

Universität Bayreuth
Fakultät für Mathematik und Physik

Diplomarbeit

Portfolio-Optimierung mit dynamischer Programmierung

CAROLINE ÖHRLEIN

30. September 2006

Betreut durch
PROF. DR. LARS GRÜNE

Inhaltsverzeichnis

1	Einleitung	1
2	Das finanzmathematische Portfolio-Problem	5
2.1	Einführung in Portfolio-Entscheidungen	5
2.2	Analyse der Gestalt der Power-Nutzenfunktion	8
3	Theoretische Grundlagen	13
3.1	Ein deterministisches optimales Steuerungsproblem	14
3.2	Eigenschaften der optimalen Wertefunktion	19
3.2.1	Das Bellman'sche Optimalitätsprinzip	19
3.2.2	Die Hamilton-Jacobi-Bellman Gleichung	22
3.3	Diskretisierung des optimalen Kontrollproblems	25
3.3.1	Diskretisierung in der Zeit	25
3.3.2	Ein Iterationsverfahren zur Berechnung der Wertefunktion . . .	27
3.3.3	Diskretisierung im Raum	27
3.4	Berechnung approximativ optimaler Trajektorien	32
3.4.1	Zeitdiskrete optimale Trajektorien	32
3.4.2	Numerische Berechnung approximativ optimaler Kontrollen . .	33
3.4.3	Das zeitkontinuierliche Problem	34
3.5	Alternative Iterationsverfahren	34
3.5.1	Das kontrollierte Gauß-Seidel-Verfahren	34
3.5.2	Die Strategie-Iteration	36
4	Numerische Lösung der Entscheidungsmodelle	39
4.1	Die elementaren C++-Programmteile	40
4.1.1	Routinen zur Verwaltung gleichmäßiger Gitter	40
4.1.2	Routinen zur numerischen Lösung des optimalen Steuerungsproblems	46
5	Diskussion der praktischen Ergebnisse	57
5.1	Analyse optimaler Steuerungsprobleme	58
5.1.1	Die Dynamik des optimal gesteuerten Systems	58
5.1.2	Vorgehen bei der Auswertung	61
5.2	Modell 1: Ein Vermögenswert - Konstante risikolose Rendite	63

5.2.1	Einführung	63
5.2.2	Lösung des Problems mithilfe der HJB-Gleichung	65
5.2.3	Diskussion der numerischen Ergebnisse	68
5.2.4	Variationen in den Parametern	72
5.3	Modell 2: Zwei Vermögenswerte - Zeitlich variierender Zinssatz und Eigenkapitalrendite	87
5.3.1	Einführung	87
5.3.2	Diskussion der praktischen Ergebnisse	89
5.3.3	Variationen in den Parametern	96
5.4	Modell 3: Erweiterung des zweiten Modells auf den zeitlich - periodischen Fall	109
5.4.1	Einführung	109
5.4.2	Diskussion der numerischen Ergebnisse	111
5.4.3	Variationen in den Parametern	115
6	Fazit der praktischen Ergebnisse und Ausblick	129
A	Notation	131
B	Finanzmathematische Grundbegriffe	133
C	C++-Quelltext	135
D	MATLAB-Quelltexte	163
D.1	Funktion: zeichnen_wertef(name)	163
D.2	Funktion: zeichnen_uopt(name)	163
D.3	Funktion: zeichnen_feld(name)	164
D.4	Funktion: zeichnen_traj(name)	166
E	Material auf der beiliegenden CD	167
	Literaturverzeichnis	169

Abbildungsverzeichnis

2.1	Power-Nutzenfunktion für variierendes γ	10
4.1	Beispiel-Rechteckgitter	40
4.2	Lokale Nummerierung der Ecken	41
5.1	Optimale Wertefunktion, Ausgangsgrafik für Modell 1	70
5.2	Optimale Kontrollfunktion, Ausgangsgrafik für Modell 1	70
5.3	Optimale Trajektorien für unterschiedliche Startwerte, Ausgangsgrafik für Modell 1	71
5.4	Optimale Wertefunktionen für variierende Risikoaversion, Modell 1	74
5.5	Optimale Kontrolle für variierende Risikoaversion, Modell 1	74
5.6	Optimale Vektordarstellung für variierende Risikoaversion, Modell 1	75
5.7	Optimale Trajektorien für variierende Risikoaversion, Modell 1	77
5.8	Optimale Wertefunktionen für variierende Rendite, Modell 1	79
5.9	Optimale und skalierte Kontrollfunktion für variierende Rendite, Modell 1	80
5.10	Optimale Vektordarstellung für variierende Rendite, Modell 1	81
5.11	Optimale Trajektorie x für unterschiedliche Rendite	81
5.12	Optimale Wertefunktion für variierende Diskontrate, Modell 1	83
5.13	Optimale Kontrollfunktion und Konsumneigung für variierende Diskontrate, Modell 1	84
5.14	Optimale Vektordarstellung für variierende Diskontrate, Modell 1	85
5.15	Optimale Trajektorien für variierende Diskontrate, Modell 1	86
5.16	Verlauf der Renditen für Modell 2	90
5.17	Optimale Wertefunktion, Ausgangsgrafik für Modell 2	92
5.18	Optimale (un)skalierte Kontrollfunktion, Ausgangsgrafik für Modell 2	92
5.19	Optimale Gleichgewichte, Ausgangswerte für Modell 2	93
5.20	Optimale Trajektorien, Ausgangswerte für Modell 2	95
5.21	Optimaler Konsum, Ausgangswerte für Modell 2	95
5.22	Optimale Wertefunktionen für Modell 2 bei Variation von γ	96
5.23	Optimale Konsumneigung für variierende Risikoaversion, Modell 2	98
5.24	Optimale Gleichgewichte für variierende Risikoaversion, Modell 2	99
5.25	Optimale Wertefunktionen für variierenden Renditen, Modell 2	101
5.26	Optimale Kontrollfunktionen und Konsumneigungen für variierende Renditen, Modell 2	102

5.27	Optimale Gleichgewichte für variierende Renditen, Modell 2	103
5.28	Optimale Wertefunktionen für variierende Diskonraten, Modell 2 . . .	105
5.29	Optimale Kontrollfunktionen für variierende Diskontrate, Modell 2 . . .	106
5.30	Optimale Konsumneigung für variierende Diskontrate, Modell 2	107
5.31	Optimale Gleichgewichte für variierende Diskontrate, Modell 2	108
5.32	Optimale Wertefunktion, Ausgangsgrafik für Modell 3	112
5.33	Optimale Kontrollfunktion und Konsumneigung, Ausgangsgrafik für Modell 3	113
5.34	Optimales Vektorfeld, Ausgangsgrafik für Modell 3	114
5.35	Optimale Trajektorien, Ausgangsgrafik für Modell 3	115
5.36	Optimale Wertefunktionen für variierende Risikoaversion, Modell 3 . .	116
5.37	Optimale Konsumneigung für variierende Risikoaversion, Modell 3 . .	118
5.38	Optimales Vektorfeld und Trajektorien für variierende Risikoaversion, Modell 3	119
5.39	Optimale Wertefunktionen für variierende Renditen, Modell 3	121
5.40	Optimales Vektorfeld und optimale Trajektorien für variierende Renditen, Modell 3	123
5.41	Optimale Wertefunktionen für variierende Diskontrate, Modell 3	126
5.42	Optimale Kontrollfunktionen und Konsumneigungen für variierende Diskontrate, Modell 3	127
5.43	Optimales Vektorfeld und optimale Trajektorien für variierende Diskontrate, Modell 3	128

Tabellenverzeichnis

2.1	Krümmungsverhalten der Nutzenfunktion und Bedeutung	8
4.1	Auflistung der elementaren Variablen bzw. Parameter mit ihrer Bedeutung für das Programm	41
4.2	Auflistung der elementaren Variablen und Parameter sowie ihre Bedeutung für das Programm	46
5.1	Die zur Auswertung des ersten Modells verwendeten Parameterwerte .	69
5.2	Die zur Analyse von Modell 1 verwendeten Risikoaversionen	73
5.3	Die zur Analyse von Modell 1 verwendeten Renditen	78
5.4	Die zur Analyse von Modell 1 verwendeten Diskontraten	82
5.5	Die verwendeten Variablen und deren Bedeutung	88
5.6	Die zur Auswertung des zweiten Modells verwendeten Parameterwerte .	89
5.7	Die zur Analyse von Modell 2 verwendeten Renditen	100
5.8	Die zur Analyse von Modell 2 verwendeten Diskontraten	104
5.9	Die zur Auswertung des dritten Modells verwendeten Parameterwerte .	111

1 Einleitung

Eines der bekanntesten Probleme in der Finanzmathematik ist das Finden einer optimalen Investmentstrategie für einen Investor, dem verschiedene risikolose und risikobehaftete Anlagemöglichkeiten zur Verfügung stehen. Den Gesamtbestand unterschiedlicher Anlageinstrumente (normalerweise Aktien, Rentenwerte und Barmittel), den ein Unternehmen oder ein Privatanleger besitzt, bezeichnet man als *Portfolio*. Der Besitz eines Portfolios ist in der Regel Teil einer Strategie, die Risiken finanzieller Investitionen durch Streuung zu senken. Das Hauptresultat der Portfoliotheorie ist somit die *Risiko-diversifikation*, wobei für jeden Investor ein sogenanntes optimales Portfolio aus allen Anlagemöglichkeiten existiert, das dessen Risiko-Chancen-Profil bestmöglich abbildet.

Mathematischer Ursprung der Theorie der Portfolio-Optimierung war 1952 die Arbeit von MARKOWITZ [12] über den Erwartungswert-Varianz-Ansatz zur Beurteilung von Investmentstrategien an Wertpapiermärkten. In dieser wurde zum ersten Mal ein Zusammenhang zwischen Rendite und Risiko eines Vermögenswertes entdeckt und berücksichtigt. Aufgrund seiner Einfachheit und Plausibilität wurde dieser Ansatz in Theorie und Praxis schnell sehr populär und wird auch heute noch häufig angewendet. Das ihm zugrunde liegende Modell ist ein sogenanntes Ein-Perioden-Modell, d.h. es werden zu Beginn einer Periode Entscheidungen über Investmentstrategien getroffen, die hieraus folgenden Konsequenzen aber erst am Ende der Periode beobachtet. Dazwischen findet also kein Eingriff in den Markt statt. Derartige Modelle nennt man auch *statische* Modelle, weil nach dem Festlegen der Strategie das Anfangsportfolio bis zum Schluß beibehalten und nicht mehr gehandelt wird. In der Einfachheit des Erwartungswert-Varianz-Ansatzes liegen allerdings erhebliche Nachteile, was später zwangsläufig zur Betrachtung zeitsteiger bzw. *dynamischer* Modelle führte (siehe z.B. MERTON [13]).

Anders als im Ein-Perioden-Modell ist es bei diesen Modellen nun möglich, zu jedem beliebigen Zeitpunkt im betrachteten, endlichen Zeitraum $[0, T]$ zu handeln.

Zur Darstellung ebenso wie zur Lösung eines Portfolio-Problems werden heute eine Vielzahl verschiedener mathematischer Modelle und Lösungsmethoden benutzt. Je nachdem welche Umstände des realen Marktes in die Modellierung mit einbezogen werden sollen (und auch je nachdem welche Modelle vom mathematischen Standpunkt aus lösbar sind), werden Investitionsentscheidungen und Aktienkurse dabei entweder nur zu *diskreten* oder *kontinuierlichen* Zeitpunkten betrachtet.

Ziel dieser Arbeit ist es, ausgehend von einer Interpretation des Portfolio-Problems als deterministisches Steuerungsproblem, sowohl die mathematischen Voraussetzungen und Grundlagen des Problems aufzuzeigen als auch die zur Lösung benötigten Methoden herzuleiten und zu beschreiben. Des Weiteren sollen unterschiedliche dynamische Beispiel-Modelle in kontinuierlicher Zeit unter Anwendung eines Algorithmus aus der dynamischen Programmierung ausgewertet und die Resultate verglichen werden. Die verwendeten Modelle basieren dabei auf einem Skriptum von SEMMLER [15].

Die Arbeit ist wie folgt aufgebaut:

Im nächsten Kapitel werden die für den weiteren Verlauf wesentlichen finanzmathematischen Grundlagen der Portfoliotheorie zusammengefaßt. Dabei sollen zum einen insbesondere das Entscheidungsproblem, dem sich ein Investor gegenüber sieht, zum anderen aber auch das Ziel seiner Entscheidungen genauer betrachtet werden. In der Realität bedeutende Einflüsse wie Wertpapierpreise, Transaktionskosten u.v.m., würden an dieser Stelle zu weit führen und werden daher in dieser Arbeit vernachlässigt.

Im dritten Kapitel werden zunächst die wesentlichen mathematischen Grundlagen optimaler Steuerungsprobleme hergeleitet. Der zweite Hauptpunkt dieses Kapitels widmet sich der Numerik des optimalen Steuerungsproblems, wobei implementierbare Methoden zu dessen Lösung hergeleitet werden. Hierbei werden insbesondere effiziente Strategien zur iterativen Berechnung der optimalen Wertefunktion als auch Verfahren zur

Berechnung approximativ optimaler Trajektorien betrachtet.

Mit dem vierten Kapitel beginnt der praktische Teil der Arbeit. Hier werden die elementaren Routinen eines Algorithmus zur Lösung optimaler Steuerungsprobleme überblickartig vorgestellt, welche im Rahmen der Vorlesung *Numerische Dynamik von Kontrollsystemen* von GRÜNE [8] entstanden sind. Dabei werden Aufbau sowie die benötigten Ein- und Ausgaben der Routinen beschrieben.

Im fünften Kapitel steht die praktische Auswertung von drei Portfolio-Modellen, welche sich vor allem in der Modellierung der Rendite eines Vermögenswertes unterscheiden, im Mittelpunkt. Nach einer kurzen Beschreibung des Vorgehens bei der Analyse der berechneten Ergebnisse werden die numerischen Resultate durch verschiedene Grafiken veranschaulicht und diskutiert. Weiterhin werden Abänderungen in den Parametern vorgenommen und Auswirkungen dieser Variationen auf die optimalen Lösungen bzw. auf das Verhalten eines Investors aufgezeigt. Um die Resultate besser vergleichen zu können, werden beim ersten Modell zudem die exakten Lösungen des Problems mit Hilfe der Hamilton-Jacobi-Bellman Gleichung hergeleitet.

Zum Abschluss der Arbeit werden die praktischen Ergebnisse noch einmal zusammengefasst und ein Ausblick auf Erweiterungs- bzw. Verbesserungsmöglichkeiten gegeben.

An dieser Stelle möchte ich mich bei Herrn Prof. Dr. Grüne für die interessante Aufgabenstellung, die motivierenden Anregungen sowie die umfangreiche Betreuung im Laufe der Arbeit bedanken.

2 Das finanzmathematische Portfolio-Problem

Zu Beginn sollen die für diese Arbeit elementaren, finanzmathematischen Grundlagen der Portfolio-Theorie in einem kurzen Überblick zusammengestellt werden.

Dabei werden wir uns im ersten Abschnitt insbesondere mit der Frage nach den Entscheidungen, welche ein Investor zu treffen hat und mit der Frage nach dem verfolgten Ziel eines Anlegers beschäftigen. Auf diese Weise sollen die im Folgenden betrachteten Portfolio-Modelle in eine mathematische Theorie eingeordnet werden.

Der zweite Abschnitt behandelt eine kurze Analyse der Gestalt der Nutzenfunktion. Der Schwerpunkt liegt dabei in der Definition der Risikoaversion eines Investors.

Wie in der Einführung dargestellt, werden dabei ausschließlich *dynamische* Portfolio-Entscheidungsmodelle betrachtet.

2.1 Einführung in Portfolio-Entscheidungen

Wie bereits erwähnt, liegt der Hauptgegenstand der Portfolio-Optimierung in der Bestimmung einer optimalen Investment- und Konsumstrategie an einem Finanzmarkt. Zur Festlegung dieser Strategie müssen wir überlegen, welche Entscheidungen ein Investor hinsichtlich seines Portfolios treffen kann. Zudem müssen wir festlegen, welches Ziel ein Anleger mit seinen Entscheidungen verfolgt.

Formulieren wir dazu in Anlehnung an KORN UND KORN [11] das allgemeine Portfolio-Problem zunächst genauer:

„Das Portfolioproblem, dem ein Investor in einem Finanzmarkt gegenüber steht, besteht darin, zu einem gegebenen Startkapital $x > 0$ eine optimale Investment- und Konsumstrategie festzulegen. Das heißt, er muss festlegen [...], *wie viele* Anteile er von *welchem* Wertpapier *wann* halten muss und *wie viel* Vermögen er *wann* konsumieren darf, um seinen Nutzen aus Konsum im Zeitraum $[0, T]$ und/oder Endvermögen im Zeithorizont $t = T$ zu maximieren“.

Wir beschäftigen uns somit im Folgenden mit einem Optimierungsproblem, welches die Dynamik des Finanzmarktes bestimmt, wobei es sich aus einem Auswahlproblem („welche“ Wertpapiere), einem Mengenproblem („wie viele“ Anteile, „wie viel“ Vermögen) und einer zeitlich dynamischen Komponente („wann“) zusammensetzt.

Derartige dynamische Systeme können nun zum einen sowohl in diskreter als auch in kontinuierlicher Zeit zum anderen unter Verwendung eines deterministischen oder aber eines stochastischen Rahmenwerkes betrachtet werden. In jedem Fall hängen sie von einem Parameter $u \in \mathbb{R}^m$ ab. Dieser Parameter kann sich verändern, da er von der Zeit und/oder vom Zustand des Systems abhängt. Man bezeichnet ihn als Kontrolle, Kontrollvariable oder häufig auch als Steuergröße, das heißt als eine Größe, die sich von außen aktiv beeinflussen lässt. Man spricht in diesem Zusammenhang nun von dynamischen Kontrollsystemen.

Da auch der Nutzen eines Investors von einer oder mehreren Entscheidungen abhängt, in gewisser Weise also von außen durch den Entscheider beeinflussbar ist, führt uns dies weiter in das Gebiet der optimalen Kontrolltheorie. Bei unseren anschließend betrachteten Beispiel-Modellen handelt es sich dabei ausschließlich um deterministische, dynamische Modelle in kontinuierlicher Zeit. Des Weiteren werden dabei analog zu SEMMLER [15] maximal zwei Kontrollen existieren, welche der Investor steuern kann. Die eine Kontrolle modelliert die Entscheidung, wie viel konsumiert werden darf, die andere wie

viele Anteile von welchem Wertpapier gehalten werden.

Ziel des optimalen Steuerungsproblems ist es nun, entweder das Endvermögen des Investors oder seinen Nutzen – definiert über den Konsum – zu maximieren. Es muss also festgelegt werden, um wie viel der Nutzen mit jeder zusätzlich gewonnenen bzw. konsumierten Einheit steigt. In der dynamischen Portfolio-Optimierung wird dies durch eine Nutzenfunktion festgelegt, mit deren Hilfe sich die erwartete Rendite, das Risiko und der sich hieraus ergebende Nutzen für einen Investor in einen direkten funktionalen Zusammenhang bringen lassen. Insbesondere können nun mit Hilfe einer solchen Funktion Investitionen in der Weise gestaltet werden, dass erwartete Rendite und Risiko in einem für den Investor Nutzen maximierenden Verhältnis zueinander stehen.

Nutzenfunktionen liegen, je nach Präferenzen des Investors, in sehr unterschiedlichen Formen vor und können, wie bereits angedeutet, sowohl über das Vermögen als auch den Konsum definiert werden. Nach CAMPBELL UND VICEIRA [4] reichern Wirtschaftssubjekte ihr Vermögen hauptsächlich deshalb an, um davon entsprechend zu konsumieren und somit ihren Lebensstandard halten oder sogar verbessern zu können. Der Profit steigt demnach simultan zur Menge, die konsumiert werden kann. Aus diesem Blickwinkel scheint es sinnvoller, den Nutzen im Folgenden über den Konsum anstatt über das Vermögen zu definieren. Zur Modellierung der Präferenzen verwenden wir an dieser Stelle die sogenannte Power-Nutzenfunktion der Form

$$g(u) = \frac{u^{1-\gamma}}{1-\gamma} . \quad (2.1)$$

Bemerkung 2.1 Ein Spezialfall der Power-Nutzenfunktion ergibt sich für $\gamma \rightarrow 1$. Wir erhalten dann eine logarithmische Nutzenfunktion $g(u) = \ln(u)$. Dies muss im Programm berücksichtigt werden (siehe Anhang C).

2.2 Analyse der Gestalt der Power-Nutzenfunktion

Abschließend wollen wir im Hinblick auf eine Analyse der Beispiel-Modelle die Gestalt der Nutzenfunktion etwas genauer untersuchen. Um unsere Ergebnisse exakt formulieren zu können, müssen wir – ausschließlich für diesen Abschnitt – den Nutzen entgegen unserer vorherigen Festlegung über das Vermögen x definieren. Dass dies keine Auswirkungen auf unsere bisherigen Überlegungen hat, wird im Folgenden klar werden.

Wie bereits erwähnt, weisen Nutzenfunktionen abhängig von den Präferenzen eines Anlegers unterschiedliche Formen auf. Allgemein gilt zum Beispiel für Nutzenfunktionen eines risikoscheuen Entscheiders, dass die erste Ableitung positiv, die zweite negativ ist. Das heißt, jeder Marktteilnehmer, der risikoavers ist, handelt nach einer streng konkaven Nutzenfunktion. Das Ausmaß der Krümmung dieser Funktion (gegeben durch die 2. Ableitung) bestimmt den Grad der Risikoaversion eines Investors, welcher im Zusammenhang mit Investitionsentscheidungen eine wichtige Rolle spielt.

Es gilt: Umso stärker die Krümmung der Nutzenfunktion desto höher die Risikoaversion.

Die nachfolgende Tabelle fasst mögliche Krümmungen und deren Bedeutung noch einmal im Überblick zusammen:

Krümmungsverhalten	Bedeutung
konkave Nutzenfunktion	Risikoscheu
konvexe Nutzenfunktion	Risikofreudig
lineare Nutzenfunktion	Risikoneutral

Tabelle 2.1: Krümmungsverhalten der Nutzenfunktion und Bedeutung

Die zweite Ableitung als Ausdruck der Krümmung der Nutzenfunktion kann allerdings nicht direkt als Maß der Risikoaversion verwendet werden. Das hängt damit zusammen, dass Nutzenfunktionen, die identische Präferenzordnungen repräsentieren, unterschiedliche zweite Ableitungen aufweisen können. Daher ist eine Normierung der zweiten Ableitungen erforderlich, so dass zur Gewährleistung von Skalierungsunabhängigkeit durch die erste Ableitung geteilt werden sollte. Auf diese Weise erhält man als erstes Maß die sogenannte *absolute Risikoaversion* (ARA). Dabei handelt es sich um eine Rate für die Veränderung von g bei der Veränderung von x . Formal ist sie definiert durch

$$ARA(x) = - \frac{g''(x)}{g'(x)}. \quad (2.2)$$

Ein weiteres, vor allem für unsere Portfolio-Modelle bedeutendes Maß ist die *relative Risikoaversion* (RRA), gegeben durch

$$RRA(x) = - \frac{x \cdot g''(x)}{g'(x)}. \quad (2.3)$$

Dabei ist die Konvention in (2.2) bzw. (2.3) notwendig, um im Regelfall positive Werte zu erhalten. Die Kehrwerte werden als *absolute* und *relative Risikotoleranz* bezeichnet. Eine formale Herleitung dieser Maße findet man zum Beispiel in den Büchern von BREUER, GÜRTLER UND SCHUHMACHER [3] sowie HUANG UND LITZENBERGER [9].

Die absolute Risikoaversion lässt sich nun in der Weise interpretieren, dass sie für spekulative Anlagen derjenige Betrag ist, den der Investor zur Vermeidung eines Risikos mit gegebener absoluter Größe bereit ist, zu zahlen. Im Allgemeinen wird davon ausgegangen, dass die absolute Risikoaversion mit zunehmendem Vermögen abnimmt, bzw. zumindest nicht zunimmt. So wird für einen Millionär, im Gegensatz zu einer armen Person, ein finanzielles Wagnis unter Umständen nur eine geringe Bedeutung haben. Er wird also zur Vermeidung des Risikos dementsprechend weniger zahlen als ein armer Anleger.

Die relative Risikoaversion ist hingegen der Anteil des Vermögens, den ein Investor zur Vermeidung eines Risikos bei gegebenem relativen Vermögen bereit ist, einzusetzen. Hierbei wird davon ausgegangen, dass die relative Risikoaversion unabhängig vom Vermögen ist. Zusammenfassend ist das ARA-Maß absolut, weil es nicht in Beziehung zu

den „Konsequenzen“ einer spekulativen Anlage (bzw. der Höhe des Vermögens) gesetzt wird. Dies geschieht beim RRA-Maß. Vergleiche hierzu EISENFÜHR UND WEBER [5].

Betrachten wir abschließend die Resultate speziell für die im Folgenden verwendete Power-Nutzenfunktion, welche in Abbildung 2.1 für verschiedene Werte von γ , definiert über das Vermögen x , dargestellt ist.

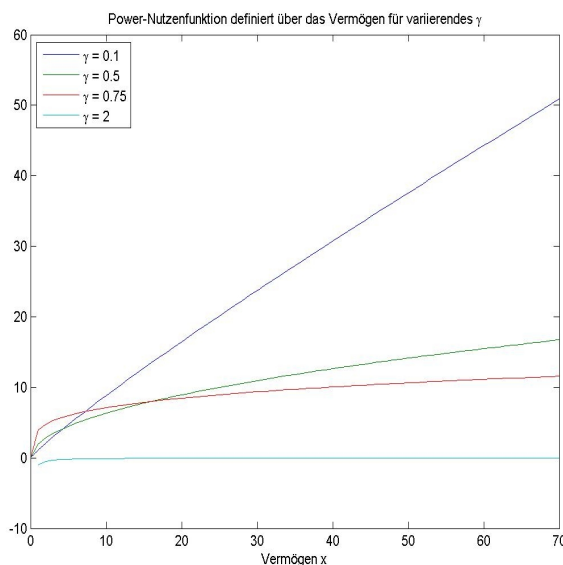


Abbildung 2.1: Die Grafik zeigt die verwendete Power-Nutzenfunktion für variierendes γ , definiert über das Vermögen.

Diese Form der Nutzenfunktion ist streng monoton wachsend, so dass jede zusätzliche Einheit an Vermögen auch zu mehr Nutzen führt. Ferner modelliert die Power-Nutzenfunktion die Präferenzen eines risikoscheuen Anlegers. Sie verläuft somit strikt konkav, was insbesondere bedeutet, dass $g'(x)$ streng monoton fällt. Damit liegt ein abnehmender Grenznutzen vor. Das heißt, der Nutzenzuwachs aus einer zusätzlichen Geldeinheit nimmt mit wachsendem Vermögen ab. Somit ist der Grenznutzen im Punkt $x = 0$ unendlich, nach dem Motto „wenig ist sehr viel besser als gar nichts“, und verschwindet für $x \rightarrow \infty$, was man als „Sättigungseffekt“ bezeichnen kann.

Während nun die absolute Risikoaversion bei Verwendung der Power-Nutzenfunktion

abnehmend in x ist, ist die relative Risikoaversion unabhängig vom Vermögen und somit konstant. In den nachfolgenden Modellen werden wir die Risikoaversion eines Investors über die relative Risikoaversion definieren, so dass mit (2.3) für die verwendete Power-Nutzenfunktion $RRA(x) = \gamma$ gilt.

Um nun besser auf das Ziel, die Kontrollen u so zu wählen, dass der Nutzen (definiert über den Konsum) maximal wird, eingehen zu können, sollen im nächsten Kapitel einige theoretische Grundlagen und Definitionen aus der Kontrolltheorie behandelt werden.

3 Theoretische Grundlagen

In diesem Kapitel sollen nun die zur Definition und Lösung eines Portfolio-Problems notwendigen mathematischen Voraussetzungen aus der Kontrolltheorie hergeleitet werden.

Im ersten Abschnitt wird das diskontierte optimale Steuerungsproblem definiert, wobei wir unsere Ausführungen auf ein deterministisches Kontrollproblem beschränken. Dabei bezeichne

- $x(t) \in \mathbb{R}^d$: den Zustandsvektor eines Systems zur Zeit t
- $u(t) \in \mathbb{R}^m$: den Steuervektor zur Zeit t
- t_0 : die Anfangszeit des Prozesses
- T : die Endzeit des Prozesses, $T > t_0$.

Die Kontrollfunktionen $u(t)$ seien zeitabhängig. Ohne Einschränkung der Allgemeinheit setzen wir dabei stets $t_0 = 0$. Desweiteren kann die Endzeit T als *fest* oder *frei* gewählt werden. Im zweiten Fall wird die Zeit ein Teil der Lösung.

Im zweiten Abschnitt werden wir einige wichtige Eigenschaften der optimalen Wertefunktion dieses Steuerungsproblems herleiten. Im Einzelnen sind dies das BELLMAN'sche Optimalitätsprinzip, die daraus resultierende Eindeutigkeitsaussage und die Stetigkeit der optimalen Wertefunktion. Anschließend wird, ausgehend vom Optimalitätsprinzip, die HAMILTON-JACOBI-BELLMAN Gleichung hergeleitet, welche wir zur Analyse unseres ersten Beispiel-Modells in Kapitel 5 benötigen.

Im Hinblick auf eine spätere Implementierung wird im dritten Abschnitt die Diskretisierung kontinuierlicher Kontrollprobleme betrachtet. Neben der Diskretisierung in der Zeit werden die räumliche Diskretisierung und ein erstes Iterationsverfahren zur Berechnung der optimalen Wertefunktion vorgestellt.

Im vierten Abschnitt wird auf die numerische Berechnung approximativ optimaler Trajektorien eingegangen. Dabei wird wieder sowohl das zeitdiskrete als auch das zeitkontinuierliche Problem betrachtet.

Im fünften Teil wird das Iterationsverfahren aus dem dritten Abschnitt aus praktischen Gründen durch alternative Iterationsverfahren ersetzt. Speziell sind dies das kontrollierte GAUSS-SEIDEL-Verfahren, die Strategie-Iteration und eine Kombination beider Verfahren.

Dieses Kapitel orientiert sich in Aufbau und Argumentation an GRÜNE [8].

3.1 Ein deterministisches optimales Steuerungsproblem

Bemerkung 3.1 *Die Begriffe „Steuerung“ und „Kontrolle“ werden im Folgenden synonym verwendet.*

Ein optimaler Steuerprozess ist durch folgende Daten gegeben:

- (1) den zulässigen Bereich für die Steuerung
- (2) die Dynamik des Prozesses
- (3) die Randbedingungen für den Zustand bei $t = t_0$ und $t = T$
- (4) die zu maximierende Zielfunktion.

Legen wir also zunächst fest, welche Werte eine Kontrolle u für den zu untersuchenden Prozess annehmen darf.

Definition 3.2 (i) *Die nichtleere und kompakte Menge $\mathbf{U} \subseteq \mathbb{R}^m$ definiert die für den Kontrollvektor $u(t)$ zulässige Wertemenge. Sie wird als Kontrollwertebereich bezeichnet.*

(ii) *Mit \mathcal{U} bzw. \mathcal{U}_h bezeichnen wir den Raum der zulässigen Kontrollfunktionen, also*

$$\mathcal{U} := \{u : \mathbb{R} \rightarrow \mathbf{U} \mid u \text{ zulässig}\} \quad \text{bzw.} \quad \mathcal{U}_h := \{u_h : h\mathbb{Z} \rightarrow \mathbf{U} \mid u \text{ zulässig}\}.$$

Im diskreten Fall sind dabei alle Funktionen zulässig. Welche Klassen von Funktionen im kontinuierlichen Fall als „zulässig“ definiert werden, wird im Folgenden genauer festgelegt.

Die Dynamik, d.h. das Langzeitverhalten eines Systems bzw. der Lösungen, wird durch ein Kontrollsystem festgelegt. Es beschreibt ausgehend von einem Anfangswert die Bewegung eines Systems über die Zeit. Bei nachfolgenden Modell-Beispielen in kontinuierlicher Zeit ist die Dynamik der betrachteten optimalen Steuerprozesse stets durch ein System gewöhnlicher Differentialgleichungen festgelegt.

Aufgrund einer späteren Diskretisierung sollen an dieser Stelle neben Systemen in kontinuierlicher Zeit auch Kontrollsysteme in diskreter Zeit definiert werden.

Definition 3.3 (i) Ein Kontrollsystem in kontinuierlicher Zeit $\mathbf{T} = \mathbb{R}$ im \mathbb{R}^d , $d \in \mathbb{N}$, ist gegeben durch die gewöhnliche Differentialgleichung

$$\frac{d}{dt}x(t) = f(x(t), u(t)) \quad (3.1)$$

wobei $f : \mathbb{R}^d \times \mathbf{U} \rightarrow \mathbb{R}^d$ ein parameterabhängiges stetiges Vektorfeld ist.

(ii) Ein Kontrollsystem in diskreter Zeit $\mathbf{T} = h\mathbb{Z} = \{hk | k \in \mathbb{Z}\}$ für ein $h > 0$ im \mathbb{R}^d , $d \in \mathbb{N}$, ist gegeben durch die Differenzengleichung

$$x(t+h) = f_h(x(t), u(t)) \quad (3.2)$$

wobei $f_h : \mathbb{R}^d \times \mathbf{U} \rightarrow \mathbb{R}^d$ eine stetige Abbildung ist.

Bemerkung 3.4 Statt „ $\frac{d}{dt}x(t)$ “ wird im Folgenden kurz „ $\dot{x}(t)$ “ geschrieben.

Für ein Kontrollsystem existiert unter genau festgelegten Eigenschaften für jeden beliebigen Anfangswert und jede Kontrollfunktion eine eindeutige Lösung.

Definition 3.5 Eine eindeutige Funktion $x(t)$, die Lösung von (3.2) zum Anfangswert $x_0 \in \mathbb{R}^d$ und zur Kontrollfunktion $u_h \in \mathcal{U}_h$ ist, wird mit $\Phi_h(t, x_0, u_h)$ bezeichnet.

Im diskreten Fall kann die Existenz dieser eindeutigen Lösung durch Induktion gezeigt werden, das heißt zu jedem Anfangswert x_0 und jeder Kontrolle $u_h \in \mathcal{U}_h$ kann eine eindeutige Lösung $\Phi_h(t, x_0, u_h)$, $\Phi : h\mathbb{N}_0 \times \mathbb{R}^d \times \mathcal{U}_h$ mit

$$\Phi_h(0, x_0, u_h) = x_0 \quad \text{und} \quad \Phi_h(t + h, x_0, u_h) = f_h(\Phi_h(t, x_0, u_h), u_h(t))$$

gefunden werden.

Für Kontrollsysteme in kontinuierlicher Zeit ist der Nachweis etwas komplizierter, es ist aber auch hier unter bestimmten Voraussetzungen möglich für jeden beliebigen Anfangswert x_0 eine eindeutige Lösung zu finden. Dazu sei auf den Existenz- und Eindeutigkeitssatz von CARATHÉODORY in GRÜNE [8], Seite 5 verwiesen, welcher zeigt, dass man einen sinnvollen Lösungsbegriff für messbare Kontrollfunktionen erhält.

Definition 3.6 *Eine eindeutige Funktion $x(t)$, die den Satz von CARATHÉODORY erfüllt, wird mit $\Phi(t, x_0, u)$ bezeichnet und ist die Lösung von (3.1) zum Anfangswert $x_0 \in \mathbb{R}^d$ und zur Kontrollfunktion $u \in \mathcal{U}$.*

Im Folgenden legen wir daher nun im kontinuierlichen Fall die messbaren Kontrollfunktionen als zulässige Funktionen fest.

Bemerkung 3.7 *Auf Randbedingungen soll an dieser Stelle nicht explizit eingegangen werden. Im Folgenden soll die Standard-Randbedingung $x(0) = x_0$ gelten.*

Betrachten wir ausschließlich nur ein Kontrollsystem, kann es unendlich viele Lösungen geben. Ziel des Investors ist es aber, eine optimale Lösung für das Portfolio-Problem zu finden. Um zu dieser zu gelangen, muss für sie zunächst ein Gütemaß bestimmt werden. Realisiert wird dies durch eine Ziel- bzw. Nutzenfunktion, wie sie im Zusammenhang mit Portfolio-Modellen auch bezeichnet wird.

Dazu wird eine Funktion

$$g : \mathbb{R}^d \times \mathcal{U} \rightarrow \mathbb{R}$$

betrachtet.

Diese wird nun entlang einer Trajektorie $\Phi(t, x_0, u)$ bzw. $\Phi_h(t, x_0, u_h)$ integriert oder im diskreten Fall summiert. Dadurch erhält man einen Wert, der sowohl vom Anfangswert x_0 als auch von der Kontrollfunktion u abhängt. Die Aufgabe ist es nun, zu einem Anfangsvermögen $x_0 > 0$ eine Kontrolle $u \in \mathcal{U}$ zu finden, die diesen Wert maximiert.

Um einen Zahlungsstrom objektiv beurteilen und den Nutzen aus diesem messen zu können, wird ein sogenanntes *diskontiertes Funktional* eingeführt. Nimmt das Funktional einen möglichst großen Wert an, handelt es sich um einen „möglichst vorteilhaften Zahlungsstrom“.

Die folgende Definition formuliert das entsprechende optimale Steuerungsproblem noch einmal formal.

Definition 3.8 *Betrachte ein Kontrollsystem (3.2) bzw. (3.1).*

Für eine Funktion $g : \mathbb{R}^d \times \mathbf{U} \rightarrow \mathbb{R}$ und einen Parameter $\delta > 0$, $\delta \in \mathbb{R}$, definieren wir das diskontierte Funktional auf unendlichem Zeithorizont in kontinuierlicher Zeit als

$$J(x, u) := \int_0^{\infty} e^{-\delta t} g(\Phi(t, x, u), u(t)) dt \quad (3.3)$$

und in diskreter Zeit als

$$J_h(x, u_h) := h \sum_{j=0}^{\infty} \beta^j \cdot g(\Phi_h(jh, x, u_h), u_h(jh)) \quad (3.4)$$

mit $\beta := (1 - \delta h)$.

Das optimale Steuerungsproblem lautet dann: Bestimme die optimale Wertefunktion

$$v(x) := \sup_{u \in \mathcal{U}} J(x, u)$$

bzw.

$$v_h(x) := \sup_{u_h \in \mathcal{U}_h} J_h(x, u_h).$$

Dabei gelten folgende Annahmen:

- (1) *Der Kontrollwertebereich \mathbf{U} ist kompakt.*

(2) Die Funktion g ist stetig und erfülle

$$|g(x, u)| \leq M_g \quad \text{und} \quad |g(x_1, u) - g(x_2, u)| \leq L_g \|x_1 - x_2\|$$

für alle $x, x_1, x_2 \in \mathbb{R}^d$, alle $u \in \mathbf{U}$ und geeignete Konstanten $M_g, L_g > 0$, mit $M_g, L_g \in \mathbb{R}$.

(3) In kontinuierlicher Zeit erfülle das Kontrollsystem (3.1) die Voraussetzungen des Satzes von CARATHÉODORY, wobei die Lipschitz-Konstante $L_R = L$ unabhängig von R sei.

In diskreter Zeit existiere eine Konstante $L > 0$, so dass die Lipschitz-Abschätzung

$$\|f_h(x_1, u) - f_h(x_2, u)\| \leq (1 + Lh) \|x_1 - x_2\|$$

gilt für alle $x_1, x_2 \in \mathbb{R}^d$ und alle $u \in \mathbf{U}$.

Bemerkung 3.9 Das hier betrachtete Modellproblem des diskontierten Funktionals J stammt ursprünglich aus der Ökonomie. Betrachtet man den Diskontfaktor $e^{-\delta t}$ aus (3.3) zeigt sich, dass er für $t \rightarrow \infty$ monoton gegen 0 strebt. Das heißt, dass die Werte von $g(\Phi(t, x, u), u(t))$ bzw. – bezogen auf das Portfolio-Problem – der Ertrag/Nutzen in naher Zukunft stärker gewichtet wird, als derjenige in ferner Zukunft.

Bemerkung 3.10 (i) Analog lässt sich auch das entsprechende Minimierungsproblem mit der optimalen Wertefunktion

$$v(x) := \inf_{u \in \mathcal{U}} J(x, u) \quad \text{bzw.} \quad v_h(x) := \inf_{u_h \in \mathcal{U}_h} J_h(x, u_h). \quad (3.5)$$

formulieren.

(ii) Bei nachfolgenden Modellen verwenden wir „max“ statt „sup“, da wir voraussetzen, dass optimale Kontrollfunktionen $u \in \mathcal{U}$ bzw. $u_h \in \mathcal{U}_h$ existieren¹.

Wir wollen nun einige wichtige Eigenschaften der optimalen Wertefunktion herleiten, welche wir in nachfolgenden Abschnitten bzw. Kapiteln benötigen.

¹Die Existenz optimaler Kontrollfunktionen kann unter gewissen Voraussetzungen nachgewiesen werden, was den Rahmen dieser Arbeit jedoch sprengen würde.

3.2 Eigenschaften der optimalen Wertefunktion

3.2.1 Das Bellman'sche Optimalitätsprinzip

Das BELLMAN'sche Optimalitätsprinzip ist eine grundlegende Eigenschaft der optimalen Wertefunktion, aus der sich alle weiteren für uns wichtigen Eigenschaften herleiten lassen. So bildet dieses Prinzip die Grundlage für die numerische Approximation einer optimalen Lösung und liefert die Basis für die *Hamilton-Jacobi-Bellman Gleichung*, welche wir in Abschnitt 3.2.2 näher betrachten werden.

BELLMAN beschreibt das Optimalitätsprinzip, welches auch als Prinzip der Dynamischen Programmierung bekannt ist, in [2] mit folgenden Worten:

„An optimal policy has the property that whatever the initial state and initial decision are the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision.“

Die Idee des Optimalitätsprinzips besteht also darin, dass Endstücke optimaler Trajektorien zu jeder Zeit selbst optimale Trajektorien sind, oder anders gesagt, dass bei Vorliegen einer optimalen Kontrolle zu jeder Zeit optimal gesteuert wird.

Den optimalen Wert $v(x)$ bzw. $v_h(x)$ erhalten wir also, indem wir für eine beliebige endliche Zeit optimal steuern und den bis dahin erzielten Wert mit dem optimalen Wert im erreichten Punkt – entsprechend gewichtet – addieren.

Mathematisch lässt sich dieses Vorgehen mit dem folgenden Satz beschreiben.

Satz 3.11 *Betrachte das optimale Steuerungsproblem aus Definition 3.8. Dann erfüllt die optimale Wertefunktion $v(x)$ für jedes $x \in \mathbb{R}^d$ und jedes $T > 0$ die Gleichung*

$$v(x) = \sup_{u \in \mathcal{U}} \left\{ \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt + e^{-\delta T} v(\Phi(T, x, u)) \right\} \quad (3.6)$$

bzw. für jedes $k \in \mathbb{N}$ und $\beta := (1 - \delta h)$ die Gleichung

$$v_h(x) = \sup_{u_h \in \mathcal{U}_h} \left\{ h \sum_{j=0}^k \beta^j g(\Phi_h(jh, x, u_h), u_h(jh)) + \beta^{k+1} v_h(\Phi_h((k+1)h, x, u_h)) \right\}$$

Beweis: Wir zeigen die Gleichheit an dieser Stelle in kontinuierlicher Zeit durch Beweisen der entsprechenden Ungleichungen in beide Richtungen. Der diskrete Fall folgt analog.

(i) “ \leq “: Seien $x \in \mathbb{R}^d$, $T > 0$ und $u \in \mathcal{U}$ beliebig. Dann gilt

$$\begin{aligned}
 J(x, u) &= \int_0^{\infty} e^{-\delta t} g(\Phi(t, x, u), u(t)) dt \\
 &= \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt + \int_T^{\infty} e^{-\delta t} g(\Phi(t, x, u), u(t)) dt \\
 &= \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt \\
 &\quad + e^{-\delta T} \int_0^{\infty} e^{-\delta t} g(\Phi(t, \Phi(T, x, u), u(T + \cdot)), u(T + t)) dt \\
 &= \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt + e^{-\delta T} J(\Phi(T, x, u), u(T + \cdot)) \\
 &\leq \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt + e^{-\delta T} v(\Phi(T, x, u))
 \end{aligned}$$

Da $u \in \mathcal{U}$ beliebig war, gilt die Ungleichung auch für das Supremum und damit für $v(x)$.

(ii) “ \geq “: Seien $x \in \mathbb{R}^d$, $T > 0$ und $\varepsilon > 0$ beliebig und $u_1 \in \mathcal{U}$ so gewählt, dass

$$\begin{aligned}
 &\sup_{u \in \mathcal{U}} \left\{ \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt + e^{-\delta T} v(\Phi(T, x, u)) \right\} \quad (3.7) \\
 &\leq \int_0^T e^{-\delta t} g(\Phi(t, x, u_1), u_1(t)) dt + e^{-\delta T} v(\Phi(T, x, u_1)) + \varepsilon
 \end{aligned}$$

Dadurch ist u_1 auf $[0, T]$ festgelegt. Wähle nun $u_1|_{(T, \infty)}$ so, dass

$$J(\Phi(T, x, u_1), u_1(T + \cdot)) \geq v(\Phi(T, x, u_1)) - \varepsilon \quad (3.8)$$

Mit (3.8) eingesetzt in (3.7) ergibt sich dann

$$\begin{aligned}
 & \sup_{u \in \mathcal{U}} \left\{ \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt + e^{-\delta T} v(\Phi(T, x, u)) \right\} \\
 & \leq \int_0^T e^{-\delta t} g(\Phi(t, x, u_1), u_1(t)) dt + e^{-\delta T} J(\Phi(T, x, u_1), u_1(T + \cdot)) + (1 + e^{-\delta T}) \varepsilon \\
 & \leq \int_0^T e^{-\delta t} g(\Phi(t, x, u_1), u_1(t)) dt + \int_T^\infty e^{-\delta t} g(\Phi(t, x, u_1), u_1(t)) dt + (1 + e^{-\delta T}) \varepsilon \\
 & = J(x, u_1) + (1 + e^{-\delta T}) \varepsilon \leq v(x) + (1 + e^{-\delta T}) \varepsilon
 \end{aligned}$$

Da $\varepsilon > 0$ beliebig war, folgt somit die Behauptung. □

Der folgende Satz zeigt nun, dass das BELLMAN'sche Optimalitätsprinzip nicht nur eine Eigenschaft der optimalen Wertefunktion beschreibt, sondern dass sich aus diesem auch eine Eindeutigkeitsaussage für $v(x)$ ergibt.

Satz 3.12 *Betrachte das optimale Steuerungsproblem aus Definition 3.8 mit optimaler Wertefunktion v . Sei ein $T > 0$ gegeben und sei $w : \mathbb{R}^d \rightarrow \mathbb{R}$ eine beschränkte Funktion, die das Optimalitätsprinzip*

$$w(x) = \sup_{u \in \mathcal{U}} \left\{ \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt + e^{-\delta T} w(\Phi(T, x, u)) \right\}$$

(bzw. das Gegenstück in diskreter Zeit) für alle $x \in \mathbb{R}^d$ erfüllt. Dann ist $w = v$ (bzw. $w = v_h$).

Für den Beweis sei auf GRÜNE [8] verwiesen.

Um die optimale Wertefunktion $v(x)$ numerisch approximieren zu können, müssten wir betrachten, welche Regularitätseigenschaften sie besitzt.

Im Allgemeinen kann nicht davon ausgegangen werden, dass $v(x)$ differenzierbar ist; tatsächlich kann noch nicht einmal Lipschitz-Stetigkeit erwartet werden. Allerdings kann eine Abschwächung der Lipschitz-Stetigkeit, die sogenannte Hölder-Stetigkeit nachgewiesen werden.

Da wir im Folgenden davon ausgehen, dass $v(x)$ sowohl stetig als auch insbesondere im ersten Beispiel-Modell differenzierbar ist, wollen wir an dieser Stelle auf ausführliche Untersuchungen zur Stetigkeit verzichten. Eine weiterführende Behandlung dieses Themas findet sich in GRÜNE [8], Kapitel 2.

Stattdessen wollen wir uns einer letzten wichtigen Eigenschaft der Wertefunktion zuwenden.

3.2.2 Die Hamilton-Jacobi-Bellman Gleichung

Diese elementare Eigenschaft ist die Tatsache, dass die Wertefunktion dort, wo sie differenzierbar ist, eine bestimmte partielle Differentialgleichung erster Ordnung erfüllt, die HAMILTON-JACOBI-BELLMAN Gleichung (HJB).

Sie ist ein klassisches Hilfsmittel zur Lösung eines Steuerungsproblems, soll aber hier weniger zur Lösung, sondern vielmehr zur Analyse unserer numerischen Resultate beim ersten Modell-Beispiel aus Kapitel 5 herangezogen werden. Aus diesem Grund, aber auch wegen ihrer Bedeutung für die Theorie der optimalen Steuerung, wollen wir sie an dieser Stelle kurz vorstellen. Zum Beweis benötigen wir noch das folgende Lemma:

Lemma 3.13 *Sei $F : \mathbf{U} \rightarrow \mathbb{R}$ eine stetige Funktion auf $\mathbf{U} \subset \mathbb{R}^m$. Es gelte*

$$\sup_{u \in \mathcal{U}} \frac{1}{T} \int_0^T F(u(t)) dt \geq C(T) \quad \forall u \in \mathcal{U}, T > 0. \quad (3.9)$$

Dann folgt $\sup_{u \in \mathbf{U}} F(u) \geq C(T)$.

Beweis: Aus (3.9) folgt wegen

$$\sup_{u \in \mathcal{U}} \frac{1}{T} \int_0^T F(u(t)) dt \in \overline{co} \{F(u) | u \in \mathbf{U}\} \quad (3.10)$$

die Ungleichung $\sup \overline{co} \{F(u) | u \in \mathbf{U}\} \geq C(T)$, wobei co die konvexe Hülle bezeichnet und \overline{co} deren Abschluss.

Zu vorgegebenem $\varepsilon > 0$ gibt es also $F_\varepsilon = \sum_{j=0}^l \alpha_j F(u_j) \geq C(T) - \varepsilon$, $u_j \in \mathbf{U}$, $\sum_{j=0}^l \alpha_j = 1$. Wähle nun $u_\varepsilon \in \mathbf{U}$ so, dass $F(u_\varepsilon) = \max_{j \in \{1, \dots, l\}} F(u_j)$ gilt.

Es folgt $F(u_\varepsilon) \geq C(T) - \varepsilon$, $u_\varepsilon \in \mathbf{U}$. Da ε beliebig gewählt war, folgt die Behauptung. \square

Satz 3.14 *Gegeben sei ein optimales Steuerungsproblem aus Definition 3.8 mit optimaler Wertefunktion v . Betrachte die partielle Differentialgleichung*

$$\delta v(x) + \sup_{u \in \mathbf{U}} \{-Dv(x) \cdot f(x, u) - g(x, u)\} = 0, \quad (3.11)$$

die sogenannte Hamilton-Jacobi-Bellman Gleichung. Ist die optimale Wertefunktion v differenzierbar in $x \in \mathbb{R}^d$, dann gilt die Hamilton-Jacobi-Bellman Gleichung für v in diesem Punkt.

Beweis: Nach dem BELLMAN'schen Optimalitätsprinzip gilt für alle $T > 0$:

$$\sup_{u \in \mathcal{U}} \left\{ \frac{1}{T} (v(x) - e^{-\delta T} v(\Phi(T, x, u))) - \frac{1}{T} \int_0^T e^{-\delta t} g(\Phi(t, x, u), u(t)) dt \right\} = 0. \quad (3.12)$$

Die Beweisidee ist nun, (3.12) abzuleiten und $T \rightarrow 0$ gehen zu lassen. Wir zeigen so in zwei Schritten die beiden Ungleichungen

$$-Dv(x)f(x, u) + \delta v(x) \leq g(x, u) \quad (3.13)$$

und

$$\sup_{u \in \mathbf{U}} \{-Dv(x)f(x, u) + \delta v(x) - g(x, u)\} \geq 0. \quad (3.14)$$

Sei $u \in \mathbf{U}$ als konstante Kontrolle fest gewählt; X bezeichne die zugehörige Trajektorie mit Anfangswert x , d.h. $X(\cdot) := \Phi(\cdot, x, u)$. Damit gilt:

$$\frac{d}{dT} X(T) = f(X(T), u), \quad X(0) = x$$

für genügend kleine $T > 0$, d.h. solange $X(T)$ in \mathbb{R}^d liegt. Wegen (3.12) gilt für diese $T > 0$ also

$$\frac{1}{T} (v(x) - e^{-\delta T} v(X(T))) \leq \frac{1}{T} \int_0^T e^{-\delta t} g(X(t), u(t)) dt. \quad (3.15)$$

Da $X(T)$ differenzierbar ist in $T = 0$ und v differenzierbar ist in $X = x$, muss auch $\tilde{v}(T) := e^{-\delta T} v(X(T))$ in $T = 0$ differenzierbar sein. Da außerdem $\tilde{v}(0) = v(x)$ gilt, ergibt sich

$$\frac{d}{dT} \tilde{v}(0) = Dv(x) \frac{d}{dT} X(0) - \delta v(x) = Dv(x)f(x, u(0)) - \delta v(x), \quad (3.16)$$

also folgt (3.13) aus (3.15).

Für die umgekehrte Ungleichung betrachte (3.12). Aus dieser folgt

$$\sup_{u \in \mathcal{U}} \left\{ \frac{1}{T} \left(v(x) - v \left(x + \int_0^T f(X(t), u(t)) dt \right) \right) - \frac{1}{T} \int_0^T g(x, u(t)) dt + \delta v(x) \right\} \geq \alpha(T)$$

mit $\alpha(T) \rightarrow 0$ für $T \searrow 0$. Dies ergibt sich durch Ableiten der Ausdrücke unter Ausnutzung von (3.16) für hinreichend kleine $T > 0$.

Wegen der Differenzierbarkeit von v in x gilt

$$v(X) = v(x) + Dv(x)(X - x) + \|X - x\| \lambda(X),$$

wobei $\lambda(X) \rightarrow 0$ für $X \rightarrow x$. Mit $X := x + \int_0^T f(X(t), u(t)) dt$ erhalten wir

$$\sup_{u \in \mathcal{U}} \left\{ \frac{1}{T} \int_0^T -Dv(x) f(X(t), u(t)) dt - \frac{1}{T} \int_0^T g(x, u(t)) dt \right\} + \delta v(x) \geq \alpha(T) \rightarrow 0.$$

Also folgt auch

$$\sup_{u \in \mathcal{U}} \left\{ \frac{1}{T} \int_0^T -Dv(x) f(x, u(t)) - g(x, u(t)) dt \right\} + \delta v(x) \geq \alpha(T) \rightarrow 0.$$

Die Behauptung ergibt sich nun aus

$$F(\cdot) = F_x(\cdot) = -Dv(x) f(x, \cdot) - g(x, \cdot)$$

mit Lemma 3.13.

□

Im Allgemeinen sind die Voraussetzungen von Satz 3.14 nicht erfüllt, da die optimale Wertefunktion v zwar stetig, nicht jedoch differenzierbar ist. Für nichtdifferenzierbare Lösungen dieser partiellen Differentialgleichung lassen sich dennoch Aussagen zur Existenz und Eindeutigkeit treffen, die auf dem Konzept der sogenannten Viskositätslösungen basieren.

Da wir im Folgenden annehmen, dass die optimale Wertefunktion auch differenzierbar ist, sei an dieser Stelle für detailliertere Informationen auf die Bücher von BARDI UND CAPUZZO DOLCETTA [1] sowie EVANS [6] verwiesen.

3.3 Diskretisierung des optimalen Kontrollproblems

Im Folgenden soll nun die Diskretisierung eines optimalen Steuerungsproblems betrachtet werden. Man unterscheidet dabei sowohl eine zeitliche als auch eine räumliche Diskretisierung.

Des Weiteren definieren wir eine erste Iterationsformel zur Berechnung zeitdiskreter optimaler Wertefunktionen v_h .

3.3.1 Diskretisierung in der Zeit

Die Diskretisierung in der Zeit wird dann notwendig, wenn optimale Steuerungsprobleme in kontinuierlicher Zeit betrachtet werden, da eine kontinuierliche Betrachtung numerisch nicht realisierbar ist. Zunächst muss also überlegt werden, wie eine solche Diskretisierung zu erreichen ist, das heißt, wie ein kontinuierliches System (3.1) durch ein diskretes (3.2) approximiert werden kann.

Eine notwendige Grundlage bilden dabei numerische Einschrittverfahren, wobei an dieser Stelle das EULER-Verfahren definiert und verwendet werden soll:

Definition 3.15 (EULER-Verfahren für Kontrollsysteme)

Gegeben sei ein kontinuierliches Kontrollsystem f gemäß Gleichung (3.1). Für einen Zeitschritt $h > 0$, $h \in \mathbb{R}$, und einen Kontrollwert $u \in \mathbf{U}$ definieren wir das EULER-Verfahren als das durch die Abbildung

$$f_h(x, u) := x + hf(x, u)$$

definierte zeitdiskrete Kontrollsystem (3.2). Die Lösungen werden mit $\tilde{\Phi}_h(t, x_0, u_h)$ bezeichnet.

Wie in GRÜNE [8] gezeigt, kann zu jedem Kontrollsystem eine diskrete EULER-Approximation konstruiert werden. Diese liefert zu jedem Anfangswert $x \in \mathbb{R}^d$ und jeder diskreten Kontrollfunktion $u_h : h\mathbb{Z} \rightarrow \mathbf{U}$ eine zeitdiskrete approximative Lösung $\tilde{\Phi}_h(t, x, u_h)$.

Der folgende Satz fasst die Eigenschaften des EULER-Verfahrens zusammen, wobei $\bar{B}_R(0)$ den abgeschlossenen Ball mit Radius R um 0 im \mathbb{R}^d bzw. \mathbb{R}^m bezeichne:

Satz 3.16 *Betrachte ein Kontrollsystem f , das die Voraussetzungen des Satzes von CARATHÉODORY erfüllt. Dann gilt für das EULER-Verfahren aus Definition 3.15 und jede Konstante $R > 0$ die folgende Aussage:*

- (i) *Es existiert eine von R abhängige Konstante $K > 0$, so dass für jede Kontrollfunktion $u \in \mathcal{U}$ mit $\|u\|_\infty \leq R$ und jeden Anfangswert $x_0 \in \bar{B}_R(0)$ eine diskrete Kontrollfunktion $u_h \in \mathcal{U}_h$ existiert, mit der die Abschätzung*

$$\|\tilde{\Phi}_h(t, x_0, u_h) - \Phi(t, x_0, u)\| \leq K\sqrt{h}e^{Lt}$$

gilt, für alle $t \in h\mathbb{N}_0$ für die die Lösungen in $\bar{B}_R(0)$ liegen.

- (ii) *Umgekehrt existiert eine von R abhängige Konstante $K > 0$, so dass für jedes $x_0 \in \bar{B}_R(0)$ und jede diskrete Kontrollfunktion $u_h \in \mathcal{U}_h$ mit $\|u_h\|_\infty \leq R$ und die durch*

$$u(\tau) := u_h(t), \quad \tau \in [t, t+h), \quad t \in h\mathbb{N}_0$$

definierte stückweise konstante Kontrollfunktion die Abschätzung

$$\|\tilde{\Phi}_h(t, x_0, u_h) - \Phi(t, x_0, u)\| \leq Kh(e^{Lt} - 1)$$

gilt, für alle $t \in h\mathbb{N}_0$ für die die Lösungen in $\bar{B}_R(0)$ liegen.

Für ein gegebenes kontinuierliches optimales Steuerungsproblem wird deshalb das zu dieser EULER-Approximation gehörige diskrete optimale Steuerungsproblem

$$\tilde{v}_h(x) := \sup_{u_h \in \mathcal{U}_h} \tilde{J}_h(x, u_h) \quad \text{mit} \quad \tilde{J}_h(x, u_h) := h \sum_{j=0}^{\infty} \beta^j g(\tilde{\Phi}_h(jh, x, u_h), u_h(jh))$$

betrachtet.

Bemerkung 3.17 *Das EULER-Verfahren für Kontrollsysteme hat die Konvergenzordnung $O(\sqrt{h})$.*

Für Abschätzungen des Diskretisierungsfehlers sei verwiesen auf GRÜNE [8], Kapitel 3.

3.3.2 Ein Iterationsverfahren zur Berechnung der Wertefunktion

Um die zeitdiskrete optimale Wertefunktion v_h , insbesondere aber auch die im vorherigen Abschnitt eingeführte zeitliche Diskretisierung \tilde{v}_h berechnen zu können, soll nun ein erstes Iterationsverfahren definiert werden. Als Grundlage betrachten wir dazu das BELLMAN'sche Optimalitätsprinzip (3.6) aus Abschnitt 3.2.1, aus welchem sich eine Iterationsformel herleiten lässt. Diese werden wir im Folgenden als Basis für unsere numerische Approximation verwenden.

Definition 3.18 Wir definieren iterativ Funktionen $v_h^i : \mathbb{R}^d \rightarrow \mathbb{R}, i = 0, 1, \dots$ mittels $v_h^0(x) = 0$ und $v_h^{i+1}(x) = T_h(v_h^i)(x)$ für alle $x \in \mathbb{R}^d$.

Der Operator $T_h : C(\mathbb{R}^d, \mathbb{R}) \rightarrow C(\mathbb{R}^d, \mathbb{R})$ ist dabei gegeben durch

$$T_h(w)(x) := \max_{u \in \mathbf{U}} \{hg(x, u) + \beta w(f_h(x, u))\}$$

mit $\beta = 1 - \delta h$.

$C(\mathbb{R}^d, \mathbb{R})$ bezeichne hierbei die Menge der stetigen Funktionen von \mathbb{R}^d nach \mathbb{R} .

Untersuchungen zur Konvergenz der Funktionenfolge v_h^i gegen v_h sind auch hier wieder in GRÜNE, Kapitel 3. nachzulesen.

3.3.3 Diskretisierung im Raum

Nachdem die Basis für die Iteration definiert wurde, folgt nun der Schritt der räumlichen Diskretisierung. Diese wird dann benötigt, wenn es an die Implementierung der Iterationsvorschrift aus Definition 3.18 geht. Auch wenn alle Größen dieser Vorschrift im Rechner auswertbar sind, kann sie nicht direkt implementiert werden. Das hängt damit zusammen, dass die Funktion v_h^i – selbst für eine beliebig kleine kompakte Menge $\Omega \subset \mathbb{R}^d$ – für unendlich viele Stellen ausgewertet werden müsste. Um eine implementierbare Form der Vorschrift zu erhalten, werden wir nun Funktionen auf *Gittern* betrachten, die nur an endlich vielen Punkten zu berechnen sind. Wir müssen uns also zur Berechnung von v_h auf einen endlichdimensionalen Funktionenraum einschränken.

Bemerkung 3.19 *In nachfolgend dargestellten Methoden beschränken wir uns auf den zweidimensionalen Fall $d = 2$. Die Methoden lassen sich aber für den Fall $d > 2$ erweitern.*

Desweiteren nehmen wir aus technischen Gründen an, dass die kompakte Menge $\Omega \subset \mathbb{R}^2$, auf der v_h berechnet wird, ein *Rechteck* ist. Betrachten wir zunächst, was wir unter einem *Rechteckgitter* verstehen.

Definition 3.20 *Sei $\Omega \subset \mathbb{R}^2$ gegeben durch $\Omega = [a_1, b_1] \times [a_2, b_2]$ mit Werten $a_1 < b_1$ und $a_2 < b_2$. Ein (regelmäßiges) Rechteckgitter Γ auf Ω ist eine Menge von Rechtecken R_i , $i = 0, \dots, P-1$, $P = P_1 P_2$, mit Kantenlängen $k_1 = (b_1 - a_1)/P_1$ und $k_2 = (b_2 - a_2)/P_2$, so dass*

$$\bigcup_{i=0}^{P-1} R_i = \Omega \quad \text{und} \quad \text{int}R_i \cap \text{int}R_j = \emptyset \quad \text{für alle } i, j = 0, \dots, P-1, i \neq j.$$

Mit E_i , $i = 0, \dots, N-1$, $N = (P_1 + 1)(P_2 + 1)$ werden die Eckpunkte des Gitters bezeichnet. Den maximalen Durchmesser eines Rechtecks bezeichnen wir mit

$$k = \sqrt{k_1^2 + k_2^2}.$$

Außerdem muss ein geeigneter Funktionenraum definiert werden, welcher zur Approximation von v_h verwendet wird.

Definition 3.21 (i) *Sei $A \subset \mathbb{R}^2$. Eine Funktion $w : A \rightarrow \mathbb{R}$ heißt affin bilinear, falls es Konstanten $\alpha_0, \dots, \alpha_3$ gibt, so dass für alle $x = (x_1, x_2)^T \in A$ die Identität $w(x) = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_1 x_2$ gilt.*

(ii) *Betrachte eine rechteckförmige Menge $\Omega \subset \mathbb{R}^2$ mit Rechteckgitter Γ . Den Raum der stetigen und stückweise affin bilinearen Funktionen auf Ω bezüglich Γ definieren wir mit*

$$\mathcal{W} := \{w : \Omega \rightarrow \mathbb{R} \mid w \text{ ist stetig und } w|_{R_i} \text{ ist affin bilinear für jedes } i=0, \dots, P-1 \}.$$

Das folgende Lemma fasst wichtige Eigenschaften von \mathcal{W} zusammen, die bei der anschließenden Programmierung benötigt werden.

Lemma 3.22 (i) *Jede Funktion $w \in \mathcal{W}$ ist eindeutig durch die Werte $w(E_i)$ in den Eckpunkten des Gitters bestimmt.*

(ii) Für jedes Rechteck $R_i = [c_1, d_1] \times [c_2, d_2]$ mit den Eckpunkten

$$E_{i_0} = (c_1, c_2)^T, \quad E_{i_1} = (d_1, c_2)^T, \quad E_{i_2} = (c_1, d_2)^T, \quad E_{i_3} = (d_1, d_2)^T$$

lässt sich $w|_{R_i}$ für $x = (x_1, x_2)^T \in R_i$ schreiben als

$$w(x) = \sum_{j=0}^3 \mu_j(x) w(E_{i_j})$$

mit

$$\begin{aligned} \mu_0(x) &= (1 - y_1(x))(1 - y_2(x)), & \mu_1(x) &= y_1(x)(1 - y_2(x)), \\ \mu_2(x) &= (1 - y_1(x))y_2(x), & \mu_3(x) &= y_1(x)y_2(x) \end{aligned}$$

und

$$y_l(x) = \frac{x_l - c_l}{d_l - c_l} \quad \text{für } l = 1, 2.$$

Insbesondere gilt hierbei $\mu_j(x) \geq 0$ für $j = 0, \dots, 3$ und $\sum_{j=0}^3 \mu_j(x) = 1$.

Nun können wir also jede Funktion $w \in \mathcal{W}$ in den Eckpunkten des Gitters identifizieren. Insbesondere ist der Funktionenraum \mathcal{W} somit ein N -dimensionaler Vektorraum über \mathbb{R} . Zur approximativen Berechnung von v_h genügt es nun also, nur die Werte in den Eckpunkten des Gitters zu berechnen. Betrachtet man die iterative Vorschrift aus Definition 3.18 heisst das, dass der Operator T_h in jedem Schritt nur noch an den Eckpunkten E_i , $i = 0, \dots, N - 1$ ausgewertet werden muss.

Es ergibt sich eine Folge von Funktionen $\hat{v}_h^j \in \mathcal{W}$ durch

$$\hat{v}_h^{j+1}(E_i) = \max_{u \in \mathbb{U}} \left\{ hg(E_i, u) + \beta \hat{v}_h^j(f_h(E_i, u)) \right\},$$

mit $\beta = 1 - \delta h$.

Bemerkung 3.23 Der Term $\hat{v}_h^j(f_h(E_i, u))$ aus der obigen Funktionenfolge kann nicht direkt berechnet werden, da \hat{v}_h^j an der Stelle $f_h(E_i)$ nicht bekannt ist. Daher muss an dieser Stelle interpoliert werden.

Setzt man $V^j = (V_1^j, \dots, V_N^j)^T \in \mathbb{R}^N$ mit $V_i^j = v_h^j(E_i)$ kann die Iteration auf \mathcal{W} nun als eine Iteration auf N -dimensionalen Vektoren formuliert werden. Zu einem gegebenen Gitter lassen sich die Vektoren $V^j \in \mathbb{R}^N$ nun nach der folgenden Vorschrift sukzessive berechnen. In der Literatur wird dieses Verfahren demnach häufig auch als sukzessive Approximation bezeichnet.

Definition 3.24 *Betrachtet wird ein zeitdiskretes optimales Kontrollproblem und ein Rechteckgitter Γ mit P Rechtecken und N Eckpunkten.*

Zu jedem $u \in \mathbf{U}$ und jedem $i = 0, \dots, N - 1$ sei $B(i, u)$ der N -dimensionale Zeilenvektor, für den für jedes $w \in \mathcal{W}$ und $W = (w(E_0), \dots, w(E_{N-1}))^T \in \mathbb{R}^N$ mit der üblichen Matrixmultiplikation gilt

$$w(\Phi_h(E_i, u)) = B(i, u)W$$

Des Weiteren sei $G(i, u) = hg(E_i, u)$.

Dann berechnen wir die Vektoren V^j iterativ durch $V^0 := (0, \dots, 0)^T$ und dem Einzelschrittverfahren

$$V^{j+1} := V^j, \quad V_i^{j+1} := \max_{u \in \mathbf{U}} \left\{ G(i, u) + \beta B(i, u)V^{j+1} \right\} \quad \text{für } i = 0, \dots, N - 1.$$

oder dem Gesamtschrittverfahren

$$V_i^{j+1} := \max_{u \in \mathbf{U}} \left\{ G(i, u) + \beta B(i, u)V^j \right\} \quad \text{für } i = 0, \dots, N - 1.$$

Im Allgemeinen ist das Einzelschrittverfahren vorteilhafter, da hier eine schnellere Konvergenz zu erwarten ist. Das hängt damit zusammen, dass für jedes $i > 1$ bereits die aktuellen Werte V_k^{j+1} für $0 \leq k < i$ berücksichtigt werden. Außerdem lässt sich die Speicherung beim Einzelschrittverfahren auf einem einzigen Vektor durchführen, während hingegen beim Gesamtschrittverfahren zwei Vektoren V^j und V^{j+1} gespeichert werden müssen.

Die einfachste Art, das Maximum in Definition 3.24 für jedes $i = 1, \dots, N$ zu bestimmen, besteht nun darin, die Werte

$$G(x, u_k) + \beta B(i, u_k)V^j, \quad k = 1, \dots, q$$

bzw.

$$G(x, u_k) + \beta B(i, u_k)V^{j+1}, \quad k = 1, \dots, q$$

zu vergleichen. Dazu müssen wir allerdings U als endliche Menge annehmen (i.Allg. als endliche Approximation einer gegebenen kontinuierlichen Menge), das heißt, wir setzen $U = \{u_1, \dots, u_q\}$ für ein $q \in \mathbb{N}$.

Der aufwändigste Teil des Algorithmus besteht nun darin, die Werte $G(i, u_k)$ und die Vektoren $B(i, u_k)$ zu berechnen. Falls genügend Speicherplatz vorhanden ist, ist es sinnvoll, diese im Voraus zu berechnen und zu speichern. Natürlich sollten dabei die Vektoren nicht komplett, sondern nur die Einträge ungleich Null gespeichert werden.

Zum Abschluss muss noch ein Abbruchkriterium für die Iteration formuliert werden, welches wir über die Konvergenz der Vektoren $V^j \xrightarrow{j \rightarrow \infty} V$ erhalten.

Lemma 3.25 *Betrachte die Iterationsvorschrift aus Definition 3.24 und sei $\delta h < 1$. Dann konvergieren die Vektoren V^j für $j \rightarrow \infty$ komponentenweise gegen den Vektor V . Dieser ist eindeutig bestimmt durch*

$$V_i = \max_{u \in U} \{G(i, u) + \beta B(i, u)V\} \quad \text{für } i = 0, \dots, N - 1.$$

Für die mit \hat{v}_h^j , $j = 1, \dots, \infty$ und \hat{v}_h bezeichneten zugehörigen Funktionen aus \mathcal{W} gilt darüber hinaus:

Falls $|V_i^j - V_i^{j+1}| \leq \varepsilon$ für alle $i = 0, \dots, N - 1$, so folgt

$$\|\hat{v}_h^j - \hat{v}_h\|_\infty \leq \frac{\varepsilon}{h\delta}.$$

Für den Beweis von Lemma 3.25 und eine Abschätzung des Fehlers, der durch die Diskretisierung im Ort entsteht, sei auf GRÜNE [8] verwiesen.

3.4 Berechnung approximativ optimaler Trajektorien

Nachdem nun weitgehend alle notwendigen mathematischen Grundlagen festgelegt sind, soll in den folgenden Abschnitten noch einmal näher auf die Numerik des optimalen Steuerungsproblems eingegangen werden. Es wird gezeigt, wie sich aus der approximativen Wertefunktion approximativ optimale Trajektorien gewinnen lassen. Wie schon im Vorfeld, sollen auch hier zunächst zeitdiskrete optimale Trajektorien und im Anschluss wieder das zeitkontinuierliche Problem betrachtet werden. Die entsprechenden Beweise sollen ebenso wie in den vorausgegangenen Abschnitten vernachlässigt werden und sind in GRÜNE [8] nachzulesen.

3.4.1 Zeitdiskrete optimale Trajektorien

Zur Betrachtung des zeitdiskreten Problems müssen wir zunächst annehmen, dass uns v_h bekannt ist. Zudem rufen wir uns noch einmal das Optimalitätsprinzip für v_h aus Abschnitt 3.2.1 in Erinnerung. Dieses kann für $k = 0$ als

$$v_h(x) = \max_{u \in \mathbf{U}} \{hg(x, u) + \beta v_h(f_h(x, u))\} \quad (3.17)$$

geschrieben werden und bildet die Grundlage der folgenden Definition:

Definition 3.26 *Wir definieren eine Abbildung $u^* : \mathbb{R}^d \rightarrow \mathbf{U}$, indem wir zu jedem $x \in \mathbb{R}^d$ ein $u_x \in \mathbf{U}$ wählen, so dass das Maximum in (3.17) angenommen wird (dieses u_x existiert, wir aber i.A. nicht eindeutig sein), und $u^*(x) = u_x$ setzen.*

Auf diese Weise wird nun eine Kontrollstrategie definiert, die nicht wie bisher von der Zeit t , sondern vom aktuellen Zustand x abhängt. Eine derartige Kontrollstrategie nennt man auch Zustandsrückführung oder Zustandsfeedback.

Zu jedem Anfangswert x lässt sich nun mittels u^* eine diskrete Kontrollfunktion $u_h^x \in \mathcal{U}_h$ gemäß der folgenden Iterationsvorschrift definieren:

$$u_h^x(0) := u^*(x), \quad u_h^x(ih) = u^*(\Phi_h(ih, x, u_h^x)) \quad \text{für } i = 1, 2, \dots \quad (3.18)$$

Die so definierte diskrete Kontrollfunktion ist optimal für das zeitdiskrete optimale Kontrollproblem, das heißt, für alle $x \in \mathbb{R}^d$ gilt

$$J_h(x, u_h^x) = v_h(x).$$

Einen entsprechenden Nachweis hierfür liefert Grüne in [8], Kapitel 4.

3.4.2 Numerische Berechnung approximativ optimaler Kontrollen

Für die Berechnung approximativ optimaler Kontrollen muss zunächst analog zu Definition 3.26 eine numerische Kontrollstrategie \hat{u}^* festgelegt werden. Dazu betrachten wir den Ausdruck

$$hg(x, u) + \beta \hat{v}_h(f_h(x, u)) \quad (3.19)$$

für die Funktion \hat{v}_h aus Lemma 3.25.

Definition 3.27 Wir definieren eine Abbildung $\hat{u}^* : \Omega \rightarrow \mathbf{U}$, indem wir zu jedem $x \in \Omega$ ein $u_x \in \mathbf{U}$ wählen, so dass das Maximum in (3.19) angenommen wird (dieses u_x existiert, wird aber i.A. wiederum nicht eindeutig sein), und $\hat{u}^*(x) = u_x$ setzen.

Analog zu (3.18) definieren wir nun zu jedem Anfangswert x eine diskrete Kontrollfunktion \hat{u}_h^x gemäß der folgenden iterativen Vorschrift:

$$\hat{u}_h^x(0) := \hat{u}^*(x), \quad \hat{u}_h^x(ih) = \hat{u}^*(\Phi_h(ih, x, \hat{u}_h^x)) \quad \text{für } i = 1, 2, \dots \quad (3.20)$$

Satz 3.28 Die in (3.20) definierte diskrete Kontrollfunktion $\hat{u}_h^x \in \mathcal{U}_h$ ist approximativ optimal für das zeitdiskrete optimale Kontrollproblem. Genauer gilt für alle $x \in \mathbb{R}^d$ die Abschätzung

$$|J_h(x, \hat{u}_h^x) - v_h(x)| \leq C \frac{k^\gamma}{h^{\gamma+1}}$$

für $\gamma = 1$, falls $\delta > L$, $\gamma \in (0, 1)$ beliebig, falls $\delta = L$, $\gamma = \delta/L$, falls $\delta < L$ und eine geeignete Konstante $C > 0$. L bezeichne hierbei, wie üblich, die Lipschitz-Konstante des Vektorfeldes f .

3.4.3 Das zeitkontinuierliche Problem

Falls es sich bei dem betrachteten zeitdiskreten Problem um eine Diskretisierung eines kontinuierlichen Problems handelt, soll nun abschließend gezeigt werden, wie aus der approximativ optimalen diskreten Kontrollfunktion \hat{u}_h^x eine messbare Kontrollfunktion \hat{u}^x konstruiert werden kann, die auch für das ursprüngliche zeitkontinuierliche Problem approximativ optimal ist. Hierzu setzen wir

$$\hat{u}^x(t) := \hat{u}_h^x(ih), \quad t \in [ih, (i+1)h). \quad (3.21)$$

Satz 3.29 *Die in (3.21) definierte stückweise konstante kontinuierliche Kontrollfunktion $\hat{u}^x \in \mathcal{U}$ ist approximativ optimal für das optimale Steuerungsproblem aus Definition 3.8. Genauer gilt für alle $x \in \mathbb{R}^d$ die Abschätzung*

$$|J(x, \hat{u}^x) - v(x)| \leq C \left(h^{\gamma/2} + \frac{k^\gamma}{h^{\gamma+1}} \right)$$

für $\gamma \in (0, 1]$ aus Satz 3.28 und eine geeignete Konstante $C > 0$.

3.5 Alternative Iterationsverfahren

Wie bereits erwähnt, zeigt sich in der Praxis, dass die Iterationsvorschrift zur Berechnung von V aus Definition 3.24 insbesondere bei Problemen mit kleinen $\delta > 0$ und $h > 0$ nicht sehr schnell konvergiert. Im Hinblick auf die Implementierung wäre es also sinnvoller, Methoden zu verwenden, mit denen die iterative Berechnung der optimalen Wertefunktion effizienter durchgeführt werden kann. Daher sollen in diesem Abschnitt zwei Methoden vorgestellt werden, die eine schnellere Konvergenz als das bisher betrachtete Verfahren aufweisen.

3.5.1 Das kontrollierte Gauß-Seidel-Verfahren

Um die Idee des kontrollierten Gauß-Seidel-Verfahrens zu verstehen, erinnern wir zunächst noch einmal an die aus Abschnitt 3.3.2 bekannte sukzessive Approximation.

In einer etwas anderen Notation lautet sie:

$$V^0 := (0, \dots, 0)^T; \quad V^{j+1} := V^j, \quad V_i^{j+1} := S(V^{j+1})_i, \quad i = 0, \dots, N-1, j = 0, 1, \dots \quad (3.22)$$

mit

$$S(W)_i = \max_{u \in \mathbf{U}} \left\{ G(i, u) + \beta \sum_{k=0}^{N-1} B(i, u)_k W_k \right\}$$

für $W \in \mathbb{R}^N$.

Die Idee des Verfahrens besteht nun darin, die letzte Zuweisung in (3.22) als Gleichung aufzufassen. Man versucht also einen Wert V_i^{j+1} zu bestimmen, so dass

$$V_i^{j+1} = S(V^{j+1})_i \quad (3.23)$$

erfüllt ist. Diese Gleichung ist explizit lösbar, denn es gilt

$$\begin{aligned} V_i^{j+1} = S(V^{j+1})_i &= \max_{u \in \mathbf{U}} \left\{ G(i, u) + \beta \sum_{k=0}^{N-1} B(i, u)_k V_k^{j+1} \right\} \\ \iff &\begin{cases} \forall u \in \mathbf{U} : V_i^{j+1} \geq G(i, u) + \beta \sum_{k=0}^{N-1} B(i, u)_k V_k^{j+1} \\ \exists u \in \mathbf{U} : V_i^{j+1} = G(i, u) + \beta \sum_{k=0}^{N-1} B(i, u)_k V_k^{j+1} \end{cases} \\ \iff &\begin{cases} \forall u \in \mathbf{U} : V_i^{j+1} \geq G(i, u) + \beta \sum_{\substack{k=0 \\ k \neq i}}^{N-1} B(i, u)_k V_k^{j+1} + \beta B(i, u)_i V_i^{j+1} \\ \exists u \in \mathbf{U} : V_i^{j+1} = G(i, u) + \beta \sum_{\substack{k=0 \\ k \neq i}}^{N-1} B(i, u)_k V_k^{j+1} + \beta B(i, u)_i V_i^{j+1} \end{cases} \\ \iff &\begin{cases} \forall u \in \mathbf{U} : V_i^{j+1} \geq \frac{G(i, u) + \beta \sum_{\substack{k=0 \\ k \neq i}}^{N-1} B(i, u)_k V_k^{j+1}}{1 - \beta B(i, u)_i} \\ \exists u \in \mathbf{U} : V_i^{j+1} = \frac{G(i, u) + \beta \sum_{\substack{k=0 \\ k \neq i}}^{N-1} B(i, u)_k V_k^{j+1}}{1 - \beta B(i, u)_i} \end{cases} \end{aligned}$$

$$\iff V_i^{j+1} = \max_{u \in \mathbf{U}} \left\{ \frac{G(i, u) + \beta \sum_{\substack{k=0 \\ k \neq i}}^{N-1} B(i, u)_k V_k^{j+1}}{1 - \beta B(i, u)_i} \right\}$$

Analog zur obigen Iteration definieren wir demnach das kontrollierte GAUSS-SEIDEL-Verfahren durch

$$V^0 := (0, \dots, 0)^T; \quad V^{j+1} := V^j, \quad V_i^{j+1} := \tilde{S}(V^{j+1})_i, \quad i = 0, \dots, N-1, j = 0, 1, \dots \quad (3.24)$$

mit

$$\tilde{S}(W)_i = \max_{u \in \mathbf{U}} \left\{ \frac{G(i, u) + \beta \sum_{\substack{k=0 \\ k \neq i}}^{N-1} B(i, u)_k W_k}{1 - \beta B(i, u)_i} \right\}$$

für $W \in \mathbb{R}^N$.

Für eine geometrische Interpretation des Verfahrens bzw. für eine Konvergenz-Analyse, aus welcher hervorgeht, dass das Abbruchkriterium aus Lemma 3.25 auch für die Iteration (3.24) gilt, sei auf GRÜNE [8], Kapitel 4, verwiesen.

3.5.2 Die Strategie-Iteration

Eine weitere Alternative, die schneller konvergiert, ist die bereits seit den 60er Jahren bekannte Strategie-Iteration. Diese unterscheidet sich von den vorherigen Verfahren vor allem dadurch, dass hier die Maximierung wegfällt.

Wählen wir zunächst einen Vektor $\tilde{u} = (\tilde{u}_0, \dots, \tilde{u}_{N-1})^T \in \mathbf{U}^N$. Statt des maximierenden $u \in \mathbf{U}$ setzen wir nun in $S(V_i^{j+1})$ den Kontrollwert \tilde{u}_i ein. Die Iteration konvergiert nun gegen einen Vektor $V^{\tilde{u}}$, der durch das Gleichungssystem

$$V_i^{\tilde{u}} = G(i, \tilde{u}_i) + \beta B(i, \tilde{u}_i) V^{\tilde{u}}, \quad i = 0, \dots, N-1 \quad (3.25)$$

eindeutig bestimmt ist.

Man sieht leicht, dass $V_i^{\tilde{u}} \leq V_i$ für alle $i = 0, \dots, N-1$ gilt.

Da nun, wie bereits erwähnt, die Maximierung wegfällt, lässt sich der Vektor $V^{\tilde{u}}$ viel schneller als V berechnen. Zur Lösung des linearen Gleichungssystems (3.25) kann zum

Beispiel das (klassische) GAUSS-SEIDEL-Verfahren verwendet werden, das durch die Iteration (3.24) ohne Maximierung gegeben ist:

$$V^{\tilde{u},j+1} := V^{\tilde{u},j}, \quad V_i^{\tilde{u},j+1} := \tilde{S}(V^{j+1}, \tilde{u})_i, \quad i = 0, \dots, N-1, \quad j = 0, 1, \dots \quad (3.26)$$

mit geeignetem Startvektor $V^{\tilde{u},0}$ und

$$\tilde{S}(W, \tilde{u})_i = \frac{G(i, \tilde{u}_i) + \beta \sum_{\substack{k=0 \\ k \neq i}}^{N-1} B(i, \tilde{u}_i)_k W_k}{1 - \beta B(i, \tilde{u}_i)_i}$$

für $W \in \mathbb{R}^N, \tilde{u} \in \mathbb{R}^N$.

Die Idee der Strategie-Iteration besteht nun darin, zu gegebenem V^j einen Kontrollvektor \tilde{u}^j so zu wählen, dass dessen Komponenten gerade die maximierenden Kontrollen für die Iterationsvorschrift $S(V^j)_i$ sind, und dann damit $V^{j+1} = V^{\tilde{u}^j}$ zu berechnen.

Abschließend soll das Verfahren noch einmal formal beschrieben werden:

- (1) Setze $V^0 := (0, \dots, 0)^T \in \mathbb{R}^N, j = 0$
- (2) Wähle $\tilde{u}^j \in \mathbf{U}^N$ mit $\tilde{S}(V^j)_i = G(i, \tilde{u}_i^j) + \beta B(i, \tilde{u}_i^j) V^j$ für $i = 0, \dots, N-1$
- (3) Berechne $V^{j+1} = V^{\tilde{u}^j}$
- (4) Falls $\|V^j - V^{j+1}\|_\infty > \varepsilon$ setze $j := j + 1$ und gehe zu (2).

Schritt (3) kann hierbei z.B. mit dem erwähnten (klassischen) Gauß-Seidel-Verfahren berechnet werden. Der Anfangsvektor ist dabei als $V^{\tilde{u},0} = V^j$ zu wählen.

In der Praxis hat sich gezeigt, dass dieses Verfahren am Anfang allerdings recht langsam konvergiert, häufig deutlich langsamer als das kontrollierte GAUSS-SEIDEL-Verfahren (3.24). Eine Möglichkeit, dieses Problem zu umgehen, besteht nun darin, beide Verfahren miteinander zu kombinieren. Das resultierende Verfahren wird im nachfolgenden Kapitel ein wichtiger Bestandteil unseres Algorithmus zur Lösung optimaler Steuerungsprobleme sein. Daher soll auch dieses abschließend kurz formal beschrieben werden.

- (1) Setze $V^0 := (0, \dots, 0)^T \in \mathbb{R}^N$, $k = 0$
- (2) (a) Setze $V^{k+1} := V^k$ und berechne $V_i^{k+1} := \tilde{S}(V^{k+1})_i$, $i = 0, \dots, N - 1$;
speichere dabei die maximierenden Kontrollwerte in $\tilde{u}^k \in \mathbb{U}^N$
- (b) Falls $\|V^{k+1} - V^k\|_\infty \leq \varepsilon$ beende den Algorithmus;
Falls ein Abbruchkriterium (s.u.) erfüllt ist, gehe zu (3);
ansonsten setze $k := k + 1$ und gehe zu (2a)
- (3) Berechne $V^{k+1} = V^{\tilde{u}^k}$, setze $k := k + 1$ und gehe zu (2)

Bemerkung 3.30 *Die Frage nach einem guten Abbruchkriterium in (2b) kann nur experimentell beantwortet werden und hängt stark vom zugrunde liegenden optimalen Steuerungsproblem ab. Sie dazu auch GRÜNE [8], Kapitel 4.*

4 Numerische Lösung der Entscheidungsmodelle

Zur numerischen Lösung des Portfolio-Problems werden wir uns im Folgenden mit der Implementierung der in Kapitel 3 behandelten Methoden beschäftigen. Grundlage bildet neben diesem Kapitel ein C++-Programm, welches an der Universität Bayreuth im Rahmen der Vorlesung *Numerische Dynamik von Kontrollsystemen* [8] entstanden ist. Der C++-Code besteht dabei aus mehreren Routinen, welche an dieser Stelle systematisch beschrieben werden sollen.

Für einen besseren Überblick werden wir im ersten Abschnitt zunächst auf Routinen zur Verwaltung gleichmäßiger Gitter eingehen.

Der zweite Abschnitt behandelt Routinen, welche zur Berechnung der approximativ optimalen Werte- und Kontrollfunktion sowie optimaler Trajektorien erforderlich sind. Die beschriebenen Routinen müssen dabei in der Regel nicht abgeändert werden. Funktionen, die speziell für die zu untersuchenden Modelle sind und daher für jedes Beispiel angepasst werden müssen, sind zum einen die Routine $rS(\cdot)$, welche die entsprechende rechte Seite der Differential- bzw. Differenzgleichung beinhaltet, zum anderen die Funktion $g(\cdot)$, die die Ziel- bzw. Nutzenfunktion beschreibt. Auf diese beiden Routinen wird hier nicht näher eingegangen, da sie ausschließlich die Vorgaben des jeweiligen Modells enthalten. Das komplette Programm mit den jeweiligen Aufrufen der einzelnen Routinen befindet sich im Anhang C dieser Arbeit.

Im Folgenden sollen nun die einzelnen Programmteile vorgestellt, sowie deren Aufbau und ihre Funktion erläutert werden.

4.1 Die elementaren C++-Programmteile

4.1.1 Routinen zur Verwaltung gleichmäßiger Gitter

Bemerkung 4.1 Analog zu Kapitel 3 beschränken wir uns bei der Darstellung der Routinen auf den Fall $d = 2$.

Da wir aufgrund der räumlichen Diskretisierung Funktionen auf Gittern betrachten, benötigen wir zunächst Routinen zur Verwaltung gleichmäßiger Gitter für ein Rechteckgebiet $\Omega = [a_1, b_1] \times [a_2, b_2] \subset \mathbb{R}^2$ mit Werten $a_1 < b_1$ und $a_2 < b_2$. In der nachfolgenden Abbildung 4.1 ist zur Veranschaulichung ein Gitter mit neun Rechtecken R_0, \dots, R_8 und 16 Eckpunkten E_0, \dots, E_{15} dargestellt. Jedes dieser Rechtecke besitzt dabei vier lokale Ecken L_0, \dots, L_3 .

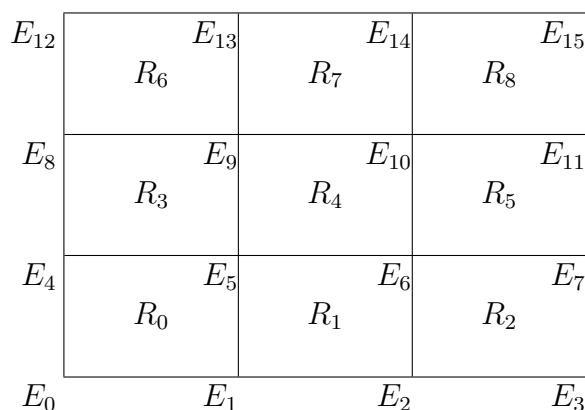


Abbildung 4.1: Beispiel-Rechteckgitter

Da die Ausgaben von der Nummerierung der Ecken abhängen, muss diese noch einheitlich festgelegt werden. Im Folgenden soll die Nummerierung konsistent mit der obigen Abbildung sein, wenn die lokale Nummerierung von unten links nach oben rechts zählt (vgl. Abbildung 4.2).

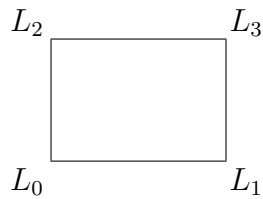


Abbildung 4.2: Lokale Nummerierung der Ecken

Des Weiteren bestehe das zugrunde liegende Gitter aus P_1 Rechtecken in x_1 -Richtung und P_2 Rechtecken in x_2 -Richtung. Es besitzt also $N = (P_1 + 1)(P_2 + 1)$ Eckpunkte. Die nachfolgende Tabelle 4.1 stellt einen kurzen Überblick über weitere, an dieser Stelle verwendete Variablen und Parameter, sowie eine kurze Erklärung ihrer Bedeutung dar.

Name in der Theorie	Name im Programm	Beschreibung
a_1, a_2	a[0], a[1]	untere Gittergrenze [Vektor]
b_1, b_2	b[0], b[1]	obere Gittergrenze [Vektor]
P_1, P_2	n[0], n[1]	Anzahl der Rechtecke in x_1 - bzw. x_2 -Richtung [Vektor]
x	x[0], x[1]	Punkt x aus Ω [Vektor]
$\mu_i, i = 0, \dots, 3$	koord[i]	bilineare Koordinaten [Vektor]
i	i	Rechteckindex
j	j	lokaler Eckenindex
l	l	globaler Eckenindex

Tabelle 4.1: Auflistung der elementaren Variablen bzw. Parameter und ihre Bedeutung.

Zur Verwaltung von Gittern muss zunächst zu einem Rechteck R_i und einer lokalen Ecke L_j der Index des globalen Eckpunktes E_l gefunden werden. Diesen liefert die Routine Gitter2d.

Die Routine: Gitter2d

Das Ziel dieser Routine ist es, zu einem gegebenen Rechteckindex i und einem lokalen Eckenindex j den globalen Index der Ecke $E_l = L_j$ zu berechnen. Man erhält diesen, indem sowohl der Rechteckindex in P_1 -Richtung i als auch der abgerundete ganzzahlige Index in P_2 -Richtung $\lfloor \frac{i}{n[0]} \rfloor$ zum lokalen Eckenindex j addiert wird. Werden die „oberen“ Ecken des Rechtecks erreicht, muss zudem die Anzahl der Rechtecke in x_1 -Richtung, vermindert um 1, addiert werden. Zuvor muss überprüft werden, ob die Rechtecknummer i bzw. die lokale Ecke j korrekt angegeben wurden.

INPUT: An die Funktion müssen die Variablen i, j und der Vektor n übergeben werden.

- Falls $j < 0$ oder $j > 3$
so gebe Warnung aus:
Falsche Angabe der lokalen Ecke, j aus $\{0, 1, 2, 3\}$.
RETURN(-1)
- Falls $i < 0$ oder $i > n[0] \cdot n[1] - 1$
so gebe Warnung aus:
Falsche Angabe der Rechtecknummer, i aus
 $(0, \dots, n[0] \cdot n[1] - 1)$.
RETURN(-1)
- Berechne den globalen Eckenindex l durch
$$(int) l = j + i + \frac{i}{n[0]}$$
- Falls $j > 1$ setze
$$l = l + n[0] - 1$$
- RETURN (l)

OUTPUT: globaler Eckenindex l .

Die Routine: FindeRechteck

Die Routine `FindeRechteck()` liefert zu einem Punkt $x = (x_1, x_2)^T$ im \mathbb{R}^2 den Index des Rechtecks, in dem dieser Punkt liegt. Das verwendete Gitter bedecke dabei wieder die rechteckförmige Menge $\Omega = [a_1, b_1] \times [a_2, b_2]$ und bestehe wie bisher aus $P_1 \cdot P_2$ Rechtecken. Von Bedeutung ist natürlich, ob der Punkt überhaupt in diesem Gebiet liegt, so dass x zunächst daraufhin getestet werden muss. Falls dies nicht der Fall ist, gibt die Routine als Rechteckindex -1 zurück.

Um den Index zu einem $x \in \Omega$ zu berechnen, werden wir das Gebiet Ω zunächst auf das Rechteck $[0, P_1] \times [0, P_2]$ transformieren. Die resultierenden Werte für x werden in x_0 und x_1 gespeichert. Um nun das gesuchte Rechteck, das heisst die zugehörige Spalte bzw. Zeile angeben zu können, muss zudem der ganzzahlige Wert von x_0 im Spaltenindex s und von x_1 im Zeilenindex z abgelegt werden.

Außerdem müssen wir noch die Ränder des Rechtecks berücksichtigen. So gehört zum einen auch der untere Rand zum Rechteck, zum anderen müssen wir den Spalten- bzw. Zeilenindex reduzieren, falls wir den oberen Rand von Ω erreichen. Wie diese Reduktion im Einzelnen umgesetzt wird, findet sich an entsprechender Stelle im Programm.

Der gesuchte Rechteckindex ergibt sich dann durch Multiplikation des Zeilenindex mit der Anzahl der Rechtecke in x_1 -Richtung, erhöht um den Spaltenindex s .

INPUT: An die Funktion müssen die Vektoren x, a, b und n übergeben werden.

- Falls der Punkt x ausserhalb von Ω liegt:

```
RETURN(-1)
```

- Transformiere Ω bzw. x durch Berechnung der Werte

$$x_0 = \left(\frac{x[0] - a[0]}{b[0] - a[0]} \right) \cdot n[0]$$

$$x_1 = \left(\frac{x[1] - a[1]}{b[1] - a[1]} \right) \cdot n[1]$$

- Speichere den ganzzahligen Teil von x_0 und x_1 im Spaltenindex s bzw. Zeilenindex z ab.

- Falls der obere Rand in $x[0]$ - bzw. $x[1]$ -Richtung erreicht wird, reduziere mit:

$$s = n[0] - 1 \quad \text{bzw.}$$

$$z = n[1] - 1$$

- Bestimme den Rechteckindex durch

$$z \cdot n[0] + s$$

- RETURN ($z \cdot n[0] + s$)

OUTPUT: Index des Rechtecks, in dem $x \in \Omega$ liegt.

Die Routine: Koordinaten

Mithilfe der Routine `Koordinaten()` sollen nun die Koordinaten des i -ten Eckpunktes in globaler Nummerierung bestimmt und im Array x zurückgegeben werden.

Aus dem globalen Eckenindex i ermitteln wir dazu zunächst den Spaltenindex s und den Zeilenindex z bezüglich der Unterteilung P_1 in x_1 -Richtung. Damit können nun die Koordinaten x_1 und x_2 auf Ω mit der gegebenen Diskretisierung P_1 bzw. P_2 ermittelt werden.

INPUT: An die Funktion müssen die Vektoren a, b, n, x und die Variable i übergeben werden.

- Falls $i < 0$ oder $i > ((n[0] + 1) \cdot (n[1] + 1))$

so gebe Warnung aus:

Globaler Eckenindex ausserhalb der Menge
 $(0, \dots, (n[0] + 1) \cdot (n[1] + 1) - 1)$. EXIT(0)

- Bestimme den Spalten- und Zeilenindex mittels

$$s = i \bmod (n[0] + 1) \quad \text{und}$$

$$z = \frac{i}{n[0] + 1}$$

- Ermittle nun die Koordinaten $x[0]$ und $x[1]$ auf Ω durch :

$$x[0] = a[0] + s \cdot \frac{b[0]-a[0]}{n[0]}$$

$$x[1] = a[1] + z \cdot \frac{b[1]-a[1]}{n[1]}$$

OUTPUT: Koordinaten des x -Vektors für den i -ten Eckpunkt.

Die Routine: **FindeRechteckKoordinaten**

Abschließend betrachten wir eine Erweiterung der im Vorfeld beschriebenen Routine `FindeRechteck()`. Auch hier wird zum einen wieder zu einem Punkt $x = (x_1, x_2)^T$ im \mathbb{R}^2 der Index des Rechtecks, in dem dieser liegt, zurückgegeben. Zum anderen werden nun aber auch zusätzlich dessen bilineare Koordinaten μ_i gemäß Lemma 3.22 berechnet und im Array `koord[i]` abgespeichert.

Wir gehen dabei zunächst analog zur vorherigen Routine vor, so dass auch an dieser Stelle die dort festgelegten Grundlagen gelten. Um die bilinearen Koordinaten zu bestimmen, benötigen wir nun allerdings auch den Nachkommaanteil von x_0 bzw. x_1 , welcher daher in y_0 bzw. y_1 abgelegt wird.

INPUT: An die Funktion müssen die Vektoren a, b, n, x und `koord` übergeben werden.

- Führe die Schritte 1 bis 4 gemäss der Funktion **FindeRechteck()** durch.
- Ermittle den Nachkommaanteil von x_0 bzw. x_1 und speichere ihn in y_0 und y_1 .
- Berechne $\mu_i, i = 0, \dots, 3$ gemäss der Funktionen aus Lemma 3.22 und speichere sie im Array `koord[i]`.
- RETURN ($z \cdot n[0] + s$).

OUTPUT: Rechteckindex, in dem $x \in \Omega$ liegt.

4.1.2 Routinen zur numerischen Lösung des optimalen Steuerungsproblems

In diesem Abschnitt werden nun Routinen zur numerischen Berechnung der optimalen Werte- und Kontrollfunktion sowie zugehöriger optimaler Trajektorien beschrieben. Neben den im vorherigen Abschnitt definierten Variablen und Parametern werden hier einige weitere benötigt, welche daher wieder kurz tabellarisch aufgeführt werden sollen.

Name in Theorie	Name im Programm	Beschreibung
x	<code>x[i] = (x[0], x[1])</code>	auszuwertender Punkt [Vektor]
\dot{x}	<code>y[i] = (y[0], y[1])</code>	linke Seite der DGL [Vektor]
v	<code>v</code>	Werte der Funktion auf den Gitterpunkten [Vektor]
u	<code>u</code>	(Vektor der) Kontrollwerte
—	<code>max</code>	maximaler Wert über alle u
—	<code>uopt</code>	u , welches das Maximum liefert
q	<code>q</code>	Länge des Vektors u
d	<code>d</code>	Dimension der DGL
h	<code>h</code>	Schrittweite
δ	<code>delta</code>	Diskontrate
ε	<code>eps</code>	Geforderte Genauigkeit
—	<code>flag</code>	flag = 1: Punkt liegt außerhalb des Gitters; flag = 0: Punkt liegt im Gitter
—	<code>ufind</code>	Gibt an, ob ein u gefunden wurde, so dass die Abbildung für x und u noch im Gitter liegt
—	<code>st</code>	Gibt die verwendete Strategie an; 0: Gauß-Seidel, 1: Strategie-Iteration

Tabelle 4.2: Auflistung der elementaren Variablen und Parameter sowie ihre Bedeutung.

Bemerkung 4.2 *Im Folgenden werden wir, wie in Kapitel 3 erwähnt, \mathbf{U} als endliche Menge annehmen. Es gilt also $\mathbf{U} = \{u_1, \dots, u_q\}$ für ein $q \in \mathbb{N}$.*

Die Routine: wert

Zu Beginn betrachten wir eine Routine, welche zum Punkt $x = (x_1, x_2)^T$ den approximierten Wert von v in diesem Punkt mittels einer affin bilinearen Funktion gemäß Definition 3.22 bestimmt. Hierbei werden die Werte $v(E_i)$ in den Eckpunkten im Array v gespeichert. Eine entscheidende Rolle spielt diese Routine zum Beispiel bei der Implementierung des Einzelschrittverfahrens aus Definition 3.24 oder, wie später in diesem Abschnitt, bei der approximativen Berechnung der optimalen Kontrolle u im Punkt x . Die Routine wird dabei zur Auswertung von $\hat{v}_h^{j+1}(f_h(E_i, u)) = B(i, u)V^{j+1}$ verwendet, da auf diese Weise die Vektoren $B(i, u)$ aus Definition 3.24 nicht explizit berechnet werden müssen. Zur Berechnung von $f_h(E_i, u)$ kann das Euler-Verfahren (3.15) verwendet werden, welches im Anschluß an diese Routine kurz beschrieben wird. Außerdem werden wir nun eine weitere (boolsche) Variable, bezeichnet mit `flag`, benötigen, um zu prüfen, ob x im Gitter liegt.

INPUT: An die Funktion müssen die Vektoren x, a, b, n und v und die boolsche Variable $flag$ übergeben werden.

- Bestimme den Rechteckindex von x und dessen bilineare Koordinaten mittels der Routine **FindeRechteckKoord()**.
Speichere den Rechteckindex für x in j (Rückgabewert).
- Falls x ausserhalb des Gitter liegt ($j < 0$)
Setze $flag$ auf 1. RETURN(0.)
- Initialisiere die Variable $wert$ mit 0.
- Berechne für alle bilinearen Koordinaten den approximierten Wert von v im Punkt x gemäß 3.22 :
$$wert = wert + koord[i] \cdot v[\mathbf{Gitter2d}(j, i, n)], i = 0, \dots, 3$$
- Setze $flag$ auf 0 zurück. RETURN(wert)

OUTPUT: 0 bzw. approximierter Wert v im Punkt x ($wert$) sowie 0 oder 1 für $flag$.

Die Routine: Euler

Mit dem Euler-Verfahren aus Definition 3.15 werden zu einem gegebenen Anfangswert x_0 und zur Schrittweite h Lösungstrajektorien berechnet.

Im Hinblick auf das kontinuierliche Kontrollsystem, das durch eine Differentialgleichung der Form

$$\dot{x}(t) = f(x(t), u(t))$$

gegeben ist, lässt sich mit diesem Einschrittverfahren durch die diskrete Funktion $f_h(x, u)$ eine zeitdiskrete approximative Lösung $\tilde{\Phi}_h$ des kontinuierlichen Kontrollsystems bestimmen. Somit wird diese Routine zu einem elementaren Baustein für unseren Algorithmus zur Lösung optimaler Steuerungsprobleme.

INPUT: An die Funktion müssen die Vektoren x, u, y sowie die Parameter d und h übergeben werden.

- Bestimme für x und u die rechte Seite der Differentialgleichung.

Rufe dazu die Funktion **rS(x,u,f)** auf.

- Berechne für jede Koordinate i :

$$y[i] = x[i] + h \cdot f[i]$$

OUTPUT: --

Die Routine: kGSV

Die Grundlage zur Berechnung des Maximums über u bildet das kontrollierte GAUSS-SEIDEL-Verfahren (3.24). Mithilfe der entsprechenden Routine `kGSV()` wird für eine Kontrolle u die rechte Seite des Optimalitätsprinzips (3.6) bzw. für $k = 0$ (3.17) berechnet. Zu berücksichtigen ist auch hier, ob der neue Punkt y im Gitter liegt (`flag=0`).

Bemerkung 4.3 i bezeichnet an dieser Stelle ebenso wie l den globalen Eckenindex.

INPUT: An die Funktion `kGSV()` müssen die Vektoren a, b, n, v, u , die Parameter $h, delta$ sowie die Variablen i und $flag$ übergeben werden.

- Setze $nenner = 1.$ und $beta = 1. - delta \cdot h.$

- Rufe folgende Routinen auf

* **Koordinaten(i, a, b, n, x)**, um die Koordinaten des x -Vektors für den globalen Eckenindex i zu bestimmen.

* **Euler(x, u, h, d, y)**, um die rechte Seite der Differentialgleichung für dieses x zu berechnen.

* **FindeRechteckKoord(y, a, b, n, koord)**. Speichere den Rückgabewert, also den Rechteckindex, in dem x liegt, in j .

- Berechne

$$wert = h \cdot g(x, u)$$

- Falls y ausserhalb des Gitters liegt ($j < 0$)

Setze $flag$ auf 1. RETURN(0.)

- Für jede bilineare Koordinate

* Berechne den globalen Eckenindex

$$l = \text{Gitter2d}(j, k, n).$$

* falls $l \neq i$ berechne

$$wert = wert + beta \cdot koord[k] \cdot v(l)$$

* ansonsten

$$nenner = nenner - beta \cdot koord[k]$$

- Bestimme den Wert für die rechte Seite des Optimalitätsprinzips nun mit

$$\text{wert} = \frac{\text{wert}}{\text{nenner}}$$

- Setze `flag` auf 0 zurück. `RETURN(wert)`

OUTPUT: 0 bzw. berechneter Wert der rechten Seite des Optimalitätsprinzips für ein $u(\text{wert})$ sowie 0 oder 1 für *flag*.

Die Routine: `gs`

Um nun den maximalen Wert (`max`) für alle Kontrollwerte u zu bestimmen, verwenden wir das GAUSS-SEIDEL-Verfahren. Die entsprechende Routine `gs()` ruft zunächst das kontrollierte GAUSS-SEIDEL-Verfahren `kGSV()` auf, welches für das übergebene u die rechte Seite des Optimalitätsprinzips berechnet. Mit der Annahme aus Abschnitt 3.3.3, dass U eine endliche Menge ist, lässt sich das Maximum nun entsprechend den Ausführungen in Kapitel 3 durch Vergleiche bestimmen. Neben dem maximalen Wert wird zusätzlich der optimale Kontrollwert (`uopt`) zurückgegeben. Der dritte Rückgabewert ist entweder eine 1, falls der neue Punkt y im Gitter liegt, ansonsten 0.

INPUT: An die Funktion `gs()` müssen die Vektoren $a, b, n, v, u, \text{max}$ und $uopt$ sowie die Variablen und Parameter i, q, h, delta übergeben werden.

- Setze `ufind` auf *false*.
- Bestimme die Koordinaten des x -Vektors für den globalen Eckenindex i durch Aufruf von **`Koordinaten(i, a, b, n, x)`**.
- Für alle u , festgelegt durch $u[k], k < q$
 - Rufe **`kGSV(a, b, n, i, v, u[k], h, delta, flag)`** auf und speichere den Rückgabewert in der Variablen `sv` ab.

- Falls `flag == 0` ist und
 - * falls bereits ein `u` gefunden wurde (`ufind == true`) und
 - falls `sv > max`
 - setze `max = sv` und `uopt = u[k]`.
 - * ansonsten setze `ufind` auf `true`, `max = sv` und `uopt = u[k]`.
 - Falls kein `u` gefunden wurde (`ufind == false`)
 - so gebe die Warnung aus:
Für jedes `u` ist `y` ausserhalb des Gitters.
RETURN(1)
 - RETURN(0)

OUTPUT: Maximum über `u` (`max`) sowie das `u` (`uopt`) das das Maximum liefert. 1 oder 0 für `flag`.

Die Routinen: `gs_iter` und `strategie_iter`

Wie bereits erwähnt, soll hier näher auf eine Kombination der GAUSS-SEIDEL-Iteration und der Strategie-Iteration aus Abschnitt 3.5 zur Berechnung des maximalen Wertes in jeder Komponente `v[i]` eingegangen werden (vergleiche dazu den zweiten Algorithmus aus Abschnitt 3.5.2). Das vollständige Programm lässt in der `main`-Funktion entweder die GAUSS-SEIDEL-Iteration (`gs_iter`) oder aber eine Kombination beider Verfahren zu (`strategie_iter`). Aufgrund der Tatsache, dass sich deren Implementierung nur in einigen wenigen Punkten unterscheidet, werden hier diejenigen Programmteile, welche sich ausschließlich auf die Strategie-Iteration beziehen, farbig unterlegt.

Vernachlässigt man den farbigen Anteil, erhält man die GAUSS-SEIDEL-Iteration.

Beide Iterationen basieren auf dem GAUSS-SEIDEL-Verfahren und werden solange fortgesetzt, bis die neuen `v[i]`'s von den alten in der Maximumnorm weniger als $h \cdot \delta \cdot \varepsilon$ abweichen. Die Strategie-Iteration wird zudem nur verwendet, wenn das optimal bestimmte `u` von dem vorherigen `u` in der 1-Norm um maximal $0.1 \cdot N$ abweicht.

Der Rückgabewert beider Routinen ist entweder 1, falls ein x existiert, so dass für jedes u der neue Punkt y außerhalb des Gitters liegt, ansonsten 0.

INPUT: An die Funktion `gs_iter` bzw. `strategie_iter` müssen die Vektoren a, b, n, v, u und die Parameter $q, h, delta$ und eps übergeben werden.

- Setze $N = (n[0] + 1) \cdot (n[1] + 1)$, $beta = 1. - delta \cdot h$ und $j = 0$.
- Setze $v[i] = 0$ für alle $i < N$, $st = 0$ oder $st = 1$.
- Solange $err > h \cdot delta \cdot eps$ führe nachfolgende Schritte aus:
 - Falls $st == 0$ erhöhe j um 1.
 - Setze err und $uerr$ auf 0.
 - Für $i < N$
 - * Falls $st == 0$
 - Falls `gs(a, b, n, i, v, u, q, h, delta, max, uopt) = 1`.
RETURN(1)
 - Falls $j > 1$
Setze $uerr = uerr + fabs(uopt[i] - uop)$.
 - Setze $uopt[i] = uop$.
 - * Ansonsten setze
 $max = kGSV(a, b, n, i, v, uopt[i], h, delta, flag)$
 - * Falls $fabs(v[i] - max) > err$
Setze $err = fabs(v[i] - max)$
 - * Setze $v[i] = max$
 - Falls $j \bmod 1 == 0$ und $st == 0$
so gebe aus:
Iteration j , Abbruchfehler err

- Falls $st == 1$ und $err < h \cdot delta \cdot eps$
Setze $st = 0$ und $err = h \cdot delta \cdot eps + 1$.
- Ansonsten, falls $st == 0$, $j > 1$ und $uerr < 0.1 \cdot N$
Setze $st = 1$.

- RETURN (0) .

OUTPUT: maximaler Wert in jeder Komponente $v[i]$
sowie 0 oder 1 für $flag$.

Die Routinen: `uoptx` und `trajopt`

Abschließend betrachten wir Routinen zur Berechnung der zeitdiskreten approximativ optimalen Trajektorien $\Phi_h(ih, x, \hat{u}_h^x)$ aus (3.20). Die Implementierung erfolgt hierbei über die Routine `trajopt ()`, welche zur Berechnung der optimalen Kontrolle u im Punkt x die Funktion `uopt ()` aufruft. Diese arbeitet ähnlich wie das GAUSS-SEIDEL-Verfahren, da auch sie das Maximum bzw. die optimalen Kontrollwerte in x über Vergleiche bestimmt. Der Wert für v_h wird dabei gemäß (3.17) berechnet. Falls kein u gefunden wird, für das die rechte Seite der Differentialgleichung bzw. an dieser Stelle y innerhalb des Gitters liegt, wird 1 zurückgegeben.

In der zweiten Routine beschreibt `x [i] [j]` eine Matrix, in welcher die Werte der Trajektorie zum Zeitpunkt ih , $i = 0, \dots, M - 1$ in der j -ten Komponente abgespeichert werden. Der Vektor `uopt` bezeichnet die optimale Steuerung zu diesen Zeitpunkten. Die Länge M der Trajektorien bestimmt sich durch die im `main`-Programm festgelegte Startzeit t , die Endzeit T und die Schrittweite h mittels $\lfloor \frac{T-t}{h} \rfloor + 1$. Für einen Punkt x wird durch den Aufruf der oben dargestellten Routine `uoptx ()` das optimale u approximativ berechnet und anschließend über das EULER-Verfahren die entsprechenden Werte der optimalen Trajektorie `y [j]` bestimmt. Zu jedem Zeitschritt ih wird dann die Matrix `x [i] [j]` mit diesen Werten belegt.

INPUT: An die Funktion **uoptx()** müssen die Vektoren x, a, b, n, v, u sowie $uopt, q, h$ und $delta$ übergeben werden.

- Setze $beta = 1. - delta \cdot h$ und $ufind$ auf *false*.
- Für $k < q$ (d.h. für jede Kontrolle)
 - Rufe **Euler(x, u[k], h, d, y)** auf.
 - Berechne die zeitdiskreten Werte für vh mit:
 - * $vh = h \cdot g(x, u[k]) + beta \cdot \mathbf{Wert}(y, a, b, n, v, \mathbf{flag})$
 - Falls $flag == 0$ und
 - * Falls bereits ein u gefunden wurde ($ufind == true$) und
 - Falls $vh > max$
 - so setze $max = vh$ und $uopt = u[k]$.
 - * Ansonsten setze $ufind$ auf *true* sowie $max = vh$ und $uopt = u[k]$.
- Falls kein u gefunden wurde ($ufind == false$)
 - so gebe die Warnung aus:
 - Für jedes u ist y ausserhalb des Gitters. RETURN(1)
- RETURN(0) .

OUTPUT: approximativ optimales u im Punkt x sowie 0 oder 1 für $flag$.

INPUT: An die Funktion `trajopt()` müssen die Vektoren x, a, b, n, v, u sowie $uopt, M, q, h$ und $delta$ übergeben werden.

- Setze $beta = 1. - delta \cdot h$.
- Für $i < M$
 - Falls $i > 0$
 - * Für $j < d$
 - Setze $x[i][j] = y[j]$.
 - Falls `uopt(x[i], a, b, n, v, u, q, uopt[i], h, delta) = 1`
 - RETURN(1) .
 - Rufe `Euler(x[i], uopt[i], h, d, y)` auf.
- RETURN(0) .

OUTPUT: optimale Trajektorie und optimale Steuerung zum Zeitpunkt $ih, i = 0, \dots, M - 1$ sowie den Wert von $flag$.

5 Diskussion der praktischen Ergebnisse

In diesem Kapitel sollen nun drei Portfolio-Entscheidungsmodelle in kontinuierlicher Zeit unter Anwendung des Algorithmus aus Kapitel 4 ausgewertet und die resultierenden Ergebnisse verglichen werden.

Unseren Betrachtungen legen wir dabei ein deterministisches Rahmenwerk zu Grunde, sodass keine der verwendeten Variablen, wie zum Beispiel die Rendite eines Vermögenswertes, einer stochastischen Dynamik folgt, also durch Unsicherheit gekennzeichnet ist. In allen drei Beispiel-Modellen wird dabei

- (1) ein Investor höchstens zwei Vermögenswerte mit entsprechender Rendite zur Auswahl haben,
- (2) sein Ziel darin bestehen, seinen Nutzen definiert über den Konsum, anstelle über das Endvermögen zu maximieren, sowie
- (3) die verwendete Zielfunktion einer Power-Nutzenfunktion, definiert über Konsum entsprechen.

Bei der Auswahl der Modelle orientieren wir uns an SEMMLER [15].

Zunächst werden im ersten Abschnitt einige weitere mathematische Grundlagen geschaffen, welche wir für die Untersuchung des dynamischen Verhaltens optimal gesteuerter Systeme benötigen sowie das Vorgehen bei der Auswertung beschrieben.

Im zweiten Abschnitt wird ein einfaches, eindimensionales Portfolio-Modell vorgestellt, wobei ein einziger Vermögenswert mit konstanter Rendite betrachtet wird. Des Weiteren werden neben der optimalen Werte- und Kontrollfunktion des Modells optimale Trajektorien für festgelegte Startparameter bestimmt und anhand von Grafiken veranschaulicht und diskutiert. Abschließend betrachten wir, wie sich Variationen in den Parametern auf

das dynamische Verhalten der optimalen Lösungen auswirken.

Die nachfolgenden Abschnitte folgen diesem Aufbau. Das im dritten Abschnitt behandelte Modell besteht nun mehr aus zwei Vermögenswerten. Um die Anzahl der konstanten Variablen möglichst klein zu halten, werden die Renditen zudem abhängig vom Vermögen periodisch dargestellt.

Der letzte Abschnitt untersucht eine Erweiterung des zweiten Beispiels auf den zeitlich periodischen Fall, um zu zeigen, wie sich diverse Änderungen in der Modellierung der Rendite auf das dynamische Verhalten des Systems auswirken.

5.1 Analyse optimaler Steuerungsprobleme

5.1.1 Die Dynamik des optimal gesteuerten Systems

Bei der Analyse optimaler Kontrollprobleme kommt der Dynamik des optimal gesteuerten Systems besondere Bedeutung zu. Um im Folgenden das Langzeitverhalten der Lösungen besser untersuchen zu können, müssen zunächst einige weitere Grundlagen geschaffen werden.

Besonders interessant sind an dieser Stelle die Punkte, gegen die die optimalen Lösungen nach einer gewissen Zeit konvergieren. Man bezeichnet derartige Punkte als Ruhelage oder als Gleichgewicht.

Um mathematisch definieren zu können, was unter einem Gleichgewicht zu verstehen ist, benötigen wir zunächst zwei Klassen von Vergleichsfunktionen:

Definition 5.1 *Wir definieren die folgenden Klassen von Funktionen*

$$\mathcal{K}_\infty := \left\{ \alpha : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+ \left| \begin{array}{l} \alpha \text{ ist stetig, streng monoton wachsend,} \\ \text{unbeschränkt und erfüllt } \alpha(0) = 0 \end{array} \right. \right\}$$

und

$$\mathcal{KL} := \left\{ \beta : \mathbb{R}_0^+ \times \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+ \left| \begin{array}{l} \beta \text{ ist stetig, } \beta(\cdot, t) \in \mathcal{K}_\infty \text{ für jedes } t \geq 0 \\ \text{und } \beta(r, t) \text{ ist streng monoton fallend in } t \\ \text{mit } \lim_{t \rightarrow \infty} \beta(r, t) = 0 \text{ für jedes } r > 0 \end{array} \right. \right\}.$$

Ferner müssen für eine Stabilitätsanalyse einige Begriffe aus der Stabilitätstheorie für unkontrollierte dynamische Systeme auf optimale Steuerungsprobleme übertragen werden. Dazu fassen wir optimale Steuerungsprobleme als dynamische Systeme auf, indem wir annehmen, dass zu jedem Anfangswert eine optimale Kontrollfunktion $u^x \in \mathcal{U}$ existiert. Dann gilt

$$v(x) = J(x, u^x).$$

Die zugehörigen Lösungen bezeichnen wir mit $\Phi^*(t, x) = \Phi(t, x, u^x)$.

Die Annahme, dass eine optimale Kontrollfunktion existiert, ist in vielen praktischen Problemen erfüllt und kann rigoros bewiesen werden. Die Eindeutigkeit dieser Funktion soll an dieser Stelle allerdings nicht vorausgesetzt werden, so dass nicht angenommen werden kann, dass die Lösung Φ^* auch eindeutig definiert ist. Es bleibt somit die Möglichkeit bestehen, dass es zu einem Anfangswert unter Umständen mehrere Lösungen geben kann. Daher werden wir im Folgenden stets von einer optimalen Lösung bzw. von den Lösungen $\Phi^*(t, x)$ mit Anfangswert x sprechen.

Das optimale Gleichgewicht und dessen Eigenschaften lassen sich nun wie folgt definieren:

Definition 5.2 (i) Einen Punkt $x^* \in \mathbb{R}^d$ nennt man *optimales Gleichgewicht*, falls eine optimale Lösung $\Phi^*(t, x^*)$ mit $\Phi^*(t, x^*) = x^*$ für alle $t \geq 0$ bzw. $t \in h\mathbb{N}_0$ existiert.

(ii) Ein optimales Gleichgewicht x^* heißt *stabil*, falls eine Umgebung $\mathcal{N}(x^*)$ und eine \mathcal{K}_∞ Funktion α existiert, so dass für alle in $\mathcal{N}(x^*)$ startende Lösungen $\Phi^*(t, x)$ die Ungleichung

$$\|\Phi^*(t, x) - x^*\| \leq \alpha(\|x - x^*\|)$$

für alle $t \geq 0$ bzw. $t \in h\mathbb{N}_0$ gilt.

(iii) Ein optimales Gleichgewicht x^* heißt asymptotisch stabil, falls eine Umgebung $\mathcal{N}(x^*)$ und eine \mathcal{KL} Funktion β existiert, so dass für alle in $\mathcal{N}(x^*)$ startende Lösungen $\Phi^*(t, x)$ die Ungleichung

$$\|\Phi^*(t, x) - x^*\| \leq \beta(\|x - x^*\|, t)$$

für alle $t \geq 0$ bzw. $t \in h\mathbb{N}_0$ erfüllt ist. Falls $\mathcal{N}(x^*) = \mathbb{R}^d$ ist, so heißt x^* global asymptotisch stabil.

(iv) Ein optimales Gleichgewicht x^* heißt instabil, falls (ii) nicht gilt, also falls für jede Umgebung $\mathcal{N}(x^*)$ und jede \mathcal{K}_∞ Funktion α mindestens eine Lösung $\Phi^*(t, x)$, die in dieser Umgebung startet, und ein $t \geq 0$ existiert mit

$$\|\Phi^*(t, x) - x^*\| > \alpha(\|x - x^*\|).$$

Falls mehrere asymptotisch stabile Gleichgewichte existieren, ist zudem interessant, für welche Anfangswerte die Lösungen gegen welches Gleichgewicht konvergieren. Formal definiert man dazu den Einzugsbereich:

Definition 5.3 Für ein Gleichgewicht x^* definieren wir den Einzugsbereich (bzw. die Stabilitätsumgebung) als

$$\mathcal{D}(x^*) := \left\{ x \in \mathbb{R}^d \mid \Phi^*(t, x) \rightarrow x^* \text{ für } t \rightarrow \infty \right\}.$$

Numerisch können optimale Gleichgewichte durch die approximative Berechnung optimaler Trajektorien bestimmt werden. Durch eine gezielte Simulation kann man damit die Gleichgewichte sowie ihre Einzugsbereiche bestimmen. Eine weitere, effizientere Möglichkeit besteht darin, die approximativ optimale Feedbackabbildung $\hat{u}^*(x)$ aus Definition 3.27 zu verwenden, um die Dynamik des optimal gesteuerten Systems zu analysieren, da diese direkt als Funktion grafisch dargestellt werden kann. Um einen globalen Überblick über das Verhalten der Trajektorien des optimal gesteuerten Systems zu erhalten, kann die Abbildung $f(x, u^*(x))$ zum Beispiel in Vektorfeldform mit Pfeilen grafisch dargestellt werden.

Im Folgenden werden wir diese Darstellungsmöglichkeiten kombinieren bzw. variieren, um einen möglichst umfassenden Überblick über das Langzeitverhalten der Lösungen unserer Beispiel-Modelle zu erhalten.

5.1.2 Vorgehen bei der Auswertung

Die folgenden Modell-Beispiele werden unter Anwendung des Algorithmus ² aus dem vorherigen Kapitel numerisch gelöst. Für die Berechnung der benötigten Werte müssen zunächst, wie in Kapitel 4 erwähnt, die entsprechenden rechten Seiten der Differentialgleichungen in der Routine `rs()` bzw. die Zielfunktion des Modells in `g()` angepasst werden. Weiterhin sind direkt im Algorithmus einige Parameter festzulegen bzw. Argumente an das Programm zu übergeben. Direkt im Algorithmus muss u.a.

- die Rendite r des Vermögenswertes (Routine: `rs()`),
- die Risikoaversion γ des Investors (Routine: `g()`),
- die Anzahl der Unterteilungen des Gitters (`main`-Funktion),
- das Intervall Ω , auf welchem die Berechnungen durchgeführt werden sollen (`main`-Funktion),
- die Schrittweite h (`main`-Funktion), sowie
- das Intervall $[t, T]$ zur Berechnung optimaler Trajektorien (`main`-Funktion)

angegeben werden.

Zum Ausführen des Programms sind mindestens drei Argumente zu übergeben:

der Startwert x , d.h in unserem Fall x_1 und x_2 , sowie die Diskontrate δ für das Modell. Auf diese Weise werden sowohl die optimale Werte- und Kontrollfunktion, die optimalen Trajektorien zum Startwert als auch das Vektorfeld berechnet. Wahlweise kann als viertes Argument der Name einer Datei übergeben werden, in der bereits im Vorfeld berechnete Werte der optimalen Wertefunktion abgespeichert wurden. Somit kann bei der

²Das vollständige Programm befindet sich im Anhang C.

Berechnung optimaler Trajektorien zu weiteren Startwerten Rechenzeit eingespart werden.

Die berechneten Werte werden in der `main`-Funktion des C++-Programms zeilen- und spaltenweise in `asc`-Dateien gespeichert. Hierbei ergibt sich für die optimale Werte- und Kontrollfunktion die Spaltenanzahl der Matrix aus der Anzahl der Zustände und aus den errechneten Werten für $\hat{v}_h(x)$ bzw. $\hat{u}^*(x)$. Die Anzahl der Zeilen richtet sich nach der Anzahl der Knoten des Gitters und ist somit von der Schrittweite h abhängig.

Die optimalen Lösungstrajektorien $x_1(ih)$, $x_2(ih)$ und $\hat{u}_h^x(ih)$, $i = 0, 1, 2, \dots$ werden gegen die Zeit t abgetragen, so dass sich Matrizen mit 4 Spalten ergeben. Die Anzahl der Zeilen wird hier durch die Größe des Intervalls $[t, T]$ und die Schrittweite h festgelegt. Alternativ können die Werte von x_1 im sogenannten Phasendiagramm gegen die von x_2 abgetragen werden.

Um das globale Verhalten der optimalen Lösungen darzustellen, kann das entsprechende Vektorfeld $f(x, \hat{u}^*(x))$ dargestellt werden. Zur Berechnung der dazu notwendigen Koordinaten x_1 , x_2 bzw. der rechten Seite der Differentialgleichung y_1 , y_2 genügt es, nur jeden `N0`-ten Wert in x_1 -Richtung und jeden `N1`-ten Wert in x_2 -Richtung auszuwerten. Auf diese Weise ergeben sich Matrizen der Größe $4 \times (\text{N0} \cdot \text{N1})$.

Um die Ergebnisse sinnvoll diskutieren und vergleichen zu können, wurden zudem die MATLAB-Routinen ³

- `zeichnen_wertef.m`,
- `zeichnen_uopt.m`,
- `zeichnen_traj.m` und
- `zeichnen_feld.m`

geschrieben. Ziel dieser Routinen ist es, die entsprechenden Werte aus den `asc`-Dateien einzulesen und zu plotten, um so eine grafische Veranschaulichung der Ergebnisse zu erhalten.

³Die zur Auswertung verwendeten MATLAB-Routinen befinden sich im Anhang D.

Da der Zustand x in den ersten beiden Beispielen eindimensional ist, wird x_2 bzw. y_2 im Programm auf 0 gesetzt. Diese Werte ändern sich folglich nicht, sodass anstatt einer dreidimensionalen eine zweidimensionale Darstellung der Ergebnisse genügt. Dazu wird in der jeweiligen MATLAB-Routine einfach diejenige Spalte, die die Werte für x_2 bzw. y_2 enthält, beim Einlesen weggelassen.

5.2 Modell 1: Ein Vermögenswert - Konstante risikolose Rendite

5.2.1 Einführung

Das erste Portfolio-Modell besteht zunächst aus nur einem einzigen Vermögenswert, welcher eine konstante - daher risikolose - Rendite liefert. Ein derartiger Wert wäre zum Beispiel ein Bond⁴, das heißt, ein risikoloses Wertpapier.

Bemerkung 5.4 *Der an dieser Stelle betrachtete Vermögenswert wird eher einem Sparguthaben als einem Bond entsprechen, aus historischen Gründen aber weiterhin so bezeichnet.*

Zudem kann ein Investor entscheiden, wie viel seines Vermögens er wann konsumieren möchte. Die Kontrollvariable u modelliert somit die Höhe des Konsums eines Anlegers. Das Modell ist also zum einen eindimensional im Zustand x und besitzt zum anderen eine eindimensionale Kontrolle u . Die Dynamik des Vermögens ist durch

$$\dot{x}(t) = x(t) \cdot r - u(t)$$

gegeben. Die Vermögensdifferenz zur nächsten Periode bestimmt sich demnach aus dem verzinnten Vermögen, vermindert um den Konsum in der aktuellen Periode.

Ziel des optimalen Steuerungsproblems ist es nun, den Nutzen, definiert über den Konsum des Investors, zu maximieren. Wie bereits erwähnt, wird dieser durch die Power-Nutzenfunktion

$$g(u) = \frac{u^{1-\gamma}}{1-\gamma} \quad (5.1)$$

festlegt.

⁴In Anhang B befindet sich eine Tabelle mit im Folgenden verwendeten finanzmathematischen Begriffen.

Bemerkung 5.5 Da die Nutzenfunktion (5.1) für $\gamma = 1$ nicht definiert ist, müssen wir dies im Programm berücksichtigen. Wie in Kapitel 2 beschrieben, wird für $\gamma = 1$ die logarithmische Funktion

$$g(u) = \ln(u)$$

verwendet.

Ein Anleger muss in unserem ersten Beispiel also zunächst keine Entscheidung zwischen unterschiedlichen Assets, sondern nur hinsichtlich des optimalen Verbrauchs bzw. Konsums treffen. Ziel der optimalen Steuerung ist es nun, die Kontrollfunktion $u \in \mathcal{U}$ so zu wählen, dass das diskontierte Funktional auf unendlichem Zeithorizont $J(x, u)$ seinen optimalen Wert annimmt.

Das optimale Steuerungsproblem besteht nun darin, die optimale Wertefunktion

$$\begin{aligned} v(x) &= \max_{u \in \mathcal{U}} J(x, u) \\ &= \max_{u \in \mathcal{U}} \int_0^{\infty} e^{-\delta t} \frac{u^{1-\gamma}}{1-\gamma} dt \end{aligned} \tag{5.2}$$

Wir werden im Folgenden das Portfolio-Problem zunächst entsprechend SEMMLER [15] mit Hilfe der HAMILTON-JACOBI-BELLMAN- Gleichung (3.11) lösen. Die Ergebnisse sollen anschließend zur Analyse der numerisch ermittelten Resultate herangezogen werden, welche wir unter Anwendung des Algorithmus aus Kapitel 4 erhalten.

5.2.2 Lösung des Problems mithilfe der HJB-Gleichung

Die Lösung des Portfolio-Optimierungsproblems mit Hilfe der HAMILTON-JACOBI-BELLMAN-Gleichung erfolgt über zwei Schritte: Im ersten Schritt sucht man die optimale Portfolio-Strategie als Funktion der unbekanntenen Wertefunktion. Anschließend wird diese Strategie in die HJB-Gleichung eingesetzt. Die Lösung dieser partiellen Differentialgleichung stellt den zweiten Schritt dar. Allerdings sind explizite Lösungen der HJB-Gleichungen nur in Spezialfällen möglich, zumeist muss man die Form der Wertefunktion bereits hinreichend gut kennen.

Die HJB-Gleichung für das Modell ist nach SEMMLER [15] gegeben durch

$$-v_t(x) = \max_u \{e^{-\delta t} g(u) + v_x(x)(r \cdot x - u)\} \quad (5.3)$$

Hierbei bezeichne v_t bzw. v_x die partielle Ableitung von v nach t bzw. x .

Die Bedingung erster Ordnung für (5.3) ist dann

$$e^{-\delta t} g'(u) - v_x(x) = 0 \quad (5.4)$$

Unter Verwendung der Power-Nutzenfunktion (5.1) erhalten wir

$$v_x(x) = e^{-\delta t} u^{-\gamma}$$

und somit, aufgelöst nach der Kontrolle u

$$u = (v_x e^{\delta t})^{-\frac{1}{\gamma}} \quad (5.5)$$

Setzen wir nun u in die HJB-Gleichung (5.3), so erhalten wir

$$\begin{aligned} 0 &= +v_t + e^{-\delta t} \frac{1}{1-\gamma} (v_x e^{\delta t})^{-\frac{1}{\gamma}(1-\gamma)} + v_x (r \cdot x - (v_x e^{\delta t})^{-\frac{1}{\gamma}}) \\ &= v_t + v_x r x + \frac{1}{1-\gamma} e^{\delta t(1-\frac{1}{\gamma}-1)} v_x^{1-\frac{1}{\gamma}} - v_x^{1-\frac{1}{\gamma}} e^{-\frac{\delta t}{\gamma}} \\ &= v_t + v_x r x + \frac{\gamma}{1-\gamma} (e^{-\frac{\delta t}{\gamma}} v_x^{1-\frac{1}{\gamma}}) \end{aligned} \quad (5.6)$$

Zudem benötigen wir eine Schätzung der Wertefunktion, wobei diese entsprechend SEMMLER [15] durch

$$v(x) = R(t) e^{-\delta t} g(x) = R e^{-\delta t} \frac{x^{1-\gamma}}{1-\gamma}$$

gegeben ist.

Dann erhalten wir für die partiellen Ableitungen v_t bzw. v_x

$$v_t = -\delta v \quad (5.7)$$

$$v_x = \frac{1-\gamma}{x} v \quad (5.8)$$

Für den zweiten Faktor der Multiplikation aus (5.6) ergibt sich damit

$$\begin{aligned} e^{-\frac{\delta t}{\gamma}} v_x^{\frac{\gamma-1}{\gamma}} &= e^{-\frac{\delta t}{\gamma}} (R e^{-\delta t} x^{-\gamma})^{\frac{\gamma-1}{\gamma}} \\ &= R^{\frac{\gamma-1}{\gamma}} e^{-\delta t} x^{1-\gamma} \\ &= (1-\gamma) R^{-\frac{1}{\gamma}} v \end{aligned} \quad (5.9)$$

Um nun zu überprüfen, ob die oben eingeführte Lösung die Lösung unseres Problems ist, setzen wir (5.7), (5.8) und (5.9) in (5.6) ein. Dann erhalten wir

$$\begin{aligned} &-\delta v + r(1-\gamma)v + \gamma R^{-\frac{1}{\gamma}} v \\ &= v (\gamma R^{-\frac{1}{\gamma}} + r(1-\gamma) - \delta) \\ &= 0 \end{aligned} \quad (5.10)$$

Erfüllt R nun (5.10), gilt

$$R = \left(\frac{\delta}{\gamma} + \frac{r(\gamma-1)}{\gamma} \right)^{-\gamma} \quad (5.11)$$

Als Lösung des Problems erhält man damit

$$v(x) = \left(\frac{\delta}{\gamma} + \frac{r(\gamma-1)}{\gamma} \right)^{-\gamma} e^{-\delta t} \frac{x^{1-\gamma}}{1-\gamma} \quad (5.12)$$

Unter Verwendung der Kontrolle u aus (5.5) ergibt sich für die optimale Kontrolle

$$\begin{aligned}
 u_{opt} &= (v_x e^{\delta t})^{-\frac{1}{\gamma}} \\
 &= R^{-\frac{1}{\gamma}} x \\
 &= \left[\left(\frac{\delta}{\gamma} + \frac{r(\gamma - 1)}{\gamma} \right)^{-\gamma} \right]^{-\frac{1}{\gamma}} \cdot x \\
 &= \left(\frac{\delta}{\gamma} + \frac{r(\gamma - 1)}{\gamma} \right) \cdot x
 \end{aligned} \tag{5.13}$$

Für die Analyse der numerischen Resultate interessiert insbesondere das Verhältnis der optimalen Kontrolle zum Vermögen. Dieses wird als Konsumneigung eines Investors bezeichnet und ist gegeben durch

$$\frac{u_{opt}}{x} = \frac{\delta}{\gamma} + \frac{r(\gamma - 1)}{\gamma}. \tag{5.14}$$

Da das erste Portfolio-Problem eine feste, das heißt nicht zeitvariante Rendite r aufweist und sich neben der Diskontrate δ des Modells auch die Risikoaversion γ eines Investors nicht ändert, ist die Konsumneigung eines Anlegers zu jedem Zeitpunkt konstant.

Im Hinblick auf spätere Variationen in den Parametern zeigt sich, dass die Konsumneigung (5.14) einen umso kleineren (konstanten) Wert aufweist, je höher die Rendite des Vermögenswertes ist. Das gleiche Verhalten erhält man für eine stärkere Risikoaversion des Investors. Wählen wir einen größeren Wert für die Diskontrate in (5.2), so nimmt die Konsumneigung des Investors hingegen zu.

Anhand dieser Feststellungen sollen nun im nächsten Abschnitt die numerisch approximierten Ergebnisse analysiert und verglichen werden.

5.2.3 Diskussion der numerischen Ergebnisse

Um die Abänderungen im Programm möglichst klein zu halten, werden die Routinen für $d = 2$ verwendet. Für eine korrekte Berechnung müssen wir Ω daher auch in der zweiten Komponente festlegen. Im Folgenden wird als Intervall ⁵ $\Omega = [0, 1] \times [0, 1]$ mit einer Diskretisierung von $P_1 = P_2 = 100$ gewählt, so dass zur Bestimmung von Werte- und Kontrollfunktion 101×101 Gitterpunkte auszuwerten sind. Um Probleme bei der Darstellung der Ergebnisse zu vermeiden, welche sich aufgrund der Unstetigkeitsstelle in der Null ergeben, wählen wir für $\gamma \geq 1$ das Intervall $\Omega = [0.0001, 1] \times [0, 1]$. Dies muss explizit im Programm angegeben werden.

Als Schrittweite h wird $h = \frac{1}{12}$ verwendet. Für den Plot des Phasendiagramms von $x = (x_1, x_2)$ und die einzelnen Trajektorien gegen die Zeit ergeben sich somit bei einer Gesamtlaufzeit von $T = 180$ Matrizen mit 4 Spalten und 2161 Zeilen. Zur Darstellung des Vektorfeldes werden wir in x_1 -Richtung nur jeden vierten Wert ($N0=4$) in x_2 -Richtung jeden zehnten Wert ($N1=10$) berechnen. Auf diese Weise ergeben sich Matrizen der Größe 4×286 .

Ferner verwenden wir für unsere Berechnungen eine Rendite von $r = 3\%$, die Risikoaversion $\gamma = 0.75$ sowie die Diskontrate $\delta = 0.06$. Für die mit dem Vermögen skalierte Kontrolle $\tilde{u} := \frac{u}{x}$ wählen wir den Kontrollwertebereich $\tilde{\mathbf{U}} = [0, 7]$, diskretisiert mit $q = 4001$ äquidistanten Kontrollwerten. Somit liegt auch u im Bereich $\mathbf{U} = [0, 7]$ für $x \in [0, 1]$. Tabelle 5.1 fasst die verwendeten Parameter noch einmal überblickartig zusammen.

⁵Die Wahl $\Omega = [0, 1] \times [0, 0]$ würde zu Fehlern führen.

Parameter	Verwendeter Wert
Ω	$[0, 1] \times [0, 1]$
$(n[0], n[1])$	$(100, 100)$
h	$\frac{1}{12}$
$[t, T]$	$[0, 180]$
$N0$	4
$N1$	10
r	3%
γ	0.75
δ	0.06

Tabelle 5.1: Parameterwerte zur Auswertung von Modell 1.

Abbildung 5.1 zeigt einen Ausschnitt der numerisch berechneten optimalen Wertfunktion des Problems. Entsprechend der verwendeten Nutzenfunktion weist sie ein konkaves Krümmungsverhalten auf.

Die zugehörige optimale Kontrollfunktion aus Abbildung 5.2 verläuft linear ansteigend. Um das Konsumverhalten eines Anlegers an dieser Stelle besser auswerten zu können, sollen die theoretischen Resultate aus Abschnitt 5.2.2 herangezogen werden. Wir betrachten daher auch an dieser Stelle das Verhältnis des optimalen Konsums zum Vermögen, das heißt die Konsumneigung des Investors. Unter Verwendung von (5.14) ergibt sich somit zusammen mit den Werten aus Tabelle 5.1 eine konstante Konsumneigung von 0.07. Numerisch erhalten wir die in der rechten Abbildung dargestellte, mit dem Vermögen skalierte optimale Kontrollfunktion. Bis auf Abweichungen nahe der Null lässt sich auch hier ein konstanter Verlauf der Konsumneigung erkennen. Die Abweichungen nahe der Null können auf die Unstetigkeitsstelle in diesem Punkt zurückgeführt werden.

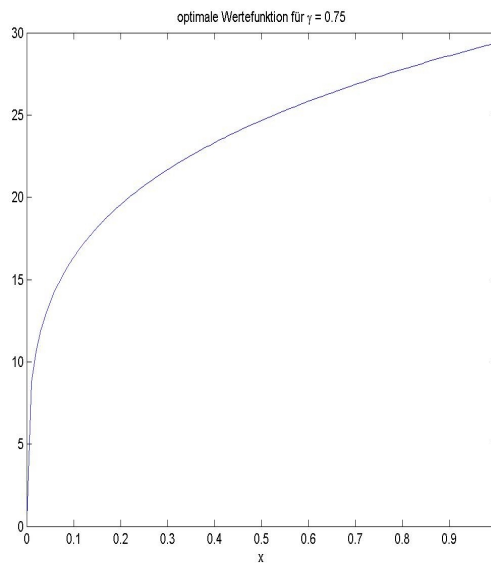


Abbildung 5.1: Die Grafik zeigt die approximativ optimale Wertefunktion des Modells auf dem Intervall $\Omega = [0, 1]$.

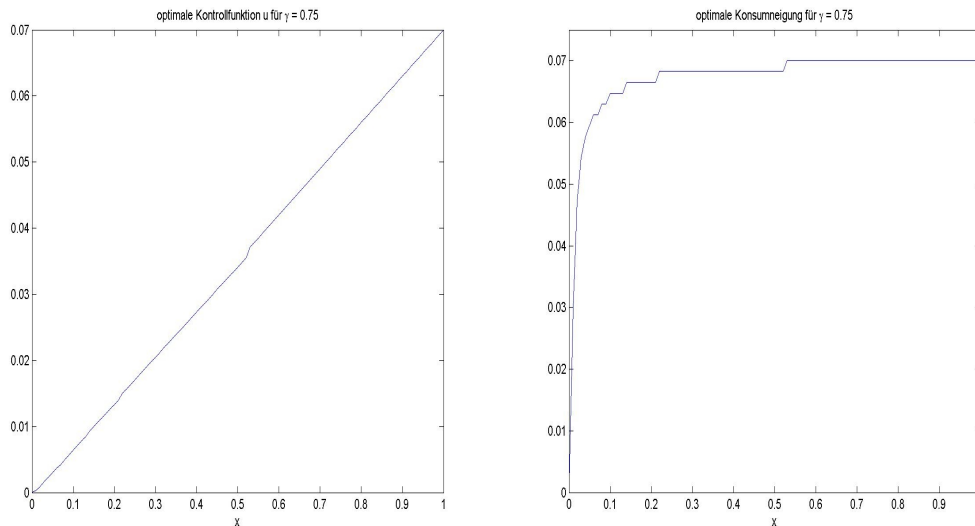


Abbildung 5.2: Die Grafik zeigt links die approximativ optimale Kontrollfunktion und rechts die zugehörige Konsumneigung eines Investors auf dem Intervall $\Omega = [0, 1]$.

Um das dynamische Verhalten des Systems analysieren zu können, soll es an dieser Stelle genügen, mehrere optimale Trajektorien gemeinsam darzustellen. Damit der Verlauf der Lösungskurven sinnvoll diskutiert werden kann, muss ein sehr großer und daher unrealistisch erscheinender Zeitraum gewählt werden. Auf diese Weise ergibt sich bei einer Schrittweite $h = \frac{1}{12}$ die im Programm festgesetzte Endzeit von $T = 180$ Jahren. Zur Erklärung dieser Wahl ist anzumerken, dass es sich bei dem Intervall $[0, T]$ nicht um die eigentliche Laufzeit der Anlage, sondern um den Betrachtungszeitraum der optimalen Lösungen handelt.

In Abbildung 5.3 stellt man fest, dass alle Lösungskurven gegen das Gleichgewicht $x^* = 0$ streben. Tatsächlich ist dieser Punkt asymptotisch stabil für das optimal gesteuerte System. Für das Vermögen des Investors gilt: Je niedriger das Startkapital des Anlegers, desto schneller erreicht die Lösungskurve die Ruhelage.

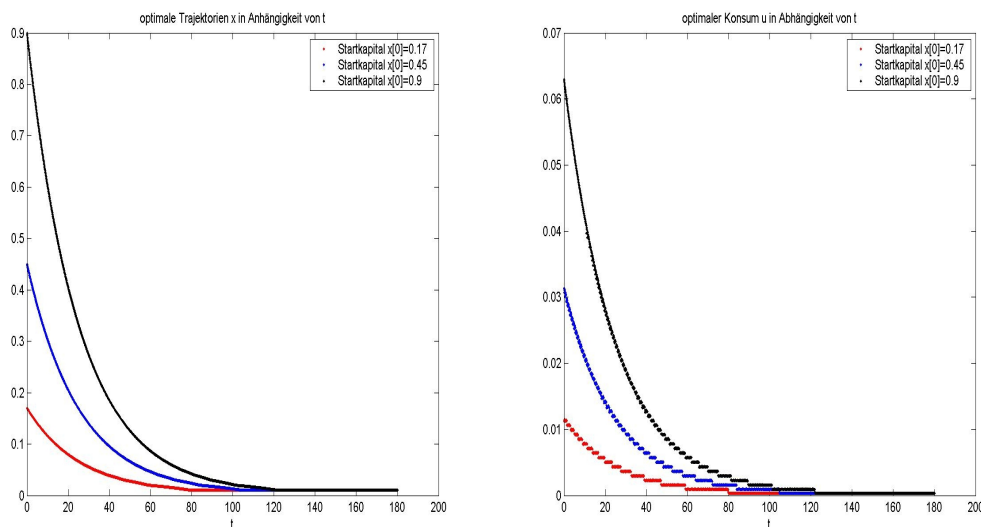


Abbildung 5.3: Die Grafik zeigt den Verlauf optimaler Trajektorien für unterschiedliches Startkapital im Zeitraum $[t, T] = [0, 180]$.

Ein vergleichbares Verhalten lässt sich auch beim Verlauf des optimalen Konsums feststellen. Zunächst richtet sich dieser nach dem eingesetzten Kapital, d.h. je höher das Anfangsvermögen, desto mehr wird ein Investor zu Beginn konsumieren. Außerdem zeigt sich auch hier, dass das optimale Gleichgewicht im Punkt $x^* = 0$ liegt und abhängig vom Startkapital zu den gleichen Zeitpunkten erreicht wird wie beim Vermögen.

Häufig ist man weniger an den Gleichgewichten und deren Stabilitätseigenschaften für ein einzelnes System interessiert, sondern daran, wie sich diese Eigenschaften in Abhängigkeit von einem Parameter verhalten. Wenn sich die Anzahl der Gleichgewichte hierbei verändert, spricht man von einer Bifurkation bzw. Verzweigung.

Die bisher verwendeten Parameter bzw. Grafiken bilden Ausgangspunkt für den nächsten Abschnitt. In diesem werden derartige Variationen in den Parametern vorgenommen und sich dadurch ergebende Auswirkungen auf die Dynamik des optimal gesteuerten Systems betrachtet.

5.2.4 Variationen in den Parametern

Variation der Risikoaversion

Zunächst betrachten wir, wie sich unterschiedliche Risikoaversionen auf die Dynamik des Systems auswirken. Wie in Kapitel 2 erwähnt, ist die Power-Nutzenfunktion (5.1) umso stärker gekrümmt, je größer der Wert für γ ist, das heißt je stärker der Investor einem Risiko abgeneigt ist. An dieser Stelle werden bis zu vier unterschiedlich risikoscheue Anleger betrachtet, welche über ein Anfangskapital in gleicher Höhe verfügen. Investor A sei im Folgenden am wenigsten einem Risiko abgeneigt. Anleger B sei risikoscheuer als Investor A, allerdings wiederum weniger risikoavers als Entscheider C bzw. D, welcher die größte Aversion aufweist. Tabelle 5.2 fasst die analysierten Risikoaversionen der Investoren nochmals zusammen.

Investor	Risikoaversion
A	$\gamma = 0.1$
B	$\gamma = 0.5$
C	$\gamma = 0.75$
D	$\gamma = 5$

Tabelle 5.2: Die verwendeten Werte für die Risikoaversion eines Investors.

Betrachtet man die approximativ optimalen Wertefunktionen aus Abbildung 5.4, so zeigt sich, dass der akkumulierte Nutzen des Anlegers bei optimaler Steuerung in Abhängigkeit vom Vermögen umso kleiner ist, je geringer die Aversion des Investors ist.

Interessant sind auch die optimalen Kontrollfunktionen bzw. die entsprechende Konsumneigung der Investoren, welche in Abbildung 5.5 grafisch veranschaulicht werden. Ein eher risikoneutraler Investor konsumiert im Verhältnis zu seinem Vermögen mehr als ein Anleger, der einem Risiko kritisch gegenüber steht. Die Konsumneigung \tilde{u} ist demnach geringer, je höher der Wert für γ ist. Mit anderen Worten wird es für einen Investor umso reizvoller, das Vermögen anzulegen anstatt zu konsumieren, um seinen Nutzen zu maximieren, je höher dessen Risikoaversion ist. Diese Feststellung lässt sich zudem mit Hilfe der Resultate, welche wir unter Verwendung der HJB-Gleichung aus Abschnitt 5.2.2 folgern konnten, bestätigen. Der auffallend abweichende Verlauf beider Funktionen bei einer Risikoaversion von $\gamma = 5$ kann an dieser Stelle auf die Unstetigkeitsstelle im Nullpunkt zurückgeführt werden.

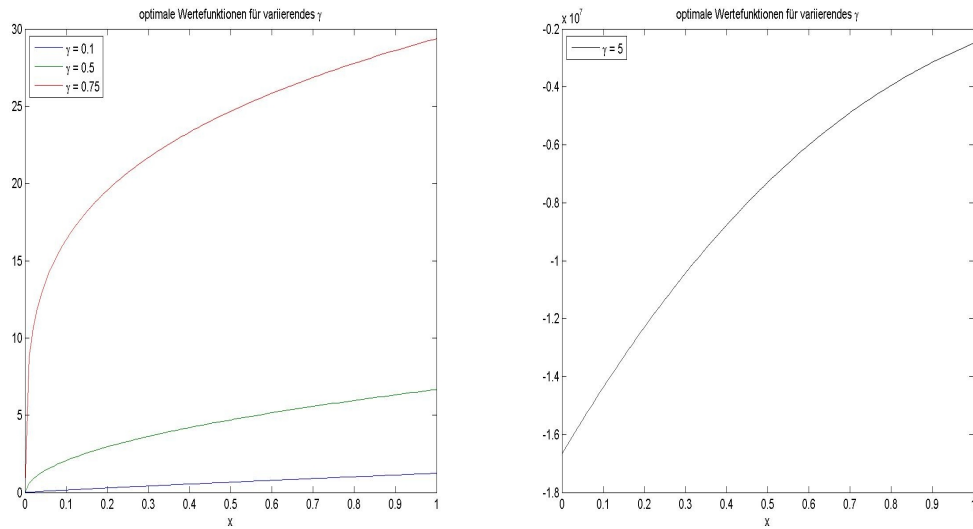


Abbildung 5.4: Die Grafik zeigt die approximativ optimalen Wertefunktionen für die unterschiedlichen Risikoaversionen zur Diskontrate $\delta = 0.06$.

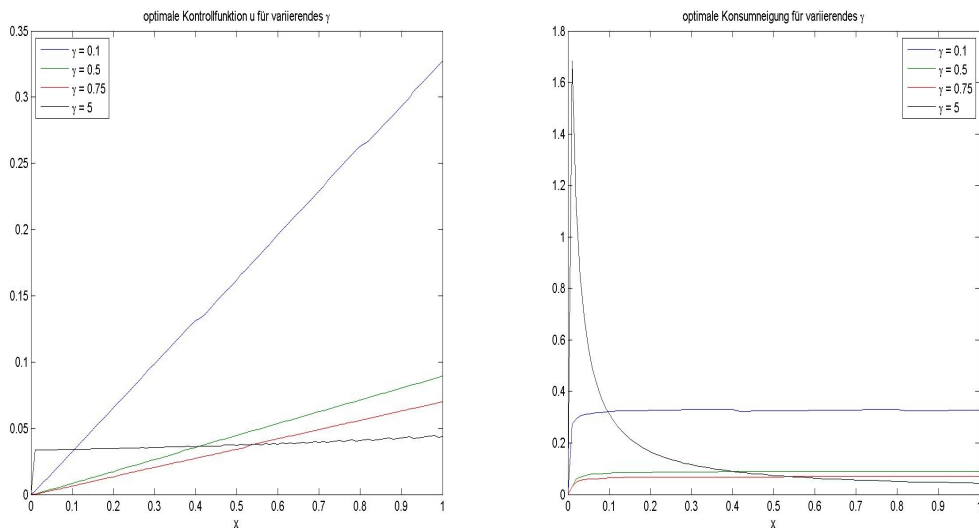


Abbildung 5.5: Die Grafik zeigt im linken Bild die optimale Kontrollfunktion, im rechten Bild die zugehörige Konsumneigung für $\gamma = 0.1, 0.5, 0.75$ und 5 zur Diskontrate $\delta = 0.06$.

Um nun mögliche Veränderungen auf die Dynamik des Systems betrachten zu können, soll zum einen ein globaler Überblick mit Hilfe des Vektorfeldes, zum anderen auch hier eine gemeinsame Darstellung einzelner optimaler Trajektorien für die zu untersuchenden Risikoaversionen gegeben werden.

Mithilfe der Vektordarstellung $f(x, u^*(x))$ aus Abbildung 5.6 erkennt man, dass die Lösungen des optimal gesteuerten Systems unabhängig von der Risikoabneigung gegen das bisherige Gleichgewicht $x^* = 0$ streben.

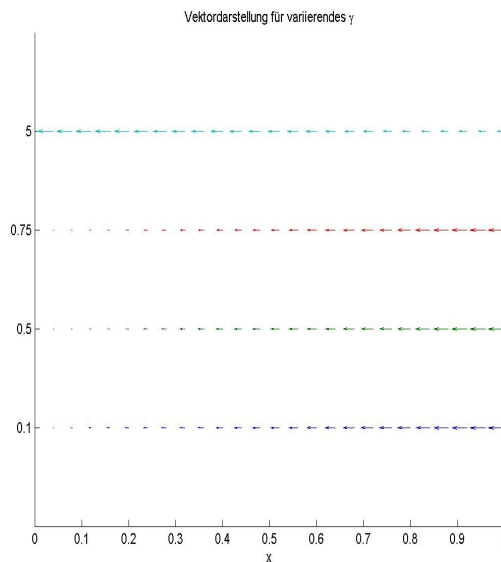


Abbildung 5.6: Die Grafik zeigt die Vektordarstellung für $\gamma = 0.1, 0.5, 0.75$ und 5 zur Diskontrate $\delta = 0.06$.

Vor diesem Resultat sollen nun auch hier die optimalen Trajektorien für den festen Startwert $x_1 = 0.9$ in Abhängigkeit von der Zeit betrachtet werden.

Neigt ein Investor dazu, umso weniger seines Vermögen zu konsumieren, je stärker er einem Risiko abgeneigt ist, müsste Anleger A zunächst einen höheren Anfangskonsum aufweisen als Investor B, C oder D. Abbildung 5.7 bestätigt dieses Verhalten. Weiterhin wird aus dieser Darstellung ersichtlich, dass die optimalen Lösungen bei Investor A schneller gegen das optimale Gleichgewicht streben als bei den übrigen Anlegern. Be-

trachtet man allerdings die optimalen Lösungskurven für $\gamma = 5$, weist der Verlauf des Vermögens zum einen ein abweichendes Krümmungsverhalten auf, zum anderen wird die Ruhelage hier erneut zu einem früheren Zeitpunkt erreicht. Der Verlauf des optimalen Konsums bleibt in diesem Fall bis zu einer gewissen Zeit nahezu unverändert, bis er schließlich zeitgleich mit dem Vermögen in die Ruhelage „springt“. Untersucht man weitere Werte für γ in diesem Bereich, so zeigt sich, dass sich diese Veränderung ungefähr ab einer Risikoaversion zwischen 3 und 4 einstellt. Dieses Verhalten ist allerdings eher numerisch bedingt als auf die gewählte Risikoaversion zurückzuführen.

5.2 MODELL 1: EIN VERMÖGENSWERT - KONSTANTE RISIKOLOSE RENDITE

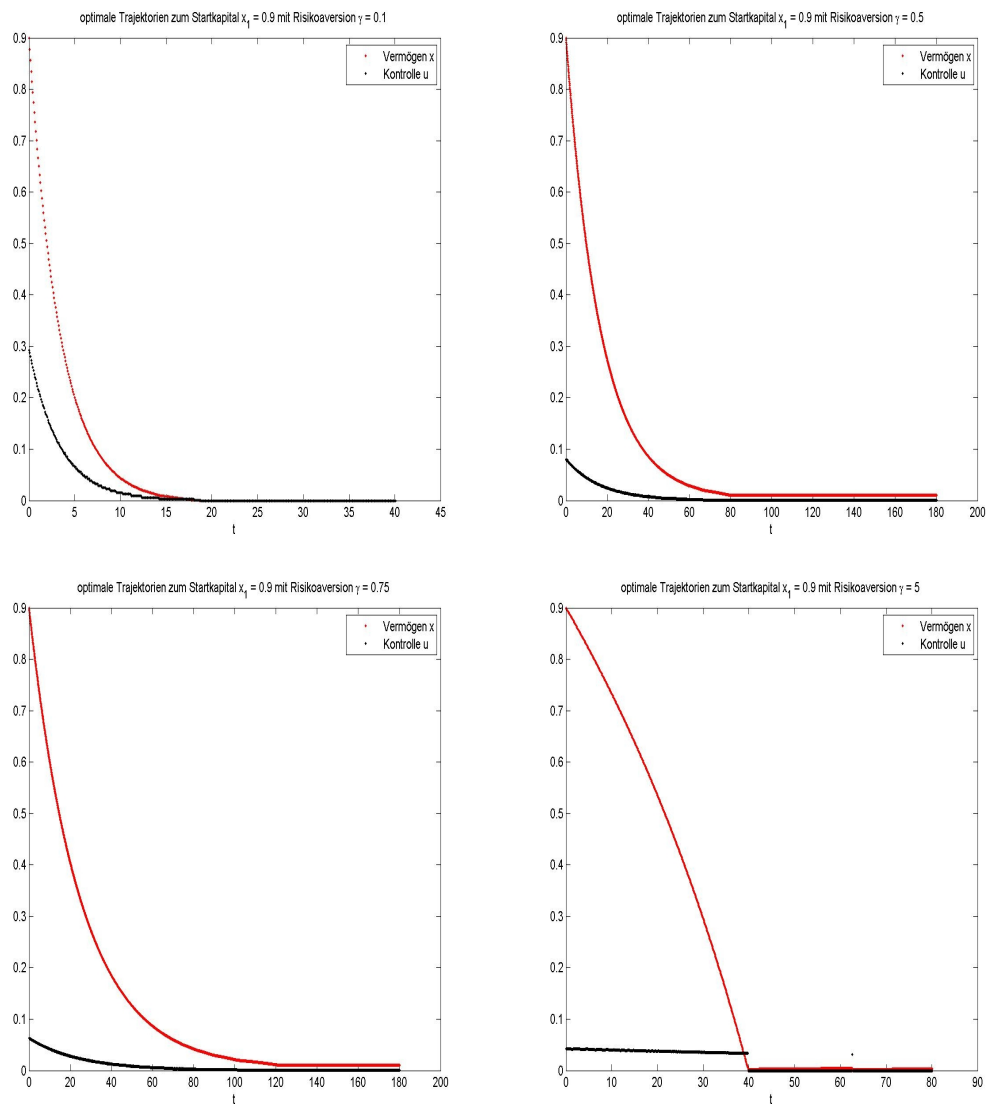


Abbildung 5.7: Die Grafik zeigt die optimalen Trajektorien für das Anfangskapital $x[0] = 0.9$ für $\gamma = 0.1, 0.5, 0.75$ und 5 zur Diskontrate $\delta = 0.06$.

Variation der Rendite

In diesem Abschnitt stehen Auswirkungen auf die Lösungen des optimal gesteuerten Systems im Mittelpunkt, welche auftreten, wenn die Berechnungen mit unterschiedlichen, über die Zeit aber konstanten Renditen durchgeführt werden.

Betrachtet werden daher einige „Portfolios“, welche wie bisher aus einem Vermögenswert bestehen, nun allerdings unterschiedliche risikolose Zinssätze aufweisen.

Beginnen wir die Analyse derartiger „Portfolios“ mit dem Aufstellen einer Vermutung: Wir erwarten, dass sich mit höherem Zinssatz auch der akkumulierte Nutzen bei optimaler Steuerung über die gesamte Zeit, in Abhängigkeit vom Vermögen, erhöht.

Um diese Vermutung zu überprüfen, werden wir das Modell zunächst für die in Tabelle 5.3 aufgeführten Werte testen. Alle übrigen Parameter aus Tabelle 5.1 werden auch hier beibehalten, so dass $\gamma = 0.75$ und $\delta = 0.06$ beträgt.

Asset	Rendite des Assets
A	$r = 3\%$
B	$r = 5\%$
C	$r = 6\%$
D	$r = 9\%$

Tabelle 5.3: Die zur Analyse verwendeten Renditen eines Vermögenswertes.

Die numerisch berechneten optimalen Wertefunktionen des Modells werden gemeinsam in Abbildung 5.8 grafisch veranschaulicht. Auf diese Weise lässt sich eine Veränderung des Verlaufs der optimalen Wertefunktion bei Änderung des Zinssatzes leichter erkennen. Wie erwartet, zeigt sich, dass der akkumulierte Nutzen bei optimaler Steuerung umso größer ist, je höher die Verzinsung des Vermögenswertes ist.

Betrachten wir nun auch hier insbesondere den Verlauf der optimalen Konsumneigung eines Anlegers bei einem sich ändernden Zinssatz. Die Herleitung der Formel für die Konsumneigung in Abschnitt 5.2.2 hat gezeigt, dass diese umso niedriger wird, je höher

5.2 MODELL 1: EIN VERMÖGENSWERT - KONSTANTE RISIKOLOSE RENDITE

die Rendite $r \in [0, 1]$ ist. Ein analoges Resultat liefern die numerisch approximierten Funktionen, welche in der rechten Grafik in Abbildung 5.9 dargestellt sind. Je höher die Vermögensverzinsung, desto reizvoller wird es für einen Anleger das Vermögen anzulegen, anstatt auszugeben, um seinen Nutzen zu maximieren.

Auffallend ist in dieser Darstellung der Verlauf der skalierten Kontrollfunktion zur Rendite 6% bzw. 9%. Im ersten Fall erhalten wir, im Vergleich zu unseren bisherigen Ergebnissen, eine vollkommen konstante Funktion (natürlich mit Ausnahme in der Null). Bei $r = 9\%$ erwarten wir einen konstanten Verlauf bei 0.05. Die Funktion steigt allerdings bis an den Rand des Gitters auf 0.09 an. Verschieben wir nun die rechte Gittergrenze, indem wir den Wert bei $b[0]$ im Programm auf einen größeren Wert setzen, verlagert sich dieser Anstieg weiter nach rechts. Der Verlauf der Konsumneigung ab $r > 6\%$ kann somit auf die Einschränkung auf Ω zurückgeführt und daher vernachlässigt werden. Zudem zeigt die linke Grafik in Abbildung 5.9 die zugehörigen, unskalierten optimalen Kontrollfunktionen des Problems. Auch hier ist zu erkennen, dass die Funktion für $r = 9\%$ vom bisherigen linearen Verlauf abweicht.

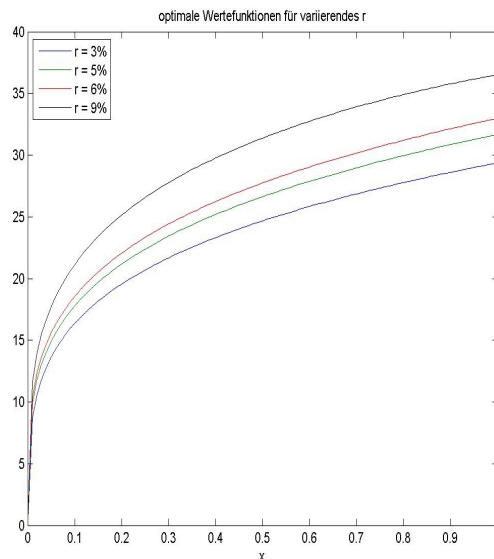


Abbildung 5.8: Die Grafik zeigt die Wertfunktionen für $r = 3\%$, 5% , 6% und 9% für Diskontrate $\delta = 0.06$.

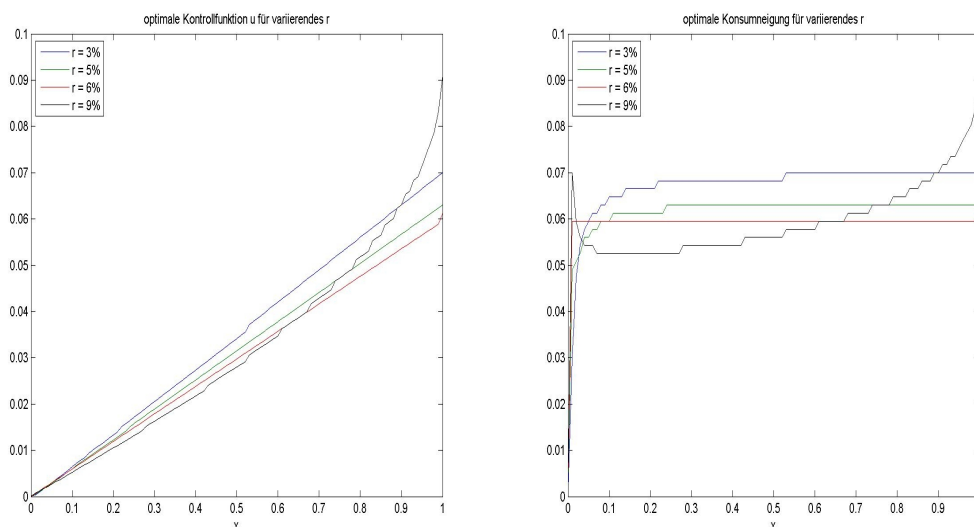


Abbildung 5.9: Die Grafik zeigt die optimale und die skalierte Kontrollfunktion für 3%, 5%, 6% und 9% zur Diskontrate $\delta = 0.06$.

Aufgrund des veränderten Verlaufs dieser Funktionen ab einem Zinssatz von $r \geq 6\%$ analysieren wir das dynamische Verhalten des optimal gesteuerten Systems vor allem in diesem Bereich. Insbesondere die globale Darstellung in Abbildung 5.10 macht deutlich, inwieweit sich die optimalen Gleichgewichte des Systems in Abhängigkeit vom verwendeten Zinssatz ändern. Die rechte Grafik veranschaulicht das Verhalten der Lösungen für Renditen $r \in [3, 10]$, während die linke Grafik speziell den Bereich für $r \in [5.5, 6.5]$ genauer herausstellt. Dem entsprechend zeigt die zeitabhängige Darstellung der optimalen Trajektorien in Abbildung 5.11, dass die Lösungen bis zu einer Rendite $r < 6\%$ in das Gleichgewicht $x^* = 0$ streben. Beträgt die Verzinsung 6%, verlaufen die optimalen Trajektorien konstant, so dass das Vermögen auf der Höhe des Anfangskapital bleibt und auch der Konsum über die Zeit unverändert ist. Ab einer Rendite über 6% nehmen die Funktionen hingegen permanent über die Zeit zu. In der Grafik begrenzt dabei allerdings die Einschränkung auf Ω diesen Anstieg. Die „dickeren“ Linien bei einem Zinssatz von $r = 6\%$ bzw. $r = 9\%$ resultieren aus der Tatsache, dass die Kontrollfunktion in diesem Bereich permanent zwischen zwei verschiedenen Kontrollwerten springt.

5.2 MODELL 1: EIN VERMÖGENSWERT - KONSTANTE RISIKOLOSE RENDITE

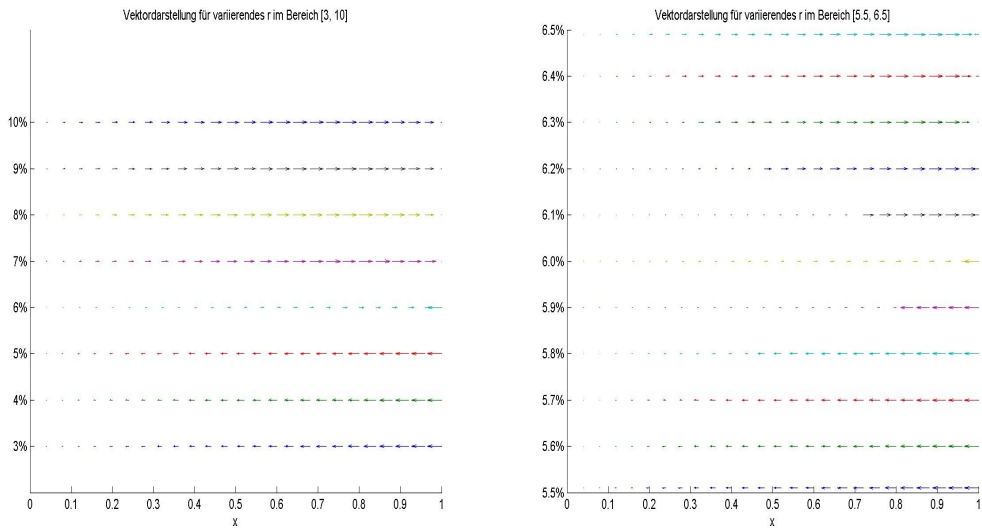


Abbildung 5.10: Die Grafik zeigt links die optimale Vektordarstellung für $r \in [3, 10]$ und rechts den Bereich für $r \in [5.5, 6.5]$.

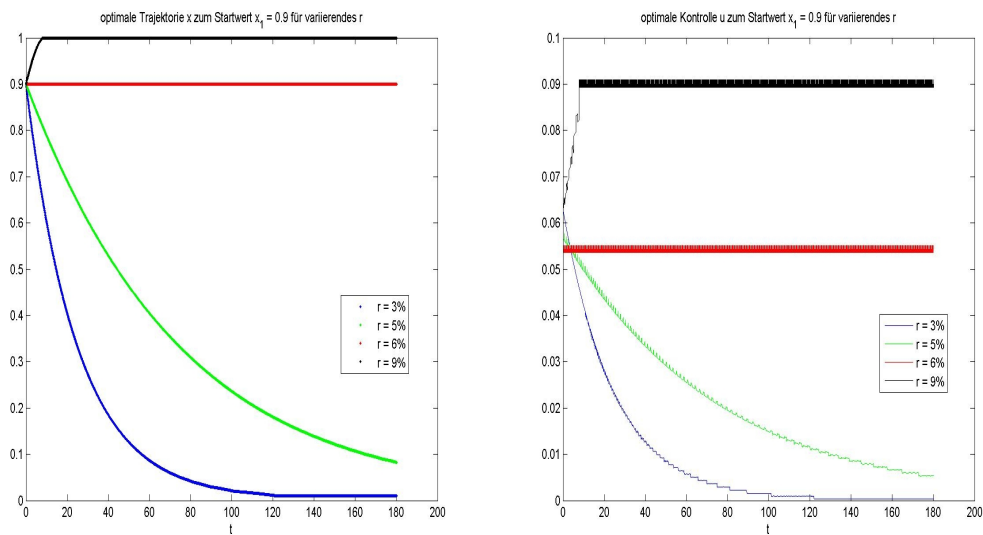


Abbildung 5.11: Die Grafik zeigt die optimalen Trajektorien für 3%, 5%, 6% und 9%.

Variation der Diskontrate

Abschließend soll die Dynamik des optimal gesteuerten Systems bei Veränderungen der Diskontrate δ und damit implizit des Zeithorizonts, über den optimiert wird, analysiert werden. Wie bereits erwähnt, stammt das hier betrachtete Modellproblem des diskontierten Funktional mit dem exponentiellen Diskontfaktor $e^{-\delta t}$ und positiver Diskontrate $\delta > 0$ ursprünglich aus der Ökonomie. Betrachtet man den Diskontfaktor genauer, so konvergiert er für $t \rightarrow \infty$ gegen 0, so dass zeitlich weit in der Zukunft liegende Werte von $g(\Phi(t, x, u), u(t))$ schwächer gewichtet werden (vergleiche 3.9). Ökonomisch gesehen wird somit die Tatsache modelliert, dass der Ertrag bzw. der Nutzen in naher Zukunft wichtiger ist als derjenige in ferner Zukunft. Je größer man nun die Diskontrate δ wählt, desto bedeutender wird für einen Investor der Nutzen in naher Zukunft. Bei einem kleineren Wert für δ ist man hingegen zudem auch an der Maximierung des Nutzens interessiert, der verhältnismäßig weit in der Zukunft liegt.

Die zu untersuchende Frage bei diesem Problem ist nun, wie das Langzeitverhalten der optimalen Lösungen von dem Parameter δ abhängt. Dabei analysieren wir das Modell insbesondere für die in der nachfolgenden Tabelle aufgeführten Diskonraten. Neben diesen Werten wurden die Parameterwerte $\gamma = 0.75$ und $r = 3\%$ zur Lösung des optimalen Steuerungsproblems verwendet.

	Diskontrate $\delta > 0$
A	$\delta = 0.01$
B	$\delta = 0.03$
C	$\delta = 0.06$
D	$\delta = 1$

Tabelle 5.4: Die zur Analyse verwendeten Diskonraten δ .

Analog zu den vorherigen Betrachtungen stellen wir zunächst einen Ausschnitt der approximativ optimalen Wertefunktionen des Problems in Abbildung 5.12 grafisch dar. Je höher der Wert für δ gewählt wurde, desto „flacher“ ist der Verlauf der optimalen Wer-

tefunktion, so dass der akkumulierte Nutzen also umso geringer ausfällt, je wichtiger für einen Investor der Nutzen in naher Zukunft ist.

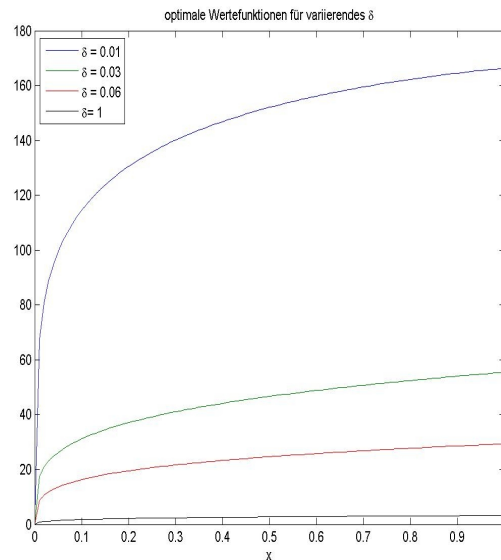


Abbildung 5.12: Die Grafik zeigt die optimalen Wertfunktionen für Diskontraten 0.01, 0.03, 0.06 und 1.

Die Approximationen der optimalen Kontrollfunktion sind in den linken Grafiken von Abbildung 5.13 veranschaulicht, rechts ist die zugehörige optimale Konsumneigung eines Investors dargestellt. Interessant ist hierbei natürlich wieder vor allem die Konsumneigung eines Anlegers, welche nach unseren theoretischen Resultaten aus Abschnitt 5.2.2 mit größerer Diskontrate zunehmen müsste. Auch hier zeigen die numerischen Resultate, dass die Neigung eines Investors, sein Vermögen auszugeben, zunimmt, je größer wir diesen Parameter wählen. Mit anderen Worten ist die Konsumneigung eines Anlegers bei variierendem δ umso größer, je wichtiger für ihn der Nutzen in naher Zukunft ist. Schreibt ein Investor hingegen dem Nutzen in ferner Zukunft eine größere Bedeutung zu, fällt sein Konsum in Abhängigkeit seines Vermögens dementsprechend niedriger aus. An diese Überlegung wollen wir nun anknüpfen und die Vektordarstellung bzw. insbesondere den Verlauf einiger Lösungskurven diskutieren.

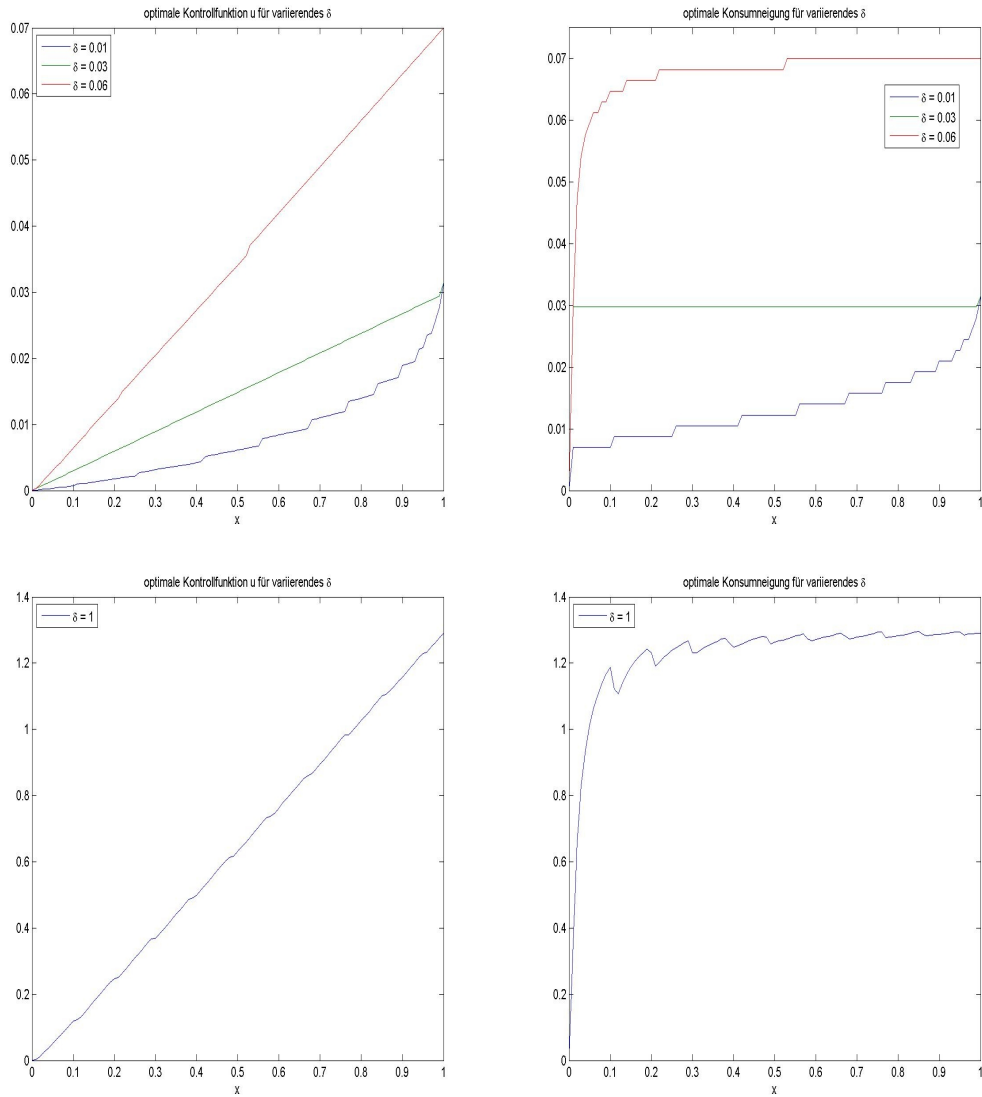


Abbildung 5.13: Die Grafik zeigt links die optimalen Kontrollfunktionen und rechts die zugehörige Konsumneigung für Diskontraten 0.01, 0.03, 0.06 und 1.

5.2 MODELL 1: EIN VERMÖGENSWERT - KONSTANTE RISIKOLOSE RENDITE

Abbildung 5.14 zeigt die Vektordarstellung mit einer feineren Skalierung in der Umgebung von $\delta = 0.03$. In diesem Bereich findet man ein ähnliches Bifurkationsszenario wie im Vorfeld bei variierender Rendite vor. Ab einer Diskontrate von ungefähr $\delta > 0.03$ verlaufen die optimalen Trajektorien nun in die umkehrte, negative Richtung. Ergänzend analysieren wir den Verlauf einiger optimaler Lösungskurven, welche in Abbildung 5.15 grafisch veranschaulicht sind. Auch hier steigen die Lösungskurven ähnlich wie im vorherigen Abschnitt für Parameterwerte $\delta < 0.03$ permanent über die Zeit an. Dies kann unter Verwendung unterschiedlicher Gittergrenzen bestätigt werden. Für $\delta = 0.03$ erhalten wir nahezu konstante Kurven. Ab einer Diskontrate $\delta > 0.03$ hingegen streben die Kurven gegen das Gleichgewicht $x^* = 0$.

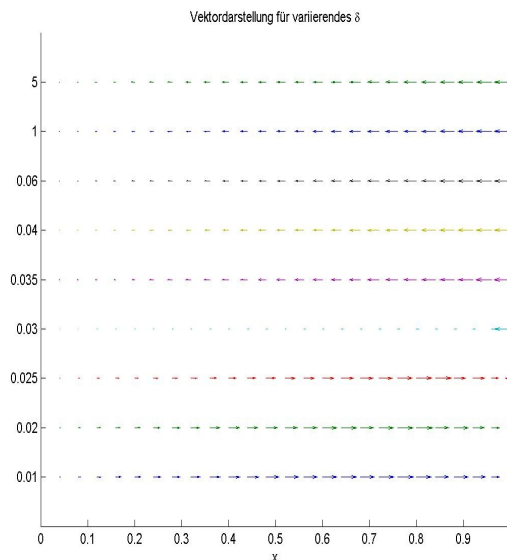


Abbildung 5.14: Die Grafik zeigt die Vektordarstellung für unterschiedliche Diskonraten im Bereich $[0, 5]$.

Zusammenfassend wird ein Anleger, für den sowohl die Optimierung des Nutzens in naher als auch in ferner Zukunft bedeutend ist, sein Vermögen anlegen, anstatt es zu verbrauchen. Ist hingegen insbesondere die Maximierung in naher Zukunft bedeutend, wird sich ein Investor gegen eine Anlage und für den Konsum entscheiden. Abhängig von der Diskontrate kehrt sich auch hier das Konsumverhalten eines Investors um.

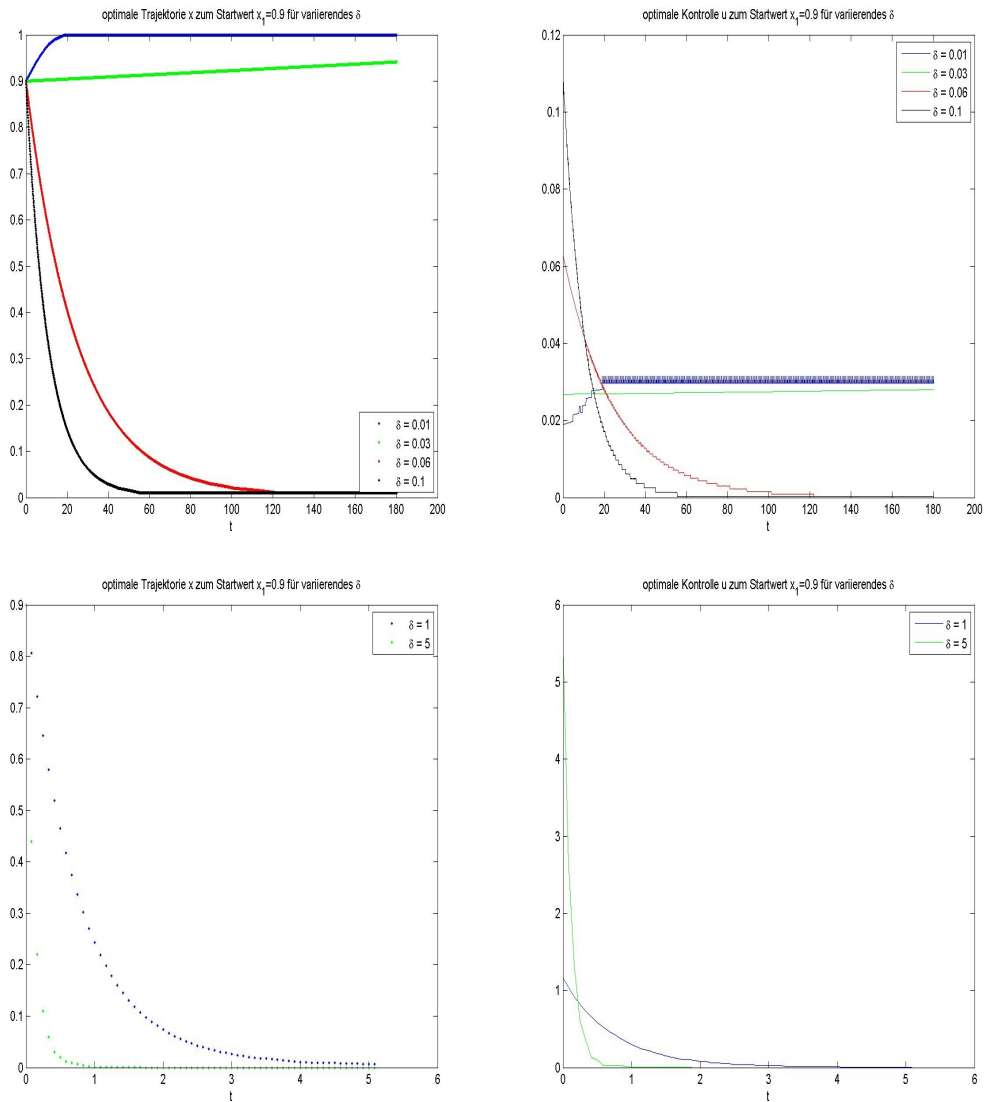


Abbildung 5.15: Die Grafik zeigt die optimalen Trajektorien für Diskontraten 0.01, 0.03, 0.06, 0.1, 1 und 5.

5.3 Modell 2: Zwei Vermögenswerte - Zeitlich variierender Zinssatz und Eigenkapitalrendite

5.3.1 Einführung

Um die Anzahl der bisher verwendeten, festen Variablen möglichst klein zu halten, betrachten wir ein weiteres Modell, bei welchem die Rendite nun in Abhängigkeit vom Vermögen x dargestellt wird. Wie bereits erwähnt, wird auch diesem Portfolio-Modell ein deterministisches Rahmenwerk zugrunde gelegt, so dass auch hier der Unsicherheitsfaktor keine Rolle spielt und die Rendite eines Vermögenswertes unabhängig von zufälligen äußeren Einflüssen ist.

Im Unterschied zum vorherigen Modell werden allerdings zwei risikolose Vermögenswerte betrachtet. Der erste Wert stellt ein risikoloses Asset dar, welches einen zeitlich variierenden Zinssatz $R_{f,t}$ aufweist, der zweite beschreibt das Eigenkapital eines Anlegers und besitzt eine sich zeitlich ändernde Eigenkapitalrendite $R_{e,t}$.

Um unterschiedliche Verzinsungen und verschieden starke Schwankungen der Renditen zu realisieren, werden diese zum einen in Abhängigkeit vom Vermögen x betrachtet, zum anderen aus Gründen der Einfachheit mithilfe der Sinus-Funktion modelliert. Auf diese Weise erhält man einen periodischen Verlauf der Renditen, wobei sich die Amplitude sowie die Periode der Funktion mit Hilfe der im Folgenden verwendeten Parameter $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ und α_5 beeinflussen lässt. Als zeitlich variierenden Zinssatz wählen wir entsprechend SEMMLER [15]

$$R_{f,t}(x) = \alpha_1 \sin(\alpha_3 x(t))$$

und für die Eigenkapitalrendite

$$R_{e,t}(x) = \alpha_2 \sin(\alpha_4 x(t)) + \alpha_5.$$

Ein weiterer Unterschied zum ersten Beispiel besteht darin, dass ein Investor nun die Möglichkeit hat, neben dem Konsum u_1 auch die Allokation seines Vermögens zu beeinflussen. Das heißt, er entscheidet, welchen Bruchteil seines Vermögens er wann in welchen Vermögenswert investiert. Im Folgenden modelliere daher eine zweite Kontrollvariable u_2 den Anteil an Vermögen, welcher in das Eigenkapital fließt. $(1 - u_2)$ ist

folglich der Teil, welcher im risikolosen Asset gehalten wird.

Das zweite zu betrachtende Modell ist demnach eindimensional im Zustand x und besitzt eine zweidimensionale Kontrolle $u = (u_1, u_2)^T \in \mathbf{U}$. Die Dynamik des Vermögens aus dem ersten Modell kann folgendermaßen erweitert werden:

$$\begin{aligned}\dot{x}(t) &= u_2(t) R_{e,t}(x) x(t) + (1 - u_2(t)) R_{f,t}(x) x(t) - u_1(t) \\ &= (u_2(t) (R_{e,t}(x) - R_{f,t}(x)) + R_{f,t}(x)) x(t) - u_1(t)\end{aligned}$$

Die nachfolgende Tabelle fasst die an dieser Stelle verwendeten Variablen und deren Bedeutung kurz zusammen:

Variable	Bedeutung der Variable
x	Gesamtvermögen
u_1	Konsum
u_2	Anteil des Vermögens investiert im Eigenkapital
$R_{e,t}$	zeitlich variierende Eigenkapitalrendite
$R_{f,t}$	zeitlich variierender Zinssatz des risikolosen Assets

Tabelle 5.5: Die verwendeten Variablen und deren Bedeutung.

Das Ziel eines Anleger besteht weiterhin darin, den Nutzen, definiert über den Konsum, zu maximieren. Um die Präferenzen eines Entscheiders zu modellieren, wird auch hier die Power-Nutzenfunktion (5.1) verwendet. Das optimale Steuerungsproblem lautet dann:

$$v(x) = \max_{u_1, u_2 \in \mathbf{U}} \int_0^{\infty} e^{-\delta t} \frac{u^{1-\gamma}}{1-\gamma} dt$$

5.3.2 Diskussion der praktischen Ergebnisse

Bei der Auswertung des zweiten Beispiels wird analog zur Analyse des ersten Modells vorgegangen. Daher werden in diesem Abschnitt zunächst Ausgangswerte festgelegt und Lösungen für das Modell berechnet. Für die ermittelten Ergebnissen werden schließlich Grafiken erstellt, welche die Basis für anschließende Variationen in den Parametern bilden. Für einen besseren Überblick fasst Tabelle 5.6 die zur Auswertung verwendeten Parameter zusammen.

Parameter	Verwendeter Wert
Ω	$[0, 100]$
$x[0]$	1200
$(x[0], x[1])$	$(x, 0)$ mit $x \in [0, 100]$
h	$\frac{1}{12}$
$[t, T]$	$[0, 12]$
N_0	1
γ	0.75
δ	0.05
α_1	0.1
α_2	0.2
α_3	0.2
α_4	0.2
α_5	0.005

Tabelle 5.6: Parameterwerte zur Auswertung von Modell 2.

Als Kontrollwertebereich wählen wir für die mit dem Vermögen skalierte Kontrolle $\tilde{u}_1 := \frac{u_1}{x}$ die Menge $\tilde{U}_1 = [0, 2]$, diskretisiert mit $q = 2001$ äquidistanten Kontrollwerten. Auf diese Weise erhalten wir für $x \in [0, 100]$ Kontrollwerte u_1 im Bereich $U_1 = [0, 2x] = [0, 200]$. Für die zweite Kontrolle gilt: $u_2 \in [-3, 4.5]$ mit Unterteilung $r = 16$.⁶

⁶siehe entsprechende Stelle im Programm.

Die resultierenden Renditen sind in Grafik 5.16 abgebildet. Die Differenz von $R_{e,t}(x)$ und $R_{f,t}(x)$ bezeichnet man als Eigenkapitalprämie.

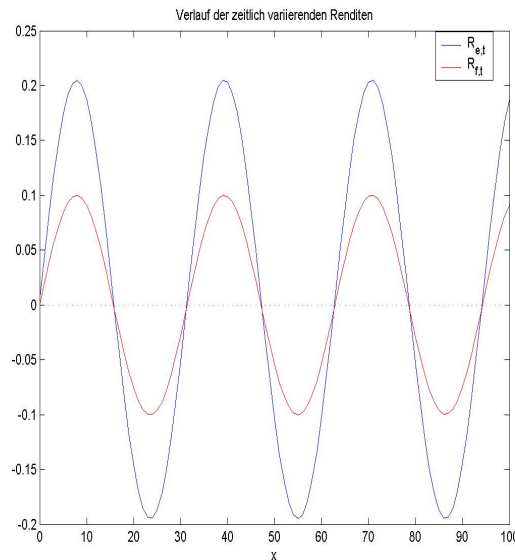


Abbildung 5.16: Die Grafik zeigt die Renditen $R_{e,t}(x)$ bzw. $R_{f,t}(x)$ für $\alpha_1 = 0.1$, $\alpha_{2-4} = 0.2$ und $\alpha_5 = 0.005$.

Da sowohl die zugrunde gelegte Eigenkapitalrendite als auch die risikolose Zinsrate periodisch gewählt wurden, verläuft auch die Eigenkapitalprämie zyklisch. Das hat Auswirkungen auf die optimale Wertefunktion, welche in Abbildung 5.17 (S.92) grafisch veranschaulicht ist. Auch sie verläuft zyklisch und weist mehrere „Knicke“ auf.

Derartige „Knicke“ sind ein typisches und sehr interessantes Phänomen optimal gesteuert Systemen. Betrachtet man in diesen Fällen die optimale Kontrollfunktion bzw. das optimale Vektorfeld, existiert an entsprechender Stelle eine Kurve, welche die Einzugsbereiche der optimalen Gleichgewichte trennt. An dieser Trennlinie sind das Vektorfeld und das optimale Feedback $u^*(x)$ unstetig. Bei einem ungesteuerten dynamischen System würde man hier die Existenz eines instabilen Gleichgewichts erwarten, bei einem optimal gesteuerten System tritt dies hingegen nicht auf. In diesem Fall besteht die ganze Trennlinie aus Punkten, für die die optimale Steuerung nicht eindeutig ist. An späterer

Stelle werden wir darauf noch einmal zurückkommen. In der Ökonomie bezeichnet man eine derartige Kurve oft als *Skiba-Kurve*.

Auf Grund der zyklischen Gestalt der Eigenkapitalprämie kann auch ein periodischer Verlauf sowohl bei den optimalen Kontrollfunktionen als auch bei der optimalen Konsumneigung eines Investors erwartet werden. Abbildung 5.18 veranschaulicht diese Funktionen grafisch. Der optimale Konsum verläuft in gewisser Weise zyklisch, nimmt im Durchschnitt über die Zeit allerdings zu. Dabei fällt sowohl der optimale Konsum als auch die Konsumneigung beim Übergang von einer negativen zu einer positiven Prämie – und somit im Bereich der „Knicke“ der optimalen Wertefunktion – auf 0 ab.

Der ungleichmäßig gezackte Verlauf der Kontrollfunktion u_1 unterbricht die Periodizität dieser Funktion, er kann aber auf numerische Ungenauigkeiten zurückgeführt werden. Um die Gesamtgenauigkeit der Berechnung zu verbessern, müsste somit die Anzahl der Kontrollwerte im Kontrollwertebereich erhöht werden. Auf diese Weise würden sich numerische Fehler verringern und die genannten Zacken verschwinden. Eine Erhöhung der äquidistanten Kontrollwerte bei den Berechnungen hat allerdings nicht den gewünschten Effekt bewirkt, so dass an dieser Stelle eine gewisse Ungenauigkeit akzeptiert werden muss.

Der Anteil des Vermögens, welcher in das Eigenkapital investiert wird, bewegt sich hingegen periodisch zwischen den festgesetzten Grenzen -3 und 4.5. Das Vermögen wird dabei entweder vollständig im Eigenkapital oder im risikolosen Asset angelegt. Diese Entscheidung des Anlegers lässt sich erklären, wenn man Abbildung 5.16 mit der linken Grafik aus Abbildung 5.18 vergleicht. Die optimale Kontrolle u_2 , das heißt der Anteil, welcher in das Eigenkapital fließt, nimmt einen sehr großen positiven bzw. negativen Wert an, wenn die Eigenkapitalprämie positiv bzw. negativ ist.

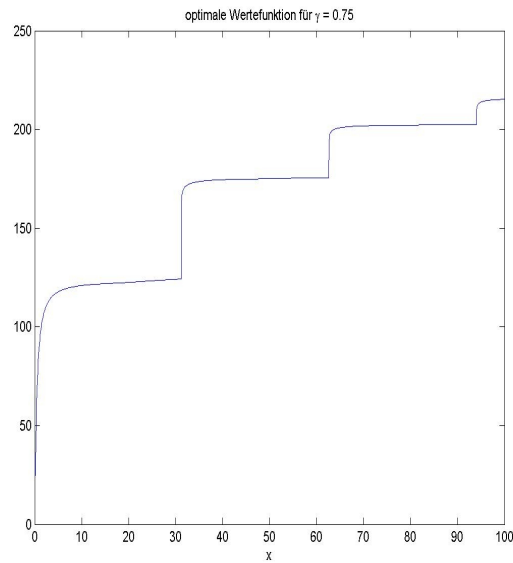


Abbildung 5.17: Die Grafik zeigt die optimale Wertefunktion des Modells für $\gamma = 0.75$ und $\delta = 0.05$.

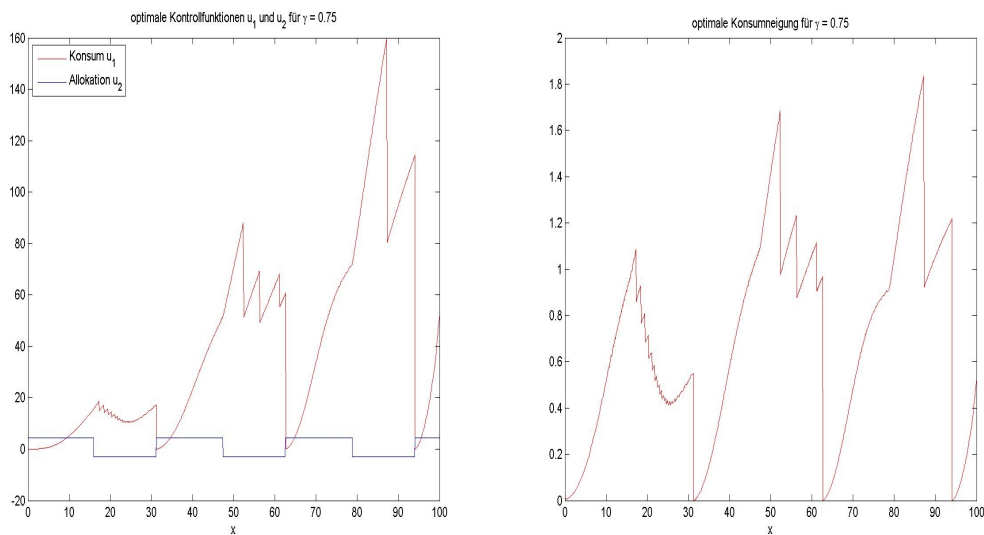


Abbildung 5.18: Die Grafik zeigt links die optimalen Kontrollfunktionen u_1 und u_2 , rechts die optimale Konsumneigung \tilde{u}_1 für $\gamma = 0.75$ und $\delta = 0.05$.

5.3 MODELL 2: ZWEI VERMÖGENSWERTE - ZEITLICH VARIIERENDER ZINSSATZ UND EIGENKAPITALRENDITE

Um eine anschauliche Darstellung der optimalen Gleichgewichte zu erhalten, soll anstelle des Vektorfeldes ein optimales Gleichgewicht nun in Form eines grün ausgefüllten Kreises, die Trennlinien ihrer Einzugsbereiche durch einen rot gefärbten Kreis veranschaulicht werden (vgl. Abbildung 5.19). Besonders deutlich wird somit einerseits die Anzahl der existierenden Gleichgewichte andererseits die Tatsache, dass sich derartige Trennlinien ungefähr an den Stellen befinden, an denen wir zuvor in der optimalen Wertefunktion einen „Knick“ feststellen konnten.

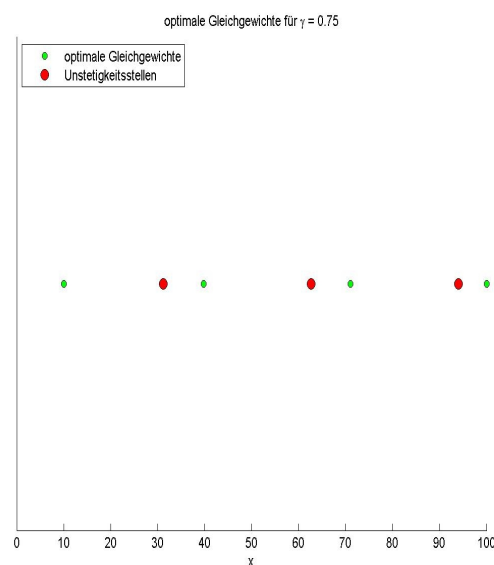


Abbildung 5.19: Die Grafik zeigt die optimalen Gleichgewichte sowie die Trennlinien der Einzugsbereiche für $\gamma = 0.75$ und $\delta = 0.05$.

Eine alternative Darstellung bieten auch hier die optimalen Trajektorien, welche in Abbildung 5.20 (S. 95) grafisch veranschaulicht sind. Insbesondere wird hierbei deutlich, dass die optimale Steuerung für Punkte auf den Trennlinien nicht eindeutig ist. In diesem Fall zeigt sich, dass es gleichermaßen optimal ist, sowohl nach links als auch nach rechts zu steuern. Für keinen dieser Punkte jedenfalls ist es optimal, in diesem Punkt zu bleiben. Im Beispiel-Modell finden wir derartige Trennlinien immer dann, wenn die negative Eigenkapitalprämie zu positiven Werten übergeht. Asymptotisch stabile Gleichgewichte liegen hingegen ungefähr bei $x_1^* \approx 10$, $x_2^* \approx 40$ und $x_3^* \approx 71$ und somit in etwa in den

Punkten, in welchen die positive Prämie maximal ist. An dieser Stelle wird der Punkt $x = 100$ ebenfalls als Gleichgewicht dargestellt, da bei der Einschränkung auf die Menge Ω im oberen Bereich ein weiterer Vorzeichenwechsel stattfindet. Allerdings handelt es sich hierbei nicht um ein weiteres Gleichgewicht für unser Modell. Bei Wahl eines größeren Bereichs Ω zeigt sich, dass das Vermögen weiter steigen würde und erst ungefähr im Punkt $x \approx 103$ in ein weiteres Gleichgewicht laufen würde, also erneut dann, wenn die positive Prämie maximal wird.

Besonders auffallend bei der Betrachtung der einzelnen Bereiche ist, dass das Vermögen bei einer ansteigenden positiven Prämie (vgl. Abbildung 5.16) über die Zeit zunimmt, obwohl es jede Periode durch den Konsum reduziert wird. Bei einer fallenden positiven bzw. einer negativen Prämie nimmt das Vermögen hingegen ab.

Den Verlauf des zugehörigen optimalen Konsums in Abhängigkeit von der Zeit t zeigt Abbildung 5.21. Insgesamt lässt sich feststellen, dass die optimalen Kurven in ein Gleichgewicht laufen, das von der Wahl des Startwertes abhängt: Nimmt das Vermögen über die Zeit zu, so nimmt auch der optimale Konsum zu, fällt es hingegen über die Zeit, so verhält sich der Konsum entsprechend. Wählt man als Startwert einen Gleichgewichtspunkt, so bleiben die Kurven ebenfalls konstant.

Auf die Darstellung der optimalen Allokation des Vermögens kann an dieser Stelle verzichtet werden, da sie wie bereits beschrieben, periodisch zwischen den festgesetzten Grenzen verläuft.

Analog zum ersten Beispiel-Modell sollen nun auch hier Auswirkungen auf das Langzeitverhalten der Lösungen des optimal gesteuerten Systems bei variierenden Parametern analysiert werden.

5.3 MODELL 2: ZWEI VERMÖGENSWERTE - ZEITLICH VARIIERENDER ZINSSATZ UND EIGENKAPITALRENDITE

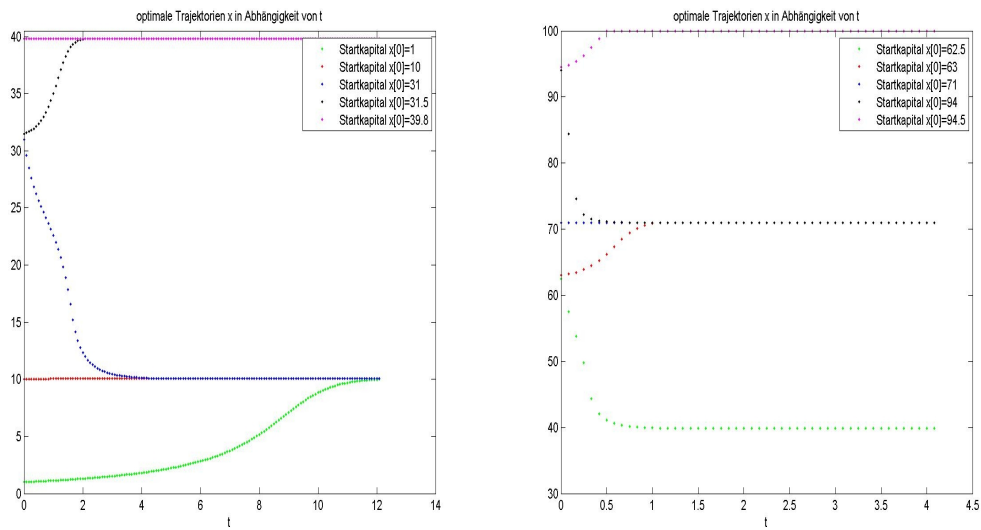


Abbildung 5.20: Die Grafik zeigt die optimalen Trajektorien x für $\gamma = 0.5$ und $\delta = 0.05$.

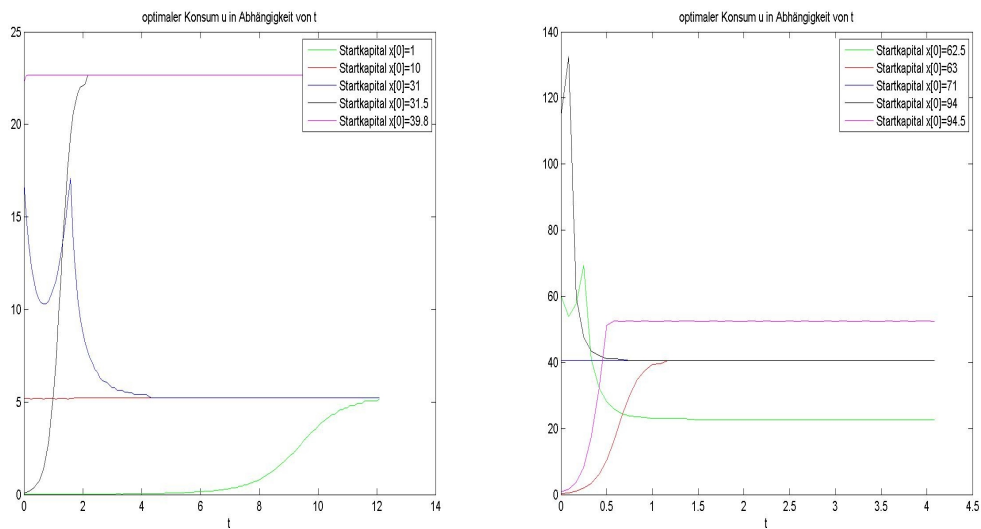


Abbildung 5.21: Die Grafik zeigt den optimalen Konsum für $\gamma = 0.75$ und $\delta = 0.05$.

5.3.3 Variationen in den Parametern

Variation der Risikoaversion

Zunächst betrachten wir, wie sich unterschiedliche Risikoaversionen auf die Dynamik des optimal gesteuerten Systems auswirken. Analog zu Modell 1 werden hier für γ die Parameterwerte 0.1, 0.5, 0.75 bzw. 1 zur Auswertung herangezogen.

Die resultierenden approximativ optimalen Wertefunktionen werden in der nachfolgenden Abbildung 5.22 veranschaulicht.

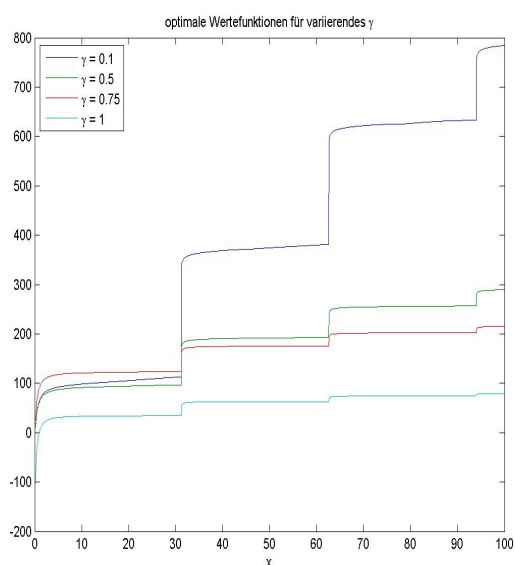


Abbildung 5.22: Die Grafik zeigt die optimalen Wertefunktionen für $\gamma : 0.1, 0.5, 0.75$ und 1 für $\delta = 0.05$.

Im Unterschied zum ersten Modell nimmt der akkumulierte Nutzen bei optimaler Steuerung in Abhängigkeit vom Vermögen nun mit höherer Risikoaversion ab. Die „Knicke“ der Wertefunktion bleiben dabei allerdings unverändert, so dass wir in Abhängigkeit von der Risikoaversion keine Veränderung der Anzahl optimaler Gleichgewichte zu erwarten haben. Daher soll es an dieser Stelle auch genügen, ausschließlich die optimale Konsumneigung eines Investors sowie das globale Verhalten der optimalen Lösungen auszuwerten.

5.3 MODELL 2: ZWEI VERMÖGENSWERTE - ZEITLICH VARIIERENDER ZINSSATZ UND EIGENKAPITALRENDITE

In Abbildung 5.23 erkennt man, dass die Konsumneigung umso höhere Werte annimmt, je risikofreudiger ein Investor ist. So steigt die Funktion bei $\gamma = 1$ maximal auf den Wert 1.6 an, während man bei $\gamma = 0.1$ häufig einen Anstieg bis über 2 vorfindet. Umso reizvoller wird es demnach auch hier für einen Investor, das Vermögen anzulegen anstatt zu konsumieren, je stärker er einem Risiko abgeneigt ist.

Die optimale Allokation u_2 weist bei variierendem γ ebenfalls keine Veränderungen auf, so dass u_2 wie bisher einen sehr großen positiven bzw. negativen Wert annimmt, wenn die Eigenkapitalprämie positiv bzw. negativ ist.

Abbildung 5.24 (S. 99) veranschaulicht das globale Verhalten optimaler Lösungen bei veränderter Risikoaversion. Unabhängig von γ erhalten wir jeweils drei optimale Gleichgewichte. Dass der hier als Gleichgewicht eingezeichnete Punkt $x = 100$ kein Gleichgewicht für unser Modell darstellt, lässt sich wie im vorherigen Abschnitt begründen.

Zusammenfassend können wir folgern, dass sich Variationen in der Risikoaversion kaum im Verhalten eines Investors niederschlagen. Allerdings fällt die Konsumneigung mit zunehmender Aversion.

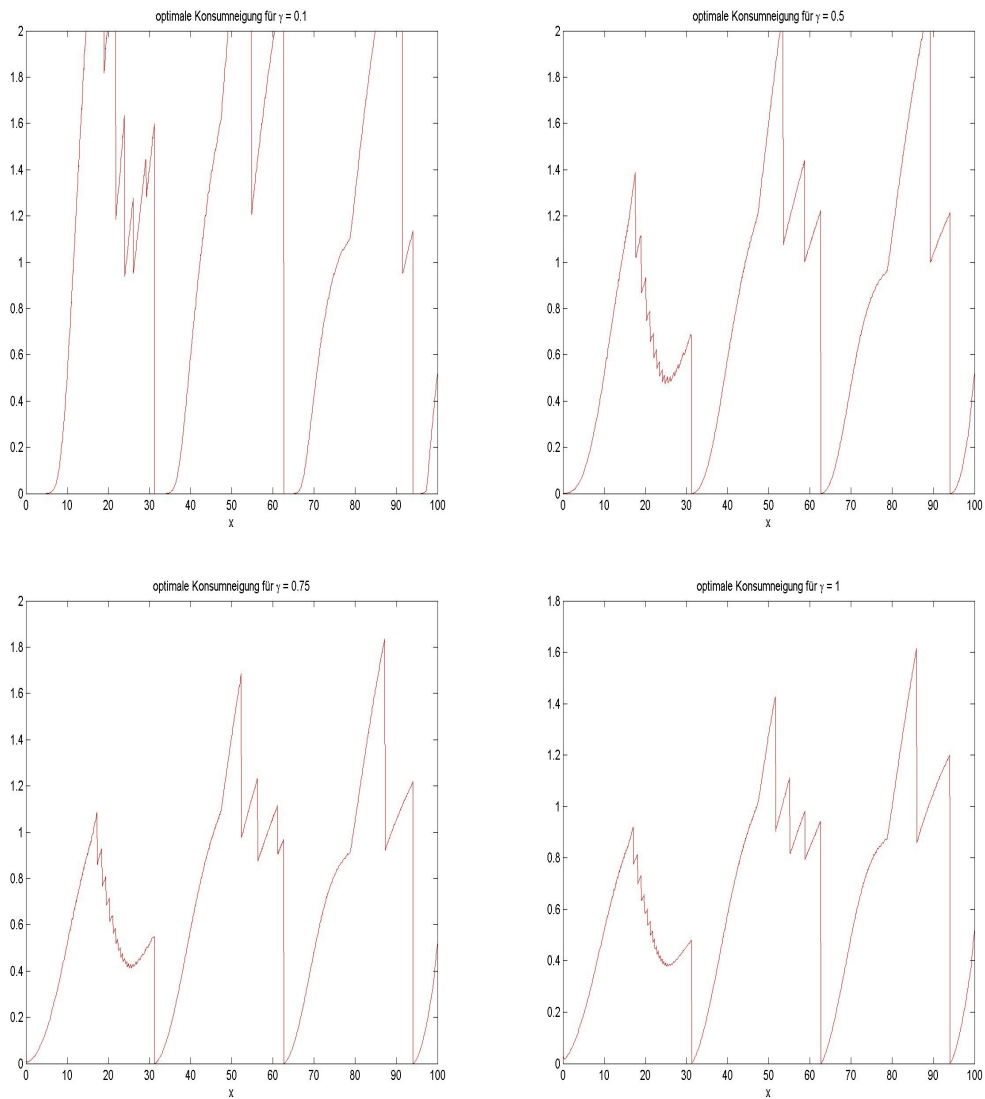


Abbildung 5.23: Die Grafik zeigt die optimale Konsumneigung für $\gamma : 0.1, 0.5, 0.75$ bzw. 1 sowie $\delta = 0.05$.

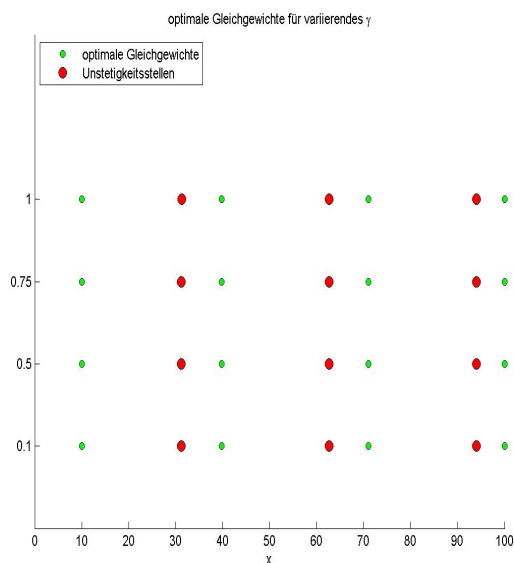


Abbildung 5.24: Die Grafik zeigt die optimalen Gleichgewichte für $\gamma : 0.1, 0.5, 0.75$ und 1 sowie $\delta = 0.05$.

Variation der Renditen

Durchaus mehr Veränderung im Verhalten eines Anlegers zeigt sich für unterschiedliche Renditen, wobei wir uns auf die Betrachtung der in Tabelle 5.7 (S. 100) zusammengefassten Variationen beschränken. Unter Verwendung dieser Parameterwerte können wir nun analysieren, inwieweit sich Veränderungen ergeben, wenn der Zinssatz des risikolosen Vermögenswertes stärkeren Schwankungen unterworfen ist. In unserem Fall heißt das konkret, dass die Periode der Funktion $R_{f,t}$ kleiner ist als diejenige der Funktion $R_{e,t}$.

Wir betrachten dazu zwei Varianten:

1. $R_{f,t}$ durchläuft zwei Perioden, während $R_{e,t}$ nur eine Periode aufweist
2. $R_{f,t}$ durchläuft vier Perioden, während $R_{e,t}$ nur eine Periode aufweist

Ferner werden wir als dritte Variante das Verhalten eines Anlegers untersuchen, wenn nun die Eigenkapitalrendite stärkeren Schwankungen ausgesetzt ist.

	Ausgangsmodell	Variation 1	Variation 2	Variation 3
α_1	0.1	0.1	0.1	0.1
α_2	0.2	0.2	0.2	0.2
α_3	0.2	0.4	0.8	0.2
α_4	0.2	0.2	0.2	0.4
α_5	0.005	0.005	0.005	0.005

Tabelle 5.7: Für Modell 2 verwendete Renditen.

Abbildung 5.25 (S. 101) veranschaulicht die approximativ optimalen Wertefunktionen der unterschiedlichen Variationen. Während sie sich im Vergleich zur Ausgangsdarstellung in der Höhe nur gering unterscheiden, weisen alle drei Varianten im unteren Bereich mindestens einen weiteren „Knick“ auf. An diesen Stellen müssten wir also bei der Darstellung der Gleichgewichte weitere Einzugsbereiche bzw. Trennlinien finden. Weiterhin auffallend bei der Betrachtung der zweiten Variante ist die im oberen Verlauf stark abnehmende Periodizität der optimalen Wertefunktion. Inwieweit sich dadurch Änderungen für das Verhalten eines Investors ergeben, werden wir nun im Folgenden untersuchen.

Abbildung 5.26 (S. 102) zeigt links die optimalen Kontrollfunktionen u_1 und u_2 , rechts die zugehörige Konsumneigung eines Investors. Um Veränderungen auszuwerten, soll an dieser Stelle allerdings die Betrachtung der Konsumneigung ausreichen. Vergleicht man die ersten beiden Varianten, zeigt sich, dass die Konsumneigung eines Anlegers umso stärker variiert, je stärker die Zinsschwankungen des risikolosen Assets sind. Insgesamt bleibt die Konsumneigung dabei im zweiten Fall weitaus geringer als bei der ersten Variante, so dass es bei stark schwankenden Zinssätzen umso reizvoller wird, das Vermögen anzulegen anstatt zu konsumieren. Verkürzt man hingegen die Periode der Funktion mit der größeren Amplitude, in unserem Fall somit die Rendite des Eigenkapitals (vgl. Abbildung 5.16), erhalten wir einen mit der ersten Variante vergleichbaren Verlauf. Abweichend sind hierbei kleine Schwankungen vor einem erneuten Anstieg der Funktion.

5.3 MODELL 2: ZWEI VERMÖGENSWERTE - ZEITLICH VARIIERENDER ZINSSATZ UND EIGENKAPITALRENDITE

Insbesondere treten nun auch bei der zweiten Kontrollfunktion stärkere Schwankungen auf. Weist die Rendite des zinslosen Assets sowohl eine geringere Amplitude als auch eine kleinere Periode als die des Eigenkapitals auf, wird ein Anleger bei positiver Eigenkapitalrendite sein Vermögen weiterhin komplett im Eigenkapital investieren. Schwankt die Eigenkapitalrendite hingegen stärker als der Zinssatz des zweiten Vermögenswertes, verläuft die Rendite $R_{e,t}$ dementsprechend häufiger unter der Funktion für $R_{f,t}$. Aus diesem Grund wird ein Investor hier nun häufiger die Allokation seines Vermögens ändern, um seinen Nutzen zu maximieren.

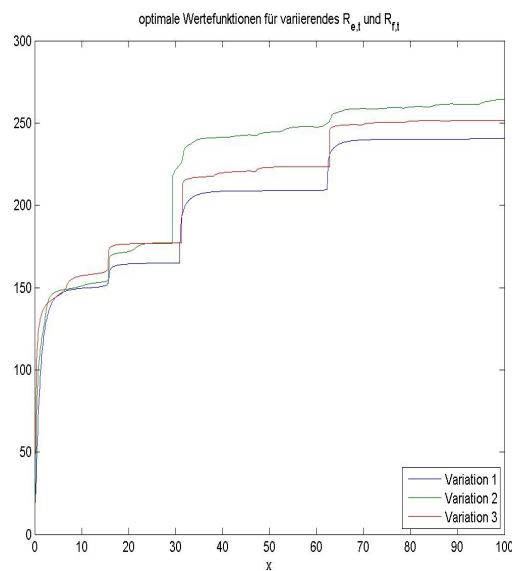


Abbildung 5.25: Die Grafik zeigt die optimalen Wertefunktionen für variierende Renditen $R_{e,t}$ und $R_{f,t}$.

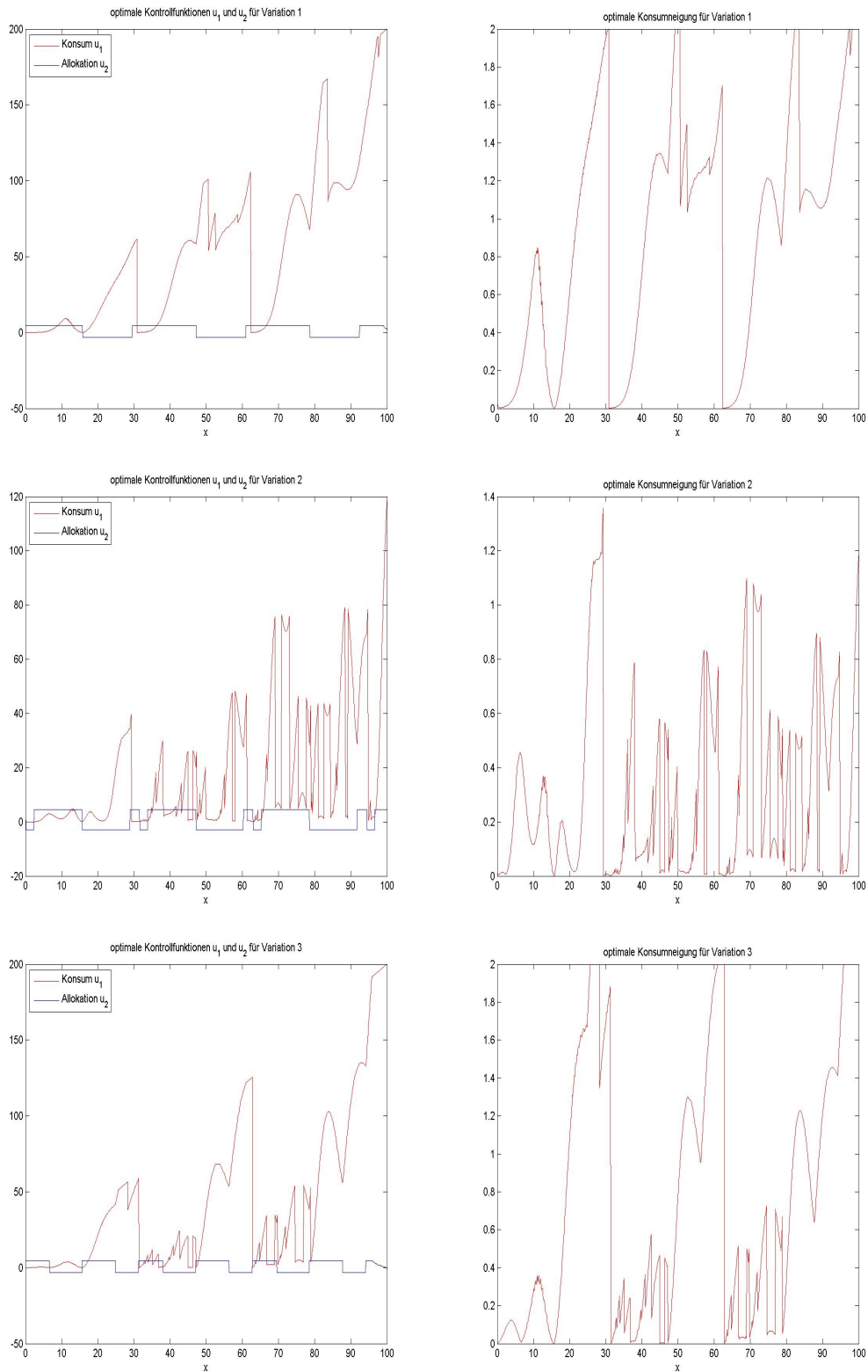


Abbildung 5.26: Die Grafik zeigt links die optimalen Kontrollfunktionen, rechts die zugehörigen Konsumneigungen für variierende Renditen.

5.3 MODELL 2: ZWEI VERMÖGENSWERTE - ZEITLICH VARIIERENDER ZINSSATZ UND EIGENKAPITALRENDITE

Abschließend werden wir analysieren, inwieweit sich Änderungen in Anzahl und Lage der optimalen Gleichgewichte ergeben, um so auf den Verlauf optimaler Trajektorien schließen zu können. Abbildung 5.27 veranschaulicht dazu die optimalen Gleichgewichte sowie die Trennlinien ihrer Einzugsbereiche. Dabei werden für einen besseren Überblick unter der Bezeichnung „Variation 0“ die Ergebnisse der Ausgangsgrafik dargestellt. Für kleinere Perioden des risikolosen Zinssatzes tritt, wie zu Beginn des Abschnitts vermerkt, ein weiteres Gleichgewicht auf, während sich das ursprüngliche Gleichgewicht $x^* \approx 10$ weiter nach rechts verlagert. Für Startwerte $x \leq 10$ steigt das Vermögen somit weiter an. Alle übrigen Gleichgewichte und deren Einzugsbereiche bleiben nahezu unverändert. Betrachten wir die zweite und dritte Variation, treten neben den bisherigen Gleichgewichten weitere Gleichgewichtspunkte auf. Da sich die Anzahl teilweise verdoppelt, wird das Vermögen vor allem im oberen Bereich nur wenig ansteigen bzw. abfallen. Teilweise sind dabei die Einzugsbereiche wieder so klein, dass die Punkte in der Darstellung verdeckt werden. Zu beachten ist auch hier, dass mit dem Punkt $x = 100$ kein Gleichgewichtspunkt gegeben ist.

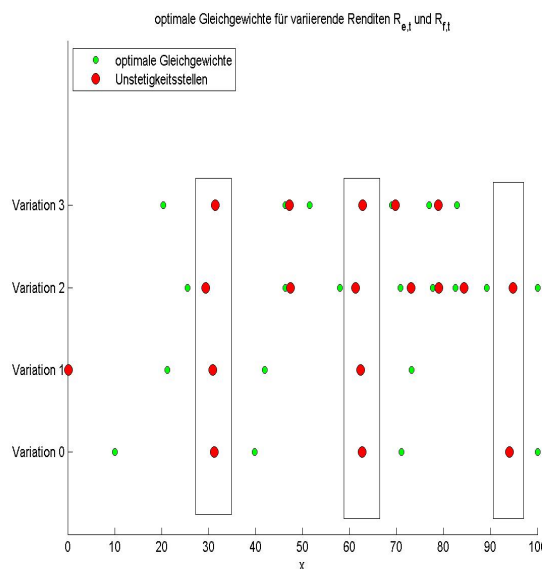


Abbildung 5.27: Die Grafik zeigt die optimalen Gleichgewichte für variierende Renditen $R_{e,t}$ und $R_{f,t}$.

Variation der Diskontrate

Analog zum ersten Modell wollen wir abschließend untersuchen, wie sich die Lösungen des zweiten Beispiels in Abhängigkeit der Diskontrate δ verhalten. Um den effektiven Optimierungshorizont zu verändern, variieren wir diese entsprechend der in Tabelle 5.8 aufgeführten Werte.

	Diskontrate $\delta > 0$		Diskontrate $\delta > 0$
A	$\delta = 0.01$	F	$\delta = 0.9$
B	$\delta = 0.03$	G	$\delta = 1$
C	$\delta = 0.05$	H	$\delta = 2$
D	$\delta = 0.1$	I	$\delta = 11$
E	$\delta = 0.5$		

Tabelle 5.8: Für Modell 2 verwendete Diskonraten.

Auffallend bei der Analyse der approximativ optimalen Wertefunktionen in Abbildung 5.28 ist die umso stärker abnehmende Periodizität, je kleiner der betrachtete Zeithorizont wird. Für sehr große Diskonraten nähert sich dabei der Verlauf der optimalen Wertefunktion dem Verlauf der Power-Nutzenfunktion an. Bei der Optimierung über einen kurzen Horizont überträgt sich demnach der konkave Verlauf der Nutzenfunktion fast direkt auf die optimale Wertefunktion. Optimiert man hingegen über den unendlichen Zeithorizont, werden die periodischen Effekte der Dynamik durch den langen Zeitraum viel stärker sichtbar. Im Hinblick auf die Trajektorien bedeutet das, dass bei kurzem Horizont die Lösung das Durchlaufen der „Täler“ der Dynamik vermeidet, bei längerem Horizont kann sich das aber lohnen, wenn man dafür später in einem Bereich mit höherem Nutzen landet.

5.3 MODELL 2: ZWEI VERMÖGENSWERTE - ZEITLICH VARIIERENDER ZINSSATZ UND EIGENKAPITALRENDITE

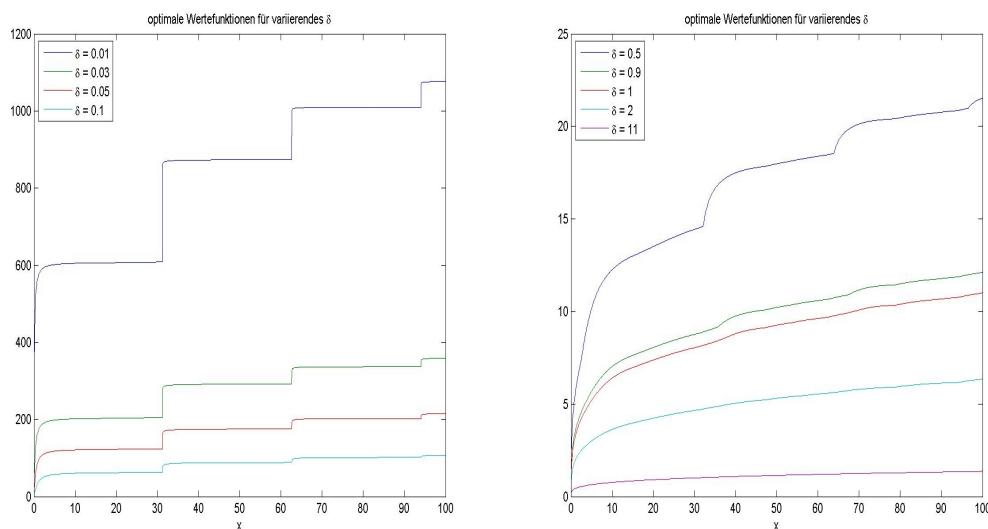


Abbildung 5.28: Die Grafik zeigt die optimalen Wertefunktionen für variierende Diskontraten.

Im Folgenden soll eine Auswahl der optimalen Kontrollfunktionen bzw. der zugehörigen Konsumneigung genügen, um den veränderten Verlauf der Funktionen in Abhängigkeit der Diskontrate zu analysieren. Bis zu einer Diskontrate $\delta < 0.5$ lässt sich ausschließlich eine zunehmende Höhe der optimalen Kontrolle feststellen, weshalb wir auf grafische Darstellungen in diesem Bereich verzichten wollen. Das Verschwinden der „Knicke“ der optimalen Wertefunktion bewirkt, dass die optimale Kontrollfunktion mit größer werdender Diskontrate in diesen Bereichen immer seltener abfällt und statt dessen weiter ansteigt (vgl. Abbildung 5.29). Je weiter man nun die Diskontrate erhöht, desto stärker verschwindet die Periodizität, bis sich schließlich ungefähr ab $\delta > 2$ ein linearer Verlauf der optimalen Kontrollfunktion abzeichnet. Entsprechend dem ersten Modell können wir demnach bei der Optimierung über einen kurzen Zeithorizont eine konstante Konsumneigung des Investors erwarten. Abbildung 5.30 (S. 107) zeigt zudem, dass ein Investor stellenweise überhaupt nicht oder nur sehr wenig konsumiert, wenn neben dem Nutzen in naher Zukunft auch derjenige in ferner Zukunft bedeutend ist. Die Allokation des Vermögens bewegt sich auch hier unverändert zwischen den Grenzen -3 und 4.5.

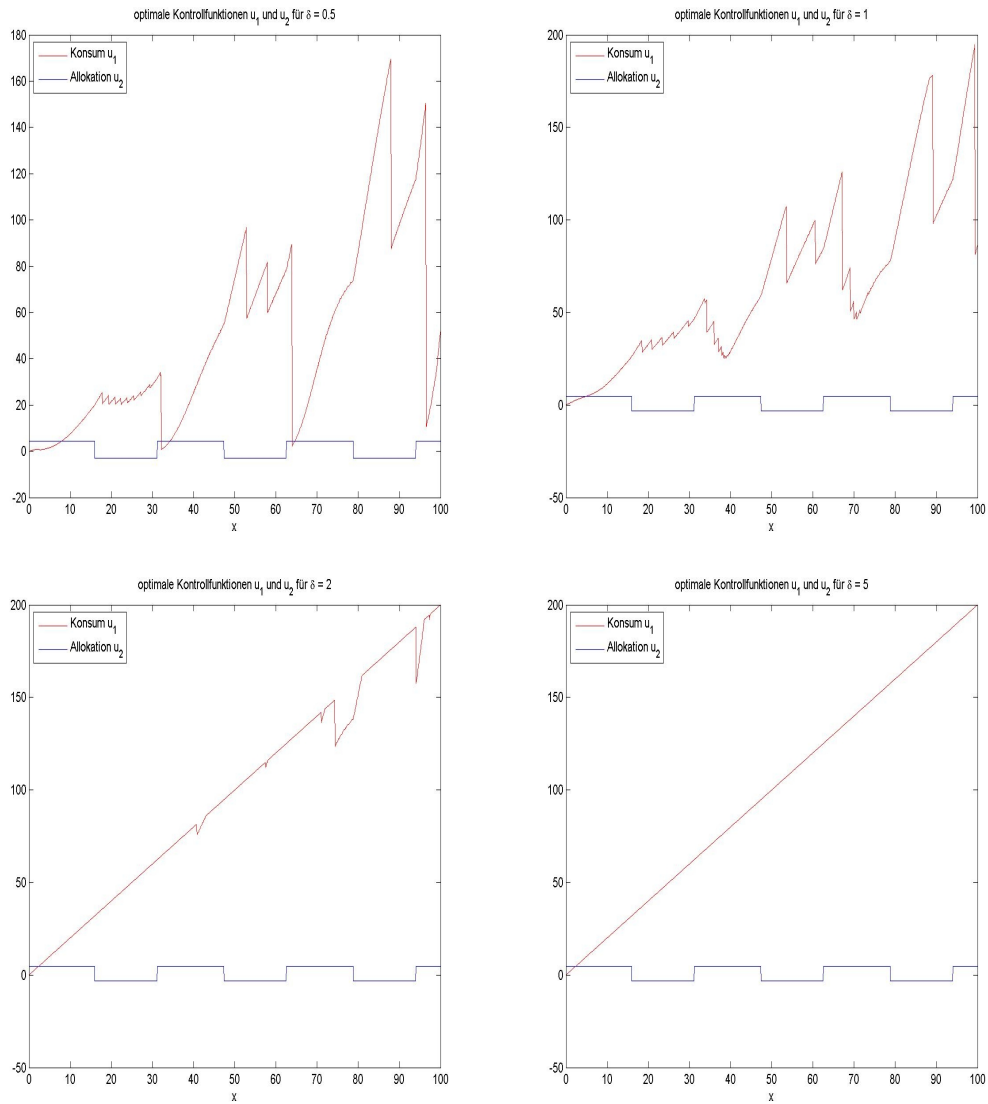


Abbildung 5.29: Die Grafik zeigt die optimalen Kontrollfunktionen für $\delta : 0.5, 1, 2$ und 5 .

5.3 MODELL 2: ZWEI VERMÖGENSWERTE - ZEITLICH VARIIERENDER ZINSSATZ UND EIGENKAPITALRENDITE

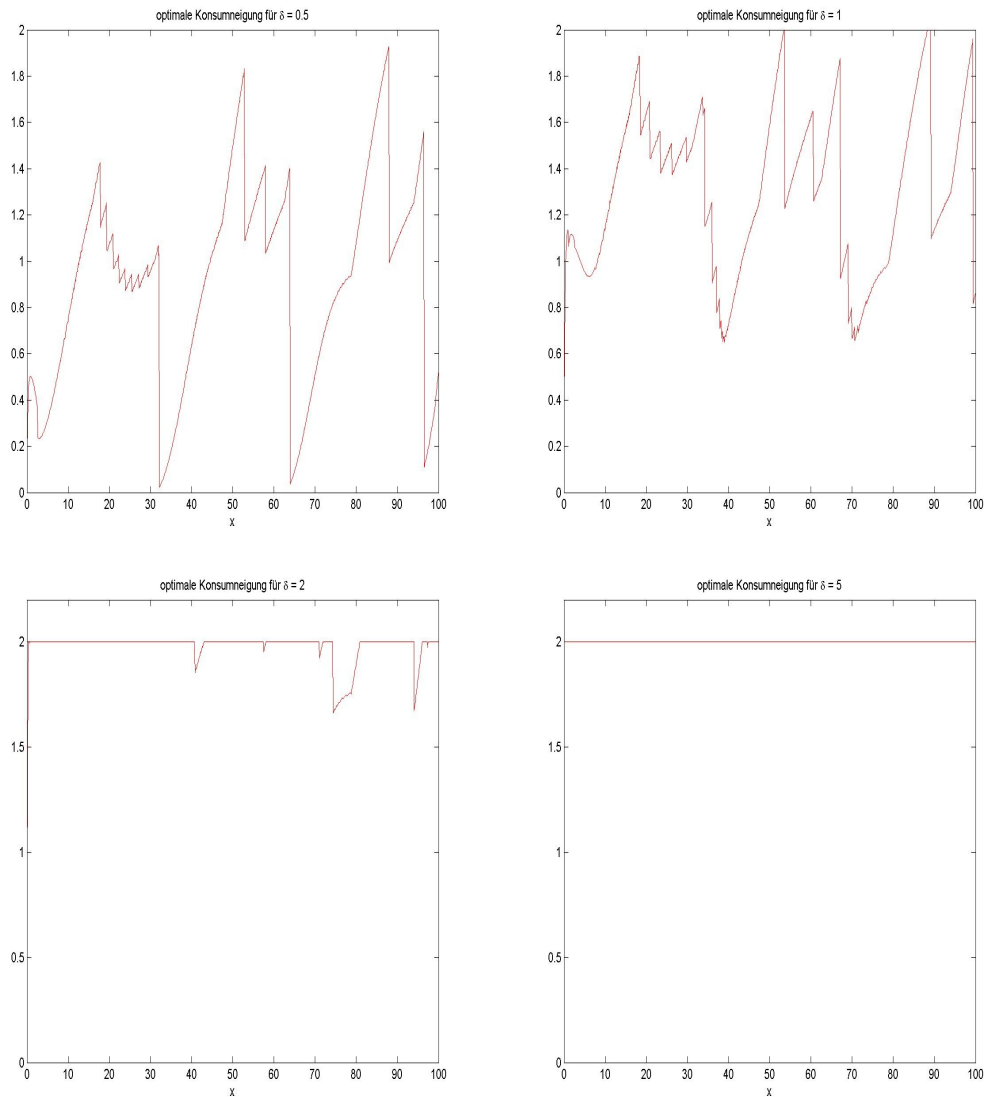


Abbildung 5.30: Die Grafik zeigt die optimalen Konsumneigungen für $\delta : 0.5, 1, 2$ und 5 .

Abschließend veranschaulicht Abbildung 5.31 das globale Verhalten der optimalen Trajektorien. Ab einer Diskontrate von $\delta \geq 0.1$ tritt im unteren Bereich ein weiterer Gleichgewichtspunkt auf. Da der Einzugsbereich recht klein ist, wird der grüne Kreis durch den roten überdeckt. Deutlich sichtbar wird der Gleichgewichtspunkt für $\delta = 0.5$. Ursache für dieses optimale Gleichgewicht ist ein weiterer „Knick“ der optimalen Wertefunktion, welcher allerdings nur sehr schwer zu erkennen ist. Bei erneutem Betrachten der optimalen Kontrollfunktion bzw. der Konsumneigung wird ein weiterer kurzzeitiger Anstieg in diesem Bereich erkennbar. Verkleinert man nun den Zeithorizont, verschwindet das zweite Gleichgewicht im Punkt $x^* \approx 10$, bis letztendlich alle optimalen Lösungen gegen 0 streben. Die Trajektorien verhalten sich nun wie im ersten Portfolio-Modell.

Zusammenfassend wird es für einen Investor reizvoller, das Vermögen anzulegen anstatt zu konsumieren, wenn der effektive Optimierungshorizont groß gewählt wird. Durch die in diesem Zusammenhang auftretenden Gleichgewichte ist der Spielraum, in welchem sich Vermögen und Konsum bewegen können, eingeschränkt. Optimiert man hingegen über einen kurzen Horizont, steigt die Konsumneigung an und das Vermögen bzw. der Konsum konvergiert gegen 0.

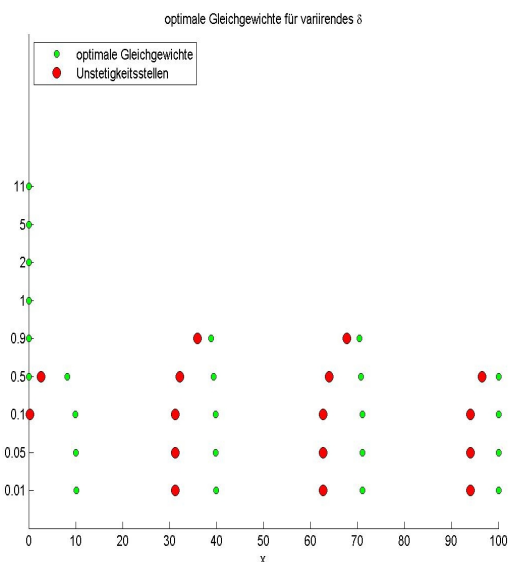


Abbildung 5.31: Die Grafik zeigt die optimalen Gleichgewichte für $\delta \in [0.01, 11]$.

5.4 Modell 3: Erweiterung des zweiten Modells auf den zeitlich - periodischen Fall

5.4.1 Einführung

Abschließend wollen wir eine Erweiterung des zweiten Portfolio-Modells auf den zeitlich periodischen Fall analysieren. Das heißt, anstatt die Renditen in Abhängigkeit des Vermögens zu modellieren, werden sie nun abhängig von der Zeit t periodisch dargestellt.

Analog zum vorherigen Beispiel-Modell liegen einem Investor zwei Vermögenswerte vor, wobei der zeitlich variierende Zinssatz des risikolosen Assets dementsprechend wieder mit $R_{f,t}$ und die sich zeitlich ändernde Eigenkapitalrendite mit $R_{e,t}$ bezeichnet werden. Allerdings werden diese nun mithilfe der Sinus-Funktion abhängig von der Zeit $t := x_2$ dargestellt, so dass wir als zeitlich variierenden Zinssatz

$$R_{f,t}(x_2) = \alpha_1 \sin(\alpha_3 x_2)$$

und für die Eigenkapitalrendite

$$R_{e,t}(x_2) = \alpha_2 \sin(\alpha_4 x_2) + \alpha_5.$$

erhalten.

Die Dynamik des Systems ist folglich gegeben durch:

$$\begin{aligned} \dot{x}_1(t) &= u_2(t) R_{e,t} x_1(t) + (1 - u_2(t)) R_{f,t} x_1(t) - u_1(t) \\ &= (u_2(t) (R_{e,t} - R_{f,t}) + R_{f,t}) x_1(t) - u_1(t) \\ \dot{x}_2(t) &= 1. \end{aligned}$$

Unter dieser Erweiterung erhalten wir ein Modell, welches nun auch *zweidimensional* im Zustand $x = (x_1, x_2)^T$ ist. Dabei bezeichne x_1 wie bisher das Vermögen eines Investors. Wie bisher besitzt das Beispiel weiterhin eine zweidimensionale Kontrolle $u = (u_1, u_2)^T \in \mathbf{U}$, so dass ein Anleger zum einen seinen Konsum, zum anderen den Anteil an Vermögen, welcher in das Eigenkapital investiert wird, beeinflussen kann.

Als Kontrollwertebereich wählen wir für die mit dem Vermögen skalierte Kontrolle \tilde{u}_1 die Menge $\widetilde{U}_1 = [0, 0.7]$, diskretisiert mit $q = 701$ äquidistanten Kontrollwerten. Auf diese Weise erhalten wir für $x \in [0, 350]$ Kontrollwerte im Bereich $U_1 = [0, 245]$. Der Bereich für die zweite Kontrolle entspricht den Festlegungen im vorherigen Abschnitt, so dass $U = [0, 245] \times [-3, 4.5]$ gilt.

Zur Auswertung des Modells sind kleine Abänderungen im Programm notwendig. Einerseits müssen wir natürlich die rechten Seiten der Differentialgleichungen in der Routine $rS()$ anpassen, andererseits überlegen, wie die Menge $\Omega = [a_1, b_1] \times [a_2, b_2]$ zu wählen ist. Für das Vermögen x_1 behalten wir dabei das Intervall $[a_1, b_1] = [0, 100]$ bei. Da wir ausschließlich positive Zeiten betrachten, setzen wir $a_2 = 0$. Aufgrund der verwendeten Sinus-Funktionen stimmen die Werte von $R_{f,t}$ bzw. $R_{e,t}$ zu den Zeiten $j \cdot 2\pi$ für $j \in \mathbb{N}_0$ überein. Dies wollen wir für die Wahl des Intervalls $[a_2, b_2]$ ausnutzen, indem wir es so festlegen, dass es genau eine Periode der Funktion $R_{f,t}$ bzw. $R_{e,t}$ umfasst. Überschreitet die Zeit x_2 die obere Grenze b_2 – wird also eine weitere Periode durchlaufen – können wir reduzieren, indem wir an entsprechender Stelle im Programm⁷ $x_2 = x_2 - b_2$ setzen. Im Folgenden sei nun entweder α_3 ein Vielfaches von α_4 oder umgekehrt. Für $\alpha_3 = k \cdot \alpha_4$ bzw. $\alpha_4 = k \cdot \alpha_3$ mit $k \in \mathbb{N}_0$ und $x_2 = b_2$ gilt dann:

$$\alpha_3 x_2 = \alpha_3 b_2 = j_1 2\pi$$

$$\alpha_4 x_2 = \alpha_4 b_2 = j_2 2\pi$$

und somit entweder $j_1 = 1, j_2 = k$ oder $j_2 = 1, j_1 = k$.

Falls $j_1 = 1$ gilt, folgt für b_2

$$b_2 = \frac{2\pi}{\alpha_3},$$

falls hingegen $j_2 = 1$ gilt, verwenden wir

$$b_2 = \frac{2\pi}{\alpha_4},$$

womit auch die obere Intervallgrenze für unsere Berechnungen festgelegt ist.

⁷Notwendig in den Routinen *FindeRechteck()* und *FindeRechteckKoord()*.

5.4.2 Diskussion der numerischen Ergebnisse

Die folgenden Abschnitte entsprechen in Aufbau unseren bisherigen Auswertungen, so dass an dieser Stelle zunächst Ausgangswerte für das zugrunde liegende Modell berechnet und grafisch veranschaulicht werden. Die hierfür verwendeten Parameter befinden sich wieder zusammengefaßt in Tabelle 5.9.

Parameter	Verwendeter Wert
Ω ($n[0], n[1]$) ($x[0], x[1]$)	$[0, 350] \times [0, \frac{2\pi}{\alpha_3}]$ bzw. $[0, 350] \times [0, \frac{2\pi}{\alpha_4}]$ (350,30) $(x_1, x_2) \in \Omega$
h	$\frac{1}{12}$
$[t, T]$	$[0, 90]$
N0	8
N1	1
γ	0.75
δ	0.05
α_1	0.1
α_2	0.2
α_3	0.2
α_4	0.2
α_5	0.005

Tabelle 5.9: Parameterwerte zur Auswertung von Modell 2.

Abbildung 5.32 veranschaulicht die approximativ optimale Wertefunktion des Modells. Analog zum ersten Beispiel-Modell weist die Funktion ein konkaves Krümmungsverhalten entsprechend der verwendeten Power-Nutzenfunktion auf. Betrachtet man zudem die zeitliche Komponente x_2 , zeigen sich mehrere, wellenförmige Schwankungen, wobei vor allem eine starke Krümmung im unteren Bereich zu erkennen ist.

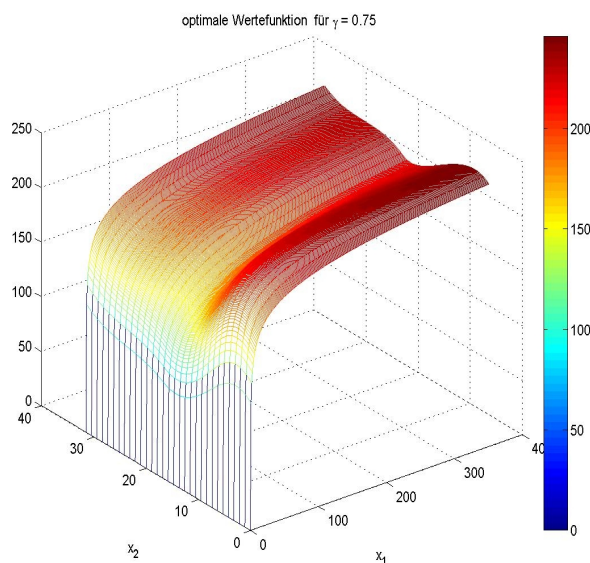


Abbildung 5.32: Die Grafik zeigt die optimale Wertfunktion für $\gamma = 0.75$ und $\delta = 0.05$.

Auch bei der optimalen Kontrollfunktion und der zugehörigen Konsumneigung in Abbildung 5.33 lassen sich Parallelen zum Verlauf der entsprechenden Funktionen im ersten Modell erkennen. Insgesamt ist ein Anstieg der Konsumneigung eines Investors in Abhängigkeit des Vermögens erkennbar. Deutlich wird insbesondere auch hier der wellenförmige Verlauf, wobei der Anstieg zum Rand des Gitters in diesen Bereichen besonders stark ausfällt. Verschiebt man die Gittergrenze, verlagert sich der Anstieg entsprechend. Somit lässt sich dieses Verhalten auf die Einschränkung auf Ω zurückführen und kann daher vernachlässigt werden.

Da sich die zweite Kontrollfunktion u_2 hier ebenfalls periodisch zwischen den festgesetzten Grenzen -3 und 4.5 bewegt, soll sie an dieser Stelle nicht explizit grafisch veranschaulicht werden. Bei Vorliegen einer positiven Eigenkapitalprämie wird das Gesamtvermögen dabei vollständig in das Eigenkapital investiert. Beim Übergang zu einer negativen Prämie legt ein Investor sein Kapital vollständig im risikolosen Asset an, da es in diesem Fall einen höheren Zinssatz aufweist.

5.4 MODELL 3: ERWEITERUNG DES ZWEITEN MODELLS AUF DEN ZEITLICH - PERIODISCHEN FALL

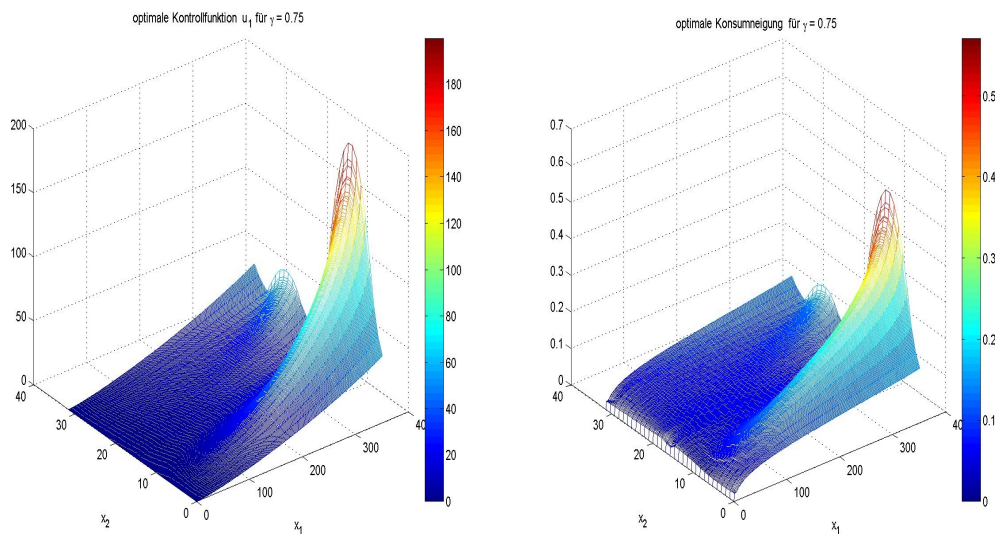


Abbildung 5.33: Die Grafik zeigt links die optimale Kontrollfunktion und rechts die optimale Konsumneigung für $\gamma = 0.75$ und $\delta = 0.05$.

Das globale Verhalten optimaler Trajektorien zeigt Abbildung 5.34. Zur besseren Veranschaulichung wurde dabei das Vektorfeld in zwei Teilbereichen vergrößert dargestellt. Die linke Grafik verdeutlicht das Verhalten im Bereich $[0, 170]$, die rechte im Bereich $[170, 350]$. Zudem beinhaltet die Grafik den Verlauf der optimalen Lösung zum Startwert $x_0 = (1, 0)$. Besonders auffallend ist dabei der Anstieg zum Zeitpunkt einer positiven Eigenkapitalprämie.

Um die optimalen Lösungen über einen längeren Zeitraum auswerten zu können, werden wir diese abschließend in Abhängigkeit von der Zeit plotten. In Abbildung 5.35 (S. 115) erkennt man, dass die optimalen Lösungen periodisch verlaufen, wobei das Vermögen zu Beginn jeder weiteren Periode zunächst abfällt, anschließend aber erneut steigt. Beim Erreichen der maximalen positiven Eigenkapitalprämie liegt der Konsum eines Investors am höchsten. Am Übergang zu einer negativen Prämie erreicht er sein Minimum, woraufhin er erneut ansteigt. Allerdings liegt er in diesen Bereichen deutlich unter den Werten, welche man für positive Prämien erreicht. In jedem Fall wird ein Investor

das Vermögen und dementsprechend auch seinen Konsum möglichst hoch halten. Der obere Verlauf des ansteigenden Vermögens ist in der Abbildung abgeschnitten. Um diesen komplett darstellen zu können, müsste das Gitter mehr als doppelt so groß gewählt werden. Der damit verbundene Rechenaufwand steht allerdings in keiner Relation zum Nutzen, so dass wir uns auf diese Darstellung beschränken wollen. Wegen des starken Anstiegs des Konsums für Kontrollwerte $u_2 = 4.5$ lässt sich aber zumindest ein ähnlich starker Zuwachs an Vermögen vermuten. Wird das Vermögen hingegen im risikolosen Asset angelegt, fällt dieser geringer aus.

Im Folgenden sollen, analog zu unseren bisher betrachteten Modellen, Variationen der Risikoaversion, der Renditen und der Diskontrate berechnet und ausgewertet werden.

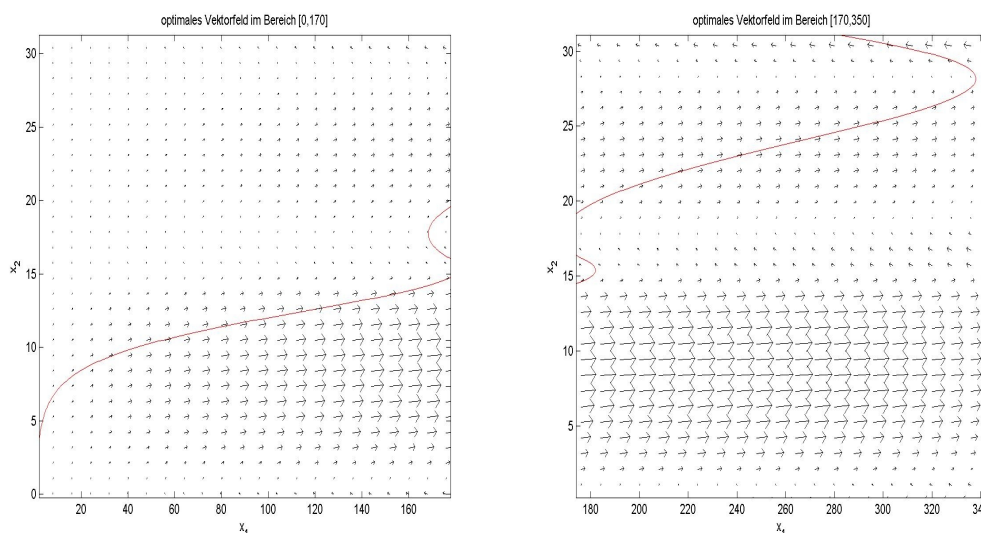


Abbildung 5.34: Die Grafik zeigt das optimale Vektorfeld mit optimaler Trajektorie zum Startwert $x_0 = (1, 0)$ für $\gamma = 0.75$.

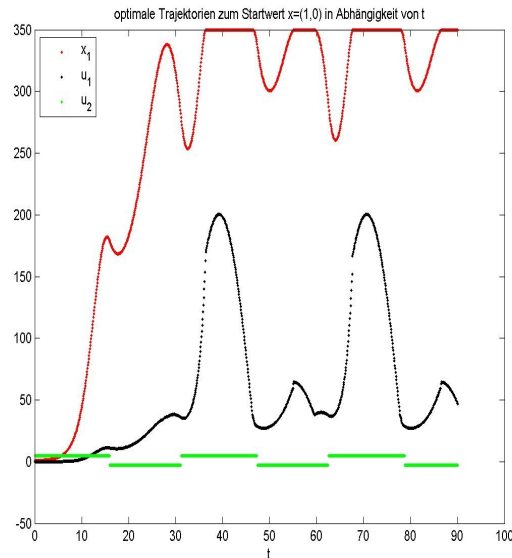


Abbildung 5.35: Die Grafik zeigt optimale Trajektorien zum Startwert $x_0 = (1, 0)$ für $\gamma = 0.75$.

5.4.3 Variationen in den Parametern

Variation der Risikoaversion

Zunächst werden wir Berechnungen für unterschiedliche Risikoaversionen eines Investors durchführen. Dabei verwenden wir für γ die Werte 0.1, 1 und 2.

Abbildung 5.36 veranschaulicht die resultierenden optimalen Wertefunktionen. Der akkumulierte Nutzen bei optimaler Steuerung ist dabei umso größer, je weniger ein Investor einem Risiko abgeneigt ist. Allerdings wird die wellenförmige Krümmung der Funktion für geringe Risikoaversionen deutlich stärker. Ab einem Wert von ungefähr $\gamma = 2$ verschwindet die Krümmung, so dass die Wertefunktion flach verläuft.

Im Folgenden werden wir nun vor allem das Verhalten in diesem Übergangsbereich untersuchen.

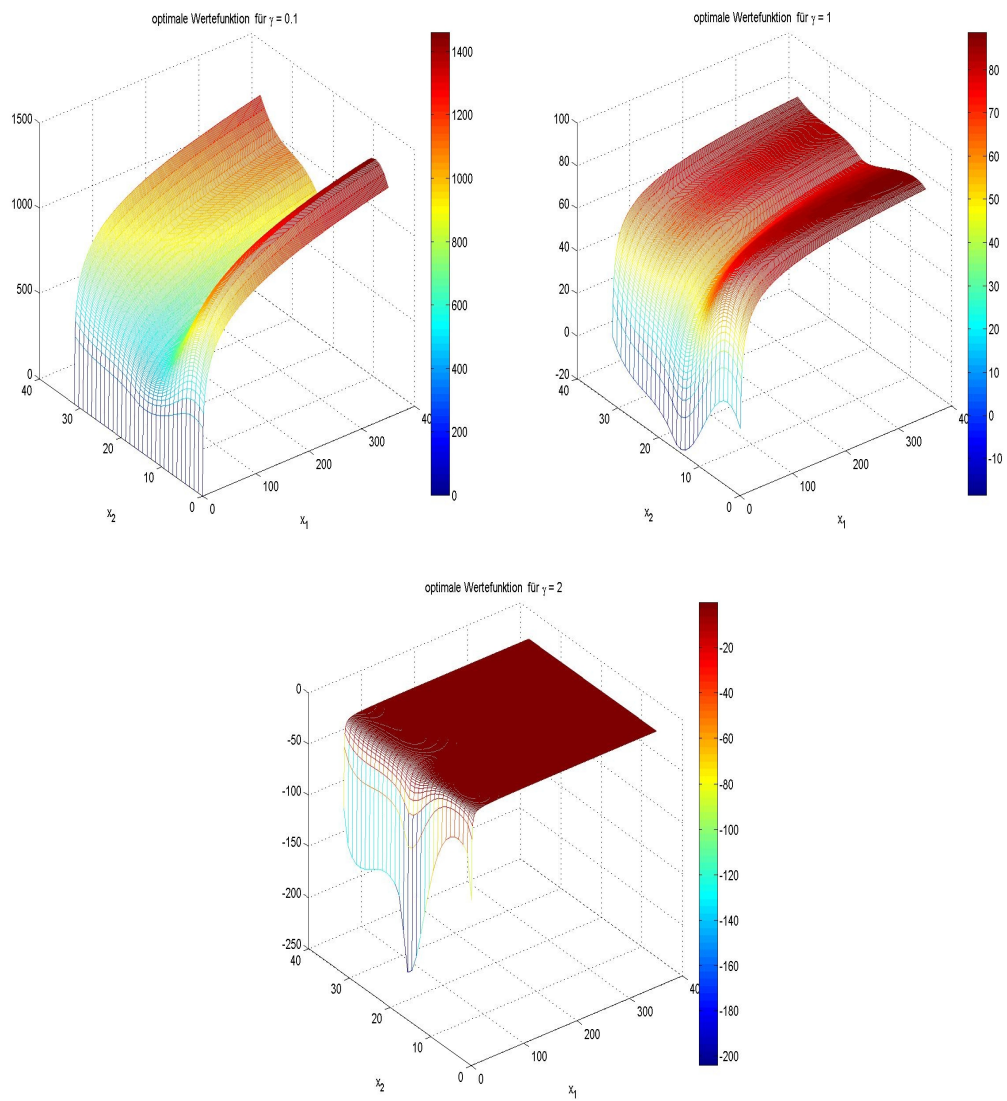


Abbildung 5.36: Die Grafik zeigt die optimalen Wertefunktionen für $\gamma : 0.1, 1$ und 2 .

Um Aussagen über den Konsum eines Anlegers bei variierender Risikoaversion zu erhalten, soll an dieser Stelle die Betrachtung der optimalen Konsumneigung genügen. Betrachten wir die mit dem Vermögen skalierten optimalen Kontrollen für $\gamma : 0.1, 1$ und 2 in Abbildung 5.37 (S. 118), zeigt sich nun im Vergleich zu den vorherigen Modellen, dass ein Investor umso mehr zum Konsum neigt, je höher die Risikoaversion ist. Das würde bedeuten, dass es unter diesen Bedingungen für einen Investor mit geringer Risikoaversion reizvoller ist, sein Vermögen anzulegen, anstatt zu konsumieren.

Zur Analyse des Langzeitverhaltens der optimalen Lösungen werden wir sowohl das optimale Vektorfeld als auch die optimalen Trajektorien in Abhängigkeit von der Zeit parallel darstellen. Abbildung 5.38 (S. 119) veranschaulicht die Resultate.

Je geringer die Risikoaversion eines Investors, desto schneller steigen Vermögen und Konsum zunächst an. Dabei gelingt es dem eher risikofreudigen Anleger, sein Vermögen möglichst hochzuhalten. Je stärker er hingegen einem Risiko abgeneigt ist, desto stärker fällt das Vermögen beim Übergang einer positiven (negativen) zu einer negativen (positiven) Eigenkapitalprämie.

Der optimale Konsum eines Investors verhält sich dem Vermögen entsprechend und erreicht bei positiver Eigenkapitalprämie sein Maximum. Investiert der Anleger hingegen im risikolosen Asset, fällt der Konsum entsprechend niedriger aus. Insgesamt zeigt sich für unterschiedliche Startwerte, dass die Funktionen jeweils unabhängig von der Zeit x_2 für unterschiedliches γ zyklisch verlaufen, wobei das beschriebene Verhalten umso schneller eintritt, je höher das Startkapital gewählt wird.

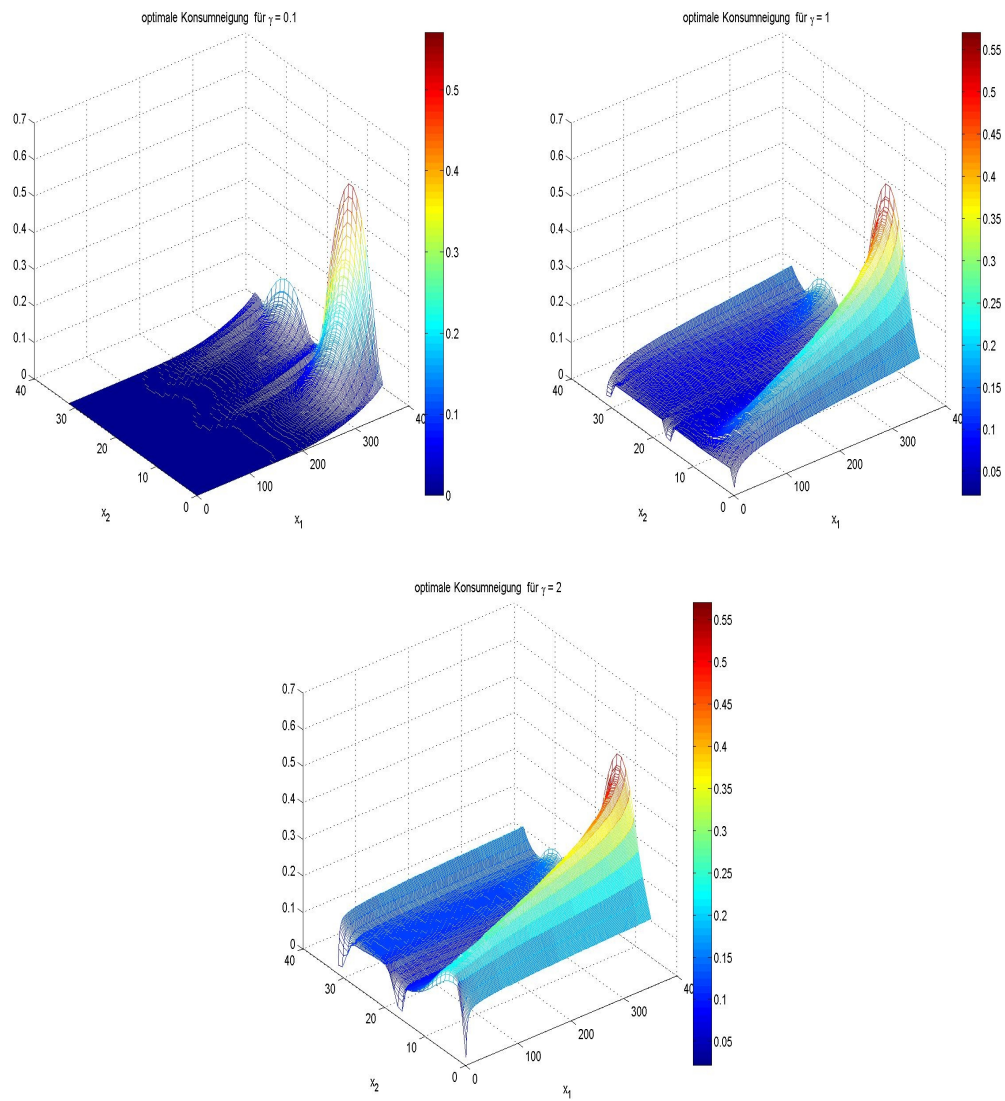


Abbildung 5.37: Die Grafik zeigt die optimalen Konsumneigungen für $\gamma : 0.1, 1$ und 2 .

5.4 MODELL 3: ERWEITERUNG DES ZWEITEN MODELLS AUF DEN ZEITLICH - PERIODISCHEN FALL

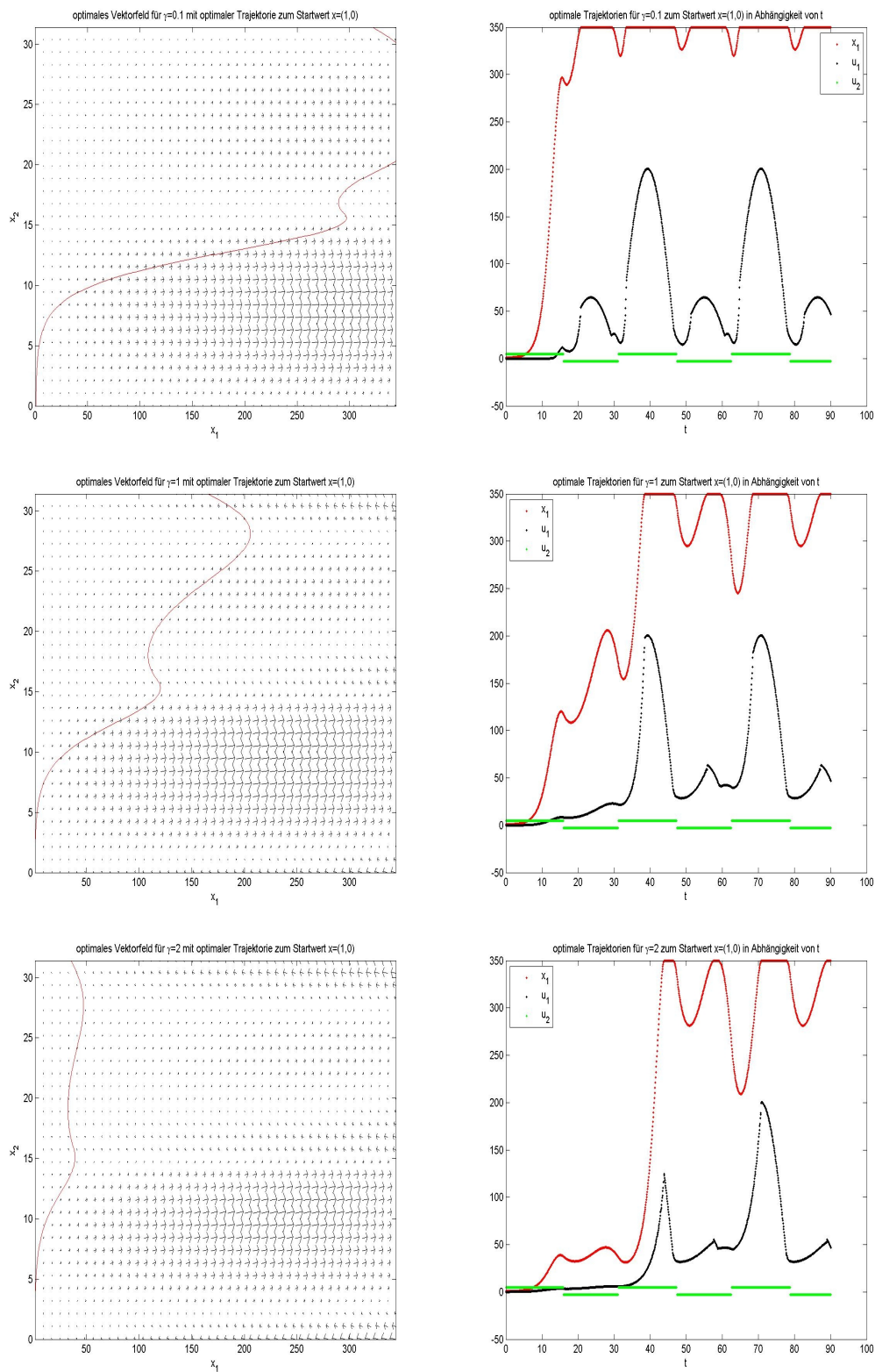


Abbildung 5.38: Die Grafik zeigt links das optimale Vektorfeld und rechts die optimalen Trajektorien für $\gamma : 0.1, 1$ und 2 .

Variation der Renditen

In diesem Abschnitt wollen wir Auswirkungen auf die Lösungen des optimal gesteuerten Systems für unterschiedliche Renditen untersuchen. Analog zum vorherigen Modell werden wir an dieser Stelle die in Tabelle 5.7 (S. 100) aufgeführten Varianten nun auch in Abhängigkeit von der Zeit testen.

Abbildung 5.39 veranschaulicht die approximativ optimalen Wertefunktionen des Modells. Insgesamt erkennt man anhand der ersten beiden Varianten, dass Zinsschwankungen den akkumulierten Nutzen bei optimaler Steuerung nur geringfügig beeinflussen. Allerdings weist die Funktion im Vergleich zur Ausgangsdarstellung einen umso stärkeren, wellenförmigen Verlauf auf, je stärker diese Schwankungen sind. Eine Zunahme der „Wellen“ zeigt sich zudem genau dann, wenn diejenige Funktion stärkeren Schwankungen ausgesetzt ist, welche eine höhere Amplitude aufweist. Dies lässt sich insbesondere beim Vergleich der Ausgangssituation aus Abbildung 5.32 mit der dritten Variante erkennen.

Ein ähnliches Verhalten können wir an dieser Stelle auch für die optimale Kontrollfunktion u_1 bzw. die zugehörige optimale Konsumneigung des Investors erwarten. Da sich auch hier kaum Änderungen in der Höhe der Funktionen ergeben, wollen wir an dieser Stelle auf eine grafische Darstellung verzichten. Zu erwähnen ist allerdings analog zum zweiten Modell der abweichende Verlauf der zweiten Kontrollfunktion. Weist die Rendite des zinslosen Assets sowohl eine geringere Amplitude als auch eine kleinere Periode als die Eigenkapitalrendite auf, wird ein Investor weiterhin sein Vermögen komplett im Eigenkapital anlegen, wenn die Eigenkapitalrendite positiv ist. Schwankt nun die Eigenkapitalrendite stärker als der (geringere) Zinssatz des zweiten Vermögenswertes, verläuft die Rendite $R_{e,t}$ dementsprechend häufiger unter der Funktion für $R_{f,t}$. Aus diesem Grund wird ein Investor hier nun öfters die Allokation seines Vermögens ändern, um seinen Nutzen zu maximieren.

5.4 MODELL 3: ERWEITERUNG DES ZWEITEN MODELLS AUF DEN ZEITLICH - PERIODISCHEN FALL

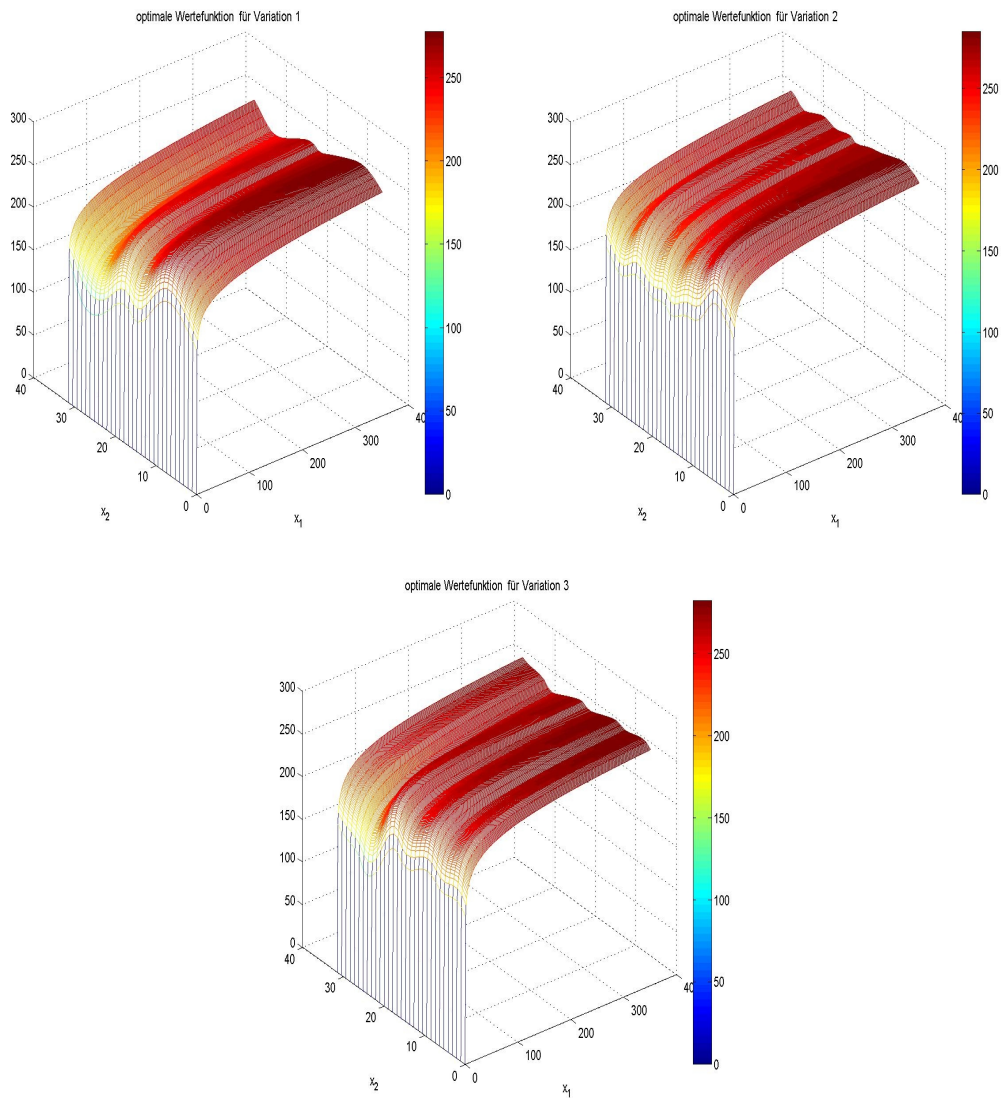


Abbildung 5.39: Die Grafik zeigt die optimalen Wertefunktionen für variierende Renditen $R_{e,t}$ und $R_{f,t}$.

Interessant ist auch das Langzeitverhalten der optimalen Lösungen. Daher werden wir im Folgenden sowohl das optimale Vektorfeld als auch den Verlauf optimaler Trajektorien näher betrachten. Abbildung 5.40 zeigt jeweils links das Vektorfeld mit der optimalen Trajektorie zum Startwert $x_0 = (1, 0)$, rechts den Verlauf des Vermögens bzw. der beiden Kontrollen in Abhängigkeit von der Zeit t .

Betrachten wir zunächst die optimalen Vektorfelder, nehmen diejenigen Bereiche, in denen das Vermögen steigt bzw. fällt umso mehr zu, je stärker die Schwankungen in den Renditen sind. Dabei werden diese Bereiche natürlich entsprechend kleiner, so dass eine stärkere Bewegung im Verlauf der optimalen Trajektorien zu erwarten ist. Eine grafische Veranschaulichung liefern die optimalen Lösungskurven zum Startwert $x_0 = (1, 0)$, welche jeweils in der rechten Grafik dargestellt sind. Beim Vergleich der beiden ersten Varianten wird das Vermögen mit zunehmenden Schwankungen zum einen umso höher gehalten, zum anderen steigt bzw. fällt es aber auch umso häufiger. Beim Übergang zu einer positiven Eigenkapitalrendite findet man dabei jeweils den stärksten Abfall des Vermögens. Dieser verschwindet hingegen bei der dritten Variation, so dass in diesem Fall das Vermögen am höchsten gehalten werden kann. Angesichts der im oberen Bereich abgeschnittenen Funktionen lässt sich an dieser Stelle allerdings nur vermuten, dass die Amplitude der Vermögensschwankungen beim dritten Fall am geringsten sein müsste. Der zugehörige Konsum eines Investors verhält sich auch hier dem Vermögen entsprechend, wobei die obere und untere Grenze nahezu gleich zu bleiben scheinen. Analog bewegt sich auch die zweite Kontrolle zwischen den festgesetzten Grenzen, weist aber im Vergleich zu unseren Ausgangswerten zunehmend mehr Übergänge auf.

5.4 MODELL 3: ERWEITERUNG DES ZWEITEN MODELLS AUF DEN ZEITLICH - PERIODISCHEN FALL

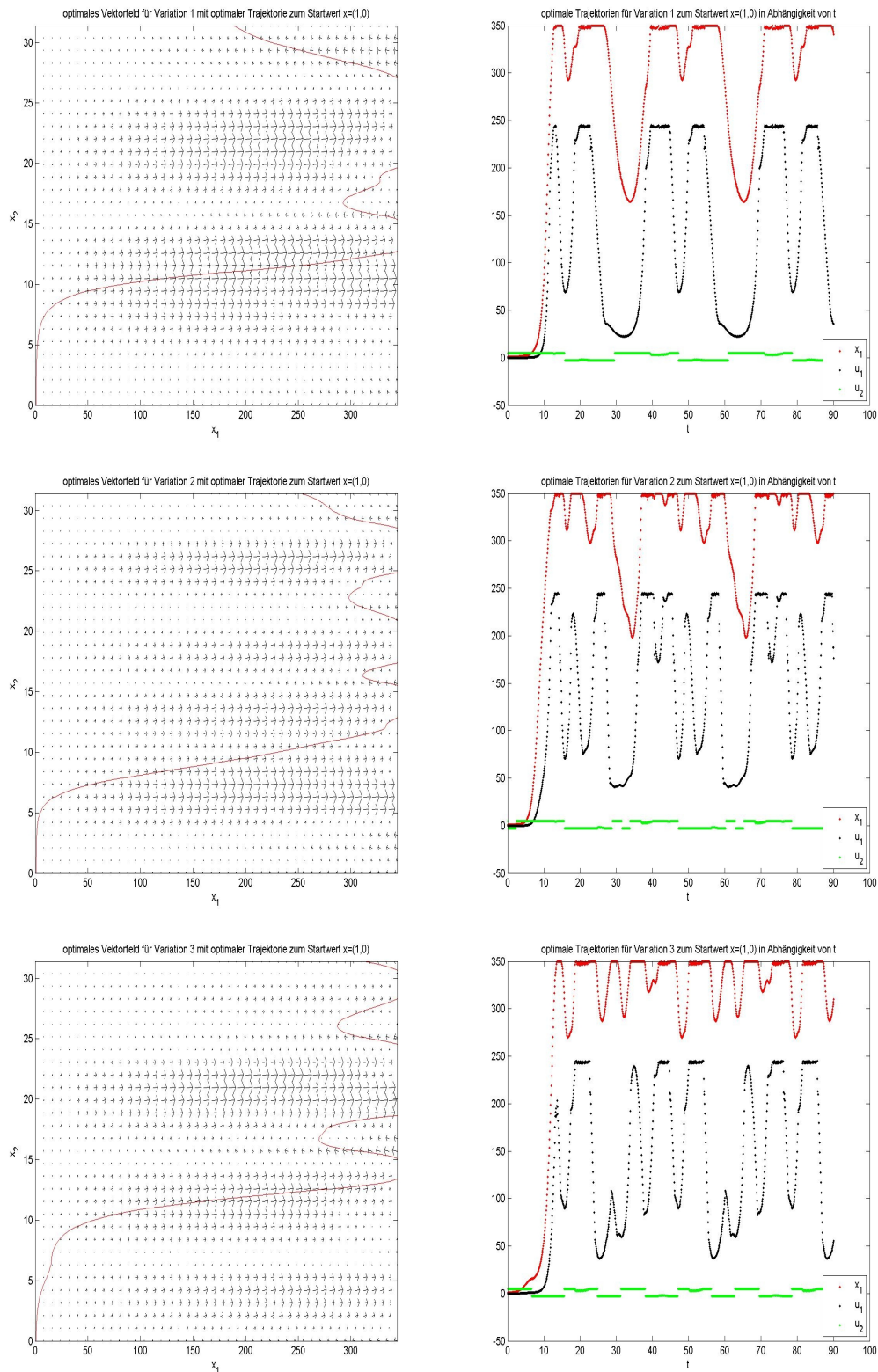


Abbildung 5.40: Die Grafik zeigt das optimale Vektorfeld und optimale Trajektorien für variierende Renditen $R_{e,t}$ und $R_{f,t}$.

Variation der Diskontrate

Abschließend soll untersucht werden, wie sich die Lösungen des Modells in Abhängigkeit des effektiven Optimierungshorizonts verhalten. Um diesen zu verändern, variieren wir die Diskontrate δ entsprechend den vorherigen Modellen. Dabei verwenden wir an dieser Stelle jeweils eine geeignete Auswahl aus dem Intervall $[0.1, 11]$, welche wir mit den Auswertungen für $\delta = 0.05$ in Beziehung setzen.

Auffallend bei der Betrachtung der approximativ optimalen Wertefunktionen in Abbildung 5.41 (S. 126) ist der umso stärker abnehmende wellenförmige Verlauf, je kleiner der betrachtete Zeithorizont wird. Analog zum zweiten Modell nähert sich dabei der Verlauf der optimalen Wertefunktion für sehr große Diskonraten dem Verlauf der Power-Nutzenfunktion an. Bei der Optimierung über einen kurzen Horizont überträgt sich demnach die konkave Krümmung der Nutzenfunktion zu allen Zeiten x_2 fast direkt auf die optimale Wertefunktion. Optimiert man hingegen über den unendlichen Zeithorizont, werden die periodischen Effekte der Dynamik durch den langen Zeitraum viel stärker sichtbar. Im Hinblick auf die Trajektorien bedeutet das wieder, dass bei kurzem Horizont die Lösung das Durchlaufen der „Täler“ der Dynamik vermeidet, bei längerem Horizont kann sich das aber lohnen, wenn man dafür später in einem Bereich mit höherem Nutzen landet. Desweiteren kann man erkennen, dass der akkumulierte Nutzen bei optimaler Steuerung umso geringer ist, je kleiner der betrachtete Zeithorizont wird.

Betrachtet man die optimalen Kontrollfunktionen bzw. die zugehörige Konsumneigung aus Abbildung 5.42 (S. 127), erkennt man, dass die optimale Kontrolle auch hier bis zu einer Diskontrate $\delta < 0.5$ ausschließlich in der Höhe zunimmt. Aus diesem Grund soll dieser Bereich in der grafischen Darstellung ausgespart werden. Je weiter man nun die Diskontrate erhöht, desto stärker verschwinden die Krümmungen, bis sich schließlich ungefähr ab $\delta \geq 1$ ein linearer Verlauf der optimalen Kontrollfunktion abzeichnet. Demnach können wir auch bei diesem Modell bei der Optimierung über einen kurzen Zeithorizont eine höhere und schließlich konstante Konsumneigung des Investors erwar-

ten. Die zweite Kontrolle u_2 hingegen bleibt unverändert und bewegt sich zwischen den festgesetzten Grenzen -3 und 4.5.

Abschließend veranschaulicht Abbildung 5.43 (S. 128) das globale Verhalten der optimalen Lösungen und speziell den Verlauf der Lösungen in Abhängigkeit von t für den Startwert $x_0 = (1, 0)$. Bei der Optimierung über einen kurzen Zeithorizont konvergieren die Lösungen nun gegen 0. Dabei verschwinden die Krümmungen, je höher man die Diskontrate wählt, so dass ab ungefähr $\delta \geq 1$ der Verlauf mit unserem ersten Beispielmodell identisch ist. Um das Verhalten geeignet darzustellen, haben wir an dieser Stelle neben dem Startwert $x_0 = (1, 0)$ für höhere Diskonraten den Wert $x_0 = (300, 0)$ betrachtet, sowie den Betrachtungszeitraum $[t, T]$ verkleinert. Dadurch scheint die zweite Kontrolle u_2 verändert, was natürlich nicht der Fall ist.

Zusammenfassend empfiehlt es sich auch hier, den effektiven Optimierungshorizont möglichst groß zu wählen, da auf diese Weise bei optimaler Steuerung ein höherer Nutzen erzielt werden kann.

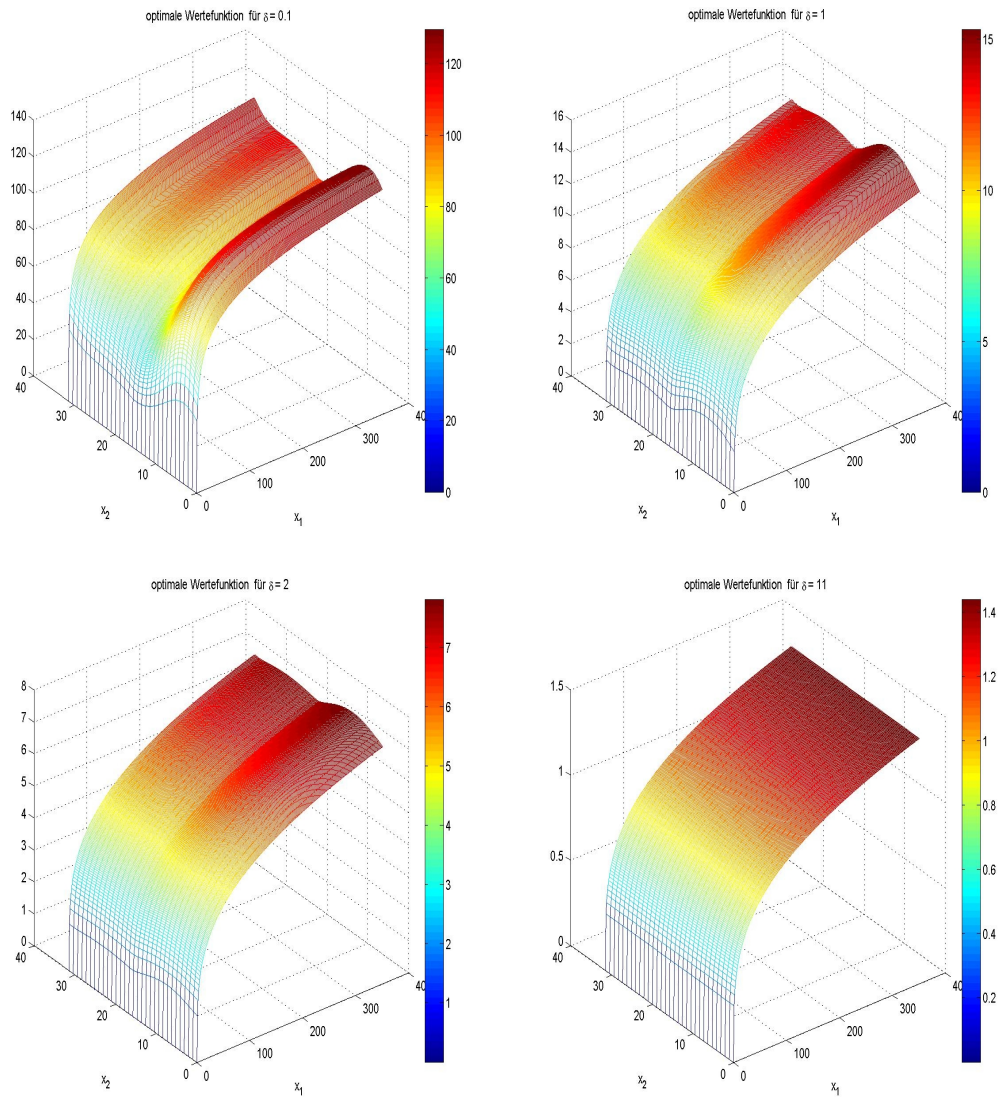


Abbildung 5.41: Die Grafik zeigt die optimalen Wertefunktionen für $\delta : 0.1, 1, 2$ und 11 .

5.4 MODELL 3: ERWEITERUNG DES ZWEITEN MODELLS AUF DEN ZEITLICH - PERIODISCHEN FALL

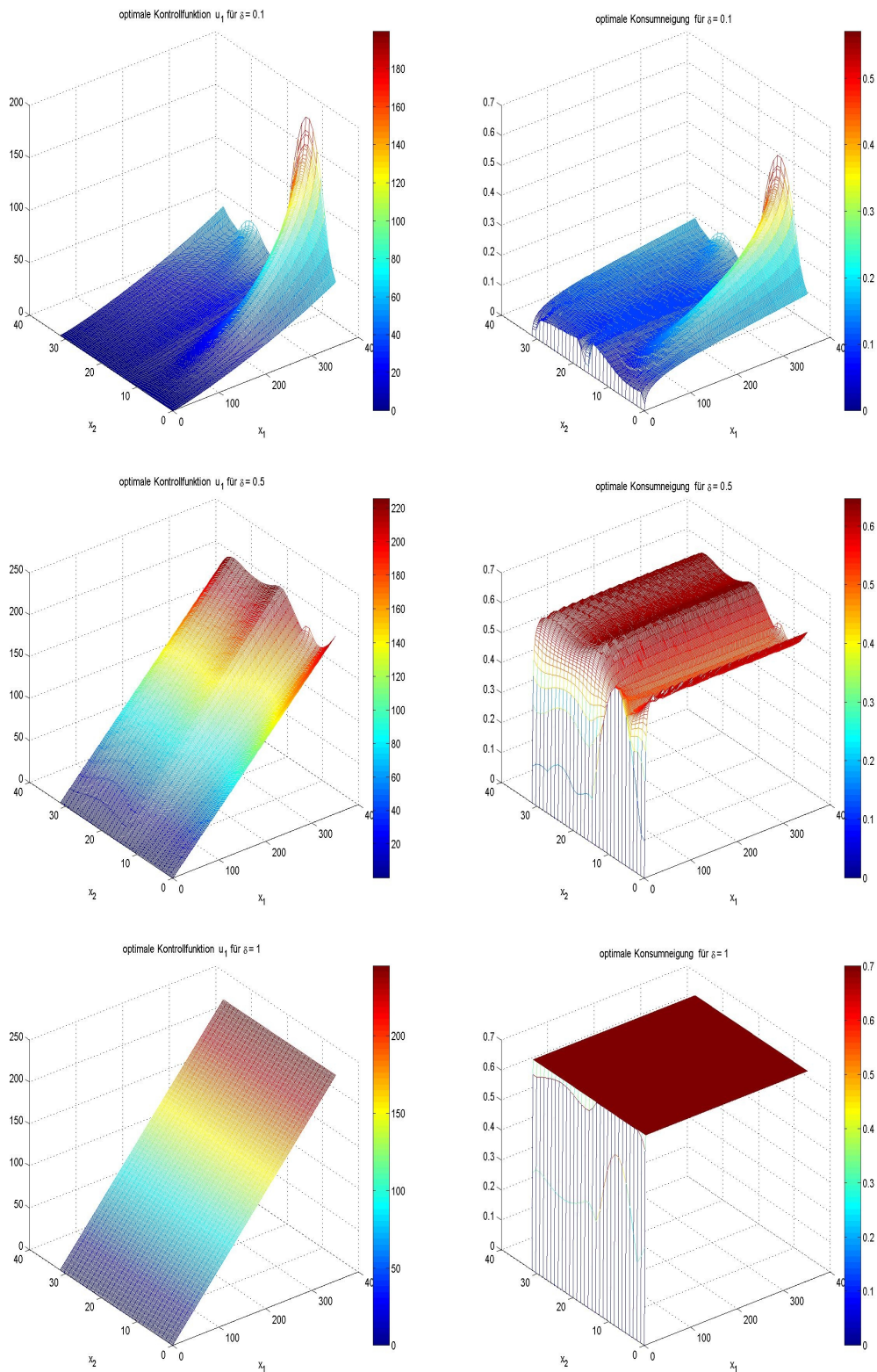


Abbildung 5.42: Die Grafik zeigt links die optimalen Kontrollfunktionen und rechts die zugehörige Konsumneigung für $\delta : 0.1, 0.5$ und 1 .

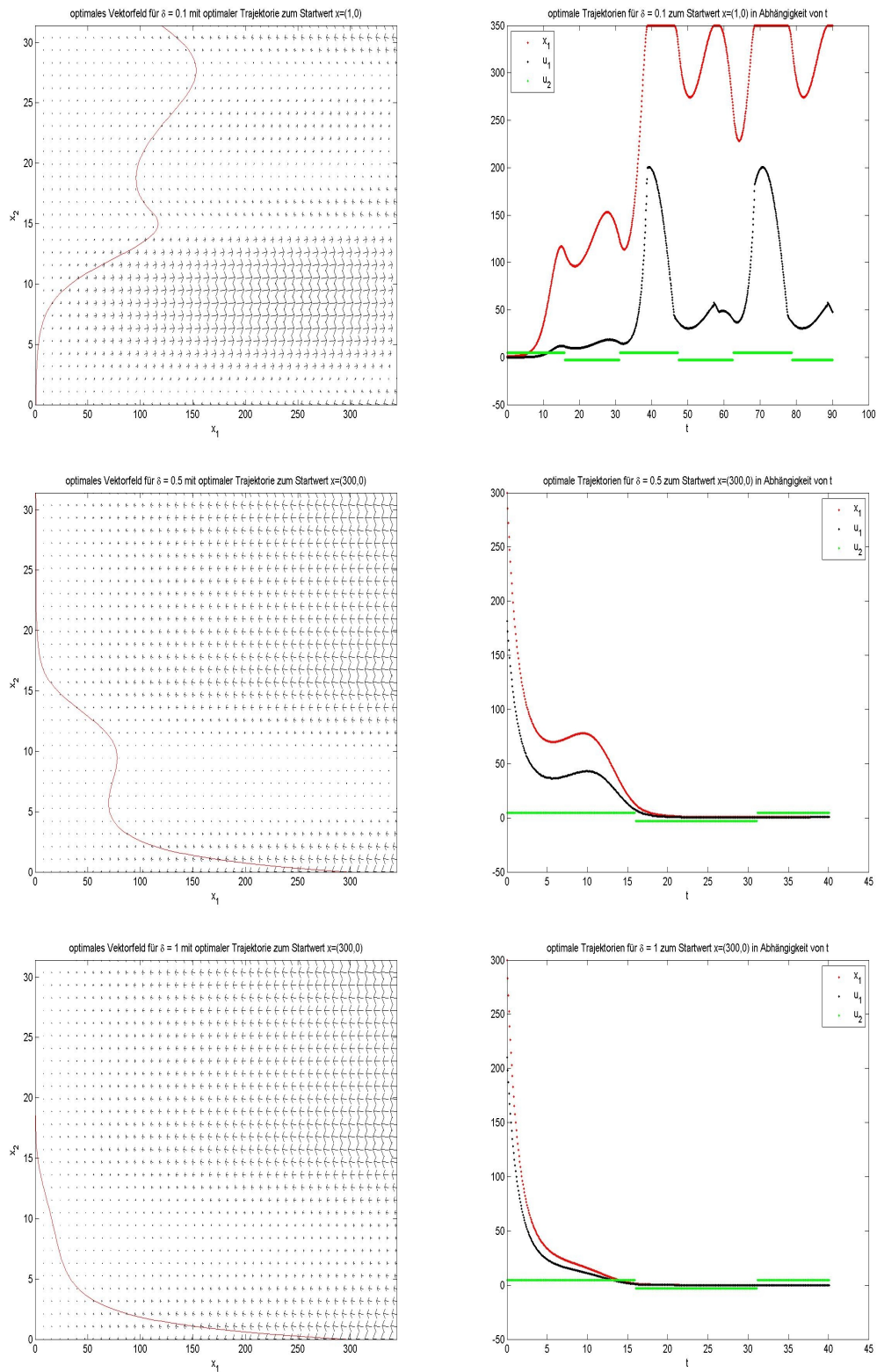


Abbildung 5.43: Die Grafik zeigt links die optimalen Vektorfelder und rechts den Verlauf der optimalen Trajektorien für $\delta : 0.1, 0.5$ und 1 .

6 Fazit der praktischen Ergebnisse und Ausblick

An dieser Stelle soll das Ergebnis der praktischen Auswertungen noch einmal zusammengefasst und ein Ausblick auf Verbesserungsmöglichkeiten gegeben werden.

Bei der Auswertung der Beispiel-Modelle in Kapitel 5 haben wir uns ausschließlich auf die Modellierung der Grundlagen eines Portfolios beschränkt. Trotz der sehr einfachen Modelle zeigt sich allerdings insbesondere in den Erweiterungen ein sehr komplexes Verhalten. Vor allem der Übergang von einer konstanten und daher risikolosen Rendite zu zeitlich variierenden Parametern verursacht dabei eine enorme Wirkung auf das analysierte Langzeitverhalten der optimalen Lösungen. Während bei Variationen im ersten Beispiel-Modell maximal ein Gleichgewicht existiert, nimmt die Anzahl der Gleichgewichtspunkte im zweiten Modell so stark zu, dass sich in vielen Bereichen aufgrund der recht kleinen Einzugsbereiche kaum Veränderungen in der Höhe der optimalen Lösungen zeigen. In diesem Zusammenhang sind vor allem auch die Bifurkationsszenarien zu erwähnen, welche zum Beispiel bereits im ersten Modell für unterschiedliche Renditen bzw. variierende Diskontraten aufgetreten sind. Interessant ist hierbei auch der Verlauf der Konsumneigung eines Investors, wobei sich mit Ausnahme des dritten Modells gezeigt hat, dass diese umso geringer ist, je höher die Risikoaversion des Anlegers ist. Um den Nutzen zu maximieren, wird es für Investoren in diesem Fall umso reizvoller, das Vermögen anzulegen anstatt zu konsumieren. Dieses Resultat erhalten wir auch ab einer bestimmten Vermögensverzinsung, sowie bei einem unendlichen Optimierungshorizont. In diesem Zusammenhang konnten wir auch eine stark variierende Konsumneigung in Abhängigkeit der Zinsschwankungen beobachten.

Da Zinssätze und Aktienkurse von Unsicherheit geprägt sind, werden im Zusammenhang mit Portfolio-Optimierungen hauptsächlich stochastische Modelle untersucht, so dass eine weitere nahe liegende Erweiterungsmöglichkeit nun sicherlich darin besteht, den Betrachtungen ein stochastisches Rahmenwerk anstelle eines deterministischen zugrunde zu legen. Auf diese Weise folgen die verwendeten Variablen, wie zum Beispiel die Rendite eines Vermögenswertes, einer stochastischen Dynamik, sind also von Unsicherheit gekennzeichnet. SEMMLER [15] bietet auch hier einige Ansätze. Die Methode der stochastischen Steuerung geht auf MERTON zurück und bildet unter anderem die Basis für Methoden zur Behandlung des Portfolio-Problems unter Transaktionskosten. Ein weiterer interessanter Ansatzpunkt wäre zum Beispiel auch die Portfolio-Optimierung unter Benchmarks und Liabilities.

Die Frage, welche Ergänzungen notwendig sind, bzw. wie sie unter Verwendung der dynamischen Programmierung sinnvoll umsetzbar sein können, bietet Platz für weitere Untersuchungen.

A Notation

An dieser Stelle werden die wichtigsten Variablenbezeichnungen und Notationen übersichtsartig aufgeführt. Die Reihenfolge entspricht dabei dem erstmaligen Auftreten im Text.

g	Ziel- bzw. Nutzenfunktion
ARA	absolute Risikoaversion
RRA, γ	relative Risikoaversion
$x(t)$	Zustand zum Zeitpunkt t
\mathbf{T}	Zeitachse
h	Schrittweite
\mathbf{U}	Kontrollwertebereich
$\mathcal{U}, \mathcal{U}_h$	Raum der zulässigen Kontrollfunktionen
$u(t)$	Kontrolle in kontinuierlicher Zeit
$u_h(t)$	Kontrolle zum Zeitpunkt t in diskreter Zeit
f	Kontrollsystem in kontinuierlicher Zeit
f_h	Kontrollsystem in diskreter Zeit
Φ	Trajektorie in kontinuierlicher Zeit
Φ_h	Trajektorie in diskreter Zeit
J	diskontiertes Funktional in kontinuierlicher Zeit
δ	Diskontrate in kontinuierlicher Zeit
J_h	diskontiertes Funktional in diskreter Zeit
β^j	Diskontrate in diskreter Zeit
$v(x)$	optimale Wertefunktion in kontinuierlicher Zeit
$v_h(x)$	optimale Wertefunktion in diskreter Zeit
$\tilde{\Phi}_h$	Lösung eines mit Euler diskretisierten Kontrollsystems
\tilde{J}_h	zur Euler-Diskretisierung gehöriges diskontiertes Funktional
\tilde{v}_h	zur Euler-Diskretisierung gehörige optimale Wertefunktion

v_h^i	optimale Wertefunktion im Laufe des Iterationsverfahrens nach der i -ten Iteration
T_h	Operator zur Berechnung der v_h^i
Ω	das untersuchte Intervall
Γ	(regelmäßiges) Rechteckgitter auf Ω
\mathcal{W}	Raum der stetigen, stückweise affin bilinearen Funktionen auf Ω bzgl. Γ
μ	bilineare Koordinaten
\hat{u}_h	Approximation der Kontrolle u
\hat{v}_h	Approximation der optimalen Wertefunktion
\hat{v}_h^j	An den Eckpunkten des Gitters approximierte Wertefunktion nach der j -ten Iteration aus \mathcal{W}
V^j	Vektor der Werte von \hat{v}_h^j an den Knoten
$u^*(x) = u_x$	Kontrollstrategie abhängig vom aktuellen Zustand x
u_h^x	diskrete Kontrollfunktion abhängig vom aktuellen Zustand x
$\hat{u}^*(x) = u_x$	numerische Kontrollstrategie abhängig vom aktuellen Zustand x
\hat{u}_h^x	approximative diskrete Kontrollfunktion abhängig vom aktuellen Zustand x
\hat{u}^x	stückweise konstante (messbare) kontinuierliche Kontrollfunktion
$S(V^{j+1})_i$	Iterationsoperator
\tilde{u}^j	zum Vektor V^j gewählter Kontrollvektor
x^*	optimales Gleichgewicht
$\mathcal{N}(x^*)$	Umgebung des optimalen Gleichgewichts x^*
$\mathcal{D}(x^*)$	Einzugsbereich des Gleichgewichts x^*
r	Rendite
v_t	partielle Ableitung von v nach t
v_x	partielle Ableitung von v nach x
\tilde{u}	Konsumneigung eines Investors
$R_{f,t}$	zeitlich variierender Zinssatz des risikolosen Asset
$R_{e,t}$	zeitlich variierende Eigenkapitalrendite

B Finanzmathematische Grundbegriffe

In einer kurzen Übersicht werden die elementaren finanzmathematischen Grundbegriffe alphabetisch aufgeführt.

Asset	Englische Bezeichnung für Vermögenswerte (Aktiva). Gegensatz: Liabilities (Verbindlichkeiten)
Benchmarks	Die Entwicklung eines Fonds wird in Relation zu einem Vergleichsindex (der Benchmark, auch Vergleichsmassstab genannt) gesetzt. Aktives Fondsmanagement ist darauf ausgerichtet, diesen Vergleichsindex zu schlagen, passiv gemanagte Fonds haben zum Ziel, sich in etwa so wie der Vergleichsmassstab zu entwickeln.
Bond	Engl. Bezeichnung für festverzinsliche Anleihen.
Liabilities	Liabilities ist eine Bezeichnung für Verbindlichkeiten (Passiva). Gegensatz: Assets (Vermögenswerte)
Portfolio	Teil oder Gesamtheit der Anlage in Wertpapieren, die ein Kunde oder ein Unternehmen besitzt (Wertpapierbestand). Ein Portefeuille dient primär dem Zweck der Risikostreuung.
Rendite	Gesamterfolg einer Kapitalanlage, gemessen als tatsächliche Verzinsung (Effektivverzinsung) des eingesetzten Kapitals. Die Rendite wird zumeist in Prozenten angegeben.
Risikomaße	Um zu einer ausgewogenen Anlageentscheidung zu kommen, benötigen Investoren Entscheidungshilfen, die ihnen Informationen über das Verhältnis von Chancen und Risiko bestimmter Anlageformen geben. Zur Beurteilung von Aktien wurden verschiedene Kennzahlen entwickelt, die zusammengefasst als Risikomaße bezeichnet werden.
Stock	Angelsächsische Bezeichnung für das Grundkapital einer Gesellschaft. Der Begriff steht allgemein für Effekten, u.a. Aktien und Anleihen .

C C++-Quelltext

```
#include <cstdio>
#include <cstdlib>
#include <iostream>
#include <fstream>
#include <cmath>
#include <new>
#include <string>
using namespace std;

int const d=2;

void speichervek(double * (&vek), long dim, char * name)
{
    long i;
    if ((vek=new (nothrow) double [dim])==NULL)
    {
        cout << "Kein Speicherplatz mehr da für den Zeiger %s!"
             << name << endl;
        exit(1);
    }
    else
    for(i=0; i<dim; i++)
        vek[i]=0.0;
}
```

```
void speichermat(double ** (&mat), long dim1, long dim2,
                char * name)
{
    long i;
    if((mat=new (nothrow) double *[dim1])==NULL)
    {
        cout << "Kein Speicherplatz mehr da für den Zeiger %s!"
              << name << endl;
        exit(1);
    }
    else
    for(i=0; i<dim1; i++)
        speichervek(mat[i], dim2, name);
}

int Gitter2d(int i, int j, int *n)
{
    /* i: Rechteckindex
       j: lokaler Eckenindex
       n: Vektor der Länge 2, n[i]: Anzahl der Rechtecke in
          Richtung i
       Rückgabe: globaler Eckenindex l
                = lokaler Eckenindex j
                + Rechteckindex n[0]-Richtung i
                + Rechteckindex n[1]-Richtung i/n0 (Ganzzahl)
       obere Ecke: zusätzlich n[0]-1.
    */
}
```

```

int l;
if (j<0 || j>3)
{
    cout << "Falsche Angabe der lokalen Ecke, j aus {0,1,2,3}"
        << endl;
    return (-1);
}
if (i<0 || i>n[0]*n[1]-1)
{
    cout << "Falsche Angabe der Rechtecknummer, i aus {0,...,"
        << n[0]*n[1]-1 << "}" << endl;
    return (-1);
}
l=j+i+i/n[0];
if (j>1) // "obere" Ecken
    l=l+n[0]-1;

return (l);
}

```

```

int FindeRechteck(double *x, double *a, double *b, int *n)
{
    /* Berechnung des Rechteckindex (Rückgabewert) für einen
    Vektor x.
    *x: Vektor der Länge 2 des Punktes x.
    Transformation von Omega = [a[0],b[0]] x [a[1],b[1]]
    auf das Rechteck [0,n[0]] x [0,n[1]].
    Die Werte stehen in x0 und x1.
    Die ganzzahligen Werte stehen im Spaltenindex s (für x0)

```

und Zeilenindex z (für x_1)
 Der untere Rand eines Rechtecks gehört zum Rechteck.
 Wird der obere Rand des Gebietes erreicht, wird reduziert.
 Der Rechteckindex ergibt sich dann durch $z*n[0]+s$.
 */

```
double x0, x1;
int s, z;

if (x[0]<a[0] || x[0]>b[0] || x[1]<a[1] || x[1]>b[1])
    return (-1);

x0 = ((x[0]-a[0])/(b[0]-a[0]))*n[0];
s = (int)x0;
if ( s > n[0]-1) // oberer Rand in x[0]-Richtung
    s = n[0]-1;

x1 = ((x[1]-a[1])/(b[1]-a[1]))*n[1];
z = (int)x1;
if ( z > n[1]-1) // oberer Rand in x[1]-Richtung
    z = n[1]-1;

return (z*n[0]+s);
}
```

```
void Koordinaten(int i, double *a, double *b, int *n,
                double *x)
{
    /* Berechnung der Koordinaten des x-Vektors für den
       globalen Eckenindex i
```

Berechne aus dem globalen Eckenindex i den Spaltenindex s und den Zeilenindex z bezüglich der Unterteilung $n[0]$ in $x[0]$ -Richtung.

Berechne daraus die Koordinaten $x[0]$ und $x[1]$ auf dem Gebiet $\Omega = [a[0],b[0]] \times [a[1],b[1]]$ mit der Diskretisierung $n[0]$ bzw. $n[1]$.

```

*/

int s, z;

if (i<0 || i >= (n[0]+1)*(n[1]+1))
{
    cout << "Globaler Eckenindex auSSerhalb der Menge {0,...,"
        << (n[0]+1)*(n[1]+1)-1 << "}. " << endl;
    exit(0);
}

s = i % (n[0]+1);
z = i / (n[0]+1);

x[0] = a[0] + s * (b[0]-a[0])/n[0];
x[1] = a[1] + z * (b[1]-a[1])/n[1];
}

int FindeRechteckKoord(double *x, double *a, double *b,
                      int *n, double *koord)
{
/* Berechnung des Rechteckindex (Rückgabewert) für einen
Vektor x.
*x: Vektor der Länge 2 des Punktes x.

```

Transformation von $\Omega = [a[0], b[0]] \times [a[1], b[1]]$
auf das Rechteck $[0, n[0]] \times [0, n[1]]$.

Die Werte stehen in x_0 und x_1 .

Die ganzzahligen Werte stehen im Spaltenindex s (für x_0)
und Zeilenindex z (für x_1)

Der untere Rand eines Rechtecks gehört zum Rechteck.

Wird der obere Rand des Gebietes erreicht, wird reduziert.

Der Rechteckindex i ergibt sich dann durch $z \cdot n[0] + s$.

Die Werte für y_0 und y_1 ergeben sich aus dem Nachkomma-
anteil von x_0 bzw. x_1 .

*/

```
int s, z;
double x0, x1, y0, y1;

if (x[0]<a[0] || x[0]>b[0] || x[1]<a[1] || x[1]>b[1])
    return(-1);

x0 = ((x[0]-a[0])/(b[0]-a[0]))*n[0];
s = (int)x0;
if (s > n[0]-1) // oberer Rand in x[0]-Richtung
    s = n[0]-1;

x1 = ((x[1]-a[1])/(b[1]-a[1]))*n[1];
z = (int)x1;
if (z > n[1]-1) // oberer Rand in x[1]-Richtung
    z = n[1]-1;

y0 = x0 - s;
y1 = x1 - z;
```

```

    koord[0] = (1.0 - y0) * (1.0 - y1);
    koord[1] = y0 * (1.0 - y1);
    koord[2] = (1.0 - y0) * y1;
    koord[3] = y0 * y1;

    return(z*n[0]+s);
}

double Wert(double *x, double *a, double *b, int *n,
            double *v, int *flag)
{
/* Liefert den approximierten Wert v im Punkt x durch eine
   affin bilineare Funktion.
   x: auszuwertender Punkt x=(x[0],x[1])
   auf dem Rechteck Omega = [a[0],b[0]] x [a[1],b[1]]
   mit n[0] Spalten in x[0]-Richtung und n[1] Zeilen in
   x[1]-Richtung
   v: Werte der Funktion auf den Gitterpunkten von
   {0,...,(n[0]+1)*(n[1]+1)-1}.
   Rückgabewert flag: 1 auSSerhalb des Gitters, 0 im Gitter.
*/

    int i;
    int j;
    double koord[4], wert;

    j=FindeRechteckKoord(x, a, b, n, koord);
    if (j < 0)
    {

```

```
        *flag=1;
        return 0.;
    }

    wert=0;
    for (i=0; i<=3; i++)
        wert+=koord[i]*v[Gitter2d(j,i,n)];

    *flag=0;
    return(wert);
}

void rS(double *x, double uu, double *y)
{
    /* rechte Seite der Differentialgleichung oder
       Differenzgleichung */

    double r = 0.03;

    y[0] = r*x[0]-(uu*x[0]); // u = uu * x
    y[1] = 0;
}

double g(double *x, double uu)
{
    double gamma = 0.75;
    double eps=1e-6;

    if (gamma == 1) // ln u
    {
```

```

    if ((uu*x[0]) < eps)
        return(-1e6);
    else
        return(log(uu*x[0]));
}

if (gamma > 1)
{
    if ((uu*x[0]) < eps)
        return(-1e6);
    else
        return(pow((uu*x[0]), (1-gamma))/(1-gamma));
}

else
    return(pow((uu*x[0]), (1-gamma))/(1-gamma));
}

void Euler(double *x, double uu, double h, int d, double *y)
{
    /* Euler-Verfahren */

    int i;
    double f[2];

    rS(x, uu, f);

    for (i=0; i<d; i++)
        y[i] = x[i]+h*f[i];
}

```

```
double kGSV(double *a, double *b, int *n, int i, double *v,
            double uu, double h, double delta, int *flag)
{
/* kontrolliertes Gau"s-Seidel-Verfahren. Berechnet für ein
u die rechte Seite des
Optimalitätsprinzips.
Eingabe: a,b: Gittergrenzen,
n : Unterteilung des Gitters,
i : globaler Eckenindex,
v : Vektor der Wertefunktion,
uu : (skalierter) Kontrollwert uu = u/x,
h : Schrittweite
delta: Wert in der Exponentialfunktion unter dem Integral
Rückgabe: flag: 0, falls der neue Punkte y im Gitter liegt,
sonst 1, Berechneter Wert der rechten Seite des
Optimalitätsprinzips.
*/

double beta, wert, nenner;
double y[d], x[d], koord[4];
int j,k,l;

nenner = 1.;
beta = 1.-delta*h;

Koordinaten(i, a, b, n, x);
Euler(x, uu, h, d, y);
j=FindeRechteckKoord(y, a, b, n, koord);
```

```

wert = h*g(x,uu);

if (j < 0)
{
    *flag=1;
    return 0.;
}

for (k=0; k<=3; k++)
{
    l=Gitter2d(j,k,n);
    if (l!=i)
        wert+=beta*koord[k]*v[l];
    else
        nenner-=beta*koord[k];
}

wert/=nenner;

*flag=0;
return(wert);
}

int gs(double *a, double *b, int *n, int i, double *v,
        double *uu, int q, double h, double delta,
        double *max, double *uopt)
{
/* Gau"s-Seidel-Verfahren. Bestimmt den maximalen Wert
(max) f"ur alle u und liefert zus"atzlich den optimalen Wert
uopt mit zur"uck. Ruft dazu kGSV auf, die f"ur jedes u den

```

Wert berechnet.

Eingabe: a,b: Gittergrenzen,

n: Unterteilung des Gitters,

i: globaler Eckenindex,

v: Vektor der Wertefunktion,

u: Vektor der Kontrollwerte,

q: Länge des Vektors u,

h: Schrittweite

delta: Wert in der Exponentialfunktion unter dem Integral

Rückgabe: max: Maximum über u

uopt: u, dass das Maximum liefert

l: falls für jedes u der neue Punkt y außerhalb des Gitters liegt, sonst 0.

*/

```

    int k, flag;
    double sv, x[d];
    bool ufind;    // gibt an, ob bereits ein u gefunden wurde,
// so dass die Abbildung für x und u noch im Gitter liegt

    ufind=false;
    Koordinaten(i, a, b, n, x);

    for (k=0; k<q; k++)
    {
        sv=kGSV(a, b, n, i, v, uu[k], h, delta, &flag);

        if (flag == 0)
        {
            if (ufind == true)

```

```

        {
            if (sv > *max)
            {
                *max = sv;
                *uopt = uu[k];
            }
        }
    else
    {
        ufind = true;
        *max = sv;
        *uopt = uu[k];
    }
} // end for k

if (ufind == false)
{
    cout << "Fuer jedes u ist y auSSerhalb des Gitters!"
        << endl;
    return(1);
}

return(0);
}

int gs_iter(double *a, double *b, int *n, double *v,
            double *uu, int q, double eps, double h,
            double delta)
{
    /* Gau"s-Seidel-Iteration zur Berechnung des maximalen

```

```

    Wertes in jeder komponente v[i]
    Die Iteration wird solange fortgesetzt, bis die neuen
    v[i] von den alten in der Maximumnorm weniger als
    h*delta*eps abweichen
Eingabe: a,b: Gittergrenzen,
n: Unterteilung des Gitters,
v: Vektor der Wertefunktion,
u: Vektor der Kontrollwerte,
q: Länge des Vektors u,
eps: Geforderte Genauigkeit,
h: Schrittweite,
delta: Wert in der Exponentialfunktion unter dem Integral
Rückgabe: 1: falls ein x existiert, so dass für jedes u der
           neue Punkt y außerhalb des Gitters liegt, sonst 0.
*/

```

```

double beta, max, err, uopt;
int i, N;
long j;

N=(n[0]+1)*(n[1]+1);
beta=1.-delta*h;
j=0;
for (i=0;i<N;i++)
    v[i]=0.;

do
{
    j++;
    err=0.;

```

```

    for (i=0;i<N;i++)
    {
        if (gs(a,b,n,i,v,uu,q,h,delta,&max,&uopt) == 1)
            return(1);
        if (fabs(v[i]-max) > err)
            err = fabs(v[i]-max);

        v[i] = max;
    } // end for i

    if (j % 1 == 0)
        cout << "Iteration " << j << ", Abbruchfehler="
            << err << endl;
    }
    while (err > h*delta*eps);

    return(0);
}

int strategie_iter(double *a, double *b, int *n, double *v,
                 double *uu, int q, double eps, double h,
                 double delta)
{
    /* Kombination der Gau"s-Seidel-Iteration und der Strategie-
       Iteration zur Berechnung des maximalen Wertes in jeder
       Komponente v[i]
       Die Iteration wird solange fortgesetzt, bis die neuen v[i]
       von den alten in der Maximumnorm weniger als h*delta*eps
       abweichen.
       Die Strategie-Iteration wird nur verwendet, wenn das optimal

```

bestimmte u , von dem vorherigen u in der 1-Norm um maximal $0.1 \cdot N$ abweicht.

Eingabe: a, b : Gittergrenzen,
 n : Unterteilung des Gitters,
 v : Vektor der Wertefunktion,
 u : Vektor der Kontrollwerte,
 q : Länge des Vektors u ,
 eps : Geforderte Genauigkeit,
 h : Schrittweite,
 delta : Wert in der Exponentialfunktion unter dem Integral
Rückgabe: 1: falls ein x existiert, so dass für jedes u der
neue Punkt y außerhalb des Gitters liegt, sonst 0.
*/

```
double beta, max, err, uerr, uop;
double *uopt;
int i, N, flag;
int st; // gibt die Strategie an, 0: Gau"s-Seidel,
        // 1: Stratgie-Iteration

long j;

N= (n[0]+1) * (n[1]+1);
speichervek(uopt, N, "uopt");
beta=1.-delta*h;
j=0;
for (i=0; i<N; i++)
    v[i]=0.;
```

```

st=0;

do
{
    if (st == 0) j++;
    err=0.;
    uerr=0.;
    for (i=0;i<N;i++)
    {
        if (st == 0) // (2a) in Kapitel 3
        {
            if (gs(a,b,n,i,v,uu,q,h,delta,&max,&uop) == 1)
                return(1);

            if (j > 1)
                uerr += fabs(uopt[i]-uop);
            uopt[i] = uop;
        }
        else // (3) in Kapitel 3
            max = kGSV(a,b,n,i,v,uopt[i],h,delta,&flag);

        if (fabs(v[i]-max) > err)
            err = fabs(v[i]-max);
        v[i] = max;
    } // end for i

    if (j % 1 == 0 && st == 0)
        cout << "Iteration " << j << ", Abbruchfehler="
            << err << endl;

```

```
    if ((st == 1) && (err < h*delta*eps))
    {
        st = 0;
        err = h*delta*eps+1.;
    }
    else if ((st == 0) && (j > 1) && (uerr < 0.1*N))
    // (2b) im Skript
        st = 1;
    }
    while (err > h*delta*eps);

    delete []uopt;
    return(0);
}

int uoptx(double *x, double *a, double *b, int *n, double *v,
          double *uu, int q, double *uopt, double h,
          double delta)
{
    /* Berechnet approximativ das optimale u im Punkt x
    Eingabe: x: Werte der Trajektorie,
    a,b: Gittergrenzen
    n: Anzahl der Gitterunterteilungen, N=(n[0]+1)*(n[1]+1),
    v: Vektor der Länge N, Optimale Wertefunktion (approximativ)
    in den Gitterpunkten,
    u: Vektor der Länge q, Kontrollwerte,
    h: Schrittweite,
    delta: Wert in der Exponentialfunktion unter dem Integral.
    Rückgabe: 1: falls ein x existiert, so dass für jedes u
              der neue Punkt y außerhalb des
```

Gitters liegt, sonst 0.

uopt: optimale Steuerung im Punkt x.

*/

```
int k, flag;
```

```
double max, vh, y[d], beta;
```

```
bool ufind;
```

```
beta=1.-delta*h;
```

```
ufind=false;
```

```
for (k=0; k<q; k++)
```

```
{
```

```
    Euler(x,uu[k],h,d,y);
```

```
    vh=h*g(x,uu[k])+beta*Wert(y,a,b,n,v,&flag);
```

```
    if (flag == 0)
```

```
    {
```

```
        if (ufind == true)
```

```
        {
```

```
            if (vh > max)
```

```
            {
```

```
                max = vh;
```

```
                *uopt = uu[k];
```

```
            }
```

```
        }
```

```
    else
```

```
    {
```

```
        ufind = true;
```

```
        max = vh;
```

```
        *uopt = uu[k];
```

```
        }
    }
} // end for k

if (ufind == false)
{
    cout << "Fuer jedes u ist y auSSerhalb des Gitters!"
        << endl;
    return(1);
}

return(0);
}

int trajopt(double **x, int M, double *a, double *b, int *n,
            double *v, double *uu, int q, double *uopt,
            double h, double delta)
{
    /* Berechnet approximativ die optimale zeitdiskreten
    Trajektorien x und u
    Eingabe: x[0][j]: Werte der Trajektorie zum Zeitpunkt 0 in
            der j-ten Komponente,
    M: Länge der Trajektorien x und uopt,
    a,b: Gittergrenzen
    n: Anzahl der Gitterunterteilungen, N=(n[0]+1)*(n[1]+1),
    v: Vektor der Länge N, Optimale Wertefunktion (approximativ)
    in den Gitterpunkten,
    u: Vektor der Länge q, Kontrollwerte,
    h: Schrittweite,
    delta: Wert in der Exponentialfunktion unter dem Integral.
```

Rückgabe: 1: falls ein x existiert, so dass für jedes u der neue Punkt y außerhalb des

Gitters liegt, sonst 0.

$x[i][j]$: Matrix; Werte der Trajektorie zum Zeitpunkt h_i , $i=0, \dots, M-1$ in der j -ten Komponente,

$uopt$: Vektor der Länge M , optimale Steuerung zu den Zeitpunkten h_i , $i=0, \dots, M-1$.

*/

```
int i, j;
```

```
double y[d], beta;
```

```
beta=1.-delta*h;
```

```
for (i=0; i<M; i++)
```

```
{
```

```
    if (i > 0)
```

```
        for (j=0; j<d; j++)
```

```
            x[i][j]=y[j];
```

```
    if (uoptx(x[i], a, b, n, v, uu, q, &uopt[i], h, delta) == 1)
```

```
        return (1);
```

```
    Euler(x[i], uopt[i], h, d, y);
```

```
}
```

```
return(0);
```

```
}
```

```
int main(int argc, char **argv)
```

```
{
```

```
int i,spalte, zeile, n[d], N, M, q, tmpn[d], tmpN, N0, N1;
double a[d], b[d], *v, *uu, *uopt, x[d], **xopt, y[d];
// uu = u/x
double eps, h, delta, t, T, uopt2;
char s[30];

double max = 0.;

FILE *fp, *fp2, *fp3;

if (argc < 4 || argc > 5)
{
    cout << "Benutzung: " << argv[0] << " x[0] x[1] delta s"
        << endl;
    cout << "\t x[0], x[1]: Anfangswerte der zu berechnenden
        optimalen Trajektorie" << endl;
    cout << "\t delta: Parameter für das Optimalsteuerungs-
        problem" << endl;
    cout << "\t s: Die Werte der Value-Funktion wurden
        bereits berechnet und liegen in der Datei s"
        << endl;
    exit(0);
}

n[0]=100;
n[1]=100;

a[0]=0.; // für gamma > 1: a[0]=0.0001
b[0]=1.;
```

```

a[1]=0.;
b[1]=1.;

q=4001;
speichervek(uu,q,"uu");

for (i=0; i<q; i++)
    uu[i]= i*0.7/400;

eps = 0.001;
h = (double)1/12;
N = (n[0]+1)*(n[1]+1);
speichervek(v,N,"v");

delta = atof(argv[3]); // 0.06

strcpy(s,"werte");
strcat(s,argv[3]);
strcat(s,".asc");
if (argc == 4)
{
    if (strategie_iter(a,b,n,v,uu,q,eps,h,delta) == 1)
        return(1);

    fp = fopen(s,"w+");
    fprintf(fp,"%d %d %d\n",n[0]+1,n[1]+1,N);
    for (i=0; i<N; i++)
    {
        Koordinaten(i, a, b, n, x);
        fprintf(fp,"%f %f %f\n",x[0],x[1],v[i]);
    }
}

```

```
    }
    fclose(fp);
}
else
{
    fp = fopen(s, "r");
    if (fp == NULL)
        cout << "Datei existiert nicht!" << endl;
    else
    {
        fscanf(fp, "%d %d %d\n", &tmpn[0], &tmpn[1], &tmpN);
        if (tmpN != N)
        {
            cout << "N ist verschieden! Bitte überprüfen!"
                 << endl;
            return(1);
        }
        else
        {
            for (i=0; i<N; i++)
                fscanf(fp, "%lf %lf %lf\n", &x[0], &x[1], &v[i]);
            fclose(fp);
        }
    }
}

t=0;
T=180;
M=((int)ceil((T-t)/h))+1;
speichermat(xopt, M, d, "xopt");
```

```

speichervek(uopt,M,"uopt");
xopt[0][0] = atof(argv[1]);
xopt[0][1] = atof(argv[2]);

if (xopt[0][0]<a[0] || xopt[0][0]>b[0] || xopt[0][1]<a[1]
    || xopt[0][1]>b[1])
{
    cout << "Startpunkt auSSerhalb des Gitters!" << endl;
    return(1);
}
else
    if (trajopt(xopt,M,a,b,n,v,uu,q,uopt,h,delta) == 1)
        return(1);

strcpy(s,"trajektorie");
strcat(s,argv[3]);
strcat(s,"_");
strcat(s,argv[1]);
strcat(s,".asc");
fp = fopen(s,"wt");

for (i=0; i<M; i++)
{
    fprintf(fp,"%f %f %f %f\n",i*h,xopt[i][0],xopt[i][1],
           uopt[i]*xopt[i][0]);
}
fclose(fp);

// Werte der optimalen Feedbacksteuerung und des Vektor-
// feldes schreiben

```

```
strcpy(s, "uoptSkal"); // skalierte Feedbacksteuerung
strcat(s, argv[3]);
strcat(s, ".asc");
    fp = fopen(s, "wt");
fprintf(fp, "%d %d %d\n", n[0]+1, n[1]+1, N);

strcpy(s, "uopt"); // unskalierte Feedbacksteuerung
strcat(s, argv[3]);
strcat(s, ".asc");
    fp3 = fopen(s, "wt");
fprintf(fp3, "%d %d %d\n", n[0]+1, n[1]+1, N);

N0=4; // schreibe nur jeden N0-ten Wert in x-Richtung.
N1=10; // schreibe nur jeden N1-ten Wert in y-Richtung
strcpy(s, "feld");
strcat(s, argv[3]);
strcat(s, ".asc");
    fp2 = fopen(s, "wt");
fprintf(fp2, "%d %d %d %d\n", n[0]/N0+1, n[1]/N1+1,
        (n[0]/N0+1) * (n[1]/N1+1), (n[0]/N0+1) * (n[1]/N1+1));

for (i=0; i<N; i++)
{
    Koordinaten(i, a, b, n, x);

    if (uoptx(x, a, b, n, v, uu, q, &uopt2, h, delta) == 1)
        return(1);
    fprintf(fp, "%f %f %f\n", x[0], x[1], uopt2);
    fprintf(fp3, "%f %f %f\n", x[0], x[1], uopt2*x[0]);
}
```

```
    spalte = i % (n[0]+1);
    zeile = i / (n[0]+1);
    if ((spalte % N0 == 0) && (zeile % N1 == 0))
    {
        Koordinaten(i, a, b, n, x);
        rS(x,uopt2,y);
        fprintf(fp2,"%f %f %f %f\n",x[0],x[1],y[0],y[1]);
    }
}
fclose(fp);
fclose(fp2);
fclose(fp3);

for (i=0; i<M; i++)
    delete []xopt[i];
delete []xopt;
delete []uu;
delete []uopt;
delete []v;

return(0);
}
```


D MATLAB-Quelltexte

D.1 Funktion: zeichnen_wertef(name)

```
function zeichnen_wertef(name)

%Zeichnen der optimalen Wertefunktion - 3-dimensional

figure
b = load('name.asc');
n0 = b(1,1);
n1 = b(1,2);
N = b(1,3);
x = b(2:n0+1,1);
y = b(2:n0:N+1,2);
z = reshape(b(2:N+1,3),n0,n1);

mesh(x,y,z');
xlabel('x');
ylabel('y');
```

D.2 Funktion: zeichnen_uopt(name)

```
function zeichnen_uopt(name)

% Zeichnen der optimalen Feedbacksteuerung und der skalierten
% Feedbacksteuerung.
```

```
figure
b = load('name.asc');
n0 = b(1,1);
n1 = b(1,2);
N = b(1,3);
x = b(2:n0+1,1);
y = b(2:n0:N+1,2);
z = reshape(b(2:N+1,3),n0,n1);

mesh(x,y,z');
xlabel('x');
ylabel('y');
```

D.3 Funktion: zeichnen_feld(name)

Zeichnen des Vektorfeldes

```
function zeichnen_feld(name)

% Zeichnen des Vektorfeldes

figure
b = load('name.asc');
size(b);
n0 = b(1,1);
n1 = b(1,2);
N = b(1,3);
x1 = b(2:n0+1,1);
x2 = b(2:n0:N+1,2);
y1 = reshape(b(2:N+1,3),n0,n1);
```

```
y2 = reshape(b(2:N+1,4),n0,n1);
```

```
xmin1=min(x1);
```

```
xmax1=max(x1);
```

```
xmin2=min(x2);
```

```
xmax2=max(x2);
```

```
quiver(x1,x2,y1',y2');
```

```
axis([xmin1 xmax1 xmin2 xmax2]);
```

Darstellung der Gleichgewichte und Trennlinien durch Punkte

```
function zeichnen_feld(name)
```

```
% Zeichnen der Gleichgewichte und Trennlinien - 1-dimensional
```

```
figure
```

```
b = load('name.asc');
```

```
n0 = b(1,1);
```

```
n1 = b(1,2);
```

```
N = b(1,3);
```

```
x1 = b(2:n0+1,1);
```

```
y1 = reshape(b(2:N+1,3),n0,n1);
```

```
l = y1(:,1);
```

```
length(l);
```

```
for i=2:length(l)
```

```
    if (y1(i-1,1)>=0 && y1(i,1)<0)
```

```
        if (abs(y1(i-1,1))<abs(y1(i,1)))
```

```
        plot(x1(i-1,1),1,'o','MarkerEdgeColor','k',
            'MarkerFaceColor','g','MarkerSize',5);
    else
        plot(x1(i,1),1,'o','MarkerEdgeColor','k',
            'MarkerFaceColor','g','MarkerSize',5);
    end
elseif (y1(i-1,1)<0 && y1(i,1)>0)
    plot(x1(i,1),1,'o','MarkerEdgeColor','k',
        'MarkerFaceColor','r','MarkerSize',8);
else
    i=i+1;
end
end
```

D.4 Funktion: zeichnen_traj(name)

```
function zeichnen_traj(name)

%Zeichnen der optimalen Trajektorien

b = load('name.asc');

figure; % Zeichnen des Phasendiagramms
plot(b(:,2),b(:,3),'*');
xlabel('x1');
ylabel('x2');
hold on;

figure; % Zeichnen der Trajektorien gegen die Zeit
plot(b(:,1),b(:,2),'r.', b(:,1),b(:,3),'b.', b(:,1),b(:,4),'k.')
```


E Material auf der beiliegenden CD

Die beiliegende CD enthält die verwendeten C++-Programme im Verzeichnis C++ ,sowie alle angesprochenen MATLAB-Dateien im Verzeichnis MATLAB. Alle in der Arbeit verwendeten Abbildungen sind im Verzeichnis Bilder gespeichert, alle zur Auswertung verwendeten Datensätze im Verzeichnis Datensätze, sowie die komplette Arbeit als pdf-Datei und als Latex-Datei im Verzeichnis Diplomarbeit.

Das Verzeichnis C++

Es sind die folgenden Programme gespeichert

- Modell_1.cpp
- Modell_2.cpp
- Modell_3.cpp

Das Verzeichnis Matlab

Es sind die folgenden Dateien jeweils für die 2- als auch für die 3-dimensionale Darstellung gespeichert

- `zeichnen_wertef(name)`
- `zeichnen_uopt(name)`
- `zeichnen_feld(name)` sowohl für das Vektorfeld als auch die Darstellung der Gleichgewichte mittels Punkten
- `zeichnen_traj(name)`

Das Verzeichnis Bilder

Um die Abbildungen besser zuordnen zu können, haben die Datei-Namen die gleiche Bezeichnung wie die entsprechenden Abbildungsnummern im Text.

Das Verzeichnis Datensätze

Dieses Verzeichnis enthält die benutzten Datensätze, sortiert nach den einzelnen Modellen und der jeweiligen Parametervariation.

Literaturverzeichnis

- [1] M. BARDI UND I. CAPUZZO DOLCETTA, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman equations*, Birkhäuser, Boston, 1997.
- [2] R. BELLMANN, *Introduction to the Mathematical Theory of Control Processes, Volume I*, in R. Bellmann, Hrsg., *Mathematics in Science and Engineering, Volume 40*, Academic Press, 1967.
- [3] W. BREUER, M. GÜRTLER UND F. SCHUHMACHER, *Portfoliomanagement*, Gabler-Verlag, 1999.
- [4] J.Y. CAMPBELL UND L. M. VICEIRA, *Strategic Asset Allocation, Portfolio Choice for Long-Term Investors*, Oxford University Press, 2003.
- [5] F. EISENFÜHR UND M. WEBER, *Rationales Entscheiden*, 3. Auflage, Springer-Verlag, 1999.
- [6] L.C. EVANS, *Partial Differential Equations*, American Mathematical Society, 2002.
- [7] L. GRÜNE, *Numerik Dynamischer Systeme*, Ausarbeitung einer Vorlesung gehalten an der Universität Bayreuth im Wintersemester 2003/2004.
- [8] L. GRÜNE, *Numerische Dynamik von Kontrollsystemen*, Ausarbeitung einer Vorlesung gehalten an der Universität Bayreuth im Sommersemester 2004.
- [9] CH. HUANG UND R.H. LITZENBERGER, *Foundations for financial economics*, North-Holland, 1988.

- [10] M.I. KAMIEN UND N.L. SCHWARTZ, *Dynamic Optimization*, North-Holland, 1991.
- [11] R. KORN UND E. KORN, *Optionsbewertung und Portfolio-Optimierung, Moderne Methoden der Finanzmathematik*, 2. Auflage, Vieweg-Verlag, 2001.
- [12] H. MARKOWITZ, *Portfolio selection*, Journal of Finance 7, pp. 77-91, 1952.
- [13] R.C. MERTON, *Lifetime portfolio selection under uncertainty: the continuous case*, Review of Economical Statistics 51, pp. 247-257, 1969.
- [14] R.C. MERTON, *Continuous Time Finance*, Blackwell, 1990.
- [15] W. SEMMLER, *Static and Dynamic Portfolio Choice - Lecture Notes*, Ausarbeitung einer Vorlesung gehalten an der New School University im Jahr 2004.

Erklärung

Hiermit erkläre ich, daß ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.

Diese Arbeit hat in dieser oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen.

Bayreuth, den 30. September 2006

.....
Caroline Öhrlein