schnittpunktregel in der Weise, dass für eine



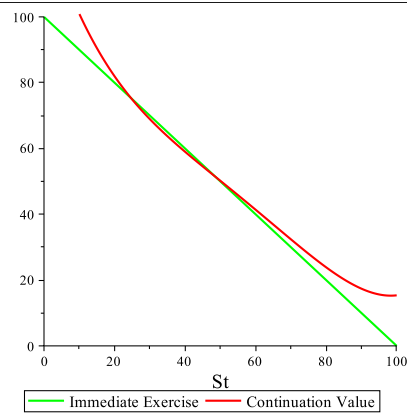
Poly1; Schnittpunkt: {62.74}



Poly2; keine Schnittpunkte



Poly3; Schnittpunkt: {47.48}



Poly4; Schnittpunkte: {24.68, 48.58}

Tabelle 4.1: Approximierter Continuation Value für verschiedene Ansatzfunktionen. Parameter: $K = 100, S_t = 100, T = 1, \sigma = 0.5, r = 0.06, L = 50, t = 0.4$

vorher definierte kleine Schranke ε überprüft wird, ob die Regressionsfunktion bei $(1 - \varepsilon) \cdot S_t$ kleiner ist als die Auszahlungsfunktion.

Die Prozedur `EEBoundary()` verlangt als Eingabewerte die Liste mit den Regressionsfunktionen `RegFkt`, wie sie von der `ValueAmerican()` Prozedur ausgegeben wird, weiters die Auszahlungsfunktion `v`, den Zeitvektor `Tv` und den Strike Preis `K`.

```
EEBoundary := proc (RegFkt, v, Tv,K)
  local eps, A, L, R, n, i, k, Lnew, temp, j;
```

Die Schranke `eps` wird definiert und einige Variablen initialisiert:

```
  eps := 0.05;
  L := [];
  n := nops(RegFkt);
```

Die Prozedur `ValueAmerican()` läuft rückwärts in der Zeit, das heißt, die erste Regressionsfunktion, die in die Liste gespeichert wird ist jene zum Zeitpunkt t_{L-1} . Also muss auch diese Prozedur rückwärts durch die Liste mit den Regressionsfunktionen laufen. Der erste kritische Preis, der nun berechnet wird, ist somit der zum Zeitpunkt t_1 .

```
  for i from n by -1 to 1 do;
```

Die in Maple integrierte Funktion `solve()` löst die übergebene Gleichung. Wir suchen jenen Punkt, an dem der geschätzte Continuation Value gleich dem Wert des sofortigen Ausübens ist. Alle erhaltenen Lösungen, ob reell oder komplex, werden in der Variable `temp` gespeichert. Aus ihnen sollen die sinnvollen ausgewählt werden.

```
    temp := [solve(RegFkt[i] = v(x), x)];
    A := [];
```

Die nächste Schleife durchläuft alle Lösungen:

```
      for k from 1 to nops(temp) do;
```

Das erste Kriterium ist, dass die Lösung in den reellen Zahlen liegt.

```
        if is(temp[k], 'real') then;
```

Weiters soll die Lösung im Intervall $[0, K]$ liegen.

```
          if 0 <= temp[k] and temp[k] <= K then;
```

Nun wird überprüft, ob die Lösung die oben definierte Schnittpunktbedingung erfüllt. Falls ja, wird sie in die Liste `A` der möglichen Lösungen aufgenommen.

```
            if evalf((unapply(RegFkt[i], x))((1-eps)*temp[k]))
              < evalf(v((1-eps)*temp[k])) then;
              A := [op(A), temp[k]];
            end if;
          end if;
        else A := [op(A), 0];
        end if;
      end do;
```

Von allen Lösungen in der Liste A, die die Schnittpunktbedingung erfüllen, wird die größte in die finale Liste L übernommen.

```

    L := [op(L), max(A, 0)];
end do;

```

Zusammen mit den Zeitpunkten aus dem Zeitvektor Tv und komplettiert mit dem letzten Punkt (T, K) wird die Liste der kritischen Preise ausgegeben.

```

    R := [seq([Tv[i+1], L[i]], i = 1 .. n), [Tv[n+2], K]];
    return R;
end proc;

```

Bevor nun die Resultate dieser Prozedur vorgestellt werden, betrachten wir noch zwei alternative Möglichkeiten, den kritischen Preis zu berechnen. Eine einfache Modifikation der obigen Prozedur `EEBoundary()` verwendet anstatt der Schnittpunktbedingung und der Maximumbildung den Mittelwert aller Schnittpunkte. Die Prozedur `EEBoundaryMean()` wird dafür verwendet. Sie verlangt die gleichen Eingabewerte wie `EEBoundary()` und ist auch ansonsten weitgehend sehr ähnlich.

```

EEBoundaryMean := proc (RegFkt, v, Tv, K)
    local eps, A, L, R, n, i, k, Lnew, temp, j, z;
    L := [];
    n := nops(RegFkt);
    for i from n by -1 to 1 do;
        temp := [solve(RegFkt[i] = v(x), x)];
        A := 0;
        z := 0;
        for k from 1 to nops(temp) do;
            if is(temp[k], 'real') then;
                if 0 < temp[k] and temp[k] < K then;

```

Anstatt die Schnittpunktbedingung zu überprüfen, werden hier zunächst alle reellen Schnittpunkte im Intervall $[0, K]$ aufsummiert.

```

                A := A+temp[k];
                z := z+1;
            end if;
        end if;
    end do;

```

Der summierte Wert wird noch durch die Anzahl der sinnvollen Schnittpunkte geteilt, was dem Mittelpunkt der Schnittpunkte entspricht.

```

    L := [op(L), A/max(z, 1)]
end do;
    R := [seq([Tv[i+1], L[i]], i = 1 .. n-1), [Tv[n+2], K]];
    return R;
end proc

```

Eine weitere Möglichkeit setzt die Prozedur `EEBoundaryV2()` um. Als Eingabewerte werden die Pfadmatrix S, die errechnete Ausübungsstrategie E und der Zeitvektor Tv verlangt. Die Prozedur sucht dann für jeden Zeitpunkt den größten Stockpreis, bei dem noch ausgeübt wird.

```
EEBoundaryV2 := proc (S, E, Tv)
  local i, j, n, m, L, zupper;
  m := ColumnDimension(E);
  n := RowDimension(E);
```

Zum Zeitpunkt 0 wird der Wert S_0 verwendet. Die Matrix E wird spalten- und dann zeilenweise durchlaufen, also vorwärts in der Zeit und zu jedem Zeitpunkt durch alle simulierten Pfade. Sie korrespondiert elementenweise mit der Preis-matrix S .

```
  L := [[Tv[1], S[1, 1]]]
  for i from 2 to m do;
```

Zu jedem Zeitpunkt verwenden wir 0 als ersten kritischen Preis, gespeichert in `zupper`. Wird in der Spalte von E ein Ausübungsindikator gefunden, wird aus der Matrix S der korrespondierende Preis mit dem bisherigen Wert von `zupper` verglichen und falls er größer ist, in `zupper` gespeichert.

```
    zupper := 0;
    for j from 1 to n do;
      if E[j, i] = 1 then;
        zupper := max(zupper, S[j, i]);
      end if;
    end do;
```

Letztendlich wird somit zu jedem Zeitpunkt der größte aller simulierten Werte gefunden, bei dem noch ausgeübt wird.

```
    L := [op(L), [Tv[i], zupper]];
  end do;
  return L;
end proc;
```

Bei der Prozedur `EEBoundaryV2()` wird somit ein tatsächlich simulierter Preis ausgegeben, während die beiden ersten Prozeduren von den Pfaden über die Regressionsfunktionen abhängen. Der erhaltene Preis muss nicht tatsächlich in den Pfaden vorkommen.

Abbildung 4.5 zeigt die simulierte Early Exercise Boundary für die angegebenen Parameter nach allen drei Methoden. Als Ansatzfunktion wurde ein Polynom 1. Grades gewählt. Die Prozedur `ValueAmerican()` liefert dafür einen Optionspreis von 16.43. Die nach `EEBoundaryV2()` berechnete Schranke wird einige Male auf 0 gesetzt, da zu diesem Zeitpunkt den simulierten Pfaden nach nicht ausgeübt wird. Mit steigender Anzahl an simulierten Pfaden tritt dieses Problem seltener auf.

Zum Vergleich wird auch die analytische Approximation in geschlossener Form (β) nach Formel (4.4) im Bereich $[0.8, 1]$ angeführt.

Die analytische Approximation verläuft bei einem wesentlich höheren Preis als die simulierten Schranken. Dass alle drei simulierten Schranken annähernd gleich verlaufen, ist eine Besonderheit dieser einfachen Ansatzfunktion. Es ist klar, dass die nach der Schnittpunktregel (`EEBPoly1`) und die nach der Mittelwertregel (`EEBMeanPoly1`) gebildeten Schranken übereinstimmen, da bei einem

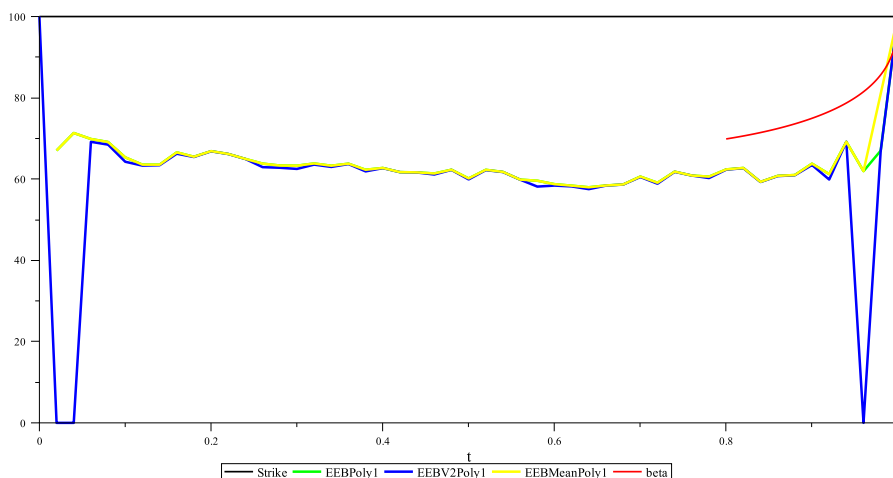


Abbildung 4.5: Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 = 100, N = 2000, L = 50$; Ansatzfunktion: Polynom 1. Grades

Polynom 1. Grades maximal ein Schnittpunkt im kritischen Bereich $[0, K]$ auftreten kann. Und da der Schnittpunkt den Bereich $[0, K]$ eindeutig in zwei Bereiche teilt, der Schnittpunkt damit jener Preis ist, ab dem nicht mehr ausgeübt wird, verläuft auch die mit `EEBoundaryV2()` berechnete Schranke annähernd gleich. Die winzigen Abweichungen sind dadurch erklärbar, dass ein wirklich simulierter Preis als Schranke gewählt wird und nicht die Lösung einer Gleichung.

Als nächstes betrachten wir den simulierten kritischen Preis bei Polynomen 2. Grades als Ansatzfunktion. Die Parameter der Option sind gleich wie vorher und es wurden auch die gleichen Pfade wie für Abbildung 4.5 verwendet.

Abbildung 4.6 zeigt die simulierte Schranke. Sofort fallen die vielen fehlenden Datenpunkte in der ersten Hälfte der Laufzeit auf. In der zweiten Hälfte der Laufzeit hingegen verläuft die Schranke annähernd glatt. Der Sprung ganz zum Ende der Laufzeit auf den Wert K ist - wie die oben angeführten Prozeduren zeigen - nicht Resultat von Berechnungen, sondern manuell hinzugefügt.

Die von `EEBoundary()` (im Diagramm mit `EEBPoly2` bezeichnet) und von `EEBoundaryV2()` (`EEBV2Poly2`) berechneten Schranken stimmen wieder weitestgehend überein. Einzig die von `EEBoundaryMean()` (`EEBMeanPoly2`) berechnete Schranke liegt signifikant tiefer. Diese bezieht wie oben beschrieben alle gefundenen Schnittpunkte mit ein und mittelt darüber. Hat der Koeffizient des quadratischen Terms positives Vorzeichen, so ist der Schnittpunkt, bei dem die Auszahlungsfunktion von unten geschnitten wird, der größere. Dieser wird der Schnittpunktregel nach für `EEBPoly2` verwendet und der zweite Schnittpunkt ignoriert. Bei `EEBMeanPoly2` wird dieser zweite, kleinere, Schnittpunkt miteinbezogen und verschiebt somit die Schranke nach unten. Zu Beginn der Laufzeit fallen einige Zeitpunkte auf, zu denen alle drei simulierten Schranken beinahe gleich verlaufen. Zu diesen Zeitpunkten schneidet der geschätzte Continuation Value die Auszahlungsfunktion im kritischen Bereich nur einmal.

Abbildungen 4.7 und Abbildung 4.8 zeigen analog zu oben noch die simulierten kritischen Preise für Polynome 3. Grades bzw. Polynome 4. Grades als

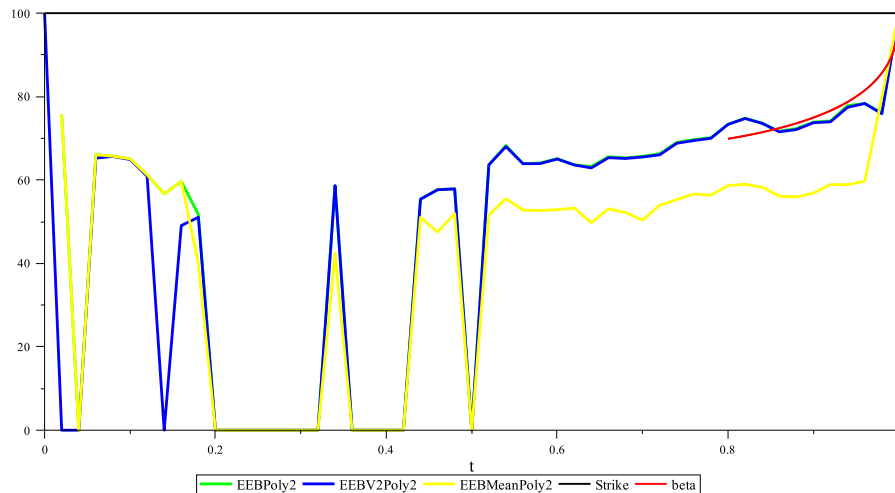


Abbildung 4.6: Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 = 100, N = 2000, L = 50$; Ansatzfunktion: Polynom 2. Grades

Ansatzfunktionen. Es zeigt sich, dass die Qualität der simulierten Schranke mit der Ordnung der Ansatzfunktion sinkt. Für Polynome 3. Grades erhalten wir erst im letzten Drittel der Laufzeit einigermaßen glatte Ergebnisse, im Falle von Polynomen 4. Grades erst im letzten Zehntel, wobei zu bemerken ist, dass sich dieses Phänomen über alle Methoden erstreckt.

Wir betrachten nun wieder die in Kapitel 3.4 untersuchten Optionen mit Startpreis $S_0 = 42$ und berechnen dazu die kritischen Preise. Abbildung 4.9 zeigt alle vier jeweils mit der Schnittpunktregel berechneten kritischen Preise basierend auf polynomialen Ansatzfunktionen 1. bis 4. Ordnung. EEBPoly1 verläuft wieder relativ tief im Vergleich zu den anderen Schranken und der analytischen Approximation (beta). EEBPoly2 weist viele Sprünge zu Beginn der Laufzeit auf, verläuft dann aber relativ glatt. EEBPoly3 zeigt ab etwa der Mitte der Laufzeit glattes Verhalten, während EEBPoly4 zu Beginn bereits relativ glatt ist aber im letzten Dritten einen großen Sprung zeigt.

Als nächstes wollen wir analysieren, wie sich die Volatilität auf die simulierten Schranken auswirkt. Dazu simulieren wir für vier Werte für σ die kritischen Preise. Als Ansatzfunktionen werden Polynome 2. Grades verwendet und die Schranke wird nach der Schnittpunktregel, also mit der Prozedur `EEBoundary()` berechnet. Zusätzlich zu den simulierten Schranken zeigt Abbildung 4.10 auch die analytischen Schranken. Je größer σ ist, desto tiefer verläuft der kritische Preis, vergleiche dazu auch Abbildung 4.4. Die simulierten Preise folgen zu Ende der Laufzeit dem von der analytischen Schranke vorgegebenen Trend. Die Schnittpunktregel wählt somit die richtigen Schnittpunkte aus und liefert einen stimmigen kritischen Preis.

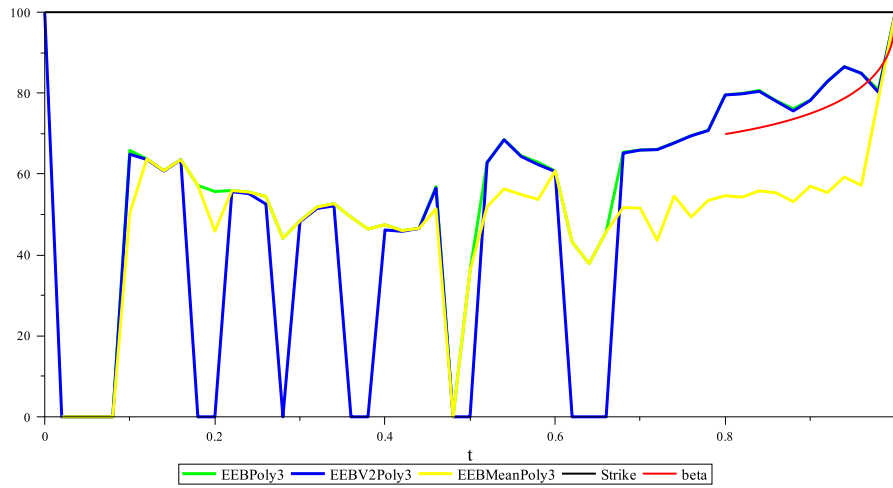


Abbildung 4.7: Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 = 100, N = 2000, L = 50$; Ansatzfunktion: Polynom 3. Grades

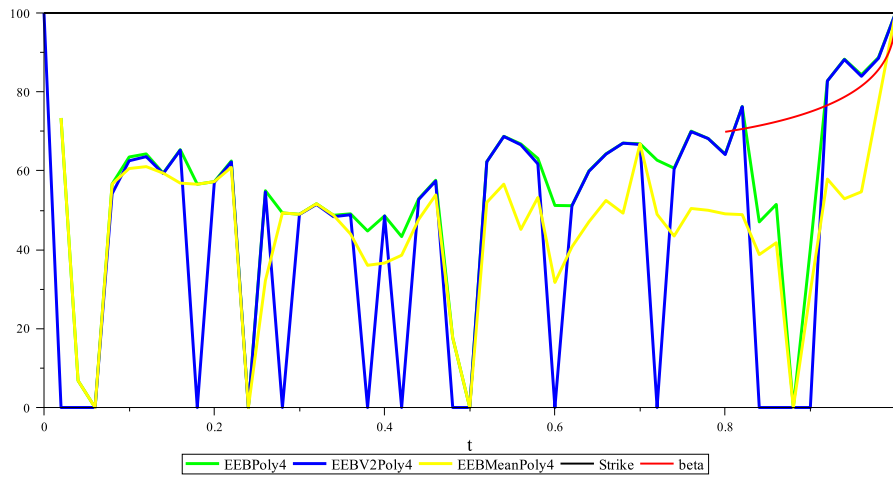


Abbildung 4.8: Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 = 100, N = 2000, L = 50$; Ansatzfunktion: Polynom 4. Grades

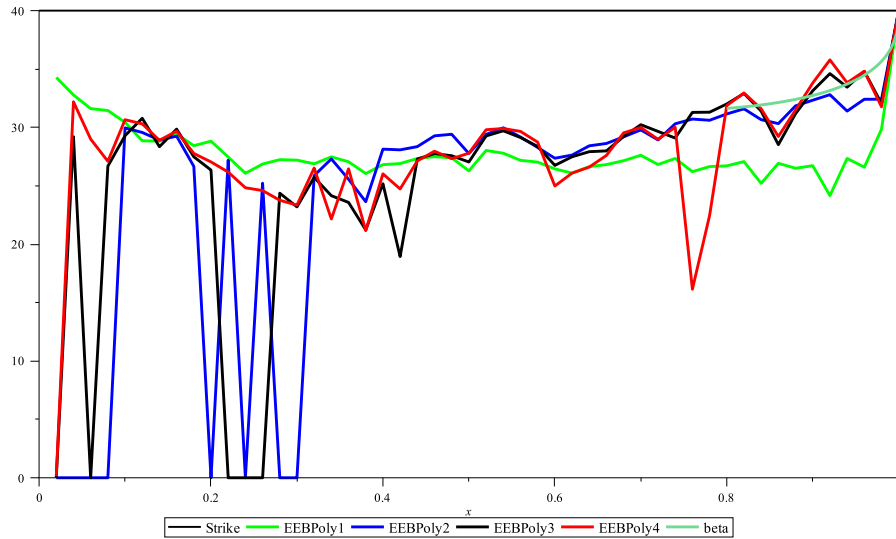


Abbildung 4.9: Kritischer Preis für $T = 1, K = 40, r = 0.06, \sigma = 0.4, S_0 = 36, N = 2000, L = 50$; verschiedene Ansatzfunktionen

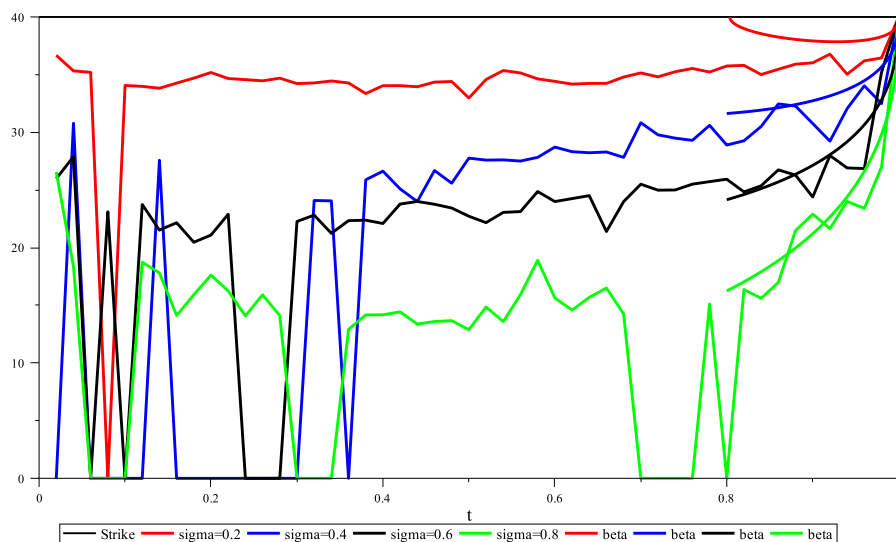


Abbildung 4.10: Kritischer Preis für $T = 1, K = 40, r = 0.06, S_0 = 40, N = 2000, L = 50$, Ansatzfunktion: Polynom 2. Grades, verschiedene Werte für σ .

4.4 Adaptionen am LSM Algorithmus

Mit dem in den vorhergehenden Kapitel erworbenen Einsichten über die Natur des Continuation Values wenden wir uns nun wieder dem ursprünglichen LSM Algorithmus zur Bepreisung von amerikanischen Optionen zu. Wir wissen nun, dass es problematisch für die Berechnung des kritischen Preises ist, wenn der geschätzte Continuation Value mehrere Schnittpunkte mit der Ausübungsfunktion hat. Das gleiche Problem tritt auch bei der Optionsbepreisung auf, wenn auch nicht so offensichtlich. Im Wesentlichen geht es dabei auch um das Suchen des kritischen Preises, unter dem die Option ausgeübt wird. Wenn nun mehrere Schnittpunkte nahe beieinander liegen, dann kann es vorkommen, dass der LSM Algorithmus die Ausübungsentscheidung falsch trifft. Dass dieses Problem vorkommt, zeigt Abbildung 4.11. Für die Bepreisung einer Option mit Parametern $K = 40$, $S_0 = 36$, $r = 0.06$, $\sigma = 0.4$, $T = 1$ und $N = 2000$ Pfade erhalten wir zum Zeitpunkt $\frac{6}{25}$ die beiden Schnittpunkte $\{23.65, 27.18\}$. Als Ansatzfunktion wurde ein Polynom 2. Grades verwendet.

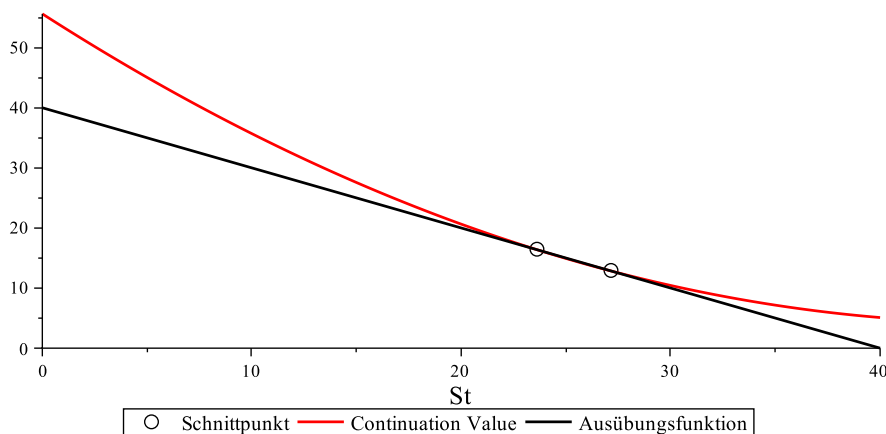


Abbildung 4.11: Continuation Value und Ausübungsfunktion; Schnittpunkte $\{23.65, 27.18\}$

Nur zwischen diesen beiden Schnittpunkten wird vorzeitig ausgeübt, für simulierte Preise kleiner als 23.18 entscheidet der Algorithmus, dass die Option beibehalten werden soll. In den 2000 simulierten Pfaden sind 24 enthalten, die zu diesem Zeitpunkt kleiner als 23.18 sind und genau für diese wird falsch entschieden.

Wir können uns nun analog zur Berechnung des kritischen Preises mit der Schnittpunktregel für einen der beiden Schnittpunkte entscheiden. Jener Schnittpunkt wird ausgewählt, bei dem der Continuation Value die Ausübungsfunktion von unten schneidet. In obigem Beispiel ist das 27.18. Damit wird eine konsistente Entscheidungsregel etabliert, welche den relevanten Preisbereich $[0, K]$ eindeutig in den Ausübungs- bzw. Beibehaltbereich teilt.

Diese Modifikation kann relativ einfach in die schon von oben bekannte Prozedur `ValueAmerican()` eingebaut werden. Wir betrachten hier die neuen Teile der Prozedur `ValueAmericanSR()`. Zur Erinnerung: j ist die Laufvariable, die

durch die Pfadmatrix rückwärts durchläuft und $E()$ ist der in jedem Schritt geschätzte Continuation Value. Zuerst werden von der Funktion `solve()` alle Schnittpunkte berechnet und diese dann analog zur Prozedur `EEBoundary()` verarbeitet. Die Konstante `eps` wird mit 0.05 festgesetzt.

```
temp := [solve(E(x) = v(x), x)];
A := [];
for k from 1 to nops(temp) do;
  if is(temp[k], 'real') then;
    if 0 <= temp[k] and temp[k] <= K then;
      if evalf(E((1-eps)*temp[k])) <
         evalf(v((1-eps)*temp[k])) then;
        A := [op(A), temp[k]];
      end if;
    end if;
    else A := [op(A), 0];
  end if;
end do;
L := max(A);
```

Ist der eindeutige Schnittpunkt L nach der Schnittpunktregel identifiziert, kann für jeden in the money Pfad entschieden werden, ob ausgeübt wird.

```
Xcont := map(x->E(x),X);
Xex := map(x->v(x),X);
z := 1;
for i from 1 to m do;
  if U[i] = 1 then;
```

Anstatt den Preis in die geschätzte Regressionsfunktion einzusetzen und mit diesem den sofortigen Payoff zu vergleichen, wird nun mit dem Schnittpunkt L verglichen. Ist der aktuelle Kurs größer als der Schnittpunkt, dann wird beibehalten, ansonsten ausgeübt.

```
  if X[z] > L then;
    for k from j to n do;
      C[i, k] := S[i, k]*v(M[i, k]);
    end do;
  else;
    S[i, j-1] := 1;
    S[i, j .. n] := 0;
    C[i, j-1] := v(M[i, j-1]);
    C[i, j .. n] := 0;
  end if;
  z := z+1;
end if;
end do;
```

Die Prozedur `ValueAmericanSR()` wird mit den gleichen Parametern wie die `ValueAmerican()` aufgerufen. Wir untersuchen nun, wie sich die Änderungen auf die berechneten Optionspreise auswirken. Dazu betrachten wir wieder das

klassische Black-Scholes Aktienpreismodell und auch Merton's Jump Diffusion. Die vom klassischen LSM Algorithmus ohne Schnittpunktregel errechneten Preise werden von oben übernommen. Tabelle 4.2 enthält die Ergebnisse des Vergleichs. Die mit LS betitelte Spalte gibt die in [LS] berechneten Preise wieder, die Spalte JD beinhaltet die Preise unter einer Jump Diffusion mit Parametern $\lambda = 0.1, \xi = -0.1, \rho = 0.03$. Die Spalte SR enthält jeweils die vom LSM mit Schnittpunktregel berechneten Preise und die Spalte Abw die relative Abweichung des SR-Preises. Der Strikepreis liegt wieder bei $K = 40$, die risikolose Zinsrate bei $r = 0.06$ und Polynome 2. Grades bilden die Ansatzfunktionen.

Parameter			Ergebnisse					
S_0	T	σ	LS	SR	Abw	JD	SR	Abw
36	1	0.2	4.472	4.502	0.68%	7.306	7.162	-1.97%
		0.4	7.091	7.329	3.36%	8.603	8.634	0.36%
	2	0.2	4.821	4.778	-0.90%	8.236	8.038	-2.41%
		0.4	8.488	8.606	1.39%	10.908	10.853	-0.51%
38	1	0.2	3.244	3.217	-0.84%	6.431	6.068	-5.65%
		0.4	6.139	6.389	4.07%	8.452	7.539	-10.80%
	2	0.2	3.735	3.784	1.31%	7.435	7.3	-1.82%
		0.4	7.669	7.501	-2.20%	10.133	10.053	-0.79%
40	1	0.2	2.313	2.313	0.01%	5.626	5.243	-6.81%
		0.4	5.308	5.471	3.07%	7.195	7.403	2.89%
	2	0.2	2.879	2.966	3.02%	7.097	6.702	-5.57%
		0.4	6.921	7.033	1.61%	9.352	9.349	-0.04%
42	1	0.2	1.617	1.601	-0.96%	4.958	4.507	-9.11%
		0.4	4.588	4.756	3.65%	6.402	6.182	-3.43%
	2	0.2	2.206	2.23	1.07%	6.134	6.127	-0.12%
		0.4	6.243	6.411	2.69%	9.183	8.729	-4.94%
44	1	0.2	1.118	x	x	4.222	4.014	-4.93%
		0.4	3.957	4.011	1.37%	5.741	5.749	0.13%
	2	0.2	1.675	x	x	5.634	5.567	-1.20%
		0.4	5.622	5.476	-2.59%	8.165	8.17	0.06%

Tabelle 4.2: Vergleich der Preise mit und ohne Schnittpunktregel

Es fällt sofort auf, dass im Falle des Black-Scholes Modells beinahe alle Abweichungen positiv sind, also die mit der Schnittpunktregel berechneten Preise sind größer als die mit dem klassischen LSM Algorithmus berechneten Preise. Im Falle der Jump Diffusion hingegen sind beinahe alle Abweichungen negativ. Auch ist die durchschnittliche absolute Abweichung im Falle von Sprüngen im Aktienkurs deutlich höher (3.18% vs. 1.74%). Tendenziell ist auch die Abweichung der Optionen mit größerer Volatilität größer, der Korrelationskoeffizient zwischen σ und der Abweichung beträgt 0.29 bzw. 0.31. Je mehr also die Preise schwanken, desto größer ist die Abweichung zwischen klassischem LSM Algorithmus und LSM mit Schnittpunktregel.

Um diese Tatsache zu verstehen, sei vorerst angenommen, der Koeffizient beim quadratischen Term der Ansatzfunktion ist positiv und beide Schnittpunkte liegen im Intervall $[0, K]$. Im Fall von Sprüngen kommt der Aktienpreis öfters in eine Region, in der der geschätzte Continuation Value für kleine Preise

bereits wieder Beibehalten der Option empfiehlt. Im Algorithmus mit Schnittpunktregel wird die Option jedoch ausgeübt, das heißt wir erhalten einen Payoff. Abbildung 4.12 verdeutlicht diese Tatsache noch einmal.

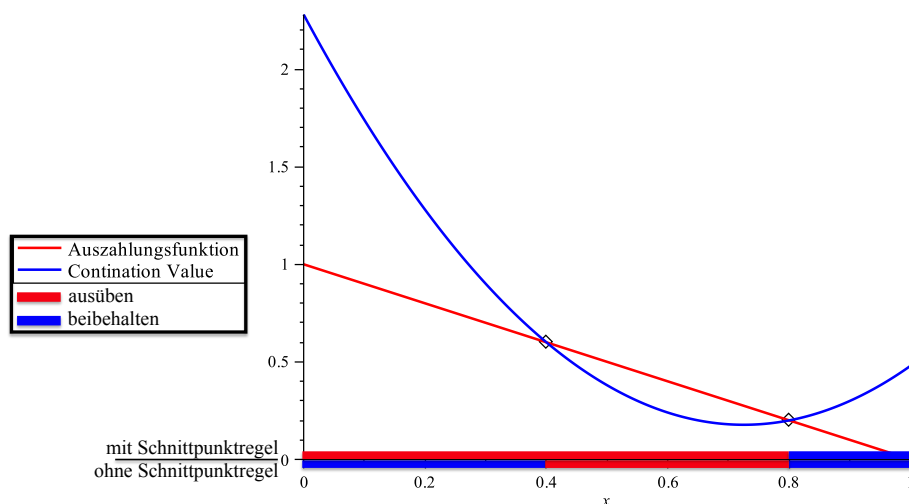


Abbildung 4.12: Continuation Value und Ausübungsfunktion mit den dazugehörigen Bereichen im Falle eines positiven Koeffizienten beim quadratischen Term der Ansatzfunktion

Ähnliches gilt, falls der geschätzte Koeffizient beim quadratischen Term negatives Vorzeichen hat. Dann erhalten wir eine ungenaue Abtrennung der Bereiche nach oben. Liegen beide Schnittpunkte im Intervall $[0, K]$, so wird von der Schnittpunktregel dann der kleinere Schnittpunkt ausgewählt, für alle Preise größer als der ausgewählte Schnittpunkt wird beibehalten, darunter ausgeübt. Ohne Schnittpunktregel würde jedoch ab dem zweiten, größeren Schnittpunkt wieder ausgeübt.

Die Schnittpunktregel stellt also - ähnlich wie bei der Berechnung des kritischen Preises - eine gewisse Konsistenz mit der Intuition her. Irrationale Ausübungs- oder Beibehaltungsentscheidungen können dadurch vermieden werden.

Kapitel 5

Kalibrierung an Marktdaten in einem Varianz-Gamma Aktienpreismodell

5.1 Das Varianz-Gamma Modell¹

Aufgrund seiner hohen Anpassungsgüte an Finanzdaten wurde das Varianz-Gamma Modell in den letzten Jahren sehr populär für logarithmische Aktienreturns. Es existieren verschiedene Darstellungen und Interpretationen des Modells, etwa als Differenz zweier Gamma-verteilter Zufallsvariablen oder als geometrische Brownsche Bewegung mit stochastischer Zeit. Wir halten uns in diesem Kapitel an die im Software Paket R verwendete Darstellung², welche mit der in [CW] präsentierten Herleitung konsistent ist. Die Dichte der Varianz-Gamma Verteilung ist gegeben durch

$$f(x; vgC, \sigma, \theta, \nu) = c(vgC, \sigma, \theta, \nu) \cdot \exp\left(\frac{\theta(x - vgC)}{\sigma^2}\right) \cdot |x - vgC|^{\frac{1}{\nu} - \frac{1}{2}} \cdot K_{\frac{1}{\nu} - \frac{1}{2}}\left(\frac{|x - vgC| \sqrt{\frac{2\sigma^2}{\nu} + \theta^2}}{\sigma^2}\right). \quad (5.1)$$

K_α bezeichnet dabei die Bessel Funktion zweiter Art der Ordnung α und

$$c(vgC, \sigma, \theta, \nu) = \frac{2}{\sigma \sqrt{2\pi} \nu^{1/\nu} \Gamma(1/\nu)} \left(\frac{1}{\sqrt{2\sigma^2/\nu + \theta^2}}\right)^{1/\nu - 1/2}. \quad (5.2)$$

Die Parameter $(vgC, \sigma, \theta, \nu)$ besitzen einfache Interpretationen: vgC ist der Lageparameter, σ ist für die Spreizung verantwortlich, θ für die Asymmetrie und ν

¹Vgl. [CW]

²Vgl. [R]

für die Form. Die in Tabelle 5.1 enthaltenen Grafiken verdeutlichen den Einfluss der Parameter auf die Dichtefunktion.

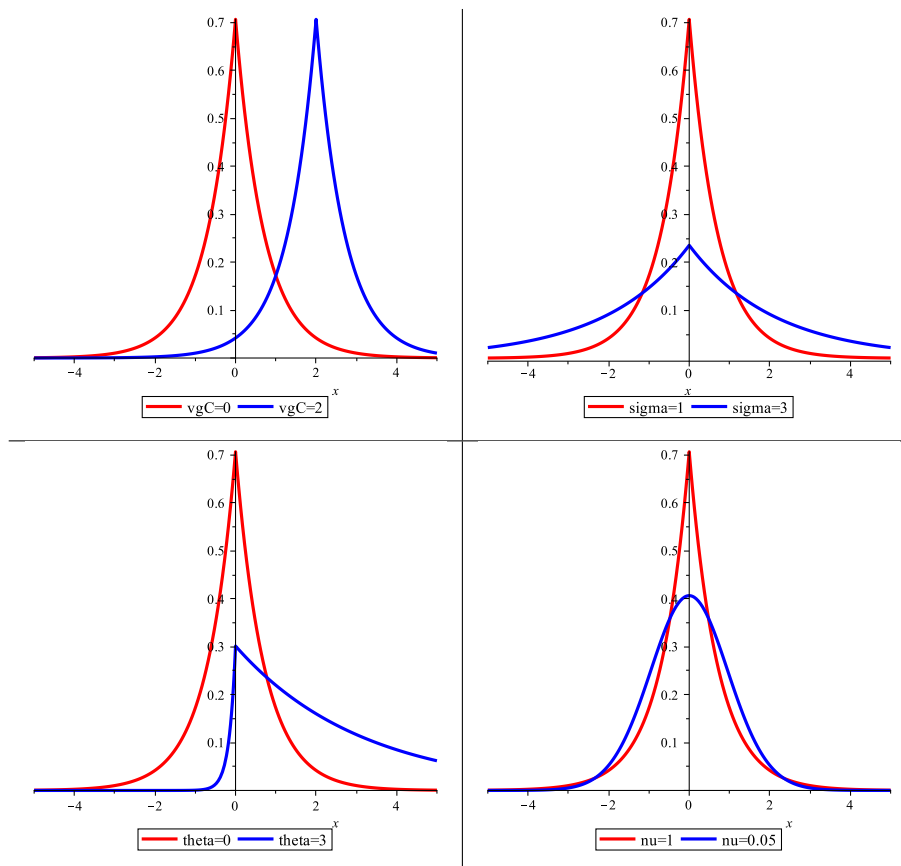


Tabelle 5.1: Dichte der Varianz Gamma Verteilung mit verschiedenen Parametern ausgehend von den Standardwerten $vgC = 0, \sigma = 1, \theta = 0, \nu = 1$

Der Varianz-Gamma Prozess ist ein Lévy Prozess. Dafür wird eine Brownsche Bewegung betrachtet und zu zufälligen, gammaverteilten Zeitpunkten ausgewertet. Daher kommt auch der Name Varianz-Gamma.

Zur Herleitung der Verteilung betrachten wir zuerst ihre charakteristische Funktion

$$\Phi(u) = e^{i\theta tu} \left(\frac{1}{1 + \frac{1}{2}\sigma^2\nu u^2 - i\delta\nu u} \right)^{\frac{t}{\nu}} \quad (5.3)$$

und beginnen nun, diese ausgehend von der klassischen Laplace Verteilung zu konstruieren. Diese besitzt auf \mathbb{R} die Dichte

$$f(x) = \frac{1}{2s} \exp\left(-\frac{1}{s}|x - \theta|\right) \quad (5.4)$$

mit Lageparameter $\theta \in \mathbb{R}$ und Skalenparameter $s > 0$. Da die Varianz $\sigma^2 = 2s^2$

ist, ist die Dichte auch darstellbar als

$$f(x) = \frac{1}{\sigma\sqrt{2}} \exp\left(-\frac{\sqrt{2}}{\sigma}|x - \theta|\right) \quad (5.5)$$

Durch Einführung eines neuen Parameters β , der jeder Seite um θ unterschiedliches Gewicht zuweist, können wir eine schiefe Verteilung erzeugen:

$$g(x) = \frac{\beta}{1 + \beta^2} \frac{\sqrt{2}}{\sigma} \begin{cases} \exp\left(-\frac{\sqrt{2}\beta}{\sigma}(x - \theta)\right) & x \geq \theta \\ \exp\left(-\frac{\sqrt{2}}{\beta\sigma}(-x + \theta)\right) & x < \theta \end{cases} \quad (5.6)$$

Die dazugehörige momentenerzeugende Funktion ist

$$\psi(u) = \exp(i\theta u) \left(\frac{1}{1 - iu\frac{1}{\beta}\frac{\sigma}{\sqrt{2}}} \right) \left(\frac{1}{1 - iu\beta\frac{\sigma}{\sqrt{2}}} \right) = \frac{\exp(i\theta u)}{1 + \frac{1}{2}\sigma^2 u^2 - i\delta u} \quad (5.7)$$

β und δ stehen dabei in der Beziehung

$$\delta = \left(\frac{1}{\beta} - \beta \right) \frac{\sigma}{\sqrt{2}} \quad (5.8)$$

Ausgehend von (5.7) erhalten wir die verallgemeinerte asymmetrische Laplace Verteilung:

$$\Psi(u) = (\psi(u))^t = \left(\frac{\exp(i\theta u)}{1 + \frac{1}{2}\sigma^2 u^2 - i\delta u} \right)^t \quad (5.9)$$

deren Dichte mithilfe der Besselfunktion dritter Ordnung geschrieben werden kann.

Betrachten wir nun eine Gamma Verteilung. Deren Dichtefunktion ist

$$f_{\mathcal{G}(t,\lambda)}(x) = \frac{\lambda^t}{\Gamma(t)} x^{t-1} e^{-\lambda x} \quad (5.10)$$

und ihre momentenerzeugende Funktion ist

$$\Phi_{\mathcal{G}(t,\lambda)}(u) = \left(\frac{1}{1 - \frac{i u}{\lambda}} \right)^t. \quad (5.11)$$

Wir definieren

$$\lambda_1 := \frac{\beta\sqrt{2}}{\sigma} \quad \text{und} \quad \lambda_2 := \frac{\sqrt{2}}{\sigma\beta} \quad (5.12)$$

Der Vergleich von (5.7) und (5.11) zeigt, dass eine Varianz-Gamma Zufallsvariable Y als Differenz zweier Gamma Zufallsvariablen geschrieben werden kann mit

$$Y_t \stackrel{d}{=} \theta t + G_1 - G_2 \quad (5.13)$$

wobei G_1 und G_2 nach $\mathcal{G}(1, \lambda_1)$ bzw. $\mathcal{G}(1, \lambda_2)$ verteilt sind.

Wir lösen nun (5.8) nach β und erhalten

$$\beta = \beta(\sigma, \delta) = -\frac{\delta}{\sigma\sqrt{2}} + \sqrt{1 + \frac{\delta^2}{2\sigma^2}}. \quad (5.14)$$

Nun führen wir noch einen weiteren Parameter ν ein, um die ersten vier Momente der Verteilung ausreichend kontrollieren zu können und die Verteilung optimal an Finanzdaten anpassen zu können. Folgende Transformationen werden verwendet:

$$\sigma \rightarrow \sigma\sqrt{\nu} \quad (5.15)$$

$$\delta \rightarrow \delta\nu \quad (5.16)$$

$$\theta \rightarrow \theta\nu \quad (5.17)$$

$$t \rightarrow \frac{t}{\nu} \quad (5.18)$$

Dadurch erhalten wir $G'_1 \rightarrow \mathcal{G}(t, \lambda'_1)$ und $G'_2 \rightarrow \mathcal{G}(t, \lambda'_2)$ mit

$$\lambda'_1 = \frac{\hat{\beta}\sqrt{2}}{\sigma\sqrt{\nu}} \quad \text{und} \quad \lambda'_2 = \frac{\sqrt{2}}{\hat{\beta}\sigma\sqrt{\nu}} \quad (5.19)$$

und $\hat{\beta} = \beta(\sigma\sqrt{\nu}, \delta\nu)$

Damit erhalten wir die Zufallsvariable

$$X_t \stackrel{d}{=} \theta t + G'_1 - G'_2 \quad (5.20)$$

mit ihrer charakteristischen Funktion

$$\Phi(u) = e^{i\theta t u} \left(\frac{1}{1 + \frac{1}{2}\sigma^2\nu u^2 - i\delta\nu u} \right)^{\frac{t}{\nu}} \quad (5.21)$$

Der Driftterm wird im Folgenden vernachlässigt, indem durchgängig $t = 1$ gesetzt wird. Der Momente der Verteilung betragen dann

$$\mathbb{E}(X) = \delta + \theta \quad (5.22)$$

$$\mathbb{E}((X - \mathbb{E}(X))^2) = \sigma^2 + \delta^2\nu \quad (5.23)$$

$$\mathbb{E}((X - \mathbb{E}(X))^3) = 2\delta^3\nu^2 + 3\delta\sigma^2\nu \quad (5.24)$$

$$\mathbb{E}((X - \mathbb{E}(X))^4) = 3\sigma^4\nu + 12\sigma^2\delta^2\nu^2 + 6\delta^4\nu^3 + 3\sigma^4 + 6\sigma^2\delta^2\nu + 3\delta^4\nu^2 \quad (5.25)$$

Falls $\nu = 0$ erhalten wir eine Kurtosis von $3\sigma^2$, was auf einen Kurtosiskoeffizienten von 3 führt.

Die Parameterschätzung für die Anpassung an Marktdaten wird in R durchgeführt, welches im Paket "VarianceGamma" die Funktion `vgFit()` zur Verfügung stellt. Abbildung 5.1 zeigt die Anpassung einer Varianz-Gamma Verteilung an die Log>Returns der Schlusskurse der Microsoft Aktie von 23.9.2011-20.9.2012. Zum Vergleich ist auch die Dichte einer Normalverteilung enthalten. Die asymmetrische Varianz-Gamma Verteilung kann die Schiefe abbilden und legt mehr Gewicht auf die Ränder.

Mit den geschätzten Parametern können wir nun Aktienkurse simulieren und diese zur Optionsbepreisung verwenden. Für Konformität mit der Parametrisierung der Varianz-Gamma Verteilung in R wurde die R Prozedur `rvg()`,

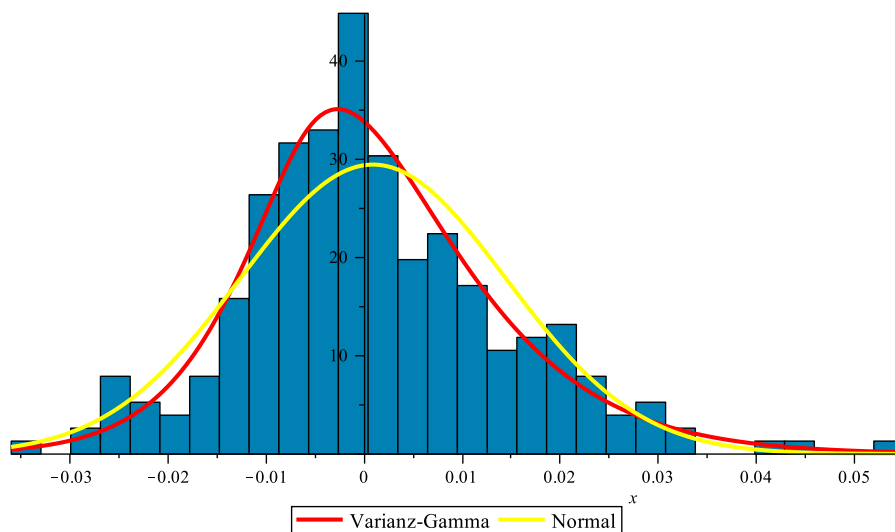


Abbildung 5.1: Log>Returns der Microsoft Aktie von 23.9.2011-20.9.2012 mit Dichte der Normalverteilung und Varianz-Gamma

welche n Realisationen einer Varianz-Gamma Zufallsvariable erstellt, in Maple nachgebildet. Durch die Darstellung einer Varianz-Gamma Zufallsvariable als Differenz von zwei Gamma Zufallsvariablen können wir relativ einfach Zufallszahlen nach der gewünschten Verteilung erzeugen. Die Maple Prozedur `simVG()` gibt n Zufallszahlen nach der gewünschten Varianz-Gamma Verteilung zurück.

Zuerst erfolgen einige Umparametrisierungen der übergebenen Parameter:

```
simVG := proc (n, vgC, sigma, theta, nu)
  local kappa, mu, kkp1sigma, kkp1theta, tau, RG1, RG2, X;
  kappa := (sqrt(2*sigma^2*sqrt(nu)^2+theta^2*nu^2)-theta*nu)/
    (sigma*sqrt(nu)*sqrt(2));
  mu := theta*nu;
  kkp1sigma := sigma*sqrt(nu);
  tau := 1/nu;
  kkp1theta := vgC;
  RG1 := Sample(RandomVariable(Gamma(1, tau)), n);
  RG2 := Sample(RandomVariable(Gamma(1, tau)), n);
```

In `X` wird nun eine Liste von n skalierten und um `kkp1theta` verschobenen Differenzen von Gamma Zufallszahlen gespeichert und ausgegeben:

```
  X := [seq(kkp1theta+kkp1sigma*(RG1[i]/kappa-
    kappa*RG2[i])/sqrt(2), i = 1 .. n)];
  return X;
end proc
```

Mit dieser Prozedur kann nun eine Pfadmatrix erstellt werden.

Die historischen Kursinformationen sind tägliche Schlusskurse. Aus diesen werden zuerst die täglichen Log>Returns $r_t = \log\left(\frac{S_{t+1}}{S_t}\right)$ errechnet. Dadurch

können wir aus n simulierten Log>Returns eine Zeile der Pfadmatrix aufbauen:

$$S_k = S_0 \exp\left(\sum_{i=0}^{k-1} r_i\right) \quad (5.26)$$

Die dafür verwendete Prozedur ist `SimStockLogVGMatrix()`. Aus den in `simVG()` simulierten Log>Returns werden die Aktienkurse errechnet. Die zur Schätzung verwendeten Log>Returns sind tägliche Returns, das heißt, wir erhalten auch wieder tägliche Returns. Wenn wir einen größeren Zeitraster wählen, werden nur die Spalten zu diesen Zeitpunkten aus der Pfadmatrix ausgegeben. Eingabeparameter sind die Parameter der Varianz-Gamma Verteilung, der Startpreis S_0 , die Anzahl der Zeitpunkte n , die Anzahl der Pfade N und der Zeithorizont T .

```
SimStockLogVGMatrix := proc (vgC, sigma, theta, nu, S0, n, N, T)
  local i, S, L, Tv, s, L2, S2, j;
  S := Matrix(N, 250*T+1);
  Tv := <0, seq(j*T/n, j = 1 .. n)>;
  for i from 1 to N do;
```

Zeilenweise wird nun eine Liste mit Log>Returns simuliert und daraus auf den Aktienkurs mithilfe des Startwerts S_0 rückgerechnet.

```
    L := simVG(250*T, vgC, sigma, theta, nu);
    S[i, 1 .. 250*T+1] := <S0,
      seq(S0*exp(add(L[k], k = 1 .. j)), j = 1 .. 250*T)>;
  end do;
```

Falls ein größerer als ein täglicher Zeitraster erforderlich ist, werden die entsprechenden Spalten aus der Pfadmatrix ausgewählt.

```
    if n != 250*T then;
      S2 := Matrix(N, n+1);
      S2[1 .. N, 1] := Vector(N, S0);
      s := floor(250*T/n);
      for j from 1 to n do;
        S2[1 .. N, j+1] := S[1 .. N, j*s+1] ;
      end do;
      S := S2;
    end if;
  return Tv, S;
end proc;
```

Zur Kontrolle der bisherigen Berechnungen verwenden wir Abbildung 5.2 welche die Returns aus diesen simulierten Pfaden zeigt. Die Prozedur liefert also zum ursprünglichen Aktienkurs passende Beobachtungen. Wir können uns nun der Optionsbepreisung zuwenden.

5.2 Vergleich von errechneten Preisen mit Marktpreisen

Um den LSM Algorithmus einem Test zu unterziehen wollen wir nun echte Preise von amerikanischen Optionen, wie sie an einem liquiden Markt notieren, und

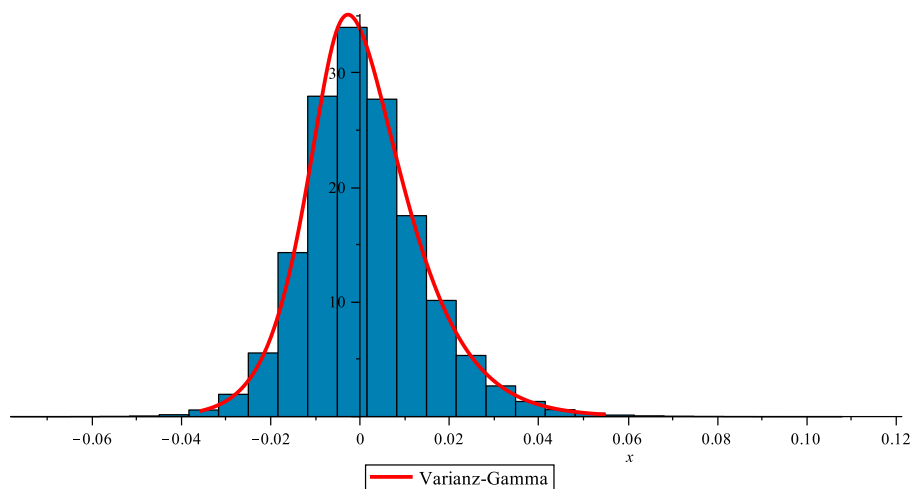


Abbildung 5.2: Simulierte Log>Returns der Microsoft Aktie mit Dichte der Varianz-Gamma Verteilung

errechnete Preise, vergleichen. An der Börse Stuttgart³ wird eine Vielzahl von amerikanischen Optionen gehandelt und die Preise dieser Optionen sind frei verfügbar. Für jede Option sind mehrere Preise verfügbar. Für den freien Handel sind die Geld- und Briefkurse, auch Bid und Ask Kurse genannt, relevant. Der Unterschied zwischen diesen beiden Kursen ist folgender: Der Geldkurs ist jener Kurs, zu dem ein Marktteilnehmer kaufen will, der Briefkurs hingegen ist der Kurs, zu dem ein Marktteilnehmer verkaufen will. Weiters wird ein Kassakurs angegeben. Dieser wird einmal täglich festgesetzt und gilt bis zur nächsten Festsetzung. Wir werden die berechneten Preise mit diesem Kassakurs vergleichen.

Die Kurshistorie, mit der die Varianz-Gamma Parameter berechnet werden, wird von Google Finance⁴ bezogen. Auf täglicher Basis steht die gesamte Kurshistorie von vielen Aktien als Download im csv-Format zur Verfügung. Die Parameter werden mit täglichen Schlusskursen geschätzt. Der Zeitraum, der zur Schätzung herangezogen wird, wird jeweils angegeben. Als risikoloser Zinssatz wird im Folgenden immer der Durchschnittswert des 12-monats-EURIBOR⁵ im September 2012 gewählt, welcher 0.74% beträgt.

Die erste betrachtete Aktie ist die Daimler AG Aktie (ISIN: DE0007100000). Der betrachtete Stichtag ist der 1.10.2012, den die Aktie mit einem Kurs von €37.855 abschloss. Wir betrachten dazu eine amerikanische Put-Option (ISIN: DE000CG0T6J4), die von der Citigroup Bank begeben wurde. Laufzeitende der Option ist der 17.12.2012 und Strike Preis beträgt €50. Damit ergibt sich eine Restlaufzeit von 54 Handelstagen. Die Option liefert ein Bezugsverhältnis von 10:1, das bedeutet für zehn ausgeübte Optionen erhält man das Recht, eine Aktie zum Strike Preis zu verkaufen. In der Praxis werden Optionen jedoch

³<http://www.boerse-stuttgart.de>

⁴<http://www.google.com/finance>

⁵<http://www.euribor-ebf.eu/euribor-org/euribor-rates.html>

meist mit einer Geldzahlung beglichen. Bei einem Bezugsrecht von 10:1 wird demnach nur ein Zehntel des von der Auszahlungsfunktion gegebenen Wertes ausbezahlt. Für die Bewertung mit dem LSM Algorithmus bedeutet das, dass der errechnete Preis durch 10 dividiert werden muss, um den Preis mit den Börsenpreisen vergleichen zu können.

Zur Berechnung werden 5000 Pfade nach dem Varianz-Gamma Modell mit den Parametern $\text{vgC}=-0.003401$, $\text{sigma}=0.022830$, $\text{delta}=0.004021$, $\text{nu}=0.118405$ basierend auf den Schlusskursen der Aktie von 3.10.2011 bis 1.10.2012, simuliert.

Die Option notiert zu einem Kassakurs von €1.24. Der LSM Algorithmus mit den oben angeführten Parametern liefert einen Preis von €12.107 bzw. korrigiert um das Bezugsverhältnis €1.2107. Das heißt der Kassakurs enthält eine Marge von 2.36%. Der Einfluss des risikolosen Zinssatzes ist in diesem Beispiel mit kurzer Laufzeit relativ gering. Wird die gleiche Option mit einem risikolosen Zinssatz von 2% bewertet, ergibt das einen Unterschied in der vierten Nachkommastelle.

Als nächstes betrachten wir wieder eine Option auf die Daimler AG Aktie, aber mit kürzerer Laufzeit (ISIN: CH0141641602). Die Option wurde von der UBS Bank begeben. Ihr Laufzeitende ist der 15.10.12, was einer Restlaufzeit von 10 Handelstagen entspricht. Das Bezugsverhältnis ist wieder 10:1 und der Strike Preis beträgt €45. Die Parameterschätzer für das Varianz-Gamma Modell sind die gleichen wie oben.

Die Option notiert zu einem Kassakurs von €0.77. Wir erhalten als Ergebnis des LSM Algorithmus einen Preis von €7.104, bzw. dividiert durch das Bezugsverhältnis €0.7104. Die Marge ist demnach 7.74%.

Wir wenden uns nun Optionen auf die Aktie der Deutschen Bank AG zu. Bewertungstag ist der 3.10.2012. Die Kurshistorie bilden die Schlusskurse der Aktie vom 5.10.2011 bis zum 2.10.2012. Damit erhalten wir die Varianz-Gamma Parameter $\text{vgC}=-0.0014009$, $\text{sigma}=0.0301494$, $\text{theta}=0.0007294$, $\text{nu}=0.6795836$. Der Schlusskurs der Aktie am 2.10.2012 ist 31.48 und wird als Startpreis S_0 verwendet.

Die erste zu bewertende Option (ISIN: DE000GT3R5L5) läuft am 25.10.2012 ab, es verbleiben somit 17 Handelstage. Die Bank Goldman Sachs hat diese Option begeben. Der Strike Preis beträgt €31 und die Option notiert zu einem Kassakurs von €0.083. Der LSM Algorithmus liefert einen Preis von €0.139, was einer Abweichung von -67,4% entspricht.

Die nächste Option (ISIN: DE000GT3BSH9) hat Strike €33 und Laufzeitende 15.11.2012, also 31 verbleibende Handelstage. Emittent ist auch hier Goldman Sachs. Sie notiert zu €0.22, LSM liefert €0.307 und somit eine Abweichung von -39,6%.

Die dritte bewertete Option (ISIN: DE000CG1J3D5) wurde von der Citigroup Bank begeben. Ihr Strike Preis beträgt €44 bei einem Laufzeitende am 17.12.12. Es verbleiben damit 53 Handelstage. Die Option notiert zu einem Kassakurs von €1.21 und der LSM Algorithmus liefert €1.31 als Preis.

Je länger die Restlaufzeit ist, desto besser kann der LSM Algorithmus für diese Aktie die Option bewerten, wenn der Kassakurs als Referenzpreis hergenommen wird. Für die teilweise doch großen Abweichungen kommen mehrere Ursachen in Frage. Eine Überlegung wäre, dass für kurze Laufzeiten aktuelle Informationen, die nicht in die Parameterschätzer für das Aktienpreismodell eingeflossen sind, relevant und den Marktteilnehmern bekannt sind. Langfristig

nähert sich die Aktie dann wieder ihrem historischen Verhalten an und somit liegen auch die berechneten Preise wieder näher an den Kassakursen.

Die Strikepreise der ersten beiden Optionen auf die Aktie der deutschen Bank liegen mit €31 bzw. €33 sehr nahe am Startkurs S0 (€31.48). Die dritte Option liegt mit einem Strikepreis von €44 tief im Geld. Das Verhältnis von Startkurs zu Strike Preis beeinflusst somit auch maßgeblich die Qualität der Berechnung.

Betrachten wir nun auf der VoestAlpine AG Aktie (ISIN: AT0000937503) basierende Optionen. Bewertungstag ist wieder der 3.10.12. Zur Parameterschätzung stehen die Schlusskurse der Aktie vom 3.10.2011 bis zum 2.10.2012 zur Verfügung und wir erhalten für das Varianz-Gamma Modell die Parameter $\text{vgC}=0.006029$, $\text{sigma}=0.032187$, $\text{theta}=-0.005427$, $\text{nu}=0.465100$. Die Aktie schloss am 2.10.2012 zu €24.4.

Um einen direkten Einfluss des Strike Preises auf die Qualität der LSM Bewertung zu sehen, betrachten wir zwei Optionen, die beide am 19.12.2012 ablaufen (das entspricht 55 Handelstagen). Eine Option kann zum Strike Preis von €24 (ISIN: DE000CK3CBW5) ausgeübt werden, der Strike Preis der anderen Option (ISIN: DE000CK0XSR1) beträgt €30. Bei beiden Optionen tritt die Commerzbank als Emittent auf. Die Option mit Strike €24 notiert zu einem Kassakurs von €0.13, die mit Strike €30 zu €0.56.

Wir verwenden wieder 5000 Pfade für die Monte Carlo Simulation und erhalten als Ergebnis Preise von €0.167 beziehungsweise €0.579. Die Abweichungen betragen somit -28.6% beziehungsweise -3.3%. Die Option, die tiefer in the money liegt, kann also viel besser über den LSM Algorithmus bewertet werden.

Eine dritte Option (ISIN: DE000CK7KNU8), ebenfalls von der Commerzbank begeben, für die VoestAlpine Aktie wird noch bewertet: Der Strike Preis liegt mit €25 nahe am Strike Preis der ersten VoestAlpine Option. Ihr Laufzeitende ist der 13.3.2013, was 110 Handelstagen entspricht. Der Kassakurs der Option beträgt €0.24 und der LSM Algorithmus liefert einen Preis von €0.266, was einer Abweichung von -11% entspricht. Es zeigt sich also wieder, dass Optionen mit längerer Laufzeit mit dem LSM Algorithmus besser bewertet werden können, natürlich nur unter der Prämisse, dass die Kassakurse als Referenzwert gültig sind.

Tabelle 5.2 gibt einen Überblick über die oben beschriebenen Optionen. Die Spalte T beinhaltet das Ablaufdatum der Option, die Spalte d die Handelstage bis zum Ablaufdatum. In der Spalte LSM werden die berechneten Preise - das Bezugsverhältnis wurde hier schon berücksichtigt - angegeben. Die Abweichung wird berechnet als

$$\text{Abw.} = \frac{\text{Kassa} - \text{LSM}}{\text{Kassa}} \quad (5.27)$$

In der Praxis ist der nächste Schritt nach der Bewertung die Überlegung, wie die Option optimal ausgeübt werden kann. Ein gutes Hilfsmittel hierfür ist der kritische Preis, wie in Kapitel 4 beschrieben. Zur Optionsbewertung wurde die Prozedur `ValueAmerican()` verwendet, welche auch die zu jedem Zeitpunkt verwendeten Regressionsfunktionen zurückgibt. Damit kann über die Prozedur `EEBoundary()` der kritische Preis berechnet werden. Wir betrachten hierfür exemplarisch die oben beschriebene VoestAlpine AG Option mit Strike Preis €30 und 55 Restlaufzeiten (ISIN: DE000CK0XSR1).

ISIN	K	T	d	S0	Kassa	LSM	Abw.
Basistitel: Daimler AG; Bewertungstag: 2.10.2012							
CH0141641602	45	15.10.12	10	37,855	0,77	0,710	7,74%
DE000CG0T6J4	50	17.12.12	54	37,855	1,24	1,211	2,36%
Basistitel: Deutsche Bank AG; Bewertungstag: 3.10.2012							
DE000GT3R5L5	31	25.10.12	17	31,48	0,083	0,139	-67,35%
DE000GT3BSH9	33	15.11.12	31	31,48	0,22	0,307	-39,59%
DE000CG1J3D5	44	17.12.12	53	31,48	1,21	1,310	-8,27%
Basistitel: VoestAlpine AG; Bewertungstag: 3.10.2012							
DE000CK3CBW5	24	19.12.12	55	24,4	0,13	0,167	-28,62%
DE000CK0XSR1	30	19.12.12	55	24,4	0,56	0,579	-3,34%
DE000CK7KNU	25	13.03.13	110	24,4	0,24	0,266	-10,95%

Tabelle 5.2: Übersicht über die bewerteten amerikanischen Put-Optionen

Abbildung 5.3 zeigt einige simulierte Pfade, den Strike Preis sowie die Early Exercise Boundary. Wir beobachten, dass diese weitgehend horizontal verläuft. Ihr durchschnittlicher Wert zwischen $t = 0.032$ (Tag 8) und $t = 0.204$ (Tag 51) beträgt €23.94. Gegen Ende der Laufzeit nähert sie sich wie erwartet dem Strike Preis an.

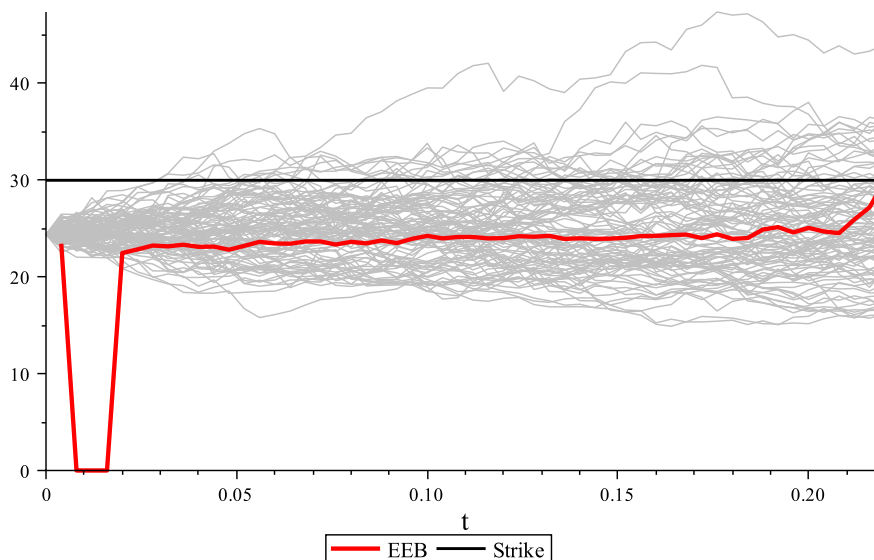


Abbildung 5.3: Early Exercise Boundary und simulierte Pfade für eine Option auf die VoestAlpine Aktie mit Strike Preis 30 und 55 Tagen Restlaufzeit

Literaturverzeichnis

- [LS] FRANCIS A. LONGSTAFF, EDUARDO SCHWARTZ (2001). *Valuing American Options by Simulation: A Simple Least-Squares Approach*. Rev. Financ. Stud. 14 (1), pp. 113-147.
- [K] IOANNIS KARATZAS (1988). *On the pricing of American Options*. Appl. Math. Optim. 17, pp. 37-60.
- [W] HARALD WORACEK, MICHAEL KALTENBÄCK, MARTIN BLÜMLINGER (2011). *Funktionalanalysis*.
- [S] LARS STENTOFT (2004). *Least Squares Monte Carlo Approach to American Option Valuation*. Manag. Sci. 50, pp. 1193-1203.
- [N] WHITNEY K. NEWEY (1997). *Convergence rates and asymptotic normality for series estimators*. J. Econometrics 79, pp. 147-168.
- [C] EMMANUELLE CLÉMENT, DAMIEN LAMBERTON, PHILIP PROTTER (2002). *An analysis of least squares regression method for American option pricing*. Finance Stoch. 6, no. 4, pp. 449-471.
- [P] HUYÊN PHAM (1997). *Optimal Stopping, Free Boundary, and American Option in a Jump-Diffusion Model*. Appl. Math. Optim. 35, no. 2, pp. 145-164.
- [B] GUY BARLES (1995). *Critical stock price near expiration*. Mathematical Finance 5, no. 2, pp. 77-95.
- [KK] RACHEL A. KUSKE, JOSEPH B. KELLER (1998). *Optimal exercise boundary for an American put option*. Applied Mathematical Finance 5, no. 2, pp. 107-116
- [R] DAVID SCOTT, CHRISTINE YANG DONG (2012). *Package 'Variance-Gamma'*
- [CW] OLIVIER LE COURTOIS , CHRISTIAN WALTER (2009). *A Study on Value-at-Risk and Lévy Processes*, Working paper.
- [EQ] *Encyclopedia of Quantitative Finance, Eintrag Jump Diffusion Models*
- [TV] PETER TANKOV, EKATERINA VOLTCHKOVA (2009). *Jump-diffusion models: a practitioner's guide*. Banque et Marchés, 99.
- [M] ROBERT C. MERTON (1976). *Option pricing when underlying stock returns are discontinuous*. Journal of Financial Economics 3, no. 1-2, pp. 125-144

Abbildungsverzeichnis

3.1	Pfade einer simulierten Aktie	18
3.2	Regressionsfunktionen und Auszahlungsfunktion für ein Polynom 2. Grades als Ansatzfunktion	24
3.3	Regressionsfunktionen und Auszahlungsfunktion zu $t = 1$	25
3.4	Regressionsfunktionen und Auszahlungsfunktion für ein Polynom 1. Grades als Ansatzfunktion	27
3.5	Zwei Approximationen an den Continuation Value	28
3.6	Pfade einer Jump Diffusion im Vergleich zu klassischen Black- Scholes Pfaden	34
4.1	Continuation Region und Exercise Region einer amerikanischen Put Option	38
4.2	Kritischer Preis für $T = 1, K = 1, r = 0.1, \sigma = 0.238$	40
4.3	Kritischer Preis für $T = 1, K = 1, r = 0.1, \sigma = 0.4$	41
4.4	Kritischer Preis für $T = 1, K = 1, r = 0.1$, verschiedene σ	42
4.5	Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 =$ $100, N = 2000, L = 50$; Ansatzfunktion: Polynom 1. Grades	48
4.6	Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 =$ $100, N = 2000, L = 50$; Ansatzfunktion: Polynom 2. Grades	49
4.7	Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 =$ $100, N = 2000, L = 50$; Ansatzfunktion: Polynom 3. Grades	50
4.8	Kritischer Preis für $T = 1, K = 100, r = 0.06, \sigma = 0.5, S_0 =$ $100, N = 2000, L = 50$; Ansatzfunktion: Polynom 4. Grades	50
4.9	Kritischer Preis für $T = 1, K = 40, r = 0.06, \sigma = 0.4, S_0 =$ $36, N = 2000, L = 50$; verschiedene Ansatzfunktionen	51
4.10	Kritischer Preis für $T = 1, K = 40, r = 0.06, S_0 = 40, N =$ $2000, L = 50$, Ansatzfunktion: Polynom 2. Grades, verschiedene Werte für σ	51
4.11	Continuation Value und Ausübungsfunktion; Schnittpunkte {23.65, 27.18}	52
4.12	Continuation Value und Ausübungsfunktion mit den dazu- gehörigen Bereichen im Falle eines positiven Koeffizienten beim quadratischen Term der Ansatzfunktion	55
5.1	Log>Returns der Microsoft Aktie von 23.9.2011-20.9.2012 mit Dichte der Normalverteilung und Varianz-Gamma	60
5.2	Simulierte Log>Returns der Microsoft Aktie mit Dichte der Varianz-Gamma Verteilung	62

5.3 Early Exercise Boundary und simulierte Pfade für eine Option auf die VoestAlpine Aktie mit Strike Preis 30 und 55 Tagen Restlaufzeit 65

Tabellenverzeichnis

2.1	zugrunde liegende Pfade	4
2.2	Auszahlungsmatrix zum Zeitpunkt $t = 3$	4
2.3	Regression zum Zeitpunkt $t = 2$	5
2.4	Optimales Ausüben zu $t = 2$	5
2.5	Auszahlungsmatrix zum Zeitpunkt $t = 2$	5
2.6	Regression zum Zeitpunkt $t = 1$	6
2.7	Optimales Ausüben zu $t = 1$	6
2.8	Stoppzeiten	7
2.9	Cashflows der Option	7
3.1	Verwendete Ansatzfunktionen für numerische Tests des LSM Algorithmus	28
3.2	Ergebnisse des LSM Algorithmus mit verschiedenen Ansatzfunktionen; Strike $K = 40$, risikoloser Zins $r = 0.06$ und 1000 Pfade	29
3.3	Relative Abweichungen zu den Ergebnissen aus [LS] in % mit verschiedenen Ansatzfunktionen; Strike $K = 40$, risikoloser Zins $r = 0.06$ und 1000 Pfade	30
3.4	Korrelationskoeffizienten der spaltenweisen Abweichung zu den Parametern	30
3.5	Early Exercise Premium der amerikanischen Option	31
3.6	Korrelationskoeffizienten der spaltenweisen Abweichung zu den Parametern	32
3.7	Korrelationen der Preise zu den Eingabeparametern	35
3.8	Preise für Optionen mit einer Jump Diffusion als Aktienpreismodell	35
4.1	Approximierter Continuation Value für verschiedene Ansatzfunktionen. Parameter: $K = 100, S_t = 100, T = 1, \sigma = 0.5, r = 0.06, L = 50, t = 0.4$	44
4.2	Vergleich der Preise mit und ohne Schnittpunktregel	54
5.1	Dichte der Varianz Gamma Verteilung mit verschiedenen Parametern ausgehend von den Standardwerten $\text{vgC} = 0, \sigma = 1, \theta = 0, \nu = 1$	57
5.2	Übersicht über die bewerteten amerikanischen Put-Optionen	65