Numerische optimale Steuerung und Stabilisierung

Diplomarbeit
von
Lars Grüne

eingereicht beim
Institut für Mathematik
der Mathematisch-Naturwissenschaftlichen Fakultät
der Universität Augsburg
Januar 1994

Erstgutachter: *Prof. Dr. Fritz Colonius*
Zweitgutachter: *Prof. Dr. Bernd Aulbach*

Inhaltsverzeichnis

Kapitel 1

Einleitung

In dieser Arbeit wird ein Verfahren zur numerischen Stabilisierung bilinearer Kontrollsysteme entwickelt. Im Gegensatz zu Verfahren wie z.B. der direkten Methode oder dem Zugang über das Maximumsprinzip wird hier zur Stabilisierung ein *diskontiertes optimales Steuerungsproblem* gelöst, das eine Annäherung an das *optimale Durchschnittskostenproblem* darstellt, welches das eigentliche Stabilisierungsproblem löst.

Dazu wird zunächst eine *optimale Wertefunktion* berechnet, die dann als Approximation für die *Lyapunov-Exponenten* der betrachteten Kontrollsysteme in gewissen Teilmengen des Zustandsraumes interpretiert wird. Diese wiederum können als Maß für die Stabilität der Systeme aufgefaßt werden. Ausgehend von dieser Wertefunktion werden dann *stabilisierende Kontrollen* für die Systeme berechnet. Dies bedeutet, daß ein großer Teil des Rechenaufwandes bereits vor der eigentlichen Berechnung der Kontrollen erledigt wird. Die tatsächliche Berechnung dieser Kontrollen läßt sich anschließend mit einem relativ einfachen numerischen Verfahren durchführen.
Zusätzlich zu den vielfältigen Methoden, die Stabilisierbarkeit von Kontrollsystemen mit analytischen Methoden zu untersuchen (vgl. z.B. Bacciotti [1]) liefert dieses Verfahren also neben der eigentlichen Stabilisierung eine Möglichkeit, Lyapunov-Exponenten eines Kontrollsystems zu berechnen, und so die Stabilisierbarkeit von Kontrollsystemen numerisch zu analysieren.

Ein großer Teil der Arbeit beschäftigt sich demnach mit Theorie und Numerik diskontierter optimaler Steuerungsprobleme und deren Zusammenhänge mit dem Durchschnittskostenproblem. Diese Zusammenhänge zeigen aber nicht nur die Möglichkeiten und Grenzen dieses Verfahrens im Zusammenhang mit Stabilisierungsproblemen auf; vielmehr kann die Theorie und das entwickelte numerische Verfahren auch unabhängig von Stabilisierungsproblemen betrachtet werden. So spielen diskontierte optimale Steuerungsprobleme z.B. in Modellen der mathematischen Wirtschaftstheorie eine Rolle, vgl. Seierstad, Sydsæter [15], Kapitel 2 und 3.

Die Arbeit gliedert sich in drei Hauptabschnitte. Im ersten, der die Kapitel 2 – 4 umfaßt werden die theoretischen Grundlagen des diskontierten optimalen Steuerungsproblems und die Zusammenhänge mit dem Durchschnittskostenproblem behandelt.

In Kapitel 2 wird zunächst das diskontierte optimale Steuerungsproblem definiert und dann

speziell auf die optimale Wertefunktion eingegangen. Die grundlegende Eigenschaft, die diese Wertefunktion erfüllt, ist das *Bellman'sche Optimalitätsprinzip*, das in diesem Kapitel hergeleitet wird. Desweiteren wird die Stetigkeit der Wertefunktion gezeigt.
Aus dem Optimalitätsprinzip wird dann die *Bellman-Gleichung* hergeleitet, eine partielle Differentialgleichung, die von der Wertefunktion erfüllt wird. Eine Diskretisierung dieser Gleichung wird dann später als Grundlage für das numerische Verfahren dienen. Da die Wertefunktion im Allgemeinen nicht differenzierbar ist, muß der Lösungsbegriff von partiellen Differentialgleichungen erweitert werden, was zur Theorie der Viskositätslösungen führt. Zum Abschluß des Kapitels wird über ein Existenz- und Eindeutigkeitsresultat die optimale Wertefunktion als eindeutige Viskositätslösung der Bellman-Gleichung charakterisiert.

In Kapitel 3 wird dann gezeigt, in welchem Sinne die optimale Wertefunktion des diskontierten optimalen Steuerungsproblems eine Annäherung an die optimale Wertefunktion des Durchschnittskostenfunktionals darstellt. Da diese Wertefunktion im Allgemeinen unstetig ist, kann eine gleichmäßige Konvergenz im gesamten Zustandsraum nicht erwartet werden. Diese kann jedoch auf gewissen Teilmengen des Zustandsraumes gezeigt werden. Dazu wird der Begriff der *Kontrollmengen* eingeführt. Eine gleichmäßige Konvergenzaussage läßt sich dann für kompakte Teilmengen von Kontrollmengen formulieren, ein schwächeres Resultat können wir für *Einzugsbereiche* von Kontrollmengen formulieren.

Im vierten Kapitel wird dann eine wichtiger Grundlage für das Stabilisierungsverfahren gezeigt. Das numerische Verfahren liefert – wie oben bereits erwähnt – eine Lösung des diskontierten optimalen Steuerungsproblems. Es stellt sich nun die Frage, ob die Kontrollen, die dieses Verfahren liefert, auch für das Durchschnittskostenproblem „sinnvoll" sind. Der in diesem Kapitel bewiesene *Approximationssatz* zeigt, daß dies unter gewissen Voraussetzungen der Fall ist.

Der zweite Hauptabschnitt der Arbeit beschäftigt sich mit der Numerik des diskontierten optimalen Steuerungsproblems. Die grundlegenden Ideen dazu stammen von Maurizio Falcone [10], [11]. In dieser Arbeit wird ein anderer Konvergenzbeweis hergeleitet, der eine etwas bessere Konvergenzrate liefert. Außerdem wird ein – auf derselben Theorie beruhendes – neues numerisches Verfahren vorgestellt, das bei allen durchgeführten numerischen Tests mit erheblich weniger Rechenaufwand auskommt.

Kapitel 5 deckt diesen zweiten Hauptabschnitt ab. Zunächst wird die Bellman-Gleichung diskretisiert und ein *zeitdiskretes Kontrollsystem* definiert, dessen optimale Wertefunktion die eindeutige Lösung der diskretisierten Bellman-Gleichung ist. Analog zum zweiten Kapitel wird dann ein diskretes Optimaltätsprinzip und die Stetigkeit der diskretisierten Wertefunktion gezeigt. Im Anschluß daran wird der Diskretisierungsfehler der Zeitdiskretisierung abgeschätzt; dies geschieht gleichmäßig auf dem gesamten Zustandsraum.
Nach der Zeitdiskretisierung wird die Diskretisierung im Zustandsraum diskutiert. Auch hier wird wieder ein Existenz- und Eindeutigkeitsresultat bewiesen und dann der Diskretisierungsfehler – wiederum gleichmäßig – abgeschätzt.
Nachdem alle theoretischen Grundlagen abgehandelt sind, werden verschiedene Berechnungstrategien für die diskretisierte Wertefunktion vorgestellt. Der numerische Vergleich der Strategien ist in Kapitel 7 aufgeführt.
Der letzte Abschnitt des Kapitels befaßt mit der Berechnung der ε-optimalen Kontrollen.

Es wird ein Algorithmus hergeleitet, der gleichmäßig optimale Zustandsfeedbacks für das diskretisierte Steuerungsproblem liefert und es wird gezeigt, daß diese Kontrollen auch für das nichtdiskretisierte (also das ursprüngliche) Kontrollsystem gleichmäßig ε-optimal sind.

Der dritte Hauptabschnitt behandelt nun die Anwendung der vorherigen Abschnitte auf das Stabilisierungsproblem, d.h. wir diskutieren, wie und unter welchen Voraussetzungen die numerisch berechneten Kontrollen zur Stabilisierung bilinearer Kontrollsysteme verwendet werden können. Anhand von einigen Beispielsystemen wird der Algorithmus dann numerisch getestet.

Im sechsten Kapitel werden zunächst die Lyapunov-Exponenten definiert und gezeigt, wie sich diese durch das Lösen eines Durchschnittkostenproblems berechnen lassen. Hierzu wird das ursprüngliche System auf den *projektiven Raum* projiziert, was eine wesentliche Vereinfachung darstellt, da die Dimension des Zustandsraumes so um 1 abnimmt und der neue Zustandsraum kompakt, also insbesondere beschränkt ist. Weiterhin werden einige Eigenschaften von Kontrollmengen im projektiven Raum und den Lyapunov-Exponenten und -Spektren zitiert, die zur Interpretation der numerischen Ergebnisse nötig sind.
Der eigentliche Algorithmus wird am Ende des Kapitels vorgestellt. Ausgehend von den Kontrollen, die das numerische Verfahren liefert, wird eine stabilisierende Kontrolle für das Kontrollsystem konstruiert.

In Kapitel 7 wird dieses Verfahren dann getestet. Zunächst werden zwei verschiedene, in Kapitel 5 vorgestellte Algorithmen zur Berechnung der Wertefunktion an verschiedenen Beispielen gegeneinander getestet, um den Rechenaufwand abzuschätzen. Danach werden die numerischen Ergebnisse des eigentlichen Stabilisierungsalgorithmus vorgsetellt. Am Beispiel des zweidimensionalen sowie des dreidimensionalen linearen Oszillators mit verschiedenen Parametern werden sowohl die Berechnung der Lyapunov-Exponenten als auch der stabilisierenden Kontrollen bzw. Trajektorien durchgeführt. In allen diesen Beispielen war die Stabilisierung mit dem Algorithmus möglich.

Im Anhang A finden sich einige Eigenschaften über Trajektorien im projektiven Raum $\mathbb{P}^{n-1}$. Da das numerische Verfahren in lokalen Koordinaten (also im $\mathbb{R}^{n-1}$) durchgeführt wird, mußte das Problem des Parameterwechsels gelöst werden. Um zu zeigen, daß bei einen Parameterwechsel, der mit den exakten sowie den diskretisierten lokalen Trajektorien durchgeführt wird, Eigenschaften wie stetige Abhängigkeit vom Anfangswert und Konvergenz des Euler-Verfahrens nicht verlorengehen, sind hier einige differentialgeometrische Überlegungen zusammengefaßt, die im laufenden Text der vorhergehenden Kapitel inhaltlich keinen Platz mehr fanden. In den Kapiteln, in denen diese Eigenschaften benötigt werden, wird auf die entsprechenden Aussagen in diesem Anhang verwiesen.

In Anhang B und C sind die Programme beschrieben, die zur numerischen Berechnung verwendet wurden. Im Anhang B ist die Implementierung beschrieben. Eine große Hilfe dabei war, daß ich von einem bereits bestehenden Programm von Uwe Sorgenfrei ausgehen konnte, welches dieser im Rahmen seiner Diplomarbeit [16] erstellt hat. Lediglich der neue Algorithmus sowie die zusätzlichen Funktionen zur Behandlung des Parameterwechsels für den projektiven Raum mußten von mir ergänzt werden.
Anhang C beschreibt die Bedienung der Programme.

Im Anhang D ist der C-Quelltext der einzelnen Module des Programms abgedruckt. Die

Module, die aus [16] unverändert übernommen wurden, sind hier – soweit sie nicht für das Verständnis des Programms nötig sind – nicht abgedruckt. Bei Modulen, die nur geringfügig gegenüber [16] verändert wurden, ist dies im Kopf vermerkt.

Für die in Kapitel 7 abgedruckten grafischen Darstellungen von Kontrollmengen wurde das Programm CS2DIM zur numerischen Berechnung von Kontrollmengen von Gerhard Haeckl verwendet. Ich möchte mich an dieser Stelle für die vielen Hilfestellungen im Umgang mit diesem Programm und bei der Berechnung der Kontrollmengen bedanken. Ebenfalls möchte ich mich bei Götz Grammel für eine Reihe von Anregungen speziell zu den im Anhang A ausgeführten Überlegungen zu Trajektorien im Projektiven Raum bedanken. Besonders bedanken möchte ich mich zuletzt noch bei Fritz Colonius für die hervorragende Betreuung und Motivierung während der gesamten Zeit, in der diese Arbeit entstanden ist.

Augsburg, Januar 1994

Kapitel 2

Das diskontierte optimale Steuerungsproblem

In diesem Kapitel werden wir das *diskontierte optimale Steuerungsproblem* definieren, und 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. Zum Abschluß des zweiten Abschnitts werden wir eine *Hamilton-Jacobi-Gleichung* für die Wertefunktion, die *Bellman-Gleichung*, herleiten.

Da die Wertefunktion im Allgemeinen nicht differenzierbar ist, kann sie diese partielle Differentialgleichung im klassischen Sinne nicht lösen. Deshalb werden wir im dritten Abschnitt den Lösungsbegriff partieller Differentialgleichungen erweitern und das Konzept der *Viskositätslösungen* einführen, das uns dann zu einem *Existenz- und Eindeutigkeitsresultat* führt.

Dieses Kapitel orientiert sich in Aufbau und Argumentation an Sorgenfrei [16], Kapitel 2 und Colonius [4], Kapitel 2.

2.1 Definition des diskontierten optimalen Steuerungsproblems

Wir betrachten hier ein Problem der optimalen Steuerung auf einer offenen Menge $W \subseteq \mathbb{R}^n$, dessen Dynamik gegeben ist durch die autonome gewöhnliche Differentialgleichung

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ x(0) &= x \in W \end{aligned} \tag{2.1}$$

mit Kontrollfunktionen

$$u(\cdot) \in \mathcal{U} := \{u : \mathbb{R} \to U, \;\; u \text{ meßbar}\} \tag{2.2}$$

wobei der *Kontrollwertebereich* $U \in \mathbb{R}^m$ kompakt ist.

In den in dieser Arbeit diskutierten Kontrollsystemen ist diese Menge W nicht notwendigerweise invariant, d.h. es kann Lösungstrajektorien von (2.1) geben, die aus W herauslaufen. Es existiert aber in jedem Fall eine Funktion $\Psi : \mathbb{R}^N \to W$, die eingeschränkt auf W gerade die Identität auf W ist und verschiedene Eigenschaften erfüllt, die wir im weiteren benötigen werden. Die Funktion Ψ ergibt sich dadurch, daß die betrachteten Systeme als Zustandsraum den *Projektiven Raum* besitzen. W ist hierbei eine lokale Parameterumgebung und Ψ entspricht einem Parameterwechsel. Nähere Einzelheiten hierzu sowie die Beweise zu den benötigten Eigenschaften finden sich im Anhang A. An den Stellen, an denen diese Eigenschaften eingehen, wird auf die entsprechenden Aussagen im Anhang verwiesen. Wir nehmen an, daß die Lösungstrajektorien $\hat{\varphi}(t, x_0, u(\cdot))$ von (2.1) für alle Startwerte $x_0 \in W$ eindeutig existieren, solange sie in W bleiben (vgl. Proposition 2.4). Dann definieren wir Trajektorien für alle Zeiten $t > 0$ durch

Definition 2.1 Es sei $\hat{\varphi}(t, x_0)$ die Trajektorie zum Anfangswert $x_0 \in W$. Dann definieren wir mittels der Funktion Ψ zu $x \in \mathbb{R}^{n-1}$ induktiv

$$\begin{aligned} \varphi(t, x, u(\cdot)) &:= \hat{\varphi}(t, \Psi(x), u(\cdot)) \ \forall 0 \le t < T_1 \ \text{ mit } \ \hat{\varphi}(t, \Psi(x)) \in W \\ \varphi(T_i + t, x, u(\cdot)) &:= \hat{\varphi}(t, \Psi(\varphi(T_i, x, u(\cdot))), u(T_i - \cdot)) \ \forall 0 \le t < T_{i+1} - T_i \\ &\qquad \text{mit } \ \hat{\varphi}(t, \Psi(\varphi(T_i, x))) \in W \end{aligned} \tag{2.3}$$

für $i \in \{1, 2, \ldots\}$. Hierbei seien die T_i jeweils maximal gewählt und Ψ garantiert uns, daß $T_{i+1} - T_i \ge T > 0$ gilt (vgl. Forderung (A.4) und Bemerkung A.5 im Anhang).

Bemerkung 2.2 Die Definition der Trajektorien in Definition 2.1 bedeutet keine Einschränkung der Allgemeinheit. Für Systeme auf dem $\mathbb{R}^n$ oder auf invarianten Teilmengen $\mathcal{O} \subset \mathbb{R}^n$ können wir $\Psi \equiv \mathrm{id}_{\mathbb{R}^n}$ setzen. Dieser Sonderfall wird in einigen Aussagen speziell berücksichtigt.

Definition 2.3 Das *Kostenfunktional* J_ρ zur Diskontrate $\rho \in \mathbb{R}^+$ weist jedem Anfangswert $x \in W$ und jeder Kontrolle $u(\cdot) \in \mathcal{U}$ eine reelle Zahl zu. J_ρ ist definiert durch

$$\begin{aligned} J_\rho : W \times \mathcal{U} &\to \mathbb{R} \\ (x, u(\cdot)) &\mapsto \int_0^\infty e^{-\rho t} g(\varphi(t, x, u(\cdot)), u(t)) dt. \end{aligned}$$

Wir stellen folgende Bedingungen an die Funktionen f und g:

$$f : W \times U \to W \ \text{ sei stetig auf } \ W \times U \tag{2.4}$$

$$\|f(x,u) - f(y,u)\| \le L_f \|x - y\| \quad \forall x, y \in W \ \forall u \in U \ \text{ für ein } \ L_f \in \mathbb{R} \tag{2.5}$$

$$\|f(x,u)\| \le M_f \quad \forall (x,u) \in W \times U \ \text{ für ein } \ M_f \in \mathbb{R} \tag{2.6}$$

$$g : W \times U \to \mathbb{R} \ \text{ sei stetig auf } \ W \times U \tag{2.7}$$

$$|g(x,u) - g(y,u)| \le L_g \|x - y\| \quad \forall x, y \in W \ \forall u \in U \ \text{ für ein } \ L_g \in \mathbb{R} \tag{2.8}$$

$$0 \le g(x,u) \le M_g \quad \forall (x,u) \in W \times U \ \text{ für ein } \ M_g \in \mathbb{R} \tag{2.9}$$

Proposition 2.4 Zu allen $(x, u(\cdot)) \in W \times \mathcal{U}$ existiert unter den obigen Voraussetzungen an f eine eindeutig bestimmte Lösungstrajektorie $\varphi(t, x, u(\cdot))$.

Beweis: Nach Colonius [3], Satz 2.3 existieren die Trajektorien $\hat{\varphi}(t, x, u(\cdot))$ eindeutig, solange sie in W bleiben. Die Definition der $\varphi(t, x, u(\cdot))$ impliziert dann die Existenz und Eindeutigkeit für alle $t \geq 0$.

Bemerkung 2.5 Die Funktion $\varphi : \mathbb{R} \times W \times \mathcal{U} \to W$ hat für bel. $x \in W$, $u(\cdot) \in \mathcal{U}$ folgende *Flußeigenschaften*:

$$\begin{aligned} &\varphi(0, x, u(\cdot)) = x \\ &\varphi(t, \varphi(s, x, u(\cdot)), u(s + \cdot)) = \varphi(t + s, x, u(\cdot)) \text{ für } s, t \in \mathbb{R}^+. \end{aligned}$$

Bemerkung 2.6 Bedingung (2.9) kann auch durch die Forderung

$$|g(x, u)| \leq M_g \quad \forall (x, u) \in W \times U \text{ für ein } M_g \in \mathbb{R}$$

ersetzt werden, da sich durch die Addition einer Konstanten zu g qualitativ keine Veränderung ergibt.
Wir verwenden in diesem und dem folgenden Kapitel Bedingung (2.9), da sich so an späterer Stelle technische Vereinfachungen ergeben.

Das *diskontierte optimale Steuerungsproblem* liegt nun darin, das Kostenfunktional J_ρ zu jedem $x \in W$ über alle $u \in \mathcal{U}$ zu minimieren.

Definition 2.7 Die *optimale Wertefunktion* v_ρ zur Diskontrate $\rho > 0$ ist definiert durch

$$\begin{aligned} v_\rho : W &\to \mathbb{R} \\ x &\mapsto \inf_{u(\cdot) \in \mathcal{U}} J_\rho(x, u(\cdot)) \end{aligned}$$

Bemerkung 2.8 Aus Bedingung (2.9) und $\rho > 0$ folgt sofort die Beschränktheit der Wertefunktion v_ρ:

$$|v_\rho(x)| \leq \int_0^\infty e^{-\rho t} M_g \, dt \leq \frac{M_g}{\rho}$$

Definition 2.9 Ein Paar $(x_0, u_{x_0}(\cdot)) \in W \times \mathcal{U}$ heißt *optimal* oder *optimales Paar* zur Diskontrate $\rho > 0$, falls $v_\rho(x_0) = J_\rho(x_0, u_{x_0}(\cdot))$. Die Kontrollfunktion $u_{x_0}(\cdot) \in \mathcal{U}$ heißt dann *optimale Kontrolle*.

Wir wollen nun einige wichtige Eigenschaften der optimalen Wertefunktion herleiten, die wir in späteren Kapiteln benötigen werden.

2.2 Eigenschaften der optimalen Wertefunktion

2.2.1 Bellman's Optimalitätsprinzip

Das Bellman'sche Optimalitätsprinzip ist die grundlegende Eigenschaft der optimalen Wertefunktion, aus der alle anderen für uns wichtigen Eigenschaften abgeleitet werden.
Die Idee, die hinter dem Bellman'schen Optimalitätsprinzip steht, ist, daß Endstücke optimaler Trajektorien zu jeder Zeit selbst optimale Trajektorien sind, oder anders gesagt, daß bei einer optimalen Kontrolle zu jeder Zeit optimal gesteuert wird.

Satz 2.10 (Bellman's Optimalitätsprinzip)
Unter den Voraussetzungen (2.4) – (2.9) an f und g gilt für beliebiges $\rho > 0$ und alle $t > 0$, $x \in W$:

$$v_\rho(x) = \inf_{u(\cdot)\in\mathcal{U}} \Big\{ \int_0^t e^{-\rho\tau} g(\varphi(\tau, x, u(\cdot)), u(\tau))\, d\tau + e^{-\rho t} v_\rho(\varphi(t, x, u(\cdot))) \Big\} \tag{2.10}$$

Beweis: Wir zeigen die Gleichheit durch Beweisen der entsprechenden Ungleichungen in beiden Richtungen.

„≥": Sei $u(\cdot) \in \mathcal{U}$ eine beliebige Kontrollfunktion und $x(\tau) := \varphi(\tau, x, u(\cdot))$. Dann gilt für $t > 0$:

$$\begin{aligned}
J_\rho(x, u(\cdot)) &= \int_0^\infty e^{-\rho\tau} g(x(\tau), u(\tau)) d\tau \\
&= \int_0^t e^{-\rho\tau} g(x(\tau), u(\tau)) d\tau + \int_t^\infty e^{-\rho\tau} g(x(\tau), u(\tau)) d\tau \\
&= \int_0^t e^{-\rho\tau} g(x(\tau), u(\tau)) d\tau + e^{-\rho t} \int_0^\infty e^{-\rho\tau} g(\varphi(\tau, x(t), u(t+\cdot)), u(t+\tau)) d\tau \\
&\geq \int_0^t e^{-\rho\tau} g(x(\tau), u(\tau)) d\tau + e^{-\rho\tau} v_\rho(x(t))
\end{aligned}$$

Da aber $u \in \mathcal{U}$ beliebig gewählt war, folgt die Ungleichung aus $v_\rho(x) = \inf_{u(\cdot)\in\mathcal{U}} J_\rho(x, u(\cdot))$.

„≤": Seien $t > 0$, $\varepsilon > 0$ gegeben und $u_1(\cdot) \in \mathcal{U}$ so gewählt, daß

$$\begin{aligned}
&\inf_{u(\cdot)\in\mathcal{U}} \Big\{ \int_0^t e^{-\rho\tau} g(\varphi(\tau, x, u(\cdot)), u(\tau))\, d\tau + e^{-\rho t} v_\rho(\varphi(t, x, u(\cdot))) \Big\} \\
&\geq \int_0^t e^{-\rho\tau} g(\varphi(\tau, x, u_1(\cdot)), u_1(\tau))\, d\tau + e^{-\rho t} v_\rho(\varphi(t, x, u_1(\cdot)) - \varepsilon.
\end{aligned} \tag{2.11}$$

Sei $u_2(\cdot) \in \mathcal{U}$ so gewählt, daß

$$J_\rho(\varphi(t,x,u_1(\cdot)),u_2(\cdot)) \le v_\rho(\varphi(t,x,u_1(\cdot))) + \varepsilon. \tag{2.12}$$

Wir konstruieren nun eine Kontrollfunktion $\bar{u}(\cdot) \in \mathcal{U}$ durch

$$\bar{u}(\tau) = \begin{cases} u_1(\tau) & \text{falls } 0 \le \tau \le t \\ u_2(\tau - t) & \text{falls } \tau \ge t. \end{cases}$$

Zu $\bar{u}(\cdot)$ gehört also die Lösung $\bar{\varphi}(\cdot)$ gegeben durch

$$\bar{\varphi}(\tau,x,\bar{u}(\cdot)) = \begin{cases} \varphi(\tau,x,u_1(\cdot)) & \text{falls } 0 \le \tau \le t \\ \varphi(\tau - t, \varphi(t,x,u_1(\cdot)), u_2(\tau - \cdot)) & \text{falls } \tau \ge t \end{cases}$$

und damit gilt

$$\begin{aligned}
&\inf_{u(\cdot)\in\mathcal{U}} \Big\{ \int_0^t e^{-\rho\tau} g(\varphi(\tau,x,u(\cdot)),u(\tau))\,d\tau + e^{-\rho t} v_\rho(\varphi(t,x,u(\cdot))) \Big\} \\
&\overset{(2.11)}{\ge} \int_0^t e^{-\rho\tau} g(\varphi(\tau,x,u_1(\cdot)),u_1(\tau))d\tau + e^{-\rho t} v_\rho(\varphi(t,x,u_1(\cdot))) - \varepsilon \\
&\overset{(2.12)}{\ge} \int_0^t e^{-\rho\tau} g(\varphi(\tau,x,u_1(\cdot)),u_1(\tau))d\tau + e^{-\rho t}\Big(J_\rho(\varphi(t,x,u_1(\cdot)),u_2(\cdot)) - \varepsilon \Big) - \varepsilon \\
&= \int_0^t e^{-\rho\tau} g(\bar{\varphi}(\tau,x,\bar{u}(\cdot)),\bar{u}(\tau))d\tau + \int_t^\infty e^{-\rho\tau} g(\bar{\varphi}(\tau,x,\bar{u}(\cdot)),\bar{u}(\tau))d\tau - e^{-\rho t}\varepsilon - \varepsilon \\
&= \int_0^\infty e^{-\rho\tau} g(\bar{\varphi}(\tau,x,\bar{u}(\cdot)),\bar{u}(\tau))d\tau - (1 + e^{-\rho t})\varepsilon \;\ge\; v_\rho(x) - (1 + e^{-\rho t})\varepsilon.
\end{aligned}$$

Da aber $\varepsilon > 0$ beliebig gewählt war, folgt hieraus die Behauptung. □

Das folgende Korollar zeigt, daß das Bellman'sche Optimalitätsprinzip nicht nur eine Eigenschaft der optimalen Wertefunktion beschreibt, sondern auch eine Eindeutigkeitsaussage liefert.

Korollar 2.11 Sei $w : W \to \mathbb{R}$ eine beschränkte Funktion, die für positive $t \in \mathbb{R}$ und $x \in W$ die Gleichung (2.10) für $\rho > 0$ erfüllt. Dann gilt $v_\rho \equiv w$, d.h. v_ρ ist durch die Gleichung (2.10) eindeutig charakterisiert.

Beweis: w erfüllt (2.10), also gilt nach Voraussetzung für $x \in W$:

$$\begin{aligned}
|v_\rho(x) - w(x)| &\le \sup_{u(\cdot)\in\mathcal{U}} \Big| e^{-\rho t}\Big(v_\rho(\varphi(t,x,u(\cdot))) - w(\varphi(t,x,u(\cdot))) \Big) \Big| \\
&\le e^{-\rho t} \sup_{u(\cdot)\in\mathcal{U}} |v_\rho(\varphi(t,x,u(\cdot))) - w(\varphi(t,x,u(\cdot)))| \\
&\le e^{-\rho t} \sup_{\tilde{x}\in W} |v_\rho(\tilde{x}) - w(\tilde{x})|
\end{aligned}$$

Da dies für beliebige $x \in W$ gilt, folgt auch

$$\sup_{x \in W} |v_\rho(x) - w(x)| \leq e^{-\rho t} \sup_{x \in W} |v_\rho(x) - w(x)|.$$

Wegen $e^{-\rho t} < 1$ folgt nun $v(x) = w(x) \quad \forall x \in W$. □

2.2.2 Stetigkeit der Wertefunktion

Mit Hilfe des Bellman'schen Optimalitätsprinzips läßt sich eine weitere wichtige Eigenschaft der Wertefunktion zeigen, nämlich die Stetigkeit. Anschaulich ist die Stetigkeit gut zu erklären, da sich die Trajektorien zu nahe beieinanderliegenden Anfangswerten wegen der stetigen Abhängigkeit vom Anfangswert auf einem endlichen Zeitintervall zu einer vorgegebenen Kontrollfunktion $u(\cdot) \in \mathcal{U}$ nur wenig unterscheiden. Die Stetigkeit von v_ρ folgt dann aus der Stetigkeit der Kostenfunktion g. Wir werden zeigen daß v *Hölder-stetig* ist.

Definition 2.12 Eine Funktion $w : W \to \mathbb{R}$ heißt *Hölder-stetig* mit *Hölder-Exponent* $\gamma \in (0, 1]$, falls gilt

$$|w(x_1) - w(x_2)| \leq C\|x_1 - x_2\|^\gamma$$

für alle $x_1, x_2 \in W$ mit einer Konstanten $C \in \mathbb{R}^+$.

Definition 2.13 $\mathcal{H}_\gamma$ bezeichne den Raum der Hölder-stetigen Funktionen w auf W mit Hölder-Exponenten $\gamma \in (0, 1]$.
$\mathcal{H}_\gamma$ ist normiert durch

$$\|w\|_{\mathcal{H}_\gamma} := \sup_{x \in W} |w(x)| + |w|_{0,\gamma},$$

wobei $|w|_{0,\gamma}$ definiert ist durch

$$|w|_{0,\gamma} := \sup_{\substack{x_1, x_2 \in W \\ x_1 \neq x_2}} \frac{|w(x_1) - w(x_2)|}{\|x_1 - x_2\|^\gamma}$$

Um die Hölder-Stetigkeit der Wertefunktion zu zeigen, benötigen wir noch ein vorbereitendes Lemma.

Lemma 2.14 Sei $\Phi : [0, \infty) \to \mathbb{R}^+$ eine meßbare Funktion mit $0 \leq \Phi(t) \leq \min\{Ae^{Bt}, C\}$, $t \geq 0$ für positive Konstanten $A < C$ und B aus $\mathbb{R}$.
Dann gilt für positives $\rho \in \mathbb{R}$:

$$\int_0^\infty e^{-\rho t} \Phi(t) dt \leq KA^\sigma$$

mit $K = K(\sigma) > 0$, wobei $\sigma = 1$, falls $\rho > B$, $\sigma \in (0, 1)$ beliebig, falls $\rho = B$ und $\sigma = \frac{\rho}{B}$, falls $\rho < B$.

Beweis: Nach Voraussetzung gilt für alle $0 \leq T \leq \infty$

$$\int_0^\infty e^{-\rho t}\Phi(t)dt \leq A\int_0^T e^{(B-\rho)t}dt + C\int_T^\infty e^{-\rho t}dt$$

Durch Ausrechnen der Integrale auf der rechten Seite mit $T = \infty$ für $\rho > B$, $T = \frac{1}{\rho}\ln\frac{C}{A}$ für $\rho = B$ und $T = \frac{1}{B}\ln\frac{C}{A}$ für $\rho < B$ erhält man

$$\int_0^\infty e^{-\rho t}\Phi(t)dt \leq \begin{cases} \frac{A}{\rho - B} & \text{falls } \rho > B \\ A\left(\frac{1}{\rho} + \frac{1}{\rho}\ln\frac{C}{A}\right) & \text{falls } \rho = B \\ A^{\frac{\rho}{B}}\left(\frac{1}{B-\rho} + \frac{1}{\rho}\right)C^{1-\frac{\rho}{B}} & \text{falls } \rho < B. \end{cases}$$

Aus diesen Ungleichungen folgt nun die Behauptung. □

Satz 2.15 Es seien die Bedingungen (2.4) – (2.9) an f und g erfüllt, $\rho > 0$. Dann ist $v \in \mathcal{H}_\gamma$, d.h. die Wertefunktion v_ρ ist Hölder-stetig mit Hölder Exponenten $\gamma \in (0, 1]$. Hierbei ist $\gamma = 1$ für $\rho > L_f + C^*$, $\gamma = \frac{\rho}{L_f + C^*}$ für $\rho < L_f + C^*$ und $\gamma \in (0, 1)$ beliebig für $\rho = L_f + C^*$, wobei C^* eine Konstante ist, die von der Funktion Ψ abhängt mit $C^* = 0$ für $\Psi \equiv \mathrm{id}_{\mathbb{R}^n}$.

Beweis: Mit der Definition der optimalen Wertefunktion gilt für alle $x_1, x_2 \in W$:

$$\begin{aligned} |v_\rho(x_1) - v_\rho(x_2)| &= \left|\inf_{u(\cdot)\in\mathcal{U}} J_\rho(x_1, u(\cdot)) - \inf_{u(\cdot)\in\mathcal{U}} J_\rho(x_2, u(\cdot))\right| \\ &\leq \sup_{u(\cdot)\in\mathcal{U}} |J_\rho(x_1, u(\cdot)) - J_\rho(x_2, u(\cdot))| \\ &= \sup_{u(\cdot)\in\mathcal{U}} \left|\int_0^\infty e^{-\rho t}\Big(g(\varphi(t, x_1, u(\cdot)), u(t)) - g(\varphi(t, x_2, u(\cdot)), u(t))\Big)dt\right| \\ &\leq \int_0^\infty e^{-\rho t} \sup_{u(\cdot)\in\mathcal{U}} \underbrace{|g(\varphi(t, x_1, u(\cdot)), u(t)) - g(\varphi(t, x_2, u(\cdot)), u(t))|}_{=:\Phi(t)} \end{aligned}$$

Zum einen gilt nun wegen (2.9) die Abschätzung

$$\Phi(t) \leq 2M_g \tag{2.13}$$

zum anderen folgt aus der stetigen Abhängigkeit der Trajektorien vom Anfangswert

$$\Phi(t) \leq CL_g\|x_1 - x_2\|e^{(L_f + C^*)t}. \tag{2.14}$$

was aus Lemma A.8 (bewiesen im Anhang) folgt. Für $\Psi = \mathrm{id}_{\mathbb{R}^n}$ folgt diese Ungleichung mit $C = 1$, $C^* = 0$ direkt aus dem Gronwall-Lemma A.7.
Aus (2.13) und (2.14) folgt nun das Zwischenergebnis

$$\Phi(t) \leq \min\{CL_g\|x_1 - x_2\|e^{(L_f + C^*)t}, 2M_g\},$$

und hieraus mit Lemma 2.14 die Behauptung. □

Bemerkung 2.16 Wenn $W \neq \mathbb{R}^n$ ist und $\Psi \neq \mathrm{id}|_{\mathbb{R}^n}$, so läßt sich v_ρ hölder-stetig auf $\overline{W}$ erweitern, indem wir $v_\rho(x) = v_\rho(\Psi(x))$ für $x \in \partial W$ setzen. Der Beweis verläuft analog.

Korollar 2.17 Die Wertefunktion v_ρ ist in der Norm beschränkt, d.h. $\|v_\rho\|_{\mathcal{H}_\gamma} < \infty$, wobei γ der Hölder-Exponent von v_ρ ist.

Beweis: Bemerkung 2.8 sagt, daß $|v_\rho(x)| \leq \frac{M_g}{\rho} \ \forall x \in W$.
Mit der Definition der Norm auf $\mathcal{H}_\gamma$ und dem Satz 2.15 folgt die Behauptung. □

2.2.3 Die Bellman-Gleichung

Eine weitere wichtige Eigenschaft der Wertefunktion ist die Tatsache, daß sie dort, wo sie differenzierbar ist, eine bestimmte partielle Differentialgleichung erfüllt, die *Bellman-Gleichung*.
Diese werden wir im folgenden Unterabschnitt herleiten.

Lemma 2.18 Sei $F : U \to \mathbb{R}$ eine stetige Funktion auf $U \subset \mathbb{R}^n$. Es gelte

$$\sup_{u(\cdot)\in\mathcal{U}} \frac{1}{t} \int_0^t F(u(\tau))d\tau \geq C(t) \quad \forall u(\cdot) \in \mathcal{U}, t > 0. \tag{2.15}$$

Dann folgt $\sup\limits_{u\in U} F(u) \geq C(t)$.

Beweis: Aus (2.15) folgt wegen

$$\sup_{u(\cdot)\in\mathcal{U}} \frac{1}{t} \int_0^t F(u(\tau))d\tau \ \in \ \overline{co}\{F(u)|u \in U\}$$

die Ungleichung $\sup \overline{co}\{F(u)|u \in U\} \geq C(t)$, wobei *co* die konvexe Hülle bezeichnet und $\overline{co}$ deren Abschluß.
Zu vorgegebenem $\varepsilon > 0$ gibt es also $F_\varepsilon = \sum_{j=0}^{l} \alpha_j F(u_j) \geq C(t) - \varepsilon$, $u_j \in U$, $\sum_{j=0}^{l} \alpha_j = 1$. Wähle nun $u_\varepsilon \in U$ so, daß $F(u_\varepsilon) = \max_{j\in\{1,\ldots,l\}} F(u_j)$ gilt. Es folgt $F(u_\varepsilon) \geq C(t) - \varepsilon$, $u_\varepsilon \in U$. Da ε beliebig gewählt war, folgt die Behauptung. □

Satz 2.19 (Bellman-Gleichung)
Betrachte das durch (2.1) – (2.9) definierte diskontierte optimale Steuerungsproblem auf W zu $\rho > 0$. Die optimale Wertefunktion v_ρ sei differenzierbar im Punkt $x_0 \in W$. Dann gilt:

$$\sup_{u\in U}\{\rho v_\rho(x_0) - g(x_0, u) - Dv_\rho(x_0)f(x_0, u)\} = 0 \tag{2.16}$$

Beweis: Nach dem Bellman'schen Optimalitätsprinzip gilt für alle $t > 0$:

$$\sup_{u(\cdot)\in\mathcal{U}} \left\{ \frac{1}{t} \left(v_\rho(x_0) - e^{-\rho t} v_\rho(\varphi(t, x_0, u(\cdot))) \right) - \frac{1}{t} \int_0^t e^{-\rho\tau} g(\varphi(\tau, x_0, u(\cdot)), u(\tau)) d\tau \right\} = 0. \tag{2.17}$$

Die Beweisidee ist nun wie folgt: Wir leiten (2.17) ab und lassen $t \to 0$ gehen. Wir zeigen so in zwei Schritten die beiden Ungleichungen

$$-Dv_\rho(x_0) f(x_0, u) + \rho v_\rho(x_0) \le g(x_0, u) \tag{2.18}$$

und

$$\sup_{u\in U} \{-Dv_\rho(x_0) f(x_0, u) + \rho v_\rho(x_0) - g(x_0, u)\} \ge 0. \tag{2.19}$$

Sei $u \in U$ als konstante Kontrolle fest gewählt; $x(\cdot)$ bezeichne die zugehörige Trajektorie mit Anfangswert x_0, d.h. $x(\cdot) := \varphi(\cdot, x_0, u)$. Damit gilt:

$$\frac{d}{dt} x(t) = f(x(t), u), \quad x(0) = x_0$$

für genügend kleine $t > 0$, d.h. solange $x(t)$ in W liegt. Wegen (2.17) gilt für diese $t > 0$ also

$$\frac{1}{t} \left(v_\rho(x_0) - e^{-\rho t} v_\rho(x(t)) \right) \le \frac{1}{t} \int_0^t e^{-\rho\tau} g(x(\tau), u(\tau)) d\tau. \tag{2.20}$$

Da $x(t)$ differenzierbar ist in $t = 0$ und v_ρ differenzierbar ist in $x = x_0$, muß auch $\tilde{v}_\rho(t) := e^{-\rho t} v_\rho(x(t))$ in $t = 0$ differenzierbar sein. Da außerdem $\tilde{v}_\rho(0) = v_\rho(x_0)$ gilt, ergibt sich

$$\frac{d}{dt}\tilde{v}_\rho(0) = Dv_\rho(x_0)\frac{d}{dt}x(0) - \rho v_\rho(x_0) = Dv_\rho(x_0) f(x_0, u(0)) - \rho v_\rho(x_0), \tag{2.21}$$

also folgt (2.18) aus (2.20).

Für die umgekehrte Ungleichung betrachte Gleichung (2.17). Aus dieser folgt

$$\sup_{u(\cdot)\in\mathcal{U}} \left\{ \frac{1}{t} \left(v_\rho(x_0) - v_\rho(x_0 + \int_0^t f(x(\tau), u(\tau)) d\tau) \right) - \frac{1}{t} \int_0^t g(x_0, u(\tau)) d\tau + \rho v_\rho(x_0) \right\} \ge \alpha(t)$$

mit $\alpha(t) \to 0$ für $t \searrow 0$. Dies ergibt sich durch Ableiten der Ausdrücke unter Ausnutzung von (2.21) für hinreichend kleine $t > 0$.
Wegen der Differenzierbarkeit von v_ρ in x_0 gilt

$$v_\rho(x) = v_\rho(x_0) + Dv_\rho(x_0)(x - x_0) + \|x - x_0\|\delta(x),$$

wobei $\delta(x) \to 0$ für $x \to x_0$. Mit $x := x_0 + \int_0^t f(x(\tau), u(\tau)) d\tau$ erhalten wir

$$\sup_{u(\cdot)\in\mathcal{U}} \left\{ \frac{1}{t} \int_0^t -Dv_\rho(x_0) f(x(\tau), u(\tau)) d\tau - \frac{1}{t} \int_0^t g(x_0, u(\tau)) d\tau \right\} + \rho v_\rho(x_0) \ge \alpha(t) \to 0.$$

Also folgt auch

$$\sup_{u(\cdot)\in\mathcal{U}} \left\{ \frac{1}{t} \int_0^t -Dv_\rho(x_0) f(x_0, u(\tau)) - g(x_0, u(\tau)) d\tau \right\} + \rho v_\rho(x_0) \geq \alpha(t) \to 0.$$

Die Behauptung ergibt sich nun aus

$$F(\cdot) = F_{x_0}(\cdot) = -Dv_\rho(x_0) f(x_0, \cdot) - g(x_0, \cdot)$$

mit Lemma 2.18. □

2.3 Verallgemeinerte Lösungen der Bellman-Gleichung

Im Allgemeinen sind die Voraussetzungen von Satz 2.19 nicht erfüllt, da die optimale Wertefunktion v_ρ zwar stetig, nicht jedoch differenzierbar ist. Um dieses Problem zu lösen, müssen wir den Lösungsbegriff für partielle nichtlineare Differentialgleichungen erweitern. Dies führt zur Theorie der *Viskositätslösungen*.

2.3.1 Viskositätslösungen

In diesem Abschnitt wird - in aller Kürze - das Konzept der Viskositätslösungen nichtlinearer partieller Differentialgleichungen der Form

$$F(y, w(y), Dw(y)) = 0 \text{ für } y \in \mathcal{O} \tag{2.22}$$

vorgestellt. Hierbei ist $\mathcal{O} \subseteq \mathbb{R}^n$ offen, $F : \mathcal{O} \times \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ stetig und $Dw = (\frac{\partial w}{\partial x_1}, \ldots, \frac{\partial w}{\partial x_n})$ bezeichnet den Gradienten von w.
Die Viskositätslösungen sind verallgemeinerte Lösungen, die nicht unbedingt differenzierbar sein müssen, lediglich die Stetigkeit ist in der Definition vorausgesetzt.
Mit diesem Konzept ist es möglich, Aussagen über Existenz, Eindeutigkeit und Stabilität von Lösungen einer großen Klasse von Gleichungen der Form (2.22) aufzustellen. Dies ermöglicht uns dann, einen Existenz- und Eindeutigkeitssatz für Lösungen der Bellman-Gleichung aufzustellen.

In diesem Unterabschnitt werden wir zwei alternative Definitionen von Viskositätslösungen zur Gleichung (2.22) angeben und zeigen, daß diese Viskositätslösungen konsistent zu den klassischen Lösungen von (2.22) sind. Für detailliertere Informationen zu diesem und dem folgenden Unterabschnitt sei auf die Arbeiten von Crandall, Evans, Lions [7], Crandall, Lions [8], Elliott [9] und Lions [14] verwiesen.

Für die erste Definition der Viskositätslösungen erinnern wir uns an die Definition der Differenzierbarkeit von reellwertigen Funktionen mit höherdimensionalem Definitionsbereich.

Definition 2.20 Sei $\mathcal{O} \subseteq \mathbb{R}^n$ offen. Eine Funktion $w : \mathcal{O} \to \mathbb{R}$ heißt *differenzierbar* in $y_0 \in \mathcal{O}$, wenn ein $p_0 \in \mathbb{R}^n$ existiert, so daß gilt

$$\lim_{y \to y_0} \frac{w(y) - w(y_0) - p_0 \cdot (y - y_0)}{\|y - y_0\|} = 0 \tag{2.23}$$

Hier bezeichnet $\cdot$ das euklidische Skalarprodukt im $\mathbb{R}^n$.

Wenn so ein $p_0 \in \mathbb{R}^n$ existiert, so ist dies der eindeutig bestimmte Gradient $Dw(y_0)$ von w in y_0. (2.23) ist offensichtlich die Konjunktion der beiden Ungleichungen

$$\limsup_{y \to y_0} \frac{w(y) - w(y_0) - p_0 \cdot (y - y_0)}{\|y - y_0\|} \leq 0 \tag{2.24}$$

$$\liminf_{y \to y_0} \frac{w(y) - w(y_0) - p_0 \cdot (y - y_0)}{\|y - y_0\|} \geq 0 \tag{2.25}$$

Für eine reellwertige Funktion w, die stetig aber nicht unbedingt differenzierbar ist, gibt es Punkte, in denen (2.23) für kein $p_0 \in \mathbb{R}^n$ erfüllt ist, trotzdem können entweder (2.24) oder (2.25) für bestimmte Vektoren $p_0 \in \mathbb{R}^n$ erfüllt sein.

Beispiel 2.21 Betrachte die Funktion $w : \mathbb{R} \to \mathbb{R}$, $w(x) = |x|$, die in $x = 0$ nicht differenzierbar ist. (2.25) ist aber erfüllt für $p_0 \in [-1, 1]$.

Aus diesem Grunde ist die folgende Definition sinnvoll:

Definition 2.22 Sei $w : \mathcal{O} \to \mathbb{R}$ und $y_0 \in \mathcal{O}$.
Dann ist das *Superdifferential von* w im Punkte y_0 definiert durch

$$D^+w(y_0) := \{p_0 \in \mathbb{R}^n | p_0 \text{ erfüllt } (2.24)\}. \tag{2.26}$$

Analog ist das *Subdifferential von* w im Punkte y_0 definiert durch

$$D^-w(y_0) := \{p_0 \in \mathbb{R}^n | p_0 \text{ erfüllt } (2.25)\}. \tag{2.27}$$

Bemerkung 2.23 $D^+w(y_0)$ und $D^-w(y_0)$ sind konvex und abgeschlossen.
Die Konvexität ist durch Nachrechnen leicht zu sehen, die Abgeschlossenheit folgt aus der Stetigkeit des euklidischen Skalarproduktes im $\mathbb{R}^n$.

Mithilfe des Super- und Subdifferentials werden wir nun die Viskositätslösung der partiellen Differentialgleichung (2.22) definieren.

Definition 2.24 Eine stetige Funktion $w : \mathcal{O} \to \mathbb{R}$ heißt *Viskositätslösung* von (2.22), falls gilt:

$$F(y, w(y), p) \leq 0 \quad \forall y \in \mathcal{O}, \forall p \in D^+w(y) \tag{2.28}$$

$$\text{und } F(y, w(y), p) \geq 0 \quad \forall y \in \mathcal{O}, \forall p \in D^-w(y) \tag{2.29}$$

$w \in C(\mathcal{O})$ heißt *Viskositäts-Superlösung*, wenn w die Bedingung (2.28) erfüllt und *Viskositäts-Sublösung*, wenn (2.29) gilt.

Diese Art der Definition der Viskositätslösung ist recht anschaulich, da der Zusammenhang zu den klassischen Lösungen von (2.22) offensichtlich ist.
Zur Durchführung von Beweisen empfiehlt sich aber eine andere Definition.

Definition 2.25 $w \in C(\mathcal{O})$ heißt Viskositätslösung von (2.22), wenn für alle $\phi \in C^1(\mathcal{O})$ gilt:

$$\begin{aligned} w-\phi \text{ nimmt in } y_0 \in \mathcal{O} \text{ ein lokales Maximum an } &\implies F(y_0, w(y_0), D\phi(y_0)) \leq 0 \\ w-\phi \text{ nimmt in } y_0 \in \mathcal{O} \text{ ein lokales Minimum an } &\implies F(y_0, w(y_0), D\phi(y_0)) \geq 0 \end{aligned}$$

Mit dem folgenden Lemma können wir zeigen, daß die Definitionen (2.24) und (2.25) tatsächlich äquivalemt sind:

Lemma 2.26 Sei $w \in C(\mathcal{O})$, $y_0 \in \mathcal{O}$ und $p \in \mathbb{R}^n$. Dann sind äquivalent:

1. $p_0 \in D^+w(y_0)$ (bzw. $p_0 \in D^-w(y_0)$).
2. Es existiert $\phi \in C^1(\mathcal{O})$ mit $D\phi(y_0) = p_0$, so daß $w - \phi$ ein lokales Maximum (bzw. Minimum) in y_0 annimmt.

Beweis: Wir zeigen zuerst (1) $\Rightarrow$ (2):
Sei dazu $p_0 \in D^+w(y_0)$; daraus folgt nach Definition $\limsup_{y\to y_0} \frac{w(y)-w(y_0)-p_0\cdot(y-y_0)}{\|y-y_0\|} \leq 0$. Dann gibt es auf einer Umgebung $N = N(y_0)$ von y_0 eine auf $N\setminus\{y_0\}$ stetig differenzierbare Funktion und auf ganz N stetige Funktion ε mit $\varepsilon(y_0) = 0$ und

$$\frac{w(y) - w(y_0) - p_0 \cdot (y - y_0)}{\|y - y_0\|} \leq \varepsilon(y). \tag{2.30}$$

Um die Existenz dieser Funktion zu beweisen, betrachten wir die Ringe um y_0, die gegeben sind durch $R_n := B(y_0, \frac{1}{2^n}) \setminus B(y_0, \frac{1}{2^{n+1}})$. Die linke Seite von (2.30) ist auf allen Ringen R_n beschränkt durch Schranken r_n. Diese können so gewählt werden, daß $\lim_{n\to\infty} r_n = 0$ gilt, da der Limes superior der linken Seite von (2.30) für $y \to y_0$ kleiner oder gleich Null ist. Wir können so also eine Treppenfunktion $y \mapsto r_n$, $y \in R_n$ definieren, die in y_0 durch 0 stetig fortgesetzt werden kann. Durch differenzierbares Abschneiden zwischen den einzelnen Treppenstufen kann so die oben angegebene Funktion ε konstruiert werden.
(2.30) kann also umformuliert werden zu

$$w(y) - w(y_0) - p_0 \cdot (y - y_0) \leq \varepsilon\|y - y_0\| \tag{2.31}$$

Setzt man nun $\phi(y) := w(y_0)+p_0\cdot y+\varepsilon(y)\|y-y_0\|$, so ist ϕ differenzierbar mit $D\phi(y_0) = p_0$. Wegen (2.31) kann man nun abschätzen

$$w(y) - \phi(y) = w(y) - w(y_0) - p \cdot y - \varepsilon\|y - y_0\| \leq -p \cdot y_0 = w(y_0) - \phi(y_0).$$

Diese Abschätzung gilt für alle $y \in N$, also hat $w - \phi$ ein lokales Maximum in y_0.

(2) $\Rightarrow$ (1):
Sei $\phi \in C^1(\mathcal{O})$ und $w - \phi$ habe ein lokales Maximum in y_0. Dann gilt auf einer Umgebung $N = N(y_0)$ von y_0

$$w(y) - \phi(y) \leq w(y_0) - \phi(y_0),$$

also

$$w(y) - w(y_0) - (\phi(y) - \phi(y_0)) \leq 0.$$

Wenn wir ϕ in x_0 als Taylorreihe entwickeln und das erste Glied betrachten, so erhalten wir auf N:

$$w(y) - w(y_0) - D\phi(y_0) \cdot (y - y_0) + o(\|y - y_0\|) \leq 0.$$

Für $y \neq y_0$ ergibt sich somit

$$\frac{w(y) - w(y_0) - D\phi(y_0) \cdot (y - y_0)}{\|y - y_0\|} \leq \frac{o(\|y - y_0\|)}{\|y - y_0\|},$$

und daher auch

$$\limsup_{y \to y_0} \frac{w(y) - w(y_0) - D\phi(y_0) \cdot (y - y_0)}{\|y - y_0\|} \leq \lim_{y \to y_0} \frac{o(\|y - y_0\|)}{\|y - y_0\|} = 0,$$

weswegen $D\phi(y_0) \in D^+w(y_0)$ gilt.
Der Beweis für $D^-w(y_0)$ läuft in beiden Richtungen analog. □

Mit diesem Lemma ist sofort ersichtlich, daß die beiden Definitionen der Viskositätslösungen äquivalent sind.

Satz 2.27 w ist genau dann eine Viskositätslösung im Sinne von Definition (2.24), wenn w eine Viskositätslösung im Sinne von Definition (2.25) ist.

Aus dem Lemma ergibt sich zudem noch eine weitere Folgerung

Proposition 2.28 Ist w in $y_0 \in \mathbb{R}^n$ differenzierbar, so gilt:
$D^+w(y_0) = D^-w(y_0) = \{Dw(y_0)\}$

Beweis: Klar ist nach Definition, daß $Dw(y_0) \in D^+w(y_0), D^-w(y_0)$.
Umgekehrt sei (o.B.d.A.) $p_0 \in D^+w(y_0)$ und w differenzierbar in y_0. Dann folgt mit dem Lemma (2.26) die Existenz einer Funktion $\phi \in C^1(\mathcal{O})$ mit $D\phi(y_0) = p_0$, so daß $w - \phi$ ein lokales Minimum in y_0 annimmt. Wegen der Differenzierbarkeit von w in y_0 ist auch $w - \phi$ in y_0 differenzierbar und es gilt $D(w - \phi)(y_0) = 0 \Rightarrow Dw(y_0) = D\phi(y_0)$. Also ist $p_0 = Dw(y_0)$ und damit wegen der Eindeutigkeit des Gradienten das einzige Element von $D^+w(y_0)$. □

Der folgende Satz zeigt, daß der Begriff der Viskositätslösung eine Verallgemeinerung des klassischen Lösungsbegriffes darstellt, d.h. klassische Lösungen sind auch Viskositätslösungen und umgekehrt sind differenzierbare Viskositätslösungen auch klassische Lösungen.

Satz 2.29 Ist $w \in C^1(\mathcal{O})$ eine klassische Lösung von (2.22), d.h.

$$F(y, w(y), Dw(y)) = 0 \quad y \in \mathcal{O},\ w \in C^1(\mathcal{O}),$$

dann ist w auch eine Viskositätslösung von (2.22).
Ist umgekehrt w eine Viskositätslösung von (2.22) und in $y_0 \in \mathcal{O}$ differenzierbar, so gilt

$$F(y_0, w(y_0), Dw(y_0)) = 0.$$

Beweis: Folgt sofort aus Proposition 2.28. □

2.3.2 Existenz und Eindeutigkeit

Der Grund, warum wir Viskositätslösungen betrachten, liegt darin, daß die optimale Wertefunktion v_ρ als Viskositätslösung der Bellman-Gleichung charakterisiert werden kann und darüber hinaus durch diese eindeutig bestimmt ist.
Diese Aussage soll nun zum Abschluß dieses Kapitels bewiesen werden.

Satz 2.30 Es seien v_1, $v_2 \in C(\mathbb{R}^n)$ gleichmäßig stetig und beschränkt.
Sei $U \subseteq \mathbb{R}^m$ kompakt und f, g_1, $g_2 \in C(\mathbb{R}^n \times U)$ erfüllen die Bedingungen (2.4) – (2.9), d.h. sie seien Lipschitz-stetig mit Lipschitz-Konstante L und beschränkt durch eine Konstante M.
Seien nun $G_i \in C(\mathbb{R}^n \times U)$, $i = 1, 2$ gegeben durch

$$G_i(x, p) = \sup_{u \in U} \{g_i(x, u) + p \cdot f(x, u)\}.$$

Ist v_i eine Viskositätslösung von $v_i(x) + G_i(x, Dv_i(x)) = 0$, $i = 1, 2$, so gilt

$$\|v_1 - v_2\|_\infty \leq \sup_{x \in \mathbb{R}^n, u \in U} |g_1(x, u) - g_2(x, u)|. \tag{2.32}$$

Beweis: Sei $0 < \delta \leq 1$. Wähle $y_0 = y_0(\delta) \in \mathbb{R}^n$ mit

$$v_1(y_0) - v_2(y_0) \geq \sup_{x \in \mathbb{R}^n} \{v_1(x) - v_2(x)\} - \delta.$$

Sei nun $\Phi = \Phi_{\varepsilon,\delta}$ definiert durch

$$\begin{aligned} \Phi : \mathbb{R}^n \times \mathbb{R}^n &\to \mathbb{R} \\ (x, y) &\mapsto v_1(x) - v_2(y) - \left\| \frac{x - y}{\varepsilon} - y_0 \right\|^2 - \|y - y_0\|^2. \end{aligned}$$

Dann gilt

$$\begin{aligned} \Phi(\varepsilon y_0 + y_0, y_0) &= v_1(\varepsilon y_0 + y_0) - v_2(y_0) - \|\varepsilon y_0\|^2 \\ &= v_1(\varepsilon y_0 + y_0) - v_1(y_0) + v_1(y_0) - v_2(y_0) - \|\varepsilon y_0\|^2 \\ &\geq \sup_{x \in \mathbb{R}^n} \{v_1(x) - v_2(x)\} - \delta - \omega(\|\varepsilon y_0\|) - \|\varepsilon y_0\|^2 \end{aligned}$$

und

$$\begin{aligned} \Phi(x, y) &\leq v_1(x) - v_2(x) + |v_2(x) - v_2(y)| - \left\| \frac{x - y}{\varepsilon} - y_0 \right\|^2 - \|y - y_0\|^2 \\ &\leq v_1(x) - v_2(x) + \omega(\|x - y\|) - \left\| \frac{x - y}{\varepsilon} - y_0 \right\|^2 - \|y - y_0\|^2, \end{aligned}$$

wobei $\omega(\|x - y\|)$ ein *Stetigkeitsmodul* der gleichmäßig stetigen Funktionen v_i, $i = 1, 2$ ist, d.h.:

$$|v_i(x) - v_i(y)| \leq \omega(\|x - y\|), \quad i = 1, 2$$

für $x, y \in \mathbb{R}^n$ mit $\omega(\|x-y\|) \to 0$ für $\|x-y\| \to 0$. Da die v_i beschränkt sind, kann sicher auch ω beschränkt gewählt werden.

Sei nun $\Phi(x,y) \geq \Phi(\varepsilon y_0 + y_0, y_0)$. Dann folgt

$$\omega(\|x-y\|) - \left\|\frac{x-y}{\varepsilon} - y_0\right\|^2 - \|y-y_0\|^2 \geq -\delta - \omega(\|\varepsilon y_0\|) - \|\varepsilon y_0\|^2$$

also

$$\left\|\frac{x-y}{\varepsilon} - y_0\right\|^2 + \|y-y_0\|^2 \leq \omega(\|x-y\|) + \delta + \omega(\|\varepsilon y_0\|) + \|\varepsilon y_0\|^2.$$

Wir wählen im Folgenden $\varepsilon > 0$ so klein, daß $\|\varepsilon y_0\|^2 \leq \delta$ gilt. Dann folgt

$$\left\|\frac{x-y}{\varepsilon} - y_0\right\|^2 + \|y-y_0\|^2 \leq \omega(\|x-y\|) + 2\delta + \omega(\|\varepsilon y_0\|). \tag{2.33}$$

Wegen der Beschränktheit von ω existiert nun ein $r > 0$, so daß (2.33) für x, y mit $x, y \notin B_r(y_0)$ nicht gelten kann. Also gilt

$$\Phi(x,y) \leq \Phi(\varepsilon y_0 + y_0, y_0) \quad \forall x, y \in \mathbb{R}^n \setminus B_r(y_0).$$

Daher gibt es $x_1, y_1 \in \mathbb{R}^n$, so daß Φ in (x_1, y_1) sein Maximum annimmt. Desweiteren folgt aus (2.33), daß für hinreichend kleine $\varepsilon > 0$ ein $m > 0$ existiert mit

$$\|x_1 - y_1\| \leq m\varepsilon, \quad m \text{ unabhängig von } \varepsilon, \delta, \tag{2.34}$$

da sonst die linke Seite von (2.33) unbeschränkt wäre für $\varepsilon \to 0$.
Betrachte jetzt die Abbildungen

$$\begin{aligned} x &\mapsto \Phi(x, y_1) = v_1(x) - \left(v_2(y_1) + \left\|\frac{x-y_1}{\varepsilon} - y_0\right\|^2 + \|y_1 - y_0\|^2\right) \\ y &\mapsto -\Phi(x_1, y) = v_2(y) - \left(v_1(x_1) - \left\|\frac{x_1-y}{\varepsilon} - y_0\right\|^2 - \|y-y_0\|^2\right) \end{aligned}$$

Diese Abbildungen nehmen ihr Maximum bzw. Minimum in $x = x_1$ bzw. $y = y_1$ an.
Mit der Viskositätseigenschaft aus Definition 2.25 folgt:

$$v_1(x_1) + G_1\left(x_1, 2\left(\frac{x_1-y_1}{\varepsilon} - y_0\right)\frac{1}{\varepsilon}\right) \leq 0$$

und

$$v_2(y_1) + G_2\left(y_1, 2\left(\frac{x_1-y_1}{\varepsilon} y_0\right)\frac{1}{\varepsilon} - 2(y_1 - y_0)\right) \geq 0.$$

Mit

$$p := 2\left(\frac{x_1-y_1}{\varepsilon} - y_0\right)\frac{1}{\varepsilon}, \quad q := -2(y_1 - y_0),$$

folgt also

$$\begin{aligned} v_1(x_1) + G_1(x_1, p) &\leq 0 \\ v_2(y_1) + G_2(y_1, p+q) &\geq 0. \end{aligned}$$

Diese beiden Ungleichungen liefern

$$\begin{aligned} v_1(x_1) - v_2(y_1) &\leq G_2(y_1, p+q) - G_2(x_1, p) + G_2(x_1, p) - G_1(x_1, p) \\ &\leq \sup_{x\in\mathbb{R}^n, u\in U} \Big(g_2(x,u) - g_1(x,u)\Big) + L\|y_1 - x_1\| + L\|y_1 - x_1\|\,\|p\| + \sup_{u\in U}\|f(y_1,u)\|\,\|q\|. \end{aligned}$$

Die Bedingungen (2.33) und (2.34) liefern nun

$$\|q\| = 2\|y_1 - y_0\| \leq 2\sqrt{\omega(\|x_1 - y_1\|) + 2\delta + \omega(\varepsilon\|y_0\|)} \leq 2\sqrt{\omega(m\varepsilon) + 2\delta + \omega(\varepsilon\|y_0\|)}$$

sowie

$$\|x_1 - y_1\|\,\|p\| \leq 2m\varepsilon \left\| \frac{x_1 - y_1}{\varepsilon} - y_0 \right\| \frac{1}{\varepsilon} \leq 2m\sqrt{\omega(m\varepsilon) + 2\delta + \omega(\varepsilon\|y_0\|)}.$$

Es folgt also

$$v_1(x_1) - v_2(y_1) \leq \sup_{x\in\mathbb{R}^n, u\in U} (g_2(x,u) - g_1(x,u)) + g_{\varepsilon,\delta},$$

mit $g_{\varepsilon,\delta} := Lm\varepsilon + 2\sqrt{\omega(m\varepsilon) + 2\delta + \omega(\varepsilon\|y_0\|)}\Big(Lm + M\Big) \to 0$ für $\varepsilon, \delta \to 0$. Also gilt für alle $0 < \delta \leq 1$

$$\begin{aligned} \sup_{x\in\mathbb{R}^n}(v_1(x) - v_2(x)) &\leq \Phi(\varepsilon y_0 + y_0, y_0) + \delta + \omega(\|\varepsilon y_0\|) + \|\varepsilon y_0\|^2 \\ &\leq \Phi(x_1, y_1) + \delta + \omega(\|\varepsilon y_0\|) + \|\varepsilon y_0\|^2 \\ &\leq v_1(x_1) - v_2(y_1) + \delta + \omega(\|\varepsilon y_0\|) + \|\varepsilon y_0\|^2 \\ &\leq \sup_{u\in U, x\in\mathbb{R}^n} (g_2(x,u) - g_1(x,u)) + g_{\varepsilon,\delta} + \delta + \omega(\|\varepsilon y_0\|) + \|\varepsilon y_0\|^2 \end{aligned}$$

Für $\varepsilon \to 0$, $\delta \to 0$ ergibt sich nun die Behauptung. □

Wenn wir mit Hilfe der Funktion Ψ aus Abschnitt 2.1 eine Randbedingung definieren, können wir die gleiche Aussage auch für Viskositätslösungen auf beschränkten offenen Teilmengen $W \subset \mathbb{R}^n$ beweisen.

Satz 2.31 Es seien v_1, $v_2 \in C(\overline{W})$ gleichmäßig stetig und beschränkt auf einer beschränkten offenen Teilmenge $W \subset \mathbb{R}^n$.
Sei $U \subseteq \mathbb{R}^m$ kompakt und f, g_1, $g_2 \in C(W \times U)$ erfüllen die Bedingungen (2.4) – (2.9), d.h. sie seien auf W Lipschitz-stetig mit Lipschitz-Konstante L und beschränkt durch eine Konstante M. Weiterhin sei $\Psi : \mathbb{R}^n \to W$ wie in Abschnitt 2.1 definiert.
Seien nun $G_i \in C(W \times U)$, $i = 1, 2$ gegeben durch

$$G_i(x,p) = \sup_{u\in U}\{g_i(x,u) + p \cdot f(x,u)\}.$$

Ist v_i eine Viskositätslösung von $v_i(x) + G_i(x, Dv_i(x)) = 0$ mit $v_i(x) = v_i(\Psi(x))$ $\forall x \in \partial W$, $i = 1, 2$, so gilt

$$\|v_1 - v_2\|_\infty \leq \sup_{x\in W, u\in U} |g_1(x,u) - g_2(x,u)|. \tag{2.35}$$

Beweis: Wähle $y_0 \in \overline{W}$ mit

$$v_1(y_0) - v_2(y_0) = \sup_{x \in \overline{W}} \{v_1(x) - v_2(x)\}.$$

Dieses existiert, da $\overline{W}$ kompakt ist und kann wegen der Randbedingung an die v_i so gewählt werden, daß es in W liegt. Sei nun $\Phi = \Phi_\varepsilon$ definiert durch

$$\begin{aligned} \Phi : \mathbb{R}^n \times \mathbb{R}^n &\to \mathbb{R} \\ (x, y) &\mapsto v_1(x) - v_2(y) - \left\| \frac{x-y}{\varepsilon} - y_0 \right\|^2 - \|y - y_0\|^2. \end{aligned}$$

Dann gilt analog zum Beweis von Satz 2.30

$$\begin{aligned} \Phi(\varepsilon y_0 + y_0, y_0) &= v_1(\varepsilon y_0 + y_0) - v_2(y_0) - \|\varepsilon y_0\|^2 \\ &= v_1(\varepsilon y_0 + y_0) - v_1(y_0) + v_1(y_0) - v_2(y_0) - \|\varepsilon y_0\|^2 \\ &\geq \sup_{x \in \overline{W}} \{v_1(x) - v_2(x)\} - \omega(\|\varepsilon y_0\|) - \|\varepsilon y_0\|^2 \end{aligned}$$

und

$$\begin{aligned} \Phi(x, y) &\leq v_1(x) - v_2(x) + |v_2(x) - v_2(y)| - \left\| \frac{x-y}{\varepsilon} - y_0 \right\|^2 - \|y - y_0\|^2 \\ &\leq v_1(x) - v_2(x) + \omega(\|x - y\|) - \left\| \frac{x-y}{\varepsilon} - y_0 \right\|^2 - \|y - y_0\|^2, \end{aligned}$$

wobei $\omega(\|x-y\|)$ wieder ein beschränktes Stetigkeitsmodul der gleichmäßig stetigen Funktionen v_i, $i = 1, 2$ ist.

Sei nun $\Phi(x, y) \geq \Phi(\varepsilon y_0 + y_0, y_0)$. Dann folgt

$$\omega(\|x - y\|) - \left\| \frac{x-y}{\varepsilon} - y_0 \right\|^2 - \|y - y_0\|^2 \geq -\omega(\|\varepsilon y_0\|) - \|\varepsilon y_0\|^2$$

also

$$\left\| \frac{x-y}{\varepsilon} - y_0 \right\|^2 + \|y - y_0\|^2 \leq \omega(\|x - y\|) + \omega(\|\varepsilon y_0\|) + \|\varepsilon y_0\|^2.$$

Wegen der Beschränktheit von ω folgt, daß ein $m > 0$ existiert mit

$$\|x - y\| \leq m\varepsilon, \quad m \text{ unabhängig von } \varepsilon,$$

für alle x, y, die diese Ungleichung erfüllen, da sonst die linke Seite unbeschränkt wäre für $\varepsilon \to 0$.

Also folgt auch, daß die rechte Seite für $\varepsilon \to 0$ gegen Null geht, weshalb die Ungleichung nur für x, y aus einer Umgebung $B_r(y_0)$ von y_0 erfüllt sein kann mit $r \to 0$ für $\varepsilon \to 0$. Deshalb nimmt Φ für hinreichend kleine $\varepsilon > 0$ sein Maximum in $x_1, y_1 \in B_r(y_0) \subset W$ an. Nun können wir fortfahren wie im Beweis zu Satz 2.30, wenn wir dort $\delta = 0$ setzen. □

Wir kommen nun zum Hauptresultat dieses Abschnittes, dem Existenz- und Eindeutigkeitssatz für die Lösung der Bellman-Gleichung.

Satz 2.32 Sei $W \subset \mathbb{R}^n$ offen und beschränkt oder $W = \mathbb{R}^n$ und es sei ein Kontrollsystem wie in Abschnitt 2.1 definiert auf W gegeben.
Dann ist die Wertefunktion v_ρ die eindeutig bestimmte beschränkte und gleichmäßig stetige Viskositätslösung der Bellman-Gleichung

$$\sup_{u \in U} \Big(\rho v_\rho(x) - g(x,u) - Dv_\rho(x) \cdot f(x,u) \Big) = 0,$$

die $v_\rho(x) = v_\rho(\Psi(x))$ $\forall x \in \partial W$ erfüllt. Für $W = \mathbb{R}^n$ und $\Psi \equiv \mathrm{id}|_{\mathbb{R}^n}$ gilt die Aussage ohne die Randbedingung.

Beweis: Wir definieren zunächst $H(x,t,p) := \sup_{u \in U}\{\rho t - g(x,u) - p \cdot f(x,u)\}$.
Der Beweis gliedert sich in drei Teile. Im ersten Teil des Beweises soll gezeigt werden, daß für alle $x \in W$ gilt:

Für alle $\phi \in C^1(W,\mathbb{R})$ mit der Eigenschaft $v_\rho - \phi$ hat ein lokales Maximum in x, gilt $H(x, v_\rho(x), D\phi(x)) \leq 0$.

Im zweiten Teil zeigen wir für alle $x \in W$:

Für alle $\phi \in C^1(W,\mathbb{R})$ mit der Eigenschaft $v_\rho - \phi$ hat ein lokales Minimum in x, gilt $H(x, v_\rho(x), D\phi(x)) \geq 0$.

Im dritten Teil werden wir die Eindeutigkeit beweisen.

Teil 1: Sei $x \in W$ fest gewählt. O.B.d.A. sei $D^+ v_\rho \neq \emptyset$, da sonst nichts zu zeigen ist. Sei nun $\phi \in C^1(W,\mathbb{R})$ gegeben, so daß $v_\rho - \phi$ ein lokales Maximum in x hat.
Ohne Einschränkung können wir $v_\rho(x) = \phi(x)$ wählen; ansonsten ersetzen wir ϕ durch $\tilde{\phi}(y) := \phi(y) + (v_\rho(x) - \phi(x))$. Für beliebige konstante Kontrollen $u(\cdot) \equiv u$ und Zeiten $t > 0$ folgt aus (2.10)

$$v_\rho(x) \leq \int_0^t e^{-\rho\tau} g(x_u(\tau), u) d\tau + e^{-\rho t} v_\rho(x_u(t)),$$

mit $x_u(t) := \varphi(t,x,u)$. Da $v_\rho(x) - \phi(x) = 0$ ein lokales Maximum ist, folgt

$$v_\rho(y) \leq \phi(y) \quad \forall y \in N,$$

wobei $N = N(x) \subset W$ eine Umgebung von x ist.
Wegen der Beschränktheit von f existiert ein $t_1 > 0$, so daß für alle $t \in (0, t_1)$ gilt $x_u(t) \in N$, d.h. es gilt für $t \in (0, t_1)$

$$\begin{aligned} \phi(x) = v_\rho(x) &\leq \int_0^t e^{-\rho\tau} g(x_u(\tau), u) d\tau + e^{-\rho t} v_\rho(x_u(t)) \\ &\leq \int_0^t e^{-\rho\tau} g(x_u(\tau), u) d\tau + e^{-\rho t} \phi(x_u(t)), \end{aligned}$$

also

$$\begin{aligned} 0 &\leq \int_0^t e^{-\rho\tau} g(x_u(\tau), u) d\tau + e^{-\rho t}\phi(x_u(t)) - \phi(x) \\ \Rightarrow 0 &\leq \frac{\int_0^t e^{-\rho\tau} g(x_u(\tau), u) d\tau + e^{-\rho t}\phi(x_u(t)) - \phi(x)}{t}. \end{aligned}$$

Für $t \to 0$ ergibt sich also

$$\begin{aligned} 0 &\leq g(x,u) - \rho\phi(x) + D\phi(x) \cdot f(x,u) \\ \Rightarrow 0 &\geq \rho\phi(x) - g(x,u) - D\phi(x) \cdot f(x,u) \\ (v_\rho(x) = \phi(x)) \Rightarrow 0 &\geq \rho v_\rho(x) - g(x,u) - D\phi(x) \cdot f(x,u). \end{aligned}$$

Da diese Abschätzung für alle $u \in U$ gilt, muß auch

$$0 \geq \sup_{u \in U} \{\rho v_\rho(x) - g(x,u) - D\phi \cdot f(x,u)\} = H(x, v(x), D\phi(x))$$

gelten, und somit ist Teil 1 bewiesen.

Teil 2: Sei wiederum $x \in W$ fest gewählt und o.B.d.A. $D^- v_\rho(x) \neq \emptyset$. Weiterhin sei $\phi \in C^1(W, \mathbb{R})$ gegeben, so daß $v_\rho - \phi$ in x ein lokales Minimum hat.
Wähle wie oben o.E. ϕ derart, daß $v_\rho(x) = \phi(x)$ gilt. Dann gilt ebenfalls analog zu oben

$$v_\rho(y) \geq \phi(y) \quad \forall y \in N = N(x) \subset W.$$

Die Ungleichung $H(x, v_\rho(x), D\phi(x)) \geq 0$ ist sicherlich erfüllt, wenn ein $u \in U$ existiert mit

$$\rho v_\rho(x) - g(x,u) - D\phi(x) \cdot f(x,u) \geq 0.$$

Wähle $t_2 > 0$, so daß $x_u(t) \in N$ für alle $u \in U$ und alle $t \in (0, t_2)$. (x_u wie oben definiert.)
Definiere zu $t_n := \frac{1}{n}$, $\varepsilon_n := \frac{1}{n^2}$ eine Folge $(u_n(\cdot))_{n \in \mathbb{N}} \subset \mathcal{U}$ mit

$$v_\rho(x) + \varepsilon_n \geq \int_0^{t_n} e^{-\rho\tau} g(\varphi(\tau, x, u_n(\cdot)), u_n(\tau)) d\tau + e^{-\rho t_n} v_\rho(\varphi(t_n, x, u_n(\cdot))),$$

wobei $n \geq n_0 \geq \frac{1}{t_2}$. Eine solche Folge existiert nach dem Optimalitätsprinzip (2.10).
Da $v_\rho(x) = \phi(x)$ und $v_\rho(\varphi(t, x, u_n(\cdot))) \geq \phi(\varphi(t, x, u_n(\cdot)))$ für alle $t \in (0, t_2)$ gilt somit

$$\phi(x) + \frac{1}{n^2} \geq \int_0^{\frac{1}{n}} e^{-\rho\tau} g(\varphi(\tau, x, u_n(\cdot)), u_n(\tau)) d\tau + e^{-\frac{\rho}{n}} \phi(\varphi(\frac{1}{n}, x, u_n(\cdot))),$$

also

$$\phi(x) - e^{-\frac{\rho}{n}} \phi(\varphi(\frac{1}{n}, x, u_n(\cdot))) - \int_0^{\frac{1}{n}} e^{-\rho\tau} g(\varphi(\tau, x, u_n(\cdot)), u_n(\tau)) d\tau \geq -\frac{1}{n^2}.$$

Mit dem Hauptsatz der Differential- und Integralrechnung folgt nun

$$\int_0^{\frac{1}{n}} \Big(\rho\phi(\varphi(\tau,x,u_n(\cdot))) - D\phi(\varphi(\tau,x,u_n(\cdot))) \cdot f(\varphi(\tau,x,u_n(\cdot)),u_n(\tau)) \\ -g(\varphi(\tau,x,u_n(\cdot)),u_n(\tau))\Big)e^{-\rho\tau}d\tau \quad \geq \quad -\frac{1}{n^2},$$

also

$$n\int_0^{\frac{1}{n}} -\Big(\rho\phi(\varphi(\tau,x,u_n(\cdot))) - D\phi(\varphi(\tau,x,u_n(\cdot))) \cdot f(\varphi(\tau,x,u_n(\cdot)),u_n(\tau)) \\ -g(\varphi(\tau,x,u_n(\cdot)),u_n(\tau))\Big)e^{-\rho\tau} \quad \leq \quad \frac{1}{n}.$$

Der Integrand ist auf $\overline{N \times U}$ gleichmäßig stetig. Wegen der Beschränktheit von f auf $\overline{N \times U}$ wird der Abstand $\|x - \varphi(\frac{1}{n}, x, u_n(\cdot))\|$ für alle $u_n(\cdot) \in \mathcal{U}$ gleichmäßig durch eine gegen 0 konvergierende Folge beschränkt.
Daher gibt es für alle $\varepsilon > 0$ ein $n(\varepsilon)$, so daß für alle $n \geq \max\{n(\varepsilon), n_0\}$ gilt:

$$n\int_0^{\frac{1}{n}} -\Big(\rho\phi(x) - D\phi(x)\cdot f(x,u_n(\tau)) - g(x,u_n(\tau))\Big)e^{-\rho\tau}d\tau \leq \frac{1}{n} + \varepsilon.$$

Weiterhin gilt

$$n\int_0^{\frac{1}{n}} -\Big(\rho\phi(x) - D\phi(x)\cdot f(x,u_n(\tau)) - g(x,u_n(\tau))\Big)e^{-\rho\tau}d\tau$$

$$= n\int_0^{\frac{1}{n}} -\Big(\rho\phi(x) - D\phi(x)\cdot f(x,u_n(\tau)) - g(x,u_n(\tau))\Big)\Big(1-(1-e^{-\rho\tau})\Big)d\tau$$

und

$$\lim_{n\to\infty} n\int_0^{\frac{1}{n}} 1 - e^{-\rho\tau}d\tau = 0.$$

Also existiert eine Folge $(d_n)_{n\in\mathbb{N}}$ mit $\lim_{n\to\infty} d_n = 0$, so daß für alle $n \geq \max\{n(\varepsilon), n_0\}$ gilt

$$n\int_0^{\frac{1}{n}} -\Big(\rho\phi(x) - D\phi(x)\cdot f(x,u_n(\tau)) - g(x,u_n(\tau))\Big)d\tau \leq d_n.$$

Es bleibt nun noch ein $u \in U$ zu konstruieren, das diese Ungleichung erfüllt. Betrachte dazu

$$g_n := n\int_0^{\frac{1}{n}} g(x,u_n(\tau))d\tau, \quad f_n := n\int_0^{\frac{1}{n}} f(x,u_n(\tau))d\tau.$$

Für genügend großes $n_1 \in \mathbb{N}$ und $n \geq n_1$ gilt

$$(f_n, g_n) \in co\{(f(x,u), g(x,u)|u \in U\}.$$

Da U kompakt ist, ist die konvexe Hülle ebenfalls kompakt; d.h. es existiert eine konvergente Teilfolge (f_{n_k}, g_{n_k}) mit

$$\lim_{k\to\infty}(f_{n_k}, g_{n_k}) = (\tilde{f}, \tilde{g}) \in co\{(f(x,u), g(x,u)|u \in U\}.$$

Insbesondere folgt

$$-\Big(\rho\phi(x) - D\phi(x)\cdot\tilde{f} - \tilde{g}\Big) \leq 0.$$

Da $(\tilde{f}, \tilde{g}) \in co\{(f(x,u), g(x,u))|u \in U\}$ existieren α_i, $i = 1, 2, \ldots, l$ mit

$$0 \leq \alpha_i \leq 1, \quad \sum_{i=1}^{l}\alpha_i = 1 \text{ und}$$

$$\tilde{g} = \sum_{i=1}^{l}\alpha_i g(x,u_i), \quad \tilde{f} = \sum_{i=1}^{l}\alpha_i f(x,u_i).$$

Wähle nun $i \in \{1, \ldots, l\}$, so daß

$$D\phi(x)\cdot f(x,u_i) + g(x,u_i) = \max_{j\in\{1,\ldots,l\}}\{D\phi(x)\cdot f(x,u_i) + g(x,u_i)\}.$$

Dann gilt

$$-\Big(\rho\phi(x) - D\phi(x)\cdot f(x,u_i(t)) - g(x,u_i(t))\Big) \leq 0,$$

also

$$\Big(\rho\phi(x) - D\phi(x)\cdot f(x,u_i(t)) - g(x,u_i(t))\Big) \geq 0.$$

Somit ist Teil 2 ebenfalls bewiesen.

Teil 3: Die Eindeutigkeit folgt mit Satz 2.30 bzw. 2.31. Aus Teil 1 und 2 folgt, daß v_ρ eine Viskositätslösung der Gleichung

$$v_\rho(x) + \sup_{u\in U}\left\{-\frac{g(x,u)}{\rho} - \frac{Dv_\rho(x)\cdot f(x,u)}{\rho}\right\} = 0 \tag{2.36}$$

ist. Die Funktionen $\frac{f}{\rho}$, $\frac{g}{\rho}$, $\frac{H}{\rho}$ und v_ρ erfüllen die Voraussetzungen von Satz 2.30 bzw. 2.31 für beliebiges $\rho > 0$. Ist also w eine weitere beschränkte Lösung der Bellman-Gleichung, so löst w auch (2.36) und es folgt

$$\|v_\rho - w\|_\infty \leq \sup_{x\in W, u\in U}\left|\frac{g(x,u)}{\rho} - \frac{g(x,u)}{\rho}\right| = 0,$$

also die Eindeutigkeit. □

Bemerkung 2.33 Diese Aussage gilt auch, falls das System auf einer invarianten Teilmenge $\mathcal{O} \subset \mathbb{R}^n$ definiert ist. Für jede beliebige Lipschitz-stetige und beschränkte Erweiterung von f und g auf $\mathbb{R}^n$ erhalten wir die Aussage; wegen der Invarianz von $\mathcal{O}$ ist $v_\rho|_\mathcal{O}$ aber unabhängig von diesen Erweiterungen, weshalb die Eindeutigkeit auf $\mathcal{O}$ folgt.

Kapitel 3

Konvergenz der Wertefunktion für $\rho \to 0$

Im letzten Kapitel haben wir uns mit dem diskontierten optimalen Steuerungsproblem beschäftigt. Obwohl dabei ein Problem auf unendlichem Zeitintervall betrachtet wird, genügt es zur Betrachtung von ε-Optimalität zu vorgegebenem $\varepsilon > 0$ immer, endliche Zeitintervalle zu betrachten, da aus der Beschränktheit der Zielfunktion sofort folgt, daß $\int_T^\infty e^{-\rho t} g(t)dt < \varepsilon$ ist für hinreichend großes $T > 0$.

Wir wollen nun zusätzlich $\limsup_{T\to\infty} \frac{1}{T} \int_0^T g(\varphi(t, x, u(\cdot)), u(t))dt$, das *Durchschnittskostenfunktional* betrachten. Es stellt sich die Frage, in welcher Weise das diskontierte Zielfunktional eine Näherung für das Durchschnittskostenfunktional darstellt.

Wir werden hier zunächst einige technische Voraussetzungen zeigen, unter denen sich Konvergenz zeigen läßt, die uns dann zu einem allgemeineren Resultat führen. Hierzu werden wir den Begriff der *Kontrollmengen* einführen, der auch in folgenden Kapiteln eine große Rolle spielen wird. Die Überlegungen hierzu gehen im Wesentlichen auf Wirth [17] zurück.

Am Ende des Kapitels werden wir noch eine Aussage über die Wertefunktion im Einzugsbereich von Kontrollmengen betrachten.

3.1 Definition der optimalen Steuerungsprobleme

Wir werden in diesem Kapitel das Kontrollsystem etwas verallgemeinern; wir betrachten als Zustandsraum eine *zusammenhängende C^∞-Mannigfaltigkeit M* der Dimension n, d.h. wir betrachten das folgende Kontrollsystem:

$$\dot{x}(t) = X(x(t), u(t)) \quad \forall t \geq 0 \tag{3.1}$$
$$x(0) = x_0 \in M \tag{3.2}$$
$$u(\cdot) \in \mathcal{U} := \{u : \mathbb{R}_+ \to U \,|\, u \text{ meßbar}\} \tag{3.3}$$
$$U \subseteq \mathbb{R}^d \text{ kompakt} \tag{3.4}$$
$$X(\cdot, u) \text{ sei ein } C^\infty\text{−Vektorfeld auf } M \text{ stetig in } u \in U \tag{3.5}$$

$$g : M \times U \to \mathbb{R} \text{ stetig auf } M \times U \tag{3.6}$$
$$0 \le g(x,u) \le M_g \quad \forall (x,u) \in M \times U \tag{3.7}$$
$$\forall x \in M,\ u \in \mathcal{U} \text{ existiert die Trajektorie } \varphi(t,x,u(\cdot)) \ \forall t \ge 0 \tag{3.8}$$

Das ρ-diskontierte Kostenfunktional und das Durchschnittskostenfunktional sind für $\rho > 0$ definiert durch

$$J_\rho(x,u(\cdot)) := \int_0^\infty e^{-\rho t} g(\varphi(t,x,u(\cdot)),u(t))dt \tag{3.9}$$

$$J_0(x,u(\cdot)) := \limsup_{T\to\infty} \frac{1}{T} \int_0^T g(\varphi(t,x,u(\cdot)),u(t))dt; \tag{3.10}$$

die zugehörigen Wertefunktionen lauten also

$$v_\rho(x) := \inf_{u(\cdot)\in\mathcal{U}} J_\rho(x,u(\cdot)) \tag{3.11}$$
$$v_0(x) := \inf_{u(\cdot)\in\mathcal{U}} J_0(x,u(\cdot)). \tag{3.12}$$

Bemerkung 3.1 Statt Bedingung (3.7) kann auch die Beschränktheit von $|g(x,u)|$ gefordert werden, vgl. Bemerkung 2.6.

Bemerkung 3.2 Aus (3.7), (3.9) und (3.10) folgt sofort $0 \le v_0(x) \le M_g$ und ebenso $0 \le \rho v_\rho(x) \le M_g \ \forall x \in M$, vgl. Bemerkung 2.8.

Wir werden nun untersuchen, ob eine Konvergenzaussage der Form

$$\rho v_\rho(x) \to v_0(x) \quad \text{für } \rho \to 0, \quad x \in M$$

gilt. Wirth [17], Beispiel 1.6 zeigt, daß $\rho v_\rho(x)$ nicht notwendigerweise konvergiert. Im nächsten Abschnitt werden wir erste Bedingungen angeben, unter denen diese Konvergenz gilt.

3.2 Punktweise Konvergenz der Wertefunktion

Die Bedingungen, die in diesem Abschnitt hergeleitet werden, lassen sich vereinfacht wie folgt beschreiben:

Punktweise Konvergenz folgt, wenn approximativ optimale Trajektorien existieren, für die gilt:

(i) Sie werden periodisch nach endlicher Zeit T_ρ.

(ii) Die Länge der Periode und die Zeit T_ρ wachsen nicht zu schnell für $\rho \to 0$.

Diese Bedingungen scheinen sehr technisch und unhandlich in der Anwendung, dienen aber dazu, im nächsten Abschnitt allgemeinere Aussagen herzuleiten. Wir beginnen nun mit einigen vorbereitenden Aussagen, bevor wir das Hauptresultat, d.h. die oben genannten Bedingungen formulieren.

Lemma 3.3 Für alle $u(\cdot) \in \mathcal{U}$ und alle $\tau > 0$ gilt

$$\limsup_{T\to\infty} \frac{1}{T} \int_0^T g(\varphi(t,x,u(\cdot)),u(t))dt = \limsup_{T\to\infty} \frac{1}{T} \int_\tau^T g(\varphi(t,x,u(\cdot)),u(t))dt.$$

Beweis: Folgt sofort aus der Definition des Limes superior. □

Lemma 3.4 Seien ρ, $t > 0$. Dann gilt

$$\frac{\rho}{1-e^{-\rho t}} \geq \frac{1}{t}.$$

Beweis: Folgt durch einfache Umformung aus $e^x \geq 1 + x \ \forall x \in \mathbb{R}$. □

Lemma 3.5 Sei $f : \mathbb{R} \to \mathbb{R}$ stetig und $T \geq 0$, $s > 0$ gegeben mit $f(t + s) = f(t)$ für alle $t > T$. Dann gilt

$$\int_T^\infty \rho e^{-\rho t} f(t)dt = \frac{\rho}{1-e^{-\rho s}} \int_T^{T+s} e^{-\rho t} f(t)dt.$$

Beweis: Durch Aufteilen von $[T, \infty)$ in die periodischen Teilintervalle und Ausrechnen der entstehenden unendlichen Summe. □

Wir werden jetzt das Durchschnittskostenfunktional und das diskontierte Kostenfunktional für periodische Trajektorien berechnen.

Proposition 3.6 Sei $x \in M$, $u(\cdot) \in \mathcal{U}$ und $T \geq 0$, $s > 0$, so daß für alle $t > T$ gilt

$$u(t) = u(t + s) \quad \text{und} \quad \varphi(t,x,u(\cdot)) = \varphi(t + s,x,u(\cdot)).$$

Dann folgt

$$J_0(x,u(\cdot)) = \frac{1}{s} \int_T^{T+s} g(\varphi(t,x,u(\cdot)),u(t))dt = \lim_{\rho\to 0} \rho J_\rho(x,u(\cdot)).$$

Beweis: Es ist

$$\begin{aligned}
J_0(x,u(\cdot)) &= \limsup_{\tau\to\infty}\frac{1}{\tau}\int_0^\tau g(\varphi(t,x,u(\cdot)),u(t))dt\\
&\overset{\text{Lemma 3.3}}{=} \limsup_{\tau\to\infty}\frac{1}{\tau}\int_T^\tau g(\varphi(t,x,u(\cdot)),u(t))dt\\
&= \limsup_{\nu\to\infty,\,\nu\in\mathbb{N}}\frac{1}{T+\nu s}\int_T^{T+\nu s} g(\varphi(t,x,u(\cdot)),u(t))dt\\
&= \limsup_{\nu\to\infty,\,\nu\in\mathbb{N}}\frac{\nu}{T+\nu s}\int_T^{T+s} g(\varphi(t,x,u(\cdot)),u(t))dt\\
&= \frac{1}{s}\int_T^{T+s} g(\varphi(t,x,u(\cdot)),u(t))dt
\end{aligned}$$

und andererseits

$$\begin{aligned}
&\lim_{\rho\to 0}\rho J_\rho(x,u(\cdot))\\
&= \lim_{\rho\to 0}\int_0^\infty \rho e^{-\rho t} g(\varphi(t,x,u(\cdot)),u(t))dt\\
&= \lim_{\rho\to 0}\left(\underbrace{\int_0^T \rho e^{-\rho t} g(\varphi(t,x,u(\cdot)),u(t))dt}_{\to 0 \text{ für } \rho\to 0} + \int_T^\infty \rho e^{-\rho t} g(\varphi(t,x,u(\cdot)),u(t))dt\right)\\
&\overset{(*)}{=} \lim_{\rho\to 0}\left(\frac{\rho}{1-e^{-\rho s}}\int_T^{T+s} e^{-\rho t} g(\varphi(t,x,u(\cdot)),u(t))dt\right)\\
&\overset{(**)}{=} \frac{1}{s}\int_T^{T+s} g(\varphi(t,x,u(\cdot)),u(t))dt,
\end{aligned}$$

wobei $(*)$ wegen Lemma 3.5 angewandt auf den rechten Summanden gilt und $(**)$ aus der Regel von de l'Hospital folgt. □

Im folgenden Satz werden wir die oben bereits genannten Bedingungen für die punktweise Konvergenz präzisieren und das Konvergenzresultat beweisen.

Satz 3.7 Sei $x \in M$ fest. Seien folgende Voraussetzungen erfüllt:

(i) Für alle $\varepsilon > 0$ existiert eine Kontrolle $u_0^\varepsilon(\cdot) \in \mathcal{U}$ sowie $t_0^\varepsilon > 0$, $T_0^\varepsilon \geq 0$ mit

(a) $J_0(x, u_0^\varepsilon(\cdot)) - v_0(x) < \varepsilon$

(b) Für alle $t > T_0^\varepsilon$ gilt:
$u_0^\varepsilon(t) = u_0^\varepsilon(t + t_0^\varepsilon)$ und $\varphi(t, x, u_0^\varepsilon(\cdot)) = \varphi(t + t_0^\varepsilon, x, u_0^\varepsilon(\cdot))$

(ii) Für alle $\rho > 0$ und alle $\varepsilon > 0$ existieren Kontrollen $u_\rho^\varepsilon(\cdot) \in \mathcal{U}$ sowie t_ρ^ε, $T_\rho^\varepsilon > 0$ mit

(a) $\rho J_\rho(x, u_\rho^\varepsilon(\cdot)) - \rho v_\rho(x) < \varepsilon$

(b) Für alle $t > T_\rho^\varepsilon$ gilt:
$u_\rho^\varepsilon(t) = u_\rho^\varepsilon(t + t_\rho^\varepsilon)$ und $\varphi(t, x, u_\rho^\varepsilon(\cdot)) = \varphi(t + t_\rho^\varepsilon, x, u_\rho^\varepsilon(\cdot))$

(c) $\lim\limits_{\rho \to 0} \rho t_\rho^\varepsilon = 0$, für alle $\varepsilon > 0$

(d) $\lim\limits_{\rho \to 0} \rho T_\rho^\varepsilon = 0$, für alle $\varepsilon > 0$

Dann folgt $\lim\limits_{\rho \to 0} \rho v_\rho(x) = v_0(x)$.

Beweis: Sei $\varepsilon > 0$ beliebig. Wähle $u_0(\cdot)$ so, daß Bedingung (i) erfüllt ist. Dann gilt mit Proposition 3.6:

$$v_0(x) \geq J_0(x, u_0^\varepsilon(\cdot)) - \varepsilon = \lim_{\rho \to 0} \rho J_\rho(x, u_0^\varepsilon(\cdot)) - \varepsilon \geq \limsup_{\rho \to 0} \rho v_\rho(x) - \varepsilon.$$

Es bleibt also noch zu zeigen $\liminf\limits_{\rho \to 0} \rho v_\rho(x) \geq v_0(x)$.
Sei dazu wieder $\varepsilon > 0$ beliebig. Wähle nach (ii) Kontrollen u_ρ^ε für alle $\rho > 0$. Dann gilt

$$\begin{aligned}
\liminf_{\rho \to 0} \rho v_\rho(x) &\geq \liminf_{\rho \to 0} \rho J_\rho(x, u_\rho^\varepsilon(\cdot)) - \varepsilon \\
&= \liminf_{\rho \to 0} \int_0^\infty \rho e^{-\rho t} g(\varphi(t, x, u_\rho^\varepsilon(\cdot)), u_\rho^\varepsilon(t)) dt - \varepsilon \\
&\overset{(ii)(d)}{=} \liminf_{\rho \to 0} \int_{T_\rho^\varepsilon}^\infty \rho e^{-\rho t} g(\varphi(t, x, u_\rho^\varepsilon(\cdot)), u_\rho^\varepsilon(t)) dt - \varepsilon \\
&\overset{\text{Lemma 3.5}}{=} \liminf_{\rho \to 0} \frac{\rho}{1 - e^{-\rho t_\rho^\varepsilon}} \int_{T_\rho^\varepsilon}^{T_\rho^\varepsilon + t_\rho^\varepsilon} e^{-\rho t} g(\varphi(t, x, u_\rho^\varepsilon(\cdot)), u_\rho^\varepsilon(t)) dt - \varepsilon \\
&\overset{\text{Lemma 3.4}}{\geq} \liminf_{\rho \to 0} e^{-\rho(T_\rho^\varepsilon + t_\rho^\varepsilon)} \frac{1}{t_\rho^\varepsilon} \int_{T_\rho^\varepsilon}^{T_\rho^\varepsilon + t_\rho^\varepsilon} g(\varphi(t, x, u_\rho^\varepsilon(\cdot)), u_\rho^\varepsilon(t)) dt - \varepsilon \\
&\overset{\text{Prop. 3.6}}{=} \liminf_{\rho \to 0} J_0(x, u_\rho^\varepsilon(\cdot)) - \varepsilon \;\geq\; v_0(x) - \varepsilon.
\end{aligned}$$

Bei der letzten Gleichheit geht ein, daß $\rho(T_\rho^\varepsilon + t_\rho^\varepsilon)$ gegen Null konvergiert für $\rho \to 0$ wegen den Voraussetzungen (ii)(c) und (d). □

3.3 Konvergenz in Kontrollmengen

In diesem Abschnitt werden wir nun allgemeinere Bedingungen herleiten, unter denen die Bedingungen von Satz 3.7 erfüllt sind. Hierzu werden wir den Begriff der *Kontrollmengen* einführen und einige Eigenschaften dieser Kontrollmengen betrachten. Kontrollmengen werden auch in späteren Kapiteln eine wichtige Rolle spielen.

Anschließend werden wir ein Resultat über gleichmäßige Konvergenz der Wertefunktion des diskontierten optimalen Steuerungsproblems auf gewissen Teilmengen von M herleiten.

Definition 3.8 Der *positive Orbit von $x \in M$ bis zur Zeit T* ist definiert durch

$$O^+_{\leq T}(x) := \{y \in M \,|\, \text{es gibt } 0 \leq t \leq T \text{ und } u(\cdot) \in \mathcal{U}, \text{ so daß } \varphi(t, x, u(\cdot)) = y\}.$$

Der *positive Orbit* von $x \in M$ ist definiert durch

$$O^+(x) := \bigcup_{T \geq 0} O^+_{\leq T}(x).$$

Definition 3.9 Der *negative Orbit von $x \in M$ bis zur Zeit T* ist definiert durch

$$O^-_{\leq T}(x) := \{y \in M \,|\, \text{es gibt } 0 \leq t \leq T \text{ und } u(\cdot) \in \mathcal{U}, \text{ so daß } \varphi(t, y, u(\cdot)) = x\}.$$

Der *negative Orbit* von $x \in M$ ist definiert durch

$$O^-(x) := \bigcup_{T \geq 0} O^-_{\leq T}(x).$$

Definition 3.10 Eine Teilmenge $D \subseteq M$ heißt *Kontrollmenge*, falls gilt:

(i) $D \subseteq \overline{O^+(x)}$ für alle $x \in D$

(ii) D ist maximal mit Eigenschaft (i)

(iii) Wenn $D = \{x\}$, so gibt es $u \in \mathcal{U}$ mit $\varphi(t, x, u(\cdot)) = x \ \forall t \geq 0$.

Definition 3.11 Eine Kontrollmenge C heißt *invariant* falls gilt

$$\overline{C} = \overline{O^+(x)} \ \forall x \in C.$$

Eine Kontrollmenge, die nicht invariant ist, heißt *variant*.

Bemerkung 3.12 Periodische Trajektorien liegen immer in Kontrollmengen.

Im weiteren sei stets die folgende **Annahme** erfüllt:
Sei L die Lie-Algebra, die von den Vektofeldern $X(\cdot, u)$, $u \in U$ erzeugt wird. Sei Δ_L die Distribution, die durch L in TM, dem Tangentialbündel an M, definiert wird. Wir nehmen an, daß gilt

$$\dim \Delta_L(x) = n \ (= \dim M) \ \text{für alle} \ x \in M. \tag{3.13}$$

Nach dem Satz von Krener ([13], Theorem 1) garantiert uns diese Annahme, daß der der Schnitt beliebiger offener Umgebungen $U(x)$ mit dem positiven sowie dem negative Orbit von $x \in M$ zu beliebiger Zeit $T > 0$ nichtleeres Inneres hat. Diese Eigenschaft werden wir für das folgende Lemma ausnutzen:

Lemma 3.13 Gegeben sei ein Kontrollsystem auf M, mit den Eigenschaften (3.1) – (3.8) und (3.13). Dann gilt:
Wenn D Kontrollmenge ist, dann ist $\text{int}D \subset O^+(x)$ für alle $x \in D$.

Beweis: Wir geben uns beliebige $x \in D$, $y \in \text{int}D$ vor.
Da $\text{int}D$ eine Umgebung von y ist, gibt es wegen (3.13) eine offene Menge A in D, die in $O^-(y)$ liegt, nämlich gerade das Innere des Schnitts von $\text{int}D$ und $O^-(y)$. Da nach der Definition der Kontrollmenge $D \subseteq \overline{O^+(x)}$ gilt, gilt ebenfalls $A \subseteq \overline{O^+(x)}$, wegen der Offenheit von A gibt es also einen Punkt $z \in A \cap O^+(x)$, der damit in $O^-(y) \cap O^+(x)$ liegt. Also ist $y \in O^+(x)$, was zu zeigen war. □

Die Definition von Kontrollmengen setzt nur approximative Kontrollierbarkeit voraus, d.h. die Existenz von Kontrollen, die in beliebige Umgebungen eines gegebenen Punktes steuern. Lemma 3.13 zeigt, daß mit Annahme (3.13) im Inneren von Kontrollmengen exakte Kontrollierbarkeit gilt. Diese werden wir ausnutzen, um periodische Lösungen zu konstruieren.

Definition 3.14 Wir definieren eine „minimale Trefferzeit-Funktion“ durch:

$$\begin{aligned} k : M \times M &\to \mathbb{R} \cup \{\infty\} \\ (x, y) &\mapsto \inf\{t \geq 0 \,|\, \text{es gibt } u(\cdot) \in \mathcal{U} \text{ so daß } \varphi(t, x, u(\cdot)) = y\} \end{aligned}$$

Proposition 3.15 Gegeben sei ein Kontrollsystem auf M, das die Bedingungen (3.1) – (3.8) und (3.13) erfüllt.
Desweiteren existiere eine Kontrollmenge $D \subset M$ mit $\text{int}D \neq \emptyset$ und zwei kompakte Mengen $K_1 \subset D$, $K_2 \subset \text{int}D$.
Dann gibt es eine Konstante $r \in \mathbb{R}$ abhängig von K_1 und K_2, so daß gilt:

(i) $k(x, y) \leq r$ für alle $x \in K_1$, $y \in K_2$

(ii) $K_1 \subset \text{int}O^-_{\leq r}(y)$ für alle $y \in K_2$

(iii) $K_2 \subset \text{int}O^+_{\leq r}(x)$ für alle $x \in K_1$.

Beweis: Siehe Colonius, Kliemann [5], Proposition 2.3.

Mit diesen Vorbereitungen können wir uns nun an die Konstruktion approximativ optimaler periodischer Lösungen machen. Wir beginnen mit dem Durchschnittskostenfunktional.

Proposition 3.16 Gegeben sei ein Kontrollsystem auf M, das die Bedingungen (3.1) – (3.8) und (3.13) erfüllt.
Weiterhin seien eine Kontrollmenge $D \subset M$, ein $x \in M$, eine kompakte Teilmenge $K \subset \text{int}D$, eine Kontrolle $u(\cdot) \in \mathcal{U}$ sowie ein $T > 0$ gegeben, so daß $\varphi(t, x, u(\cdot)) \in K$ für alle $t \geq T$.
Dann existiert für jedes $\varepsilon > 0$ eine Kontrolle $u_\varepsilon(\cdot)$, so daß gilt

(i) $u_\varepsilon(T + \cdot)$ und $\varphi(T + \cdot, x, u_\varepsilon(\cdot))$ sind periodisch mit der gleichen Periode

(ii) $J_0(x, u_\varepsilon(\cdot)) - J_0(x, u(\cdot)) \leq \varepsilon$.

Beweis: Wegen Lemma 3.3 können wir o.B.d.A. $T = 0$ annehmen. Nach Proposition 3.15 gibt es $r \geq 0$, so daß $k(x, y) \leq r$ für alle $x, y \in K$.
Nach Definition des Durchschnittskostenfunktionals gibt es nun ein t_ε, so daß

$$\frac{1}{t_\varepsilon} \int_0^{t_\varepsilon} g(\varphi(t, x, u(\cdot)), u(t))dt \;<\; J_0(x, u(\cdot)) + \frac{\varepsilon}{2}. \tag{3.14}$$

Wenn t_ε genügend groß gewählt wird, gilt für alle $v(\cdot) \in \mathcal{U}$:

$$\frac{1}{t_\varepsilon + r} \int_0^{t_\varepsilon + r} g(\varphi(t, x, v(\cdot)), v(t))dt \;\leq\; \frac{1}{t_\varepsilon} \int_0^{t_\varepsilon} g(\varphi(t, x, v(\cdot)), v(t))dt + \frac{\varepsilon}{2}. \tag{3.15}$$

Da $\varphi(t_\varepsilon, x, u(\cdot)) \in K$, gibt es eine Kontrolle $w(\cdot)$, so daß $\varphi(t_1, \varphi(t_\varepsilon, x, u(\cdot)), w(\cdot)) = x$ für $t_1 \leq r$. Definiere nun $u_\varepsilon(\cdot)$ durch

$$u_\varepsilon(t) := \begin{cases} u(t), & 0 \leq t \leq t_\varepsilon \\ w(t - t_\varepsilon), & t_\varepsilon \leq t \leq t_\varepsilon + t_1. \end{cases}$$

Wegen $\varphi(t_\varepsilon + t_1, x, u_\varepsilon(\cdot)) = x$ können wir $u_\varepsilon(\cdot)$ $t_\varepsilon + t_1$-periodisch fortsetzen. Nach Proposition 3.6 und den Ungleichungen (3.14) und (3.15) ergibt dies

$$\begin{aligned} J_0(x, u_\varepsilon(\cdot)) &= \frac{1}{t_\varepsilon + t_1} \int_0^{t_\varepsilon + t_1} g(\varphi(t, x, u_\varepsilon(\cdot)), u_\varepsilon(t))dt \\ &\leq \frac{1}{t_\varepsilon} \int_0^{t_\varepsilon} g(\varphi(t, x, u(\cdot)), u(t))dt + \frac{\varepsilon}{2} \;<\; J_0(x, u(\cdot)) + \varepsilon. \end{aligned}$$

□

Eine ähnliche Aussage machen wir jetzt für das diskontierte Zielfunktional.

Proposition 3.17 Gegeben sei ein Kontrollsystem auf M, das die Bedingungen (3.1) – (3.8) und (3.13) erfüllt.
Sei $x \in M$ fest und eine Kontrollmenge $D \subset M$, eine kompakte Teilmenge $K \subset \text{int}D$, eine Kontrolle $u(\cdot) \in \mathcal{U}$ sowie ein $T > 0$ gegeben, so daß $\varphi(t, x, u(\cdot)) \in K$ für alle $t \geq T$.
Dann existiert eine Konstante $r = r(K)$ und für alle $\varepsilon > 0$ und alle $\rho > 0$ existieren positive Konstanten s_1, s_2 sowie eine Kontrolle $w(\cdot)$, alle abhängig von ε und ρ, so daß gilt:

(i) $\rho J_\rho(x, w(\cdot)) - \rho J_\rho(x, u(\cdot)) \leq \varepsilon$

(ii) Für alle $t \geq T + s_1$ gilt
$w(t) = w(t + s_2)$ und $\varphi(t, x, w(\cdot)) = \varphi(t + s_2, x, w(\cdot))$

(iii) $s_1,\ s_2 \ \leq \frac{\ln\left(3\frac{M_g(1-e^{-\rho r})}{\varepsilon}+1\right)}{\rho} + r$

(iv) $\lim\limits_{\rho\to 0} \rho s_1(\varepsilon, \rho) = \lim\limits_{\rho\to 0} \rho s_2(\varepsilon, \rho) = 0$ gilt für alle $\varepsilon > 0$.

Beweis: Zunächst einmal ist (iv) eine sofortige Konsequenz aus (iii), denn da r unabhängig von ε und ρ ist, folgt

$$\lim_{\rho\to 0} \rho \left(\frac{\ln\left(3\frac{M_g(1-e^{-\rho r})}{\varepsilon} + 1\right)}{\rho} + r \right) = \lim_{\rho\to 0} \ln\left(3\frac{M_g(1-e^{-\rho r})}{\varepsilon} + 1\right) = 0.$$

Wähle nun $\varepsilon > 0$ und $\rho > 0$ und definiere $r := \sup\{k(x, y) \,|\, x, y \in K\} < \infty$.
Zur Abkürzung setzen wir

$$a := \frac{\ln\left(3\frac{M_g(1-e^{-\rho r})}{\varepsilon} + 1\right)}{\rho}. \tag{3.16}$$

Für jedes $y \in K$ definiere $\mathcal{U}(y) := \{u(\cdot) \in \mathcal{U} \,|\, \varphi(a, y, u(\cdot)) \in K\}$.
Nach Voraussetzung wissen wir, daß $\mathcal{U}(y)$ nicht für alle $y \in K$ leer ist. Wenn $a \geq r$ ist, also $e^{-\rho r} \geq \varepsilon/3M_g$, was für hinreichend kleine ρ und ε immer gilt, folgt $\mathcal{U}(y) \neq \emptyset$ für alle $y \in K$.
Wähle nun $\bar{x} \in K$ und $\bar{u}(\cdot) \in \mathcal{U}(\bar{x})$, so daß

$$\int_0^a \rho e^{-\rho t} g(\varphi(t, \bar{x}, \bar{u}(\cdot)), \bar{u}(t))dt - \inf_{x\in K} \inf_{u(\cdot)\in\mathcal{U}(x)} \int_0^a \rho e^{-\rho t} g(\varphi(t, x, u(\cdot)), u(t))dt \leq \frac{\varepsilon}{3}(1 - e^{-\rho a}). \tag{3.17}$$

Nach Voraussetzung gibt es eine Kontrolle $w_1(\cdot)$ und $r_1 \leq r$ mit

$$\varphi(r_1, \varphi(T + a, x, u(\cdot)), w_1(\cdot)) = \bar{x}$$

sowie eine Kontrolle $w_2(\cdot)$ und $r_2 < r$ mit

$$\varphi(r_2, \varphi(a, \bar{x}, \bar{u}(\cdot)), w_2(\cdot)) = \bar{x}.$$

Wir konstruieren nun die Kontrolle $w(\cdot)$, die die Behauptung erfüllt, indem wir zuerst mit $u(\cdot)$ aus der Voraussetzung nach K steuern. Wir werden noch für die Zeit a diese Kontrolle beibehalten und dann nach $\bar{x}$ steuern, welches der Startpunkt der periodischen Trajektorie ist. Von dort aus steuern wir für die Zeit a mit $\bar{u}(\cdot)$ und kehren mit $w_2(\cdot)$ zu $\bar{x}$ zurück.
Insgesamt gibt dies folgende (rekursive) Definition:

$$w(t) := \begin{cases} u(t), & 0 & \leq t \leq & T + a \\ w_1(t - (T + a)), & T + a & < t \leq & T + a + r_1 \\ \bar{u}(t - (T + a + r_1)), & T + a + r_1 & < t \leq & T + 2a + r_1 \\ w_2(t - (T + 2a + r_1)), & T + 2a + r_1 & < t \leq & T + 2a + r_1 + r_2 \\ w(t - (a + r_2)), & T + 2a + r_1 + r_2 & < t & \end{cases}$$

Für diese Kontrolle gelten mit $s_1 = a + r_1$ und $s_2 = a + r_2$ sicher die Abschätzungen (ii) – (iv). Es bleibt also noch zu zeigen, daß gilt

$$\rho J_\rho(x, w(\cdot)) - \rho J_\rho(x, u(\cdot)) \leq \varepsilon.$$

Aus der Periodizität von $w(\cdot)$ folgt die Abschätzung

$$\begin{aligned}
\rho J_\rho(x, w(\cdot)) - \rho J_\rho(x, u(\cdot)) &= \int_{T+a}^{\infty} \rho e^{-\rho t}\Big(g(\varphi(t, x, w(\cdot)), w(t)) - g(\varphi(t, x, u(\cdot)), u(t))\Big)dt \\
&\leq \int_{T+a}^{T+s_1} \rho e^{-\rho t} M_g dt \\
&\quad + \sum_{\nu=0}^{\infty} \int_{T+s_1+\nu s_2}^{T+s_1+\nu s_2+a} \rho e^{-\rho t}\Big((g(\varphi(t, x, w(\cdot)), w(t)) - g(\varphi(t, x, u(\cdot)), u(t))\Big)dt \\
&\quad + \sum_{\nu=0}^{\infty} \int_{T+s_1+\nu s_2+a}^{T+s_1+(\nu+1)s_2} \rho e^{-\rho t} M_g dt.
\end{aligned}$$

Wegen Ungleichung (3.17) und $\varphi(T+s_1+\nu s_2, x, w(\cdot)) = \bar{x}$ sowie $\varphi(T+s_1+\nu s_2, x, u(\cdot)) \in K$ können wir fortfahren, indem wir die Integrale ausrechnen

$$\begin{aligned}
&\leq M_g e^{-\rho(T+a)}(1 - e^{-\rho r_1}) + e^{-\rho(T+s_1)} \sum_{\nu=0}^{\infty} e^{-\rho\nu s_2} \frac{\varepsilon}{3}(1 - e^{-\rho a}) \\
&\quad + M_g e^{-\rho(T+s_1+a)} \sum_{\nu=0}^{\infty} e^{-\rho\nu s_2}(1 - e^{-\rho r_2}) \\
&\leq M_g e^{-\rho a}(1 - e^{-\rho r_1}) + \frac{\varepsilon(1 - e^{-\rho a})}{3(1 - e^{-\rho s_2})} + M_g e^{-\rho a} \frac{1 - e^{-\rho r_2}}{1 - e^{-\rho s_2}}
\end{aligned}$$

Aus (3.16) folgt nun

$$e^{\rho a} = 3\frac{M_g(1 - e^{-\rho r})}{\varepsilon} + 1 \tag{3.18}$$

und daraus

$$e^{-\rho a} < \frac{\varepsilon}{3M_g(1 - e^{-\rho r})}. \tag{3.19}$$

Mit (3.19) ergibt sich für den ersten Term

$$M_g e^{-\rho a}(1 - e^{-\rho r_1}) \leq M_g \frac{\varepsilon(1 - e^{-\rho r_1})}{3M_g(1 - e^{-\rho r})} \leq \frac{\varepsilon}{3}$$

und für den dritten Term gilt mit (3.18)

$$M_g e^{-\rho a} \frac{1 - e^{-\rho r_2}}{1 - e^{-\rho s_2}} \leq M_g e^{-\rho a} \frac{1 - e^{-\rho r}}{1 - e^{-\rho(a+r_2)}} = M_g \frac{1 - e^{-\rho r}}{e^{\rho a} - e^{-\rho r_2}}$$

$$\begin{aligned} &= M_g \frac{1-e^{-\rho r}}{3\frac{M_g(1-e^{-\rho r})}{\varepsilon} + \underbrace{1-e^{-\rho r_2}}_{\geq 0}} \\ &\leq \frac{\varepsilon}{3}\ \frac{1-e^{\rho r}}{1-e^{\rho r}} \ = \ \frac{\varepsilon}{3}. \end{aligned}$$

Also folgt

$$M_g e^{-\rho a}(1-e^{-\rho r_1}) + \frac{\varepsilon(1-e^{-\rho a})}{3(1-e^{-\rho s_2})} + M_g e^{-\rho a}\frac{1-e^{-\rho r_2}}{1-e^{-\rho s_2}} \ \leq \ \frac{\varepsilon}{3}+\frac{\varepsilon}{3}+\frac{\varepsilon}{3} \ = \ \varepsilon,$$

und damit die Behauptung. □

Wir können nun aus Satz 3.7 ein Konvergenzresultat ableiten, daß uns unter relativ einfachen Voraussetzungen gleichmäßige Konvergenz der diskontierten optimalen Wertefunktion auf kompakten Teilmengen von Kontrollmengen liefert.
Eine unmittelbare Folge von Lemma 3.3 und Lemma 3.13 ist, daß v_0 auf dem Inneren von Kontrollmengen konstant ist. Diese Eigenschaft nutzen wir aus, um die gleichmäßige Konvergenz zu zeigen.

Satz 3.18 Gegeben sei ein Kontrollsystem auf M, das die Bedingungen (3.1) – (3.8) und (3.13) erfüllt.
Außerdem seien eine Kontrollmenge $D \subset M$, ein $x \in \text{int}D$, eine kompakte Teilmenge $K \subset \text{int}D$ sowie optimale Kontrollen $u_\rho(\cdot)$, $u_0(\cdot) \in \mathcal{U}$ gegeben, so daß

$$\begin{aligned} &\varphi(t,x,u_\rho(\cdot)) \in K, \qquad \forall t \geq 0,\ \forall \rho > 0 \\ &\varphi(t,x,u_0(\cdot)) \in K, \qquad \forall t \geq 0 \end{aligned}$$

Dann gilt $\rho v_\rho \to v_0$ für $\rho \to 0$ gleichmäßig auf kompakten Teilmengen von int$D$.

Beweis: Aus den Propositionen 3.16 und 3.17 sowie Satz 3.7 folgt sofort $\rho v_\rho(x) \to v_0(x)$ für $\rho \to 0$.
Wähle nun eine kompakte Menge $Q \subset \text{int}D$. Nach Proposition 3.15 gibt es dann eine Konstante $\infty > r = \sup\{k(y,z) \,|\, y,z \in Q \cup \{x\}\} \geq 0$.
Für alle $y \in Q$ gibt es also eine Kontrolle $u(\cdot)$ und ein $T \leq r$, so daß $\varphi(T,x,u(\cdot)) = y$. Nach Bellman's Optimalitätsprinzip (2.10) wissen wir, daß für alle $\rho > 0$ gilt:

$$\rho v_\rho(x) \ \leq \ \int_0^T \rho e^{-\rho t} g(\varphi(t,x,u(\cdot)),u(t))dt + e^{-\rho T}\rho v_\rho(y).$$

Also folgt

$$\begin{aligned} \rho v_\rho(x) - \rho v_\rho(y) \ &\leq \ (e^{-\rho T}-1)\rho v_\rho(y) + \int_0^T \rho e^{-\rho t} g(\varphi(t,x,u(\cdot)),u(t))dt \\ &\leq \ \int_0^r \rho e^{-\rho t} M_g dt \ = \ M_g(1-e^{-\rho r}) \end{aligned}$$

Die letzte Gleichheit sieht man durch Ausrechnen des Integrals. Aus Symmetriegründen gilt ebenfalls

$$\rho v_\rho(y) - \rho v_\rho(x) \;\leq\; M_g(1 - e^{-\rho r}).$$

Da $\lim_{\rho\to 0} M_g(1 - e^{-\rho r}) = 0$ folgt $\lim_{\rho\to 0} \rho v_\rho(x) = \lim_{\rho\to 0} \rho v_\rho(y)$ für alle $y \in Q$.
Die gleichmäßigen Konvergenz folgt, da für alle y, $z \in Q$ gilt:

$$\Big|\rho v_\rho(z) - \rho v_\rho(y)\Big| \leq M_g(1 - e^{-\rho r}).$$

□

Bemerkung 3.19 Falls auf M eine Metrik definiert ist und D eine Kontrollmenge ist mit $\overline{D}$ kompakt, so ist die Voraussetzung an D für Satz 3.18 äquivalent zu:
Es gibt ein $x \in \mathrm{int}D$ und ein $\delta > 0$, so daß für die optimalen Kontrollen $u_\rho(\cdot)$ und $u_0(\cdot)$ gilt

$$\varphi(t, x, u_\rho(\cdot)) \in \mathrm{int}D, \quad \varphi(t, x, u_0(\cdot)) \in \mathrm{int}D$$

und

$$d(\omega(x, u_\rho), \partial D) \geq \delta, \quad \omega(x, u_0) \in \mathrm{int}D,$$

mit $\omega(x, u(\cdot)) := \{y \in M \,|\, \exists t_k \to \infty \text{ mit } \varphi(t_k, x, u(\cdot)) \to y\}$. Hierbei bezeichnet d das Infimum der Abstände über alle Punkte der angegeben Mengen.

3.4 Die Wertefunktion auf Einzugsbereichen von Kontrollmengen

In diesem letzten Abschnitt wollen wir uns noch kurz mit der Konvergenz der diskontierten Wertefunktion auf Einzugsbereichen von Kontrollmengen beschäftigen und ein ähnliches Resultat wie oben herleiten.

Lemma 3.20 Sei $x \in M$ und $y \in O^-(x)$. Dann gilt

$$v_0(y) \leq v_0(x).$$

Beweis: Da es eine Kontrolle gibt, mit der in endlicher Zeit von y nach x gesteuert werden kann, folgt die Behauptung sofort mit Lemma 3.3. □

Definition 3.21 Sei $B \subset M$ eine Menge. Der *Einzugsbereich von B bis zur Zeit T* ist definiert durch

$$A_{\leq T}(B) := \bigcup_{x\in B} O^-_{\leq T}(x).$$

Der *Einzugsbereich* von B ist definiert durch

$$A(B) := \bigcup_{T\geq 0} A_{\leq T}(B).$$

Bemerkung 3.22 Sei $D \in M$ eine Kontrollmenge. Dann gilt für alle $Q \subset \text{int}D$ und alle $T > 0 \quad A_{\leq T}(Q) \subseteq A_{\leq T}(D)$ und $A(Q) = A(D)$, letzteres als Folgerung aus Lemma 3.13.

Korollar 3.23 Gegeben sei ein Kontrollsystem auf M, das die Bedingungen (3.1) – (3.8) und (3.13) erfüllt.
Außerdem sei eine Kontrollmenge $D \subset M$ gegeben. Dann gilt

$$v_0(x) \leq \inf_{y \in \text{int}D} v_0(y) \quad \forall x \in A(D).$$

Beweis: Sei $x \in A(D)$ und $\varepsilon > 0$ gegeben. Wähle $\tilde{y} \in \text{int}D$, so daß $v_0(\tilde{y}) \leq \inf_{y \in \text{int}D} v_0(y) + \varepsilon$. Da nach Lemma 3.13 $\tilde{y}$ von ganz D aus erreichbar ist, ist $x \in O^-(\tilde{y})$. Mit Lemma 3.20 gilt nun $v_0(x) \leq v_0(\tilde{y}) \leq \inf_{y \in \text{int}D} v_0(y) + \varepsilon$ und da $\varepsilon > 0$ beliebig war folgt so die Behauptung. □

Für gewisse Teilmengen des Einzugsbereiches der Kontrollmenge D können wir nun eine Art gleichmäßige Approximation durch die diskontierte optimale Wertefunktion zeigen.

Satz 3.24 Gegeben sei ein Kontrollsystem auf M, das die Bedingungen (3.1) – (3.8) und (3.13) erfüllt.
Desweiteren sei $D \subset M$ eine Kontrollmenge, die die Voraussetzungen von Satz 3.18 erfüllt. Dann gibt es für beliebige kompakte Mengen $Q \subset D, T > 0$ und $\varepsilon > 0$ ein $R = R(Q, T) > 0$, so daß für alle $\rho < R$ gilt:

$$\rho v_\rho(x) \leq v_0(z) + \varepsilon \quad \forall x \in A_{\leq T}(Q), \;\; z \in \text{int}D.$$

Beweis: Wegen Satz 3.18 gibt es R_1, so daß gilt

$$\rho v_\rho(y) \leq v_0(y) + \frac{\varepsilon}{2} = v_0(z) + \frac{\varepsilon}{2} \quad \forall \rho < R_1, \;\; y \in Q, \;\; z \in \text{int}D. \tag{3.20}$$

Zu beliebigem $x \in A_{\leq T}(Q)$ gibt es nun ein $y \in Q$ und eine Kontrolle $u(\cdot)$ mit

$$\varphi(t, x, u(\cdot)) = y \;\text{ für }\; t \leq T.$$

Aus Bellman's Optimalitätsprinzip (2.10) folgt nun für genügend kleines $R_2 > 0$ analog zum Beweis von Satz 3.18:

$$\rho v_\rho(x) - \rho v_\rho(y) \leq M_g(1 - e^{-\rho T}) \leq \frac{\varepsilon}{2} \quad \forall \rho < R_2.$$

Zusammen mit Ungleichung (3.20) ergibt dies die Behauptung mit $R = \min\{R_1, R_2\}$. □

Korollar 3.25 Unter den Voraussetzungen von Satz 3.24 gibt es für alle $x \in A(D)$ und alle $\varepsilon > 0$ ein $R > 0$, so daß für alle $\rho < R$ gilt

$$\rho v_\rho(x) \leq v_0(z) + \varepsilon \quad \forall z \in \text{int}D.$$

Beweis: Mit Bemerkung 3.22 können wir ein kompaktes $Q \subset D$ sowie $T > 0$ finden, so daß $x \in A_{\leq T}(Q)$ gilt. Nun folgt die Behauptung sofort mit Satz 3.24. □

Kapitel 4

Ein Approximationssatz für das diskontierte optimale Steuerungsproblem

Im letzten Kapitel haben wir eine Aussage über die Konvergenz der optimalen Wertefunktion des diskontierten optimalen Steuerungsproblems gegen die Wertefunktion des Durchschnittskostenfunktionals hergeleitet. In diesem Kapitel wollen wir untersuchen, wie die Werte der beiden Funktionale zusammenhängen, wenn die gleiche Kontrolle eingesetzt wird.

Da wir uns in den nächsten Kapiteln speziell mit Teilmengen des Zustandsraumes beschäftigen wollen, in denen die optimale Wertefunktion negative Werte annimmt (also insbesondere nach oben beschränkt ist), wird hier nun ein Resultat hergeleitet, das Trajektorien betrachtet, entlang denen das Zielfunktional beschränkt ist.

4.1 Herleitung des Approximationssatzes

Zur Herleitung des Satzes benötigen wir zwei vorbereitende Lemmas, in denen Eigenschaften diskontierter Integrale gezeigt werden.

Lemma 4.1 Sei $g : \mathbb{R} \to \mathbb{R}$ eine integrierbare Funktion mit
$|g(t)| \leq M_g \ \ \forall t \in \mathbb{R}$ und $\int_0^\infty e^{-\rho t} g(t) dt < -\delta, \quad \delta > 0.$
Dann gibt es ein $\tau \in [a, b], \ \ a := -\frac{\ln(-\frac{\delta\rho}{2M_g}+1)}{\rho}, \ \ b := -\frac{\ln(\frac{\delta\rho}{2M_g})}{\rho}$
mit $\int_0^\tau e^{-\rho t} g(t) dt \leq -\frac{\delta}{2}$.

Beweis: Es ist

$$\left|\int_c^d e^{-\rho t} g(t) dt\right| \leq \int_c^d |e^{-\rho t} g(t)| dt \leq \int_c^d e^{-\rho t} M_g dt = \frac{M_g}{\rho}(e^{-\rho c} - e^{-\rho d}).$$

Also gilt:

$$\delta \leq \lim_{d\to\infty} |\int_0^d e^{-\rho t} g(t)dt| \leq \lim_{d\to\infty} \frac{M_g}{\rho}(e^{-\rho 0} - e^{-\rho d}) = \frac{M_g}{\rho}$$

und daher $\frac{\delta\rho}{2M_g} < \frac{\delta\rho}{M_g} \leq 1$, womit $\ln(-\frac{\delta\rho}{2M_g} + 1)$ definiert ist und $a,\ b > 0$ gilt. Außerdem gilt für beliebiges $\tilde{a} \in [0, a)$

$$\begin{aligned} |\int_0^{\tilde{a}} e^{-\rho t} g(t)dt| &\leq \frac{M_g}{\rho}(e^{-\rho 0} - e^{-\rho\tilde{a}}) = -\frac{M_g}{\rho}(e^{-\rho\tilde{a}} - 1) \\ &< -\frac{M_g}{\rho}(e^{-\rho a} - 1) = -\frac{M_g}{\rho}(e^{-\rho(-\frac{\ln(-\frac{\delta\rho}{2M_g}+1)}{\rho})} - 1) \\ &= -\frac{M_g}{\rho}(-\frac{\delta\rho}{2M_g} + 1 \ - 1) = \frac{\delta}{2} \end{aligned}$$

weshalb $\tau \geq a$ sein muß, und für b gilt:

$$\begin{aligned} |\int_b^{\infty} e^{-\rho t} g(t)dt| &= \lim_{d\to\infty} |\int_b^d e^{-\rho t} g(t)dt| \leq \lim_{d\to\infty} \frac{M_g}{\rho}(e^{-\rho b} - e^{-\rho d}) \\ &= \frac{M_g}{\rho} e^{-\rho b} = \frac{M_g}{\rho} e^{-\rho(-\frac{\ln(\frac{\delta\rho}{2M_g})}{\rho})} = \frac{\delta}{2}, \end{aligned}$$

weswegen

$$\int_0^b e^{-\rho t} g(t)dt \leq -\frac{\delta}{2}$$

ist und daher $\tau \leq b$ gewählt werden kann. □

Lemma 4.2 Sei $g : \mathbb{R} \to \mathbb{R}$ eine integrierbare Funktion mit
$|g(t)| \leq M_g \ \ \forall t \in \mathbb{R}$, $\rho > 0$ und $\int_0^{\tau} e^{-\rho t} g(t)dt \leq -\delta, \quad \delta > 0$, für ein $\tau \in \mathbb{R}$.
Dann gibt es ein $\tilde{\tau} \in \mathbb{R},\ \ c \leq \tilde{\tau} \leq \tau$, $c := \frac{\delta}{M_g}$ mit $\int_0^{\tilde{\tau}} g(t)dt \leq -\delta$.

Beweis: Für beliebiges $\tilde{c} \in [0, c)$ gilt:

$$|\int_0^{\tilde{c}} g(t)dt| \leq \int_0^{\tilde{c}} |g(t)|dt \leq \int_0^{\tilde{c}} M_g dt = \tilde{c}M_g < cM_g = \frac{\delta}{M_g} M_g = \delta,$$

woraus folgt, daß $\tilde{\tau} \geq c$ sein muß, falls es existiert.

Sei nun $f : \mathbb{R} \to \mathbb{R},\ \ f(t) := e^{-\rho t} g(t)$. Auf $[0, \tau]$ sind dann sowohl f als auch $e^{\rho t} f$ ebenfalls

durch M_g beschränkt. Zu zeigen ist nun: $\int_0^{\tilde{\tau}} e^{\rho t} f(t)dt \leq -\delta$ für ein $\tilde{\tau} \leq \tau$. Es seien $f^+,\ f^- : \mathbb{R} \to \mathbb{R}$ wie folgt definiert:

$$f^+(t) = \begin{cases} f(t) & \text{falls } f(t) \geq 0 \\ 0 & \text{sonst} \end{cases}, \quad f^-(t) = \begin{cases} -f(t) & \text{falls } f(t) \leq 0 \\ 0 & \text{sonst} \end{cases}$$

Dann gilt

$$\int_0^{\sigma} f(t)dt = \int_0^{\sigma} f^+(t) - f^-(t)dt = \int_0^{\sigma} f^+(t)dt - \int_0^{\sigma} f^-(t)dt \quad \forall \sigma \in \mathbb{R}.$$

Wähle nun $\tilde{\tau} := \min\{\sigma \in [0,\tau] | \int_0^{\sigma} f(t)dt = -\delta\}$. Dieses existiert, da $F(\cdot) := \int_0^{\cdot} f(t)dt$ eine stetige Funktion ist und $F(\tau) \leq -\delta$, $F(0) = 0 > -\delta$ ist.

Im Folgenden wird mehrmals folgende Ungleichung verwendet:
Wenn $\psi : [a,b] \to \mathbb{R}$ eine integrierbare Funktion ist mit $\psi(t) \geq 0\ \forall t \in [a,b]$, $0 \leq a < b$, $\rho > 0$ beliebig, so gilt: $e^{\rho a} \int_a^b \psi(t)dt \leq \int_a^b e^{\rho t}\psi(t)dt \leq e^{\rho b} \int_a^b \psi(t)dt$. Dies ist eine Folgerung aus der Monotonie der e-Funktion und des Integrals sowie der Positivität von ψ.

1. Fall: Sei $\int_0^{\tilde{\tau}} f^+(t)dt = 0 \implies \int_0^{\tilde{\tau}} e^{\rho t} f^+(t)dt \leq e^{\rho\tilde{\tau}} \int_0^{\tilde{\tau}} f^+(t)dt = 0$. Daraus folgt:

$$\int_0^{\tilde{\tau}} e^{\rho t} f(t)dt = -\int_0^{\tilde{\tau}} e^{\rho t} f^-(t)dt \leq -\int_0^{\tilde{\tau}} f^-(t)dt = \int_0^{\tilde{\tau}} f(t)dt \leq -\delta$$

Also ist die Behauptung erfüllt.

2. Fall: Sei $\int_0^{\tilde{\tau}} f^+(t)dt > 0$. Also ist $-\int_0^{\tilde{\tau}} f^-(t)dt < -\delta$.

Wähle $\gamma \in [0,\tilde{\tau})$ so, daß $-\int_0^{\gamma} f^-(t)dt = -\delta$. Dieses existiert ebenfalls, da $\int_0^{\cdot} f^-(t)dt$ stetig ist.

Setze nun $f_1^-,\ f_2^- : \mathbb{R} \to \mathbb{R}$ wie folgt:

$$f_1^-(t) = \begin{cases} f^-(t) & \text{falls } t \leq \gamma \\ 0 & \text{sonst} \end{cases}, \quad f_2^-(t) = \begin{cases} f^-(t) & \text{falls } t > \gamma \\ 0 & \text{sonst} \end{cases}$$

und $f_2 : \mathbb{R} \to \mathbb{R}$ als $f_2(t) := f^+(t) - f_2^-(t)$. (Dann folgt $f = -f_1^- + f_2$.) Wegen $\int_0^{\tilde{\tau}} f^+(t)dt > 0$ kann o.B.d.A. angenommen werden, daß $\int_0^{\varepsilon} f^+(t)dt > 0$ für beliebiges $\varepsilon > 0$. (Ansonsten kann die untere Integrationsgrenze entsprechend nach oben geschoben werden. f_2^- muß wegen der Minimalität von $\tilde{\tau}$ mindestens bis zu diesem Zeitpunkt gleich Null sein, ansonsten wäre das Integral über f bereits kleiner oder gleich $-\delta$.) Ebenfalls wegen der Minimalität von $\tilde{\tau}$ gilt

$$\int_0^{t_1} f_2(t)dt > 0\ \forall t_1 \in (0,\tilde{\tau}) \quad \text{und} \quad \int_0^{\tilde{\tau}} f_2(t)dt = 0,$$

also auch

$$\int_0^{t_1} f_2(t)dt - \int_{t_1}^{\tilde{\tau}} f_2^-(t)dt \leq \int_0^{\tilde{\tau}} f_2(t)dt = 0$$

Setze nun $\Psi(t_1) := \min\{t_2 > t_1 | \int_{t_1}^{t_2} f_2^-(t_1)dt = \int_0^{t_1} f_2(t)dt\}$. Sei $\varepsilon \in (0, \tilde{\tau})$ beliebig. Dann ist $\int_0^{t_1} f_2(t)dt > \delta(\varepsilon)\ \forall t_1 \in [\varepsilon, \tilde{\tau} - \varepsilon]$, da das Integral stetig in t_1 ist und auf dem gesamten kompakten Intervall $[\varepsilon, \tilde{\tau} - \varepsilon]$ echt grösser als Null ist. Zusammen mit der Beschränktheit von f ergibt sich

$$\delta(\varepsilon) \leq \int_{t_1}^{\Psi(t_1)} f_2^-(t)dt \leq (\Psi(t_1) - t_1)M_g \quad \Longrightarrow \quad \Psi(t_1) - t_1 \geq \frac{\delta(\varepsilon)}{M_g}\ \forall t_1 \in [\varepsilon, \tilde{\tau} - \varepsilon]$$

Man kann also $[0, \tilde{\tau}]$ wie folgt in k Teilintervalle $[\tau_i, \tau_{i+1}]$ zerlegen:
$\tau_0 := 0,\ \tau_1 := \varepsilon,\ \tau_{i+1} := \Psi(\tau_i)$ solange, bis $\tau_{i+1} \geq \tilde{\tau} - \varepsilon$ ist, $\tau_k := \tilde{\tau}$. Dann ist $\tau_k \geq \Psi(\tau_{k-1})$ und $\tau_k - \tau_{k-1} < \varepsilon$. Wegen $\Psi(\tau_i) - \tau_i \geq \frac{\delta(\varepsilon)}{M_g}$ ist die Anzahl der Intervalle tatsächlich endlich und es gilt für $i = 2, \ldots, k-1$:

$$\begin{aligned}
&\int_{\tau_{i-1}}^{\tau_i} f^+(t)dt - \int_{\tau_i}^{\tau_{i+1}} f_2^-(t)dt \leq \int_{\tau_{i-1}}^{\tau_i} f^+(t)dt - \int_{\tau_i}^{\Psi(\tau_i)} f_2^-(t)dt \\
&= \int_{\tau_{i-1}}^{\tau_i} f^+(t)dt - \int_0^{\tau_i} f_2(t)dt \\
&= \int_{\tau_{i-1}}^{\tau_i} f^+(t)dt - \int_0^{\tau_{i-1}} f_2(t)dt - \int_{\tau_{i-1}}^{\tau_i} f^+(t)dt + \int_{\tau_{i-1}}^{\tau_i} f_2^-(t)dt \\
&= -\int_0^{\tau_{i-1}} f_2(t)dt + \int_{\tau_{i-1}}^{\tau_i} f_2^-(t)dt = 0, \qquad (4.1)
\end{aligned}$$

da $\tau_i = \Psi(\tau_{i-1})$ für $i = 2, \ldots, k-1$. Also ist:

$$\begin{aligned}
\int_0^{\tilde{\tau}} e^{\rho t} f_2(t)dt &= \underbrace{\int_0^{\tau_1} e^{\rho t} f_2(t)dt}_{\leq \varepsilon M_g} + \int_{\tau_1}^{\tau_{k-1}} e^{\rho t} f^+(t)dt + \underbrace{\int_{\tau_{k-1}}^{\tau_k} e^{\rho t} f^+(t)dt}_{\leq \varepsilon M_g} - \int_{\tau_1}^{\tau_k} e^{\rho t} f_2^-(t)dt \\
&\leq \sum_{i=1}^{k-2} \int_{\tau_i}^{\tau_{i+1}} e^{\rho t} f^+(t)dt - \sum_{i=1}^{k-1} \int_{\tau_i}^{\tau_{i+1}} e^{\rho t} f_2^-(t)dt + 2\varepsilon M_g \\
&= \sum_{i=2}^{k-1} \left(\int_{\tau_{i-1}}^{\tau_i} e^{\rho t} f^+(t)dt - \int_{\tau_i}^{\tau_{i+1}} e^{\rho t} f_2^-(t)dt \right) - \underbrace{\int_{\tau_1}^{\tau_2} e^{\rho t} f_2^-(t)dt}_{\geq 0} + 2\varepsilon M_g
\end{aligned}$$

$$\begin{aligned} &\leq \sum_{i=2}^{k-1} e^{\rho\tau_i} \underbrace{\left(\int_{\tau_{i-1}}^{\tau_i} f^+(t)dt - \int_{\tau_i}^{\tau_{i+1}} f_2^-(t)dt \right)}_{\leq 0 \text{ wegen (4.1)}} + 2\varepsilon M_g \\ &\leq 2\varepsilon M_g \end{aligned}$$

Da aber $\varepsilon \in (0, \tilde{\tau})$ beliebig gewählt war, folgt $\int_0^{\tilde{\tau}} e^{\rho t} f_2(t)dt \leq 0$.

Für f gilt daher

$$\begin{aligned} \int_0^{\tilde{\tau}} e^{\rho t} f(t)dt &= \int_0^{\tilde{\tau}} e^{\rho t} f_2(t)dt - \int_0^{\tilde{\tau}} e^{\rho t} f_1^-(t)dt \\ &\leq 0 - \int_0^{\tilde{\tau}} f_1^-(t)dt = -\delta \end{aligned}$$

und somit die Behauptung. □

Mit diesen Vorbereitungen können wir nun den Approximationssatz beweisen.

Satz 4.3 (Approximationssatz)
Seien $\rho > 0$, $x_0 \in \mathbb{R}^n$, $u(\cdot) \in \mathcal{U}$, $C \in \mathbb{R}$, $\delta > 0$ gegeben, so daß
$\rho J_\rho(\varphi(t, x_0, u(\cdot)), u(t + \cdot)) \leq C - \delta \ \forall t \geq 0$. Dann gilt:

$$J_0(x_0, u(\cdot)) < C.$$

Beweis: O.B.d.A. kann $C = 0$ angenommen werden. Ansonsten kann statt g die Funktion $g - C$ betrachtet werden, für die gilt:

$$\begin{aligned} &\rho \int_t^\infty e^{-\rho(\tau - t)} \Big(g(\varphi(\tau, x_0, u(\cdot)), u(\tau + \cdot)) - C \Big) d\tau \\ &= \rho \int_t^\infty e^{-\rho(\tau - t)} g(\varphi(\tau, x_0, u(\cdot)), u(\tau + \cdot))\, d\tau - \rho \frac{C}{\rho} \leq -\delta \end{aligned}$$

und

$$\begin{aligned} &\limsup_{T \to \infty} \frac{1}{T} \int_0^T \Big(g(\varphi(\tau, x_0, u(\cdot)), u(\tau + \cdot)) - C \Big) d\tau \\ &= \limsup_{T \to \infty} \frac{1}{T} \int_0^T g(\varphi(\tau, x_0, u(\cdot)), u(\tau + \cdot))\, d\tau \ - \ C \end{aligned}$$

Für $C = 0$ gibt es nun unter Anwendung von Lemma 4.1 zu jedem $t > 0$ ein $\tau(t)$, so daß gilt: $\int_t^{\tau(t)} e^{-\rho(s-t)} g(\varphi(s, x, u(\cdot)), u(s))dt < -\frac{\delta}{2\rho}$. Nach Lemma 4.2 gibt es dann auch ein $\tilde{\tau}(t) \leq \tau(t)$ mit $\int_t^{\tilde{\tau}(t)} g(\varphi(s, x, u(\cdot)), u(s))dt < -\frac{\delta}{2\rho}$. Außerdem gilt $\tilde{\tau}(t) - t \in [c, b]$, wobei c, b nach Lemma 4.2 und 4.1 nur von δ, ρ und M_g abhängen, nicht von t.
Sei nun $T > 0$ gegeben. Das Intervall $[0, T]$ kann wie folgt in k Teilintervalle $[\tilde{\tau}_i, \tilde{\tau}_{i+1}]$ zerlegt werden:
Setze $\tilde{\tau}_0 := 0$, $\tilde{\tau}_{i+1} := \tilde{\tau}(\tilde{\tau}_i)$, solange $\tilde{\tau}(\tilde{\tau}_i) \leq T$, $\tilde{\tau}_k := T$.
Dann ist $c \leq \tilde{\tau}_{i+1} - \tilde{\tau}_i \leq b \ \forall i = 0, \ldots, k-1$ und daher $\frac{T}{b} \leq k \leq \frac{T}{c}$. Außerdem ist $\int_{\tilde{\tau}_i}^{\tilde{\tau}_{i+1}} g(\varphi(t, x, u(\cdot)), u(t))dt < -\frac{\delta}{2\rho} \ \forall i = 0, \ldots, k-2$. Damit gilt:

$$\begin{aligned}
& \int_0^T g(\varphi(t, x, u(\cdot)), u(t))dt \\
= & \sum_{i=0}^{k-2} \int_{\tilde{\tau}_i}^{\tilde{\tau}_{i+1}} g(\varphi(t, x, u(\cdot)), u(t))dt + \int_{\tilde{\tau}_{k-1}}^{\tilde{\tau}_k} g(\varphi(t, x, u(\cdot)), u(t))dt \\
\leq & -\frac{k\delta}{2\rho} + (\tilde{\tau}_k - \tilde{\tau}_{k-1})M_g \quad \leq \quad -\frac{T\delta}{2b\rho} + bM_g
\end{aligned}$$

Also folgt:

$$\limsup_{T\to\infty} \frac{1}{T} \int_0^T g(\varphi(t, x, u(\cdot)), u(t))dt \leq \limsup_{T\to\infty} \ -\frac{\delta}{2b\rho} + \frac{bM_g}{T} = -\frac{\delta}{2b\rho} < 0$$

□

Bemerkung 4.4 Analog zeigt sich die „symmetrische“ Behauptung:

Seien $\rho > 0$, $x_0 \in \mathbb{R}^n$, $u(\cdot) \in \mathcal{U}$, $C \in \mathbb{R}$, $\delta > 0$ gegeben, so daß
$\rho J_\rho(\varphi(t, x_0, u(\cdot)), u(t + \cdot)) \geq C + \delta \ \forall t \geq 0$.
Dann gilt: $J_0(x_0, u(\cdot)) > C$.

Wir können ein schwächeres Resultat formulieren, wenn es kein $\delta > 0$ gibt, wie es in der Voraussetzung des Approximationssatzes gefordert ist.

Korollar 4.5 Seien $\rho > 0$, $x_0 \in \mathbb{R}^n$, $u(\cdot) \in \mathcal{U}$ und $C \in \mathbb{R}$ gegeben, so daß
$\rho J_\rho(\varphi(t, x_0, u(\cdot)), u(t + \cdot)) \leq C \ \forall t \geq 0$.
Dann gilt: $J_0(x_0, u(\cdot)) \leq C$.

Beweis: Zu beliebigem $\varepsilon > 0$ setze $\tilde{C} := C + \varepsilon$. Dann ist $\rho J_\rho(\varphi(t, x_0, u(\cdot)), u(t + \cdot)) \leq \tilde{C} - \varepsilon$ $\forall t \geq 0$. Nach Satz 4.3 ist also $J_0(x_0, u(\cdot)) < \tilde{C} = C + \varepsilon$. Da $\varepsilon > 0$ beliebig war, folgt so die Behauptung. □

Kapitel 5

Numerische Lösung des diskontierten optimalen Steuerungsproblems

In diesem Kapitel werden wir in den ersten drei Abschnitten ein Verfahren zur Berechnung der optimalen Wertefunktion diskontierter optimaler Steuerugsprobleme herleiten. Dazu werden wir zunächst eine Diskretisierung der Bellman-Gleichung

$$\rho v_\rho(x_0) + \sup_{u \in U}\{-g(x_0,u) - Dv_\rho(x_0)\cdot f(x_0,u)\} = 0 \tag{5.1}$$

betrachten (vgl. (2.16)) und einige Eigenschaften dieser *diskretisierten Bellman-Gleichung* herleiten. Die Ideen hierzu gehen zurück auf Falcone [10], [11] und sind auch im vierten Kapitel von Sorgenfrei [16] ausgeführt; einige Beweise sind hier jedoch abgeändert oder vollständig anders geführt.

Im Anschluß werden wir ein numerisches Verfahren herleiten, das uns diese diskretisierte Bellman-Gleichung löst. Aufbauend auf der so berechneten Wertefunktion werden wir im vierten Abschnitt dann Zustandsfeedbacks für das diskretisierte optimale Steuerungsproblem konstruieren und zeigen, daß diese auch das ursprüngliche diskontierte optimale Steuerungsproblem approximativ optimal lösen.

5.1 Die diskretisierte Bellman-Gleichung

In diesem Abschnitt werden wir die *diskretisierte Bellman-Gleichung*

$$\sup_{u \in U}\{v_h(x) - \beta v_h(\Psi(x + hf(x,u))) - hg(x,u)\} = 0 \tag{5.2}$$

mit $x \in W$, $\rho, h > 0$ und $\beta := 1-\rho h$ betrachten. Zunächst werden wir zeigen, daß Gleichung (5.2) tatsächlich eine Approximation von Gleichung (5.1) ist.

Danach werden wir zeigen, daß die Lösung der diskretisierten Bellman-Gleichung die Wertefunktion des diskretisierten optimalen Steuerungsproblems ist; wie bei Gleichung (5.1) läßt sich auch hier ein Eindeutigkeitsresultat formulieren.

Zum Abschluß wird noch eine Abschätzung des Diskretisierungsfehlers hergeleitet.

Bei den Abschätzungen in diesem Kapitel wird davon ausgegangen, daß die Parameter ρ und h nach oben beschränkt sind. So können Terme, die von ρ oder h abhängen, durch Konstanten abgeschätzt werden. Dies bedeutet keine Einschränkung, da es in den hier behandelten Beispielen nicht sinnvoll ist, diese Parameter beliebig groß zu wählen.

5.1.1 Approximation der Bellman-Gleichung

Lemma 5.1 Seien ρ, h positive reelle Zahlen und $\beta := 1 - \rho h$.
Dann ist die Gleichung (5.2) eine Approximation der Gleichung (5.1).

Beweis: $v : W \to \mathbb{R}$ erfülle (5.2) und sei differenzierbar in $x \in W$. Dann folgt

$$\begin{aligned} & v(x) + \sup_{u\in U}\{-\beta v(\Psi(x + hf(x,u))) - hg(x,u)\} = 0 \\ \iff & \sup_{u\in U}\{v(x) - v(\Psi(x + hf(x,u))) + \rho h v(\Psi(x + hf(x,u))) - hg(x,u)\} = 0 \\ \iff & \sup_{u\in U}\left\{-\frac{v(\Psi(x + hf(x,u))) - v(x)}{h} + \rho v(\Psi(x + hf(x,u))) - g(x,u)\right\} = 0. \end{aligned}$$

Da W offen ist und Ψ auf W die Identität ist, gilt für hinreichend kleine $h > 0$:

$$\sup_{u\in U}\left\{-\frac{v(x + hf(x,u)) - v(x)}{h} + \rho v(x + hf(x,u)) - g(x,u)\right\} = 0.$$

Mit Grenzübergang $h \to 0$ ergibt sich:

$$\begin{aligned} & \lim_{h\to 0}\sup_{u\in U}\left\{-\frac{v(x + hf(x,u)) - v(x)}{h} + \rho v(x + hf(x,u)) - g(x,u)\right\} \\ & \quad = \sup_{u\in U}\{Dv(x)\cdot f(x,u) + \rho v(x) - g(x,u)\}. \end{aligned}$$

□

5.1.2 Das diskretisierte optimale Steuerungsproblem

Wir werden nun zeigen, daß die Lösung von (5.2) die Wertefunktion des mittels des Euler-Rekursionsschemas diskretisierten optimalen Steuerungsproblems ist. Dazu ersetzen wir die rechte Seite (2.1) für $x \in W$ durch

$$x_0 := x, \quad x_{j+1} := \Psi(x_j + hf(x_j, u_j)), \quad j = 0,\, 1,\, 2, \dots, \tag{5.3}$$

wobei gilt

$$h > 0, \quad u_j := u(jh). \tag{5.4}$$

Die diskretisierte Trajektorie ist also definiert durch

$$\varphi_h(t,x,u(\cdot)) := x_j \text{ für } t \in [jh,(j+1)h).$$

Wir bezeichnen mit $\mathcal{U}_h \subset \mathcal{U}$ die Menge aller Kontrollfunktionen, die auf jedem Intervall $[jh,(j+1)h)$ konstant sind. Jeder Kontrolle $u \in \mathcal{U}$ kann dann eindeutig ein $u_h \in \mathcal{U}_h$ zugeordnet werden, so daß $\varphi_h(t,x,u(\cdot)) = \varphi_h(t,x,u_h(\cdot))$ gilt.
Das diskretisierte Kostenfunktional J_h zu einem Anfangswert $x \in \mathbb{R}^n$ und $u(\cdot) \in \mathcal{U}_h$ ist gegeben durch die Reihe

$$J_h(x,u(\cdot)) := h\sum_{j=0}^{\infty} \beta^j g(x_j,u_j).$$

Die diskretisierte Wertefunktion ist nun definiert durch

$$v_h(x) := \inf_{u(\cdot)\in\mathcal{U}_h} J_h(x,u(\cdot)). \tag{5.5}$$

Wir zeigen jetzt, daß die diskretisierte Wertefunktion die eindeutig bestimmte beschränkte Lösung der diskretisierten Bellman-Gleichung ist. Hierzu zeigen wir zunächst ein Analogon zum Bellman'schen Optimalitätsprinzip.

Satz 5.2 (Diskretes Bellman-Prinzip)
Für alle $x = x_0 \in W$ und alle $p \in \mathbb{N}$ gilt

$$v_h(x) = \inf_{u(\cdot)\in\mathcal{U}_h} \left\{ h\sum_{j=0}^{p-1} \beta^j g(x_j,u_j) + \beta^p v_h(x_p) \right\}. \tag{5.6}$$

Beweis: Sei $x = x_0 \in W$ beliebig aber fest gewählt. Wegen (5.5) existiert zu jedem $\varepsilon > 0$ ein $u^\varepsilon(\cdot) \in \mathcal{U}$, so daß

$$v_h(x) + \varepsilon \geq J_h(x,u^\varepsilon(\cdot)).$$

Für beliebiges $p \geq 1$ gilt

$$J_h(x,u^\varepsilon(\cdot)) = h\sum_{j=0}^{p-1} \beta^j g(x_j^\varepsilon,u_j^\varepsilon) + \underbrace{\beta^p h\sum_{j=0}^{\infty} \beta^j g(x_{j+p}^\varepsilon,u_{j+p}^\varepsilon)}_{=\beta^p J_h(x_p^\varepsilon,u^\varepsilon(\cdot+ph))},$$

mit $x_j^\varepsilon = \varphi_h(jh,x,u^\varepsilon(\cdot))$. Also folgt

$$\begin{aligned} v_h(x)+\varepsilon &\geq h\sum_{j=0}^{p-1} \beta^j g(x_j^\varepsilon,u_j^\varepsilon) + \beta^p J_h(x_p^\varepsilon,u^\varepsilon(\cdot+ph)) \\ &\geq h\sum_{j=0}^{p-1} \beta^j g(x_j^\varepsilon,u_j^\varepsilon) + \beta^p v_h(x_p^\varepsilon) \\ &\geq \inf_{u(\cdot)\in\mathcal{U}_h} \left\{ h\sum_{j=0}^{p-1} \beta^j g(x_j,u_j) + \beta^p v_h(x_p) \right\}, \end{aligned}$$

woraus die eine Richtung folgt. Die andere Richtung wird ähnlich bewiesen, vgl. den Beweis zu Satz 2.10. □

Satz 5.3 Sei g nach (2.9) beschränkt für alle $(x,u) \in W \times U$ und $h \in (0, \frac{1}{\rho})$.
Dann ist die durch (5.5) definierte Funktion v_h die eindeutig bestimmte, beschränkte Lösung von (5.2).

Beweis: Die Beschränktheit von v_h folgt sofort aus der Beschränktheit von g. Wir zeigen nun zunächst, daß v_h die diskrete Bellman-Gleichung (5.2) erfüllt.
Betrachte dazu (5.6) mit $p = 1$. Dann folgt

$$\begin{aligned} v_h(x) &= \inf_{u(\cdot)\in\mathcal{U}_h} \{hg(x_0,u_0) + \beta v_h(x_1)\} \\ &= \inf_{u\in U} \{hg(x,u) + \beta v_h(\Psi(x+hf(x,u)))\} \\ &= -\sup_{u\in U} \{-hg(x,u) - \beta v_h(\Psi(x+hf(x,u)))\}, \end{aligned}$$

also erfüllt v_h Gleichung (5.2).
Seien nun v_h^1 und v_h^2 zwei beschränkte Lösungen von (5.2). Analog zum Beweis von Korollar 2.11 ergibt sich dann

$$\sup_{x\in W} |v_h^1(x) - v_h^2(x)| \le \beta \sup_{x\in W} |v_h^1(x) - v_h^2(x)|,$$

also wegen $\beta = 1 - \rho h < 1$ die Eindeutigkeit. □

5.1.3 Eigenschaften der Lösung der diskretisierten Bellman-Gleichung

In diesem Unterabschnitt werden wir die diskrete Bellman-Gleichung in eine äquivalente Fixpunktgleichung umformen. Dies ermöglicht es uns dann, mit dem Banach'schen Fixpunktsatz die Existenz einer eindeutig bestimmten, Hölder stetigen Lösung v_h nachzuweisen. Darüber hinaus wird sich zeigen, daß der Hölder-Exponent der diskreten Wertefunktion mit dem der exakten Wertefunktion identisch ist.

Danach werden wir Abschätzungen für v_h herleiten, aus denen dann die lokal gleichmäßige Konvergenz von v_h gegen die Viskositätslösung v von (5.1) für $h \searrow 0$ folgt.

Satz 5.4 Sei $\gamma \in (0,1]$ der Hölder-Exponent der Wertefunktion v. Weiterhin sei $\rho > 0$ und $h \in [0, \frac{1}{\rho})$.
Dann besitzt die diskrete Bellman-Gleichung (5.2) eine eindeutig bestimmte Lösung $v_h \in \mathcal{H}_\gamma$.
Außerdem gelten mit ρ, M_g und einer Konstanten $L = L(\gamma)$ die Abschätzungen

$$\sup_{x\in\mathbb{R}^n} |v_h(x)| \le \frac{M_g}{\rho} \text{ und} \tag{5.7}$$

$$|v_h|_{0,\gamma} \le L. \tag{5.8}$$

Beweis: Äquivalent zu (5.2) ist die Fixpunktgleichung

$$v_h(x) = T_h v_h(x), \quad x \in \mathbb{R}^n, \tag{5.9}$$

wobei der Operator T_h gegeben ist durch

$$T_h v(x) := \min_{u \in U} \Big(\beta v(\Psi(x + hf(x,u))) + hg(x,u) \Big), \quad x \in \mathbb{R}^n. \tag{5.10}$$

Im ersten Schritt zeigen wir die Eindeutigkeit der Lösung. Seien dazu $v_h^1, v_h^2 \in \mathcal{H}_\gamma$ zwei Lösungen und $u^1, u^2 \in U$ zwei Kontrollwerte, so daß das Minimum in (5.10) angenommen wird. Dann gilt

$$\begin{aligned}
&T_h v_h^1(x) - T_h v_h^2(x) \\
&\quad = \beta\Big(v_h^1(\Psi(x + hf(x,u^1))) - v_h^2(\Psi(x + hf(x,u^2))) \Big) + h\Big(g(x,u^1) - g(x,u^2) \Big) \\
&\quad \le \beta\Big(v_h^1(\Psi(x + hf(x,u^2))) - v_h^2(\Psi(x + hf(x,u^2))) \Big) + h\Big(g(x,u^2) - g(x,u^2) \Big) \\
&\quad \le \beta \sup_{x \in W} |(v_h^1 - v_h^2)(x)|,
\end{aligned}$$

und aus Symmetriegründen folgt

$$\sup_{x \in W} |T_h v_h^1(x) - T_h v_h^2(x)| \;\le\; \beta \sup_{x \in W} |(v_h^1 - v_h^2)(x)|, \tag{5.11}$$

d.h. der Operator T_h ist eine kontrahierende Abbildung.
Mit dem Banach'schen Fixpunktsatz folgt nun, daß es eine eindeutig bestimmte, beschränkte Funktion v_h gibt, die (5.2) erfüllt.

Im zweiten Schritt ist nun zu zeigen, daß v_h tatsächlich in $\mathcal{H}_\gamma$ liegt. Nach der Definition von $\mathcal{H}_\gamma$ sind dafür die beiden Ungleichungen (5.7) und (5.8) zu zeigen.
Abschätzung (5.7) ist eine Folgerung aus Satz 5.3; für beliebiges $u \in \mathcal{U}_h$ gilt

$$\begin{aligned}
v_h(x) \;\le\; J_h(x,u) \;&=\; h \sum_{j=0}^{\infty} \beta^j g(x_j, u_j) \\
&\le\; h \sum_{j=0}^{\infty} \beta^j M_g \;=\; h \frac{1}{1-\beta} M_g \;=\; \frac{M_g}{\rho}.
\end{aligned}$$

Um Abschätzung (5.8) zu zeigen, betrachten wir

$$\begin{aligned}
|v_h(x) - v_h(y)| &= |\inf_{u \in \mathcal{U}_h} J_h(x,u) - \inf_{u \in \mathcal{U}_h} J_h(y,u)| \\
&\le \sup_{u \in \mathcal{U}_h} |J_h(x,u) - J_h(y,u)| \\
&= \sup_{u \in \mathcal{U}_h} \left| h \sum_{j=0}^{\infty} \beta^j g(\varphi_h(jh, x, u(\cdot)), u(jh)) + h \sum_{j=0}^{\infty} \beta^j g(\varphi_h(jh, y, u(\cdot)), u(jh)) \right| \\
&= \sup_{u \in \mathcal{U}_h} \left| \int_0^{\infty} \beta^{[\frac{t}{h}]} (g(\varphi_h(t, x, u(\cdot)), u(t)) - g(\varphi_h(t, y, u(\cdot)), u(t))) dt \right|
\end{aligned}$$

$$\leq \sup_{u\in\mathcal{U}_h} \left| \int_0^\infty e^{-\rho[\frac{t}{h}]h}(g(\varphi_h(t,x,u(\cdot)),u(t)) - g(\varphi_h(t,y,u(\cdot)),u(t)))dt \right|$$

$$\leq \sup_{u\in\mathcal{U}_h} \left| e^{\rho h} \int_0^\infty e^{-\rho t} \underbrace{(g(\varphi_h(t,x,u(\cdot)),u(t)) - g(\varphi_h(t,y,u(\cdot)),u(t)))}_{=:\Phi_h(t)} dt \right|$$

Hierbei bezeichnet die *Gaußklammer* $[x]$ für reelle x die größte ganze Zahl, die kleiner oder gleich x ist.
Für Φ_h gilt nun wegen (2.9) zum einen die Abschätzung

$$\Phi_h(t) \leq 2M_g. \tag{5.12}$$

Zum anderen folgt aus Lemma A.9 im Anhang, daß gilt

$$\Phi_h(t) \leq CL_g\|x-y\|e^{(L_f+C^*)t}$$

mit C, C^* wie in Ungleichung (2.14). Für $\Psi = \mathrm{id}_{\mathbb{R}^n}$ folgt aus (A.17) diese Abschätzung mit $C = 1$ und $C^* = 0$. Damit ergibt sich die Zwischenbehauptung

$$\Phi_h(t) \leq \min\{CL_g\|x-y\|e^{(L_f+C^*)t}, 2M_g\}.$$

Analog zum Beweis der Hölder-Stetigkeit von v_ρ (Satz 2.15) folgt nun mit Lemma 2.14 die Behauptung. □

Bemerkung 5.5 Mit Blick auf das Lemma 2.14 sieht man, daß die Konstante L in (5.8) für $\rho \leq L_f$ vom Faktor $\frac{1}{\rho}$ abhängt, d.h. für kleine Diskontraten ρ ist $L \approx \frac{C}{\rho}$ mit C unabhängig von ρ. (Die gleiche Abschätzung gilt für v_ρ.)

Das folgende Korollar ergibt sich sofort aus Satz 5.4:

Korollar 5.6 Für die Norm der diskreten Wertefunktion v_h gilt $\|v_h\|_{\mathcal{H}_\gamma} < \infty$.

5.1.4 Diskretisierungsfehler

In diesem Abschnitt wird gezeigt, daß v_h für $h \searrow 0$ gleichmäßig gegen v_ρ konvergiert. Dies zeigen wir, indem wir den maximalen Diskretisierungsfehler auf dem $\mathbb{R}^n$ abschätzen.

Zunächst machen wir diese Abschätzung für die Zielfunktionale:

Satz 5.7 Es seien die Bedingungen (2.4) – (2.9) an f und g erfüllt. Dann gilt

$$|J_h(x,u(\cdot)) - J_\rho(x,u(\cdot))| \leq L_1 h^\delta + L_2 h \quad \forall u(\cdot)\in\mathcal{U},\ \ x\in W,\ \ h\in(0,\frac{1}{\rho}).$$

Hierbei ist $0 < \delta \leq 1$ und $\delta = \gamma$, falls $\Psi \equiv \mathrm{id}_{\mathbb{R}^n}$. Für kleine $\rho > 0$ gilt $L_1 \approx \frac{C_1}{\rho}$ und $L_2 \approx \frac{C_2}{\rho}$ für positive Konstanten C_1, $C_2 > 0$.

Beweis:

$$
\begin{aligned}
&|J_h(x,u(\cdot)) - J_\rho(x,u(\cdot))| \\
&\quad \le \int_0^\infty |g(\varphi_h(s,x,u(\cdot)),u(s)) - g(\varphi(s,x,u(\cdot)),u(s))| e^{-\rho s} ds \qquad (5.13)\\
&\quad + \int_0^\infty |g(\varphi_h(s,x,u(\cdot)),u(s))|\, |e^{-\rho s} - e^{-\theta\rho\left[\frac{s}{h}\right]h}| ds, \qquad (5.14)
\end{aligned}
$$

wobei gilt $\theta = \theta(\rho,h) = -\frac{1}{\rho h}\ln(1-\rho h)$. Hierbei gilt $\theta > 1$, wegen $1-\rho h < e^{-\rho h} \Rightarrow \ln(1-\rho h) < -\rho h \Rightarrow -\frac{1}{\rho h}\ln(1-\rho h) > 1$.
Für den Term (5.13) können wir den Integranden mit Lemma A.10 aus dem Anhang abschätzen durch

$$|g(\varphi_h(t,x,u(\cdot)),u(s)) - g(\varphi(t,x,u(\cdot)),u(s))| \le ChL_g e^{(C^{**}+L_f)t},$$

wobei wir für $\Psi \equiv \mathrm{id}_{\mathbb{R}^n}$ diese Abschätzung mit $C = M_f$ und $C^{**} = 0$ durch das Gronwall-Lemma (A.7) erhalten. Unter Beachtung der Beschränktheit von g ergibt sich

$$|g(\varphi_h(t,x,u(\cdot)),u(s)) - g(\varphi(t,x,u(\cdot)),u(s))| \le \min\{ChL_g e^{(C^{**}+L_f)t}, 2M_g\}.$$

Mit Lemma 2.14 folgt also

$$\int_0^\infty |g(\varphi_h(s,x,u(\cdot)),u(s)) - g(\varphi(s,x,u(\cdot)),u(s))| e^{-\rho s} ds \le L_1 h^\delta$$

mit einer Konstanten $L_1 > 0$ und $0 < \delta \le 1$, wobei δ für $C^{**} = 0$ genau der Hölder-Exponent der Wertefunktionen ist (vgl. Beweis zu Satz 2.15).
Den Term (5.14) können wir wie folgt abschätzen:

$$
\begin{aligned}
&\int_0^\infty |g(\varphi_h(s,x,u(\cdot)),u(s))|\, |e^{-\rho s} - e^{-\theta\rho\left[\frac{s}{h}\right]h}| ds \\
&\quad \le M_g \int_0^\infty |e^{-\rho s} - e^{-\theta\rho\left[\frac{s}{h}\right]h}| ds \\
&\quad \le M_g \int_0^\infty \left|\rho s - \theta\rho\left[\frac{s}{h}\right]h\right| \max\left\{e^{-\rho s}, e^{-\theta\rho s}\right\} ds \\
&\quad \le M_g\rho(|1-\theta| + h) \underbrace{\int_0^\infty (s+1)e^{-\rho s} ds}_{=\frac{1}{\rho}+\frac{1}{\rho^2}} \\
&\quad = M_g(|1-\theta|+h)(1+\frac{1}{\rho}) \;=\; M_g h(1+\frac{1}{\rho})(\frac{\theta-1}{h}+1) \\
&\quad \le M_g(1+\frac{1}{\rho})(1+\frac{\rho}{2}+o(\rho^2 h))h \;\le\; L_2 h.
\end{aligned}
$$

In der zweiten Ungleichung wird hier der Zwischenwertsatz für dehnungsbeschränkte Funktionen angewendet und die Monotonie der Exponentialfunktion ausgenutzt. Die dritte Ungleichung sieht man, wenn man die Integranden für festes $s \geq 0$ betrachtet und die letzte Ungleichung sieht man durch Taylor-Entwicklung von $\ln(1-\rho h)$ und Einsetzen in θ. □

Satz 5.8 Es seien die Bedingungen (2.4) – (2.9) an f und g erfüllt und es sei zu festem $\rho > 0$ v_ρ die Wertefunktion des diskontierten optimalen Steuerungsproblems und v_h die diskretisierte Wertefunktion.
Dann gilt die Abschätzung

$$\sup_{x \in W} |(v_\rho - v_h)(x)| \leq L_1 h^\delta + L_2 h \tag{5.15}$$

für alle $h \in (0, \frac{1}{\rho})$ mit den Konstanten L_1, L_2, $\delta > 0$ aus Satz 5.7.

Beweis: Sei zunächst $\varepsilon > 0$ gegeben. Zu vorgegebenem $x \in W$ seien $u_h(\cdot)$, $u_\rho(\cdot) \in \mathcal{U}$ Kontrollen, so daß die optimalen Werte bis auf ε angenommen werden. Dann gilt:

$$\begin{aligned} v_\rho(x) - v_h(x) &\leq v_\rho(x) - J_h(x, u_h(\cdot)) + \varepsilon \leq J_\rho(x, u_h(\cdot)) - J_h(x, u_h(\cdot)) + \varepsilon \\ &\leq \sup_{u(\cdot) \in \mathcal{U}} |J_\rho(x, u(\cdot)) - J_h(x, u(\cdot))| + \varepsilon. \end{aligned}$$

Aus Symmetriegründen und da $\varepsilon > 0$ beliebig war folgt

$$|v_\rho(x) - v_h(x)| \leq \sup_{u(\cdot) \in \mathcal{U}} |J_\rho(x, u(\cdot)) - J_h(x, u(\cdot))|$$

und damit mit Satz 5.7 die Behauptung. □

Korollar 5.9 Unter den Annahmen von Satz 5.8 gelten die folgenden Abschätzungen für kleine ρ mit Konstanten $C, C^{**} > 0$:

$$\begin{aligned} \sup_{\mathbb{R}^n} |v_\rho - v_h| &\leq \frac{C}{\rho} h \text{ für } \rho > L_f + C^{**}, \\ \sup_{\mathbb{R}^n} |v_\rho - v_h| &\leq \frac{C}{\rho}(h^\delta + h) \text{ mit } \delta \in (0,1) \text{ beliebig für } \rho = L_f + C^{**}, \\ \sup_{\mathbb{R}^n} |v_\rho - v_h| &\leq \frac{C}{\rho}(h^{\frac{\rho}{(L_f + C^{**})}} + h) \text{ für } \rho < L_f + C^{**}. \end{aligned}$$

Hierbei ist $C^{**} = 0$ falls $\Psi \equiv \mathrm{id}_{\mathbb{R}^n}$.

5.2 Diskretisierung im Zustandsraum

In diesem Abschnitt wird die diskrete Bellman-Gleichung (5.2) durch die Diskretisierung des Zustandsraumes auf ein endlichdimensionales Problem gebracht.

Um die Diskretisierung zu verwirklichen, benötigen wir eine beschränkte Teilmenge des $\mathbb{R}^n$. Dies ist möglich, wenn eine offene, beschränkte Teilmenge $\Omega \subset \mathbb{R}^n$ existiert, die positiv

invariant ist bezüglich der Dynamik (2.1) für alle Kontrollfunktionen.
Falcone hat in [11], Proposition 2.5 gezeigt, daß dann eine reguläre Triangulierung von Ω in eine endliche Zahl $P \in \mathbb{N}$ von Simplizes S_j durchgeführt werden kann, so daß die Menge $\Omega_k := \bigcup_{1\le j\le P} S_j$ positiv invariant für die diskretisierten Trajektorien ist, d.h. es gilt

$$\exists h > 0 : \quad x + hf(x,u) \in \Omega^k, \quad \forall (x,u) \in \Omega^k \times U,$$

wobei k den maximalen Durchmesser $\max_{1\le j\le P} \operatorname{diam}(S_j)$ der Simplizes bezeichnet.

Wenn die gegebene Menge $W \subset \mathbb{R}^n$ bereits beschränkt ist und die Invarianz durch die Funktion Ψ sichergestellt ist, können wir diese Menge als zu diskretisierenden Bereich wählen, d.h. wir setzen $\Omega := W$.

Wir bezeichnen nun mit x_i, $i = 1, \dots, N$ die Ecken der Simplizes, also die Knotenpunkte der Triangulierung. Anstelle von (5.2) betrachten wir nun das folgende System von N Gleichungen

$$v_h^k(x_i) + \sup_{u\in U} \Big(-\beta v_h^k(\Psi(x_i + hf(x_i,u))) - hg(x_i,u)\Big) = 0 \tag{5.16}$$

für $i = 1, \dots, N$. Für die weitere Betrachtung schreiben wir die Gleichung analog zu (5.9), (5.10) in Fixpunktform, d.h. wir erhalten für $i = 1, \dots, N$:

$$\begin{gathered} v_h^k(x_i) = T_h v_h^k(x_i) \\ \text{mit } \ T_h v_h^k(\cdot) = \min_{u\in U} \Big(\beta v_h^k(\Psi(\cdot + hf(\cdot,u))) + hg(\cdot,u)\Big). \end{gathered} \tag{5.17}$$

5.2.1 Finite-Differenzen-Approximation

Wir wollen nach einer Lösung von (5.16) im Raum $\mathcal{W}^k$ der stetigen, stückweise affinen Funktionen

$$\mathcal{W}^k := \Big\{ w \in C(\Omega^k) \,|\, \nabla w(x) = c_j \ \text{ in } \ S_j \Big\} \tag{5.18}$$

suchen. Dadurch kann dann die Zwischenwertberechnung in einem Punkt x mittels linearer Interpolation der Werte an den Ecken des Simplizes erfolgen, in dem x liegt.

Bemerkung 5.10 Ein Punkt $x \in \mathbb{R}^n$, der innerhalb eines Simplizes S mit den Eckpunkten $x_1, \dots, x_{n+1}$ liegt. läßt sich als eindeutige Konvexkombination

$$x = \sum_{j=1}^{n+1} \mu_j x_j \quad \text{mit} \quad \sum_{j=1}^{n+1} \mu_j = 1, \ \ \mu_j \ge 0 \ \ \forall j = 1, \dots, n+1$$

darstellen. Die μ_j heißen dann *baryzentrische Koordinaten* .

Auch für diesen eingeschränkten Lösungsraum läßt sich ein Existenz- und Eindeutigkeitsresultat formulieren.

Satz 5.11 Es seien (2.4) und (2.7) erfüllt (Stetigkeit von f und g).
Dann existiert für alle $h \in [0, \frac{1}{\rho})$ mit $\Psi(x + hf(x,u)) \in \Omega \ \ \forall (x,u) \in \Omega \times U$ eine eindeutige Lösung von (5.16) in $\mathcal{W}^k$.

Beweis: Für jede affine Funktion ergibt sich die Fixpunktgleichung (5.17) zu

$$v_h^k(x_i) = \min_{u\in U}\left(\beta\sum_{j=1}^{N}\lambda_j(x_i,u)v_h^k(x_j) + hg(x_i,u)\right), \tag{5.19}$$

wobei gilt

$$\begin{aligned}\lambda_j &\geq 0, \quad \forall j = 1,\ldots,N\\ \sum_{j=1}^{N}\lambda_j &= 1\\ \sum_{j=1}^{N}\lambda_j(x_i,u)x_j &= \Psi(x_i + hf(x_i,u)),\end{aligned}$$

d.h. jeder Punkt ist als Konvexkombination der Knotenpunkte darstellbar. Definiert man nun $\Lambda(u) \in \mathbb{R}^{N\times N}$ und $G(u) \in \mathbb{R}^N$ durch

$$\Lambda_{ij}(u) := \lambda_j(x_i,u), \quad G_i(u) := g(x_i,u), \tag{5.20}$$

so ergibt sich (5.19) zu

$$V^{n+1} = T_h(V^n) := \min_{u\in U}\Big(\beta\Lambda(u)V^n + hG(u)\Big) \tag{5.21}$$

mit $V^n = ([v_h^k]_n(x_1),\ldots,[v_h^k]_n(x_N)) \in \mathbb{R}^N$.
Wenn man die i-te Zeile von Λ mit Λ_i bezeichnet, gilt mit V, $W \in \mathbb{R}^N$:

$$|(T_h(V) - T_h(W))_i| \leq \underbrace{\beta}_{<1}\underbrace{\max_{u\in U}|\Lambda_i(u)|}_{\leq 1}\|V - W\|,$$

d.h. der Operator T_h ist eine kontrahierende Abbildung bzgl. der Maximumsnorm $\|\cdot\|$.
Also folgt die Existenz eines eindeutig bestimmten $V^* \in \mathbb{R}^N$ mit $T_h(V^*) = V^*$ mit dem Banachschen Fixpunktsatz. □

5.2.2 Diskretisierungsfehler

Wir werden nun den Diskretisierungsfehler bei der Diskretisierung des Zustandsraumes abschätzen:

Satz 5.12 Mit den Voraussetzungen von Satz 5.11 gilt für kleine $\rho > 0$ und für alle $k > 0$, $h \in (0, \frac{1}{\rho})$ die folgende Abschätzung:

$$\sup_{x\in\Omega^k}|v_h^k(x) - v_h(x)| \leq \frac{C}{\rho^2}\frac{k^\gamma}{h}, \tag{5.22}$$

mit einer Konstanten $C > 0$. Hierbei bezeichnet $\gamma \in (0,1]$ den Hölder-Exponenten der Wertefunktion.

Beweis: Für jedes $x \in \overline{\Omega^k}$ gilt

$$|v_h^k(x) - v_h(x)| \leq \sum_{j=1}^{N} \mu_j |v_h^k(x_j) - v_h(x_j)| + \sum_{j=1}^{N} \mu_j |v_h(x_j) - v_h(x)|.$$

Für den zweiten Term gilt mit Satz 5.4 und wegen $\|x_j - x\| \leq k$ die Abschätzung

$$\sum_{j=1}^{N} \mu_j |v_h(x_j) - v_h(x)| \leq \sum_{j=1}^{N} \mu_j L \|x_j - x\|^\gamma \leq L k^\gamma \sum_{j=1}^{N} \mu_j = L k^\gamma. \tag{5.23}$$

Aus (5.2) und (5.16) folgt für alle $j = 1, \ldots, N$:

$$\begin{aligned} |v_h^k(x_j) - v_h(x_j)| &\leq \beta |v_h^k(\Psi(x_j + hf(x_j, u^*))) - v_h(\Psi(x_j + hf(x_j, u^*)))| \\ &\leq \beta |v_h^k(\Psi(x_j + hf(x_j, u^*))) - v_h(\Psi(x_j + hf(x_j, u^*)))| \\ &\leq \beta \max_{x \in \overline{\Omega^k}} |v_h^k(x) - v_h(x)|, \end{aligned}$$

wobei $u^* \in U$ einen der beiden Kontrollwerte bezeichnet, die das Maximum in (5.2) bzw. (5.16) ergeben. Insgesamt erhalten wir

$$\max_{x \in \overline{\Omega^k}} |v_h^k(x) - v_h(x)| \leq \beta \max_{x \in \overline{\Omega^k}} |v_h^k(x) - v_h(x)| + L k^\gamma$$

und unter Berücksichtigung von Bemerkung 5.5

$$\max_{x \in \overline{\Omega^k}} |v_h^k(x) - v_h(x)| \leq \frac{C}{\rho^2} \frac{k^\gamma}{h}.$$

□

Die Abschätzungen über die Diskretisierungsfehler können wir zusammenfassen:

Korollar 5.13 Unter den obigen Voraussetzungen gilt für positive Konstanten C_1 und C_2 die Fehlerabschätzung

$$\max_{x \in \overline{\Omega^k}} |v_h^k(x) - v_\rho(x)| \leq \frac{C_1}{\rho} (h^\delta + h) + \frac{C_2}{\rho^2} \frac{k^\gamma}{h}. \tag{5.24}$$

Für $\Psi \equiv \mathrm{id}_{\mathbb{R}^n}$ ist $\delta = \gamma$.

Beweis: Folgt sofort aus den Sätzen 5.8 und 5.12. □

Bemerkung 5.14 Aus dieser Abschätzung ergibt sich zum einen die Forderung, daß die Zeitschrittweite h gegenüber der Raumschrittweite k nicht zu klein werden darf.
Zum anderen folgt aus diesem Korollar, daß die Konvergenzrate mit kleiner werdendem ρ immer schlechter wird, d.h. kleine ρ bewirken zwar nach Satz 3.18 eine Annäherung an das Durchschnittskostenfunktional, ziehen aber eine sehr feine Diskretisierung nach sich, damit die numerischen Fehler nicht zu groß werden.

5.3 Berechnungsstrategien

In diesem Abschnitt wollen wir verschiedene Berechnungsstrategien diskutieren, mit denen die Fixpunktgleichung (5.17) gelöst werden kann. Zunächst werden kurz zwei Strategien vorgestellt, die in Falcone [10] vorgeschlagen wurden. Danach wird ein neuer Algorithmus entwickelt, der auch für die in Kapitel 7 beschriebenen numerischen Berechnungen verwendet wurde, und dort mit dem alten Algorithmus verglichen wird.

5.3.1 Sukzessive Approximation

Die erste Strategie zur Berechnung der Lösung der Gleichung (5.17) ergibt sich sofort aus der Eigenschaft, daß der Operator T_h eine kontrahierende Abbildung ist. Der Banach'sche Fixpunktsatz garantiert dann, daß die durch den Operator gegebene Rekursionsvorschrift gegen die Lösung von (5.17) konvergiert.

Das Grundgerüst des Verfahrens läßt sich also wie folgt beschreiben:

Wähle beliebige Anfangswerte in den Knotenpunkten

$$[v_h^k]_0(x_i) = v_h^k(x_i), \quad i = 1, \ldots, N.$$

Mit Berechnung des Zwischenwertes $v_h^k(x_i + hf(x_i, u))$ durch lineare Interpolation der Werte an den Knotenpunkten definiert man die folgende Rekursionsvorschrift:

$$\begin{aligned} [v_h^k]_{n+1}(x_i) &= T_h[v_h^k]_n(x_i), \quad i = 1, \ldots, N, \\ \text{mit } T_h[v_h^k]_n(x_i) &= \inf_{u \in U} \Big(\beta [v_h^k]_n(x_i + hf(x_i, u)) + hg(x_i, u) \Big). \end{aligned}$$

Der wesentliche Nachteil dieses Verfahrens liegt darin, daß die Kontraktionsrate $\beta = 1 - \rho h$ des Operators für kleine h und ρ nahe bei 1 liegt, und daher die Konvergenz sehr langsam wird.

Um diesen Mangel zu beheben, kann das Verfahren beschleunigt werden.

5.3.2 Das beschleunigte Verfahren

Die Hauptidee des beschleunigten Verfahrens liegt darin, durch geschickte Wahl des Startvektors $V_0 = ([v_h^k]_0(x_i))_{i=1,\ldots,N}$ *monotone Konvergenz* zu erhalten.

Definition 5.15 Die Teilmenge $\mathcal{V} \in \mathbb{R}^N$ ist definiert durch

$$\mathcal{V} := \Big\{ V \in \mathbb{R}^N \,\Big|\, V \leq T_h(V) \Big\},$$

wobei die Relation $\leq$ komponentenweise zu verstehen ist, d.h. $V_1 \leq V_2$ genau dann, wenn $[V_1]_i \leq [V_2]_i$ für alle $i = 1, \ldots, N$ gilt.

Lemma 5.16 Die Menge $\mathcal{V}$ ist eine konvexe, abgeschlossene Teilmenge des $\mathbb{R}^N$.

Beweis: Die Abgeschlossenheit folgt sofort aus der Definition mittels $\leq$.
Zum Beweis der Konvexität seien $V_1, V_2 \in \mathcal{V}$ gegeben, also $V_i \leq T_h(V_i)$, $i = 1, 2$. Sei nun V eine beliebige Konvexkombination von V_1 und V_2, also $V = \mu V_1 + (1-\mu)V_2$ für $\mu \in [0,1]$. Dann gilt

$$\begin{aligned} V &= \mu V_1 + (1-\mu)V_2 \;\leq\; \mu T_h(V_1) + (1-\mu)T_h(V_2) \\ &= \mu \min_{u\in U}\{\beta\Lambda(u)V_1 + hG(u)\} + (1-\mu)\min_{u\in U}\{\beta\Lambda(u)V_2 + hG(u)\} \\ &= \min_{u\in U}\{\beta\Lambda(u)\mu V_1 + \mu hG(u)\} + \min_{u\in U}\{\beta\Lambda(u)(1-\mu)V_2 + (1-\mu)hg(u)\} \\ &\leq \min_{u\in U}\{\beta\Lambda(u)(\mu V_1 + (1-\mu)V_2) + hG(u)\} \\ &= T_h(\mu V_1 + (1-\mu)V_2) \;=\; T_h(V) \end{aligned}$$

und damit die Behauptung. □

Lemma 5.17 Für beliebige Vektoren $V_1, V_2 \in \mathbb{R}^N$ gilt $V_1 \leq V_2 \Longrightarrow T_h(V_1) \leq T_h(V_2)$.

Beweis: Sei $1 \leq i \leq N$ beliebig. Dann folgt die Behauptung aus

$$\begin{aligned} [T_h(V_1)]_i &= [\min_{u\in U} \beta\Lambda(u)V_1 + hG(u)]_i \\ &= \min_{u\in U} \beta \sum_{j=1}^{N} \lambda_{ij}(u)[V_1]_j + hg(x_i,u) \\ ([V_1]_j \leq [V_2]_j \;\forall j = 1,\ldots,N) \quad &\leq \min_{u\in U} \beta \sum_{j=1}^{N} \lambda_{ij}(u)[V_2]_j + hg(x_i,u) \;=\; [T_h(V_2)]_i. \end{aligned}$$

□

Wir können nun die Hauptaussage dieses Abschnitts formulieren, die sowohl für das beschleunigte Verfahren als auch für das in folgenden Abschnitt beschriebene Koordinatenaufstiegsverfahren die notwendigen Grundlagen liefert.

Satz 5.18 Sei $V_0 \in \mathcal{V}$. Dann gilt für die Folge $V_{n+1} = T_h(V_n)$, $n \geq 0$ die Ungleichung $V_n \leq V_{n+1}$, was gleichbedeutend ist mit der Tatsache, daß V_n in $\mathcal{V}$ liegt für alle $n \geq 0$. Desweiteren ist der Fixpunkt des Rekursionsverfahrens das komponentenweise Maximum der Menge $\mathcal{V}$, d.h. der Vektor $V^* \in \mathcal{V}$ für den gilt $V^* \geq V \;\forall V \in \mathcal{V}$.

Beweis: Die erste Aussage folgt unmittelbar aus Lemma 5.17, denn es folgt
$V_0 \leq T_h(V_0) \;\Rightarrow\; V_1 = T_h(V_0) \leq T_h(T_h(V_0)) = V_2 \;\Rightarrow\; \ldots$
Für den Beweis der zweiten Aussage nehmen wir an, daß ein $V \in \mathcal{V}$ existiert mit $T_h(V) = V$ und $[V]_i < [\tilde{V}]_i$ für ein $\tilde{V} \in \mathcal{V}$ und ein $i \in \{1,\ldots,N\}$.
Wenn wir nun $\tilde{V}$ als neuen Startwert des Iterationsverfahrens wählen, so liefert uns dieses Verfahren einen Fixpunkt $V^* \in \mathcal{V}$ für den wegen der Monotonie der Iteration gilt: $[V^*]_i \geq [\tilde{V}]_i > [V]_i$, was aber einen Widerspruch zur Eindeutigkeit des Fixpunktes bedeutet. □

Das folgende Lemma zeigt uns, daß ein Startwert $V \in \mathcal{V}$ leicht zu finden ist:

Lemma 5.19 Sei $\rho > 0$ der Diskontfaktor des optimalen Steuerungsproblems und es gelte $\max_{1\le i\le N,\, u\in U} |g(x_i,u)| = M$. (Die Existenz von M folgt sofort aus (2.9).)
Dann ist $V_0 := \left(-\frac{M}{\rho}, \ldots, -\frac{M}{\rho}\right)^T \in \mathcal{V}$.

Beweis: Sei $1 \le i \le N$ beliebig. Dann gilt

$$\begin{aligned}
[T_h(V_0)]_i &= \min_{u\in U} \beta \sum_{j=1}^{N} \lambda_{ij}[V_0]_j + hg(x_i,u) \\
&= \min_{u\in U} \beta \left(-\frac{M}{\rho}\right) \sum_{j=1}^{N} \lambda_{ij} + hg(x_i,u) \\
&= \left(-\frac{M}{\rho}\right) + h \underbrace{\left(M + \min_{u\in U} g(x_i,u)\right)}_{\ge 0} \\
&\ge \left(-\frac{M}{\rho}\right) = [V_0]_i.
\end{aligned}$$

□

Mit den oben genannten Überlegungen kann nun der beschleunigte Algorithmus konstruiert werden:

Schritt 1: Nimm einen beliebigen Startwert $V_0 \in \mathcal{V}$ und berechne $\tilde{V}_1 = T_h(V_0)$.

Schritt 2: Berechne $V_1 = V_0 + \alpha(\tilde{V}_1 - V_0)$, wobei α gegeben ist durch
$\alpha = \max\{\alpha \in \mathbb{R} \,|\, V_0 + \alpha(\tilde{V}_1 - V_0) \in \mathcal{V}\}$.

Schritt 3: Ersetze V_0 durch V_1 und fahre fort mit Schritt 1.

Der Algorithmus ist in der linken Grafik von Abb. 5.1 geometrisch veranschaulicht.

Bemerkung 5.20 Dieser Algorithmus ist tatsächlich wesentlich schneller als der vorher beschriebene. Ein Vergleich der beiden Algorithmen findet sich z.B. in Falcone [10] oder Sorgenfrei [16].
Der Hauptrechenaufwand beider Verfahren liegt in den Auswertungen des Operators T_h. Beim ersten Verfahren ist diese genau gleich der Anzahl der Iterationen, beim zweiten Verfahren kommt noch eine unbestimmte Anzahl zur Ermittlung des Faktors α hinzu, z.B. für ein Bisektionsverfahren.

Bei kleinen ρ zeigt sich in dem beschleunigten Verfahren folgendes Verhalten:
Die Folge ist zwar monoton wachsend, beginnt aber in $\mathcal{V}$ gewissermaßen im Kreis zu laufen, d.h. sie läuft zwischen den „Rändern“ von $\mathcal{V}$ weite Strecken hin und her, ohne sich dabei dem Fixpunkt V^* groß zu nähern.
Diese Beobachtung gab den Anstoß für das folgende Verfahren.

5.3.3 Das Koordinatenaufstiegsverfahren

Für das Verfahren ist es nötig, die Menge $\mathcal{V}$ etwas anders darzustellen.

Lemma 5.21 Für die Menge $\mathcal{V}$ aus Definition 5.15 gilt:

$$\mathcal{V} = \left\{ V \in \mathbb{R}^N \,\middle|\, [V]_i \leq \min_{u \in U} \frac{\beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V]_j + hG_i(u)}{1 - \beta\lambda_{ii}(u)} \quad \forall i \in \{1, \ldots, N\} \right\}.$$

Beweis: Es gilt

$$\begin{aligned}
& V \in \mathcal{V} \\
\iff \quad & [V]_i \leq \min_{u \in U} \beta \sum_{j=1}^{N} \lambda_{ij}(u)[V]_j + hG_i(u) \quad \forall i \in \{1, \ldots, N\} \\
\iff \quad & [V]_i \leq \beta \sum_{j=1}^{N} \lambda_{ij}(u)[V]_j + hG_i(u) \quad \forall i \in \{1, \ldots, N\}, \forall u \in U \\
\iff \quad & [V]_i - \beta\lambda_{ii}(u)[V]_i \leq \beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V]_j + hG_i(u) \quad \forall i \in \{1, \ldots, N\}, \forall u \in U \\
\iff \quad & [V]_i \leq \frac{\beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V]_j + hG_i(u)}{1 - \beta\lambda_{ii}(u)} \quad \forall i \in \{1, \ldots, N\}, \forall u \in U \\
\iff \quad & [V]_i \leq \min_{u \in U} \frac{\beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V]_j + hG_i(u)}{1 - \beta\lambda_{ii}(u)} \quad \forall i \in \{1, \ldots, N\}
\end{aligned}$$

□

Aus diesen Überlegungen ergibt sich der *Koordinatenaufstiegsalgorithmus*:

Schritt 1: Wähle einen Startvektor $V_0 \in \mathcal{V}$.

Schritt 2: Setze $V_1 = V_0$ und bestimme nacheinander

$$[V_1]_i = \min_{u \in U} \frac{\beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V_1]_j + hG_i(u)}{1 - \beta\lambda_{ii}(u)} \quad \forall i \in \{1, \ldots, N\}.$$

Schritt 3: Setzte $V_0 = V_1$ und fahre mit Schritt 2 fort.

Die Konvergenz des Verfahrens stellt das folgende Lemma sicher.

Lemma 5.22 Es sei V_1 der Vektor, der durch Anwenden von Schritt 2 in allen Komponenten aus dem Vektor $V_0 \in \mathcal{V}$ berechnet wurde. Dann gilt

$$[V_1]_i - [V_0]_i \quad \geq \quad [T_h(V_0)]_i - [V_0]_i,$$

d.h. die Konvergenz des Koordinatenaufstiegsverfahrens folgt aus der Konvergenz des sukzessiven Approximationsverfahrens.

Beweis: Wegen $V_0 \in \mathcal{V}$ ist sicher $[V_1]_i \geq [V_0]_i \ \forall i = 1, \ldots, N$. Also folgt

$$\begin{aligned}
[V_1]_i - [V_0]_i &= \min_{u \in U} \frac{\beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V_1]_j + hG_i(u) - (1 - \beta\lambda_{ii}(u))[V_0]_i}{1 - \beta\lambda_{ii}(u)} \\
&\geq \min_{u \in U} \beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V_0]_j + hG_i(u) - (1 - \beta\lambda_{ii}(u))[V_0]_i \\
&= \min_{u \in U} \beta \sum_{j=1}^{N} \lambda_{ij}(u)[V_0]_j + hG_i(u) - [V_0]_i \ = \ [T_h(V_0)]_i - [V_0]_i
\end{aligned}$$

□

In Abbildung (5.1) sind der beschleunigte Algorithmus (links) und der Koordinatenaufstiegsalgorithmus (rechts) schematisch dargestellt. Zu beachten ist bei dieser Darstellung,

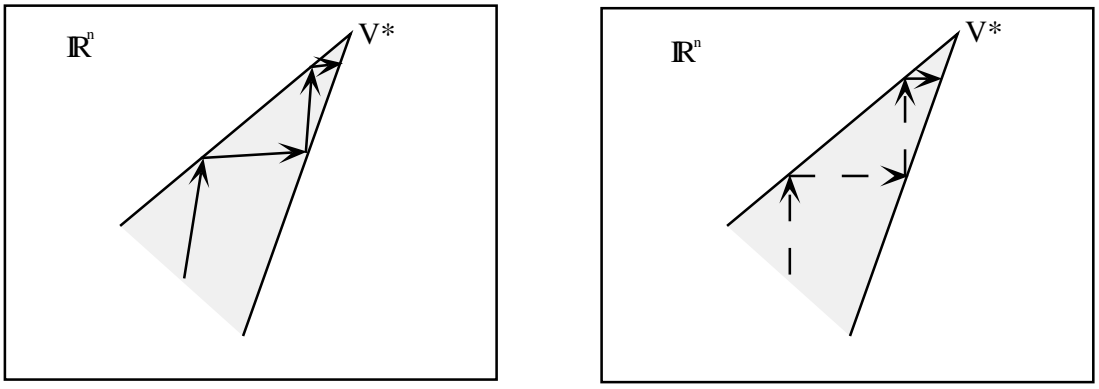

Abbildung 5.1: Geometrische Darstellung der Algorithmen, die gepunktete Menge entspricht einem Ausschnitt von $\mathcal{V}$.

daß beim beschleunigten Algorithmus für einen Iterationsschritt (entspricht einem Pfeil) mehrere Auswertungen des Operators T_h nötig sind. Demgegenüber entsprechen im Koordinatenaufstiegsalgorithmus N Pfeile (also im Schema zwei Pfeile) einer Operatorauswertung. Dieser Vorteil im Rechenaufwand kommt auch bei den numerischen Tests in Kapitel 7 deutlich heraus.

Die Konvergenzgeschwindigkeit hängt, wie aus der Grafik anschaulich gut zu erkennen ist, von den Steigungen der Hyperebenen (im Schema also der Geraden) ab, die die Menge $\mathcal{V}$ begrenzen. Je kleiner diese Steigungen sind, desto schneller konvergiert der Algorithmus.

Diese Steigungen sind gegeben durch $\frac{\beta \sum_{i \neq j} \lambda_{ij}(u)}{1 - \beta\lambda_{ii}(u)}$ und werden demnach klein, wenn entweder β oder die λ_{ij} klein werden. Beides wird jedoch nur dadurch erreicht, daß ρ und h bzw. k groß gewählt wird, was aber einen großen Diskretisierungsfehler nach sich zieht. Hohe Rechengenauigkeit und niedrige Diskontrate verlangsamen also die Konvergenz des Algorithmus unabhängig davon, daß auch die Auswertung des Operators T_h bei einer hohen Knotenanzahl aufwendiger wird.

Zum Abschluß dieses Abschnitts soll noch eine Bemerkung angefügt werden, die möglicherweise Grundlage für einen noch effizienteren Algorithmus sein könnte.

Bemerkung 5.23 Wenn der Kontrollwertebereich diskretisiert wird, d.h. nur mit einer endlichen Anzahl von Kontrollwerten aus U gerechnet wird (wie dies in der Praxis auch bei den bisher diskutierten Algorithmen üblich ist), so läßt sich das Problem in ein lineares Optimierungsproblem transformieren.
Bei Auswahl von d Kontrollwerten führt die Darstellung der Menge $\mathcal{V}$ in Lemma 5.21 auf ein Ungleichungssystem mit N Unbekannten und $N \cdot d$ Ungleichungen. Das Problem lautet dann also:
Maximiere $\sum [V]_i$ für $V \in \mathbb{R}^N$ mit $AV \leq b$. Hierbei sind die Koeffizienten von A und b durch die Ungleichungen in Lemma 5.21 gegeben.

5.4 Die Konstruktion ε-optimaler Kontrollen

In diesem Abschnitt werden wir zunächst ein Verfahren herleiten, das ausgehend von der zeit- und ortsdiskretisierten Wertefunktion v_h^k ein ε-optimales Zustandsfeedback für das diskretisierte optimale Steuerungsproblem liefert.

Für die so konstruierte Kontrolle werden wir dann den Diskretisierungsfehler abschätzen und zeigen, daß diese Kontrolle auch für das ursprüngliche diskontierte optimale Steuerungsproblem ε-optimal ist.

5.4.1 Das ε-optimale Zustandfeedback

Wir gehen zunächst von der zeitdiskretisierten Wertefunktion v_h aus. Wenn wir annehmen, daß v_h mindestens unterhalbstetig ist, wird für jedes feste $x \in \overline{\Omega}^k$ das Supremum in (5.2) in einem Wert $\tilde{u}_h(x) \in U$ angenommen. Davon ausgehend können wir definieren:

$$x_0^* = x, \quad x_{j+1}^* = \Psi(x_j^* + hf(x_j^*, \tilde{u}_h(x_j^*))), \quad j = 0, 1, 2, \ldots \tag{5.25}$$

und definieren uns ein $u_h^* \in \mathcal{U}_h$ durch

$$u_h^*(t) = \tilde{u}_h(x_j^*), \quad t \in [jh, (j+1)h) \tag{5.26}$$

Satz 5.24 Sei $h \in [0, \frac{1}{\rho})$. Ist v_h unterhalbstetig und U kompakt, so ist die stückweise konstante Kontrolle u_h^* definiert durch (5.25) und (5.26) optimal für (5.5), d.h. es gilt

$$v_h(x) = J_h(x, u_h^*(\cdot)) \quad \forall x \in \mathbb{R}^n.$$

Beweis: Nach Definition von $x \mapsto \tilde{u}_h(x)$ folgt aus (5.2), daß für jedes $p \in \mathbb{N}$ gilt:

$$\beta^p \Big(v_h(x_p^*) - \beta v_h(x_{p+1}^*) - hg(x_p^*, \tilde{u}_h(x_p^*)) \Big) = 0.$$

Durch Aufaddieren ergibt sich so

$$v_h(x) = \beta^{p+1} v_h(x^*_{p+1}) + h \sum_{j=0}^{p} \beta^j g(x^*_j, \tilde{u}_h(x^*_j))$$

und damit für $p \to \infty$ die Behauptung. □

In der Praxis haben wir die Funktion v_h jedoch nicht zur Verfügung; wir müssen ausgehen von dem Vektor $V^* \in \mathbb{R}^n$, in dem die Werte von v_h^k in den Knotenpunkten der Triangulation gespeichert sind. Für jeden Knotenpunkt $x_i \in \overline{\Omega^k}$ existiert aufgrund des Approximationsverfahrens mindestens ein Kontrollwert $\tilde{u}_i \in U$, so daß in (5.17) Gleichheit gilt. Dieser muß aber nicht eindeutig sein, weshalb wir die folgende lexikographische Ordnung auf dem $\mathbb{R}^m$ einführen:

Definition 5.25 Es seien $x, y, \in \mathbb{R}^m$. Es gelte $x \prec y$ genau dann, wenn ein $q \in \{1, \ldots, m\}$ existiert mit $x_q < y_q$ und $x_i \leq y_i \ \forall 1 \leq i \leq q$.

Mit dieser Ordnungsrelation können wir nun einen eindeutigen Kontrollwert auswählen, der Gleichheit in (5.17) liefert. Dies erweitern wir nun auf ganz $\overline{\Omega^k}$.

Definition 5.26 Zu gegebenen ρ, h, $k > 0$ definieren wir zu jedem $x \in \overline{\Omega^k}$ einen Kontrollwert $\tilde{u}^k_{x,h} \in U$ als den bezüglich der lexikographischen Ordnung kleinsten Kontrollwert, für den gilt

$$\beta v_h^k(\Psi(x + hf(x, \tilde{u}^k_{x,h}))) + hg(x, \tilde{u}^k_{x,h}) = \min_{u \in U} \Big(\beta v_h^k(\Psi(x + hf(x,u))) + hg(x,u) \Big).$$

Die stückweise konstante Kontrolle $u^k_{x,h}(\cdot)$ definieren wir nun durch

$$u^k_{x,h}(t) = \tilde{u}^k_{x_j,h}, \quad t \in [jh, (j+1)h),$$

wobei die diskretisierte Trajektorie x_i gegeben ist durch

$$x_0 = x, \quad x_{j+1} = \Psi(x_j + hf(x_j, \tilde{u}^k_{x_j,h})), \quad j = 0, 1, 2, \ldots$$

Mit dieser Definition können wir nun folgendes Ergebnis formulieren:

Satz 5.27 Es seien die Voraussetzungen (2.4) – (2.9) an f und g erfüllt und $u^k_{x,h}(\cdot)$ sei die in Definition 5.26 definierte Kontrolle.
Dann gibt es für jedes $\varepsilon > 0$ ein $K = K(h) > 0$, so daß für alle $k \leq K$ gilt:

$$|J_h(x, u^k_{x,h}(\cdot)) - v_h(x)| \leq \varepsilon \quad \forall x \in \mathbb{R}^n.$$

Beweis: Nach der Konstruktion der Kontrolle ist $u^k_{x,h}|_{[ih,(i+1)h]} \equiv u^{k,i}_{x,h}$, wobei $u^{k,i}_{x,h} \in U$ derjenige Wert ist, so daß $hg(x_i, u^{k,i}_{x,h}) + \beta v^k_h(\Psi(x_i + hf(x_i, u^{k,i}_{x,h})))$ minimal wird. Hierbei ist die Folge $(x_i)_{i\in\mathbb{N}}$ definiert durch: $x_0 := x,\ \ x_{i+1} := \Psi(x_i + hf(x_i, u^{k,i}_{x,h}))$. Also gilt unter Berücksichtigung der gleichmäßigen Konvergenz von $v^k_h(\cdot)$ gegen $v_h(\cdot)$ für beliebiges $\varepsilon > 0$ und genügend kleines $\tilde{K}(h) > 0$ für alle $k \le \tilde{K}(h)$:

$$\begin{aligned} hg(x_i, u^{k,i}_{x,h}) + \beta v^k_h(\Psi(x_i + hf(x_i, u^{k,i}_{x,h}))) &\ge hg(x_i, u^{k,i}_{x,h}) + \beta v_h(\Psi(x_i + hf(x_i, u^{k,i}_{x,h}))) - \frac{\varepsilon}{2} \\ &\ge v_h(x_i) - \frac{\varepsilon}{2} \ \ge\ v^k_h(x_i) - \varepsilon \end{aligned}$$

und andererseits

$$\begin{aligned} hg(x_i, u^{k,i}_{x,h}) + \beta v^k_h(\Psi(x_i + hf(x_i, u^{k,i}_{x,h}))) &\le hg(x_i, u^{0,i}_{x,h}) + \beta v^k_h(\Psi(x_i + hf(x_i, u^{0,i}_{x,h}))) \\ &\le hg(x_i, u^{0,i}_{x,h}) + \beta v_h(\Psi(x_i + hf(x_i, u^{0,i}_{x,h}))) + \frac{\varepsilon}{2} \\ &= v_h(x_i) + \frac{\varepsilon}{2} \ \le\ v^k_h(x_i) + \varepsilon, \end{aligned}$$

wobei $u^{0,i}_{x,h} \in U$ der Wert ist, bei dem $hg(x_i, u^{0,i}_{x,h}) + \beta v_h(\Psi(x_i + hf(x_i, u^{0,i}_{x,h})))$ minimal wird. Zusammen ist also

$$|hg(x_i, u^{k,i}_{x,h}) + \beta v^k_h(\Psi(x_i + hf(x_i, u^{k,i}_{x,h}))) \ - \ v^k_h(x)| \ \le\ \varepsilon \quad \forall x \in \overline{\Omega^k} \tag{5.27}$$

Per Induktion ergibt sich damit, daß für alle $\varepsilon > 0,\ \ p \in \mathbb{N},\ \ h > 0$ ein $k > 0$ existiert, so daß gilt:

$$|h\sum_{j=0}^{p} \beta^j g(x_j, u^{k,j}_{x,h}) + \beta^{p+1} v^h_k(x_{p+1}) \quad - \quad v^k_h(x)| \ \le\ \varepsilon \quad \forall x \in \overline{\Omega^k} \tag{5.28}$$

Da $\beta < 1$ ist für alle $h > 0$ und g sowie v^k_h beschränkt sind auf ganz $\overline{\Omega^k}$, können wir nun zu vorgegebenem $\varepsilon > 0$ ein $p_h \in \mathbb{N}$ finden, so daß

$$|h\sum_{j=0}^{\infty} \beta^j g(x, u^{k,j}_{x,h}) - h\sum_{j=0}^{p_h} \beta^j g(x, u^{k,j}_{x,h}) - \beta^{p_h+1} v^k_h(x)| < \frac{\varepsilon}{2} \quad \forall x \in \overline{\Omega^k}, u \in \mathcal{U}_h. \tag{5.29}$$

Zu diesen h, p_h können wir nun nach den obigen Überlegungen $K(h) > 0$ finden, so daß (5.28) mit $k < K(h)$ ebenfalls für $\frac{\varepsilon}{2}$ erfüllt ist. Durch Anwendung der Dreiecksungleichung ergibt sich so aus (5.28) und (5.29) die Behauptung. □

Aus der Definiton 5.26 können wir also unseren Algorithmus zur Bestimmung des Zustandsfeedbacks ableiten:

Schritt 1: Setze $x_0 = x$.

Schritt 2: Bestimme den lexikographisch kleinsten Kontrollwert $\tilde{u}^k_{x_0,h} \in U$, so daß $\beta v^k_h(\Psi(x + hf(x, \tilde{u}^k_{x_0,h}))) + hg(x, \tilde{u}^k_{x_0,h})$ minimal wird.

Schritt 3: Setze $u^k_{x,h}(t) = \tilde{u}^k_{x_0,h}$ für alle $t \in [nh, (n+1)h]$.

Schritt 4: Setze $x_0 = \Psi(x_0 + hf(x, \tilde{u}^k_{x_0,h}))$ und fahre mit Schritt 2 fort.

5.4.2 Die Kontrolle für das ursprüngliche optimale Steuerungsproblem

Zum Abschluß dieses Kapitels wollen wir nun zeigen, daß die durch Definition 5.26 gegebene Kontrolle auch für das ursprüngliche diskontierte optimale Steuerungsproblem ε-optimal ist. Hierfür müssen wir nur die bisherigen Resultate zusammenfassen.

Satz 5.28 Sei $\rho > 0$ gegeben und $u^k_{x,h}(\cdot)$ die durch Definition 5.26 gegebene stückweise konstante Kontrolle. Dann gibt es für jedes $\varepsilon > 0$ ein $H > 0, K(h) > 0$, so daß $|J_\rho(x, u^k_{x,h}(\cdot)) - v_\rho(x)| < \varepsilon \quad \forall x \in \overline{\Omega^k},\ h < H,\ k < K(h)$. D.h. $J_\rho(x, u^k_{x,h}(\cdot))$ konvergiert gleichmäßig gegen den optimalen Wert.

Beweis: Es ist

$$\begin{aligned} |J_\rho(x, u^k_{x,h}(\cdot)) - v_\rho(x)| \quad \leq \quad & |J_\rho(x, u^k_{x,h}(\cdot)) - J_h(x, u^k_{x,h}(\cdot))| \\ & + |J_h(x, u^k_{x,h}(\cdot)) - v_h(x)| \\ & + |v_h(x) - v_\rho(x)|. \end{aligned}$$

Für diese Terme haben wir bereits Abschätzungen durch die Sätze 5.7, 5.27 und 5.8.

Zu vorgegebenem $\varepsilon > 0$ können wir also H und $K(h)$ so wählen, daß alle drei Terme kleiner als $\frac{\varepsilon}{3}$ werden, womit die Behauptung erfüllt ist. □

Kapitel 6

Bilineare Kontrollsysteme und Stabilität

Die Systeme, für die wir in diesem Kapitel einen Stabilisierungsalgorithmus herleiten wollen, sind *bilineare Kontrollsysteme* im $\mathbb{R}^n$, d.h. Systeme der Form

$$\dot{x}(t) = \Big(A_0 + \sum_{i=1}^{d} u_i(t) A_i\Big) x(t), \quad x(0) = x_0 \in \mathbb{R}^n \setminus \{0\} \tag{6.1}$$

mit $A_j \in gl(n, \mathbb{R})$, $j = 0, \ldots, d$, $u(\cdot) \in \mathcal{U} := \{u : \mathbb{R} \to U,\ u \text{ meßbar}\}$ wobei der Kontrollwertebereich $U \subset \mathbb{R}^d$ kompakt mit nichtleerem Inneren ist.

Wir nehmen an, daß die Lösungstrajektorie zu jedem Startwert $x_0 \in \mathbb{R}^n \setminus \{0\}$ und jeder Kontrollfunktion $u \in \mathcal{U}$ eindeutig existiert, sie sei bezeichnet mit $x(t, x_0, u(\cdot))$.

Im ersten Abschnitt dieses Kapitels werden wir die *Lyapunov-Exponenten* betrachten, die ein Maß für die Stabilität der Lösungen von (6.1) darstellen. Im zweiten Abschnitt werden einige Zusammenhänge von Kontrollmengen im Projektiven Raum $\mathbb{P}^{n-1}$ und den Lyapunov-Exponenten zitiert. Im letzten Abschnitt wird dann mithilfe der Lyapunov-Exponenten und den Hilfsmitteln aus den vorhergehenden Kapiteln ein Stabilisierungsalgorithmus hergeleitet.

6.1 Lyapunov-Exponenten

Wir wollen nun das exponentielle Wachstum der Lösungen von (6.1) charakterisieren. Dazu definieren wir

Definition 6.1 Der Lyapunov-Exponent einer Lösung von (6.1) ist definiert durch

$$\lambda(x_0, u(\cdot)) := \limsup_{t \to \infty} \frac{1}{t} \ln \|x(t, x_0, u(\cdot))\|. \tag{6.2}$$

Der minimale Lyapunov-Exponent zu $x_0 \in \mathbb{R}^n \setminus \{0\}$ ist definiert als

$$\lambda^*(x_0) := \inf_{u(\cdot) \in \mathcal{U}} \lambda(x_0, u(\cdot)) \tag{6.3}$$

und der minimale Lyapunov-Exponent des Kontrollsystems als

$$\kappa^* := \inf_{x_0 \neq 0} \lambda^*(x_0). \tag{6.4}$$

Bemerkung 6.2 Der Lyapunov-Exponent ist ein Maß für das exponentielle Wachstum von Trajektorien. Trajektorien, deren Lyapunov-Exponent negativ ist, laufen asymptotisch nach Null, sind also stabil.

Eine einfache Überlegung erlaubt es uns, den Bereich der Anfangswerte, die wir betrachten, einzuschränken.

Lemma 6.3 Es sei ein bilineares Kontrollsystem der Form (6.1) gegeben. Dann gilt

$$\lambda(x_0, u(\cdot)) = \lambda(\alpha x_0, u(\cdot)) \quad \forall x_0 \in \mathbb{R}^n \setminus \{0\}, \;\; \alpha \in \mathbb{R} \setminus \{0\}, \;\; u \in \mathcal{U}.$$

Beweis: Wegen der Linearität gilt $x(t, \alpha x_0, u(\cdot)) = \alpha x(t, x_0, u(\cdot))$ für alle $x_0 \neq 0$, $\alpha \neq 0$. Also folgt

$$\begin{aligned} \limsup_{t\to\infty} \frac{1}{t} \ln \|x(t, \alpha x_0, u(\cdot))\| &= \limsup_{t\to\infty} \frac{1}{t} (\ln|\alpha| + \ln \|x(t, x_0, u(\cdot))\|) \\ &= \limsup_{t\to\infty} \frac{1}{t} \ln \|x(t, x_0, u(\cdot))\| \end{aligned}$$

und damit die Behauptung. □

Es genügt also, Anfangswerte s_0 aus $\mathbb{P}^{n-1}$, dem Projektiven Raum, zu betrachten, da dort alle Punkte auf einer Geraden, die durch den Nullpunkt läuft, identifiziert werden. Wir können das System also zur Diskussion der Lyapunov-Exponenten mittels $s = x/\|x\|$ auf die Sphäre projizieren und erhalten dann durch Identifikation gegenüberliegender Punkte ein System auf dem Projektiven Raum. Das folgende Lemma zeigt, wie sich das System auf $\mathbb{P}^{n-1}$ beschreiben läßt.

Lemma 6.4 Es sei ein bilineares Kontrollsystem (6.1) gegeben. Für das auf $\mathbb{P}^{n-1}$ projizierte System gilt

$$\dot{s}(t) = h_0(s(t)) + \sum_{i=1}^{d} u_i(t) h_i(s(t)) \tag{6.5}$$

mit

$$h_i(s) = [A_i - s^t A_i s \cdot \mathrm{Id}]s \quad \forall i = 0, \ldots, d.$$

Für den Lyapunov-Exponenten (6.2) gilt mit $s_0 = x_0/\|x_0\|$

$$\lambda(x_0, u(\cdot)) = \lambda(s_0, u(\cdot)) = \limsup_{t\to\infty} \frac{1}{t} \int_0^t q(s(\tau, s_0, u(\cdot)), u(\tau)) d\tau$$

mit

$$q(s, u) = s^t \Big(A_0 + \sum_{i=0}^{d} u_i A_i \Big) s.$$

Beweis: Für die projizierte Trajektorie $s(t)$ gilt

$$\begin{aligned}
\dot{s}(t) &= \frac{d}{dt}\frac{x(t)}{\|x(t)\|} = \frac{1}{\|x(t)\|^2}\Big(\frac{d}{dt}x(t)\|x(t)\| - x(t)\frac{d}{dt}\|x(t)\|\Big) \\
&= \frac{1}{\|x(t)\|}\dot{x}(t) - \Big(\frac{1}{\|x(t)\|^3}\langle x(t), \dot{x}(t)\rangle \cdot \mathrm{Id}\Big)x(t) \\
&= \Big(A_0 + \sum_{i=1}^{d} u_i A_i\Big)s(t) - \langle s(t), \Big(A_0 + \sum_{i=1}^{d} u_i A_i\Big)s(t)\rangle \cdot \mathrm{Id} \cdot s(t) \\
&= [A_0 - s(t)^t A_0 s(t) \cdot \mathrm{Id}]s(t) + \sum_{i=1}^{d}[u_i A_i - s(t)^t (u_i A_i)s(t) \cdot \mathrm{Id}]s(t)
\end{aligned}$$

und damit die erste Behauptung.
Für die zweite Behauptung betrachten wir

$$\begin{aligned}
\frac{d}{dt}2\ln\|x(t)\| &= \frac{d}{dt}\ln\|x(t)\|^2 = \frac{d}{dt}\ln\langle x(t), x(t)\rangle \\
&= \frac{2\langle x(t), \dot{x}(t)\rangle}{\|x(t)\|^2} = 2s(t)^t\Big(A_0 + \sum_{i=0}^{d} u_i(t)A_i\Big)s(t) = 2q(s(t), u(t)).
\end{aligned}$$

Also folgt

$$\|x(t)\| = \|x_0\| \exp \int_0^t q(s(\tau), u(\tau))d\tau$$

und damit die Behauptung. □

Bemerkung 6.5 Das projizierte System erfüllt als optimales Steuerungsproblem die Bedingungen (3.1) – (3.8) aus Kapitel 3.

Im Folgenden bezeichnen wir die Lösungstrajektorie von (6.5) zum Anfangswert $s_0 \in \mathbb{P}^{n-1}$ wieder mit $\varphi(t, s_0, u(\cdot))$ und nehmen an, daß die **Annahme (3.13)** für das projizierte System erfüllt ist. Wenn wir uns dann Korollar 3.23 ansehen, können wir daraus für den Lyapunov-Exponenten die gleiche Behauptung aufstellen.

Lemma 6.6 Sei D eine Kontrollmenge des nach Lemma 6.4 gegebenen Kontrollsystems auf $\mathbb{P}^{n-1}$ und $A(D)$ deren Einzugsbereich nach Defnition 3.21. Dann gilt:

$$\inf_{s\in A(D)} \lambda^*(s) \leq \inf_{s\in D} \lambda^*(s).$$

Beweis: Folgt sofort aus Korollar 3.23. □

Es gibt also einen Zusammenhang zwischen dem Lyapunov-Exponenten und den Kontrollmengen des Systems in $\mathbb{P}^{n-1}$.

6.2 Kontrollmengen im Projektiven Raum

Im diesem Abschnitt werden wir einige Eigenschaften über Kontrollmengen im Projektiven Raum sowie über Lyapunov-Exponenten in diesen Kontrollmengen zitieren. Es handelt sich um die Theoreme 1 und 3 aus Colonius, Kliemann [6]. Die Betrachtungen sind zum einen nützlich für das Verständnis der numerischen Beispiele, zum anderen aber auch für die numerische Berechnung, da die Konvergenz der Wertefunktion v_ρ für ρ gegen 0 von den Kontrollmengen und ihren Einzugsbereichen abhängt.

Wir gehen nun aus von einem gemäß Lemma 6.4 projizierten Kontrollsystem. Wir nehmen an, daß das projizierte System die Annahme (3.13) erfüllt, wobei L hier genau die Lie-Algebra ist, die von den Vektorfeldern $h_i(\cdot),\ i = 0, \ldots, d$ erzeugt wird (vgl. Isidori [12], Theorem 2.7).

Wenn wir stückweise konstante Kontrollen betrachten, so können wir die zu System (6.1) gehörige *Systemhalbgruppe* definieren als

$$\mathcal{S} := \{g = \exp(t_n A(u_n)) \ldots \exp(t_1 A(u_1)) \,|\, t_j > 0,\ \ u_j \in U,\ \ j = 1, \ldots, n,\ \ n \in \mathbb{N}\}.$$

Die Annahme (3.13) impliziert dann, daß das Innere von $\mathcal{S}$ in der System(Lie-)Gruppe $\mathcal{G}$ (definiert wie $\mathcal{S}$ aber mit $t_j \in \mathbb{R}$) nichtleer ist. Mit dieser Notation läßt sich für die Kontrollmengen von (6.5) in $\mathbb{P}^{n-1}$ folgender Satz beweisen:

Satz 6.7 Sei Annahme (3.13) erfüllt. Dann gilt

(i) Es gibt k Kontrollmengen $D_1, \ldots, D_k$, $1 \leq k \leq n$ mit nichtleerem Inneren in $\mathbb{P}^{n-1}$. Diese nennen wir die *Hauptkontrollmengen*.

(ii) Das Innere der Hauptkontrollmengen besteht aus den Zusammenhangskomponenten von $\{\mathbb{P}E(\lambda) \,|\, \lambda \in spec(g),\ \ g \in \text{int}\mathcal{S}\}$, wobei $\mathbb{P}E(\lambda)$ die Projektion nach $\mathbb{P}^{n-1}$ der verallgemeinerten Eigenräume von g zum Eigenwert λ von g bezeichnet.

(iii) Die Hauptkontrollmengen sind linear geordnet durch

$$D_i \prec D_j \iff \exists x_i \in D_i,\, x_j \in D_j,\, g \in \mathcal{S}_t \text{ mit } gx_i = x_j.$$

Wir numerieren die Hauptkontrollmengen mittels dieser Ordnung $D_1 \prec D_2 \ldots \prec D_k$.

(iv) Die Kontrollmenge $C := D_k$ ist abgeschlossen und als einzige invariant, die Kontrollmenge $C^* := D_1$ ist offen. Alle anderen Hauptkontrollmengen sind weder offen noch abgeschlossen.

Bemerkung 6.8 Die lineare Ordnung impliziert, daß die minimalen Lyapunov-Exponenten in den Hauptkontrollmengen ebenfalls linear geordnet sind. Eigenschaft (ii) impliziert außerdem, daß jeder Punkt in $\mathbb{P}^{n-1}$ im Einzugsbereich mindestens einer Hauptkontrollmenge liegt. Die Menge aller Punkte aus $\mathbb{P}^{n-1}$, die negativen Lyapunov-Exponenten besitzen, läßt sich also als Vereinigung von Hauptkontrollmengen und ihren Einzugsbereichen darstellen.

Die Tatsache, daß wir uns mit der Systemhalbgruppe auf stückweise konstante Kontrollen eingeschränkt haben, führt zu folgender Definition:

Definition 6.9 Sei D eine Hauptkontrollmenge von (6.5). Das *Floquet-Spektrum von (6.1) über D* ist definiert als

$$\Sigma_{Fl}(D) := \left\{ \lambda(p,u(\cdot)) \,|\, (p,u(\cdot)) \in \mathrm{int}D \times \mathcal{U}, \begin{array}{l} u(\cdot) \text{ ist stückweise konstant und} \\ \tau-\text{periodisch mit } \varphi(\tau,p,u(\cdot)) = p \end{array} \right\}$$

und das *Floquet-Spektrum* Σ_{Fl} von (6.1) ist definiert als

$$\Sigma_{Fl} := \bigcup_{i=1}^{k} \Sigma_{Fl}(D_i).$$

Auf ähnliche Weise ist das *Lyapunov-Spektrum von (6.1) über $\overline{D}$* erklärt als

$$\Sigma_{Ly}(\overline{D}) := \{\lambda(p,u(\cdot)) \,|\, \varphi(t,p,u(\cdot)) \in \overline{D} \text{ für alle } t \geq T \text{ für ein } T \geq 0\}$$

und das *Lyapunov-Spektrum* als

$$\Sigma_{Ly} := \{\lambda(p,u(\cdot)) \,|\, p \in \mathbb{P}^{n-1}, \;\; u(\cdot) \in \mathcal{U}\}.$$

Sicherlich gilt $\Sigma_{Fl} \subseteq \Sigma_{Ly}(\overline{D}) \subseteq \Sigma_{Ly}$. Für die umgekehrte Richtung ist zunächst nicht klar, wann Gleichheit gilt. Wir können aber ein Resultat formulieren, wenn wir System (6.1) in eine Familie von Systemen einbetten, indem wir die Größe des Kontrollwertebereichs mit einem Parameter σ variieren.

Sei dazu $U^\sigma := \sigma U, \;\; \sigma \geq 0$. Dann hängen alle Größen in Definition 6.9 von σ ab. Die Annahme (3.13) gilt für $\sigma > 0$ genau dann, wenn sie für $\sigma = 1$ gilt.

Seien $\lambda_1, \ldots, \lambda_k$ die verschiedenen Realteile der Eigenwerte der Matrix A_0 und $E(\lambda_i)$, $1 \leq i \leq k \leq n$, die zugehörigen Summen von verallgemeinerten Eigenräumen.
Für $i = 1, \ldots, k$ betrachten wir die folgende Abbildung von $[0,\infty)$ mit Werten in der Menge der kompakten Teilmengen von P^{n-1} versehen mit der Hausdorff-Metrik.

$$\begin{array}{l} \sigma \mapsto \overline{D_i(\sigma)}, \\ \text{wobei } D_i(\sigma) \text{ eine Hauptkontrollmenge ist mit} \\ \mathbb{P}E(\lambda_i) \subset \mathrm{int}D_i(\sigma) \text{ für } \sigma > 0 \text{ und } D_i(0) = \mathbb{P}E(\lambda_i). \end{array} \tag{6.6}$$

Die Anzahl der Hauptkontrollmengen $k(\sigma)$ nimmt für wachsendes σ ab, deshalb können einige der $D_i(\sigma)$, $i = 1, \ldots, k$ in (6.6) zusammenfallen. Wir bezeichnen deshalb die unterschiedlichen $D_i(\sigma)$ mit D_j^σ, $j = 1, \ldots, k(\sigma)$. Der folgende Satz zeigt zunächst die Wohldefiniertheit von (6.6), d.h. die Existenz der $D_i(\sigma)$. Außerdem wird eine Aussage über Zusammenhänge zwischen den oben definierten Spektren gemacht.

Satz 6.10 Es gelte die Annahme (3.13) und außerdem die folgende Innere-Punkt-Bedingung für $\sigma' > 0$:
Für alle $0 < \sigma < \sigma'$ und alle $(p, u(\cdot)) \in \mathbb{P}^{n-1} \times \mathcal{U}^\sigma$ existieren $T > 0$ und $S > 0$ mit

$$\varphi(T, p, u(\cdot)) \in \mathrm{int}\mathcal{O}^{\sigma',+}_{\leq T+S}(p),$$

wobei $\mathcal{O}^{\sigma',+}_{\leq T+S}(p)$ der positive Orbit bis zur Zeit $T+S$ bezüglich der Kontrollen aus $\mathcal{U}^{\sigma'}$ ist. Dann gelten die folgenden Aussagen:

(i) Für $i = 1, \ldots, k$ und $\sigma \geq 0$ gibt es eine Hauptkontrollmenge $D_i(\sigma)$, die die Bedingung (6.6) erfüllt und es gilt $\overline{D_i(\sigma)} \subset \mathrm{int}D_i(\sigma')$ für $0 < \sigma < \sigma'$.

(ii) Die Abbildungen $\sigma \mapsto cl\Sigma_{Fl}(D_i(\sigma))$ sind monoton wachsend und stetig auf $[0, \infty)$ bis auf abzählbar viele Stellen. Die Bilder dieser Abbildungen sind Intervalle. An den Stellen, in denen die Abbildung stetig ist, gilt $cl\Sigma_{Fl}(D_i) = \Sigma_{Ly}(cl(D_i))$, $i = 1, \ldots, k$, und $cl\Sigma_{Fl} = \Sigma_{Ly}$.

(iii) Sei σ eine Stetigkeitsstelle, dann gibt es für jedes $(s, u(\cdot)) \in \mathbb{P}^{n-1} \times \mathcal{U}^\sigma$ eine Hauptkontrollmenge D mit $\pi_{\mathbb{P}}\omega(s, u(\cdot)) := \{q \in \mathbb{P}^{n-1} \mid \exists t_k \to \infty \text{ mit } \varphi(t_k, s, u) \to q\} \subset \overline{D}$ und $\lambda(s, u(\cdot)) \in \Sigma_{Ly}(\overline{D})$.

(iv) An jeder Stetigkeitsstelle σ definiere
$E^\sigma_j(u(\cdot)) := \{x \in \mathbb{R}^n \mid x \neq 0 \text{ impliziert } \varphi(t, x, u(\cdot)) \in D^\sigma_j \ \ \forall t \in \mathbb{R}\}$
für $j = 1, \ldots, k(\sigma)$, $u(\cdot) \in \mathcal{U}$
Dann sind diese $E^\sigma_j(u(\cdot))$ Unterräume mit Dimension unabhängig von $u(\cdot) \in \mathcal{U}$ und es gilt $\mathbb{R}^n = E^\sigma_1(u(\cdot)) \oplus \ldots \oplus E^\sigma_{k(\sigma)}(u(\cdot))$.

Bemerkung 6.11 Die Spektralintervalle sind durch $\Sigma_{Fl}(D^\sigma_i) \prec \Sigma_{Fl}(D^\sigma_j)$ für $i < j$ im folgenden Sinne geordnet:
$\inf \Sigma_{Fl}(D^\sigma_i) \leq \inf \Sigma_{Fl}(D^\sigma_j)$ und $\sup \Sigma_{Fl}(D^\sigma_i) \leq \sup \Sigma_{Fl}(D^\sigma_j)$.
Die Spektralintervalle können sich dabei aber überlappen.

6.3 Der Stabilisierungsalgorithmus

In diesem Abschnitt werden wir die Resultate aus den vorangegangenen Kapiteln und Abschnitten verwenden, um einen Stabilisierungsalgorithmus für bilineare Kontrollsysteme zu entwickeln. Die Idee des Algorithmus ist dabei wie folgt:
Wenn wir eine Kontrolle konstruieren, so daß der Lyapunov-Exponent entlang der zugehörigen Trajektorie stets negativ ist, so ist diese Trajektorie exponentiell stabil.

Indem wir das optimale Steuerungsproblem aus Lemma 6.4 mit dem Algorithmus aus Abschnitt 5.3.3 zu kleinem $\rho > 0$ numerisch lösen, erhalten wir nach Satz 3.18 und Satz 3.24 mit der Wertefunktion eine Approximation der minimalen Lyapunov-Exponenten in jedem Punkt $s \in \mathbb{P}^{n-1}$, insbesondere gibt es dort Bereiche, in denen die Wertefunktion negativ ist. Mit dem Algorithmus aus Abschnitt 5.4.1 können wir dann nach Satz 5.28 zu jedem Anfangswert ε-optimale Kontrollen konstruieren.

Wir wollen nun diese Kontrollen dafür verwenden, eine neue Kontrolle entlang der exakten Trajektorie zu konstruieren, deren Endstücke das optimale Steuerungsproblem für jeden Punkt auf der Trajektorie approximativ optimal lösen, und damit die Voraussetzungen des Approximationssatzes 4.3 erfüllen.

Definition 6.12 Zu jedem $x \in \overline{\Omega^k}$ sei mit $u_x(\cdot) \in \mathcal{U}$ eine Kontrolle gegeben. Sei nun $(\tau_i)_{i\in\mathbb{N}}$ eine Folge in $\mathbb{R}$ mit $\tau_1 = 0$, $\tau_{i+1} > \tau_i$ und $\tau_{i+1} - \tau_i \in [\frac{a}{\rho}, \frac{b}{\rho}]$ $\forall i \in \mathbb{N}$ und $a, b \in \mathbb{R}$, $a \leq b$. Dann ist die Kontrolle $\bar{u}_x(\cdot) \in \mathcal{U}$ definiert durch:

$$\bar{u}_x|_{[\tau_i,\tau_{i+1})} \equiv u_{\varphi(x,\tau_i,\bar{u}_x(\cdot))}|_{[0,\tau_{i+1}-\tau_i)} \quad \forall i \in \mathbb{N}$$

Lemma 6.13 Sei zu jedem $x \in \overline{\Omega^k}$ eine Kontrolle $u_x(\cdot) \in \mathcal{U}$ gegeben, so daß $|J_\rho(x, u_x(\cdot)) - v_\rho(x)| < \varepsilon$. Dann gilt für $\bar{u}_x(\cdot) \in \mathcal{U}$ aus Definition 6.12:

$$J_\rho(\varphi(\sigma, x, \bar{u}_x(\cdot)), \bar{u}_x(\sigma + \cdot)) \leq v_\rho(\varphi(\sigma, x, \bar{u}_x(\cdot))) + e^b \frac{1 + e^{-a}}{1 - e^{-a}} \cdot \varepsilon \quad \forall \sigma \geq 0.$$

Beweis: Nach Wahl der τ_i ist $e^{-b} \leq e^{-\rho(\tau_{i+1}-\tau_i)} \leq e^{-a}$. Damit ergibt sich mit $x_{\tau_i} := \varphi(x, \tau_i, u_x(\cdot))$:

$$\begin{aligned}
&\int_0^{\tau_{i+1}-\tau_i} e^{-\rho t} g(\varphi(t, x_{\tau_i}, u_{x_{\tau_i}}(\cdot)), u_{x_{\tau_i}}(t)) dt \\
&+ e^{-\rho(\tau_{i+1}-\tau_i)} \int_0^{\infty} e^{-\rho t} g(\varphi(t, x_{\tau_{i+1}}, u_{x_{\tau_{i+1}}}(\cdot)), u_{x_{\tau_{i+1}}}(t)) dt \\
\leq &\int_0^{\tau_{i+1}-\tau_i} e^{-\rho t} g(\varphi(t, x_{\tau_i}, u_{x_{\tau_i}}(\cdot)), u_{x_{\tau_i}}(t)) dt + e^{-\rho(\tau_{i+1}-\tau_i)} v_\rho(x_{\tau_{i+1}}) + e^{-\rho(\tau_{i+1}-\tau_i)} \cdot \varepsilon \\
\leq &\int_0^{\infty} e^{-\rho t} g(\varphi(t, x_{\tau_i}, u_{x_{\tau_i}}(\cdot)), u_{x_{\tau_i}}(t)) dt + e^{-\rho(\tau_{i+1}-\tau_i)} \cdot \varepsilon \\
\leq &\, v_\rho(x_{\tau_i}) + \varepsilon + e^{-\rho(\tau_{i+1}-\tau_i)} \cdot \varepsilon \;\leq\; v_\rho(x_{\tau_i}) + (1 + e^{-a})\varepsilon
\end{aligned}$$

Per Induktion ergibt sich so:

$$\begin{aligned}
J_\rho(x, \bar{u}_x(\cdot)) &= \sum_{i=0}^{\infty} \int_{\tau_i}^{\tau_{i+1}} e^{-\rho t} g(\varphi(t, \varphi(\tau_i, x, \bar{u}_x(\cdot)), u_{\varphi(\tau_i,x,\bar{u}_x(\cdot))}), u_{\varphi(\tau_i,x,\bar{u}_x(\cdot))}(t)) dt \\
&\leq v_\rho(x) + (1 + e^{-a})\varepsilon \sum_{j=0}^{\infty} (e^{-a})^i = v_\rho(x) + \frac{1 + e^{-a}}{1 - e^{-a}} \cdot \varepsilon
\end{aligned}$$

Wegen der rekursiven Definition von $\bar{u}_x(\cdot)$ gilt dies für alle $J_\rho(\varphi(\tau_i, x, \bar{u}_x(\cdot)), \bar{u}_x(\tau_i + \cdot))$, $i \in \mathbb{N}$.

Für die Zwischenstellen betrachte folgende Ungleichung:
Sei $\sigma > 0$ und $u_{x_0}(\cdot) \in \mathcal{U}$ gegeben, so daß $|J_\rho(x_0, u_{x_0}(\cdot)) - v_\rho(x_0)| \leq \varepsilon$. Dann gilt

$$\begin{aligned}
v_\rho(x_0) + \varepsilon &\geq \int_0^\infty e^{-\rho t} g(\varphi(t, x_0, u_{x_0}(\cdot)), u_{x_0}(t)) dt \\
&= \int_0^\sigma e^{-\rho t} g(\varphi(t, x_0, u_{x_0}(\cdot)), u_{x_0}(t)) dt \\
&\quad + e^{-\rho\sigma} \underbrace{\int_0^\infty e^{-\rho t} g(\varphi(t, \varphi(\sigma, x_0, u_{x_0}(\cdot)), u_{x_0}(\sigma + \cdot)), u_{x_0}(\sigma + t)) dt}_{= J_\rho(\varphi(\sigma, x_0, u_{x_0}(\cdot)), u_{x_0}(\sigma + \cdot))} \\
&\geq \int_0^\sigma e^{-\rho t} g(\varphi(t, x_0, u_{x_0}), u_{x_0}(t)) dt \;+\; e^{-\rho\sigma} v_\rho(\varphi(\sigma, x_0, u_{x_0})) \\
&\geq v_\rho(x_0).
\end{aligned}$$

Aus dieser Einschachtelung folgt

$$|v_\rho(\varphi(\sigma, x_0, u_{x_0}(\cdot))) - J_\rho(\varphi(\sigma, x_0, u_{x_0}(\cdot)), u_{x_0}(\sigma + \cdot))| \;\leq\; \varepsilon e^{\rho\sigma}.$$

Wenn $i \in \mathbb{N}$ nun maximal gewählt wird mit $\tau_i \leq \sigma$ und $x_0 := \varphi(\tau_i, x, \bar{u}_x(\cdot))$, folgt:

$$|v_\rho(\varphi(\sigma, x, \bar{u}_x(\cdot))) - J_\rho(\varphi(\sigma, x, \bar{u}_x(\cdot)), \bar{u}_x(\sigma + \cdot))| \leq \frac{1 + e^{-a}}{1 - e^{-a}} \cdot \varepsilon e^{\rho(\sigma - \tau_i)} \leq e^b \frac{1 + e^{-a}}{1 - e^{-a}} \cdot \varepsilon$$

□

Bemerkung 6.14 Die obige Abschätzung wird minimal für $a = b = \ln(1 + \sqrt{2})$. Es gilt dann $e^b \frac{1+e^{-a}}{1-e^{-a}} = 3 + 2\sqrt{2}$. Dies ist leicht zu sehen durch Gleichsetzen von a und b und Ableiten des Ausdrucks.

Bemerkung 6.15 Für die diskretisierte Trajektorie erfüllt (für genügend kleines $k > 0$) bereits die vom Algorithmus gelieferte Kontrolle $u_x(\cdot) \in \mathcal{U}_h$ eine ähnliche Behauptung, d.h es gilt

$$|J_h(\varphi_h(jh, x, u_x(\cdot)), u_x(jh + \cdot)) - v_h(\varphi_h(jh, x, u_x(\cdot)))| \leq \varepsilon.$$

Dies ergibt sich sofort aus Satz 5.28 und der Konstruktion von $u_x(\cdot)$.

Der folgende Satz formuliert nun das Ergebnis der Überlegungen zur numerischen Berechnung einer stabilisierenden Kontrolle.

Satz 6.16 Sei $\rho > 0$ und v_h^k die numerisch berechnete Wertefunktion von (6.5). Sei $A \subseteq \overline{\Omega^k}$ gegeben, so daß $\rho v_h^k(x) \leq -\delta < 0$ für beliebig kleine $h, k > 0$ und alle $x \in A$.
Zu $x \in A$ und $a \leq b \in \mathbb{R}$ sei $\bar{u}_x$ nach 6.12 mit $u_x(\cdot) = u_{x,h}^h(\cdot)$ aus Definition 5.26 konstruiert und es gelte $\varphi(x, t, \bar{u}_x(\cdot)) \in A \quad \forall t \geq 0$ ebenfalls für beliebig kleine $h, k > 0$.
Dann gibt es $H > 0, K(h) > 0$, so daß $\bar{u}_x$ das System (6.1) mit zu x korrespondierenden Startwerten stabilisiert für $h < H, k < K(h)$.

Beweis: Nach Satz 5.28 gibt es zu jedem $\varepsilon > 0$ $H > 0, K(h) > 0$, so daß $J_\rho(x, u_x(\cdot)) \leq -\delta + e^{-b}\frac{1-e^{-a}}{1+e^{-a}} \cdot \varepsilon \; \forall x \in A$. Da φ in A bleibt, folgt mit Lemma 6.13, daß $J_\rho(\varphi(t, x, \bar{u}(\cdot)), \bar{u}_x(t+\cdot)) \leq -\delta + \varepsilon \quad \forall t \geq 0$. Für hinreichend kleines $\varepsilon > 0$ sind also die Voraussetzungen von Satz 4.3 erfüllt, womit die zu $\bar{u}_x$ gehörige Trajektorie des ursprünglichen Systems negativen Lyapunov-Exponenten hat. □

Bemerkung 6.17 Für die Betrachtung der Stabilisierungsprobleme ist es sinnvoll, zusätzlich zu der oben definierten „numerischen“ Menge A die Menge $B := \{x \in \overline{\Omega^k} | \lambda^*(x) \leq 0\}$ zu betrachten. Diese Menge läßt sich nach Bemerkung 6.8 als Vereinigung von Kontrollmengen und ihren Einzugsbereichen darstellen. Verläßt eine Trajektorie die Menge B für alle $t \geq t_0 \in \mathbb{R}$, so muß es auch $t_1 \in \mathbb{R}$ geben, so daß die Trajektorie nicht mehr für alle $t \geq t_1$ in A liegt, da sich ansonsten ein Widerspruch zu Satz 4.3 ergäbe.
Nach den Sätzen 3.18 und 3.24 läßt sich – unter den dortigen Voraussetzungen – außerdem eine kompakte Menge $K \subset \overline{\Omega^k}$ finden, die sowohl in A als auch in B liegt.

Bemerkung 6.18 Es reicht i.A. nicht, bei Satz 6.16 vorauszusetzen, daß die Trajektorie zu einer *optimalen* Kontrolle in einer solchen Menge A bzw. B bleibt. Läuft z.B. diese Trajektorie „nahe am Rand“ der Menge, so kann eine Trajektorie herauslaufen, obwohl sie zu einer ε-*optimalen* Kontrolle gehört. Das gleiche gilt für Startwerte, die nahe am Rand von A oder B liegen.

Kapitel 7

Numerische Beispiele

In diesem Kapitel werden die numerischen Beispiele vorgestellt, die mit dem entwickelten Algorithmus berechnet wurden. Im ersten Teil wird zunächst der neue Algorithmus mit dem alten „beschleunigten Algorithmus “ verglichen.

Im zweiten Teil werden dann zwei Kontrollsysteme vorgestellt, die mit diesem Algorithmus stabilisiert wurden.

7.1 Vergleich der Algorithmen

Im Folgenden soll der Koordinatenaufstiegsalgorithmus aus Abschnitt 5.3.3 mit dem beschleunigten Algorithmus aus Abschnitt 5.3.2 verglichen werden. Als Vergleichskriterium wurde die Anzahl der Aufrufe von *iter_step* bzw. die Anzahl der Iterationen in *iterate* verwendet, was der Anzahl der Auswertungen des Operators T entspricht.
In den folgenden Tabellen ist der Aufwand für wechselnde Parameter h, k und ρ beschrieben. Alle Berechnungen wurden auf einer IBM6000-Workstation durchgeführt. Tabelle 7.1 zeigt den Vergleich der Algorithmen für das in [16] beschriebene System SAMPLE, mit dem auch dort die numerischen Tests durchgeführt wurden. Hier wie dort wurden $N = 16$ Knoten und $h = 0.01$ gewählt. Ops_1 bezieht sich auf den neuen Algorithmus, Ops_2 auf den alten beschleunigten Algorithmus. In Tabelle 7.1 bezeichnen err_1 bzw. err_2 den durchschnittlichen Fehler durch Gegenüberstellung mit den exakten Werten aus [16], Lemma 5.3.

ρ	V_0	Ops_1	err_1	Ops_2	err_2
2.0	-0.715	5	0.0022	198	0.0091
0.5	-2.86	5	0.0197	452	0.0336
0.0625	-22.88	5	0.0415	1396	0.0616

Tabelle 7.1: Abhängigkeit vom Diskontfaktor ρ im Beispiel SAMPLE zu oben angegebenen Parametern.

Die folgenden Vergleiche wurden mit dem projizierten zweidimensionalen linearen Oszillator mit $b = 1.5$ durchgeführt; das System ist im Abschnitt 7.2 beschrieben.

ρ	V_0	Ops_1	Ops_2
5.0	-0.66	16	2001
2.0	-1.66	35	5543
1.0	-3.32	42	11477
0.1	-33.23	194	121187
0.01	-332.27	1707	-
0.001	-3322.72	16836	-

Tabelle 7.2: Abhängigkeit von der Diskontrate ρ bei $h = 0.1$, $k = 0.032$.

h	Ops_1	Ops_2
1.0	13	11477
0.1	42	11477
0.01	51	11477

Tabelle 7.3: Abhängigkeit von h bei $k = 0.032$, $\rho = 1.0$.

k	Ops_1	Ops_2
0.063	28	5918
0.032	42	11477
0.0063	233	49625

Tabelle 7.4: Abhängigkeit von k bei $h = 0.1$, $\rho = 1.0$.

Die Anzahl der Operationen ist bei diesen Rechnungen bestimmt durch die Abbruchgenauigkeit. Hier wurde grundsätzlich der Wert 10^{-5} vorgegeben. Die berechnete Wertefunktion stimmt bei beiden Algorithmen bis auf eine Abweichung von etwa 10^{-3} überein. Bei den Gegenüberstellungen muß berücksichtigt werden, daß hier nur der Aufwand des Iterationsverfahrens gemessen wurde. Gerade bei kleinem k und vielen Kontrollwerten benötigt das Aufstellen der Λ-Matrix viel Zeit; dieser Abschnitt des Verfahrens ist bei beiden Algorithmen gleich.
In Tabelle 7.2 wurde beim alten Verfahren zu den Diskontraten $\rho = 0.01$ und $\rho = 0.001$ bei erlaubten 20000 Iterationen (dies entspricht in diesem Beispiel 460000 Operationen) keine Konvergenz erreicht.
Festzuhalten ist in jedem Fall, daß das hier entwickelte Verfahren in der Iteration gerade bei hoher Genauigkeit (speziell bei kleinen ρ) mit erheblich weniger Rechenaufwand auskommt. Einige der folgenden Berechnungen, bei denen mit recht hoher Genauigkeit gerechnet wird, wären mit dem alten Algorithmus in vernünftiger Zeit kaum zu rechnen gewesen.

In den nächsten beiden Abschnitten werden zwei Systeme vorgestellt, an denen der Stabilisierungsalgorithmus getestet wurde. Im einzelnen sind dies der zweidimensionale und der dreidimensionale lineare Oszillator zu verschiedenen Parametern, so daß verschiedene Fälle bezüglich Anzahl und Lage der Kontrollmengen auftreten.

7.2 Der zweidimensionale Lineare Oszillator

Der zweidimensionale lineare Oszillator ist beschrieben durch

$$\ddot{x} + 2b\dot{x} + (1+u)x = 0$$

bzw. als zweidimensionales System mit $x_1 = x,\ \ x_2 = \dot{x}$

$$\begin{pmatrix} \dot{x_1} \\ \dot{x_2} \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & 1 \\ -1-u & -2b \end{pmatrix}}_{=:A(u)} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \tag{7.1}$$

Für das auf $\mathbb{P}^1$ projizierte System $s = \frac{x}{\|x\|}$ gilt nach Lemma 6.4:

$$\dot{s} = \begin{pmatrix} s_2(1 + us_1^2 + 2bs_1s_2) \\ -(1+u)s_1 - 2bs_2 + s_2^2(us_1 + 2bs_2) \end{pmatrix} \tag{7.2}$$

Mit der Parametrisierung $s(t) = (s_1(t), s_2(t)) = (\cos\varphi(t), \sin\varphi(t))$ ergibt sich so

$$\dot{s}(t) = \begin{pmatrix} -\sin\varphi(t) \\ \cos\varphi(t) \end{pmatrix} \dot{\varphi}(t) = \begin{pmatrix} -s_2(t) \\ s_1(t) \end{pmatrix} \dot{\varphi}(t) \tag{7.3}$$

Aus den ersten Zeilen von (7.2) und (7.3) folgt nun die Differentialgleichung für φ:

$$\dot{\varphi} = -(1 + u\cos^2\varphi + 2b\sin\varphi\cos\varphi) \tag{7.4}$$

Die Kostenfunktion lautet mit dieser Parametrisierung:

$$g(\varphi, u) = -\sin\varphi(u\cos\varphi + 2b\sin\varphi) \tag{7.5}$$

Für den zweidimensionalen linearen Oszillator wurden zwei Fälle untersucht. Für die Wahl der Parameter betrachtet man die Eigenwerte von $A(u)$ für ein $u \in U$. Diese sind $\lambda_{1,2} = -b \pm \sqrt{b^2 - 1 - u}$. Nach Satz 6.7 kann man nun die Eigenwerte z.B. zu $u = 0$ betrachten, und dadurch auf die Kontrollmengen für den Kontrollwertebereich $U = [-a, a]$, a genügend klein, schliessen.

7.2.1 Das System mit einer Kontrollmenge

Im ersten Fall wurde der Parameter $b = 0$ gewählt. In diesem Fall existieren die komplexen Eigenwerte $\pm i$ und damit eine Kontrollmenge, die ganz $\mathbb{P}^1$ umfaßt. Zur Berechnung wurden die numerischen Parameter wie folgt gesetzt: $k = 0.016$, $h = 0.01$, $\rho = 0.01$, $U = \{-0.5, 0.5\}$. Die berechneten Werte $v_h^k(x)$ lagen dabei im Intervall $[-0.162009, -0.161327]$, sind also bis auf Abweichungen durch die Rechengenauigkeit konstant auf ganz $\mathbb{P}^1$. Bild 7.1 zeigt die Trajektorien zu den konstanten Kontrollen $u \equiv -0.5$ und $u \equiv 0.5$, sowie die Trajektorie zur vom Programm berechneten Kontrolle. Es zeigt sich, daß diese tatsächlich exponentiell stabil ist.

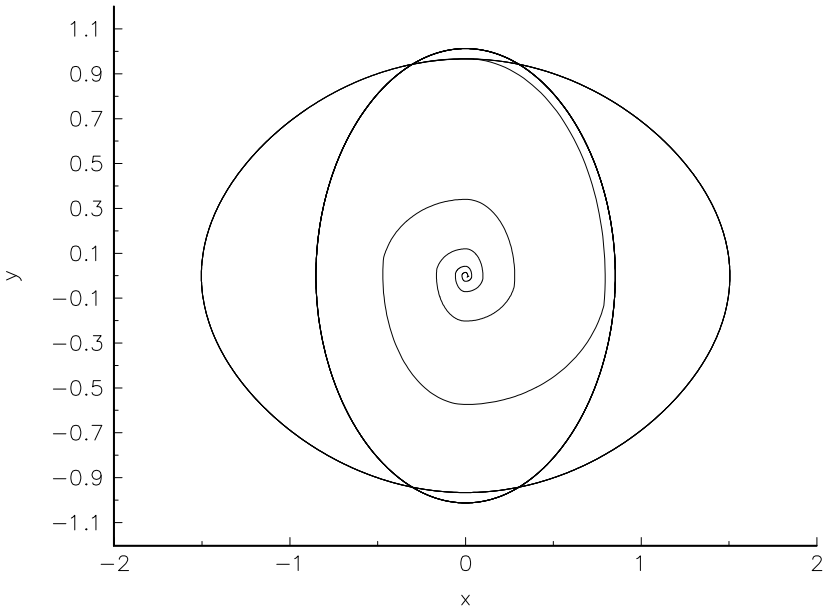

Abbildung 7.1: Stabilisierung bei einer Kontrollmenge.

7.2.2 Das System mit zwei Kontrollmengen

Im zweiten Fall wurde $b = 1.5$ gesetzt. In diesem Fall gibt es zwei unterschiedliche reelle Eigenwerte und damit zwei Kontrollmengen, in denen sich die Lyapunov-Exponenten stark unterscheiden. Abbildung 7.2 zeigt die vom Programm berechnete optimale Wertefunktion für verschiedene $\rho > 0$. Die Berechnungen wurden durchgeführt mit den Parametern $k = 0.0016$ in einer Umgebung der varianten Kontrollmenge, $k = 0.016$ sonst, $h = 0.01$ und $U = \{-0.5, 0.5\}$. Die Sprungstelle, die sich bei kleinen ρ abzeichnet, deckt sich genau mit dem Rand der varianten Kontrollmenge.

Es liegt nun nahe, anzunehmen, daß für Startwerte, deren Projektionen nach $\mathbb{P}^1$ in dieser varianten Kontrollmenge liegen, die mit der Kontrolle des Programms gesteuerte Trajektorie schneller nach Null läuft, wenn die entsprechenden Voraussetzungen aus Satz 6.16 erfüllt sind. Die folgenden Trajektorien in Bild 7.3 (berechnet mit $\rho = 0.01$) zeigen, daß genau dieses der Fall ist. Die von links oben nach Null laufenden Trajektorien sind diejenigen, deren Projektionen in der varianten Kontrollmenge liegen, während die anderen Trajektorien zuerst in die invariante Kontrollmenge laufen und dann in den Nullpunkt. Die variante Kontrollmenge ist in der Grafik durch die gepunkteten Linien angedeutet, die invariante durch die gestrichelten Linien.

Die Ränder der Kontrollmengen können ausgerechnet werden, wenn man berücksichtigt, daß sie bei diesem einfachen System gerade durch die Eigenräume zu den extremalen Kontrollen gegeben sind. Deren Repräsentanten auf $\mathbb{P}^1$ lassen sich leicht explizit ausrechnen; es gilt $s_1 = \pm\sqrt{\frac{1}{1+(-b\pm\sqrt{b^2-1-u})^2}}$.

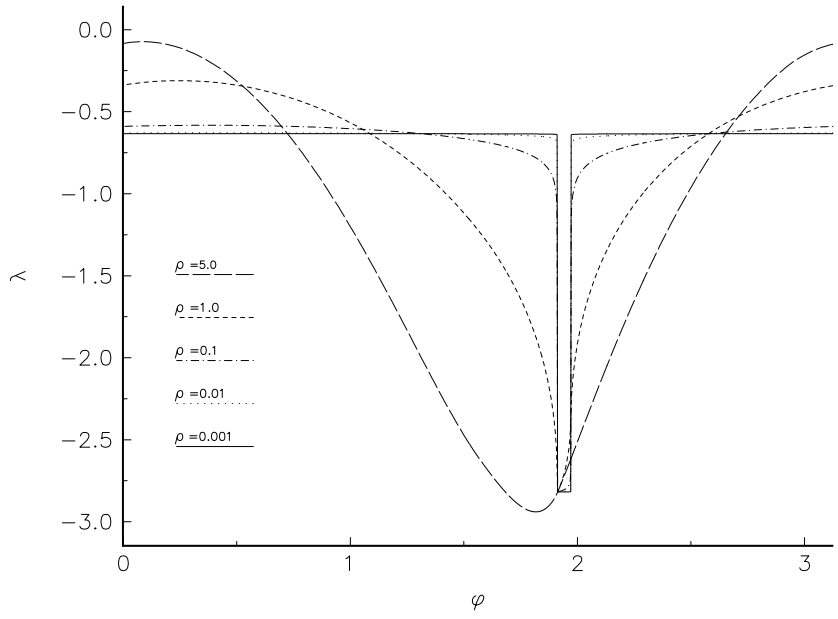

Abbildung 7.2: Wertefunktion zu wechselnden $\rho > 0$ bei zwei Kontrollmengen.

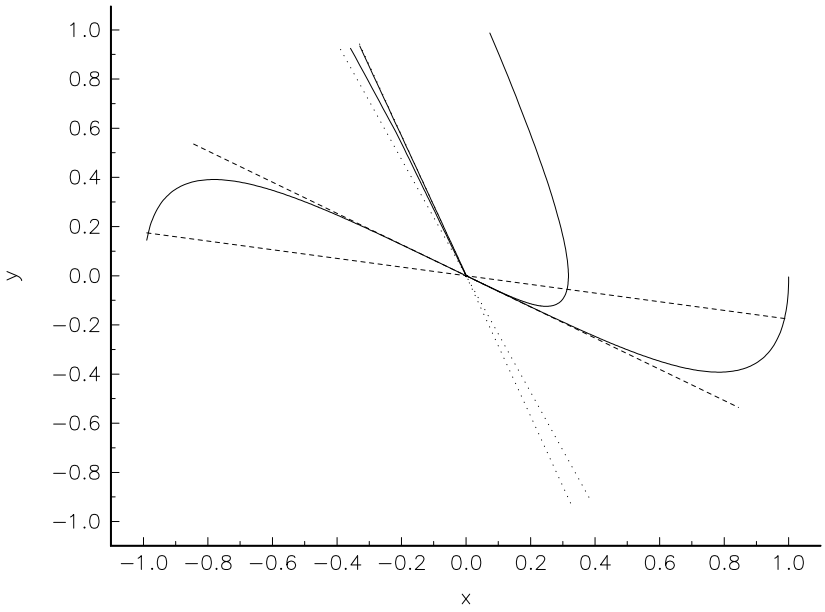

Abbildung 7.3: Stabilisierung bei zwei Kontrollmengen.

Als weiterer numerischer Test mit diesem System wurden die Spektralintervalle numerisch berechnet. Um den maximalen Lyapunov-Exponent in der varianten Kontrollmenge D_1 zu berechnen, wurden zwei Strategien verwendet. Zum einen wurden nur die $u \in \mathcal{U}_h$ betrachtet, für die die zugehörigen Trajektorien in dieser Kontrollmenge blieben. Zum anderen wurde das zeitumgekehrte System betrachtet, für das diese Kontrollmenge gerade die invariante ist, die Lyapunov-Exponenten aber gleichbleiben, was aus der Betrachtung der Floquet-Exponenten folgt. Mit $\rho = 0.01$, $h = 0.001$ und $k = 0.0006$ ergab sich in beiden Fällen ein Wert von -2.37. Zusammen mit dem Wert -2.81, der sich beim Minimieren ergab, lautet die numerische Berechnung des Spektralintervalls also $cl\Sigma_{Ly}(D_1) = [-2.81, -2.37]$. Für die invariante Kontrollmenge D_2 ergab sich $cl\Sigma_{Ly}(D_2) = [-0.62, -0.19]$.

7.3 Der dreidimensionale Lineare Oszillator

Der dreidimensionale lineare Oszillator ist gegeben durch die Gleichung

$$\dddot{y} + a\ddot{y} + b\dot{y} + (c + u)y = 0 \tag{7.6}$$

oder als dreidimensionales System beschrieben durch

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -(c+u) & -b & -a \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \tag{7.7}$$

mit $a, b, c \in \mathbb{R}$ und $u \in U$.
Mit der Projektion nach $\mathbb{P}^2$ mittels Lemma 6.4 und $s = \frac{x}{\|x\|}$ ergibt sich also:

$$\dot{s} = \begin{pmatrix} s_2 - s_1(-(c+u)s_1s_3 + s_1s_2 + (1-b)s_2s_3 - as_3^2) \\ s_3 - s_2(-(c+u)s_1s_3 + s_1s_2 + (1-b)s_2s_3 - as_3^2) \\ -(c+u)s_1 - bs_2 - as_3 - s_3(-(c+u)s_1s_3 + s_1s_2 + (1-b)s_2s_3 - as_3^2) \end{pmatrix} \tag{7.8}$$

Für die Parametrisierung der Sphäre wurde die im Anhang A beschriebene Stereographische Projektion (A.6) gewählt.
Durch Ableiten von (A.6) erhält man

$$\begin{aligned} \dot{s}_1 &= \frac{2\dot{x}_1}{1+\|x\|^2} - \frac{4\langle \dot{x}, x\rangle x_1}{(1+\|x\|^2)^2} \\ \dot{s}_2 &= \frac{2\dot{x}_2}{1+\|x\|^2} - \frac{4\langle \dot{x}, x\rangle x_2}{(1+\|x\|^2)^2} \\ \dot{s}_3 &= -\frac{4\langle \dot{x}, x\rangle}{(1+\|x\|^2)^2} \end{aligned}$$

also lautet die Umkehrung

$$\begin{aligned} \dot{x}_1 &= \frac{(\dot{s}_1 - \dot{s}_3 x_1)(1+\|x\|^2)}{2} \\ \dot{x}_2 &= \frac{(\dot{s}_2 - \dot{s}_3 x_2)(1+\|x\|^2)}{2} \end{aligned} \tag{7.9}$$

Die rechte Seite ergibt sich also aus den Gleichungen (7.8), (A.6) und (7.9).

Die Zielfunktion lautet nach Lemma 6.4 mit $\phi = \phi_S$ aus (A.6):

$$g(x,u) = (-c-u)\phi_1(x)\phi_3(x) + \phi_1(x)\phi_2(x) + (1-b)\phi_2(x)\phi_3(x) - a\phi_3(x)^2 \tag{7.10}$$

Die folgenden grafischen Darstellungen der Ergebnisse in $\mathbb{P}^2$ sind der besseren Übersichtlichkeit wegen in Kugelkoordinaten ($s_1 = \sin\theta\cos\varphi$, $s_2 = \sin\theta\sin\varphi$, $s_3 = \cos\theta$) dargestellt. Außerdem wurde das System für die Ausgabe mittels der Standard-Transformation $z(t) := e^{\frac{1}{3}at}y(t)$ transformiert.

Die Analysen der Kontrollmengen sowie die Grafiken, die die Kontrollmengen zeigen, wurden mit dem Programm CS2DIM zur numerischen Berechnung von Kontrollmengen von Gerhard Häckl durchgeführt.

Auch für den dreidimensionalen linearen Oszillator wurden zwei Fälle mit unterschiedlicher Anzahl von Kontrollmengen untersucht. Hierzu wurden wiederum die Eigenwerte bzw. Eigenräume von A(0) betrachtet.
Das charakteristische Polynom von A(0) lautet $\chi(x) = x^3 + b_1x + b_0$ mit $b_1 = b - \frac{1}{3}a^2$, $b_0 = c - \frac{1}{3}ab + \frac{2}{27}a^3$. Für die Eigenwerte betrachtet man nun $D = (\frac{b_1}{3})^3 + (\frac{b_0}{2})^2$. Für $D > 0$ existieren ein reeller Eigenwert und zwei konjugiert komplexe Eigenwerte, deren Realteil gleich Null ist, also existieren zwei generalisierte Eigenräume. Für $D < 0$ existieren drei reelle Eigenwerte, also drei Eigenräume.
Mit diesen Vorüberlegungen kann man nun Parameter finden, so daß das Kontrollsystem für $U = [-a, a]$, $a \in \mathbb{R}$, zwei bzw. drei Kontrollmengen besitzt.

7.3.1 Das System mit zwei Kontrollmengen

In diesem ersten Fall wurden die Parameter wie folgt gewählt: $a = 1,\ b = 0,\ c = 0.5$ ($\Rightarrow D = 0.094$), $U = \{-0.3, -0.25, \ldots, 0.25, 0.3\}$. Mit diesen Parametern ergibt sich eine invariante und eine variante Kontrollmenge. Bild 7.4 zeigt diese Mengen.

Da die variante Kontrollmenge (links unten und rechts oben im Bild) recht klein ist, gerade aber die Wertefunktion in dieser Menge und in einer Umgebung dieser Menge interessant ist, wurde zur Berechnung der Wertefunktion das Gitter in einer Umgebung dieser Kontrollmenge feiner gewählt als auf dem Rest von $\overline{\Omega}$. Die Werte wurden gewählt als $k = 0.1$ außerhalb einer Umgebung der varianten Kontrollmenge und $k = 0.0007$ um diese Menge. Außerdem wurde $h = 0.1$ und $\rho = 0.01$ gewählt. Die optimale Wertefunktion ist in Bild 7.5 dargestellt. (Die Gitterpunkte in der Netzgrafik entsprechen nicht den Knoten bei der Berechnung.)

Bei Verkleinerung von ρ ohne gleichzeitige Verkleinerung von h und k konnte keine Verbesserung der Wertefunktion, d.h. ein „schärferer Knick“ am Rand der Kontrollmenge, mehr erzielt werden.
Es zeigt sich jedoch, daß die Genauigkeit der Wertefunktion zur Stabilisierung des Systems für Anfangswerte, deren Projektion nach $\mathbb{P}^2$ in der varianten Kontrollmenge und nicht zu nahe am Rand liegen, ausreicht. In Bild 7.6 ist der Verlauf einer optimalen Trajektorie in $\mathbb{P}^2$ dargestellt. In Bild 7.7 sind die stabilisierte Trajektorie in $\mathbb{R}^3$ sowie gestrichelt die Trajektorien zu den konstanten Kontrollen $u \equiv -0.3$ und $u \equiv 0.3$ dargestellt.

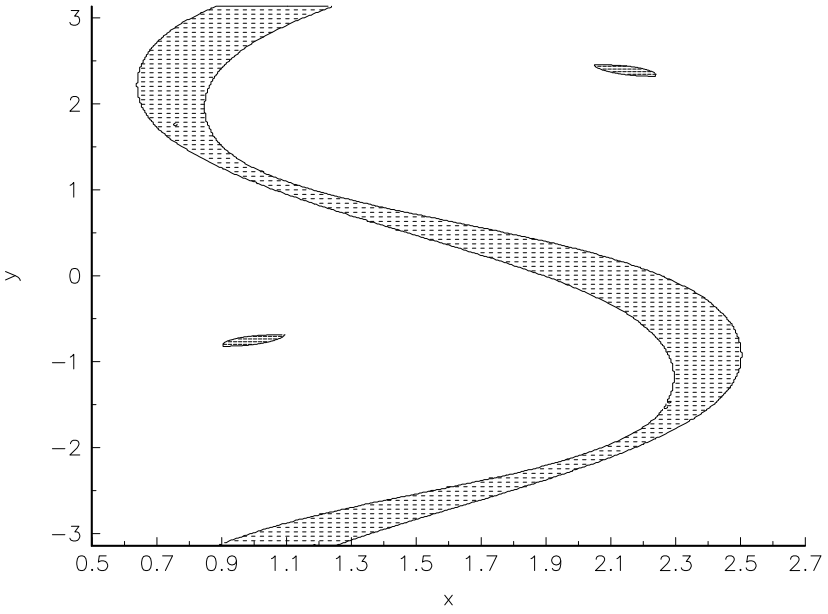

Abbildung 7.4: Lage der zwei Kontrollmengen.

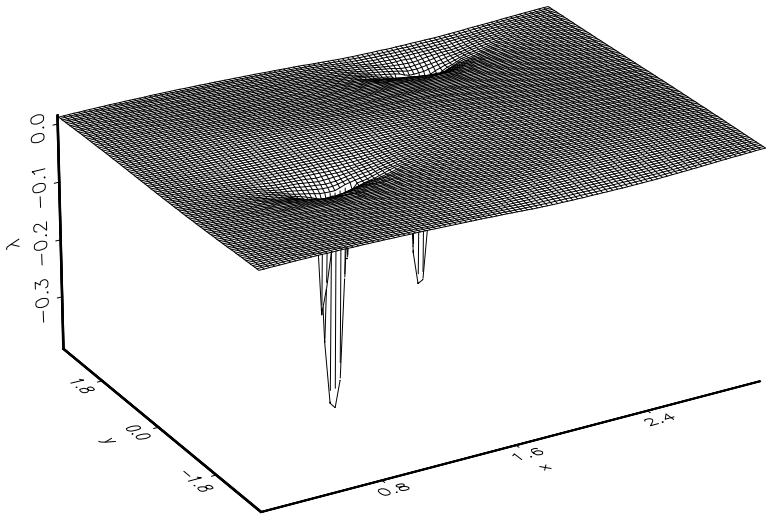

Abbildung 7.5: Wertefunktion bei zwei Kontrollmengen.

Für Anfangswerte, die nicht zu nahe am Rand der Kontrollmenge liegen, zeigt sich nach einigen Zeitschritten jeweils ein ähnlicher Verlauf ungefähr in der Mitte der Kontrollmenge. Bei Anfangswerten, die nahe am Rand der Kontrollmenge liegen, kommt es vor, wie nach Bemerkung 6.18 zu erwarten war, daß die berechnete Trajektorie aus der Kontrollmenge herausläuft und die Lösung dann nicht mehr stabil ist.

7.3.2 Das System mit drei Kontrollmengen

In diesem Fall wurden die Parameter $a = -1$, $b = -3$, $c = 0.5$ ($\Rightarrow D = -1.28$), $U = \{-1.0, -0.9, \ldots, 0.9, 1.0\}$ gewählt. Bild 7.8 zeigt die Lage der Kontrollmengen sowie die Einzugsbereiche von D_2, der zweiten varianten Kontrollmenge. Die Kontrollmenge unten links bzw. oben rechts ist die offene variante, die Kontrollmenge unten rechts bzw. oben links ist die invariante, in der Mitte liegt die zweite variante Kontrollmenge.

Die Berechnung wurde durchgeführt mit $k = 0.1$, $h = 0.05$, $\rho = 0.01$. Bild 7.9 zeigt die optimale Wertefunktion, in Bild 7.10 ist der Bereich markiert, in dem die Wertefunktion negativ ist.

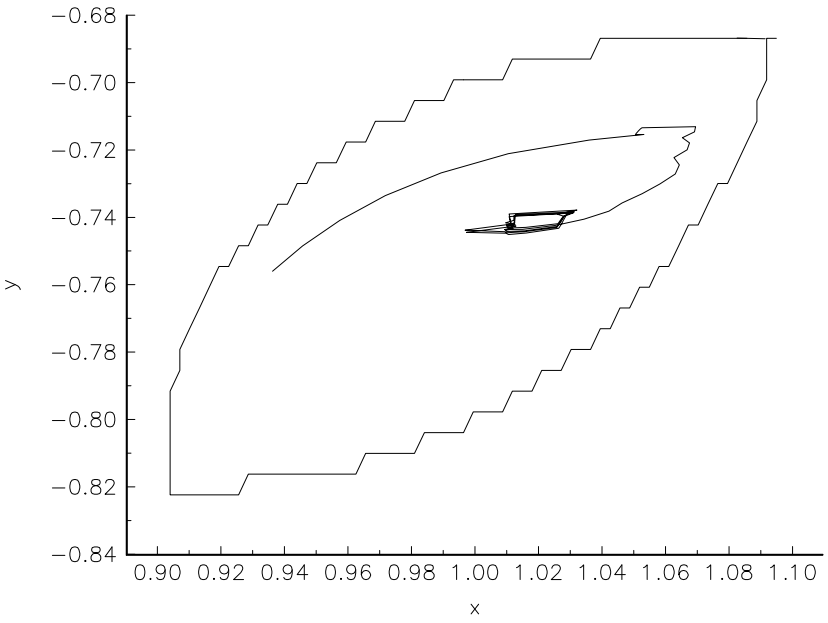

Abbildung 7.6: Optimale Trajektorie in $\mathbb{P}^2$ bei zwei Kontrollmengen.

Bild 7.11 zeigt eine optimale Trajektorie in $\mathbb{P}^2$, die im Einzugsbereich der zweiten varianten Kontrollmenge startet. Die Trajektorie läuft schnell in diese Kontrollmenge und bleibt dort.

In Bild 7.12 ist die zugehörige Trajektorie im $\mathbb{R}^3$ dargestellt. Zusätzlich sind dort die Trajektorien zu gleichem Anfangswert und konstanten Kontrollen $u \equiv -1$ sowie $u \equiv 1$ gestrichelt abgebildet. Auch hier reicht also die Genauigkeit der Wertefunktion aus, um das System zu stabilisieren.

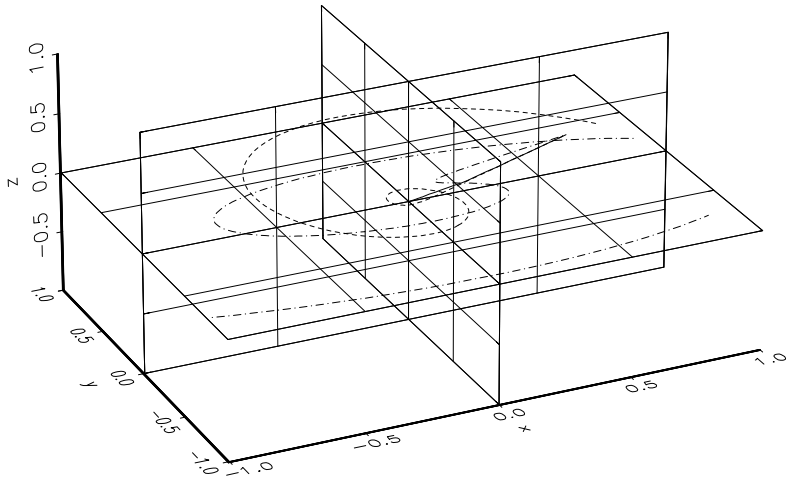

Abbildung 7.7: Optimale Trajektorie in $\mathbb{R}^3$ bei zwei Kontrollmengen.

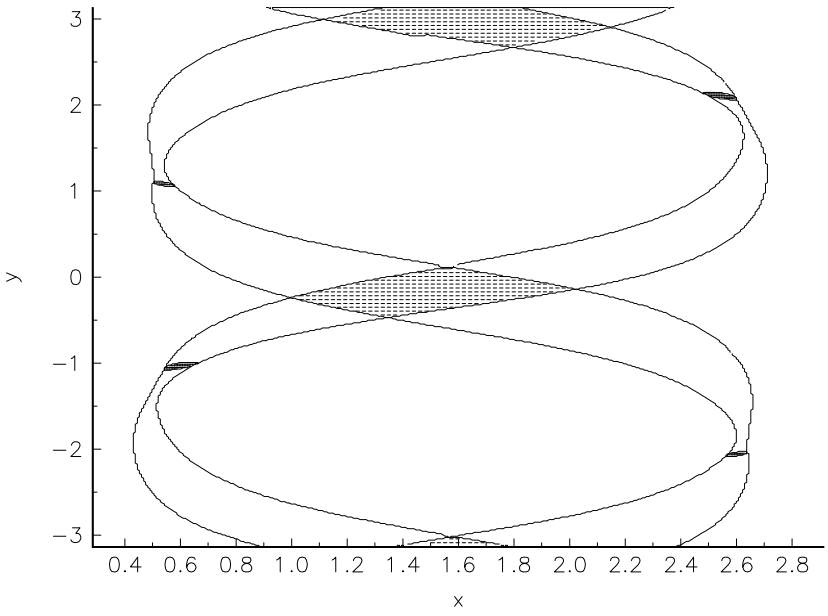

Abbildung 7.8: Lage der drei Kontrollmengen.

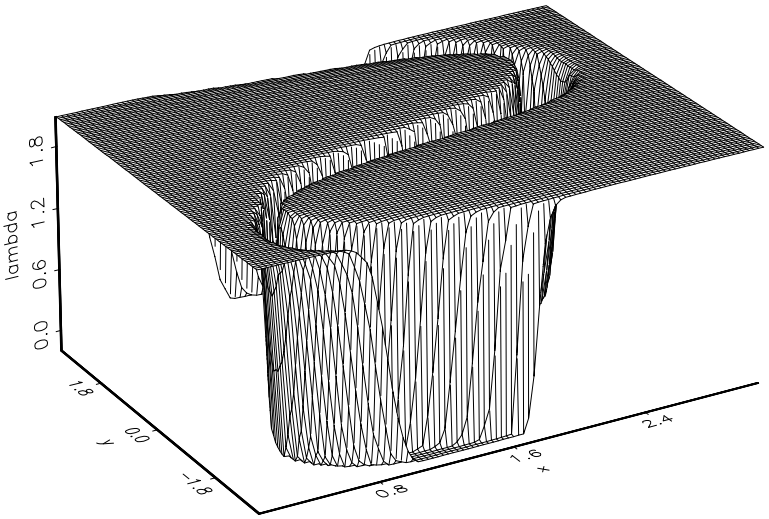

Abbildung 7.9: Wertefunktion bei drei Kontrollmengen.

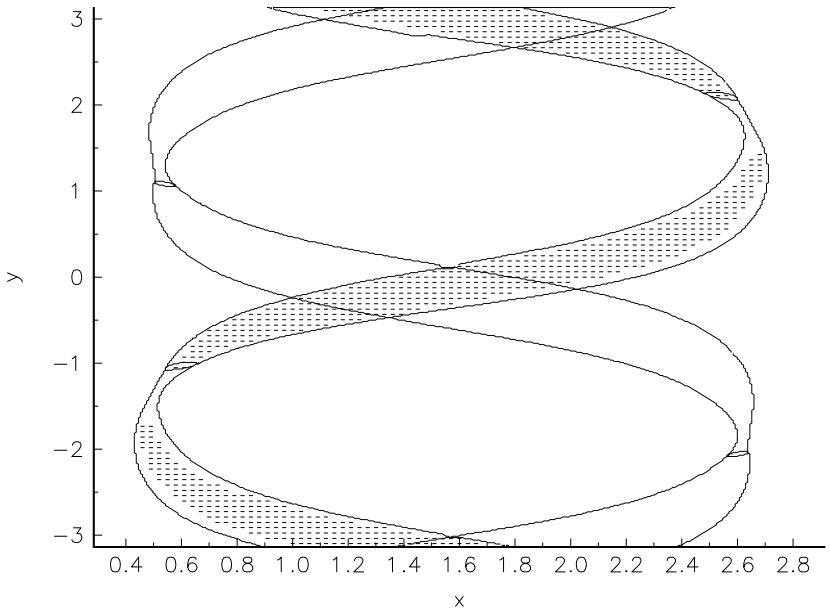

Abbildung 7.10: Negativer Bereich bei drei Kontrollmengen.

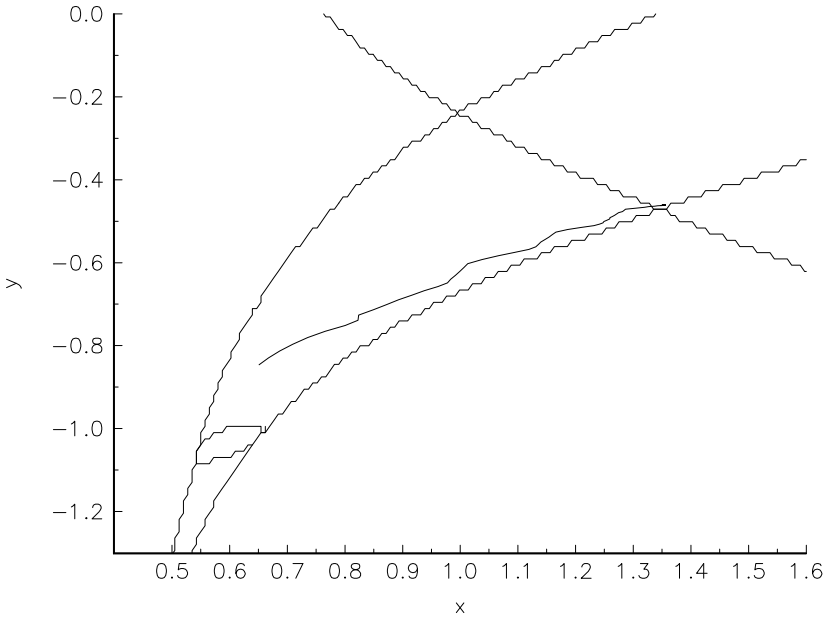

Abbildung 7.11: Optimale Trajektorie in $\mathbb{P}^2$ bei drei Kontrollmengen.

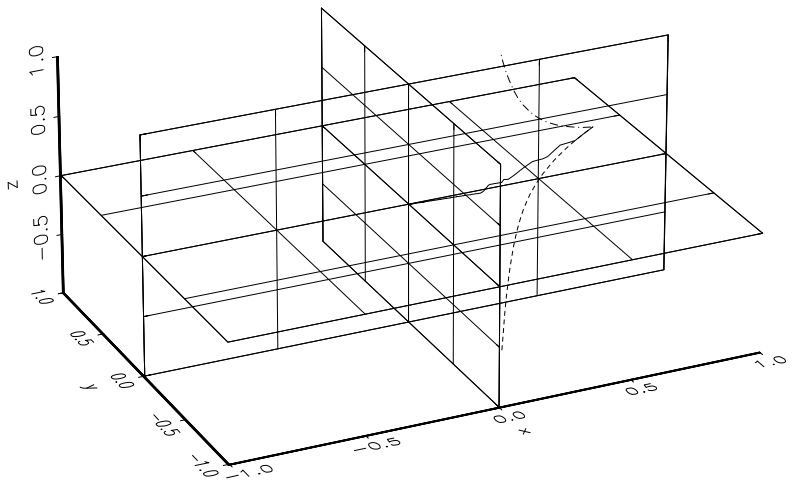

Abbildung 7.12: Optimale Trajektorie in $\mathbb{R}^3$ bei drei Kontrollmengen.

Anhang A

Trajektorien im Projektiven Raum

In den Kapiteln 2 und 5 sind wir von einem System im $\mathbb{R}^n$ und einer Funktion Ψ mit gewissen Eigenschaften (im Wesentlichen Erhaltung der stetigen Abhängigkeit vom Anfangswert) ausgegangen. In diesem Abschnitt werden wir zeigen, daß sich ein System im Projektiven Raum in ein solches umformen läst.

Wir werden hier wie in Kapitel 6 den Projektiven Raum als Sphäre auffassen, bei der gegenüberliegende Punkte identifiziert werden, da dies der Projektion mittels $x/\|x\|$ entspricht.

A.1 Umformulierung in ein lokales Problem

Wir haben aus Lemma 6.4 ein Kontrollsystem auf $\mathbb{P}^{n-1}$ gegeben. Um die Theorie aus den Kapiteln 2 und 5 anzuwenden, müssen wir dies nun in ein Problem auf einer offenen Menge $W \subset \mathbb{R}^{n-1}$ umformulieren.

Der einfacheren Notation wegen wird in diesem Abschnitt keine Abhängigkeit der rechten Seite von einem Kontrollwert u angenommen. Alle Aussagen gelten aber analog.

Wir werden nun Voraussetzungen für beliebige kompakte Mannigfaltigkeiten formulieren, unter denen sich eine Funktion Ψ, wie wir sie benötigen, finden läßt; danach wird gezeigt, daß $\mathbb{P}^{n-1}$ diese Voraussetzungen erfüllt.

Wir betrachten also eine kompakte n-dimensionale Riemannsche Mannigfaltigkeit M mit lokalen Parametrisierungen (ϕ_i, U_i), $i \in I$ und $\bigcup_{i\in I} \phi_i(U_i) = M$. Wegen der Kompaktheit von M können wir I endlich wählen.

Auf M sei ein Vektorfeld $X : M \to TM$ gegeben, so daß zu jedem $p \in M$ die Lösungstrajektorie $p(t)$ für alle $t \geq 0$ existiert. Die lokale Darstellung von X in U_i bezeichnen wir mit f_i, d.h. $f_i(x) := D\phi_i^{-1} X(\phi_i(x))$. Eine Riemann'sche Metrik auf M sei bezeichnet mit $d(\cdot,\cdot)$.

Die diskretisierte Trajektorie $p_h(\cdot)$ zu $p \in M$, $h > 0$ ist definiert durch

$$p_h(t) = p_j \quad \text{für } t \in [jh, (j+1)h),$$

mit

$$p_0 = p, \quad p_{j+1} = \phi_{i_j}(\phi_{i_j}^{-1}(p_j) + hf_{i_j}(\phi_{i_j}^{-1}(p_j))), \tag{A.1}$$

wobei die unten stehende Forderung (A.4) für hinreichend kleine $h > 0$ die Existenz einer solchen Parametrisierung ϕ_{i_j} zu jedem p_j sicherstellt.
Für die Zeit, in der $p(\cdot)$ bzw. $p_h(\cdot)$ in einer Koordinatenumgebung $\phi_i(U_i)$ bleiben, können wir die entsprechenden lokalen Trajektorien definieren durch

$$\hat{\varphi}(\cdot, \phi_i^{-1}(p)) := \phi_i^{-1}(p(\cdot)) \quad \text{und} \quad \hat{\varphi}_h(\cdot, \phi_i^{-1}(p)) := \phi_i^{-1}(p_h(\cdot)).$$

$\hat{\varphi}(\cdot, x_0)$ ist dann die Lösung der gewöhnlichen Differentialgleichung $\dot{\hat{\varphi}}(t, x_0) = f_i(\hat{\varphi}(t, x_0))$ mit Anfangswert $x_0 \in U_i$, $\hat{\varphi}_h(\cdot, x_0)$ ist deren Diskretisierung per Euler-Verfahren.

Wir stellen nun folgende Anforderungen an M, X und die (ϕ_i, U_i):

Die f_i seien auf allen U_i beschränkt und Lipschitz – stetig mit Konstanten M_f, L_f unabhängig von i. (A.2)

$\alpha\|\phi_i^{-1}(p) - \phi_i^{-1}(q)\| \le d(p,q) \le \beta\|\phi_i^{-1}(p) - \phi_i^{-1}(q)\| \quad \forall i \in I,\ p, q \in \phi_i(U_i)$
für positive reelle Konstanten α, β. (A.3)

$\exists T > 0 : \ \forall p \in M \ \exists i \in I$ mit
$p(t) \in \phi_i(U_i) \ \forall t \in [0, T], \quad \phi_i^{-1}(p_j) + hf_i(\phi_i^{-1}(p_j)) \in U_i \ \forall j \le \dfrac{T}{h}$ (A.4)

Für p und (ϕ_i, U_i), die (A.4) erfüllen, schreiben wir kurz $p \leftrightarrow \phi_i$. Mit dieser Notation können wir die letzte Anforderung formulieren.

$\forall p, q \in M$ mit $p \leftrightarrow \phi_i$, $q \leftrightarrow \phi_j$, $i \neq j$ gibt es eine Kette
$p = r_0, r_1, \ldots, r_k = q$ auf der kürzesten Verbindung von p nach q in M und
$U_i = U_{l_0}, U_{l_1}, \ldots, U_{l_{k+1}} = U_j$ mit $r_i \leftrightarrow U_{l_i}$, $r_i \leftrightarrow U_{l_{i+1}} \quad \forall i \in \{0, \ldots, k\}$ (A.5)

Falls X und damit die f_i von weiteren Parametern abhängen, fordern wir, daß (A.4) zu festem i für alle möglichen Parameter erfüllt ist.

Bevor wir jetzt zeigen, daß $\mathbb{S}^{n-1}$ und $\mathbb{P}^{n-1}$ mit „geeignetem“ Vektorfeld X diese Anforderungen erfüllen, betrachten wir zunächst noch eine Parametrisierung, die wir im weiteren verwenden werden. Hierbei beschränken wir uns wegen der einfacheren Notation auf den Fall $n = 3$, der uns für die in dieser Arbeit diskutierten numerischen Beispiele genügt. Die folgenden Aussagen gelten aber auch für beliebige höhere Dimensionen.

Definition A.1 Die stereographische Projektion ist definiert durch

$$\begin{aligned} \phi_S : \mathbb{R}^2 &\to \mathbb{S}^2 \setminus \{(0,0,-1)\} \\ \phi_S(x_1, x_2) &= \left(\frac{2x_1}{1+\|x\|^2}, \frac{2x_2}{1+\|x\|^2}, \frac{2}{1+\|x\|^2} - 1\right) \end{aligned} \tag{A.6}$$

$$\begin{aligned} \phi_N : \mathbb{R}^2 &\to \mathbb{S}^2 \setminus \{(0,0,1)\} \\ \phi_N(x_1, x_2) &= \left(\frac{2x_1}{1+\|x\|^2}, \frac{2x_2}{1+\|x\|^2}, 1 - \frac{2}{1+\|x\|^2}\right). \end{aligned} \tag{A.7}$$

Bemerkung A.2 Die stereographische Projektion ist eine Parametrisierung der Sphäre.

Lemma A.3 Sei X ein Vektorfeld auf $\mathbb{S}^2$, so daß für alle beschränkten offenen Mengen $\tilde{W} \subset \mathbb{R}^2$ für die bzgl. ϕ_S, ϕ_N zu X gehörigen lokalen Darstellungen f_S und f_N Bedingung (A.2) mit $M_f(\tilde{W})$ und $L_f(\tilde{W})$ erfüllt ist.
Dann gibt es $U_N = U_S = W \subset \mathbb{R}^2$, so daß die Bedingungen (A.2) – (A.5) erfüllt sind.

Beweis: Es bezeichne $D_\delta := \{x \in \mathbb{R}^2 \,|\, \|x\| < 1 + \delta\}$ die offene Scheibe in $\mathbb{R}^2$ mit Radius $1 + \delta$ um den Nullpunkt. Dann gilt für beliebige $\delta > 0$

$$\mathbb{S}^2 = \phi_S(D_\delta) \cup \phi_N(D_\delta).$$

Die Existenz von α und β, die (A.3) erfüllen, kann aus der Existenz von $\tilde{\alpha}$ und $\tilde{\beta}$ in einer ähnlichen Abschätzung für die Länge von Kurven (vgl. [2], Beweis zu Theorem 3.1) hergeleitet werden. Für die Sphäre ist die Existenz von α und β auf D_δ aber auch anschaulich klar, α und β hängen dabei von δ ab.
Wir wählen nun ein beliebiges $\delta > 0$ und schätzen ab, wie weit sich eine Trajektorie zu vorgegebener Zeit $T > 0$ von einem beliebigen Startpunkt auf dem Äquator (also der Menge A:=$\{(x, y, z) \in \mathbb{S}^2 \,|\, z = 0\}$) vom Äquator entfernt. Wir brauchen dabei nur Trajektorien zu berücksichtigen, die den Äquator nicht kreuzen, also o.B.d.A. ganz in $\phi_S(D_\delta)$ liegen. Daher können wir die lokalen Trajektorien $\hat{\varphi}(\cdot, x_0)$ und $\hat{\varphi}_h(\cdot, x_0)$ betrachten. Für diese gilt:

$$\|x_0 - \hat{\varphi}(T, x_0)\| = \int_0^T f_S(\hat{\varphi}(t, x_0))dt \leq M_f T \tag{A.8}$$

und

$$\|x_0 - \hat{\varphi}_h(T, x_0)\| = \sum_{j=1}^{\left[\frac{T}{h}\right]} h f_S(\hat{\varphi}_h(jh, x_0))dt \leq M_f T. \tag{A.9}$$

Also gilt für $p \in A$

$$d(p, p(T)) \leq \beta \|\phi_S^{-1}(p) - \hat{\varphi}(T, \phi_S^{-1}(p))\| \leq \beta M_f T$$

und ebenso für $p_h(T)$.
Wir wählen nun δ so groß, daß alle Punkte mit Abstand $\beta M_f T$ vom Äquator sowohl in $\phi_S(D_\delta)$ also auch in $\phi_N(D_\delta)$ liegen und setzen $W := D_\delta$.
Klar ist, daß mit diesem W die Bedingungen (A.2) bis (A.4) erfüllt sind.
Um zu zeigen, daß auch Bedingung (A.5) erfüllt ist, benötigen wir folgende Überlegung:
Wenn p und q auf einer Halbkugel der Sphäre liegen, ist (A.5) wegen (A.4) sofort erfüllt. Wenn sie auf verschiedenen Halbkugeln liegen, gibt es auf der kürzesten Verbindung sicher einen Punkt r auf dem Äquator der Sphäre. Nach Konstruktion von W erfüllt dieser Punkt $r \hookrightarrow \phi_N$ und $r \hookrightarrow \phi_S$, also ist Bedingung (A.5) ebenfalls erfüllt. □

Das folgende Lemma liefert uns die Möglichkeit, Differentialgleichungen, die durch ein Vektorfeld auf $\mathbb{P}^{n-1}$ gegeben sind, in ein Problem auf einer offenen Teilmenge des $\mathbb{R}^{n-1}$ umzuformen.

Lemma A.4 Es seien die Voraussetzungen von Lemma A.3 erfüllt. Die Funktion Ψ sei mit W aus Lemma A.3 gegeben durch:

$$\begin{aligned} \Psi : \mathbb{R}^{n-1} &\to W \\ x &\mapsto \begin{cases} x, & \text{falls } x \in W \\ \frac{x}{\|x\|^2}, & \text{sonst} \end{cases} \end{aligned} \tag{A.10}$$

Weiterhin sei $g : \mathbb{P}^{n-1} \to \mathbb{R}$ eine beliebige reellwertige und Lipschitz-stetige Funktion auf $\mathbb{P}^{n-1}$.
Dann gilt mit obiger Notation:

(i) Die Lösung von $\dot{\varphi}(t, x_0) = f_S(\varphi(t, x_0)),\ \varphi(0, x_0) = x_0 \in \mathbb{R}^{n-1}$ existiert eindeutig für alle $t > 0$, $x_0 \in \mathbb{R}^{n-1}$, für die die Trajektorie $p(\tau)$ in M mit $p = \phi_S(x_0)$ nicht durch $(0, \ldots, 0, -1) \in \mathbb{S}^{n-1}$ läuft für alle $\tau \leq t$.

(ii) Für diese Lösung gilt: $\Psi(\varphi(t, x_0)) = \Psi(\varphi(t, \Psi(x_0)))\quad \forall x_0 \in \mathbb{R}^{n-1}$.

(iii) Für g gilt: $g(\phi_S(x)) = g(\phi_S(\Psi(x)))\quad \forall x \in \mathbb{R}^{n-1}$.

Darüberhinaus ist f_S auf W beschränkt und Lipschitz-stetig, desgleichen ist $g(\phi_S(\cdot))$ auf W Lipschitz-stetig.

Beweis: (i) folgt aus der Tatsache, daß $p(\cdot) = \phi_S(\varphi(\cdot, x_0))$ mit $p = \phi_S(x_0)$ gilt. Da $p(\cdot)$ eindeutig existiert, muß dies auch für $\varphi(\cdot, x_0)$ gelten.
Durch Nachrechnen sieht man, daß

$$\Psi(x) = \phi_S^{-1}(-\phi_S(x)) \quad \forall x \in \mathbb{R}^{n-1} \setminus W.$$

Wegen der Identifikation gegenüberliegender Punkte auf der Sphäre gilt $(-p)(t) = -(p(t))$ für beliebige Punkte $p \in \mathbb{S}^{n-1}$, also folgt mit $p = \phi_S(x_0)$

$$\Psi(\varphi(t, x_0)) = \phi_S^{-1}(-p(t)) = \phi_S^{-1}((-p)(t)) = \phi_S^{-1}(-\phi_S(\varphi(t, \phi_S^{-1}(-p)))) = \Psi(\varphi(t, \Psi(x_0)),$$

also gilt (ii).
(iii) gilt mit $p = \phi_S(x)$ wegen

$$g(\phi_S(x)) = g(p) = g(-p) = g(-\phi_S(x)) = g(\phi_S(\Psi(x))).$$

Die Lipschitz-Stetigkeit und Beschränktheit von f_S auf W ist nach Voraussetzung klar, die Lipschitz-Stetigkeit von $g(\phi_S(\cdot))$ folgt aus der Existenz der Konstanten β. □

Bemerkung A.5 Eine weitere Eigenschaft der Funktion Ψ ist, daß $\Psi(x) \in D_0\ \forall x \in \mathbb{R}^{n-1} \setminus W$ (D_0 ist die offene Scheibe um 0 mit Radius 1). Zusammen mit der Wahl von W garantiert uns dies nach Lemma A.3, daß alle Trajektorien (exakte sowie diskretisierte), die in einem Punkt $\Psi(x)$ mit x außerhalb von W starten, mindestens für die Zeit T in W bleiben.
Ψ kann also als eine Funktion aufgefaßt werden, die in lokalen Koordinaten einen Wechsel der Parametrisierung vornimmt, so daß danach Bedingung (A.4) erfüllt ist.

Mit der Funktion Ψ definieren wir nun lokale Trajektorien (exakte sowie diskretisierte), die äquivalent zu denen im projektiven Raum sind.

Definition A.6 Es seien nach Lemma A.3 eine Menge $W \subset \mathbb{R}^{n-1}$ und eine Lipschitzstetige und beschränkte Funktion $f : W \to \mathbb{R}^{n-1}$ gegeben, die durch ein Vektorfeld auf $\mathbb{P}^{n-1}$ definiert ist. Es sei $\hat{\varphi}(t, x_0)$ die Trajektorie zum Anfangswert $x_0 \in W$. Dann definieren wir mittels der Funktion Ψ aus (A.10) zu $x \in \mathbb{R}^{n-1}$ induktiv

$$\begin{aligned} \varphi(t, x, u) &:= \hat{\varphi}(t, \Psi(x)) \ \forall 0 \le t \le T_1 \text{ mit } \hat{\varphi}(t, \Psi(x)) \in \overline{W} \\ \varphi(T_i + t, x) &:= \hat{\varphi}(t, \Psi(\varphi(T_i, x))) \ \forall 0 \le t \le T_{i+1} - T_i \text{ mit } \hat{\varphi}(t, \Psi(\varphi(T_i, x))) \in \overline{W} \end{aligned} \tag{A.11}$$

und analog zu $h > 0$ die diskretisierte Trajektorie

$$x_0 = \Psi(x), \quad x_{i+1} = \Psi(x_i + h f(x_i)) \tag{A.12}$$

für $i = 1, 2, \ldots$

A.2 Stetige Abhängigkeit und Diskretisierung

In diesem Abschnitt werden wir zeigen, daß für die exakten und diskretisierten Trajektorien stetige Abhängigkeit vom Anfangswert gilt. Desweiteren zeigt sich, daß wir Differentialgleichungen, die durch ein Vektorfeld auf $\mathbb{P}^{n-1}$ bzw. $\mathbb{S}^{n-1}$ gegeben sind, approximativ lösen können, indem wir das Euler-Verfahren in lokalen Koordinaten anwenden. Für dieses Verfahren können wir dann den Diskretisierungsfehler abschätzen.

Lemma A.7 (Gronwall-Lemma)
Sei $I \subset \mathbb{R}$ ein Intervall, $\psi : I \to \mathbb{R}^+$ eine stetige Funktion, die für positive Konstanten α, $\beta \in \mathbb{R}$, $t_0 \in I$ und alle $t \in I$ die Abschätzung

$$\psi(t) \le \alpha + \beta \int_{t_0}^{t} \psi(s) ds \tag{A.13}$$

erfüllt. Dann gilt für alle $t \in I$ die Ungleichung

$$\psi(t) \le \alpha e^{\beta |t - t_0|}. \tag{A.14}$$

Setzt man in (A.13) die strikte Ungleichung voraus, so ergibt sich auch in (A.14) die strikte Ungleichung.

Beweis: Setze

$$\phi(t) := \int_{t_0}^{t} \psi(s) ds.$$

Damit schreibt sich (A.13) als $\dot{\phi}(t) \le \alpha + \beta \phi(t)$. Setze nun $\theta(t) := \phi(t) e^{-\beta t}$. Durch Differenzieren von θ erhält man $\dot{\theta}(t) \le \alpha e^{-\beta t}$. Integration dieser Ungleichung (die Funktionen sind nichtnegativ wegen (A.13) und es gilt $\theta(t_0) = 0$) liefert

$$\theta(t) \le \frac{\alpha}{\beta} |e^{-\beta t} - e^{-\beta t_0}|.$$

Damit ergibt sich für ϕ

$$\phi(t) \leq \frac{\alpha}{\beta} e^{\beta|t-t_0|} - \frac{\alpha}{\beta}.$$

Für ψ erhält man daraus

$$\psi(t) \leq \alpha + \beta\phi(t) \leq \alpha + \alpha e^{\beta|t-t_0|} - \alpha.$$

Eine strikte Ungleichung in (A.13) führt zu einer strikten Ungleichung in jeder dieser Abschätzungen. □

Lemma A.8 Seien eine Mannigfaltigkeit M und ein Vektorfeld X gegeben, die (A.2) – (A.5) erfüllen.
Dann gilt für beliebige Punkte p, $q \in M$ und für alle $t \geq 0$:

$$d(p(t), q(t)) \leq d(p, q) e^{((\frac{\beta}{\alpha} - 1)\frac{1}{T} + L_f)t}$$

Beweis: Wir nehmen zunächst an, daß es ein $i \in I$ gibt mit $p \leftrightarrow \phi_i$ und $q \leftrightarrow \phi_i$. Dann folgt mit dem Gronwall-Lemma A.7 $\forall 0 \leq t \leq T$:

$$d(p(t), q(t)) \leq \beta \|\phi_i^{-1}(p(t)) - \phi_i^{-1}(q(t))\| \leq \beta e^{L_f t} \|\phi_i^{-1}(p) - \phi_i^{-1}(q)\| \leq \frac{\beta}{\alpha} d(p, q) e^{L_f t} \quad \text{(A.15)}$$

Wenn p und q nun zu verschiedenen Parametrisierungen gehören, so können wir nach (A.5) Zwischenpunkte r_0 bis r_k auf der Geodätischen finden, die die kürzeste Verbindung zwischen p und q darstellt. Wir betrachten nun die Trajektorien, die von diesen Zwischenpunkten ausgehen. Für jeweils zwei dieser Trajektorien können wir (A.15) anwenden, dies führt zu

$$\begin{aligned} d(p(t), q(t)) &= \sum_{j=1}^{k} d(r_{j-1}(t), r_j(t)) \leq \sum_{j=1}^{k} \frac{\beta}{\alpha} e^{L_f t} d(r_{j-1}, r_j) \\ &= \frac{\beta}{\alpha} e^{L_f t} \sum_{j=1}^{k} d(r_{j-1}, r_j) = \frac{\beta}{\alpha} e^{L_f t} d(p, q) \end{aligned} \quad \text{(A.16)}$$

für alle $0 \leq t \leq T$. Per Induktion ergibt sich nun für beliebige $t > 0$

$$d(p(t), q(t)) \leq \left(\frac{\beta}{\alpha}\right) \frac{t}{T} e^{L_f t} d(p, q).$$

Aus $e^C \geq (1 + C)$ folgt $e^{(C-1)} \geq C$ und auch $e^{(C-1)j} \geq C^j$ für beliebige $C \in \mathbb{R}$. Damit folgt

$$d(p(t), q(t)) \leq e^{(\frac{\beta}{\alpha} - 1)\frac{t}{T}} e^{L_f t} d(p, q),$$

also die Behauptung. □

Lemma A.9 Seien eine Mannigfaltigkeit M und ein Vektorfeld X gegeben, die (A.2) – (A.5) erfüllen.
Dann gilt für beliebige Punkte p, $q \in M$ und für alle $t \geq 0$:

$$d(p_h(t), q_h(t)) \leq d(p, q) e^{((\frac{\beta}{\alpha} - 1)\frac{1}{T} + L_f)t}$$

Beweis: Wir können vereinfachend annehmen, daß T ein Vielfaches von h ist. Für beliebige Punkte x, y in U_i gilt außerdem (solange die Trajektorien in U_i bleiben):

$$\|\hat{\varphi}_h(t,x) - \hat{\varphi}_h(t,y))\| \leq e^{L_f t}\|x-y\|, \tag{A.17}$$

was durch Induktion über $j = [\frac{t}{h}]$ gezeigt wird:
Für $j = 0$ ist (A.17) klar.
Für $j \to j+1$ gilt:

$$\begin{aligned}
&\|\hat{\varphi}_h((j+1)h,x) - \hat{\varphi}_h((j+1)h,y)\| \\
&\quad = \|x_j + hf(x_j) - y_j - h(f(y_j)\| \\
&\quad \leq \|x_j - y_j\| + h\|f(x_j) - f(y_j)\| \\
&\quad \overset{I.V.}{\leq} e^{L_f h_j}\|x-y\| + hL_f e^{L_f jh}\|x-y\| \\
&\quad = (1+hL_f)e^{L_f jh}\|x-y\| \;\leq\; e^{L_f h}e^{L_f jh}\|x-y\| \;=\; e^{L_f(j+1)h}\|x-y\|.
\end{aligned}$$

Wir nehmen nun wieder an, daß es ein $i \in I$ gibt mit $p \leftrightarrow \phi_i$ und $q \leftrightarrow \phi_i$. Dann folgt aus (A.17)

$$d(p_h(T), q_h(T)) \leq \frac{\beta}{\alpha}e^{L_f T}d(p,q).$$

Nun können wir die Aussage analog zum vorhergehenden Beweis auf beliebige p, $q \in M$ erweitern, und die Behauptung per Induktion zeigen. □

Lemma A.10 Seien eine Mannigfaltigkeit M und ein Vektorfeld X gegeben, die (A.2) – (A.5) erfüllen.
Dann gilt für beliebige $p \in M$ und für alle $t \geq 0$:

$$d(p(t), p_h(t)) \leq hM_f\frac{\beta}{\alpha}e^{((\frac{\beta}{\alpha})\frac{1}{T}+L_f)t}$$

Beweis: Wir nehmen wieder vereinfachend an, daß T Vielfaches von h ist. Wenn wir das Gronwall-Lemma A.7 auf die lokalen Trajektorien anwenden, erhalten wir

$$d(p(t), p_h(t)) \leq \frac{\beta}{\alpha}M_f h e^{L_f t} \quad \forall 0 \leq t \leq T.$$

Unter Verwendung von Abschätzung (A.16) erhalten wir daraus für beliebige p, $q \in M$

$$\begin{aligned}
d(p(t), q_h(t)) &\leq d(p(t), q(t)) + d(q(t), q_h(t)) \\
&\leq \frac{\beta}{\alpha}e^{L_f t}\Big(d(p,q) + hM_f\Big) \quad \forall 0 \leq t \leq T.
\end{aligned} \tag{A.18}$$

Wir zeigen nun zunächst per Induktion für $j \geq 0$ die Zwischenbehauptung

$$d(p(jT), p_h(jT)) \leq jhM_f e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)jT}. \tag{A.19}$$

Für $j = 0$ ist die Behauptung klar, für $j \to j+1$ gilt

$$\begin{aligned} d(p((j+1)T), p_h((j+1)T)) &\overset{(A.18)}{\leq} \frac{\beta}{\alpha} e^{L_f T}\Big(d(p(jT), p_h(jT)) + hMf\Big) \\ &\overset{I.V.}{\leq} \frac{\beta}{\alpha} e^{L_f T}\Big(jhM_f e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)jT} + hMf\Big) \\ &\leq e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)T}\Big(jhM_f e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)jT} + hMf\Big) \\ &\leq jhM_f e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)(j+1)T} + hM_f e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)T} \\ &\leq (j+1)hM_f e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)(j+1)T} \end{aligned}$$

und damit ist (A.19) bewiesen.
Um eine Abschätzung für alle $t \geq 0$ zu erhalten, setzen wir $t = jT + \tilde{t}$ mit $0 \leq \tilde{t} < T$ und wenden (A.18) an:

$$\begin{aligned} d(p(t), p_h(t)) &\leq \frac{\beta}{\alpha} e^{L_f \tilde{t}}\Big(d(p(jT), q(jT)) + hM_f\Big) \\ &\leq \frac{\beta}{\alpha} e^{L_f \tilde{t}}\Big(jhM_f e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)jT} + hM_f\Big) \\ &\leq \frac{\beta}{\alpha} hM_f\Big(je^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)t} + e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)\tilde{t}}\Big) \\ &\leq \frac{\beta}{\alpha} hM_f\Big((j+1)e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)t}\Big) \end{aligned}$$

Unter Ausnutzung von $j+1 \leq e^j$ und $j \leq \frac{t}{T}$ erhalten wir hieraus

$$d(p(t), p_h(t)) \leq hM_f \frac{\beta}{\alpha} e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f)t+j} \leq hM_f \frac{\beta}{\alpha} e^{((\frac{\beta}{\alpha}-1)\frac{1}{T}+L_f+\frac{1}{T})t}$$

und damit die Behauptung.

Bemerkung A.11 Wenn die Mannigfaltigkeit so parametrisiert ist, daß $\alpha = \beta = 1$ gilt, verbessern sich die Abschätzungen.
Bei der Sphäre $\mathbb{S}^1$ ist dies möglich durch die Parametrisierung mit Polarkoordinaten. Die Funktion Ψ entspricht dann einer Periodizitätsbedingung $\Psi(x) = x - \pi$. Mit dieser Parametrisierung wird das Vektorfeld beim Parameterwechsel nicht verzerrt, so daß für die diskretisierte Trajektorie $\Psi(\Psi(x) + hf(\Psi(x)) = \Psi(x + hf(x))$ gilt. Deshalb entsteht durch den Parameterwechsel im Euler-Verfahren kein weiterer Fehler. Im Beweis zu Lemma A.10 kann der Abstand der Trajektorien dann durch eine Integralungleichung für alle $t \geq 0$ abgeschätzt werden, auf die dann das Gronwall-Lemma angewendet werden kann. Dadurch wird der Term $1/T$ im Exponenten vermieden.

Anhang B

Implementierung des Algorithmus

Die beiden Algorithmen zur Berechnung der optimalen Wertefunktion und der ε-optimalen Kontrollen wurden auf zwei einzelne Programme aufgeteilt. Diese bauen auf der Implementierung des beschleunigten Algorithmus aus Abschnitt 5.3.2 auf, die von Uwe Sorgenfrei im Rahmen seiner Diplomarbeit [16] erstellt wurde.

Das Modul *findsimp*, das die Verwaltung der Triangulation enthält, wurde direkt übernommen. In das Modul *orbit*, das die Berechnung der ε-optimalen Trajektorien und Kontrollen vornimmt, wurde lediglich die Funktion Ψ eingefügt, die für die lokale Berechnung der Trajektorien im Projektiven Raum nötig ist. Das Modul *iterate* enthält die komplette Iterationsroutine des Koordinatenaufstiegsverfahrens und ist demnach neu geschrieben worden. Die anderen Module dienen zur Unterstützung: *inout* enthält die Ein- und Ausgaberoutinen, *lu_solve* enthält eine Routine zur Lösung von Linearen Gleichungssystemen, die zur Berechnung der λ_{ij} benötigt wird. Die rechten Seiten der Kontrollsysteme sowie die Kostenfunktionen sind im Modul *equation* implementiert, das Modul *equ_orb* schließlich enthält weitere Funktionen, die im Modul *orbit* zur Verwaltung des Projektiven Raums benötigt werden.

Bemerkung B.1 Die im Programm verwendeten Variablen- und Konstantenbezeichnungen weichen zum Teil von den Bezeichnungen ab, die in dieser Arbeit verwendet werden. Im Modul *iterate.c* sind die Bezeichnungen im Programm ausführlich erläutert.

B.1 Verwaltung der Triangulation

Wie oben bereits erwähnt, wurde dieser Programmteil direkt aus [16] übernommen. Zum Verständnis des Programms soll er hier aber trotzdem noch kurz beschrieben werden.

Für die Triangulation werden sowohl die Knotenpunkte als auch die Simplizes in jeweils einem Feld gespeichert. Die Größe dieser Felder ist gegeben durch die Anzahl der Knotenpunkte N, die Anzahl der Simplizes P und die Dimension des Systems n. Die Raumschrittweite k ergibt sich dann als der maximale Abstand zweier Knoten, die zu einem Dreieck gehören. Mit $\overline{\Omega^k}$ bezeichnen wir das gesamte triangulierte Gebiet.

Wir müssen nun zu einem beliebigem Punkt $x = (x_1, \ldots, x_n)^t$ den Index j des zugehörigen Simplizes S_j zu finden, und danach seine baryzentrischen Koordinaten $\lambda_{j_1}, \ldots, \lambda_{j_{n+1}}$ berechnen. Dies geschieht wie folgt: Wenn $x_{j_1}, \ldots, x_{j_{n+1}} \in \mathbb{R}^n$ die Eckpunkte des Simplizes S_j bezeichnen, läßt sich das Gleichungssystem

$$\begin{pmatrix} x_{j_1} & \cdots & x_{j_{n+1}} \\ 1 & \cdots & 1 \end{pmatrix} (\lambda_{j_1}, \ldots, \lambda_{j_{n+1}})^t = (x_1, \ldots, x_n, 1)^t \tag{B.1}$$

eindeutig lösen, da die Regularität der Koeffizientenmatrix direkt aus der Eigenschaft folgt, daß die x_{j_i} Eckpunkte eines Simplizes sind. Dieses Gleichungssystem wird mit einer einfachen LU-Zerlegung nach Gauß im Modul *lu_solve* gelöst.

Wenn dieses Gleichungssystem für alle Simplizes gelöst wird, ist der gesuchte derjenige, bei dem $\lambda_{j_i} \geq 0$ gilt für alle $i = 1, \ldots, n+1$. Der Aufwand, alle Simplizes zu „durchsuchen“, ist jedoch sehr hoch. Wir können deshalb ausnutzen, daß die Eckpunkte des Simplizes, der x enthält, höchstens den Abstand k von x haben können. Deshalb besteht die Suchstrategie aus drei Abschnitten:
Im ersten werden alle Knoten durchlaufen und diejenigen markiert, die einen Abstand kleiner oder gleich k von x haben.
Im zweiten werden alle Simplizes markiert, die einen der markierten Knoten als Eckpunkt haben.
Im dritten wird nun für diese Simplizes das Gleichungssystem (B.1) gelöst, und getestet, ob $\lambda_{j_i} \geq 0$ gilt.

Das Markieren geschieht hierbei durch Aufnehmen der Indizes in lineare Listen, da so eine dynamische Speicherverwaltung verwendet werden kann.

Alle diese Schritte werden in der Funktion *seek_simplex* erledigt. Der Rückgabewert dieser Funktion gibt an, ob ein Simplex gefunden wurde; als Zeigervariablen werden die λ_{j_i} und der Index des gefundenen Simplizes übergeben.

B.2 Berechnung der Wertefunktion

Die Routinen zur Berechnung der Wertefunktion v_h^k befinden sich im Modul *iterate*. In diesem wird also die in Abschnitt (5.3.3) beschriebene Iteration

$$V_{j+1} = V_j, \quad [V_{j+1}]_i = \min_{u \in U} \frac{\beta \sum_{\substack{j=1,\ldots,N \\ j \neq i}} \lambda_{ij}(u)[V_{j+1}]_j + hG_i(u)}{1 - \beta\lambda_{ii}(u)} \quad i = 1, \ldots, N \tag{B.2}$$

mit einem Startvektor $V_0 \in \mathcal{V}$ durchgeführt.

Zu diesem Zweck dienen drei Funktionen: *prepare* berechnet zu jedem $u \in U$ die Matrix $\Lambda(u) \in \mathbb{R}^{N \times N}$ mit den Einträgen λ_{ij} sowie den Vektor $G(u)$, *iterate* führt die Iteration durch und stützt sich dabei auf die Funktion *det_i*, die den Bruch in (B.2) zu vorgegebenem V_{j+1} und i berechnet.

Da jedem Punkt $x \in \overline{\Omega^k}$ von der Routine *seek_simplex* genau ein Simplex zugeordnet wird, ergibt sich eine Zeile von $\Lambda(u)$ zu

$$[\Lambda(u)]_i = (\ldots \lambda_{ij_1} \ldots \lambda_{ij_2} \ldots\ldots \lambda_{ij_{n+1}} \ldots). \tag{B.3}$$

Alle restlichen Einträge sind 0. Statt nun die gesamte Matrix zu speichern, genügt es für jede Zeile von $\Lambda(u)$ den Index j des Simplizes und die $n+1$ Nicht-Null Einträge zu speichern. Über den Index j lassen sich dann aus dem Simplex-Feld die zugehörigen Spaltenindizes $j_1, \ldots, j_{n+1}$ entnehmen. Statt N^2 Dezimalwerten müssen so nur N Integerwerte und $N(n+1)$ Dezimalwerte gespeichert werden.

Die Routine *prepare* funktioniert nun wie folgt:
Zu jedem Knotenpunkt x_i wird zu jedem Kontrollwert $u \in U$ ein Euler-Schritt durchgeführt und – falls nötig – die Funktion Ψ auf den erreichten Punkt angewendet. Mittels der Funktion *seek_simplex* werden dann der zugehörige Simplex und die λ_{ij} berechnet und in der beschriebenen Weise in der Matrix $\Lambda(u)$ abgelegt. $G(u)$ wird durch einfaches Auswerten der Kostenfunktion in den Knotenpunkten ermittelt. Die Funktion liefert in einem Diagnoseflag die Anzahl der Knotenpunkte zurück, für die kein Simplex gefunden wurde.

Die Funktion *iterate* steuert die eigentliche Iteration:
Ausgehend von einem Startvektor V wird die Routine *det_i* für alle $i = 1, \ldots, N$ aufgerufen, die das Minimum der Brüche aus(B.2) über alle $u \in U$ berechnet. Der neue Vektor wird mit dem alten verglichen, und solange die erreichte Verbesserung über einem vorgegebenem Wert δ_{iter} liegt, wird mit demselben Verfahren weitergemacht. Die Iteration bricht ebenfalls ab, wenn eine maximale Iterationsanzahl I_{max} überschritten wird.

Für bestimmte Kontrollsysteme genügt *optimale Invarianz* des triangulierten Gebietes (vgl. [16], Kapitel 3). Deshalb ist im Programm die Möglichkeit vorgesehen, daß nicht zu allen $u \in U$ ein Simplex gefunden wird. In diesem Fall setzt *prepare* den entsprechenden Simplex-Index auf -1 und gibt eine Warnung aus. Diese Kontrollwerte werden dann in *det_i* ignoriert.
Das Verfahren bricht ab, falls es Knotenpunkte gibt, bei denen für *jedes* $u \in U$ die Invarianzbedingung verletzt ist.

B.3 Berechnung der Orbits und Kontrollen

Die Berechnung der ε-optimalen Kontrollen und Orbits findet in der Funktion *calc_orbit* des Moduls *orbit* statt. Diese Funktion stützt sich auf die Funktion *calc_next*, in der die eigentlichen Berechnung durchgeführt wird.

In *calc_next* wird einem gegebenen Punkt x_j zu jedem Kontrollwert $u \in U$ per Euler-Schritt und eventueller Anwendung der Funktion Ψ der Punkt x^u_{j+1} berechnet. Nach dem in Abschnitt 5.4.1 beschriebenen Algorithmus wird nun der optimale Kontrollwert bestimmt, wobei zur Bestimmung der Werte von v^k_h in den Zwischenpunkten die Funktion *seek_simplex* benötigt wird. Für die nötige Interpolation werden die von *seek_simplex* gelieferten λ_{ij} verwendet.

Für die Berechnung der ε-optimalen Trajektorien ruft *calc_orbit* diese Routine iterativ auf und bricht nach einer vorgegebenen Anzahl von Zeitschritten I_{orb} ab.

Die hier abgedruckte Version der Funktion *calc_orbit* wurde etwas abgewandelt, um die Berechnung der Trajektorien der ursprünglichen Systeme, die stabilisiert werden sollten, zu ermöglichen. Zusätzlich zur Trajektorie in $\mathbb{P}^{n-1}$, die ja durch *calc_next* berechnet wird,

wird hier auch die Trajektorie im $\mathbb{R}^n$ durch das Euler-Verfahren berechnet. Die rechte Seite hierzu wird im Modul *equ_orb* bereitgestellt, dort befinden sich auch die Funktionen zur Projektion auf die Sphäre und deren Umkehrung. Der jeweils nächste Euler-Schritt wird mit dem berechneten optimalen Kontrollwert durchgeführt. Da das Euler-Verfahren auf $\mathbb{P}^{n-1}$ und im $\mathbb{R}^n$ i.A. nicht die gleichen Werte liefern, wird hier nach jedem Zeitschritt mit der Projektion des Wertes im $\mathbb{R}^n$ fortgefahren, was dem Verfahren aus Definition 6.12 für die exakte Trajektorie entspricht. Die ursprüngliche *calc_orbit* Routine erhält man einfach durch Weglassen der Zuweisung

```
x[j]=y2[j]
```

Die nachfolgende Anwendung der Funktion Ψ kann dann auch weggelassen werden, da dies bereits in *calc_next* geschieht.

Anhang C

Bedienung der Programme

In diesem Anhang wird die Bedienung der Programme beschrieben.

Grundsätzlich sind die Programme so ausgelegt, daß die Eingabe der Werte von *stdin* und die Ausgabe auf *stdout* erfolgt. Das bedeutet, daß Ein- und Ausgabedateien den Programmen mit den Operatoren „>“ und „<“ zugewiesen werden müssen (Beispiele folgen in den nächsten Abschnitten). Parameter der Programmaufrufe sind lediglich die Indexnummer der Systemgleichung und für die Orbitberechnung der Startpunkt der Trajektorie.

In den folgenden Abschnitten ist die Bedienung sowohl unter UNIX als auch unter MS-DOS beschrieben.

C.1 Dateneingabe

Eine Eingabedatei besteht aus fünf bzw. sechs Teilen:
Der Kontrollzeile, der Spezifikation der Funktion Ψ, der Liste der Simplizes, der Liste der Kontrollwerte und einer Liste von Systemparametern. Für die Orbitberechnung kommt noch der Ergebnisvektor des Iterationsverfahrens dazu.
Achtung: Kommentare sind in der Eingabedatei nicht möglich.

In der Kontrollzeile werden die Werte h, ρ, I_{max}, δ_{iter}, δ_{dummy}, δ_{dummy}, I_{orb} festgelegt. Die beiden $delta_{dummy}$-Werte werden für den Koordinatenaufstiegsalgorithmus nicht benötigt, sie dienen hier nur als Platzhalter für Parameter, die im beschleunigten Algorithmus benötigt werden, um weitestgehende Kompatibilität der Eingabedateien sicherzustellen.

In der zweiten Zeile der Eingabedatei wird die Funktion Ψ festgelegt. Bei einem positiven Wert wird Ψ als Periodizitätsbedingung festgelegt, wobei die Perioden in den Komponenten gerade gleich dem angegebenen Wert sind. Wird dieser Wert auf 0 gesetzt, so wird Ψ als die Identität festgelegt, also davon ausgegangen, daß Ω^k invariant ist. Für beliebige negative Werte wird die Funktion (A.10) gewählt, also die Stereographische Projektion. $\overline{\Omega^k}$ ist dann intern festgelegt als $[-1,1] \times [-1,1]$.

Die dritte Zeile enthält die Dimension des Systems und die Anzahl N der Knotenpunkte. Nach dieser Zeile folgt dann die Liste der Knotenpunkte, im Anschluß daran die Anzahl

der Simplizes und deren Liste. Danach folgt die Dimension und Anzahl der Kontrollwerte und deren Liste.

Nach diesen Angaben können dann noch drei Systemparameter festgelegt werden, auf die im Programm über die Variablen *a_d, b_d, c_d* zugegriffen werden kann.

Falls die Eingabedatei für die Orbitberechnung vorgesehen ist, muß nun noch der Vektor der Wertefunktion in den Knotenpunkten angehängt werden; diese Liste wird vom Iterationsverfahren geliefert.

Die folgende Mustereingabedatei kann als Beispiel für den Aufbau der Eingabedaten dienen.

0.01 2.0 50 1.0e-6 1.0e-3 1.0e-6 100	*Kontrollzeile*
0.0	*Spezifikation von* Ψ
2 25	*Dimension und Anzahl der Knotenpunkte*
0 0.0 0.0	*Liste der Knotenpunkte*
1 0.0 0.25	
⋮	
24 1.0 1.0	
32	*Liste der Simplizes*
0 0 5 6	
⋮	
31 19 20 24	
1 2	*Liste der Kontrollwerte*
0 0.0	
1 1.0	
1.0	*Systemparameter*
1.5	
-0.5	
0 0.000000	*Ergebnisvektor der Iteration*
⋮	
24 0.335335	

C.2 Das Iterationsverfahren

Für das Beispiel nehmen wir an, daß wir die Wertefunktion für das System mit der Nummer 0, das System *sample* berechnen wollen, und daß eine Eingabedatei in der obigen Form (ohne Wertefunktionsvektor) unter dem Namen *sample.dat* vorliegt. Dann sieht ein Aufruf des Iterationsverfahrens wie folgt aus:

mkn 0 < sample.dat > sample.out

Der Index 0 steht hierbei für die Nummer des Kontrollsystems. Bei erfolgreicher Durchführung der Iteration wird der Ergebnisvektor dann in die Datei mit dem Namen *sample.out* geschrieben. Sämtliche Fehlermeldungen werden auf *stderr*, im Normalfall also auf den Bildschirm geschrieben. Auch die fortlaufenden Iterationen werden dort ausgegeben. Auf

UNIX-Systemen gibt es die Möglichkeit, das Programm im Hintergrund laufen zu lassen und die laufenden Ausgaben z.B. in die Datei *sample.log* zu schreiben. Der Aufruf lautet dann

```
mkn 0 < sample.dat > sample.out 2>>sample.log &
```

In der Datei *sample.log* kann dann jederzeit nachgesehen werden, wie weit die Berechnung fortgeschritten ist.

C.3 Die Orbitberechnung

Für die Orbitberechnung muß das Ergebnis des Iterationsverfahrens an die alte Eingabedatei angehängt werden. Das kann unter UNIX geschehen mit

```
cp sample.dat sample.orb
cat sample.out >> sample.orb
```

bzw. unter MS-DOS mit

```
copy sample.dat sample.orb
type sample.out >> sample.orb
```

Die Datei *sample.orb* ist nun die neue Eingabedatei für das Programm *mko* zur Orbitberechnung. Der Programmaufruf benötigt nun wieder den Index des gewünschten Systems und zusätzlich die Angabe des Startpunktes für die Orbitberechnung. Mit dem Startpunkt $x = (0.0, 1.0)^t$ lautet der Aufruf also

```
mko 0 0.0 1.0 < sample.orb > sample.tra
```

bzw.

```
mko 0 0.0 1.0 < sample.orb > sample.tra 2>>sample.log &
```

wenn unter UNIX im Hintergrund gerechnet werden soll und die bestehende Protokolldatei *sample.log* fortgeführt werden soll.

In der abgedruckten Version wird die Trajektorie im $\mathbb{R}^n$ in die Datei *sample.tra* geschrieben und die Trajektorie im projektiven Raum zusammen mit der optimalen Kontrolle auf *stderr*, also bei Verwendung des zweiten Aufrufs in *sample.log*. Dies kann leicht durch die Änderung der entsprechenden *fprintf*-Anweisungen umgestellt werden.

C.4 Hinzufügen neuer Kontrollsysteme

Um weitere Kontrollsysteme hinzuzunehmen, müssen die Module *equation* und *equ_orb* geändert werden. Dies geschieht wie folgt

(i) Die Prototypen der Systemgleichung und der Kostenfunktion müssen in die Datei *equation.h* eingetragen werden. Die Typen müssen dabei gemäß der *typedef*-Anweisung im Modul *defs* gewählt werden.

(ii) Die Namen der Funktionen müssen in die Funktionenarrays am Anfang des Module *equation.c* eingetragen werden. Durch die Lage in diesen Arrays bestimmt sich der Index, den das System erhält, wobei die Zählung nach C-Konvention mit 0 beginnt.

(iii) Nun kann unten im Modul *equation.c* die Implementierung der neuen Funktionen vorgenommen werden.

(iv) Falls das System eine Projektion eines Systems im $\mathbb{R}^n$ auf die Sphäre ist und auch die Trajektorien des ursprünglichen Systems berechnet werden sollen, müssen die beschriebenen Änderungen ebenfalls in *equ_orb.h* und *equ_orb.c* vorgenommen werden, falls dies nicht der Fall ist, genügt es, in den Funktionenarrays an Anfang des Moduls *equ_orb.c* die bereits definierten Dummy-Funktionen an die entsprechenden Stellen einzutragen.

(v) Nach einer Neukompilierung sind die Programme, wenn keine Fehler gemacht wurden, wieder lauffähig. Für Compiler, die das Makefile-Konzept unterstützen, sind die entsprechenden Makefiles im Anhang abgedruckt.

Anhang D

Programmtext

Auf den folgenden Seiten ist der Programmtext der Module abgedruckt, die gegenüber dem Programm von Uwe Sorgenfrei [16] abgeändert wurden bzw. völlig neu geschrieben wurden. Sie können mit den ebenfalls abgedruckten Makefiles mit den in [16] abgedruckten Modulen zusammengebunden werden.

```
/********************************************** (c) Lars Gruene 1993


HEADER : iterate.h
PURPOSE: This module provides two functions, "iterate" and "prepare", for
     common use.
     Given all characteristic data and a start vector this function
     iterates the values of the (optimal) value function at the node
     points of the given triangulation.
     "iterate" calls the function "det_i", which determines the next step
     in the i-th component of the vector.
     To obtain all characteristic data there is the function
     "prepare", which sets up the lambda-matrices and the cost-vectors
     used in the iteration procedure.


*****************************************************************************/

#ifndef _ITERATE_H
  #define _ITERATE_H

  int iterate( void );
  void prepare( RHS, CFN, int* );

#endif
```

```
/************************************************** (c) Lars Gruene 1993

   HEADER : equation.h
   PURPOSE: Implementation of the equations for the control systems


**************************************************************************/

real testg( real *, real * );
real phi_g(real*, real*);
real osz3dp_g(real*, real*);


void testf( real *, real *, real * );
void phi_f( real *, real *, real * );
void osz3dp_f( real *, real *, real * );
```

```
/********************************************************* (c) Lars Gruene 1993

HEADER : equ_orb.c
PURPOSE: Additional equations for dealing with the projective space

*****************************************************************************/

void dummy1(real *, real *, real *);
void dummy2(real *, real *);

void osz2_f(real *, real *, real *);
void osz2_p(real *, real *);
void osz2_o(real *, real *);
void osz2_i(real *, real *);

void osz3_f(real *, real *, real *);
void osz3_p(real *, real *);
void osz3_o(real *, real *);
void osz3_i(real *, real *);
```

```
/****************************** (c) Uwe Sorgenfrei 1992, Lars Gruene 1993


HEADER : orbit.h
PURPOSE: Calculation of optimal orbits.

*****************************************************************************/

#ifndef _ORBIT_H
  #define _ORBIT_H

  #ifndef __STDIO_H
    #include <stdio.h>
  #endif

  #ifndef __STDLIB_H
    #include <stdlib.h>
  #endif

  #ifndef __MATH_H
    #include <math.h>
  #endif

  #ifndef __TIME_H
    #include <time.h>
  #endif

  #include "equ_orb.h"

  real calc_orbit( real *x, RHS f, CFN g, RHS f2, PRO p, PRO o, PRO i);

#endif
```

```
/******************************** (c) Uwe Sorgenfrei 1992, Lars Gruene 1993


HEADER : inout.h
PURPOSE: Loading and checking input data.


*****************************************************************************/

#define FACTOR 1.1

void read_data( void );
int get_nodes( void );

real det_timst( RHS f );

real det_spatial( void );

real det_iterstart( CFN g );
```

```
/************************************* (c) Sorgenfrei Uwe, December 1992

STANDARD DEFINITIONS FILE defs

*************************************************************************/

typedef
  double
  real;


typedef real (*CFN)( real*, real* ); /* cost function */
typedef void (*RHS)( real*, real*, real* ); /* system equation */
typedef void (*PRO)( real*, real* );



#define FOR( var, start, end ) \
  for( (var) = (start); (var) < (end); (var)++ )
#define DOWNTO( var, start, end ) \
  for( (var) = (start)-1; (var) ≥ (end); (var)-- )
#define SWAP( tmp, x, y ) \
  ( (tmp) = (x), (x) = (y), (y) = (tmp) )
#define INRANGE( x, y, z ) \
  ( (x) ≥ (y) && (x) ≤ (z) )
#define ABS( x ) \
  ((x) < 0 ? -(x) : (x))
#define MIN( x, y ) \
  ((x) < (y) ? (x) : (y))
#define MAX( x, y ) \
  ((x) < (y) ? (y) : (x))


#define B_D ((real) 1.0)


#ifndef __ALLOC
  #ifdef __MSDOS__
    #include <alloc.h>
  #else
    #include <malloc.h>
  #endif
#endif
#define DOALLOC( nitems, type ) \
  ( (type*) malloc( (size_t) (nitems)*sizeof( type ) ) )
```

```
#define UNALLOC( ptr ) \
  free( (void*) (ptr) )
#define ALLOC_FAILED( ptr ) \
  !((ptr))
```

```
/********************************************** (c) Lars Gruene 1993

   MODULE : iterate.c
   PURPOSE: Implementation of the iteration

**********************************************************************/

#undef DEBUG



#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <math.h>
#include <string.h>



#include "defs"
#include "iterate.h"
#include "findsimp.h"


int
   **simplices; /* field for simplices */
real
   **vertices, /* field for vertices */
   **controlvals; /* field for control values */
real
   lambda, /* discount factor */
   time_step, /* time stepsize */
   spatial, /* space step size */
   per; /* flag for periods */
int
   max_vertex, /* number of vertices */
   max_simplex, /* number of simplices */
   max_control, /* number of control values */
   dim_system, /* dimension of system state space */
   dim_control, /* dimension of control values */
   max_iter; /* maximum number of iterations */
unsigned long
   pops; /* counter of partial operations */
```

```
real
  iter_delta, /* abort criterium for iteration */
  bis_delta, /* abort criterium for bisection */
  orb_delta, /* abort criterium for optimal orbit calculation */
  max_time; /* maximum optimal orbit calculation time */
int
  **si; /* index field for Lambda-matrices */
real
  *vec, /* iteration vector */
  *v_temp, /* temporary iteration vector */
  **wa, /* field for nonnegative entries of Lambda-matrices */
  **cs; /* field for G vectors */



real det_i(real*, int);


int iterate( void )
{
  int i, j, k, par,
    niters = 0;
  int Abort = 0;
  real delta;
  void *malloc();
  void free();

  v_temp = DOALLOC( max_vertex, real );
  if( ALLOC_FAILED( v_temp ) )
    return( 1 );

  pops=0;

  do
  {
    niters++; /* incrementation of iteration counter */

    fprintf( stderr,''[i%2d]'', niters );

    FOR(i, 0, max_vertex)
      v_temp[i]=vec[i];
    delta= (real) 0.0;

    FOR(i, 0, max_vertex)
    {
```

```
      v_temp[i]=det_i(v_temp,i);
      if (v_temp[i]<vec[i])
        fprintf(stderr,''\n%d IMPROPER VALUE'',i);
    }

    FOR(i, 0, max_vertex)
    {
      delta=MAX(v_temp[i]-vec[i], delta);
      vec[i]=v_temp[i];
    }

    fprintf( stderr, ''[delta=%12.8f]\n'', delta );
    if( delta < 0.0 )
    {
      Abort = 1;
      fprintf( stderr, ''ITERATION ABORTED: IMPROPER START VALUE.\n'' );
    }
  }
  while( niters < max_iter && delta > iter_delta && !Abort );

  fprintf(stderr,''\n%d OPERATIONS\n'',pops);

  FOR( i, 0, max_vertex )
    UNALLOC( vertices[i] );
  FOR( i, 0, max_simplex )
    UNALLOC( simplices[i] );
  FOR( i, 0, max_control )
  {
    UNALLOC( controlvals[i] );
    UNALLOC( wa[i] );
    UNALLOC( cs[i] );
    UNALLOC( si[i] );
  }
  UNALLOC( vertices );
  UNALLOC( simplices );
  UNALLOC( controlvals );
  UNALLOC( si );
  UNALLOC( wa );
  UNALLOC( cs );
  UNALLOC( v_temp );
  return 0;
}


real det_i(vec, i)
```

```
real * vec;
int i;
{
  real a, sum;
  int j, k, kk, ind, first_time=1;

  pops++;
  FOR(j, 0, max_control)
  {
    if (si[j][i]==-1) continue;
    sum = (real) 0.0;
    kk=-1;
    FOR(k, 0, dim_system+1)
    {
      ind=simplices[si[j][i]][k];
      if (ind≠i) sum+=( wa[ j ][ k + i * ( dim_system + 1 ) ]
        * vec[ ind ]);
      else kk=k;
    }
    sum* =(1-lambda*time_step);
    sum+=time_step*cs[j][i];
    if (kk≥0) sum/=1-(1-lambda*time_step)*wa[j][kk+i*(dim_system+1)];
    a = (first_time) ? (first_time=0, sum) : MIN(a,sum);
  }

  if (first_time) exit(1);

  return (a);
}


void prepare( f, g, diag )
RHS f;
CFN g;
int *diag;
{
  int i, kk, c, v, code;
  int r_val = 0;
  int idx, seek_simplex();
  real *lin, *y, yh;
  void *malloc(), free();

  lin = DOALLOC( dim_system+1, real );
  y = DOALLOC( dim_system, real );
  si = DOALLOC( max_control, int* );
```

```
  wa = DOALLOC( max_control, real* );
  cs = DOALLOC( max_control, real* );
  if( ALLOC_FAILED( si ) || ALLOC_FAILED( wa ) || ALLOC_FAILED( cs ) )
  {
    *diag = -1;
    return;
  }
  else
  {
    FOR( i, 0, max_control )
    {
      si[i] = DOALLOC( max_vertex, int );
      wa[i] = DOALLOC( max_vertex*(dim_system+1), real );
      cs[i] = DOALLOC( max_vertex, real );
      if( ALLOC_FAILED( wa[i] ) ||
          ALLOC_FAILED( si[i] ) ||
          ALLOC_FAILED( cs[i] ) )
      {
        *diag = -2;
        return;
      }
    }
  }
#ifdef DEBUG
  fprintf( stderr, ''M : %d, N : %d, P : %d\n'', max_control, max_vertex,
    max_simplex );
#endif
  FOR( c, 0, max_control )
  FOR( v, 0, max_vertex )
  {
#ifdef DEBUG
    fprintf( stderr, ''.'' );
#endif
    (*f)( vertices[v], controlvals[c], y );
    FOR( kk, 0, dim_system ) /* apply Psi if necessary */
    {
      y[kk] = vertices[v][kk] + time_step * y[kk];
      if (per>(real) 0.0) /* periodicity */
      {
        while (y[kk]<0) y[kk]+=per;
        while (y[kk]>per) y[kk]-=per;
      }
    }
    if (per<(real)0.0) /* check identification */
      if ((fabs(y[0])>1.0) || (fabs(y[1])>1.0))
```

```
        {
          yh=y[0];
          y[0]=-y[0]/(y[0]*y[0]+y[1]*y[1]);
          y[1]=-y[1]/(yh*yh+y[1]*y[1]);
        }
      code = seek_simplex( y, &idx, lin );
      if( !code )
      {
        fprintf( stderr, ''ERROR in [c%d][v%d]'', c, v );
        FOR(kk, 0, dim_system) fprintf(stderr, '' %.15f '',y[kk]);
        fprintf(stderr,''\n'');
      }

      si[c][v] =
        code == 1 ? idx : ( r_val++, (int) -1 );
      FOR( kk, 0, dim_system+1 )
      {
        wa[c][v*(dim_system+1)+kk] =
          code == 1 ? lin[kk] : (real) 0.0;
      }
    }
    UNALLOC( lin );
    UNALLOC( y );
    FOR( c, 0, max_control )
    FOR( v, 0, max_vertex )
    {
      cs[c][v] = (*g)( vertices[v], controlvals[c] );
    }
    *diag = r_val;
    return;
}
```

```
/************************************************** (c) Lars Gruene 1993

   MODULE : equation.c
   PURPOSE: Implementation of the equations for the control systems


**************************************************************************/
#include "defs"
#include "equation.h"

#ifndef __MATH_H
  #include <math.h>
#endif

real a_d, b_d, c_d;

CFN cfn_ptr[] =
   {
      testg, phi_g, osz3dp_g
   };

RHS rhs_ptr[] =
   {
      testf, phi_f, osz3dp_f
   };




/************************************************************************
* *
* The first control system equations are taken from the article *
* "Numerical Solution Of Deterministic Continuous Control Problems" *
* by Maurizio Falcone. These equations are used for numerical tests. *
* *
************************************************************************/

void testf( real *x, real *u, real *y )
{
    y[0] = - x[0] + u[0] * x[1];
    y[1] = - x[1];
    return;
```

```
}

real testg( real *x, real *u )
{
  return( 1.0 - x[0] + 0.3 * u[0] );
}



/* SYSTEM: 2D LINEAR OSCILATOR on P,
parameterized by polar coordinates*/


void phi_f(real *x, real *u, real *y)
{
  y[0]=(-1.0-u[0]*cos(x[0])*cos(x[0])-2.0*b_d*sin(x[0])*cos(x[0]));
}

real phi_g(real *x, real *u)
{
  return -(sin(x[0])*(u[0]*cos(x[0])+2.0*b_d*sin(x[0])));
}


/* SYSTEM: 3D LINEAR OSCILLATOR ON P,
parameterized by stereographic projection */


void osz3dp_f(real *x, real *u, real *y)
{
  real s1, s2, s3, s1p, s2p, s3p, T, c;

  c = 1.0+x[0]*x[0]+x[1]*x[1];
  s1 = 2.0/c * x[0];
  s2 = 2.0/c * x[1];
  s3 = 2.0/c -1.0;

  T = (-c_d-u[0])*s1*s3 + s1*s2 + (1.0-b_d)*s2*s3 - a_d*s3*s3;
  s1p=s2-s1*T;
  s2p=s3-s2*T;
  s3p=(-c_d-u[0])*s1-b_d*s2-a_d*s3-s3*T;

  y[0]=(s1p-s3p*x[0])*c/2.0;
  y[1]=(s2p-s3p*x[1])*c/2.0;
}
```

```
real osz3dp_g(real *x, real *u)
{
  real s1, s2, s3, c;

  c = 1+x[0]*x[0]+x[1]*x[1];
  s1 = 2/c * x[0];
  s2 = 2/c * x[1];
  s3 = 2/c -1;

  return ( (-c_d-u[0])*s1*s3 + s1*s2 + (1.0-b_d)*s2*s3 - a_d*s3*s3 );
}
```

```
/************************************************** (c) Lars Gruene 1993


MODULE : equ_orb.c
PURPOSE: Additional equations for dealing with the projective space
         and calculating the trajectory in R^n

*************************************************************************/
#include "defs"
#include "equ_orb.h"

#ifndef __MATH_H
  #include <math.h>
#endif

extern real a_d, b_d, c_d;

RHS rhs2_ptr[] =
  {
    dummy1, osz2_f, osz3_f
  };

PRO p_ptr[] =
  {
    dummy2, osz2_p, osz3_p
  };

PRO o_ptr[] =
  {
    dummy2, osz2_o, osz3_o
  };

PRO i_ptr[] =
  {
    dummy2, osz2_i, osz3_i
  };


/*dummies*/

void dummy1(real *x, real *u, real *y)
{}

void dummy2(real *x, real *u)
{}
```

```
/* 2d linear oszillator, Projective space parameterized by cos, sin */

void osz2_f(real *x, real *u, real *y) /* System equation */
{
  y[0]=x[1];
  y[1]=(-1.0-u[0])*x[0]-2.0*b_d*x[1];
}

void osz2_p(real *x, real *y) /* Projection from S to R */
{
 y[0]=acos( fabs(x[0]/sqrt(x[0]*x[0]+x[1]*x[1]))*((x[0]*x[1]<0.0) ? -1.0 : 1.0) );
}

void osz2_o(real *x, real *w) /* Output Function */
{
  w[0]=x[0];
  w[1]=x[1];
}

void osz2_i(real *y, real *x) /* Injektion from R to S */
{
  x[0]=cos(y[0]);
  x[1]=sin(y[0]);
}


/* 3d linear oszillator, S2 parameterized by stereographic projection */

void osz3_f(real *x, real *u, real *y)
{
  y[0]=x[1];
  y[1]=x[2];
  y[2]=-((c_d+u[0])*x[0]+b_d*x[1]+a_d*x[2]);
}

void osz3_p(real *x, real *y)
{
  real yh;
  real n=sqrt(x[0]*x[0]+x[1]*x[1]+x[2]*x[2]);
  y[0]=1.0/(1.0+x[2]/n) *x[0]/n;
  y[1]=1.0/(1.0+x[2]/n) *x[1]/n;
  if ((fabs(y[0])>1.0) || (fabs(y[1])>1.0))
  {
    yh=y[0];
```

```
      y[0]=-y[0]/(y[0]*y[0]+y[1]*y[1]);
      y[1]=-y[1]/(yh*yh+y[1]*y[1]);
   }
}


void osz3_i(real *y, real *x)
{
   real c = 2.0/(1+y[0]*y[0]+y[1]*y[1]);

   x[0] = c*y[0];
   x[1] = c*y[1];
   x[2] = c-1.0;
}

void osz3_o(real *x, real *w)
{
   w[0]=x[0];
   w[1]=x[1];
}
```

```
/****************************** (c) Uwe Sorgenfrei 1992, Lars Gruene 1993

MODULE: orbit.c

*****************************************************************************/



#include "defs"
#include "orbit.h"
#include "findsimp.h"
#include "equ_orb.h"
#include <math.h>

#define TRACE fprintf( stderr, ''.'' )


int **simplices;
real **vertices, **controlvals;

real lambda, time_step, spatial;
int max_vertex, max_simplex, max_control;
int dim_system, dim_control;
int max_iter;
real iter_delta, bis_delta, orb_delta, max_time;
int **si;
real *vec, *v_temp;
real **wa, **cs;
real per;

extern real a_d, b_d;

static real dist( int, real *, real * );
static real interpolate( real *, int );
static int TheBetter( int, int );
static real calc_next( real *, int *, RHS, CFN );



static real dist( int n, real *x1, real *x2 )
{
  int i;
  real distance = 0.0, pow(), sqrt();

  FOR( i, 0, n )
```

```
    distance += pow( x1[i] - x2[i], 2.0 );
  return( sqrt( distance ) );
}



static int TheBetter( u1, u2 )
int u1, u2;
{
  int i, u_opt;

  FOR( i, 0, dim_control )
    if( controlvals[u1][i] ≠ controlvals[u2][i] )
    {
      u_opt = ( controlvals[u1][i] < controlvals[u2][i] ) ? u1 : u2;
      break;
    }
  return u_opt;
}



static real interpolate( lin, simp_no )
real *lin;
int simp_no;
{
  int i;
  double dummy = 0.0;
  FOR( i, 0, dim_system + 1 )
    dummy += lin[i]*vec[ simplices[simp_no][i] ];
  return( dummy );
}



static real calc_next( x, u_opt, f, g )
real *x;
int *u_opt;
RHS f;
CFN g;
{
   real xh, ik_val, ik_min, beta, *lin, *x_new, retval, *dir, interpolate();
   int i, c, simp_no;
```

```
void *malloc(), free();

lin = DOALLOC( dim_system+1, real );
x_new = DOALLOC( dim_system, real );
dir = DOALLOC( dim_system, real );
beta = 1 - lambda * time_step;
*u_opt = 0;
FOR( c, 0, max_control )
{
    ik_val = time_step * g( x, controlvals[c] );
    (*f)( x, controlvals[c], dir );
    FOR( i, 0, dim_system ) /* apply Psi */
    {
      x_new[i] = x[i] + time_step * dir[i];
      if (per>(real)0.0) /* check periodicity */
        if (x_new[i]<0) x_new[i]+=per;
      else if(x_new[i]>per) x_new[i]-=per;
    }
    if (per<(real)0.0) /* check identification */
      if ((fabs(x_new[0])>1.0) || (fabs(x_new[1])>1.0))
      {
        xh=x_new[0];
        x_new[0]=-x_new[0]/(x_new[0]*x_new[0]+x_new[1]*x_new[1]);
        x_new[1]=-x_new[1]/(xh*xh+x_new[1]*x_new[1]);
      }

    seek_simplex( x_new, &simp_no, lin );

    ik_val += ( beta * interpolate( lin, simp_no ) );
    if( c == 0 )
    {
      ik_min = ik_val; *u_opt = 0 ;
    }
    else
    {
      if( ik_min == ik_val )
        *u_opt = TheBetter( *u_opt, c );
      else if( ik_min > ik_val )
      {
        ik_min = ik_val;
        *u_opt = c;
      }
    }
}
(*f)( x, controlvals[*u_opt], dir );
```

```
    FOR( i, 0, dim_system )
      x_new[i] = x[i] + time_step * dir[i];
    retval = dist( dim_system, x, x_new );
    FOR( i, 0, dim_system )
      x[i] = x_new[i];
    UNALLOC( lin );
    UNALLOC( x_new );
    UNALLOC( dir );
    return retval;
}


real calc_orbit( real *x, RHS f, CFN g, RHS f2, PRO p, PRO o, PRO i)
{
  real distance, *lin, calc_next(), xh, y[10], y2[10];
  int u_opt, j, simp_no, maxval, count=0;
  clock_t clock(), start = clock(), stop;
  void *malloc(), free();

  lin = DOALLOC( dim_system+1, real );
  maxval=(int)max_time;
  fprintf(stderr,''%d steps\n'',maxval);
  FOR( j, 0, dim_system )
    fprintf( stderr, ''%.10lf '', x[j] );
  fprintf( stderr, ''\n'' );

  (*i)(x,y);

  do
  {
    distance = calc_next( x, &u_opt, f, g );
    (*f2)(y,controlvals[u_opt],y2);

    fprintf(stderr,''%d '',count);

    FOR( j, 0, dim_system+1 )
    {
      fprintf(stdout,''%f'',y[j]);
      if (j<dim_system) fprintf(stdout,'' , '');
      else fprintf(stdout,''\n'');
      y[j]+=time_step*y2[j];
      fprintf( stderr, ''%lf '', y[j] );
    }
    fprintf( stderr, ''%.3lf '', controlvals[u_opt][0] );
```

```
    (*p)(y,y2);
    FOR(j, 0, dim_system)
    {
      fprintf(stderr, '' %.5lf'',y2[j]-x[j]);
      x[j]=y2[j];
    }
    fprintf(stderr,''\n'');

    /* apply Psi (only necessary if (*p) doesn't work properly) */
    if (per>(real)0.0) /* check periodicity */
    {
      FOR( j, 0, dim_system )
      {
        if (x[j]<0) x[j]+=per;
        else if(x[j]>per) x[j]-=per;
      }
    }
    if (per<(real)0.0) /* check identification */
      if ((fabs(x[0])>1.0) || (fabs(x[1])>1.0))
      {
          xh=x[0];
          x[0]=-x[0]/(x[0]*x[0]+x[1]*x[1]);
          x[1]=-x[1]/(xh*xh+x[1]*x[1]);
      }

    seek_simplex( x, &simp_no, lin );
    stop = clock();
    TRACE;
  }
  while( distance > orb_delta && simp_no ≠ -1
   && ++count<maxval);

  UNALLOC( lin );

  return distance;
}
```

```
/****************************** (c) Uwe Sorgenfrei 1992, Lars Gruene 1993

MODULE: inout.c

*****************************************************************************/

#ifndef __STDIO_H
  #include <stdio.h>
#endif
#ifndef __STDLIB_H
  #include <stdlib.h>
#endif
#ifndef __MATH_H
  #include <math.h>
#endif

#include "defs"
#include "inout.h"
#include "equation.h"

extern int **simplices;
extern real **vertices,
  **controlvals;

extern real lambda, time_step, spatial, per;
extern int max_vertex, max_simplex, max_control;
extern int dim_system, dim_control;
extern int max_iter;
extern real iter_delta, bis_delta, orb_delta, max_time;
extern int *si;
extern double *vec, *v_temp;
extern double **wa, **cs;

extern real a_d, b_d, c_d;

static int get_para( void );
static int get_tria( void );
static int get_ctrl( void );
static real dist( int n, real *x1, real *x2 );


void read_data()
{
  get_para();
  get_tria();
```

```
  get_ctrl();

}



static real dist( n, x1, x2 ) /* euklidian distance function */
int n;
real *x1, *x2;
{
  int i;
  real distance = 0.0, pow(), sqrt();

  FOR( i, 0, n )
    distance += pow( x1[i] - x2[i], 2.0 );
  return( sqrt( distance ) );
}


double det_timst( f ) /* checking time step size */
RHS f;
{
  int i, j;
  real dummy = 0.0;
  real *zero, *y, dist();
  void *malloc(), free();

  zero = DOALLOC( dim_system, real );
  y = DOALLOC( dim_system, real );
  FOR( i, 0, dim_system )
    zero[i] = 0.0;

  FOR( i, 0, max_vertex )
  FOR( j, 0, max_control )
  {
    (*f)( vertices[i], controlvals[j], y );
    dummy = MAX( dummy, dist( dim_system, zero, y ) );
  }
  UNALLOC( zero );
  UNALLOC( y );
  /*dummy = MIN( spatial/dummy, 1/lambda );*/
  dummy=1/lambda;
  if( time_step > dummy/2.0 )
    time_step = dummy / 2.0;
  return( time_step );
```

```
}


real det_spatial() /* checking space step size */
{
  int i, j;

  FOR( i, 0, max_simplex )
  FOR( j, 0, dim_system+1 )
    spatial = MAX( spatial, dist( dim_system, vertices[ simplices[i][j] ],
      vertices[ simplices[i][(j+1) % (dim_system+1)] ] ) );
  spatial *= 1.1;
  return( spatial );
}


real det_iterstart( g ) /* determine start vector */
CFN g;
{
  int i, j;
  double dummy1, dummy2 = 0.0;
  void *malloc();

  vec = DOALLOC( max_vertex, double );
  if( ALLOC_FAILED( vec ) )
  {
    fprintf( stderr, ''NOT ENOUGH MEMORY. EXIT.\n'' );
    exit( EXIT_FAILURE );
  }
  FOR( i, 0, max_vertex )
  FOR( j, 0, max_control )
  {
    dummy1 = (*g)( vertices[i], controlvals[j] );
    dummy2 = MAX( dummy2, ABS( dummy1 ) );
  }
  dummy1 = - dummy2 * 1.1 / lambda;
  FOR( i, 0, max_vertex )
    vec[i] = dummy1;
  return( dummy1 );
}
```

```
int get_para() /* read control line */
{
  char string[80];
  real atof();

  fscanf( stdin, ''%s'', string );
  time_step = atof( string );
  fscanf( stdin, ''%s'', string );
  lambda = atof( string );
  fscanf( stdin, ''%s'', string );
  max_iter = atoi( string );
  fscanf( stdin, ''%s'', string );
  iter_delta = atof( string );
  fscanf( stdin, ''%s'', string );
  bis_delta = atof( string );
  fscanf( stdin, ''%s'', string );
  orb_delta = atof( string );
  fscanf( stdin, ''%s'', string );
  max_time = atof( string );

  fscanf(stdin, ''%s'', string);
  per=atof(string);
  if (per> (real) 0.0)
    fprintf(stderr, ''PERIODIC: %f\n'',per);
  if (per< (real) 0.0)
    fprintf(stderr, ''STEREOGRAPHIC\n'');

  return 0;
}


int get_tria() /* read triangulation */
{
  int i, j;
  char dummy_string[80];
  real atof();
  void *malloc();

  fscanf( stdin, ''%d %d'', &dim_system, &max_vertex );

  vertices = DOALLOC( max_vertex, real* );
  do
  {
    fscanf( stdin, ''%d'', &i );
```

```
      vertices[i] = DOALLOC( dim_system, real );
      for( j = 0; j < dim_system; j++ )
      {
        fscanf( stdin, ''%s'', dummy_string );
        vertices[i][j] = atof( dummy_string );
      }
    }
    while( i ≠ max_vertex-1 );

    fscanf( stdin, ''%d'', &max_simplex );
    simplices = DOALLOC( max_simplex, int* );
    do
    {
      fscanf( stdin, ''%d'', &i );
      simplices[i] = DOALLOC( dim_system+1, int );
      FOR( j, 0, dim_system + 1 )
      {
        fscanf( stdin, ''%s'', dummy_string );
        simplices[i][j] = atoi( dummy_string );
      }
    }
    while( i ≠ max_simplex-1 );
    return 0;
}

int get_ctrl() /* load control values */
{
    int i, j;
    char dummy_string[80];
    real atof();
    void *malloc();

    fscanf( stdin, ''%d %d'', &dim_control, &max_control );
    controlvals = DOALLOC( max_control, real* );
    do
    {
      fscanf( stdin, ''%d'', &i );
      controlvals[i] = DOALLOC( dim_control, real );
      for( j = 0; j < dim_control; j++ )
      {
        fscanf( stdin, ''%s'', dummy_string );
        controlvals[i][j] = atof( dummy_string );
      }
    }
    while( i ≠ max_control-1 );
```

```
   fscanf( stdin, ''%s'', dummy_string);
   a_d=atof(dummy_string);
   fscanf( stdin, ''%s'', dummy_string);
   b_d=atof(dummy_string);
   fscanf( stdin, ''%s'', dummy_string);
   c_d=atof(dummy_string);

   return 0;
}

int get_nodes() /* load iterated node values
                                    (for calculation of optimal orbits) */
{
   int i;
   char dummy_string[80];
   real atof();
   void *malloc();

   vec = DOALLOC( max_vertex, real );
   do
   {
      fscanf( stdin, ''%d'', &i );
      fscanf( stdin, ''%s'', dummy_string );
      vec[i] = atof( dummy_string );
   }
   while( i ≠ max_vertex-1 );
   return 0;
}
```

```
#
# Makefile : mk_nodes (project)
#
#
# CALL: make -f mkn.fil (without standard defaults)
#
OBJECTS=equation.o iterate.o findsimp.o inout.o \
        lu_solve.o mk_nodes.o
#
LIBES=-lm
#
mkn: $(OBJECTS); cc $(OBJECTS) $(LIBES) -o $@
#
equation.o : equation.c equation.h defs; cc -c equation.c
falcone.o : iterate.c iterate.h findsimp.h \
                lu_solve.h defs; cc -c iterate.c
findsimp.o : findsimp.c findsimp.h lu_solve.h defs; cc -c findsimp.c
inout.o : inout.c inout.h defs; cc -c inout.c
lu_solve.o : lu_solve.c lu_solve.h defs; cc -c lu_solve.c
mk_nodes.o : mk_nodes.c inout.h equation.h iterate.h defs
                cc -c mk_nodes.c


#
# Makefile : mk_orbit (project)
#
#
# CALL: make -f mko.fil (without standard defaults)
#
OBJECTS=equation.o equ_orb.o findsimp.o inout.o \
        lu_solve.o mk_orbit.o orbit.o
#
LIBES=-lm
#
mko: $(OBJECTS); cc $(OBJECTS) $(LIBES) -o $@
#
equ_orb.o : equ_orb.c equ_orb.h defs; cc -c equ_orb.c
equation.o : equation.c equation.h defs; cc -c equation.c
orbit.o : orbit.c orbit.h findsimp.h defs; cc -c orbit.c
findsimp.o : findsimp.c findsimp.h lu_solve.h defs; cc -c findsimp.c
inout.o : inout.c inout.h defs; cc -c inout.c
lu_solve.o : lu_solve.c lu_solve.h defs; cc -c lu_solve.c
mk_orbit.o : mk_orbit.c inout.h equation.h falcone.h defs
                cc -c mk_orbit.c
```

Literaturverzeichnis

[1] Andrea Bacciotti, *Local Stabilizability of Nonlinear Control Systems*, World Scientific, Singapore 1992

[2] William M. Boothby, *An Introduction to Differentiable Manifolds and Riemannian Geometry*, 2nd Edition, Academic Press, Orlando 1986

[3] Fritz Colonius, *Einführung in die Steuerungstheorie*, Ausarbeitung einer Vorlesung gehalten im Wintersemester 1991/92 an der Universität Augsburg

[4] Fritz Colonius, *Einführung in die Steuerungstheorie 2*, Ausarbeitung einer Vorlesung gehalten im Sommersemester 1992 an der Universität Augsburg

[5] Fritz Colonius, Wolfgang Kliemann, *Infinite Time Optimal Control and Periodicity*, Applied Mathematics and Optimization 20 (1989), 113-130

[6] Fritz Colonius, Wolfgang Kliemann , *Asymptotic Null Controllability of Bilinear Systems*, Proceedings of the Workshop „Geometry and Nonlinear Control Theory“, Warsaw, June 1993

[7] M. G. Crandall, L. C. Evans, P. L. Lions, *Some Properties of Viscosity Solutions of Hamilton-Jacobi Equations*, Transactions of the American Mathematical Society 282 (1984), 487-502

[8] M. G. Crandall, P. L. Lions, *Viscosity Solutions of Hamilton-Jacobi Equations*, Transactions of the American Mathematical Society 277 (1983), 1-42

[9] R. J. Elliott, *Viscosity solutions and optimal control*, Pitman Research Notes Series, Pitman, London, 1987

[10] Maurizio Falcone, *Numerical Solution of Deterministic Control Problems*, Dipartimento di Matematica, Universita di Roma *La Sapienza*, Italia

[11] Maurizio Falcone, *A Numerical Approach to the Infinite Horizon Problem of Deterministic Control Theory*, Applied Mathematics and Optimization 15 (1987), 1-13

[12] A. Isidori, *Nonlinear Control Systems: An Introduction*, 2nd ed., Springer, Berlin 1989

[13] Arthur J. Krener, *A Generalization of Chow's Theorem and the Bang-Bang Theorem to nonlinear Control Problems*, SIAM Journal on Control 12 (1974), 43-52

[14] P. L. Lions, *Generalized solutions of Hamilton-Jacobi Equations*, Pitman Research Notes Series, Pitman, London, 1982

[15] Atle Seierstad, Knut Sydsæter, *Optimal Control Theory with Economic Applications*, North-Holland, Amsterdam 1987

[16] Uwe Sorgenfrei, *Theorie und Numerik von optimalen Wertefunktionen bei Kontrollproblemen*, Diplomarbeit, Institut für Mathematik, Universität Augsburg (1992)

[17] Fabian Wirth, *Convergence of the Value Functions of Discounted Infinite Horizon Optimal Control Problems with Low Discount Rates*, Mathematics of Operations Research