

Homotopie-Methoden zum Lösen von Optimalsteuerungsproblemen

Inauguraldissertation

zur

Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften
(Dr. rer. Nat.)

der

Mathematisch-Naturwissenschaftlichen Fakultät

der

Ernst-Moritz-Arndt-Universität Greifswald

vorgelegt von

Christoph Lass

geboren am 24.03.1985

in Neubrandenburg

Greifswald, den 24.03.2015

Dekan:

Prof. Dr. Klaus Fesser

1. Gutachter:

Prof. Dr. Bernd Kugelmann

2. Gutachter:

Prof. Dr. Matthias Gerds

Tag der Promotion:

23.10.2015

Inhaltsverzeichnis

| | |
|---|-----------|
| 1. Vorwort | 3 |
| 2. Optimalsteuerung | 5 |
| 2.1. Grundlagen | 5 |
| 2.2. Problemstellung | 9 |
| 2.3. Notwendige Bedingungen bei infiniten Optimierungsproblemen | 13 |
| 2.4. Minimumprinzip | 16 |
| 3. Homotopie-Methoden | 23 |
| 3.1. Variation von T | 23 |
| 3.2. Variation von S | 28 |
| 3.3. Verallgemeinerung der Parameterabhängigkeit | 30 |
| 3.4. Unterschiede zu Costanzas Ansatz | 33 |
| 4. Numerische Methoden | 35 |
| 4.1. Lösen der ODE | 35 |
| 4.1.1. Äußere Integration | 36 |
| 4.1.2. Innere Integration | 38 |
| 4.2. Indirekte Verfahren | 41 |
| 4.3. Direkte Verfahren | 44 |
| 5. Beispiele | 46 |
| 5.1. Unbeschränkte Probleme | 46 |
| 5.1.1. Manövrierung eines starren Raumfahrzeuges | 46 |
| 5.1.2. Bilinearer, quadratischer Regler | 49 |
| 5.1.3. Beschleunigung eines Wagens | 51 |
| 5.2. Apollo-Problem | 56 |
| 5.3. Probleme mit Beschränkungen | 66 |
| 5.3.1. Beschleunigung eines Wagens | 66 |
| 5.3.2. Biegung einer Stange | 70 |
| 6. Zusammenfassung und Ausblick | 76 |
| 6.1. Zusammenfassung | 76 |
| 6.2. Ausblick | 78 |
| Abbildungsverzeichnis | 84 |
| Tabellenverzeichnis | 85 |

| | |
|------------------------------|-----------|
| Literaturverzeichnis | 86 |
| A. Veröffentlichungen | 89 |
| B. Danksagung | 90 |

1. Vorwort

Das Themengebiet der Optimalsteuerungsprobleme entstand als Verallgemeinerung von Variationsproblemen. Eines der bekanntesten Variationsprobleme, das Brachistochronenproblem, wurde am Juni 1696 von Johann Bernoulli formuliert [13]:

„Wenn in einer vertikalen Ebene zwei Punkte A und B gegeben sind, soll man dem beweglichen Punkte M eine Bahn AMB anweisen, auf welcher er von A ausgehend vermöge seiner eigenen Schwere in kürzester Zeit nach B gelangt.

Damit Liebhaber solcher Dinge Lust bekommen sich an die Lösung dieses Problems zu wagen, mögen sie wissen, dass es nicht, wie es scheinen könnte, bloße Spekulation ist und keinen praktischen Nutzen hat. Vielmehr erweist es sich sogar, was man kaum glauben sollte, auch für andere Wissenszweige, als die Mechanik, sehr nützlich. Um einem voreiligen Urteile entgegenzutreten, möge noch bemerkt werden, dass die gerade Linie AB zwar die kürzeste zwischen A und B ist, jedoch nicht in kürzester Zeit durchlaufen wird. Wohl aber ist die Kurve AMB eine den Geometern sehr bekannte, die ich angeben werde, wenn sie nach Verlauf dieses Jahres kein anderer genannt hat.“

Wie Bernoulli bereits schilderte, ist weder eine gerade Linie noch ein Kreisbogen die Lösung des Problems. Gottfried Wilhelm Leibniz war dabei der erste, der auf die Zykloide [8, S. 42-44] als Lösung kam und somit Bernoulli bereits am 16. Juni 1696 [13] innerhalb von einer Woche antworten konnte.

Das Interesse an der Forschung an Optimalsteuerungsproblemen begann etwa 1950 und war vor allem durch militärische Anwendung wie z.B. Flugbahnoptimierung von Raketen motiviert. Daraus entwickelte sich auch das Interesse an der Optimalsteuerung in der Raumfahrt. Ein bekanntes, komplexes Problem ist das sogenannte „Reentry problem“ bzw. „Apollo-Problem“ [26, S. 491-497]. Dort soll eine Raumkapsel bei minimaler Aufheizung in die Erdatmosphäre eintreten und eine gewünschte Position, Geschwindigkeit und Winkel bzgl. der Erdoberfläche erreichen. Es bedarf einiges an numerischen Aufwands und Vorwissen, um diese Aufgabe mit hinreichender Genauigkeit zu lösen. Das Apollo-Problem eignet sich also perfekt, um die Praxistauglichkeit neuer numerischer Verfahren zu testen.

Obwohl bereits frühe, wichtige Forschungsergebnisse aus den Jahren 1964 und 1966 von Lev Semenovich Pontryagin [23] und Magnus R. Hestenes [12] zu Optimalsteuerungsproblemen bekannt sind, gibt es weiterhin neue Forschungsergebnisse zu diesem Thema. So veröffentlichten Nik, Effate, Motsa und Shirazan 2014 eine Arbeit zur Methode der spektralen Homotopie-Analyse [21], um nichtlineare, unbeschränkte Optimalsteuerungsprobleme zu lösen. Einen anderen Homotopie-Ansatz benutzte Vicente Costanza in seinen Arbeiten [5, 6] aus den Jahren 2008 und 2011 zum numerischen Lösen von unbeschränkten Optimalsteuerungsproblemen, welche die Ausgangsbasis für diese Dissertation sind.

Ziel unserer Arbeit ist es, Costanzas Methode näher zu untersuchen, kritisch zu hinterfragen und zu erweitern. Dabei gehen wir neben der Theorie der optimalen Steuerung auch auf die numerische Umsetzung des Ansatzes ein. Insbesondere möchten wir überprüfen, ob der Ansatz auch auf beschränkte Optimalsteuerungsprobleme übertragen werden kann. Ferner möchten wir das Apollo-Problem mit dem Homotopie-Ansatz effizient und mit hoher Genauigkeit lösen.

Dazu befassen wir uns in dem kommenden Kapitel mit der Theorie von Optimalsteuerungsproblemen und leiten notwendige Bedingungen für ein Extremum her.

2. Optimalsteuerung

2.1. Grundlagen

Bevor wir das Optimalsteuerungsproblem definieren, werden wir die Funktionenräume beschreiben, in denen wir arbeiten.

Definition 2.1.1. Sei $1 \leq p < \infty$. $L^p([a, b], \mathbb{R}^n)$ ist der Raum aller messbaren Funktionen $u : [a, b] \rightarrow \mathbb{R}^n$ mit

$$\int_a^b \|u(t)\|^p dt < \infty$$

und der Norm

$$\|u(\cdot)\|_p := \left(\int_a^b \|u(t)\|^p dt \right)^{1/p},$$

wobei $\|\cdot\| : \mathbb{R}^n \rightarrow \mathbb{R}$ ebenfalls eine Norm sei.

Definition 2.1.2. $L^\infty([a, b], \mathbb{R}^n)$ ist der Raum aller messbaren Funktionen $u : [a, b] \rightarrow \mathbb{R}^n$ mit

$$\operatorname{ess\,sup}_{t \in [a, b]} \|u(t)\| < \infty$$

und der Norm

$$\|u(\cdot)\|_\infty := \operatorname{ess\,sup}_{t \in [a, b]} \|u(t)\|.$$

Die obigen Funktionenräume bezeichnet man auch als *Lebesgue-Räume* und wir werden sie für unsere Steuerungen benutzen. Für unsere Zustände benutzen wir die sogenannten *Sobolev-Räume*. Dazu definieren wir uns zuerst die zugehörigen Normen.

Definition 2.1.3. Sei $1 \leq q, p \leq \infty$. Die Norm $\|\cdot\|_{q,p}$ ist definiert durch

$$\|x(\cdot)\|_{q,p} := \left(\sum_{i=0}^q \|x^{(i)}(\cdot)\|_p^p \right)^{1/p} \quad 1 \leq p < \infty$$

$$\|x(\cdot)\|_{q,\infty} := \max_{0 \leq i \leq q} \|x^{(i)}(\cdot)\|_\infty.$$

Definition 2.1.4. Sei $1 \leq q, p \leq \infty$. Der Raum $W^{q,p}([a, b], \mathbb{R}^n)$ besteht aus allen absolutstetigen Funktionen $x : [a, b] \rightarrow \mathbb{R}^n$, deren Ableitungen bis zur Ordnung $q - 1$ ebenfalls absolutstetig sind, mit

$$\|x(\cdot)\|_{q,p} < \infty.$$

Definition 2.1.5. Sei X ein reeller Vektorraum und $\|\cdot\|_X : X \rightarrow \mathbb{R}$ eine Norm. Man nennt $(X, \|\cdot\|_X)$ einen *Banachraum*, sofern jede Cauchy-Folge aus X einen Grenzwert in X besitzt, d.h. der Raum ist vollständig. Dabei bezeichnen wir das Nullelement von X mit Θ_X und X^* ist der Dualraum von X .

Definition 2.1.6. Seien X und Y Banachräume. Der Operator $F : X \rightarrow Y$ heißt *Fréchet-differenzierbar in x* , falls es einen stetigen, linearen Operator $F'(x)(\cdot) : X \rightarrow Y$ gibt mit

$$F(x+h) - F(x) = F'(x)(h) + o(\|h\|_X)$$

$$\lim_{\|h\|_X \rightarrow 0} \frac{o(\|h\|_X)}{\|h\|_X} = \Theta_Y$$

für alle $h \in X$.

Ist die Abbildung $F'(\cdot) : X \rightarrow \mathcal{L}(X, Y)$, $x \mapsto F'(x)$ stetig bzgl. der Operatornorm $\|F\|_{\mathcal{L}(X, Y)}$, so heißt F *stetig Fréchet-differenzierbar*, wobei $\mathcal{L}(X, Y)$ die Menge aller linearen, stetigen Abbildung von X in Y ist.

$F : X \times Y \rightarrow Z$ heißt *partiell Fréchet-differenzierbar bzgl. x in $(\hat{x}, \hat{y}) \in X \times Y$* , falls $F(\cdot, \hat{y})$ Fréchet-differenzierbar in \hat{x} ist. Wir bezeichnen die partielle Ableitung von F bzgl. x in (\hat{x}, \hat{y}) mit $F'_x(\hat{x}, \hat{y})$. Analog ist die partielle Ableitung bzgl. y definiert.

Ist $F : X \times Y \rightarrow Z$ Fréchet-differenzierbar in (\hat{x}, \hat{y}) , dann ist F auch partiell Fréchet-differenzierbar bzgl. x und y . Es gilt

$$F'(\hat{x}, \hat{y})(x, y) = F'_x(\hat{x}, \hat{y})(x) + F'_y(\hat{x}, \hat{y})(y)$$

für alle $(x, y) \in X \times Y$.

Lemma 2.1.7. Seien X und Y Banachräume. Sei $T : X \rightarrow Y \times \mathbb{R}^n$ mit $T(\cdot) = (T_1(\cdot), T_2(\cdot))$, wobei $T_1 : X \rightarrow Y$ linear, stetig und surjektiv sei. $T_2 : X \rightarrow \mathbb{R}^n$ sei linear und stetig. Dann ist das Bild von T abgeschlossen in $Y \times \mathbb{R}^n$.

Beweis. Siehe [8, S. 210-211]. □

Satz 2.1.8. Seien X, Z Banachräume, $h : X \rightarrow Z$ eine stetig Fréchet-differenzierbare Abbildung, $\hat{x} \in X$ ein Punkt mit $h(\hat{x}) = \Theta_Z$ und sei $\hat{d} \in X$ mit $h'(\hat{x})(\hat{d}) = \Theta_Z$. Ferner sei die Fréchet-Ableitung $h'(\hat{x})$ surjektiv.

Dann existieren eine reelle Zahl $\varepsilon_0 > 0$ und eine Abbildung $r : [0, \varepsilon_0] \rightarrow X$ mit

$$\lim_{\varepsilon \searrow 0} \frac{r(\varepsilon)}{\varepsilon} = \Theta_X$$

$$h(\hat{x} + \varepsilon \hat{d} + r(\varepsilon)) = \Theta_Z. \quad \forall \varepsilon \in [0, \varepsilon_0]$$

Beweis. Siehe [18]. □

Definition 2.1.9. Sei S eine Menge. Dann ist das *relative Innere* von S die Menge $\text{relint}(S)$ mit

$$\text{relint}(S) := \{x \in S \mid \exists \varepsilon > 0 : (U_\varepsilon(x) \cap \text{aff}(S)) \subseteq S\},$$

wobei $\text{aff}(S)$ die affine Hülle von S ist.

Satz 2.1.10 (Trennbarkeit einer konvexen Menge).

Sei X ein Banachraum, $A \subseteq X$ konvex mit $\text{relint}(A) \neq \emptyset$ und $x_0 \notin \text{relint}(A)$.

Dann existiert ein Funktional $f \in X^*$, $f \neq \Theta_{X^*}$ mit

$$f(x_0) \leq f(a). \quad \forall a \in A$$

Beweis. Siehe [1, Satz 2.17]. □

Satz 2.1.11. Sei X ein Banachraum, M ein abgeschlossener Teilraum von X und $\hat{x} \in X \setminus M$.

Dann existiert ein $f \in X^*$ mit $f(\hat{x}) \neq 0$ und $f(x) = 0$ für alle $x \in M$.

Beweis. Siehe [29, S. 98]. □

Wir benötigen noch das folgende Resultat über lineare Differentialgleichungen.

Satz 2.1.12 (Existenz und Eindeutigkeit).

Sei $A \in L^\infty([0, T], \mathbb{R}^{n \times n})$ und $h \in L^\infty([0, T], \mathbb{R}^n)$. Ferner sei die folgende lineare Differentialgleichung gegeben:

$$\dot{x}(t) = A(t)x(t) + h(t). \quad t \in [0, T]$$

Dann gilt:

(i) *Das Anfangswertproblem*

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + h(t) & t \in [0, T] \\ x(0) &= x_0 \end{aligned}$$

hat eine eindeutige Lösung $x \in W^{1,\infty}([0, T], \mathbb{R}^n)$ für jeden Anfangswert $x_0 \in \mathbb{R}^n$.

Diese ist gegeben durch

$$x(t) = \Phi(t) \left(x_0 + \int_0^t \Phi^{-1}(\tau) h(\tau) d\tau \right) \quad t \in [0, T], \quad (2.1.1)$$

wobei das Fundamentalsystem $\Phi(t) \in \mathbb{R}^{n \times n}$ die eindeutige Lösung des Anfangswertproblems

$$\begin{aligned}\dot{\Phi}(t) &= A(t)\Phi(t) \\ \Phi(0) &= I_n\end{aligned}$$

ist.

(ii) Für einen beliebigen Vektor $b \in \mathbb{R}^r$ und Matrizen $C_0, C_T \in \mathbb{R}^{r \times n}$ mit

$$\text{Rang}(C_0\Phi(0) + C_T\Phi(T)) = r$$

besitzt das Randwertproblem

$$\begin{aligned}\dot{x}(t) &= A(t)x(t) + h(t) & t \in [0, T] \\ C_0x(0) + C_Tx(T) &= b\end{aligned}$$

eine Lösung.

Beweis. Der Beweis für Teil (i) kann in [11] gefunden werden. Die zweite Behauptung lässt sich mit Hilfe von (2.1.1) zeigen. Dazu untersuchen wir die Randbedingung:

$$\begin{aligned}b &= C_0x(0) + C_Tx(T) \\ &= C_0x(0) + C_T\Phi(T) \left(x(0) + \int_0^T \Phi^{-1}(\tau)h(\tau) \, d\tau \right) \\ &= (C_0 + C_T\Phi(T))x(0) + C_T\Phi(T) \int_0^T \Phi^{-1}(\tau)h(\tau) \, d\tau \\ &= (C_0\Phi(0) + C_T\Phi(T))x(0) + C_T\Phi(T) \int_0^T \Phi^{-1}(\tau)h(\tau) \, d\tau.\end{aligned}$$

Dabei folgt die letzte Zeile aus der Anfangsbedingung $\Phi(0) = I_n$. Wir stellen die obigen Gleichungen nach $x(0)$ um und erhalten das lineare Gleichungssystem

$$(C_0\Phi(0) + C_T\Phi(T))x(0) = b - C_T\Phi(T) \int_0^T \Phi^{-1}(\tau)h(\tau) \, d\tau.$$

Das System besitzt eine Lösung, weil die Matrix $(C_0\Phi(0) + C_T\Phi(T))$ laut Voraussetzung vollen Rang besitzt. Somit lässt sich das Randwertproblem auf ein Anfangswertproblem zurückführen, welches gemäß Teil (i) eine Lösung hat. \square

In dem nächsten Abschnitt definieren wir, was ein Optimalsteuerungsproblem ist, und wir stellen ferner ein äquivalentes, infinites Optimierungsproblem auf.

2.2. Problemstellung

Definition 2.2.1. Gegeben seien die Funktionen und Variablen

$$\begin{aligned} L &: \mathbb{R}^n \times \mathbb{R}^{n_u} \rightarrow \mathbb{R} \\ \varphi &: \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R} \\ f &: \mathbb{R}^n \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^n \\ \psi &: \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^{n_\psi} \\ c &: \mathbb{R}^n \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_c} \\ S, T &\in \mathbb{R}. \end{aligned}$$

Bei einem *Optimalsteuerungsproblem* werden die Zustände $x \in W^{1,\infty}([0, T], \mathbb{R}^n)$ und die Steuerungen $u \in L^\infty([0, T], \mathbb{R}^{n_u})$ gesucht, welche die folgende Aufgabe lösen

$$\min_u J(u) = \min_u \int_0^T L(x(t), u(t)) dt + \varphi(S, x(0), x(T)) \quad T \geq 0 \quad (2.2.1)$$

unter der *Differentialgleichung*

$$\dot{x}(t) = f(x(t), u(t)) \quad t \in [0, T] \quad (2.2.2)$$

und den *Randbedingungen*

$$\psi(S, x(0), x(T)) = 0_{n_\psi} \quad (2.2.3)$$

mit den *Steuer- und Zustandsbeschränkungen*

$$c(x(t), u(t)) \leq 0_{n_c} \quad t \in [0, T] \quad (2.2.4)$$

und den *Mengenbeschränkungen*

$$u(t) \in \mathcal{U} \subseteq \mathbb{R}^{n_u}. \quad t \in [0, T] \quad (2.2.5)$$

Bei den Randbedingungen und den Steuer- und Zustandsbeschränkungen ist das Relationszeichen jeweils komponentenweise zu verstehen. Den Parameter S führen wir hier bereits für unseren neuen Ansatz in Kapitel 3 ein, wobei dieser für die folgenden Überlegungen nicht relevant ist.

Wir nehmen dabei die Differentialgleichung als autonom an, d.h. die Zeit t taucht nicht explizit in der rechten Seite auf. Dies ist keine Einschränkung, weil jedes nichtautonome Problem in ein autonomes Problem umgewandelt werden kann. Sei dazu die folgende nichtautonome Differentialgleichung gegeben

$$\dot{x}(t) = g(t, x(t), u(t)). \quad t \in [0, T]$$

Wir führen den neuen Zustand $x_{n+1} : \mathbb{R} \rightarrow \mathbb{R}$ mit $x_{n+1}(t) = t$ ein und erhalten dadurch das äquivalente autonome Problem

$$\begin{aligned}\dot{x}(t) &= g(x_{n+1}(t), x(t), u(t)) & t \in [0, T] \\ \dot{x}_{n+1}(t) &= 1 \\ x_{n+1}(0) &= 0.\end{aligned}$$

Für unser Optimalsteuerungsproblem (2.2.1) haben wir also einen weiteren Zustand, eine zusätzliche Differentialgleichung und eine weitere Randbedingung.

Auch Probleme mit freier Endzeit T können wir in der Form (2.2.1) aufschreiben. Wir betrachten dazu ein Optimalsteuerungsproblem mit freier Endzeit:

$$\begin{aligned}\min_{u, T} \quad J(u) &= \min_{u, T} \int_0^T L(x(t), u(t)) dt + \varphi(S, x(0), x(T)) & T \geq 0 \\ \dot{x}(t) &= f(x(t), u(t)) & t \in [0, T] \\ \psi(S, x(0), x(T)) &= 0_{n_\psi} \\ c(x(t), u(t)) &\leq 0_{n_c} \\ u(t) &\in \mathcal{U}.\end{aligned}$$

Analog zu der Transformation der nichtautonomen Probleme führen wir den Zustand $x_{n+1} : \mathbb{R} \rightarrow \mathbb{R}$ mit $x_{n+1}(\tau) = T$ ein und bekommen dann ein äquivalentes Problem der optimalen Steuerung mit fester Endzeit:

$$\begin{aligned}\min_u \quad J(u) &= \min_u \int_0^1 x_{n+1}(\tau) L(\bar{x}(\tau), \bar{u}(\tau)) d\tau + \varphi(S, \bar{x}(0), \bar{x}(1)) \\ \dot{\bar{x}}(\tau) &= x_{n+1}(\tau) f(\bar{x}(\tau), \bar{u}(\tau)) & \tau \in [0, 1] \\ \dot{x}_{n+1}(\tau) &= 0 \\ \psi(S, \bar{x}(0), \bar{x}(1)) &= 0_{n_\psi} \\ c(\bar{x}(\tau), \bar{u}(\tau)) &\leq 0_{n_c} \\ \bar{u}(\tau) &\in \mathcal{U}\end{aligned}$$

mit den neuen Zuständen $\bar{x} \in W^{1, \infty}([0, 1], \mathbb{R}^n)$, Steuerungen $\bar{u} \in L^\infty([0, 1], \mathbb{R}^{n_u})$ und der Zeitvariable $\tau \in [0, 1]$, wobei

$$\begin{aligned}\bar{x}(\tau) &:= x(x_{n+1}(\tau)\tau) & \tau \in [0, 1] \\ \bar{u}(\tau) &:= u(x_{n+1}(\tau)\tau).\end{aligned}$$

Auch die Wahl des Zeitintervalls $[0, T]$ stellt keine Restriktion dar, weil Optimalsteuerungsprobleme mit $t \in [t_0, t_e]$ leicht auf unser Standardintervall transformiert werden können:

$$\begin{aligned} \min_u J(u) &= \min_u \int_0^{t_e} \frac{t_e - t_0}{t_e} L(\bar{x}(\tau), \bar{u}(\tau)) \, d\tau + \varphi(S, \bar{x}(0), \bar{x}(t_e)) \\ \dot{\bar{x}}(\tau) &= \frac{t_e - t_0}{t_e} f(\bar{x}(\tau), \bar{u}(\tau)) & \tau \in [0, t_e] \\ \psi(S, \bar{x}(0), \bar{x}(t_e)) &= 0_{n_\psi} \\ c(\bar{x}(\tau), \bar{u}(\tau)) &\leq 0_{n_c} \\ \bar{u}(\tau) &\in \mathcal{U} \end{aligned}$$

mit den neuen Zuständen $\bar{x} \in W^{1,\infty}([0, t_e], \mathbb{R}^n)$, Steuerungen $\bar{u} \in L^\infty([0, t_e], \mathbb{R}^{n_u})$ und der Zeitvariable $\tau \in [0, t_e]$, wobei

$$\begin{aligned} \bar{x}(\tau) &:= x \left(t_0 + \tau \frac{t_e - t_0}{t_e} \right) & \tau \in [0, t_e] \\ \bar{u}(\tau) &:= u \left(t_0 + \tau \frac{t_e - t_0}{t_e} \right). \end{aligned}$$

Im Folgenden betrachten wir nur Optimalsteuerungsprobleme ohne Steuer- und Zustandsbeschränkungen, d.h. $n_c = 0$, weil unser Ansatz in Kapitel 3 nur auf diese anwendbar ist. Wir werden in Abschnitt 5.3 auch Beispiele mit Restriktionen betrachten, welche durch unbeschränkte Optimalsteuerungsprobleme approximiert werden können.

Um notwendige Bedingungen für ein Minimum herzuleiten, stellen wir ein zu (2.2.1) äquivalentes, infinites Optimierungsproblem auf.

Definition 2.2.2. Seien X, Z Banach-Räume, $F : X \rightarrow \mathbb{R}$ ein Funktional, $H : X \rightarrow Z$ ein Operator und $S \subseteq X$ eine abgeschlossene, konvexe Menge. Bei einem *infiniten Optimierungsproblem* wird ein $\hat{x} \in X$ gesucht, welches die folgende Aufgabe löst:

$$\min_{x \in S} F(x) \tag{2.2.6}$$

unter

$$H(x) = \Theta_Z.$$

Definition 2.2.3. Sei $X := W^{1,\infty}([0, T], \mathbb{R}^n) \times L^\infty([0, T], \mathbb{R}^{n_u})$ und $F : X \rightarrow \mathbb{R}$ ist definiert durch

$$F(x, u) = \int_0^T L(x(t), u(t)) dt + \varphi(S, x(0), x(T)).$$

Sei $Z := L^\infty([0, T], \mathbb{R}^{n_u}) \times \mathbb{R}^{n_\psi}$ und $H = (H_1, H_2) : X \rightarrow Z$ ist gegeben durch

$$\begin{aligned} H_1(x, u) &= f(x(\cdot), u(\cdot)) - \dot{x}(\cdot) \\ H_2(x, u) &= -\psi(S, x(0), x(T)). \end{aligned}$$

Sei $S \subseteq X$ definiert durch $S := W^{1,\infty}([0, T], \mathbb{R}^n) \times U_{ad}$ mit

$$U_{ad} := \{u(\cdot) \in L^\infty([0, T], \mathbb{R}^{n_u}) \mid u(t) \in \mathcal{U} \text{ f.ü. in } [0, T]\}.$$

Das Optimalsteuerungsproblem (2.2.1) ist für $n_c = 0$ äquivalent zu dem infiniten Optimierungsproblem

$$\min_{(x,u) \in S} F(x, u)$$

unter

$$H(x, u) = \Theta_Z.$$

Sind L und φ stetig bzgl. aller Argumente und stetig differenzierbar bzgl. x und u , dann ist F Fréchet-differenzierbar in (\hat{x}, \hat{u}) mit

$$\begin{aligned} F'(\hat{x}, \hat{u})(\delta x, \delta u) &= \varphi'_{x_0}(\hat{x}(0), \hat{x}(T))\delta x(0) + \varphi'_{x_e}(\hat{x}(0), \hat{x}(T))\delta x(T) \\ &\quad + \int_0^T L'_x(\hat{x}(t), \hat{u}(t))\delta x(t) + L'_u(\hat{x}(t), \hat{u}(t))\delta u(t) dt \\ &=: \varphi'_{x_0}\delta x(0) + \varphi'_{x_e}\delta x(T) + \int_0^T L'_x[t]\delta x(t) + L'_u[t]\delta u(t) dt. \end{aligned}$$

Die letzte Zeile benutzen wir zur verkürzten Darstellung. Gilt außerdem, dass f und ψ stetig bzgl. aller Argumente und stetig differenzierbar bzgl. x und u sind, dann ist H stetig Fréchet-differenzierbar in (\hat{x}, \hat{u}) mit

$$\begin{aligned} H'(\hat{x}, \hat{u})(\delta x, \delta u) &= \begin{pmatrix} H'_1(\hat{x}, \hat{u})(\delta x, \delta u) \\ H'_2(\hat{x}, \hat{u})(\delta x, \delta u) \end{pmatrix} \\ H'_1(\hat{x}, \hat{u})(\delta x, \delta u) &= f'_x(\hat{x}(\cdot), \hat{u}(\cdot))\delta x(\cdot) + f'_u(\hat{x}(\cdot), \hat{u}(\cdot))\delta u(\cdot) - \dot{\delta x}(\cdot) \\ &=: f'_x[\cdot]\delta x(\cdot) + f'_u[\cdot]\delta u(\cdot) - \dot{\delta x}(\cdot) \\ H'_2(\hat{x}, \hat{u})(\delta x, \delta u) &= -\psi'_{x_0}(S, \hat{x}(0), \hat{x}(T))\delta x(0) - \psi'_{x_e}(S, \hat{x}(0), \hat{x}(T))\delta x(T) \\ &=: -\psi'_{x_0}\delta x(0) - \psi'_{x_e}\delta x(T), \end{aligned}$$

wobei wir die jeweils verkürzte Darstellung im Rest des Kapitels benutzen werden.

Als Nächstes möchten wir notwendige Bedingungen für eine mögliche Lösung des infiniten Optimierungsproblems aufstellen. Wir halten uns dabei an [8, S. 86-93].

2.3. Notwendige Bedingungen bei infiniten Optimierungsproblemen

Definition 2.3.1. In dem infiniten Optimierungsproblem (2.2.6) heißt $x \in X$ *zulässig*, falls $x \in \Sigma := \{x \in S \mid H(x) = \Theta_Z\}$.

Definition 2.3.2. $\hat{x} \in \Sigma$ ist ein *lokales Minimum* des infiniten Optimierungsproblem (2.2.6), wenn es ein $\varepsilon > 0$ gibt mit

$$F(\hat{x}) \leq F(x), \quad \forall x \in \{x \in \Sigma : \|x - \hat{x}\|_X < \varepsilon\}$$

Satz 2.3.3 (Fritz-John-Bedingungen).

Seien $F : X \rightarrow \mathbb{R}$ Fréchet-differenzierbar und $H : X \rightarrow Z$ stetig Fréchet-differenzierbar. Ferner sei \hat{x} ein lokales Minimum von (2.2.3) und $S \subset X$ eine abgeschlossene und konvexe Menge mit $\text{int}(S) \neq \emptyset$. Weiterhin sei das Bild von $H'(\hat{x})$ keine echte dichte Teilmenge von Z . Dann existieren nichttriviale Multiplikatoren $(l_0, \lambda^*) \in \mathbb{R} \times Z^*$ mit

$$l_0 \geq 0 \\ l_0 F'(\hat{x})(d) - \lambda^*(H'(\hat{x})(d)) \geq 0, \quad \forall d \in S - \{\hat{x}\}$$

Wir benötigen die folgende Definition, die in dem Beweis des obigen Satzes benutzt wird.

Definition 2.3.4. Sei $\hat{x} \in S$. Die *konische Hülle* von S in \hat{x} ist definiert durch

$$\text{coh}(S, \hat{x}) := \{\alpha(s - \hat{x}) \mid s \in S, \alpha \geq 0\}.$$

Wir orientieren uns bei dem Beweis an [8, S. 89-93], [17, Theorem 3.1, Corollar 4.2] und [16].

Beweis. Wir zeigen zunächst durch Widerspruch, dass unter der Annahme der Surjektivität von $H'(\hat{x})(\cdot) : X \rightarrow Z$ auch $F'(\hat{x})(x - \hat{x}) \geq 0$ gelten muss. Sei dazu $H'(\hat{x})(\cdot)$ surjektiv und sei $x \in \text{int}(S)$ zulässig mit

$$\begin{aligned} F(\hat{x}) + F'(\hat{x})(x - \hat{x}) &< F(\hat{x}) \\ H(\hat{x}) + H'(\hat{x})(x - \hat{x}) &= \Theta_Z. \end{aligned} \tag{2.3.1}$$

Auf Grund der Surjektivität von $H'(\hat{x})$ gibt es nach Satz 2.1.8 eine reelle Zahl $t_0 > 0$ und eine Abbildung $r : [0, t_0] \rightarrow X$ mit

$$\begin{aligned} r(0) &:= \Theta_X \\ \lim_{t \searrow 0} \frac{r(t)}{t} &= \Theta_X \\ H(\hat{x} + t(x - \hat{x}) + r(t)) &= \Theta_Z, \quad \forall t \in [0, t_0] \end{aligned}$$

Damit ist die Kurve $x(t) = \hat{x} + t(x - \hat{x}) + r(t)$ für jedes $t \in [0, t_0]$ zulässig und es gilt

$$\begin{aligned} x(0) &= \hat{x} \\ x'(0) &= \lim_{t \searrow 0} \frac{x(t) - x(0)}{t} = \lim_{t \searrow 0} \frac{x(t) - \hat{x}}{t} \\ &= x - \hat{x} + \lim_{t \searrow 0} \frac{r(t)}{t} = x - \hat{x}. \end{aligned}$$

Unter Ausnutzung von (2.3.1) gilt für die Zielfunktion

$$\left. \frac{d}{dt} F(x(t)) \right|_{t=0} = F'(\hat{x}) \frac{dx(0)}{dt} = F'(\hat{x})(x - \hat{x}) < 0.$$

Somit ist $x - \hat{x}$ eine Abstiegsrichtung von F und entsprechend gilt $F(x(t)) < F(\hat{x})$ für hinreichend kleine $t > 0$, da F stetig ist. Dies wird im Folgenden zu einem Widerspruch der Minimalität von \hat{x} führen.

Wir müssen noch überprüfen, ob $x(t) \in S$ für hinreichend kleine $t > 0$. Da x im Inneren von S liegt, gibt es eine Umgebung $U_\delta(x)$ mit $\hat{x} + t(z - \hat{x}) \in S$ für alle $t \in [0, 1]$ und alle $z \in U_\delta(x)$. Zusammen mit der Abbildung r aus Satz 2.1.8 gilt wegen

$$\begin{aligned} \lim_{t \searrow 0} \frac{r(t)}{t} &= \Theta_X \\ \left\| \frac{r(t)}{t} \right\|_X &< \delta \end{aligned}$$

für ein hinreichend kleines $\varepsilon > t > 0$ für unsere Kurve $x(t)$

$$x(t) = \hat{x} + t(x - \hat{x}) + r(t) = \hat{x} + t \left(x + \frac{r(t)}{t} - \hat{x} \right) \in S.$$

Daraus folgt, dass

$$F'(\hat{x})(d) \geq 0 \quad \forall d \in \{d \in \text{int}(\text{coh}(S, \hat{x})) \mid H'(\hat{x})(d) = \Theta_Z\}, \quad (2.3.2)$$

weil sonst \hat{x} kein lokales Minimum sein kann.

Wir betrachten nun die nichtleere konvexe Menge

$$A := \left\{ \begin{pmatrix} F'(\hat{x})(d) + r \\ H'(\hat{x})(d) \end{pmatrix} : d \in \text{int}(\text{coh}(S, \hat{x})), r > 0 \right\},$$

welche in die folgenden Mengen zerlegt werden kann:

$$\begin{aligned} A_1 &:= \left\{ \begin{pmatrix} F'(\hat{x})(d) \\ H'(\hat{x})(d) \end{pmatrix} : d \in \text{int}(\text{coh}(S, \hat{x})) \right\} \\ A_2 &:= \left\{ \begin{pmatrix} r \\ \Theta_Z \end{pmatrix} : r > 0 \right\}. \end{aligned}$$

Unter Anwendung des Satzes über offene Abbildungen kann man zeigen, dass die Projektion von A_1 auf Z als Teilmenge des topologischen Raumes Z innere Punkte enthält, da nach unserer Annahme $H'(\hat{x})$ surjektiv ist. Somit enthält A innere Punkte in $\mathbb{R} \times Z$. Allerdings gilt gleichzeitig auch $(0, \Theta_Z) \notin \text{int}(A)$, denn angenommen es wäre nicht so: Dann gibt es ein $r > 0$ und $d \in \text{int}(\text{coh}(S, \hat{x}))$ mit

$$F'(\hat{x})(d) = -r < 0.$$

Dies steht aber in Widerspruch zu (2.3.2).

Damit können wir den Trennungssatz 2.1.10 benutzen und es existiert dementsprechend ein nichtverschwindendes, lineares, stetiges Funktional $f \in (\mathbb{R} \times Z)^*$ und eine zugehörige Hyperebene, die A und $(0, \Theta_Z)$ trennt. Somit gibt es nichttriviale Multiplikatoren $(l_0, \lambda^*) \in \mathbb{R} \times Z^*$ mit

$$l_0(F'(\hat{x})(d) + r) - \lambda^*(H'(\hat{x})(d)) \geq f(0, \Theta_Z) = 0 \quad (2.3.3)$$

für alle $d \in \text{int}(\text{coh}(S, \hat{x}))$ und $r > 0$. Dies gilt auch für alle $d \in \text{coh}(S, \hat{x})$ und $r \geq 0$, da die Funktionale l_0, λ^* und die linearen Abbildungen $F'(\hat{x})(\cdot), H'(\hat{x})(\cdot)$ stetig sind. Wählen wir dabei $d = \Theta_X \in \text{coh}(S, \hat{x})$, so erhalten wir die Bedingung $l_0 \geq 0$.

Ist entgegen unserer Annahme $H'(\hat{x})$ nicht surjektiv, dann ist der Abschluss der Menge

$$M := \left\{ \begin{pmatrix} r \\ H'(\hat{x})(x) \end{pmatrix} : r \in \mathbb{R}, x \in X \right\}$$

ein echter abgeschlossener Teilraum von $\mathbb{R} \times Z$, da das Bild von $H'(\hat{x})$ gemäß der Voraussetzung des Satzes kein echter dichter Teilraum von Z ist. Somit sind die Voraussetzungen von Satz 2.1.11 erfüllt und es gibt ein nichttriviales, lineares, stetiges Funktional $f \in (\mathbb{R} \times Z)^*$ mit $f(r, z) = 0$ für alle $(r, z) \in \overline{M} \supset A$. Also existieren nichtverschwindende Multiplikatoren $(l_0, \lambda^*) \in \mathbb{R} \times Z^*$ mit

$$l_0(F'(\hat{x})(d) + r) - \lambda^*(H'(\hat{x})(d)) = 0.$$

Damit gilt natürlich auch die Ungleichung (2.3.3) und wir können wieder die obige Argumentation benutzen. \square

Satz 2.3.5 (Mangasarian-Fromowitz-Bedingung).

Seien die Voraussetzungen von Satz 2.3.3 erfüllt und es gelte weiterhin, dass $H'(\hat{x})$ surjektiv ist. Existiert ein $\hat{d} \in \text{int}(S - \{\hat{x}\})$ mit $H'(\hat{x})(\hat{d}) = \Theta_Z$, dann gelten die Behauptungen von Satz 2.3.3 mit $l_0 = 1$.

Beweis. Siehe [30]. \square

Die beiden Resultate können auch auf infinite Optimierungsprobleme mit Ungleichungsnebenbedingungen erweitert werden. Dies kann man in [8, S. 88-93] nachvollziehen. Wir wenden im nächsten Abschnitt diese Ergebnisse auf Optimalsteuerungsprobleme an, wobei wir uns an [8, S. 96-107, S. 113-114] halten.

2.4. Minimumprinzip

Notwendige Bedingungen für Optimalsteuerungsprobleme wurden bereits vor 50 Jahren untersucht. Das bekannteste Resultat ist das sogenannte „Maximumprinzip“ bzw. in unserem Fall das „Minimumprinzip“. Die ersten Resultate sind dabei in [23] und [12] zu finden. Wir werden hier neuere Ergebnisse, die wir im vorigen Abschnitt behandelt haben, benutzen, um das Minimumprinzip herzuleiten, welches eine Basis für unseren neuen Ansatz in Kapitel 3 bildet.

Zunächst definieren wir uns analog zu infiniten Optimierungsproblemen Zulässigkeit und lokale Minima für Optimalsteuerungsprobleme.

Definition 2.4.1. Ein Paar $(x, u) \in W^{1,\infty}([0, T], \mathbb{R}^n) \times L^\infty([0, T], \mathbb{R}^{n_u})$ heißt *zulässig* für das Optimalsteuerungsproblem (2.2.1), sofern die Bedingungen bzw. Beschränkungen (2.2.2)-(2.2.5) erfüllt sind.

Definition 2.4.2. Ein zulässiges Paar (\hat{x}, \hat{u}) heißt *schwaches lokales Minimum* von dem Optimalsteuerungsproblem aus Definition 2.2.1, wenn es ein $\varepsilon > 0$ gibt, so dass

$$\int_0^T L(\hat{x}(t), \hat{u}(t)) dt + \varphi(S, \hat{x}(0), \hat{x}(T)) \leq \int_0^T L(x(t), u(t)) dt + \varphi(S, x(0), x(T))$$

für alle zulässigen (x, u) mit $\|x - \hat{x}\|_{1,\infty} < \varepsilon$ und $\|u - \hat{u}\|_\infty < \varepsilon$.

Satz 2.4.3 (Existenz von Multiplikatoren).

Gegeben sei ein unbeschränktes Optimalsteuerungsproblem gemäß Definition 2.2.1, d.h. ohne Steuer- und Zustandsbeschränkungen der Form (2.2.4), wobei die Funktionen L , φ , f und ψ stetig in allen Argumenten und ferner stetig differenzierbar in x und u seien. $\mathcal{U} \subseteq \mathbb{R}^{n_u}$ sei eine abgeschlossene, konvexe Menge, deren Inneres nicht leer sei.

Ist (\hat{x}, \hat{u}) ein schwaches lokales Minimum von (2.2.1), dann existieren nichtverschwindende Multiplikatoren $l_0 \in \mathbb{R}$ und $\lambda^ \in Z^*$ mit*

$$l_0 \geq 0$$

$$l_0 F'(\hat{x}, \hat{u})(x - \hat{x}, u - \hat{u}) - \lambda^*(H'(\hat{x}, \hat{u})(x - \hat{x}, u - \hat{u})) \geq 0. \quad \forall (x, u) \in S$$

Beweis. Wir zeigen, dass die Voraussetzungen von Satz 2.3.3 erfüllt sind. Aus den Bedingungen an die Stetigkeit und Differenzierbarkeit von L , φ , f und ψ folgt, dass F Fréchet-differenzierbar und H stetig Fréchet-differenzierbar ist. Die Bedingungen an S folgen aus den Voraussetzungen an \mathcal{U} . Mit Satz 2.1.12 folgt, dass $H'_1(\hat{x}, \hat{u})$ surjektiv ist. Damit können wir Lemma 2.1.7 benutzen und erhalten, dass das Bild von $H'(\hat{x}, \hat{u})$ abgeschlossen in Z ist. Also ist dieses Bild keine echte dichte Teilmenge in Z . Wir können somit Satz 2.3.3 anwenden. \square

Satz 2.4.4 (Darstellung der Multiplikatoren).

Seien die Voraussetzungen von Satz 2.4.3 erfüllt, wobei dort $\lambda^* = (\lambda_e^*, \mu) \in (L^\infty([0, T], \mathbb{R}^n))^* \times \mathbb{R}^{n_\psi}$ ist.

Dann existiert eine Funktion $\lambda \in W^{1,\infty}([0, T], \mathbb{R}^n)$ mit

$$\lambda_e^*(h(\cdot)) = - \int_0^T \lambda^\top(t) h(t) dt \quad (2.4.1)$$

für alle $h \in L^\infty([0, T], \mathbb{R}^n)$. Ferner kann λ explizit dargestellt werden.

Beweis. Aus Satz 2.4.3 haben wir die Variationsungleichung

$$l_0 F'(\hat{x}, \hat{u})(x - \hat{x}, u - \hat{u}) - \lambda^*(H'(\hat{x}, \hat{u})(x - \hat{x}, u - \hat{u})) \geq 0. \quad \forall (x, u) \in S$$

Wir setzen $u = \hat{u}$, nutzen die Fréchet-Ableitungen aus Definition 2.2.3 aus und erhalten die Variationsgleichung

$$\begin{aligned} 0 = & (l_0 \varphi'_{x_0} + \mu^\top \psi'_{x_0}) \delta x(0) + (l_0 \varphi'_{x_e} + \mu^\top \psi'_{x_e}) \delta x(T) \\ & + \lambda_e^*(\dot{\delta}x(\cdot) - f'_x[\cdot] \delta x(\cdot)) + \int_0^T l_0 L'_x[t] \delta x(t) dt \end{aligned} \quad (2.4.2)$$

für alle $\delta x \in W^{1,\infty}([0, T], \mathbb{R}^n)$. Die Gleichheit folgt dabei aus der Bedingung, dass die Variationsungleichung für alle $\delta x \in W^{1,\infty}([0, T], \mathbb{R}^n)$ gelten muss und somit auch für $-\delta x$. Analog erhalten wir durch Setzen von $x = \hat{x}$

$$l_0 \int_0^T L'_u[t](u(t) - \hat{u}(t)) dt - \lambda_e^*(f'_u[\cdot](u(\cdot) - \hat{u}(\cdot))) \geq 0 \quad (2.4.3)$$

für alle $u \in U_{ad}$ aus Definition 2.2.3.

Wir definieren

$$h(\cdot) := \dot{\delta}x(\cdot) - f'_x[\cdot] \delta x(\cdot) \quad (2.4.4)$$

und das Anfangswertproblem

$$\begin{aligned} \dot{\delta}x(t) &= f'_x[t] \delta x(t) + h(t) & t \in [0, T] \\ \delta x(0) &= 0_n. \end{aligned}$$

Satz 2.1.12 liefert für alle $h \in L^\infty([0, T], \mathbb{R})$ die eindeutige Lösung

$$\delta x(t) = \Phi(t) \int_0^t \Phi^{-1}(\tau) h(\tau) d\tau, \quad (2.4.5)$$

wobei Φ die Lösung des Anfangswertproblems

$$\begin{aligned}\dot{\Phi}(t) &= f'_x[t]\Phi(t) & t \in [0, T] \\ \Phi(0) &= I_n.\end{aligned}$$

ist. Wir setzen (2.4.4)-(2.4.5) in Gleichung (2.4.2) und es ergibt sich

$$\begin{aligned}-\lambda_e^*(h(\cdot)) &= (l_0\varphi'_{x_e} + \mu^\top \psi'_{x_e})\Phi(T) \int_0^T \Phi^{-1}(\tau)h(\tau) d\tau \\ &\quad + \int_0^T l_0L'_x[t]\Phi(t) \left(\int_0^t \Phi^{-1}(\tau)h(\tau) d\tau \right) dt.\end{aligned}$$

Das letzte Integral kann dabei mit Hilfe der Äquivalenz

$$\int_a^b f_1(x) \left(\int_a^x f_2(y) dy \right) dx = \int_a^b f_2(x) \left(\int_x^b f_1(y) dy \right) dx,$$

die sich durch partielle Integration ergibt, umgeformt werden, so dass wir durch Ausklammern von $\Phi^{-1}(t)h(t)$

$$\begin{aligned}-\lambda_e^*(h(\cdot)) &= \int_0^T \left((l_0\varphi'_{x_e} + \mu^\top \psi'_{x_e})\Phi(T) + \int_t^T l_0L'_x[\tau]\Phi(\tau) d\tau \right) \Phi^{-1}(t)h(t) dt \\ &= \int_0^T \lambda^\top(t)h(t) dt\end{aligned}$$

erhalten, wobei

$$\lambda^\top(t) := \left((l_0\varphi'_{x_e} + \mu^\top \psi'_{x_e})\Phi(T) + \int_t^T l_0L'_x[\tau]\Phi(\tau) d\tau \right) \Phi^{-1}(t). \quad (2.4.6)$$

□

Wir benötigen noch die Hamilton-Funktion, um das berühmte Minimumprinzip zu formulieren.

Definition 2.4.5. Für das Optimalsteuerungsproblem (2.2.1) ist die *Hamilton-Funktion* $\mathcal{H} : \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$ definiert durch

$$\mathcal{H}(x, u, \lambda, l_0) = l_0L(x, u) + \lambda^\top f(x, u).$$

Definition 2.4.6. Die Hamilton-Funktion ist *regulär*, wenn es für alle $t \in [0, T]$ in einer Umgebung von $(\hat{x}(t), \lambda(t), l_0)$ eine eindeutig bestimmte Steuerung \hat{u} gibt, welche die Hamilton-Funktion bzgl. u minimiert.

Satz 2.4.7 (Lokales Minimumprinzip).

Gegeben sei ein unbeschränktes Optimalsteuerungsproblem gemäß Definition 2.2.1, d.h. ohne Steuer- und Zustandsbeschränkungen der Form (2.2.4), wobei die Funktionen L , φ , f und ψ stetig in allen Argumenten und ferner stetig differenzierbar in x und u seien. $\mathcal{U} \subseteq \mathbb{R}^{n_u}$ sei eine abgeschlossene, konvexe Menge, deren Inneres nicht leer sei.

Ist $(\hat{x}, \hat{u}) \in W^{1,\infty}([0, T], \mathbb{R}^n) \times L^\infty([0, T], \mathbb{R}^{n_u})$ ein schwaches lokales Minimum des Optimalsteuerungsproblems, dann gibt es Multiplikatoren $l_0 \in \mathbb{R}$, $\mu \in \mathbb{R}^{n_\psi}$ und $\lambda \in W^{1,\infty}([0, T], \mathbb{R}^n)$, so dass die folgenden Bedingungen gelten:

(i) $l_0 \geq 0$ und das Tupel (l_0, σ, λ) verschwindet nicht,

(ii) die **Adjungiertengleichungen**

$$\dot{\lambda}(t) = -(\mathcal{H}'_x(\hat{x}(t), \hat{u}(t), \lambda(t), l_0))^\top \quad t \in [0, T],$$

(iii) die **Transversalitätsbedingungen**

$$\begin{pmatrix} -\lambda(0) \\ \lambda(T) \end{pmatrix} = \begin{pmatrix} (l_0 \varphi'_{x_0} + \mu^\top \psi'_{x_0})^\top \\ (l_0 \varphi'_{x_e} + \mu^\top \psi'_{x_e})^\top \end{pmatrix}$$

(iv) und die **Optimalitätsbedingung**, dass fast überall in $[0, T]$ für alle $u \in \mathcal{U}$

$$\mathcal{H}'_u(\hat{x}(t), \hat{u}(t), \lambda(t), l_0)(u - \hat{u}(t)) \geq 0$$

gilt.

Beweis. Die Existenz von $\lambda \in W^{1,\infty}([0, T], \mathbb{R}^n)$ ist durch Satz 2.4.4 gesichert, so dass wir (2.4.6) aufschreiben können als

$$\lambda^\top(t)\Phi(t) = (l_0 \varphi'_{x_e} + \mu^\top \psi'_{x_e})\Phi(T) + \int_t^T l_0 L'_x[\tau]\Phi(\tau) \, d\tau.$$

Wir differenzieren die Gleichung bzgl. t und erhalten

$$\dot{\lambda}^\top(t)\Phi(t) + \lambda^\top(t)\dot{\Phi}(t) = -l_0 L'_x[t]\Phi(t).$$

Unter Ausnutzung von

$$\begin{aligned} \dot{\Phi}(t) &= f'_x[t]\Phi(t) & t \in [0, T] \\ \Phi(0) &= I_n \end{aligned}$$

aus dem Beweis von Satz 2.4.4 erhalten wir

$$\dot{\lambda}^\top(t)\Phi(t) = -(l_0 L'_x[t] + \lambda^\top(t)f'_x[t])\Phi(t)$$

$$= -\mathcal{H}'_x(\hat{x}(t), \hat{u}(t), \lambda(t), l_0)\Phi(t) =: -\mathcal{H}'_x[t]\Phi(t)$$

und somit die Adjungiertengleichungen.

Als Nächstes setzen wir (2.4.1) in (2.4.2) ein und es ergibt sich

$$\begin{aligned} 0 = & (l_0\varphi'_{x_0} + \mu^\top\psi'_{x_0})\delta x(0) + (l_0\varphi'_{x_e} + \mu^\top\psi'_{x_e})\delta x(T) \\ & + \int_0^T \mathcal{H}'_x[t]\delta x(t) - \lambda^\top(t)\delta\dot{x}(t) dt \end{aligned} \quad (2.4.7)$$

für alle $\delta x \in W^{1,\infty}([0, T], \mathbb{R}^n)$. Partielle Integration liefert

$$\int_0^T \lambda^\top(t)\delta\dot{x}(t) dt = [\lambda^\top(t)\delta x(t)]_0^T - \int_0^T \dot{\lambda}^\top(t)\delta x(t) dt$$

und wir können damit Gleichung (2.4.7) wie folgt aufschreiben:

$$\begin{aligned} 0 = & (l_0\varphi'_{x_0} + \mu^\top\psi'_{x_0} + \lambda^\top(0))\delta x(0) + (l_0\varphi'_{x_e} + \mu^\top\psi'_{x_e} - \lambda^\top(T))\delta x(T) \\ & + \int_0^T (\mathcal{H}'_x[t] + \dot{\lambda}^\top(t))\delta x(t) dt. \end{aligned}$$

Der Integrand ist durch die Adjungiertengleichungen fast überall 0. Da wir δx beliebig wählen können, müssen die beiden Summanden Null sein. Daraus folgen die Transversalitätsbedingungen.

Wir definieren uns die absolutstetigen Funktionen

$$\begin{aligned} h_1(t) &:= \int_0^t \mathcal{H}'_u[\tau] d\tau \\ h_2(t) &:= \int_0^t \mathcal{H}'_u[\tau]\hat{u}(\tau) d\tau. \end{aligned}$$

Aus der absoluten Stetigkeit folgt, dass die Funktionen auch fast überall differenzierbar sind. Somit gilt für fast alle $t \in [0, T]$

$$\begin{aligned} \lim_{\varepsilon \searrow 0} \frac{h_1(t+\varepsilon) - h_1(t)}{\varepsilon} &= \lim_{\varepsilon \searrow 0} \frac{1}{\varepsilon} \int_t^{t+\varepsilon} \mathcal{H}'_u[\tau] d\tau = \mathcal{H}'_u[t] \\ \lim_{\varepsilon \searrow 0} \frac{h_2(t+\varepsilon) - h_2(t)}{\varepsilon} &= \lim_{\varepsilon \searrow 0} \frac{1}{\varepsilon} \int_t^{t+\varepsilon} \mathcal{H}'_u[\tau]\hat{u}(\tau) d\tau = \mathcal{H}'_u[t]\hat{u}(t). \end{aligned} \quad (2.4.8)$$

Wir setzen (2.4.1) in (2.4.3) ein. Dies liefert

$$\int_0^T \mathcal{H}'_u[t](u(t) - \hat{u}(t)) dt \geq 0. \quad (2.4.9)$$

Sei $u \in \mathcal{U}$ und $\varepsilon > 0$. Wir definieren die Steuerung

$$u_\varepsilon(\tau) = \begin{cases} u & \tau \in [t, t + \varepsilon] \\ \hat{u}(\tau) & \text{sonst} \end{cases}.$$

Damit ist $u_\varepsilon \in U_{ad}$. Wir setzen u_ε für $u(\cdot)$ in (2.4.9) ein und erhalten

$$0 \leq \frac{1}{\varepsilon} \int_0^T \mathcal{H}'_u[\tau](u_\varepsilon(\tau) - \hat{u}(\tau)) d\tau = \frac{1}{\varepsilon} \int_t^{t+\varepsilon} \mathcal{H}'_u[\tau](u - \hat{u}(\tau)) d\tau.$$

$\varepsilon \searrow 0$ liefert dann für fast alle $t \in [0, T]$ die Optimalitätsbedingung. \square

Für $\mathcal{U} = \mathbb{R}^{n_u}$ impliziert die Optimalitätsbedingung für fast alle $t \in [0, T]$

$$\mathcal{H}'_u(\hat{x}(t), \hat{u}(t), \lambda(t), l_0) = 0.$$

Wir zeigen nun, dass der Multiplikator l_0 unter gewissen Voraussetzungen redundant ist.

Lemma 2.4.8. Sei $\Phi : \mathbb{R} \rightarrow \mathbb{R}^n$ das Fundamentalsystem der homogenen linearen Differentialgleichung

$$\begin{aligned} \dot{\Phi}(t) &= f'_x[t]\Phi(t) & t \in [0, T] \\ \Phi(0) &= I_n \end{aligned}$$

und sei

$$\text{Rang}(\psi'_{x_0}\Phi(0) + \psi'_{x_e}\Phi(T)) = n_\psi.$$

Dann ist $H'(\hat{x}, \hat{u})$ surjektiv.

Beweis. Seien $h_1 \in L^\infty([0, T], \mathbb{R})$ und $h_2 \in \mathbb{R}^n$ beliebig gegeben. Wir betrachten das lineare Randwertproblem

$$\begin{aligned} H'_1(\hat{x}, \hat{u})(\delta x, \delta u)(t) &= h_1(t) & \text{f.ü. in } [0, T] \\ H'_2(\hat{x}, \hat{u})(\delta x, \delta u) &= h_2. \end{aligned}$$

Gemäß Definition 2.2.3 bedeutet dies

$$\begin{aligned} f'_x[t]\delta x(t) + f'_u[t]\delta u(t) - \dot{\delta x}(t) &= h_1(t) & \text{f.ü. in } [0, T] \\ -\psi'_{x_0}\delta x(0) - \psi'_{x_e}\delta x(T) &= h_2. \end{aligned}$$

Dieses Randwertproblem hat stets eine Lösung, da durch die Rangvoraussetzung Satz 2.1.12 benutzt werden kann. Zusammen mit der freien Wahl von h_1 und h_2 haben wir die Surjektivität gezeigt. \square

Satz 2.4.9 (Mangasarian-Fromowitz-Bedingung).

Es seien die Voraussetzungen von Lemma 2.4.8 und Satz 2.4.7 erfüllt. Ferner existiere $\delta x \in W^{1,\infty}([0, T], \mathbb{R}^n)$, $\delta u \in \text{int}(U_{ad} - \{\hat{u}\})$ mit U_{ad} aus Definition 2.2.3 und

$$\begin{aligned} f'_x[t]\delta x(t) + f'_u[t]\delta u(t) - \dot{\delta}x(t) &= 0_n && \text{f.ü. in } [0, T] \\ \psi'_{x_0}\delta x(0) + \psi'_{x_e}\delta x(T) &= 0_{n_\psi}. \end{aligned}$$

Dann gilt in Satz 2.4.7 $l_0 = 1$.

Beweis. Wir zeigen, dass die Voraussetzungen von Satz 2.3.5 erfüllt sind. Die Surjektivität von $H'(\hat{x}, \hat{u})$ folgt aus Lemma 2.4.8. Die zweite Voraussetzung folgt aus den Bedingungen an δx und δu , wobei wir in Satz 2.3.5 $\hat{d} = (\hat{x}, \hat{u})$ setzen. \square

In der Praxis ist es meist einfacher zu zeigen, dass $l_0 = 0$ zu einem Widerspruch führt. Gilt dies, so kann o.B.d.A. $l_0 = 1$ gewählt werden. Wir werden deshalb im Folgenden l_0 nicht mehr als Parameter der Hamilton-Funktion benutzen.

In dem nächsten Kapitel werden wir die Bedingungen aus Satz 2.4.7 nutzen, um einen neuen Ansatz zum Lösen von unbeschränkten Optimalsteuerungsproblemen herzuleiten.

3. Homotopie-Methoden

3.1. Variation von T

Wir betrachten unbeschränkte Optimalsteuerungsprobleme der Form

$$\begin{aligned} \min_u J(u) &= \min_u \int_0^T L(x(t), u(t)) dt + \varphi(S, x(0), x(T)) \quad S, T \in \mathbb{R} \\ \dot{x}(t) &= f(x(t), u(t)) \quad t \in [0, T] \\ \psi(S, x(0), x(T)) &= 0_{n_\psi}. \end{aligned} \quad (3.1.1)$$

$S \in [S_0, S_e]$ und $T \in [T_0, T_e]$ seien beliebig aber fest. Die Dimensionen der einzelnen Funktionen sind wie im vorigen Kapitel definiert. Nun benutzen wir die indirekte Methode um das Optimalsteuerungsproblem zu lösen und stellen dazu die Hamilton-Funktion gemäß Definition 2.4.5 auf:

$$\mathcal{H}(x, u, \lambda) = L(x, u) + \lambda^\top f(x, u).$$

\mathcal{H} sei regulär, so dass die optimale Steuerung \hat{u} eindeutig aus

$$\hat{u}(t) = \arg \min_u \mathcal{H}(x(t), u, \lambda(t)) \quad t \in [0, T]$$

bestimmt werden kann. Wir stellen dann die notwendigen Bedingungen aus Satz 2.4.7 auf, wobei wir bei $x(\cdot)$, $\lambda(\cdot)$ und $\hat{u}(\cdot)$ das Argument t weglassen. Ferner verzichten wir bei den partiellen Ableitungen von $\varphi(\cdot, \cdot, \cdot)$ und $\psi(\cdot, \cdot, \cdot)$ auf die Argumente. Wir erhalten

$$\begin{aligned} \dot{x} &= \left(\frac{\partial \mathcal{H}(x, \hat{u}, \lambda)}{\partial \lambda} \right)^\top = f(x, \hat{u}) \quad \psi(x(0), x(T), S) = 0_{n_\psi} \\ \dot{\lambda} &= - \left(\frac{\partial \mathcal{H}(x, \hat{u}, \lambda)}{\partial x} \right)^\top \quad \begin{pmatrix} -\lambda(0) \\ \lambda(T) \end{pmatrix} = \begin{pmatrix} (\varphi_{x_0} + \mu^\top \psi_{x_0})^\top \\ (\varphi_{x_e} + \mu^\top \psi_{x_e})^\top \end{pmatrix} \\ \begin{pmatrix} \dot{x} \\ \dot{\lambda} \end{pmatrix} &= \begin{pmatrix} F(x, \lambda) \\ G(x, \lambda) \end{pmatrix} = \mathcal{X}(x, \lambda). \end{aligned} \quad (3.1.2)$$

Dies ist ein Randwertproblem, welches man durch Kollokations- oder Mehrfachschießverfahren lösen kann. Die *indirekten Verfahren*, welche in Abschnitt 4.2 besprochen werden, basieren auf dem Auswerten dieser notwendigen Bedingungen.

Ziel des folgenden Ansatzes ist es die Anfangs- und Endwerte von x und λ zu bestimmen. Kennen wir diese Werte, so haben wir nur noch ein reines Anfangswert- oder Endwertproblem, welches numerisch einfacher zu lösen ist.

Der neue Ansatz basiert auf den Arbeiten von Costanza [5, 6], wobei wir auf die Unterschiede seiner und unserer Methodik im letzten Abschnitt 3.4 dieses Kapitels eingehen werden.

Definition 3.1.1. Die Funktion $\phi : \mathbb{R} \times [0, T] \times \mathcal{O}_x \times \mathcal{O}_\lambda \rightarrow \mathbb{R}^{2n}$, welche das Anfangswertproblem

$$\begin{aligned} \frac{\partial}{\partial t} \phi(S, t, x_0, \lambda_0) &= \begin{pmatrix} \frac{\partial}{\partial t} \phi_1(S, t, x_0, \lambda_0) \\ \frac{\partial}{\partial t} \phi_2(S, t, x_0, \lambda_0) \end{pmatrix} = \begin{pmatrix} F(\phi^\top(S, t, x_0, \lambda_0)) \\ G(\phi^\top(S, t, x_0, \lambda_0)) \end{pmatrix} & \forall t \in [0, T] \\ &= \mathcal{X}(\phi^\top(S, t, x_0, \lambda_0)) & (3.1.3) \\ \phi(S, 0, x_0, \lambda_0) &= \begin{pmatrix} \phi_1(S, 0, x_0, \lambda_0) \\ \phi_2(S, 0, x_0, \lambda_0) \end{pmatrix} = \begin{pmatrix} x_0 \\ \lambda_0 \end{pmatrix} & \forall \begin{pmatrix} x_0 \\ \lambda_0 \end{pmatrix} \in \mathcal{O}_x \times \mathcal{O}_\lambda \end{aligned}$$

löst, heißt *Flussfunktion*. Dabei sei $\mathcal{O}_x \times \mathcal{O}_\lambda \subseteq \mathbb{R}^{2n}$ die Menge aller zulässigen Anfangswertpaare (x_0, λ_0) .

Offensichtlich gilt für die Flussfunktion

$$\frac{\partial}{\partial S} \phi(S, \cdot, \cdot, \cdot) = 0_{2n}. \quad \forall S \in \mathbb{R} \quad (3.1.4)$$

Somit könnten wir S als Argument weglassen. Wir werden aber in Abschnitt 3.3 auch Beispiele betrachten, wo S explizit in den notwendigen Bedingungen auftaucht. Dementsprechend behalten wir für eine konsistente Notation der Flussfunktion S als Argument bei.

Definition 3.1.2. Für eine Schar von Optimalsteuerungsproblemen der Form (3.1.1) beschreiben $\rho^0(S, T)$ und $\rho^e(S, T)$ die Anfangs- und Endwerte von x bei Wahl der optimalen Steuerung in Abhängigkeit von S und T . $\sigma^0(S, T)$ und $\sigma^e(S, T)$ beschreiben die Anfangs- und Endwerte der Adjungierten λ bei Wahl der optimalen Steuerung. Die Funktionen bilden dabei jeweils von dem \mathbb{R}^2 in den \mathbb{R}^n ab.

Es gilt also

$$\begin{aligned} \phi(S, 0, \rho^0(S, T), \sigma^0(S, T)) &= \begin{pmatrix} \rho^0(S, T) \\ \sigma^0(S, T) \end{pmatrix} \\ \phi(S, T, \rho^0(S, T), \sigma^0(S, T)) &= \begin{pmatrix} \rho^e(S, T) \\ \sigma^e(S, T) \end{pmatrix}. \end{aligned} \quad (3.1.5)$$

Wir leiten nun die Flussfunktion zum Zeitpunkt T total nach T ab und erhalten unter Ausnutzung der Gleichung (3.1.3):

$$\begin{aligned}
\frac{d}{dT}\phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) &= \frac{\partial}{\partial t}\phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) \\
&+ \frac{\partial}{\partial x_0}\phi_1(S, T, \rho^0(S, T), \sigma^0(S, T))\frac{\partial}{\partial T}\rho^0(S, T) \\
&+ \frac{\partial}{\partial \lambda_0}\phi_1(S, T, \rho^0(S, T), \sigma^0(S, T))\frac{\partial}{\partial T}\sigma^0(S, T) \\
&= F(\rho^e(S, T), \sigma^e(S, T)) \\
&+ \frac{\partial}{\partial x_0}\phi_1(S, T, \rho^0(S, T), \sigma^0(S, T))\rho_T^0(S, T) \\
&+ \frac{\partial}{\partial \lambda_0}\phi_1(S, T, \rho^0(S, T), \sigma^0(S, T))\sigma_T^0(S, T).
\end{aligned}$$

Andererseits ergibt sich durch Gleichung (3.1.5)

$$\frac{d}{dT}\phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) = \rho_T^e(S, T).$$

Wir fassen beide Gleichungen zusammen, gehen analog für ϕ_2 vor und lassen dabei im Folgenden die Argumente für die partiellen Ableitungen von ϕ weg, so dass

$$\begin{aligned}
\rho_T^e(S, T) &= F(\rho^e(S, T), \sigma^e(S, T)) + \frac{\partial\phi_1}{\partial x_0}\rho_T^0(S, T) + \frac{\partial\phi_1}{\partial \lambda_0}\sigma_T^0(S, T) \\
\sigma_T^e(S, T) &= G(\rho^e(S, T), \sigma^e(S, T)) + \frac{\partial\phi_2}{\partial x_0}\rho_T^0(S, T) + \frac{\partial\phi_2}{\partial \lambda_0}\sigma_T^0(S, T).
\end{aligned} \tag{3.1.6}$$

Um die Gleichungen zu lösen, benötigen wir die partiellen Ableitungen von ϕ nach den Anfangswerten. Zunächst definieren wir uns eine Matrixfunktion, welche uns die partiellen Ableitungen angibt.

Definition 3.1.3. Die Funktion $V : \mathbb{R} \times [0, T] \times \mathcal{O}_x \times \mathcal{O}_\lambda \rightarrow \mathbb{R}^{2n \times 2n}$ ist definiert durch

$$\begin{aligned}
V(S, t, x_0, \lambda_0) &:= D_{x_0, \lambda_0}\phi(S, t, x_0, \lambda_0) \\
&= \begin{pmatrix} \frac{\partial}{\partial x_0}\phi_1(S, t, x_0, \lambda_0) & \frac{\partial}{\partial \lambda_0}\phi_1(S, t, x_0, \lambda_0) \\ \frac{\partial}{\partial x_0}\phi_2(S, t, x_0, \lambda_0) & \frac{\partial}{\partial \lambda_0}\phi_2(S, t, x_0, \lambda_0) \end{pmatrix} \\
&= \begin{pmatrix} V_1(S, t, x_0, \lambda_0) & V_2(S, t, x_0, \lambda_0) \\ V_3(S, t, x_0, \lambda_0) & V_4(S, t, x_0, \lambda_0) \end{pmatrix}.
\end{aligned} \tag{3.1.7}$$

Wie bei Definition 3.1.1 hängt hier V nicht von S ab. Aus Gründen der Notation behalten wir aber S als Parameter bei, da wir später Aufgaben betrachten werden, wo S Einfluss auf die Funktion hat.

Nun differenzieren wir Gleichung (3.1.3) bezüglich $v := (x_0, \lambda_0)$ und vertauschen die Differentialoperatoren, wobei wir die zweimal stetige Differenzierbarkeit der Flussfunktion voraussetzen, d.h.

$$\begin{aligned} \frac{\partial}{\partial v} \frac{\partial}{\partial t} \phi(S, t, v) &= \frac{\partial}{\partial v} X(\phi^\top(S, t, v)) \\ \Leftrightarrow \frac{\partial}{\partial t} \frac{\partial}{\partial v} \phi(S, t, v) &= D_{x,\lambda} X(x(t), \lambda(t)) \frac{\partial}{\partial v} \phi(S, t, v). \end{aligned}$$

Wir setzen V aus Definition 3.1.3 in die untere Gleichung ein und beachten, dass $\phi(\cdot, 0, v) = v$. Dies impliziert das Anfangswertproblem

$$\begin{aligned} \dot{V}(S, t, v) &= D_{x,\lambda} X(\phi^\top(S, t, v)) V(S, t, v) \quad t \in [0, T] \\ V(S, 0, v) &= I_{2n}, \end{aligned} \quad (3.1.8)$$

welches simultan mit dem Anfangswertproblem

$$\begin{aligned} \begin{pmatrix} \dot{x}(t) \\ \dot{\lambda}(t) \end{pmatrix} &= \begin{pmatrix} F(x(t), \lambda(t)) \\ G(x(t), \lambda(t)) \end{pmatrix} \\ \begin{pmatrix} x(0) \\ \lambda(0) \end{pmatrix} &= \begin{pmatrix} \rho^0(S, T) \\ \sigma^0(S, T) \end{pmatrix} \end{aligned} \quad (3.1.9)$$

gelöst wird.

Damit können wir die Gleichungen (3.1.6) mit Hilfe von V aufschreiben, wobei wir die Argumente von V im Folgenden weglassen:

$$\begin{aligned} \rho_T^e(S, T) &= F(\rho^e(S, T), \sigma^e(S, T)) + V_1 \rho_T^0(S, T) + V_2 \sigma_T^0(S, T) \\ \sigma_T^e(S, T) &= G(\rho^e(S, T), \sigma^e(S, T)) + V_3 \rho_T^0(S, T) + V_4 \sigma_T^0(S, T). \end{aligned} \quad (3.1.10)$$

Dies ist ein lineares, implizites System mit $2n$ Gleichungen und $4n$ gesuchten Funktionen, welches in dieser Form nicht eindeutig gelöst werden kann. Allerdings haben wir durch die Randbedingungen und die Transversalitätsbedingungen weitere $2n + n_\psi$ Gleichungen gegeben, so dass die Funktionen bestimmt werden können. Dazu setzen wir ρ^0 und ρ^e in die Randgleichungen $\psi(S, x(0), x(T)) = 0$ ein und leiten sie total nach T ab, so dass

$$D_{x_0, x_e} \psi(S, \rho^0(S, T), \rho^e(S, T)) \begin{pmatrix} \rho_T^0(S, T) \\ \rho_T^e(S, T) \end{pmatrix} = 0_{n_\psi}. \quad (3.1.11)$$

Wir gehen analog für die Transversalitätsbedingungen vor. Dabei ist aber zu beachten, dass die Multiplikatoren μ auch von S und T abhängen können.

Definition 3.1.4. Für eine Schar von Optimalsteuerungsproblemen der Form (3.1.1) beschreibt $\hat{\mu}(S, T)$ die Multiplikatoren in den Transversalitätsbedingungen bei Wahl der optimalen Steuerung in Abhängigkeit von S und T . Analog zu Definition 3.1.2 bildet die Funktion von dem \mathbb{R}^2 in den \mathbb{R}^{n_ψ} ab.

Zur einfacheren Darstellung definieren wir uns zuerst die Vektorfunktion \mathcal{K} .

Definition 3.1.5. Die Funktion $\mathcal{K} : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{n_\psi} \rightarrow \mathbb{R}^{2n}$ ist definiert durch

$$\begin{aligned} \mathcal{K}(S, x_0, x_e, \mu) &:= \begin{pmatrix} (\varphi_{x_0}(S, x_0, x_e) + \mu^\top \psi_{x_0}(S, x_0, x_e))^\top \\ (\varphi_{x_e}(S, x_0, x_e) + \mu^\top \psi_{x_e}(S, x_0, x_e))^\top \end{pmatrix} \\ &= \left(D_{x_0, x_e} \left(\varphi(S, x_0, x_e) + \mu^\top \psi(S, x_0, x_e) \right) \right)^\top. \end{aligned}$$

Wir setzen ρ^0, ρ^e und $\hat{\mu}$ in \mathcal{K} ein, leiten die Transversalitätsbedingungen (3.1.2) total nach T ab und erhalten

$$\begin{aligned} \begin{pmatrix} -\sigma_T^0(S, T) \\ \sigma_T^e(S, T) \end{pmatrix} &= D_{x_0, x_e} \mathcal{K}(S, \rho^0(S, T), \rho^e(S, T), \hat{\mu}(S, T)) \begin{pmatrix} \rho_T^0(S, T) \\ \rho_T^e(S, T) \end{pmatrix} \\ &+ \left(D_{x_0, x_e} \psi(S, \rho^0(S, T), \rho^e(S, T)) \right)^\top \hat{\mu}_T(S, T). \end{aligned} \quad (3.1.12)$$

Damit haben wir zusammen mit dem linearen, impliziten System (3.1.10) $4n + n_\psi$ Gleichungen mit $4n + n_\psi$ gesuchten Funktionen. Die Werte für $\rho^0(\cdot, T_0), \rho^e(\cdot, T_0), \sigma^0(\cdot, T_0), \sigma^e(\cdot, T_0)$ und $\hat{\mu}(\cdot, T_0)$ sind in (3.1.1) gegeben oder müssen analytisch bzw. numerisch bestimmt werden.

Somit können wir die Anfangs- und Endwerte durch die folgenden linearen, impliziten Systeme berechnen, wobei wir zur Vereinfachung der Notation die Argumente (S, T) bei $\rho^0(\cdot, \cdot), \rho^e(\cdot, \cdot), \sigma^0(\cdot, \cdot), \sigma^e(\cdot, \cdot), \hat{\mu}(\cdot, \cdot)$ und deren Ableitungen weglassen:

$$\begin{aligned} \begin{pmatrix} \rho_T^e \\ \sigma_T^e \end{pmatrix} &= \begin{pmatrix} F(\rho^e, \sigma^e) \\ G(\rho^e, \sigma^e) \end{pmatrix} + V(S, T, \rho^0, \sigma^0) \begin{pmatrix} \rho_T^0 \\ \sigma_T^0 \end{pmatrix} \quad T \in [T_0, T_e] \\ 0_{n_\psi} &= D_{x_0, x_e} \psi(S, \rho^0, \rho^e) \begin{pmatrix} \rho_T^0 \\ \rho_T^e \end{pmatrix} \\ \begin{pmatrix} -\sigma_T^0 \\ \sigma_T^e \end{pmatrix} &= D_{x_0, x_e} \mathcal{K}(S, \rho^0, \rho^e, \hat{\mu}) \begin{pmatrix} \rho_T^0 \\ \rho_T^e \end{pmatrix} \\ &+ \left(D_{x_0, x_e} \psi(S, \rho^0, \rho^e) \right)^\top \hat{\mu}_T \\ \dot{V}(S, t, \rho^0, \sigma^0) &= D_{x, \lambda} \mathcal{X}(x(t), \lambda(t)) V(S, t, \rho^0, \sigma^0) \quad t \in [0, T] \\ V(S, 0, \rho^0, \sigma^0) &= I_{2n} \\ \begin{pmatrix} \dot{x}(t) \\ \dot{\lambda}(t) \end{pmatrix} &= \begin{pmatrix} F(x(t), \lambda(t)) \\ G(x(t), \lambda(t)) \end{pmatrix} \\ \begin{pmatrix} x(0) \\ \lambda(0) \end{pmatrix} &= \begin{pmatrix} \rho^0 \\ \sigma^0 \end{pmatrix}. \end{aligned} \quad (3.1.13)$$

Bei der numerischen Umsetzung ist zu beachten, dass wir für jeden Zeitschritt in den linearen, impliziten Systemen (3.1.10), (3.1.11) und (3.1.12) das Matrixdifferentialgleichungssystem (3.1.8) sowie die Zustands- und Adjungiertengleichungen (3.1.9) lösen müssen. Darauf werden wir in Kapitel 4 näher eingehen.

3.2. Variation von S

Im vorigen Abschnitt haben wir die Ableitungen nach T betrachtet. Es kann aber auch sinnvoll sein S zu variieren. Dazu leiten wir die Flussfunktion zum Zeitpunkt T total nach S ab und nutzen dabei die Gleichungen (3.1.4) und (3.1.7) aus. Dies ergibt

$$\begin{aligned} \frac{d}{dS} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) &= \frac{\partial}{\partial S} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) \\ &+ \frac{\partial}{\partial x_0} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) \frac{\partial}{\partial S} \rho^0(S, T) \\ &+ \frac{\partial}{\partial \lambda_0} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) \frac{\partial}{\partial S} \sigma^0(S, T) \\ &= 0_n + V_1 \rho_S^0(S, T) + V_2 \sigma_S^0(S, T). \end{aligned}$$

Gleichzeitig erhalten wir durch Gleichung (3.1.5)

$$\frac{d}{dS} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) = \rho_S^e(S, T).$$

Wir fassen beide Gleichungen zusammen und folgern bei gleicher Vorgehensweise für ϕ_2

$$\begin{aligned} \rho_S^e(S, T) &= V_1 \rho_S^0(S, T) + V_2 \sigma_S^0(S, T) \\ \sigma_S^e(S, T) &= V_3 \rho_S^0(S, T) + V_4 \sigma_S^0(S, T). \end{aligned} \quad (3.2.1)$$

Bei den Rand- und Transversalitätsbedingungen müssen wir beachten, dass S auch explizit in φ bzw. ψ auftreten kann. Für die Randgleichungen ergibt sich

$$\begin{aligned} 0_{n_\psi} &= D_{x_0, x_e} \psi(S, \rho^0(S, T), \rho^e(S, T)) \begin{pmatrix} \rho_S^0(S, T) \\ \rho_S^e(S, T) \end{pmatrix} \\ &+ \psi_S(S, \rho^0(S, T), \rho^e(S, T)). \end{aligned} \quad (3.2.2)$$

Wir gehen analog bei den Transversalitätsbedingungen (3.1.2) vor und erhalten

$$\begin{aligned} \begin{pmatrix} -\sigma_S^0(S, T) \\ \sigma_S^e(S, T) \end{pmatrix} &= D_{x_0, x_e} \mathcal{K}(S, \rho^0(S, T), \rho^e(S, T), \hat{\mu}(S, T)) \begin{pmatrix} \rho_S^0(S, T) \\ \rho_S^e(S, T) \end{pmatrix} \\ &+ (D_{x_0, x_e} \psi(S, \rho^0(S, T), \rho^e(S, T)))^\top \hat{\mu}_S(S, T) \\ &+ \mathcal{K}_S(S, \rho^0(S, T), \rho^e(S, T), \hat{\mu}(S, T)). \end{aligned} \quad (3.2.3)$$

Insgesamt sind also die folgenden linearen Systeme zu lösen:

$$\begin{aligned}
\begin{pmatrix} \rho_S^e \\ \sigma_S^e \end{pmatrix} &= V(S, T, \rho^0, \sigma^0) \begin{pmatrix} \rho_S^0 \\ \sigma_S^0 \end{pmatrix} && S \in [S_0, S_e] \\
0_{n_\psi} &= D_{x_0, x_e} \psi(S, \rho^0, \rho^e) \begin{pmatrix} \rho_S^0 \\ \rho_S^e \end{pmatrix} + \psi_S(S, \rho^0, \rho^e) \\
\begin{pmatrix} -\sigma_S^0 \\ \sigma_S^e \end{pmatrix} &= D_{x_0, x_e} \mathcal{K}(S, \rho^0, \rho^e, \hat{\mu}) \begin{pmatrix} \rho_S^0 \\ \rho_S^e \end{pmatrix} \\
&\quad + (D_{x_0, x_e} \psi(S, \rho^0, \rho^e))^T \hat{\mu}_S \\
&\quad + \mathcal{K}_S(S, \rho^0, \rho^e, \hat{\mu}) && (3.2.4) \\
\dot{V}(S, t, \rho^0, \sigma^0) &= D_{x, \lambda} \mathcal{X}(x(t), \lambda(t)) V(S, t, \rho^0, \sigma^0) && t \in [0, T] \\
V(S, 0, \rho^0, \sigma^0) &= I_{2n} \\
\begin{pmatrix} \dot{x}(t) \\ \dot{\lambda}(t) \end{pmatrix} &= \begin{pmatrix} F(x(t), \lambda(t)) \\ G(x(t), \lambda(t)) \end{pmatrix} \\
\begin{pmatrix} x(0) \\ \lambda(0) \end{pmatrix} &= \begin{pmatrix} \rho^0 \\ \sigma^0 \end{pmatrix}.
\end{aligned}$$

Diese Alternative zu dem Verfahren (3.1.13) aus Abschnitt 3.1 kann zum Beispiel genutzt werden bei Aufgaben mit fest vorgegebenen Anfangs- und Endwerten, bei denen T nicht variiert wird oder wo der vorgegebene Endwert von S abhängt. Die Werte für $\rho^0(S_0, \cdot)$, $\rho^e(S_0, \cdot)$, $\sigma^0(S_0, \cdot)$ und $\sigma^e(S_0, \cdot)$ sind wie im vorigen Abschnitt beschrieben entweder durch die Randbedingungen gegeben oder müssen noch analytisch oder numerisch bestimmt werden.

3.3. Verallgemeinerung der Parameterabhängigkeit

Wir betrachten in diesem Abschnitt eine verallgemeinerte Version des Optimalsteuerungsproblems (3.1.1), wobei es weiterhin unbeschränkt ist:

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^T L(x(t), u(t), S) dt + \varphi(S, x(0), x(T)) \quad S, T \in \mathbb{R} \\ \dot{x}(t) &= f(x(t), u(t), S) \quad t \in [0, T] \\ \psi(S, x(0), x(T)) &= 0_{n_\psi}. \end{aligned} \quad (3.3.1)$$

Wie in Abschnitt 3.2 sei $S \in [S_0, S_e]$ beliebig aber fest und T sei fixiert. Durch die verallgemeinerte Formulierung kann S in der Hamilton-Funktion explizit vorkommen, d.h.

$$\begin{aligned} \mathcal{H}(x, \lambda, u, S) &= L(x, u, S) + \lambda^\top f(x, u, S) \\ \hat{u}(t) &= \arg \min_u \mathcal{H}(x(t), u, \lambda(t), S). \quad t \in [0, T] \end{aligned}$$

Da S in der Hamilton-Funktion explizit vorkommt, hängen auch die Adjungiertengleichungen von S ab, insbesondere nicht nur die Anfangs- und Endwerte der Trajektorie sondern auch der Verlauf der Trajektorie wird explizit von S beeinflusst:

$$\begin{aligned} \dot{x} &= \left(\frac{\partial \mathcal{H}(x, \hat{u}, \lambda, S)}{\partial \lambda} \right)^\top = f(x, \hat{u}, S) \quad \psi(x(0), x(T), S) = 0_{n_\psi} \\ \dot{\lambda} &= - \left(\frac{\partial \mathcal{H}(x, \hat{u}, \lambda, S)}{\partial x} \right)^\top \quad \begin{pmatrix} -\lambda(0) \\ \lambda(T) \end{pmatrix} = \begin{pmatrix} (\varphi_{x_0} + \mu^\top \psi_{x_0})^\top \\ (\varphi_{x_e} + \mu^\top \psi_{x_e})^\top \end{pmatrix} \\ \begin{pmatrix} \dot{x} \\ \dot{\lambda} \end{pmatrix} &= \begin{pmatrix} F(x, \lambda, S) \\ G(x, \lambda, S) \end{pmatrix} = \mathcal{X}(x, \lambda, S). \end{aligned}$$

Definition 3.3.1. Die Funktion $\phi : \mathbb{R} \times [0, T] \times \mathcal{O}_x \times \mathcal{O}_\lambda \rightarrow \mathbb{R}^{2n}$, welche das Anfangswertproblem

$$\begin{aligned} \frac{\partial}{\partial t} \phi(S, t, x_0, \lambda_0) &= \begin{pmatrix} \frac{\partial}{\partial t} \phi_1(S, t, x_0, \lambda_0) \\ \frac{\partial}{\partial t} \phi_2(S, t, x_0, \lambda_0) \end{pmatrix} = \begin{pmatrix} F(\phi^\top(S, t, x_0, \lambda_0), S) \\ G(\phi^\top(S, t, x_0, \lambda_0), S) \end{pmatrix} \quad \forall t \in [0, T] \\ &= \mathcal{X}(\phi^\top(S, t, x_0, \lambda_0), S) \quad (3.3.2) \\ \phi(S, 0, x_0, \lambda_0) &= \begin{pmatrix} \phi_1(S, 0, x_0, \lambda_0) \\ \phi_2(S, 0, x_0, \lambda_0) \end{pmatrix} = \begin{pmatrix} x_0 \\ \lambda_0 \end{pmatrix} \quad \forall \begin{pmatrix} x_0 \\ \lambda_0 \end{pmatrix} \in \mathcal{O}_x \times \mathcal{O}_\lambda \end{aligned}$$

löst, heißt *Flussfunktion*. Dabei sei $\mathcal{O}_x \times \mathcal{O}_\lambda \subseteq \mathbb{R}^{2n}$ die Menge aller zulässigen Anfangswertpaare (x_0, λ_0) .

Im Gegensatz zur Flussfunktion in Definition 3.1.1 hat hier S durch die notwendigen Bedingungen Einfluss auf ϕ und somit verschwindet deren partielle Ableitung nach S nicht notwendigerweise.

Analog zur Flussfunktion passen wir die Definitionen 3.1.2 und 3.1.4 der Anfangs- und Randwertfunktionen an das Optimalsteuerungsproblem (3.3.1) an.

Definition 3.3.2. Für eine Schar von Optimalsteuerungsproblemen der Form (3.3.1) beschreiben $\rho^0(S, T)$, $\rho^e(S, T)$, $\sigma^0(S, T)$ und $\sigma^e(S, T)$ die Anfangs- und Endwerte von x und λ bei Wahl der optimalen Steuerung in Abhängigkeit von S und T . $\hat{\mu}(S, T)$ beschreibt die Multiplikatoren in den Transversalitätsbedingungen.

Nun leiten wir ϕ_1 zum Zeitpunkt T total nach S ab und nutzen dabei die Gleichungen (3.1.7) und (3.1.5):

$$\begin{aligned} \frac{d}{dS} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) &= \frac{\partial}{\partial S} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) \\ &+ \frac{\partial}{\partial x_0} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) \frac{\partial}{\partial S} \rho^0(S, T) \\ &+ \frac{\partial}{\partial \lambda_0} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) \frac{\partial}{\partial S} \sigma^0(S, T) \\ &= \phi_{1,S}(S, T, \rho^0(S, T), \sigma^0(S, T)) \\ &+ V_1 \rho_S^0(S, T) + V_2 \sigma_S^0(S, T). \end{aligned}$$

Aus Gleichung (3.1.5) folgt aber auch

$$\frac{d}{dS} \phi_1(S, T, \rho^0(S, T), \sigma^0(S, T)) = \rho_S^e(S, T).$$

Wir gehen analog für ϕ_2 vor und erhalten zusammengefasst das lineare, implizite System

$$\begin{aligned} \rho_S^e(S, T) &= \phi_{1,S}(S, T, \rho^0(S, T), \sigma^0(S, T)) + V_1 \rho_S^0(S, T) + V_2 \sigma_S^0(S, T) \\ \sigma_S^e(S, T) &= \phi_{2,S}(S, T, \rho^0(S, T), \sigma^0(S, T)) + V_3 \rho_S^0(S, T) + V_4 \sigma_S^0(S, T). \end{aligned} \quad (3.3.3)$$

Zum Lösen des Systems benötigen wir ϕ_S zum Zeitpunkt T . Dazu leiten wir ϕ_S partiell nach t ab, nutzen die zweimal stetige Differenzierbarkeit von ϕ und verwenden Gleichung (3.3.2) aus Definition 3.3.1, d.h.

$$\begin{aligned} \dot{\phi}_S(S, t, \rho^0(S, T), \sigma^0(S, T)) &= \frac{\partial}{\partial t} \frac{\partial}{\partial S} \phi(S, t, \rho^0(S, T), \sigma^0(S, T)) \\ &= \frac{\partial}{\partial S} \frac{\partial}{\partial t} \phi(S, t, \rho^0(S, T), \sigma^0(S, T)) \\ &= \begin{pmatrix} F_S(x(t), \lambda(t), S) \\ G_S(x(t), \lambda(t), S) \end{pmatrix} \\ &+ D_{x,\lambda} \mathcal{X}(x(t), \lambda(t), S) \phi_S(S, t, \rho^0(S, T), \sigma^0(S, T)). \end{aligned} \quad (3.3.4)$$

Wir erhalten somit wieder ein Differentialgleichungssystem, welches simultan mit der Matrixdifferentialgleichung (3.1.8) sowie den Zustands- und Adjungiertengleichungen (3.1.9) gelöst wird. Die zugehörige Anfangsbedingung $\phi_S(S, 0, x_0, \lambda_0) = 0_{2n}$ folgt aus

Definition 3.3.1.

Damit ergeben sich insgesamt die folgenden linearen Systeme

$$\begin{aligned}
\begin{pmatrix} \rho_S^e \\ \sigma_S^e \end{pmatrix} &= \phi_S(S, T, \rho^0, \sigma^0) + V(S, T, \rho^0, \sigma^0) \begin{pmatrix} \rho_S^0 \\ \sigma_S^0 \end{pmatrix} \quad S \in [S_0, S_e] \\
0_{n_\psi} &= D_{x_0, x_e} \psi(S, \rho^0, \rho^e) \begin{pmatrix} \rho_S^0 \\ \rho_S^e \end{pmatrix} + \psi_S(S, \rho^0, \rho^e) \\
\begin{pmatrix} -\sigma_S^0 \\ \sigma_S^e \end{pmatrix} &= D_{x_0, x_e} \mathcal{K}(S, \rho^0, \rho^e, \hat{\mu}) \begin{pmatrix} \rho_S^0 \\ \rho_S^e \end{pmatrix} \\
&\quad + (D_{x_0, x_e} \psi(S, \rho^0, \rho^e))^\top \hat{\mu}_S \\
&\quad + \mathcal{K}_S(S, \rho^0, \rho^e, \hat{\mu}) \\
\dot{V}(S, t, \rho^0, \sigma^0) &= D_{x, \lambda} \mathcal{X}(x(t), \lambda(t), S) V(S, t, \rho^0, \sigma^0) \quad t \in [0, T] \\
V(S, 0, \rho^0, \sigma^0) &= I_{2n} \\
\dot{\phi}_S(S, t, \rho^0, \sigma^0) &= \begin{pmatrix} F_S(x(t), \lambda(t), S) \\ G_S(x(t), \lambda(t), S) \end{pmatrix} \\
&\quad + D_{x, \lambda} \mathcal{X}(x(t), \lambda(t), S) \phi_S(S, t, \rho^0, \sigma^0) \\
\phi_S(S, 0, \rho^0, \sigma^0) &= 0_{2n} \\
\begin{pmatrix} \dot{x}(t) \\ \dot{\lambda}(t) \end{pmatrix} &= \begin{pmatrix} F(x(t), \lambda(t), S) \\ G(x(t), \lambda(t), S) \end{pmatrix} \\
\begin{pmatrix} x(0) \\ \lambda(0) \end{pmatrix} &= \begin{pmatrix} \rho^0 \\ \sigma^0 \end{pmatrix},
\end{aligned} \tag{3.3.5}$$

wobei die letzten drei Differentialgleichungssysteme für jeden Integrationsschritt von (3.3.3) simultan gelöst werden.

3.4. Unterschiede zu Costanzas Ansatz

Die Grundlage für den neuen Ansatz ist in [5, 6] zu finden. Der Spezialfall, dass x ein-dimensionale Funktionswerte hat, wird in [5] behandelt und liefert im Gegensatz zum mehrdimensionalen Fall, der in [6] untersucht wird, korrekte Ergebnisse. Wir zeigen in diesem Abschnitt, wieso Costanzas Ansatz nur für spezielle Fälle angewendet werden kann.

Wir betrachten einen Spezialfall von unserem Optimalsteuerungsproblem (3.1.1):

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^T L(x(t), u(t)) dt + x^\top(T) S x(T) \quad S, T \in \mathbb{R} \\ \dot{x}(t) &= f(x(t), u(t)) \quad t \in [0, T] \\ x(0) &= x_0. \end{aligned} \quad (3.4.1)$$

Die Adjungiertengleichungen verändern sich nicht und wir haben die spezielle Transversalitätsbedingung

$$\lambda(T) = 2S x(T).$$

Damit ergibt sich für unsere Flussfunktion gemäß Definition 3.1.1 und ρ bzw. σ wie in 3.1.2 definiert:

$$\begin{aligned} \phi(S, 0, x_0, \sigma^0(S, T)) &= \begin{pmatrix} x_0 \\ \sigma^0(S, T) \end{pmatrix} \\ \phi(S, T, x_0, \sigma^0(S, T)) &= \begin{pmatrix} \rho^e(S, T) \\ 2S\rho^e(S, T) \end{pmatrix}. \end{aligned}$$

Für V gemäß (3.1.7) und (3.1.8) gelten die Gleichungen

$$\begin{aligned} \dot{V}(S, t, v) &= D_{x,\lambda} \mathcal{X}(\phi^\top(S, t, v)) V(S, t, v) \quad t \in [0, T] \\ \dot{V}(S, T, v) &= D_{x,\lambda} \mathcal{X}(\phi^\top(S, T, v)) V(S, T, v) = \mathcal{A}(\phi(S, T, v)) V(S, T, v) \\ &= \begin{pmatrix} \mathcal{A}_1(\phi(S, T, v)) & \mathcal{A}_2(\phi(S, T, v)) \\ \mathcal{A}_3(\phi(S, T, v)) & \mathcal{A}_4(\phi(S, T, v)) \end{pmatrix} \begin{pmatrix} V_1(S, t, v) & V_2(S, t, v) \\ V_3(S, t, v) & V_4(S, t, v) \end{pmatrix} \\ &=: V_T(S, T, v) \end{aligned} \quad (3.4.2)$$

Im Folgenden lassen wir die Argumente bei den Matrixfunktionen weg. V ist symplektisch (siehe [14, S. 219-228]), so dass gilt

$$V^{-1} = \begin{pmatrix} V_4^\top & -V_2^\top \\ -V_3^\top & V_1^\top \end{pmatrix}.$$

In unserem Ansatz nutzen wir dies nicht aus, sondern berechnen V mittels (3.1.8). In [6] werden die V_i über die folgenden Hilfsmatrizen berechnet:

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} := V^{-1} \begin{pmatrix} I_n \\ 2S I_n \end{pmatrix} = \begin{pmatrix} V_4^\top - 2S V_2^\top \\ -V_3^\top + 2S V_1^\top \end{pmatrix}. \quad (3.4.3)$$

Wir leiten Gleichung (3.4.3) partiell nach T ab und nutzen dabei Gleichung (3.4.2) aus, so dass

$$\begin{aligned} \begin{pmatrix} \alpha_T \\ \beta_T \end{pmatrix} &= \begin{pmatrix} (V_4^T)_T - 2S(V_2^T)_T \\ -(V_3^T)_T + 2S(V_1^T)_T \end{pmatrix} \\ &= \begin{pmatrix} V_2^T \mathcal{A}_3^T + V_4^T \mathcal{A}_4^T - 2S(V_2^T \mathcal{A}_1^T + V_4^T \mathcal{A}_2^T) \\ -(V_1^T \mathcal{A}_3^T + V_3^T \mathcal{A}_4^T) + 2S(V_1^T \mathcal{A}_1^T + V_3^T \mathcal{A}_2^T) \end{pmatrix}. \end{aligned} \quad (3.4.4)$$

Dies tun wir nochmal für S , d.h.

$$\begin{pmatrix} \alpha_S \\ \beta_S \end{pmatrix} = \begin{pmatrix} V_{4,S}^T - 2(V_2^T + S V_{2,S}^T) \\ -V_{3,S}^T + 2(V_1^T + S V_{1,S}^T) \end{pmatrix}. \quad (3.4.5)$$

In [5] tauchen die partiellen Ableitungen der V_i nach S aber nicht in Gleichung (3.4.5) auf. Dadurch können zwar die V_i zusammen mit Gleichung (3.4.3) explizit nach α , α_S , β und β_S umgestellt werden, jedoch liefert dies falsche Werte schon für einfache, nichtlineare Probleme. Dementsprechend ist der alte Ansatz nur korrekt, wenn die V_i nicht von S abhängen wie z.B. bei linearen Problemen.

Nehmen wir die partiellen Ableitungen hinzu, so können wir die V_i nicht mehr explizit bestimmen. Aus diesem Grund berechnen wir V direkt aus der Matrix-ODE (3.1.8). Im alten Ansatz muss man eine Matrix-PDE lösen, allerdings für α und β , welche sich aus Gleichung (3.4.4) und den umgestellten V_i ergibt.

Ein weiterer Vorteil unseres Ansatzes ist die Verallgemeinerung des Optimalsteuerungsproblems und der Definition von ρ und σ . Dadurch können auch Probleme gelöst werden, bei denen nur Endwerte oder sowohl Anfangs- als auch Endwerte gegeben sind. Weiterhin können wir wie in Abschnitt 3.3 beschrieben auch Optimalsteuerungsprobleme untersuchen, bei denen das S explizit im Integranden oder den Zustandsgleichungen vorkommt. Dies kann z.B. genutzt werden, um Beschränkungen an die Steuerung oder die Zustände zu approximieren. Dies werden wir in Kapitel 5 näher beleuchten.

Das folgende Kapitel erklärt, wie wir den beschriebenen Ansatz numerisch umsetzen.

4. Numerische Methoden

4.1. Lösen der ODE

Bei den linearen Systemen (3.1.13), (3.2.4) und (3.3.5) handelt es sich im Wesentlichen um Differentialgleichungssysteme mit gegebenen Anfangswerten. Dementsprechend können wir auf bekannte Anfangswertlöser zurückgreifen, um die jeweiligen Probleme zu lösen. Wir haben uns dabei entschieden, die von MATLAB zur Verfügung gestellten Routinen zu benutzen.

Insbesondere haben wir *ode45* verwendet, welches ein Löser mit Schrittweitensteuerung ist und auf expliziten Runge-Kutta-Verfahren beruht. Im Speziellen werden dabei die Koeffizienten von RK5(4)7M aus [7] benutzt. Dabei werden zur Schrittweitensteuerung zwei Runge-Kutta-Verfahren unterschiedlicher Ordnung (4 bzw. 5) verwendet, die in vielen Koeffizienten übereinstimmen, so dass die Anzahl der benötigten Funktionsaufrufe klein gehalten wird. Weitere Details zu der Schrittweitensteuerung können z.B. in [26, S. 426-429] nachgeschlagen werden.

In Abschnitt 5.2 werden wir auch andere explizite Verfahren untersuchen, wie z.B. *ode113*, welches im Gegensatz zu *ode45* ein Mehrschrittverfahren ist. Implizite Verfahren haben wir nicht betrachtet, da die hier genannten expliziten Verfahren bereits gute Ergebnisse liefern, wie wir in Kapitel 5 sehen werden.

Im Folgenden bezeichnen wir das Lösen der linearen, impliziten Systeme (3.1.13), (3.2.4) und (3.3.5) als *äußere Integration*. Die Behandlung der restlichen Differentialgleichungssysteme (3.1.8), (3.1.9) und (3.3.4) nennen wir *innere Integration*. Wir gehen dabei in den nächsten zwei Abschnitten auf die Besonderheiten bei den numerischen Berechnungen der jeweiligen Systeme ein.

4.1.1. Äußere Integration

Die impliziten Gleichungen der äußeren Integration aus dem System (3.1.13) sind linear in den Ableitungen und können kompakt aufgeschrieben werden durch

$$\begin{aligned} \begin{pmatrix} V_1 & -I_n & V_2 & 0 & 0 \\ V_3 & 0 & V_4 & -I_n & 0 \\ D_{x_0, x_e} \psi & 0 & 0 & 0 & 0 \\ D_{x_0, x_e} \mathcal{K} & I_n & 0 & 0 & (D_{x_0, x_e} \psi)^\top \\ 0 & 0 & -I_n & 0 & 0 \end{pmatrix} \begin{pmatrix} \rho_T^0 \\ \rho_T^e \\ \sigma_T^0 \\ \sigma_T^e \\ \hat{\mu}_T \end{pmatrix} &= \begin{pmatrix} -F(\rho^e, \sigma^e) \\ -G(\rho^e, \sigma^e) \\ 0 \\ 0 \\ 0 \end{pmatrix} \\ &= \mathcal{M}(S, T, \rho^0, \rho^e, \sigma^0) \begin{pmatrix} \rho_T^0 \\ \rho_T^e \\ \sigma_T^0 \\ \sigma_T^e \\ \hat{\mu}_T \end{pmatrix} = \begin{pmatrix} -F(\rho^e, \sigma^e) \\ -G(\rho^e, \sigma^e) \\ 0 \\ 0 \\ 0 \end{pmatrix}. \end{aligned} \quad (4.1.1)$$

Die Matrixfunktion \mathcal{M} ist bei allen linearen Systemen, d.h. (3.1.13), (3.2.4) und (3.3.5), identisch. Ist diese auch invertierbar, so haben wir ein System von gewöhnlichen Differentialgleichungen der Form

$$\begin{pmatrix} \rho_T^0 \\ \rho_T^e \\ \sigma_T^0 \\ \sigma_T^e \\ \hat{\mu}_T \end{pmatrix} = \mathcal{M}^{-1}(S, T, \rho^0, \rho^e, \sigma^0) \begin{pmatrix} -F(\rho^e, \sigma^e) \\ -G(\rho^e, \sigma^e) \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

welches wir mit den vorher diskutierten Methoden lösen können. Dabei berechnen wir aber nicht explizit die Inverse von \mathcal{M} , sondern wir bestimmen die partiellen Ableitungen über das Lösen des linearen Gleichungssystems (4.1.1), welches wir mit dem Standardlöser von MATLAB realisieren.

In vielen praktischen Beispielen, wie z.B. dem Apollo-Problem, ist es möglich das lineare, implizite System zu reduzieren. Dies stellen wir an dem folgenden Optimalsteuerungsproblem dar, welches eine modifizierte Version des Apollo-Problems aus Abschnitt 5.2 ist:

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^T L(x(t), u(t)) dt + (x(T) - x_e)^\top S (x(T) - x_e) & S, T \in \mathbb{R} \\ \dot{x}(t) &= f(x, u) & t \in [0, T] \\ x(0) &= x_0. & \end{aligned} \quad (4.1.2)$$

Die dazugehörigen Transversalitätsbedingungen lauten

$$\begin{pmatrix} -\lambda(0) \\ \lambda(T) \end{pmatrix} = \begin{pmatrix} \mu \\ 2S(x(T) - x_e) \end{pmatrix} = \mathcal{K}(S, x(0), x(T), \mu). \quad (4.1.3)$$

Für die Ableitung der Randbedingungen nach den Anfangs- und Endwerten ergibt sich

$$D_{x_0, x_e} \psi(S, x(0), x(T)) = \begin{pmatrix} I_n & 0 \end{pmatrix}. \quad (4.1.4)$$

Damit erhalten wir das folgende lineare, implizite System in T :

$$\begin{pmatrix} V_1 & -I_n & V_2 & 0 & 0 \\ V_3 & 0 & V_4 & -I_n & 0 \\ I_n & 0 & 0 & 0 & 0 \\ 0 & 0 & I_n & 0 & I_n \\ 0 & 2S & 0 & -I_n & 0 \end{pmatrix} \begin{pmatrix} \rho_T^0 \\ \rho_T^e \\ \sigma_T^0 \\ \sigma_T^e \\ \hat{\mu}_T \end{pmatrix} = \begin{pmatrix} -F(\rho^e, \sigma^e) \\ -G(\rho^e, \sigma^e) \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (4.1.5)$$

Die letzten $3n$ Gleichungen können umgeschrieben werden zu

$$\begin{aligned} \rho_T^0(S, T) &= 0 \\ \hat{\mu}_T(S, T) &= -\sigma_T^0(S, T) \\ \sigma_T^e(S, T) &= 2S\rho_T^e(S, T). \end{aligned} \quad (4.1.6)$$

Das heißt, dass wir nur die ersten $2n$ linearen, impliziten Gleichungen für ρ^e und σ^0 benötigen, um alle $5n$ gesuchten Funktionen zu bestimmen. Dies können wir auch direkt aus den Gleichungen (4.1.2) und (4.1.3) ablesen, denn sie implizieren

$$\begin{aligned} \rho^0(S, T) &= x_0 \\ \hat{\mu}(S, T) &= -\sigma^0(S, T) \\ \sigma^e(S, T) &= 2S(\rho^e(S, T) - x_e). \end{aligned}$$

Diese Gleichungen setzen wir in die V_i , F und G ein, nutzen die Äquivalenzen (4.1.6) aus und reduzieren damit das lineare, implizite System auf (4.1.5) auf

$$\begin{pmatrix} -I_n & V_2 \\ -2SI_n & V_4 \end{pmatrix} \begin{pmatrix} \rho_T^e \\ \sigma_T^0 \end{pmatrix} = \begin{pmatrix} -F(\rho^e, 2S(\rho^e - x_e)) \\ -G(\rho^e, 2S(\rho^e - x_e)) \end{pmatrix}. \quad (4.1.7)$$

In den Beispielen in Kapitel 5 werden wir, sofern nicht anders explizit erwähnt, nur das reduzierte System benutzen.

Wir betrachten wieder das System (4.1.1), bei dem über $T \in [T_0, T_e]$ integriert wird. In der Praxis möchten wir auch oft die Anfangs- und Endwerte der Zustände und Adjungierten nicht nur für verschiedene T , sondern auch für verschiedene $S \in [S_0, S_e]$ bestimmen. Dazu wählen wir uns eine endliche Menge von $S_i \in [S_0, S_e]$ mit $i = 0, \dots, N$. Dies ergibt die folgenden $N + 1$ Systeme

$$\begin{aligned}
\begin{pmatrix} \rho_T^e \\ \sigma_T^e \end{pmatrix} &= \begin{pmatrix} F(\rho^e, \sigma^e) \\ G(\rho^e, \sigma^e) \end{pmatrix} + V(S_i, T, \rho^0, \sigma^0) \begin{pmatrix} \rho_T^0 \\ \sigma_T^0 \end{pmatrix} & T \in [T_0, T_e] \\
0 &= D_{x_0, x_e} \psi(S_i, \rho^0, \rho^e) \begin{pmatrix} \rho_T^0 \\ \rho_T^e \end{pmatrix} & i = 0, \dots, N \\
\begin{pmatrix} -\sigma_T^0 \\ \sigma_T^e \end{pmatrix} &= D_{x_0, x_e} \mathcal{K}(S_i, \rho^0, \rho^e, \hat{\mu}) \begin{pmatrix} \rho_T^0 \\ \rho_T^e \end{pmatrix} \\
&\quad + (D_{x_0, x_e} \psi(S_i, \rho^0, \rho^e))^\top \hat{\mu}.
\end{aligned}$$

Diese Systeme sind unabhängig voneinander und können somit parallel berechnet werden. Dies ist insofern sinnvoll, da die jeweils zugehörige innere Integration, die in jedem Zeitschritt der äußeren Integration ermittelt werden muss, auch nur von dem entsprechenden S_i abhängt. Somit kann z.B. bei einem Mehrkernprozessor jeder Kern ein S_i übernehmen und die zugehörige äußere als auch innere Integration durchführen.

In dem nächsten Abschnitt werden wir die innere Integration näher untersuchen und ferner auch angeben, wie wir die partiellen Ableitungen von ψ und \mathcal{K} bestimmen bzw. approximieren.

4.1.2. Innere Integration

Die innere Integration, bei der über $t \in [0, T]$ integriert wird, beinhaltet die Adjungierten-
gleichungen (3.1.9)

$$\begin{aligned}
\begin{pmatrix} \dot{x}(t) \\ \dot{\lambda}(t) \end{pmatrix} &= \begin{pmatrix} F(x(t), \lambda(t)) \\ G(x(t), \lambda(t)) \end{pmatrix} \\
\begin{pmatrix} x(0) \\ \lambda(0) \end{pmatrix} &= \begin{pmatrix} \rho^0 \\ \sigma^0 \end{pmatrix},
\end{aligned}$$

das Matrixdifferentialgleichungssystem (3.1.8)

$$\begin{aligned}
\dot{V}(S, t, \rho^0, \sigma^0) &= D_{x, \lambda} \mathcal{X}(x(t), \lambda(t), S) V(S, t, \rho^0, \sigma^0) \\
V(S, 0, \rho^0, \sigma^0) &= I_{2n}
\end{aligned}$$

und bei Optimalsteuerungsproblemen der Form (3.3.1) aus Abschnitt 3.3 das Differentialgleichungssystem (3.3.4)

$$\begin{aligned}
\dot{\phi}_S(S, t, \rho^0, \sigma^0) &= \begin{pmatrix} F_S(x(t), \lambda(t), S) \\ G_S(x(t), \lambda(t), S) \end{pmatrix} \\
&\quad + D_{x, \lambda} \mathcal{X}(x(t), \lambda(t), S) \phi_S(S, t, \rho^0, \sigma^0) \\
\phi_S(S, 0, \rho^0, \sigma^0) &= 0.
\end{aligned}$$

Dies sind alles Anfangswertprobleme und sie können somit mit den Methoden aus dem Abschnitt 4.1 simultan gelöst werden. Dabei müssen wir aber noch die partiellen Ableitungen von \mathcal{X} , F und G bestimmen.

Wir haben uns entschieden, diese Ableitungen numerisch zu bestimmen, damit der Anwender so wenig Vorarbeit wie möglich leisten muss, um sein Optimalsteuerungsproblem zu lösen.

Eine Möglichkeit besteht im Automatischen Differenzieren [10], welches sehr genaue Werte liefert, da nur Rundungsfehler auftreten können. Dabei werden die Ableitungen über das Ausnutzen der Kettenregel bestimmt. Dazu müssen allerdings alle Operationen wie Addition, Multiplikation, ... und Grundfunktionen wie z.B. die trigonometrischen Funktionen in der jeweiligen Programmiersprache überladen werden, d.h. sie müssen sowohl den Funktionswert als auch den Wert der Ableitung für ein konkretes Argument zurückgeben. Dies kann abhängig von der Programmiersprache aufwändig zu implementieren sein.

Eine einfachere Möglichkeit besteht in dem Verwenden von Differenzenquotienten, wie z.B. dem zentralen Differenzenquotienten

$$\frac{f(x+h) - f(x-h)}{2h} = f'(x) + O(h^2). \quad f \in C^3((x-h, x+h)) \quad (4.1.8)$$

Dieser liefert mit nur zwei Funktionsauswertungen eine Approximation an die erste Ableitung mit Konsistenzordnung 2 unter der Voraussetzung, dass die Funktion mindestens dreimal stetig differenzierbar ist. Für kleine Schrittweiten h ist der zentrale Differenzenquotient zwar sehr genau, gleichzeitig hat er aber auch für kleine h den Nachteil, dass im Zähler eine Differenz von zwei fast gleichgroßen Zahlen steht und es somit zu einer numerischen Auslöschung [26, S.8-14] kommen kann. Dies kann umgangen werden, wenn wir im Komplexen rechnen.

Die Idee, Ableitungen von reellen Funktionen durch Auswertung der Funktionen im Komplexen zu approximieren, wurde bereits 1967 von Lyness und Moler [20] entwickelt, welche auf beliebig hohe Ableitungen angewendet werden kann. Die Approximationsformel, die wir im Folgenden benutzen, stammt von Squire und Trapp [25]. Dazu sei die Funktion f analytisch. Dann können wir für reelle x und h die folgende Taylorentwicklung durchführen:

$$\begin{aligned} f(x+ih) &= f(x) + \lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{(ih)^k}{k!} f^{(k)}(x) \\ &= f(x) + ihf'(x) - \frac{h^2}{2}f''(x) - i\frac{h^3}{6}f'''(x) + \dots \end{aligned}$$

Wir brechen nun die Reihe nach dem dritten Glied ab, nehmen von beiden Seiten den Imaginärteil, wobei wir dabei ausnutzen, dass f analytisch ist, und erhalten durch Umstellen nach $f'(x)$

$$f'(x) = \frac{\Im f(x+ih)}{h} + O(h^2). \quad (4.1.9)$$

Dies bezeichnet man auch als *Complex step differentiation*. Dabei haben wir wie beim zentralen Differenzenquotient Konsistenzordnung 2 vorliegen, aber ohne den Nachteil der numerischen Auslöschung, da es nur zu einer komplexen Funktionsauswertung kommt. Dadurch können wir h sehr klein wählen, z.B. $h = 10^{-8}$, ohne negative Auswirkungen auf die Genauigkeit der Approximation. Ferner ist die eine komplexe Funktionsauswertung schneller zu berechnen als die zwei reellen Funktionsauswertungen des zentralen Differenzenquotienten, wie wir in Abschnitt 5.2 anhand des Apollo-Beispiels sehen werden. Wir haben uns deshalb entschieden, alle Ableitungen, sofern sie numerisch bestimmt werden sollen, mit Complex step differentiation zu approximieren.

Dies ist insbesondere wichtig für die numerische Berechnung von $D_{x,\lambda}\mathcal{X}$, welches wir im Folgenden mit \mathcal{A} abkürzen. \mathcal{A} taucht dabei sowohl in Gleichung (3.1.8) als auch (3.3.4) auf. Die Ableitungen müssen aber nur einmal pro Integrationsschritt approximiert werden, da die Anfangswertprobleme simultan gelöst werden. Ferner werden wir in Abschnitt 5.2 sehen, dass die Berechnung von \mathcal{A} den größten Teil der Rechenzeit ausmacht und es somit sehr wichtig ist, wie man die partiellen Ableitungen bestimmt.

Eine weitere Möglichkeit dies zu beschleunigen ist es, die partiellen Ableitungen nach den einzelnen Komponenten parallel zu ermitteln, da sie unabhängig voneinander sind. Allerdings brachte dies für die Beispiele in Kapitel 5 keinen Zeitgewinn, weil die Zustände in den Problemen von kleiner Dimension sind und somit der Overhead meist größer war als die Berechnung der einzelnen Approximationen. Für Probleme von großer Dimension kann die Parallelisierung aber Vorteile bringen.

In den nächsten Abschnitten beschäftigen wir uns mit anderen Verfahren, um Optimalsteuerungsprobleme zu lösen. Die vorgestellten Methoden werden wir benutzen, um unseren Ansatz zu überprüfen.

4.2. Indirekte Verfahren

Unser Ansatz beruht auf den indirekten Verfahren, bei denen man die notwendigen Bedingungen aus Satz 2.4.7 für eine optimale Steuerung nutzt, um ein Randwertproblem aufzustellen, mit dem man die Zustände und Adjungierten bestimmen kann. Wir nutzen dieses Verfahren zur Überprüfung der numerischen Ergebnisse unseres Ansatzes. Dazu betrachten wir ein Randwertproblem der Form

$$\begin{aligned} \dot{x}(t) &= f(x(t)) \in \mathbb{R}^n \quad t \in [a, b] \\ r(x(a), x(b)) &= 0_n. \end{aligned} \quad (4.2.1)$$

Dieses kann man z.B. in MATLAB mit der Routine *bvp4c* lösen, welche auf dem Kollokationsverfahren [15] beruht. Wir haben uns aber für einen eigenen Löser entschieden, der im Wesentlichen auf dem Mehrfachschießverfahren basiert. Bei der Beschreibung des Verfahrens halten wir uns im Wesentlichen an die Bezeichnungen aus [26, S. 483-487], wobei bei uns die Differentialgleichung o.B.d.A. autonom sei (siehe Abschnitt 2.2) und unsere Funktion x von dem Parameter t abhängt.

Wir teilen das Intervall $[a, b]$ in $m - 1$ Teilintervalle der Form $[t_i, t_{i+1}]$ auf, wobei die $t_i \in [a, b]$ äquidistante Zeitpunkte seien mit

$$a = t_1 < t_2 < \dots < t_m = b.$$

Definition 4.2.1. Sei $x : [a, b] \rightarrow \mathbb{R}^n$ Lösung des Randwertproblems (4.2.1). Dann heißen die Variablen $s_k \in \mathbb{R}^n$ mit der Eigenschaft

$$x(t_k) = s_k \quad k = 1, \dots, m$$

Startwerte des Mehrfachschießverfahrens.

Wir möchten diese Startwerte numerisch ermitteln. Die zentrale Idee des Verfahrens ist es, auf jedem Teilintervall ein Anfangswertproblem zu lösen. Dabei sind die Startwerte des Mehrfachschießverfahrens gleichzeitig die Startwerte der Anfangswertprobleme.

Definition 4.2.2. Die Funktion $\tilde{x} : \mathbb{R} \times \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ löst das Anfangswertproblem

$$\begin{aligned} \frac{\partial}{\partial t} \tilde{x}(t, t_0, x_0) &= f(\tilde{x}) & t \geq t_0 \\ \tilde{x}(t_0, t_0, x_0) &= x_0. \end{aligned}$$

Ziel ist es nun die Startwerte so zu bestimmen, dass die zusammengesetzte Funktion $\bar{x} : \mathbb{R} \rightarrow \mathbb{R}^n$ mit

$$\begin{aligned} \bar{x}(t) &:= \tilde{x}(t, t_k, s_k) & t \in [t_k, t_{k+1}) & k = 1, \dots, m-1 \\ \bar{x}(t_m) &:= s_m \end{aligned}$$

stetig und damit insgesamt eine Lösung unseres Randwertproblems (4.2.1) ist. Damit haben wir letztendlich $n \cdot m$ Bedingungen an unsere Startwerte, d.h.

$$\begin{aligned}\hat{x}(t_{k+1}, t_k, s_k) &= s_{k+1} \quad k = 1, \dots, m-1 \\ r(s_1, s_m) &= 0.\end{aligned}$$

Diese können wir als Nullstellenproblem für eine Funktion $F : \mathbb{R}^{nm} \rightarrow \mathbb{R}^{nm}$ wie folgt aufschreiben:

$$F(s) := \begin{pmatrix} F_1(s_1, s_2) \\ \vdots \\ F_{m-1}(s_{m-1}, s_m) \\ F_m(s_1, s_m) \end{pmatrix} = \begin{pmatrix} \hat{x}(t_2, t_1, s_1) - s_2 \\ \vdots \\ \hat{x}(t_m, t_{m-1}, s_{m-1}) - s_m \\ r(s_1, s_m) \end{pmatrix} = 0, \quad (4.2.2)$$

wobei $s = (s_1^\top, \dots, s_m^\top)$. Zum Lösen dieses Problems benutzen wir das Newtonverfahren

$$\begin{aligned}DF(s^{(i)}) \Delta s^{(i)} &= -F(s^{(i)}) \\ s^{(i+1)} &= s^{(i)} + \Delta s^{(i)}. \quad i = 0, 1, \dots\end{aligned} \quad (4.2.3)$$

Dies erfordert die Berechnung der Jacobi-Matrix von F , welche für große m dünnbesetzt ist und auf Grund von (4.2.2) die folgende Struktur hat

$$DF(s) = \begin{pmatrix} G_1(s) & -I_n & 0 & \cdots & 0 \\ 0 & G_2(s) & -I_n & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & G_{m-1}(s) & -I_n \\ A(s) & 0 & \cdots & 0 & B(s) \end{pmatrix}$$

mit den Jacobi-Matrizen

$$\begin{aligned}G_k(s) &:= D_{s_k} F_k(s) = D_{s_k} \hat{x}(t_{k+1}, t_k, s_k), \quad k = 1, \dots, m-1 \\ A(s) &:= D_{s_1} F_m(s) = D_{s_1} r(s_1, s_m), \\ B(s) &:= D_{s_m} F_m(s) = D_{s_m} r(s_1, s_m).\end{aligned}$$

Zur Berechnung der Jacobi-Matrizen benutzen wir die in Abschnitt 4.1.2 beschriebene Complex step differentiation (4.1.9), welche schneller ist als eine numerische Approximation über zentrale Differenzenquotienten, wie wir in Abschnitt 5.2 sehen werden. Weiterhin nutzen wir aus, dass die G_i unabhängig voneinander sind und somit parallel berechnet werden können. Dies beschleunigt die Berechnung deutlich, da zur Berechnung einer Jacobi-Matrix G_i n Anfangswertprobleme gelöst werden müssen. Zum Lösen dieser Anfangswertprobleme haben wir das klassische Runge-Kutta-Verfahren 4. Ordnung benutzt, wobei die Zeitschritte innerhalb eines Anfangswertproblems äquidistant sind.

Die Konvergenz des Newton-Verfahrens (4.2.3) ist allerdings nur gesichert [26, S. 253-256], wenn der Startwert $s^{(0)}$ hinreichend nah an der Nullstelle von F ist. Um das Verfahren robuster zu machen, betrachten wir eine modifizierte Version des Newton-Verfahrens der Form

$$\begin{aligned} DF(s^{(i)}) \Delta s^{(i)} &= -F(s^{(i)}) \\ s^{(i+1)} &= s^{(i)} + \alpha_i \Delta s^{(i)}. \quad i = 0, 1, \dots \end{aligned} \quad (4.2.4)$$

Die Idee dahinter ist, dass wir mit den $\alpha_i \in [0, 1]$ den Newton-Schritt aus Algorithmus (4.2.3) dämpfen können, falls wir uns mit einer Iteration von der Nullstelle entfernen würden. Dabei werden die α_i wie folgt bestimmt:

$$\begin{aligned} g(\cdot) &:= F^T(\cdot)F(\cdot) \\ l_{\min} &= \min_{l \in \mathbb{N}_0} \{l \leq l_{\max} \mid g(s^{(i)} + 2^{-l} \Delta s^{(i)}) < g(s^{(i)})\} & l_{\max} &\in \mathbb{N}_0 \\ \alpha_i &= 2^{-l_{\min}}. & \alpha_{\max} &\in [0, 1] \end{aligned}$$

Dabei kann l_{\max} von uns frei gewählt werden. Ist die Menge zur Bestimmung von l_{\min} leer, so setzen wir $l_{\min} = l_{\max}$. Bei den meisten Beispielen in Kapitel 5 haben wir uns für $l_{\max} = 3$ entschieden.

4.3. Direkte Verfahren

Wir werden in dem nächsten Kapitel auch Beispiele betrachten, bei denen die Steuerungen und/oder die Zustände beschränkt sind. Diese können in der folgenden Form aufgeschrieben werden

$$\begin{aligned} \min_u J(u) &= \min_u \int_0^T L(x(t), u(t)) dt + \varphi(x(0), x(T)) \quad T \geq 0 \\ \dot{x}(t) &= f(x(t), u(t)) \quad t \in [0, T] \\ \psi(x(0), x(T)) &= 0_{n_\psi} \\ c(x(t), u(t)) &\leq 0_{n_c} \quad t \in [0, T] \end{aligned} \quad (4.3.1)$$

Die Idee der *direkten Verfahren* ist es, das obige Problem mit einem diskreten, endlichen Optimierungsproblem der folgenden Form zu approximieren:

$$\begin{aligned} \min_y D(y) \quad y &\in \mathbb{R}^{N_g} \\ c_e(y) &= 0_{n_e} \\ c_i(y) &\leq 0_{n_i} \end{aligned} \quad (4.3.2)$$

Um (4.3.1) mit Hilfe von (4.3.2) zu approximieren diskretisieren wir zuerst die Zeit, in dem wir nur mit endlich vielen äquidistanten Zeitpunkten t_i rechnen. Für sie gilt

$$\begin{aligned} t_0 &= 0 \\ t_N &= T \\ t_{i+1} - t_i &= \tau \quad i = 0, \dots, N-1 \end{aligned}$$

Dementsprechend rechnen wir auch nur mit den Zuständen und Steuerungen zu diesen Zeitpunkten, d.h.

$$\begin{aligned} x_i &:= x(t_i) \quad i = 0, \dots, N \\ u_i &:= u(t_i). \end{aligned}$$

Das Integral im Zielfunktional approximieren wir mit der Trapezformel, so dass

$$\begin{aligned} \int_0^T L(x(t), u(t)) dt + \varphi(x(0), x(T)) &\approx \varphi(x_0, x_N) + \tau \sum_{i=0}^{N-1} \alpha_i L(x_i, u_i) \\ \alpha_i &:= \begin{cases} 1 & i = 1, \dots, N-1 \\ 0.5 & \text{sonst} \end{cases} \end{aligned}$$

Die Differentialgleichungen approximieren wir mit einer Differenzgleichung. Dabei beruht die Verfahrensfunktion Φ auf dem klassischen Runge-Kutta-Verfahren vierter Ordnung:

$$\begin{aligned}
k_1 &= f(x_i, u_i) \\
k_2 &= f\left(x_i + \frac{\tau}{2}k_1, \frac{u_i + u_{i+1}}{2}\right) \\
k_3 &= f\left(x_i + \frac{\tau}{2}k_2, \frac{u_i + u_{i+1}}{2}\right) \\
k_4 &= f(x_i + \tau k_3, u_{i+1}) \\
x_{i+1} &= x_i + \frac{\tau}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\
&= x_i + \tau\Phi(x_i, u_i, u_{i+1}, \tau; f).
\end{aligned}$$

Wir nehmen hier die Steuerung als stückweise linear an, so dass wir für die Berechnung von k_2 und k_3 das arithmetische Mittel der beiden Funktionswerte der Steuerung an den entsprechenden Zeitpunkten nehmen können, um den Funktionswert an der Zwischenstelle $t_i + \tau/2$ zu berechnen.

Insgesamt haben wir dann das diskrete Optimierungsproblem

$$\begin{aligned}
\min_{y=(x_0, \dots, x_N, u_0, \dots, u_N)} \quad & \varphi(x_0, x_N) + \tau \sum_{i=0}^N \alpha_i L(x_i, u_i) \quad y \in \mathbb{R}^{2N+2} \\
x_i - x_{i+1} + \tau\Phi(x_i, u_i, u_{i+1}, \tau; f) &= 0 \quad i = 0, \dots, N-1 \\
\psi(x_0, x_N) &= 0_{n_\psi} \\
c(x_i, u_i) &\leq 0_{n_c} \quad i = 0, \dots, N
\end{aligned} \tag{4.3.3}$$

Dieses Problem lösen wir mit der MATLAB-Routine *fmincon*, welches Aufgaben der Form (4.3.2) numerisch lösen kann. Als Lösungsverfahren kann dort gewählt werden zwischen Trust-Region-Verfahren [4], Methode der aktiven Mengen [9, S. 205-259], Innere-Punkte-Methode [2] und SQP-Verfahren [22, S. 529-562]. Wir haben für unsere Beispiele in Kapitel 5 die Methode der aktiven Mengen gewählt, da diese mit den wenigsten Funktionsauswertungen zum Minimum konvergiert hat.

5. Beispiele

5.1. Unbeschränkte Probleme

5.1.1. Manövrierung eines starren Raumfahrzeuges

Das folgende Beispiel stammt aus [19, S. 41-53] und beschreibt die optimale Manövrierung eines starren Raumfahrzeuges, wobei die Differentialgleichungen auch genutzt werden können, um die Bewegung eines Unterwasserfahrzeuges [3] zu modellieren.

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^{100} \frac{1}{2} u^\top(t) u(t) dt \\ \dot{x}(t) &= - \begin{pmatrix} \frac{I_3 - I_2}{I_1} x_2(t) x_3(t) \\ \frac{I_1 - I_3}{I_2} x_1(t) x_3(t) \\ \frac{I_2 - I_1}{I_3} x_1(t) x_2(t) \end{pmatrix} + \begin{pmatrix} \frac{u_1(t)}{I_1} \\ \frac{u_2(t)}{I_2} \\ \frac{u_3(t)}{I_3} \end{pmatrix} = f(x(t), u(t)) \\ x(0) &= \begin{pmatrix} 0.01 \\ 0.005 \\ 0.001 \end{pmatrix} = x_0 \\ x(100) &= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = x_e \end{aligned} \tag{5.1.1}$$

mit

$$\begin{aligned} I_1 &= 86.24, \\ I_2 &= 85.07, \\ I_3 &= 113.59. \end{aligned}$$

Dabei stehen die x_i für die Winkelgeschwindigkeiten des Weltraumfahrzeuges und die u_i sind die Steuerungen der Drehmomente. Wir modifizieren das obige Problem wie folgt, so dass wir unseren Ansatz anwenden können:

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^{100} \frac{1}{2} u^\top(t) u(t) dt \\ \dot{x}(t) &= f(x(t), u(t)) \\ x(0) &= x_0 \\ x(100) &= x_0 + S(x_e - x_0). \quad S \in [0, 1] \end{aligned} \tag{5.1.2}$$

Offensichtlich sind die beiden Probleme für $S = 1$ identisch. Diese Modifikation kann sinnvoll sein, wenn die Anfangswerte der Adjungierten für ein $S < 1$ leichter zu bestimmen sind. Wir berechnen nun die optimale Steuerung und stellen die notwendigen Bedingungen auf:

$$\hat{u}(x, \lambda) = - \begin{pmatrix} \frac{\lambda_1}{I_1} \\ \frac{\lambda_2}{I_2} \\ \frac{\lambda_3}{I_3} \end{pmatrix}$$

$$\dot{x} = f(x, \hat{u}(x, \lambda)) \quad x(0) = x_0, \quad x(100) = x_0 + S(x_e - x_0)$$

$$\dot{\lambda} = \begin{pmatrix} \lambda_2 \frac{I_1 - I_3}{I_2} x_3 + \lambda_3 \frac{I_2 - I_1}{I_3} x_2 \\ \lambda_1 \frac{I_3 - I_2}{I_1} x_3 + \lambda_3 \frac{I_2 - I_1}{I_3} x_1 \\ \lambda_1 \frac{I_3 - I_2}{I_1} x_2 + \lambda_2 \frac{I_1 - I_3}{I_2} x_1 \end{pmatrix}.$$

Für unsere Flussfunktion gilt

$$\phi(S, 0, \rho^0(S, T), \sigma^0(S, T)) = \begin{pmatrix} x_0 \\ \sigma^0(S, T) \end{pmatrix}$$

$$\phi(S, T, \rho^0(S, T), \sigma^0(S, T)) = \begin{pmatrix} x_0 + S(x_e - x_0) \\ \sigma^e(S, T) \end{pmatrix}.$$

Damit erhalten wir das lineare, implizite System

$$x_e - x_0 = V_2 \sigma_S^0$$

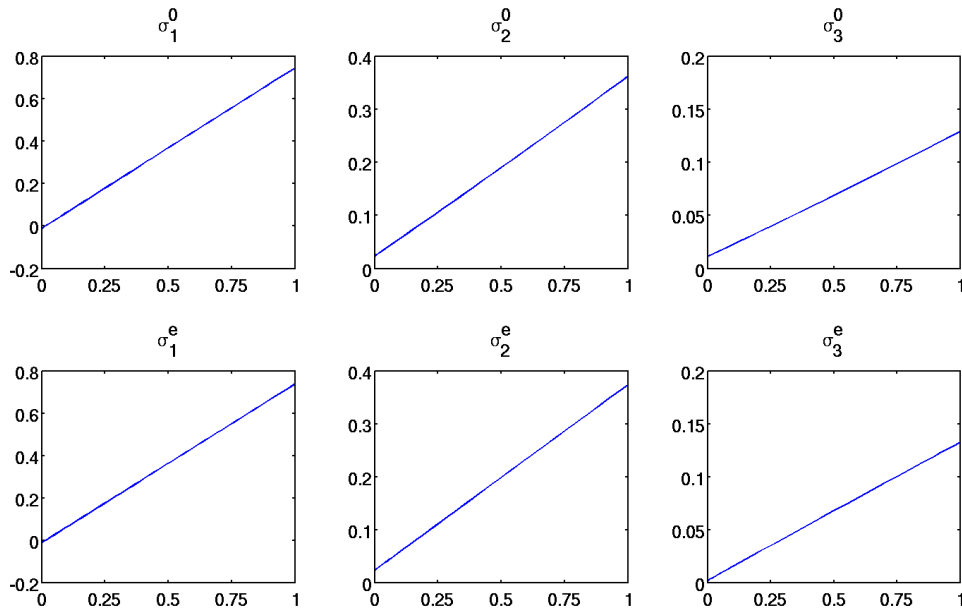
$$\sigma_S^e = V_4 \sigma_S^0.$$

Dabei könnten wir auf die zweite Gleichung verzichten, da wir meist nur an den Anfangswerten interessiert sind. Dies tun wir aber nicht, weil wir die Genauigkeit der Ergebnisse überprüfen möchten. Die Funktionswerte für $S = 0$ bestimmen wir numerisch mit dem Mehrfachschießverfahren, wobei wir als Startwerte Nullen wählen, und erhalten dann die folgenden relativen Fehler für $S = 1$:

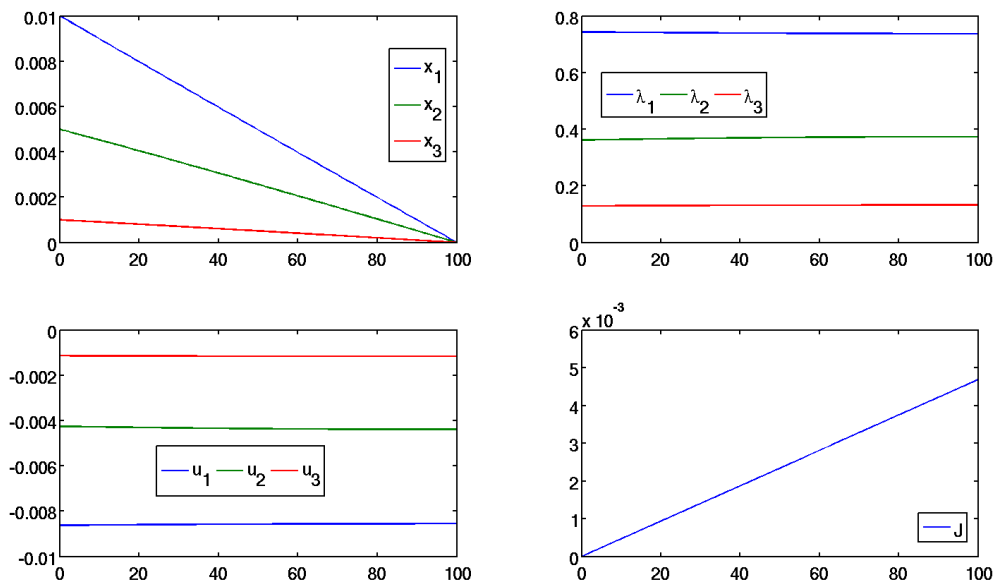
| | relativer Fehler |
|----------------------|------------------------|
| $\sigma_1^0(1, 100)$ | $2.714 \cdot 10^{-13}$ |
| $\sigma_2^0(1, 100)$ | $1.036 \cdot 10^{-13}$ |
| $\sigma_3^0(1, 100)$ | $8.407 \cdot 10^{-12}$ |
| $\sigma_1^e(1, 100)$ | $2.889 \cdot 10^{-12}$ |
| $\sigma_2^e(1, 100)$ | $1.277 \cdot 10^{-11}$ |
| $\sigma_3^e(1, 100)$ | $2.797 \cdot 10^{-11}$ |

Tabelle 5.1.1.: Relativer Fehler für $S = 1$

Die Anfangs- und Endwerte der Adjungierten werden in ca. 1.6 Sekunden sehr genau berechnet und können somit genutzt werden, um das ursprüngliche Problem zu lösen. Dies erklärt sich auch durch die einfache Struktur von σ^0 und σ^e , da diese lineare Funktionen sind.

Abbildung 5.1.1.: $\sigma^0(S, 100)$ und $\sigma^e(S, 100)$ für $S \in [0, 1]$

Die Lösung des originalen Problems (5.1.1) ist auch von einfacher Struktur, allerdings müssen die Startwerte für das Mehrschießverfahren sorgsam gewählt werden, sofern wir sie nicht mit unserem Ansatz bestimmen. Als Startwerte haben wir gleichverteilte Zufallszahlen zwischen 0 und 1 gewählt, welche für viele andere Beispiele gut funktionieren.

Abbildung 5.1.2.: Lösung von Problem (5.1.1) bzw. (5.1.2) für $S = 1$

In diesem Fall hat das Verfahren für verschiedene Zufallszahlen erst nach durchschnittlich 20 Iterationen konvergiert oder es hat vorher abgebrochen. Das Gleiche passierte auch bei $S = 0$, weswegen wir dort keine Zufallszahlen gewählt hatten.

5.1.2. Bilinearer, quadratischer Regler

Als nächstes Beispiel betrachten wir den bilinearen, quadratischen Regler, wobei hier Q symmetrisch sein soll, d.h.

$$\begin{aligned} \min_u J(u) &= \min_u \int_0^T x^\top(t) Q x(t) + u^\top(t) R u(t) dt + x^\top(0) S x(0) \\ \dot{x}(t) &= A x(t) + (B + N x(t)) u(t) = f(x(t), u(t)) \\ x(T) &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} = x_e \end{aligned} \quad (5.1.3)$$

mit

$$\begin{aligned} A &= \begin{pmatrix} -2 & 0 \\ 3 & -1 \end{pmatrix} & B &= \begin{pmatrix} 1 \\ 1 \end{pmatrix} & N &= \begin{pmatrix} 2.5 & 0.5 \\ 0 & 4 \end{pmatrix} \\ Q &= \begin{pmatrix} 1 & 0.3 \\ 0.3 & 2 \end{pmatrix} & R &= 1. \end{aligned}$$

Dabei sei $S, T \in [0, 1]$. Wir bestimmen wie im vorigen Beispiel die optimale Steuerung, die notwendigen Bedingungen und die zugehörigen Transversalitätsbedingungen.

$$\begin{aligned} \hat{u}(x, \lambda) &= -\frac{\lambda^\top (B + N x)}{2R} \\ \dot{x} &= f(x, \hat{u}(x, \lambda)) & x(T) &= x_e \\ \dot{\lambda} &= -2Qx - A^\top \lambda + N^\top \lambda \hat{u}(x, \lambda) & \lambda(0) &= -2S x(0). \end{aligned}$$

Somit gilt für die Flussfunktion

$$\begin{aligned} \phi(S, 0, \rho^0(S, T), -2S \rho^0(S, T)) &= \begin{pmatrix} \rho^0(S, T) \\ -2S \rho^0(S, T) \end{pmatrix} \\ \phi(S, T, \rho^0(S, T), -2S \rho^0(S, T)) &= \begin{pmatrix} x_e \\ \sigma^e(S, T) \end{pmatrix}. \end{aligned}$$

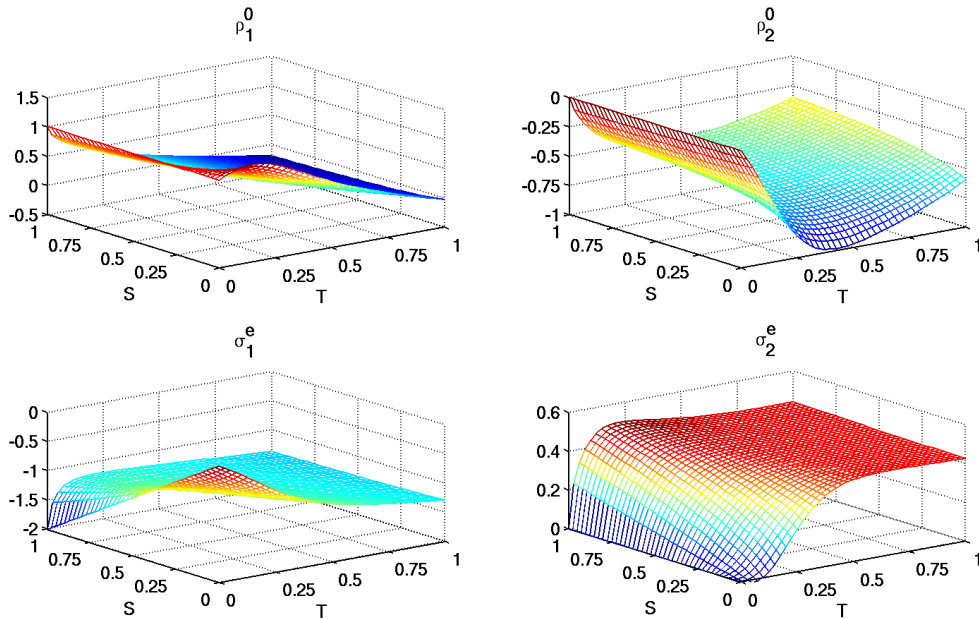
Wir leiten die Flussfunktion zum Zeitpunkt T nach T ab und erhalten das lineare System

$$\begin{aligned} 0 &= F + V_1 \rho_T^0 - 2S V_2 \rho_T^0 \\ \sigma_T^e &= G + V_3 \rho_T^0 - 2S V_4 \rho_T^0. \end{aligned}$$

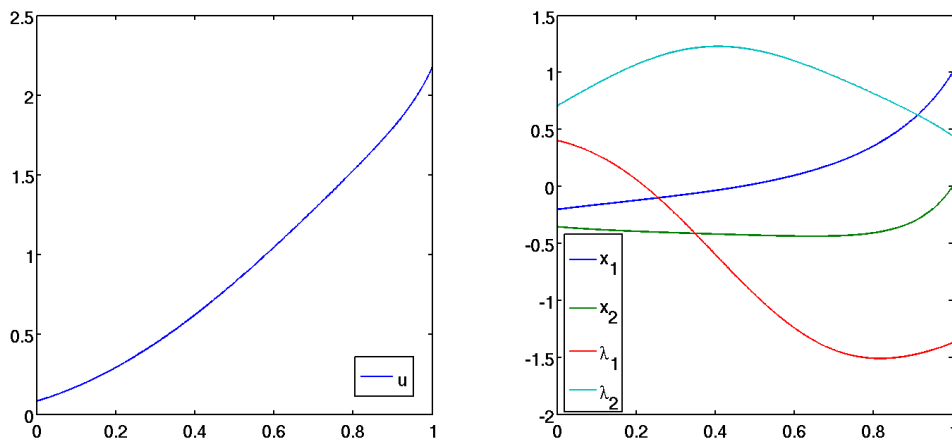
Die Funktionswerte für $T = 0$ lassen sich einfach bestimmen, denn

$$\begin{aligned} \rho^0(S, 0) &= x(T = 0) = x_e \\ \sigma^e(S, 0) &= \lambda(T = 0) = -2S x_e. \end{aligned}$$

Für die obigen Gleichungen erhalten wir die folgende numerische Lösung:

Abbildung 5.1.3.: $\rho^0(S, T)$ und $\sigma^e(S, T)$ für $S, T \in [0, 1]$

Für große S und T verändern sich die Endwerte der Adjungierten nicht signifikant und man kann vermuten, dass diese jeweils gegen einen bestimmten Wert konvergieren. Die Anfangswerte der Zustände jedoch verändern sich weiterhin für große T , wobei dort S nur einen geringen Einfluss auf die Funktionswerte hat.

Abbildung 5.1.4.: Lösung von Problem (5.1.3) für $S = T = 1$

Die obige Lösung haben wir mit dem Mehrfachschießverfahren bestimmt. Wie im vorigen Beispiel liefern Zufallszahlen zwischen 0 und 1 keine guten Startwerte für das Verfahren. In den meisten Fällen kam es nicht einmal zur Konvergenz. Somit kann unser Ansatz hier genutzt werden, um gute Startwerte zu erhalten.

Dies wird insbesondere an der folgenden Tabelle deutlich:

| | relativer Fehler |
|--------------------|-----------------------|
| $\rho_1^0(1, 1)$ | $1.304 \cdot 10^{-3}$ |
| $\rho_2^0(1, 1)$ | $3.156 \cdot 10^{-4}$ |
| $\sigma_1^e(1, 1)$ | $2.625 \cdot 10^{-4}$ |
| $\sigma_2^e(1, 1)$ | $3.626 \cdot 10^{-4}$ |

Tabelle 5.1.2.: Relativer Fehler für $S = T = 1$

Unsere Methode liefert gute Approximationen an die Anfangswerte der Zustände, wobei der Fehler auf Grund der Nichtlinearität von ρ^0 und σ^e größer ist als im vorigen Beispiel. Wir können dadurch als Startwerte $x(0) = \rho^0(S, T)$ und über die Transversalitätsbedingungen $\lambda(0) = -2S\rho^0(S, T)$ für ein konkretes Paar (S, T) wählen.

5.1.3. Beschleunigung eines Wagens

Bisher waren die Randbedingungen in unseren Beispielen stets linear in den Anfangs- und Endwerten der Zustände. Das folgende Optimalsteuerungsproblem beschreibt ein Fahrzeug, welches sich nur in eine Richtung bewegt, wobei wir die Beschleunigung des Wagen steuern können.

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^T \frac{1}{2} u^2(t) dt + S x_2^2(T) \\ \dot{x}(t) &= \begin{pmatrix} x_2(t) \\ u(t) \end{pmatrix} \\ \psi(S, x(0), x(T)) &= \begin{pmatrix} x_1(0) - 0 \\ x_2(0) - 1 \\ x_1^2(T) + x_2^2(T) - 1 \end{pmatrix} = 0_3. \end{aligned} \quad (5.1.4)$$

Neben den linearen Anfangsbedingungen haben wir hier eine zusätzliche, nichtlineare Randbedingung. Es folgt automatisch $|x_i(T)| \in [0, 1]$ und somit können wir mit dem Parameter S beeinflussen, welcher der beiden Endwerte (absolut) größer sein soll.

Wir bestimmen die optimale Steuerung und stellen die notwendigen Bedingungen auf

$$\begin{aligned} \hat{u}(x, \lambda) &= -\lambda_2 \\ \dot{x} &= \begin{pmatrix} x_2 \\ \hat{u}(x, \lambda) \end{pmatrix} & x(0) &= x_0 \\ \lambda &= \begin{pmatrix} 0 \\ -\lambda_1 \end{pmatrix}. \end{aligned}$$

Die zugehörigen Transversalitätsbedingungen lauten

$$\begin{pmatrix} -\lambda_1(0) \\ -\lambda_2(0) \\ \lambda_1(T) \\ \lambda_2(T) \end{pmatrix} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ 2x_1(T)\mu_3 \\ 2x_2(T)(S + \mu_3) \end{pmatrix} = \mathcal{K}(S, x(0), x(T), \mu). \quad (5.1.5)$$

Im Gegensatz zu den vorigen Beispielen werden wir das lineare System aus unserem Ansatz nicht, wie in Abschnitt 4.1.1 beschrieben, reduzieren, obwohl das hier möglich ist. Wir können dadurch kontrollieren, ob sich die Funktionen, die sich trivial aus den Rand- oder Transversalitätsbedingungen analytisch bestimmen lassen, hinreichend genau approximiert werden. Wir lösen also das volle System der Form (3.1.13), wobei wir die Ableitungen von \mathcal{K} und ψ nach den Anfangs- und Endzuständen mit Hilfe der Complex step differentiation (4.1.9) bestimmen.

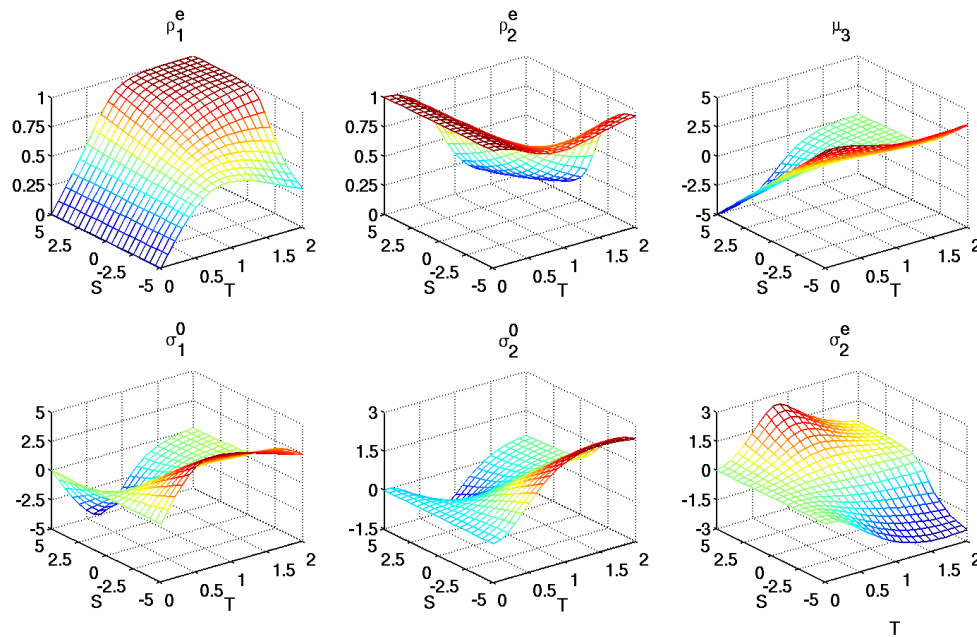
Leider ist es nicht möglich $T_0 = 0$ zu wählen, um Startwerte für unsere gesuchten Funktionen analytisch zu berechnen, da $\sigma_2^0(S, 0)$, $\sigma_2^e(S, 0)$ und $\hat{\mu}_2(S, 0)$ von $\hat{\mu}_3(S, 0)$ abhängen, welches nicht eindeutig bestimmt werden kann. Wir benutzen deshalb das Mehrfachschießverfahren für eines kleines $T_0 > 0$, z.B. $T_0 = 10^{-6}$, um Startwerte zu erhalten, wobei wir die folgenden Gleichungen als Randbedingungen benutzen

$$\begin{aligned} x(0) &= x_0 \\ x_1^2(T_0) + x_2^2(T_0) &= 1 \\ x_2(T_0)\lambda_1(T_0) &= x_1(T_0)\lambda_2(T_0) - 2S x_1(T_0)x_2(T_0). \end{aligned}$$

Die letzte Gleichung ergibt sich durch Umformen der Transversalitätsbedingungen für $\lambda_i(T_0)$ nach μ_3 und Gleichsetzen der entstehenden Terme. Ferner wurden μ_1 und μ_2 durch die Adjungierten ausgedrückt. Wir erhalten damit gute Approximationen an $x(T_0)$, $\lambda(0)$ und $\lambda(T_0)$ mit denen μ durch die Transversalitätsbedingungen (5.1.5) bestimmt werden kann. Dabei muss das Mehrfachschießverfahren nur einmal für ein S_0 aufgerufen werden. Die restlichen Startwerte können wir durch unseren Ansatz mit variiertem S bestimmen. Wir lösen dazu das volle System (3.2.4), wobei auch hier die Ableitungen von \mathcal{K} und ψ mit Complex step differentiation approximiert werden.

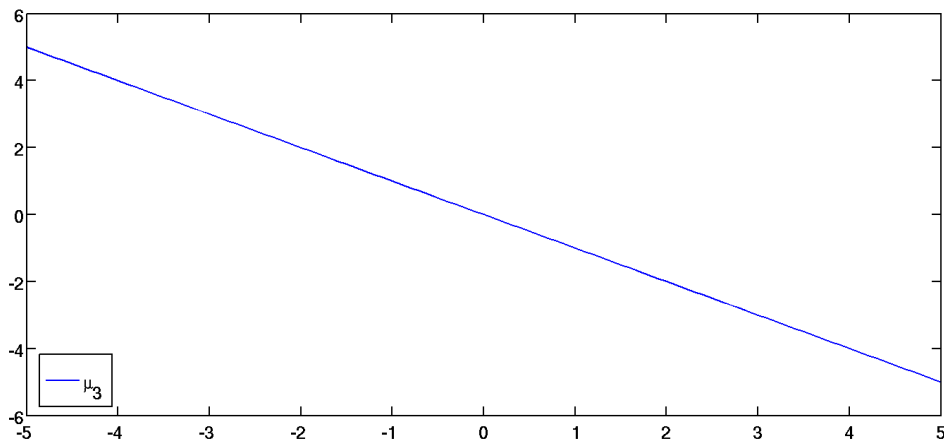
Unser Ansatz liefert dann das folgende Ergebnis, wobei wir nur sechs der elf Funktionen darstellen, da die anderen fünf Funktionen konstant sind oder leicht aus den sechs Funktionen berechnet werden können, d.h.

$$\begin{aligned} \rho^0(S, T) &= x_0 \\ \hat{\mu}_{1,2}(S, T) &= -\sigma^0(S, T) \\ \sigma_1^0(S, T) &= \sigma_1^e(S, T). \end{aligned}$$

Abbildung 5.1.5.: Zeile 1: $\rho_1^e(S, T)$, $\rho_2^e(S, T)$ und $\hat{\mu}_3(S, T)$,

Zeile 2: $\sigma_1^0(S, T)$, $\sigma_2^0(S, T)$ und $\sigma_2^e(S, T)$ für $(S, T) \in [-5, 5] \times [10^{-6}, 2]$

Wir sehen, dass wir mit dem Parameter S , wie zu erwarten, beeinflussen können, welche der beiden Endzustände größer ist. Für große S geht $\rho_1^e(S, T)$ gegen 1 während für negative S $\rho_2^e(S, T)$ Werte nahe der 1 annimmt. Weiterhin ist auffällig, dass $\hat{\mu}_3(S, T)$ für $T \rightarrow 0$ gegen $-S$ konvergiert, welches in der folgenden Abbildung verdeutlicht wird.

Abbildung 5.1.6.: $\hat{\mu}_3(S, T_0)$ für $S \in [-5, 5]$ und $T_0 = 10^{-6}$

Dies impliziert aber nicht, dass wir diesen Wert für $\hat{\mu}_3(S, 0)$ benutzen können, da die Koeffizientenmatrix des linearen Systems (4.1.5) für $T = 0$ singularär ist. Außerdem ist $\hat{\mu}_3(S, T)$ für $T = 0$, wie wir vorher beschrieben haben, frei wählbar.

In der folgenden Tabelle betrachten wir den relativen Fehler der Approximationen, wobei wir für ρ_1^0 nur den absoluten Fehler angeben, da der exakte Wert dort Null ist.

| | $S = -5$ | $S = 0$ | $S = 5$ |
|---------------------|------------------------|------------------------|------------------------|
| $\rho_1^0(S, 2)$ | $3.106 \cdot 10^{-17}$ | $9.968 \cdot 10^{-17}$ | $3.337 \cdot 10^{-17}$ |
| $\rho_2^0(S, 2)$ | $1.110 \cdot 10^{-16}$ | 0 | 0 |
| $\rho_1^e(S, 2)$ | $2.043 \cdot 10^{-05}$ | $1.673 \cdot 10^{-07}$ | $5.665 \cdot 10^{-06}$ |
| $\rho_2^e(S, 2)$ | $5.431 \cdot 10^{-06}$ | $9.631 \cdot 10^{-07}$ | $7.999 \cdot 10^{-05}$ |
| $\sigma_1^0(S, 2)$ | $4.898 \cdot 10^{-06}$ | $6.171 \cdot 10^{-08}$ | $9.018 \cdot 10^{-06}$ |
| $\sigma_2^0(S, 2)$ | $3.465 \cdot 10^{-06}$ | $1.559 \cdot 10^{-07}$ | $4.616 \cdot 10^{-06}$ |
| $\sigma_1^e(S, 2)$ | $4.898 \cdot 10^{-06}$ | $6.171 \cdot 10^{-08}$ | $9.018 \cdot 10^{-06}$ |
| $\sigma_2^e(S, 2)$ | $6.363 \cdot 10^{-06}$ | $1.075 \cdot 10^{-06}$ | $3.277 \cdot 10^{-06}$ |
| $\hat{\mu}_1(S, 2)$ | $4.898 \cdot 10^{-06}$ | $6.171 \cdot 10^{-08}$ | $9.018 \cdot 10^{-06}$ |
| $\hat{\mu}_2(S, 2)$ | $3.465 \cdot 10^{-06}$ | $1.559 \cdot 10^{-07}$ | $4.616 \cdot 10^{-06}$ |
| $\hat{\mu}_3(S, 2)$ | $3.487 \cdot 10^{-06}$ | $1.360 \cdot 10^{-06}$ | $5.602 \cdot 10^{-04}$ |

Tabelle 5.1.3.: Relativer bzw. absoluter Fehler für $S \in \{-5, 0, 5\}$ und $T = 2$

Die Anfangsbedingung $\rho^0(S, T) = x_0$ wird im numerischen Sinne exakt bestimmt und es ist somit sinnvoll, dass wir im Allgemeinen lineare Randbedingungen aus den linearen Systemen herausnehmen. Für alle drei Werte von S sind die Approximationen gut, wobei die besten Näherungen bei $S = 0$ zu verzeichnen sind.

Die optimalen Zustände und Steuerungen für die in Tabelle 5.1.3. beschriebenen Fälle sehen wie folgt aus:

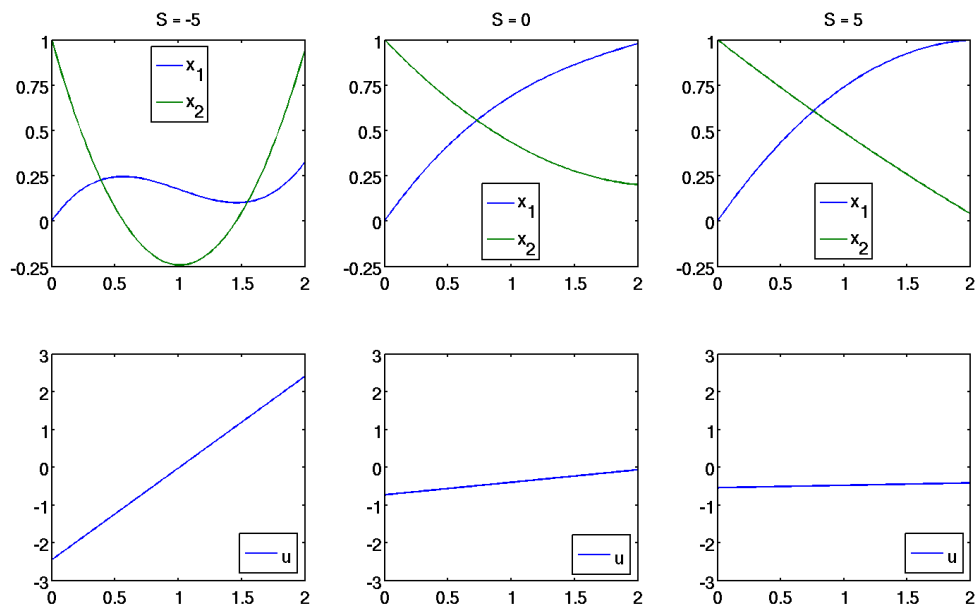


Abbildung 5.1.7.: Optimale Zustände und Steuerungen für $T = 2$ und $S \in \{-5, 0, 5\}$

Je kleiner wir S wählen, desto größer ist der Anstieg der Beschleunigung bzw. Steuerung. Für positive S scheint es so, als würde die Beschleunigungsfunktion gegen eine konstante Funktion konvergieren. Aufgrund dessen nähert sich die Geschwindigkeit für große S einer fallenden, linearen Funktion an.

5.2. Apollo-Problem

In diesem Abschnitt behandeln wir das Apollo-Problem [26, S. 491-497], welches wir in Kapitel 1 erwähnt haben. Dabei soll die Raumkapsel bei minimaler Aufheizung in die Erdatmosphäre eintreten und eine gewünschte Position, Geschwindigkeit und Winkel bzgl. der Erdoberfläche erreichen.

$$\begin{aligned}
 \min_{u,T} \quad J(u, T) &= \min_{u,T} \int_0^T 10v^3(t) \sqrt{p(t)} \, dt \\
 \dot{v}(t) &= -\frac{Sp(t)v^2(t)}{2m} C_D(u(t)) - \frac{g \sin \gamma(t)}{(\xi(t) + 1)^2} \\
 \dot{\gamma}(t) &= \frac{Sp(t)v(t)}{2m} C_L(u(t)) + \frac{v(t) \cos \gamma(t)}{R(\xi(t) + 1)} - \frac{g \cos \gamma(t)}{v(t)(\xi(t) + 1)^2} \\
 \dot{\xi}(t) &= \frac{v(t) \sin \gamma(t)}{R} \\
 \begin{pmatrix} v(0) \\ \gamma(0) \\ \xi(0) \end{pmatrix} &= \begin{pmatrix} 0.36 \\ -8.1 \frac{\pi}{180} \\ \frac{4}{R} \end{pmatrix} = x_0 \\
 \begin{pmatrix} v(T) \\ \gamma(T) \\ \xi(T) \end{pmatrix} &= \begin{pmatrix} 0.27 \\ 0 \\ \frac{2.5}{R} \end{pmatrix} = x_e \\
 p(t) &= p_0 e^{-\beta R \xi(t)} \\
 C_D(u) &= 1.174 - 0.9 \cos u \\
 C_L(u) &= 0.6 \sin u.
 \end{aligned} \tag{5.2.1}$$

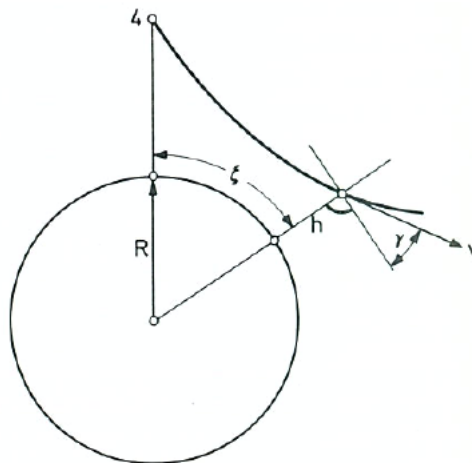


Abbildung 5.2.1.: Kenngrößen der Trajektorie aus [26, S. 492]

Die Konstanten und Variablen werden in der folgenden Tabelle erklärt, wobei in der vorigen Abbildung 5.2.1. $h = R\xi$ ist.

| Name | Wert | Einheit | Bedeutung |
|----------------|------------------------|--------------------------|--|
| R | 209 | 10^5 ft | Erdradius |
| β | 4.26 | $1/\text{ft}$ | - |
| p_0 | $2.704 \cdot 10^{-3}$ | slug/ft ³ | Luftdichte für $\xi = 0$ |
| g | $3.2172 \cdot 10^{-4}$ | 10^5 ft/s ² | Erdgravitation |
| S/m | $5.32 \cdot 10^4$ | ft ² /slug | Stirnfläche durch Masse der Kapsel |
| $v = x_1$ | - | 10^5 ft/s | Geschwindigkeit |
| $\gamma = x_2$ | - | rad | Flugwinkel bzgl. der Erdoberfläche |
| $\xi = x_3$ | - | $10^5/R$ ft | normierte Höhe bzgl. der Erdoberfläche |
| p | - | slug/ft ³ | Luftdichte |
| C_D | - | - | Luftwiderstandskoeffizient |
| C_L | - | - | Aerodynamischer Auftriebskoeffizient |
| u | - | rad | Anstellwinkel |

Tabelle 5.2.1.: Konstanten und Variablen des Apollo-Problems

Das dazugehörige Randwertproblem ist schwierig zu lösen, da die Menge der Anfangswerte, von denen aus die Endwerte erreicht werden können, sehr klein ist und man eine gute Startlösung braucht, damit das Mehrfachschießverfahren konvergiert. Unser Ziel ist es jetzt, mit der neuen Methode eine gute Startlösung für das ursprüngliche Problem zu erhalten. Dazu schwächen wir das Problem wie folgt ab, wobei wir S/m ab jetzt durch S_m ausdrücken, um Verwechslungen mit dem S aus unserer Methode zu vermeiden, und als Funktionsnamen x_1 bis x_3 wie in der obigen Tabelle angegeben benutzen:

$$\begin{aligned}
 \min_u J(u) &= \min_u \int_0^T 10x_1^3(t) \sqrt{p(t)} dt + (x(T) - x_e)^\top S (x(T) - x_e) \\
 \dot{x}_1(t) &= -\frac{S_m p(t) x_1^2(t)}{2} C_D(u(t)) - \frac{g \sin x_2(t)}{(x_3(t) + 1)^2} \\
 \dot{x}_2(t) &= \frac{S_m p(t) x_1(t)}{2} C_L(u(t)) + \frac{x_1(t) \cos x_2(t)}{R(x_3(t) + 1)} - \frac{g \cos x_2(t)}{x_1(t)(x_3(t) + 1)^2} \\
 \dot{x}_3(t) &= \frac{x_1(t) \sin x_2(t)}{R} \\
 x(0) &= x_0.
 \end{aligned} \tag{5.2.2}$$

Ziel dieses modifizierten Problems ist es, die Endbedingung möglichst gut zu erreichen ohne sie zu erzwingen und gleichzeitig die Aufwärmung zu minimieren. Jedoch haben wir hier die Möglichkeit, die feste Endzeit T zu variieren. Mit dem Parameter S können wir beeinflussen, wie gut die Randbedingung erreicht werden soll.

Im Folgenden kürzen wir die rechte Seite der Differentialgleichungen mit $f(x(\cdot), u(\cdot)) := \dot{x}(\cdot)$ ab und lassen t als Argument der Funktionen auf der rechten Seite weg. Die zugehörige Hamilton-Funktion des modifizierten Problems (5.2.2) lautet

$$\mathcal{H}(x, \lambda, u) = 10x_1^3 \sqrt{p} + \lambda^\top f(x, u),$$

Wir leiten sie nach u ab und können dann unsere optimale Steuerung \hat{u} mit Hilfe von x und λ ausdrücken, wobei wir $x_1(\cdot) > 0$ voraussetzen, so dass

$$\begin{aligned} \mathcal{H}_u(x, \lambda, u) &= \frac{S_m p x_1}{2} (0.6\lambda_2 \cos u - 0.9x_1\lambda_1 \sin u) \stackrel{!}{=} 0 \\ \Rightarrow \sin \hat{u} &= -\frac{0.6\lambda_2}{\sqrt{(0.9x_1\lambda_1)^2 + (0.6\lambda_2)^2}} \\ \Rightarrow \cos \hat{u} &= -\frac{0.9x_1\lambda_1}{\sqrt{(0.9x_1\lambda_1)^2 + (0.6\lambda_2)^2}}. \end{aligned}$$

Wir stellen die notwendigen Bedingungen auf und es ergibt sich

$$\begin{aligned} \dot{x}(t) &= f(x, \hat{u}) \\ x(0) &= x_0 \\ \dot{\lambda}_1(t) &= -30x_1^2 \sqrt{p} + \lambda_1 S_m x_1 p C_D(\hat{u}) \\ &\quad - \lambda_2 \left(\frac{S_m p}{2} C_L(\hat{u}) + \frac{\cos x_2}{R(1+x_3)} + \frac{g \cos x_2}{x_1^2(1+x_3)^2} \right) - \lambda_3 \frac{\sin x_2}{R} \\ \dot{\lambda}_2(t) &= \lambda_1 \frac{g \cos x_2}{(1+x_3)^2} + \lambda_2 \left(\frac{x_1 \sin x_2}{R(1+x_3)} - \frac{g \sin x_2}{x_1(1+x_3)^2} \right) - \lambda_3 \frac{x_1 \cos x_2}{R} \\ \dot{\lambda}_3(t) &= 5\beta R x_1^3 \sqrt{p} - \lambda_1 \left(\frac{S_m \beta R x_1^2 p}{2} C_D(\hat{u}) + \frac{2g \sin x_2}{(1+x_3)^3} \right) \\ &\quad + \lambda_2 \left(\frac{S_m \beta R x_1 p}{2} C_L(\hat{u}) + \frac{x_1 \cos x_2}{R(1+x_3)^2} - \frac{2g \cos x_2}{x_1(1+x_3)^3} \right) \\ \lambda(T) &= 2S(x(T) - x_e). \end{aligned} \tag{5.2.3}$$

Für die Flussfunktion ϕ gilt

$$\begin{aligned} \phi(S, 0, \rho^0(S, T), \sigma^0(S, T)) &= \begin{pmatrix} x_0 \\ \sigma^0(S, T) \end{pmatrix} \\ \phi(S, T, \rho^e(S, T), \sigma^e(S, T)) &= \begin{pmatrix} \rho^e(S, T) \\ 2S(\rho^e(S, T) - x_e) \end{pmatrix}. \end{aligned}$$

Flussfunktion total nach T abgeleitet ergibt

$$\begin{aligned} \rho_T^e(S, T) &= F(\rho^e(S, T), 2S(\rho^e(S, T) - x_e)) + V_2 \sigma_T^0(S, T) \\ 2S \rho_T^e(S, T) &= G(\rho^e(S, T), 2S(\rho^e(S, T) - x_e)) + V_4 \sigma_T^0(S, T). \end{aligned}$$

Die Anfangsbedingungen dabei sind

$$\begin{aligned}\rho^e(S_0, 0) &= x_0 \\ \sigma^0(S_0, 0) &= 2S_0(x_0 - x_e).\end{aligned}$$

Wir wählen $S_0 > 0$, da aus $S = 0$ die Transversalitätsbedingung $\lambda(T) = 0$ folgt und somit die optimale Steuerung \hat{u} nicht nach x und λ aufgelöst werden kann. Als numerische Lösung für $S \in [1, 200]$ und $T \in [1, 200]$ erhalten wir

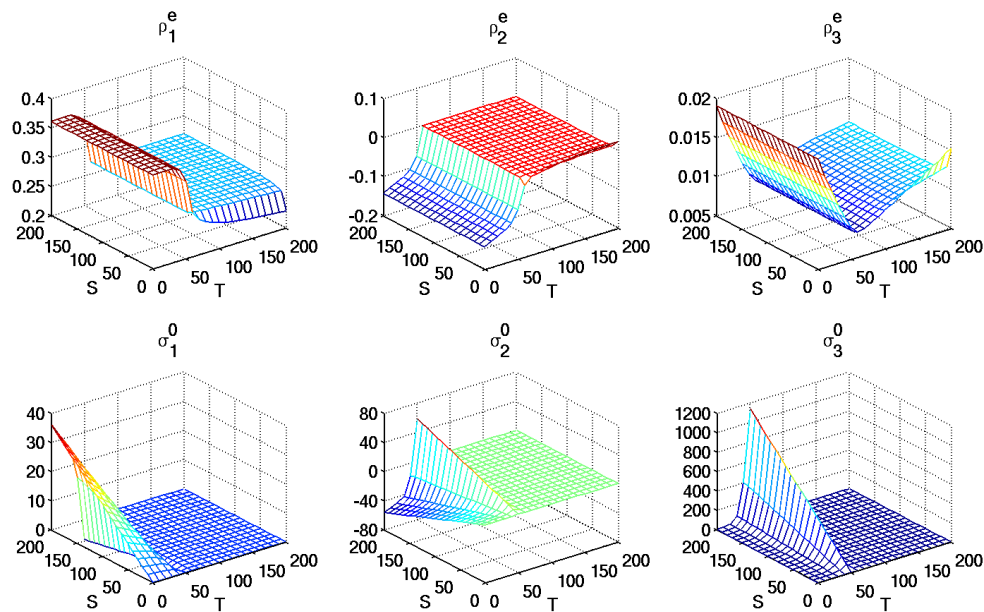
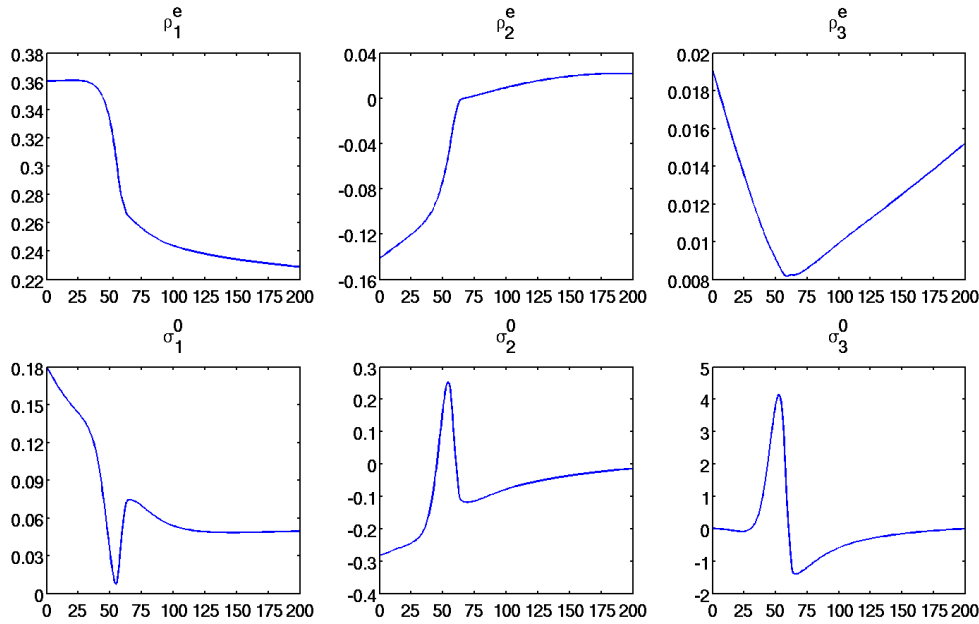


Abbildung 5.2.2.: $\rho^e(S, T)$ und $\sigma^0(S, T)$ für $S \in [1, 200]$ und $T \in [0, 200]$

Wir sehen, dass sich der Verlauf der gesuchten Funktionen bis $T \approx 50$ noch stark ändert und danach T kaum einen Einfluss auf die Funktionswerte hat. Die Veränderungen auf S bezogen sind bei den meisten Funktionen sehr gering, jedoch ist dies bei σ_{2-3}^e nicht der Fall. Dort wächst der Funktionswert vor allem bei $T \approx 50$ stark für steigende S . Dies führt auch dazu, dass die Berechnung mit größerem S mehr Zeit in Anspruch nimmt. So benötigen wir z.B. für $S = 1$ knapp 35 Sekunden und für $S = 200$ rund 128 Sekunden.

Ziel ist es im Folgenden, den Wert für die gesuchten Funktionen bei $S = 200$ und $T = 200$ schneller zu berechnen. Dazu teilen wir die Berechnungen in zwei Phasen auf. In der ersten Phase berechnen wir die Funktionswerte für ein kleines $S_0 > 0$ und $T \in [0, 200]$. Danach fixieren wir T und vergrößern S , so dass $x(T) \approx x_e$. Das ist die zweite Phase. Für die erste Phase wählen wir $S_0 = 1$ und erhalten

Abbildung 5.2.3.: $\rho^e(1, T)$ und $\sigma^0(1, T)$ für $T \in [0, 200]$

Bei x_1 und x_2 wird die gewünschte Endbedingung ungefähr bei $T \approx 60$ angenommen und entfernt sich für größere T dann von dieser. Dies sieht man auch bei den Adjungierten, die bei $T \approx 50$ ihr Maximum bzw. Minimum annehmen und deren Verlauf sich für $T \in [40, 60]$ am stärksten verändert. Für größere T scheinen sich die Anfangswerte der Adjungierten nicht mehr signifikant zu ändern.

Wir kontrollieren unsere Ergebnisse, indem wir sie als Startlösung für ein Mehrfachschießverfahren für das Randwertproblem (5.2.3) benutzen. Außerdem können wir dadurch bessere Approximationen an die Endwerte von x und die Anfangswerte von λ erhalten.

| | relativer Fehler |
|----------------------|-----------------------|
| $\rho_1^e(1, 200)$ | $3.429 \cdot 10^{-7}$ |
| $\rho_2^e(1, 200)$ | $5.038 \cdot 10^{-5}$ |
| $\rho_3^e(1, 200)$ | $1.712 \cdot 10^{-5}$ |
| $\sigma_1^0(1, 200)$ | $6.344 \cdot 10^{-5}$ |
| $\sigma_2^0(1, 200)$ | $5.732 \cdot 10^{-4}$ |
| $\sigma_3^0(1, 200)$ | $2.500 \cdot 10^{-1}$ |

Tabelle 5.2.2.: Relativer Fehler für $S_0 = 1$ und $T = 200$

Das Mehrfachschießverfahren benötigt nur zwei Iterationen, um den Fehler kleiner als 10^{-10} zu machen. Außerdem sind die Ergebnisse bis auf $\sigma_3^0(1, 200)$ sehr gut und könnten somit auch ohne Verbesserung genutzt werden.

In der zweiten Phase möchten wir durch eine Vergrößerung von S erreichen, dass $x(T) \approx x_e$. Wir leiten die Flussfunktion total nach S ab und erhalten die neuen Vektordifferentialgleichungen

$$\begin{aligned}\rho_S^e(S, T) &= V_2 \sigma_S^0(S, T) \\ 2(\rho^e(S, T) - x_e + S \rho_S^e(S, T)) &= V_4 \sigma_S^0(S, T).\end{aligned}$$

Als Startwerte für $\rho^e(1, 200)$ bzw. $\sigma^0(1, 200)$ wählen wir die Lösung, die uns das Mehrschießverfahren geliefert hat. Wir rechnen bis $S = 200$ und es ergibt sich

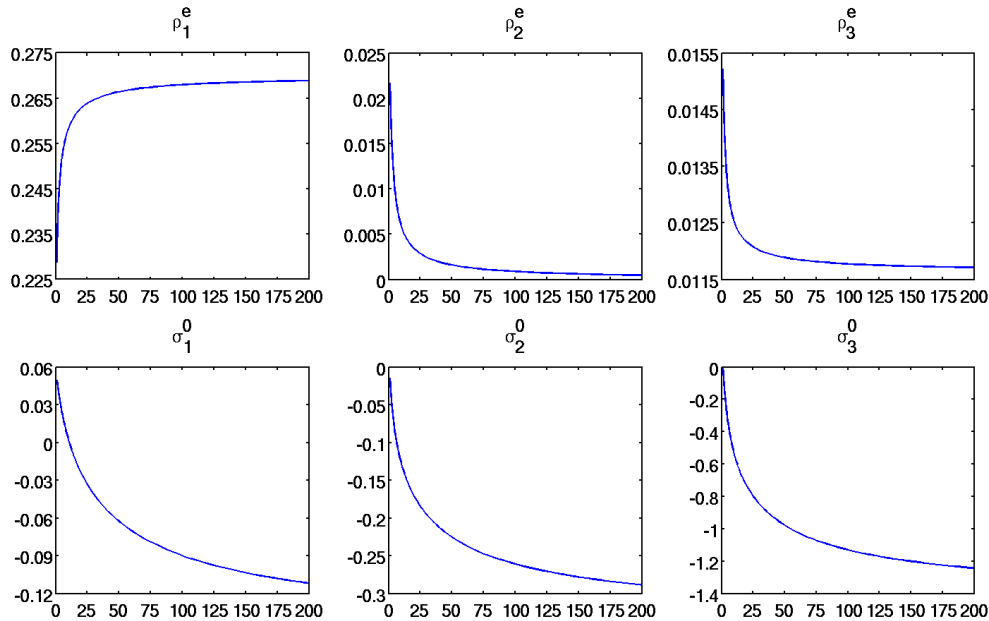


Abbildung 5.2.4.: $\rho^e(S, 200)$ und $\sigma^0(S, 200)$ für $S \in [1, 200]$

Die Randwertfunktionen $\rho^e(\cdot, 200)$ verhalten sich wie fallende Exponentialfunktionen und nähern sich somit schon für kleine S x_e an und verändern sich für $S \geq 150$ kaum noch. Die Anfangswerte der Adjungierten fallen eher wie quadratische Funktionen und scheinen deshalb auch für $S \geq 150$ noch nicht zu konvergieren.

| | relativer Fehler |
|------------------------|-----------------------|
| $\rho_1^e(200, 200)$ | $4.434 \cdot 10^{-8}$ |
| $\rho_2^e(200, 200)$ | $2.191 \cdot 10^{-5}$ |
| $\rho_3^e(200, 200)$ | $2.239 \cdot 10^{-6}$ |
| $\sigma_1^0(200, 200)$ | $1.553 \cdot 10^{-5}$ |
| $\sigma_2^0(200, 200)$ | $7.645 \cdot 10^{-6}$ |
| $\sigma_3^0(200, 200)$ | $7.087 \cdot 10^{-6}$ |

Tabelle 5.2.3.: Relativer Fehler für $S = 200$ und $T = 200$

Auch hier benötigt das Mehrfachschießverfahren nur zwei Durchläufe, um das Ergebnis zu verbessern. Im Gegensatz zum vorigen Verfahren sind hier alle Ergebnisse zufriedenstellend und sind somit auch geeignet, um als Startlösung für das Lösen des Apollo-Problems mittels Mehrfachschießverfahren benutzt zu werden. Dabei ist in den folgenden Darstellungen zu beachten, dass wir die Steuerung nicht im Bogenmaß sondern im Winkelmaß angeben.

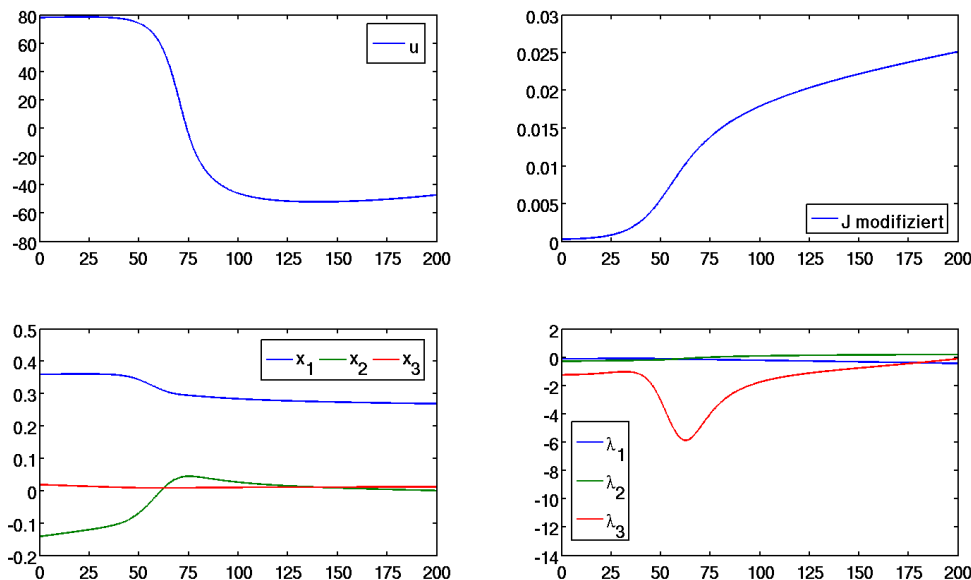


Abbildung 5.2.5.: Lösung von Problem (5.2.2) für $S = 200$ und $T = 200$

Bei der Darstellung der Zielfunktion J unseres modifizierten Problems (5.2.2) ist zu beachten, dass der Strafterm $\varphi(S, x(T)) = (x(T) - x_e)^\top S (x(T) - x_e)$ schon für $T = 0$ dazu addiert wurde und somit der Wert der Zielfunktion an dieser Stelle bereits positiv ist.

Bevor wir die obige Lösung als Approximation für das Apollo-Problem benutzen, wandeln wir das ursprüngliche Problem (5.2.1) mit freier Endzeit in ein äquivalentes Problem mit fester Endzeit um, damit wir die Ergebnisse aus der letzten Verbesserung als Startlösung benutzen können, d.h.

$$\begin{aligned}
 \min_u J(u) &= \min_u \int_0^1 x_4(\tau) \left(10x_1^3(\tau) \sqrt{p(\tau)} \right) d\tau \\
 \dot{x}_{1-3}(\tau) &= x_4(\tau) f(x_{1-3}(\tau), u(\tau)) \\
 \dot{x}_4(\tau) &= 0 \\
 x_{1-3}(0) &= x_0 \\
 x_{1-3}(1) &= x_e
 \end{aligned} \tag{5.2.4}$$

mit den zugehörigen Gleichungen für λ_4

$$\begin{aligned}\lambda_4(\tau) &= -\left(10x_1^3(\tau)\sqrt{p(\tau)} + \lambda_{1-3}^\top(\tau)f(x_{1-3}(\tau), \hat{u}(\tau))\right) \\ \lambda_4(0) &= \lambda_4(1) = 0.\end{aligned}$$

Wir wählen für die erste Iteration des Mehrfachschießverfahrens $x_4(\cdot) = 200$ und $\lambda_4(\cdot) = 0$.

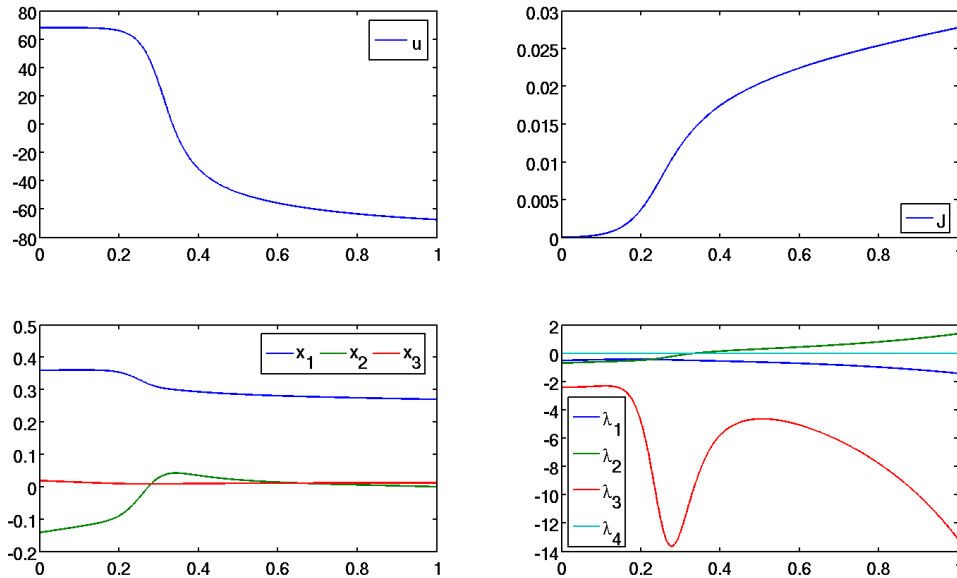


Abbildung 5.2.6.: Lösung des Apollo-Problems (5.2.4)

Nach 13 Iterationen erhalten wir die obige Lösung und sehen, dass die Startlösung bereits eine sehr gute Approximation für fast alle Funktionen des Apollo-Problems liefert, wobei wir die größte Abweichung bei λ_3 feststellen. Als Endzeit erhalten wir $T = x_4(\cdot) \approx 224.9$ Sekunden. Unsere Methode kann dieses komplexe Problem also ohne Schwierigkeiten lösen.

Ferner ist es auch möglich S und T weniger konservativ, d.h. nahe der Lösung des originalen Problems zu wählen, z.B. liefert $(S, T) = (12, 60)$ auch eine hinreichend gute Startlösung, wobei hier 22 Iterationen durchlaufen werden. Wählen wir T noch kleiner, so ist unsere Startlösung nicht mehr gut genug. Es scheint sich also zu bestätigen, dass erst ab $T \approx 60$ die Endbedingungen erreicht werden können.

Unser ursprüngliches Ziel war es, die Berechnung durch die Aufteilung in zwei Phasen zu beschleunigen. Weil in diesem Beispiel die Adjungiertengleichungen nicht trivial sind, eignet sich das Apollo-Problem auch um verschiedene Möglichkeiten zur Berechnung von \mathcal{A} (siehe Abschnitt 4.1.2) zu vergleichen, z.B. über zentralen Differenzenquotienten (4.1.8) und Complex step differentiation (4.1.9). Beide Methoden testen wir auch am Mehrfachschießverfahren.

| | Complex step | | Zentraler DQ | |
|--------------------------------|--------------|----------|--------------|----------|
| | Iterationen | Laufzeit | Iterationen | Laufzeit |
| $S_0 = 1$ und $T \in [0, 200]$ | | 34.8 s | | 31.5 s |
| Verbessere Approximation | 2 | 0.7 s | 2 | 1.1 s |
| $S \in [1, 200]$ und $T = 200$ | | 16.9 s | | 23.2 s |
| Verbessere Approximation | 2 | 0.7 s | 2 | 1.1 s |
| Löse ursprüngliches Problem | 13 | 8.5 s | 13 | 13.6 s |
| Gesamt | | 61.6 s | | 70.5 s |

Tabelle 5.2.4.: Laufzeiten für Lösung des Apollo-Problems

Zuerst können wir feststellen, dass durch die Aufteilung in zwei Phasen die Dauer der Berechnung mehr als halbiert wird, obwohl wir in unserem ersten Ansatz die Zeit zur Lösung des Apollo-Problem noch nicht berücksichtigt hatten. Hinsichtlich der Genauigkeit unterscheiden sich beide Methoden zur Approximation der partiellen Ableitungen hier nur unwesentlich. Allerdings können wir feststellen, dass die Berechnung durch den Einsatz von Complex step differentiation signifikant beschleunigt wird - und zwar um den Faktor ein Drittel. Dementsprechend haben wir für alle weiteren Beispiele den zentralen Differenzenquotienten zur Berechnung von \mathcal{A} und der Jacobi-Matrix im Mehrfachschießverfahren nicht mehr benutzt. Allerdings dauert die Berechnung in der ersten Phase mit Complex step differentiation länger, obwohl die Anzahl der Integrationssschritte nahezu gleich ist, d.h. 91,794 bzw. 91,782 Funktionsaufrufe in der inneren Integration.

Wie wir gesehen haben, nimmt die numerische Berechnung von \mathcal{A} einen großen Teil der Rechenzeit in Anspruch. Wir untersuchen dies weiter, in dem wir \mathcal{A} analytisch berechnen und somit nur ein Funktionsaufruf für die Berechnung an einer Stelle benötigt wird.

| | Complex step | | Analytisch | |
|--------------------------------|--------------|----------|-------------|----------|
| | Iterationen | Laufzeit | Iterationen | Laufzeit |
| $S_0 = 1$ und $T \in [0, 200]$ | | 34.8 s | | 10.6 s |
| Verbessere Approximation | 2 | 0.7 s | 2 | 0.7 s |
| $S \in [1, 200]$ und $T = 200$ | | 16.9 s | | 7.7 s |
| Verbessere Approximation | 2 | 0.7 s | 2 | 0.7 s |
| Löse ursprüngliches Problem | 13 | 8.5 s | 13 | 8.5 s |
| Gesamt | | 61.6 s | | 28.2 s |

Tabelle 5.2.5.: Vergleich der Laufzeiten bei verschiedener Berechnung von \mathcal{A}

Wie zu erwarten, ist die analytische Berechnung deutlich schneller. Die Laufzeit wird mehr als halbiert, wobei der relative Fehler in beiden Fällen nahezu identisch mit den Werten aus den vorigen Tabellen ist. Somit liefert Complex step differentiation eine sehr genaue Approximation an \mathcal{A} .

Eine Alternative zum Einschrittverfahren *ode45* ist das in MATLAB implementierte *ode113*, welches ein Mehrschrittverfahren ist. Dieses basiert auf einem Adams-Bashforth-Moulton-Löser [24] und benötigt somit eventuell weniger Funktionsauswertungen als ein Einschrittverfahren. Wir vergleichen zunächst die Laufzeiten beider Verfahren anhand des Apollo-Problems.

| | <i>ode45</i> | | <i>ode113</i> | |
|--------------------------------|--------------|----------|---------------|----------|
| | Iterationen | Laufzeit | Iterationen | Laufzeit |
| $S_0 = 1$ und $T \in [0, 200]$ | | 34.8 s | | 28.6 s |
| Verbessere Approximation | 2 | 0.7 s | 3 | 1.1 s |
| $S \in [1, 200]$ und $T = 200$ | | 16.9 s | | 16.1 s |
| Verbessere Approximation | 2 | 0.7 s | 2 | 0.7 s |
| Löse ursprüngliches Problem | 13 | 8.5 s | 13 | 8.5 s |
| Gesamt | | 61.6 s | | 55.0 s |

Tabelle 5.2.6.: Vergleich der Laufzeiten von Einschritt- und Mehrschrittverfahren

Das Mehrschrittverfahren ist in beiden Phasen schneller als das Einschrittverfahren, wobei die Unterschiede in der ersten Phase am größten sind. Allerdings scheint die Approximationen der ersten Phase schlechter zu sein, da das Mehrfachschießverfahren drei anstatt zwei Iterationen benötigt, um die Werte zu verbessern. Wir untersuchen die relativen Fehler beider Verfahren genauer:

| | Phase 1: $S = 1, T = 200$ | | Phase 2: $S = 200, T = 200$ | |
|--------------------|---------------------------|-----------------------|-----------------------------|-----------------------|
| | <i>ode45</i> | <i>ode113</i> | <i>ode45</i> | <i>ode113</i> |
| $\rho_1^e(S, T)$ | $3.429 \cdot 10^{-7}$ | $6.198 \cdot 10^{-6}$ | $4.434 \cdot 10^{-8}$ | $4.672 \cdot 10^{-8}$ |
| $\rho_2^e(S, T)$ | $5.038 \cdot 10^{-5}$ | $7.049 \cdot 10^{-4}$ | $2.191 \cdot 10^{-5}$ | $1.395 \cdot 10^{-5}$ |
| $\rho_3^e(S, T)$ | $1.712 \cdot 10^{-5}$ | $1.962 \cdot 10^{-4}$ | $2.239 \cdot 10^{-6}$ | $2.187 \cdot 10^{-6}$ |
| $\sigma_1^0(S, T)$ | $6.344 \cdot 10^{-5}$ | $7.802 \cdot 10^{-4}$ | $1.553 \cdot 10^{-5}$ | $3.116 \cdot 10^{-5}$ |
| $\sigma_2^0(S, T)$ | $5.732 \cdot 10^{-4}$ | $7.308 \cdot 10^{-3}$ | $7.645 \cdot 10^{-6}$ | $1.408 \cdot 10^{-5}$ |
| $\sigma_3^0(S, T)$ | $2.500 \cdot 10^{-1}$ | $3.237 \cdot 10^{-0}$ | $7.087 \cdot 10^{-6}$ | $1.240 \cdot 10^{-5}$ |

Tabelle 5.2.7.: Vergleich der relativen Fehler von Einschritt- und Mehrschrittverfahren

Das Mehrschrittverfahren liefert in der zweiten Phase Näherungen, die vergleichbar mit den Approximationen von *ode45* sind. Dies ist aber in der ersten Phase nicht der Fall. Dort sind die von *ode113* berechneten um mindestens eine Zehnerpotenz schlechter als die vorher ermittelten Näherungen. Da die zeitliche Ersparnis in keinem Verhältnis zum Verlust der Genauigkeit steht, haben wir uns entschieden, das Einschrittverfahren für alle weiteren Beispiele zu benutzen.

5.3. Probleme mit Beschränkungen

5.3.1. Beschleunigung eines Wagens

In dem folgenden Abschnitt untersuchen wir beschränkte Optimalsteuerungsprobleme. Dabei betrachten wir wieder das Beispiel aus Abschnitt 5.1.3, wobei die Steuerung hier beschränkt sei.

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^3 \frac{1}{2} u^2(t) dt \quad |u(\cdot)| \leq 0.5 \\ \dot{x}(t) &= \begin{pmatrix} x_2(t) \\ u(t) \end{pmatrix} \\ x(0) &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} = x_0 \\ x(3) &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} = x_e. \end{aligned} \tag{5.3.1}$$

Wir approximieren dieses beschränkte Optimalsteuerungsproblem durch das folgende unbeschränkte Problem, weil unser Ansatz die Unbeschränktheit voraussetzt:

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^3 \frac{1}{2} u^2(t) + S \mathcal{P}(u(t), 0.5) dt \quad S \geq 0 \\ \dot{x}(t) &= \begin{pmatrix} x_2(t) \\ u(t) \end{pmatrix} \\ x(0) &= x_0 \\ x(3) &= x_e \end{aligned} \tag{5.3.2}$$

mit der Straffunktion

$$\mathcal{P}(v, c) = \begin{cases} (v+c)^2 & v < -c \\ 0 & \text{sonst} \\ (v-c)^2 & v > c \end{cases}.$$

Wir stellen die Hamilton-Funktion auf und erhalten die optimale Steuerung, d.h.

$$\begin{aligned} \mathcal{H}(x, \lambda, u, S) &= \frac{1}{2} u^2 + S \mathcal{P}(u, 0.5) + \lambda^T \begin{pmatrix} x_2 \\ u \end{pmatrix} \\ \Rightarrow \hat{u}(x, \lambda, S) &= \begin{cases} -\frac{\lambda_2 + S}{1 + 2S} & \lambda_2 > 0.5 \\ -\lambda_2 & \text{sonst} \\ -\frac{\lambda_2 - S}{1 + 2S} & \lambda_2 < -0.5 \end{cases}. \end{aligned}$$

Wir stellen die notwendigen Bedingungen auf und es ergibt sich

$$\begin{aligned} \dot{x} &= \begin{pmatrix} x_2 \\ \hat{u}(x, \lambda, S) \end{pmatrix} \quad x(0) = x_0, \quad x(3) = x_e \\ \dot{\lambda} &= \begin{pmatrix} 0 \\ -\lambda_1 \end{pmatrix}. \end{aligned}$$

Die optimale Steuerung hängt hier explizit von S ab und somit auch die Flussfunktion. Ferner gilt für ϕ

$$\begin{aligned}\phi(S, 0, \rho^0(S, T), \sigma^0(S, T)) &= \begin{pmatrix} x_0 \\ \sigma^0(S, T) \end{pmatrix} \\ \phi(S, T, \rho^0(S, T), \sigma^0(S, T)) &= \begin{pmatrix} x_e \\ \sigma^e(S, T) \end{pmatrix}.\end{aligned}$$

Damit ergibt sich das lineare, implizite System

$$\begin{aligned}0 &= \phi_{1,S}(S, T, x_0, \sigma^0) + V_2 \sigma_S^0 \\ \sigma_S^e &= \phi_{2,S}(S, T, x_0, \sigma^0) + V_4 \sigma_S^0,\end{aligned}$$

wobei man auf die Gleichung für $\sigma_S^e(S, T)$ verzichten kann, wenn man nur an den Anfangswerten der Adjungierten interessiert ist.

Die Funktionswerte für $S = 0$, d.h. es gibt keine Beschränkung an u , bestimmen wir mittels Mehrfachschießverfahren und erhalten dadurch die folgenden Anfangs- und Endwerte der Adjungierten in Abhängigkeit von S :

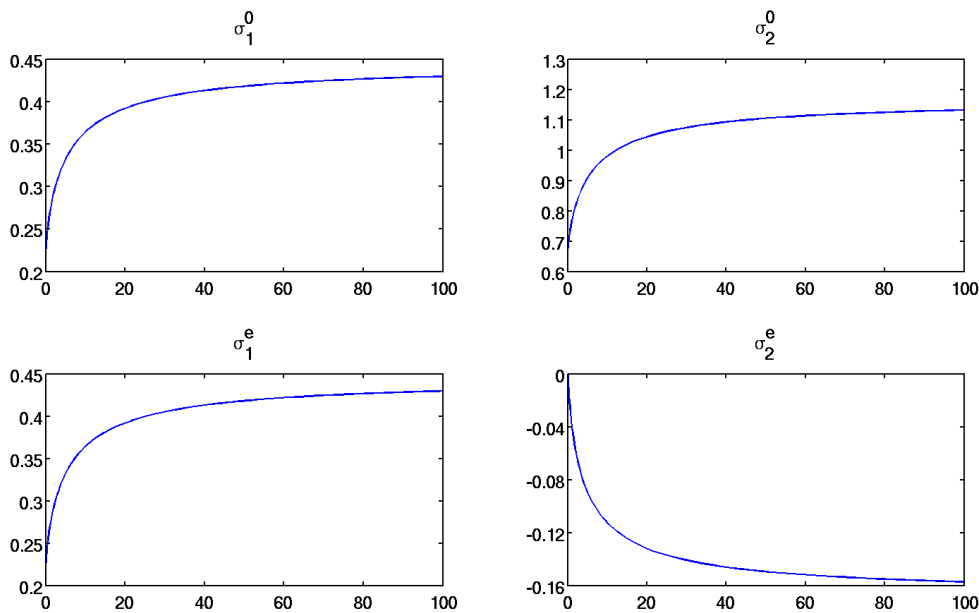


Abbildung 5.3.1.: $\sigma^0(S, 3)$ und $\sigma^e(S, 3)$ für $S \in [0, 100]$

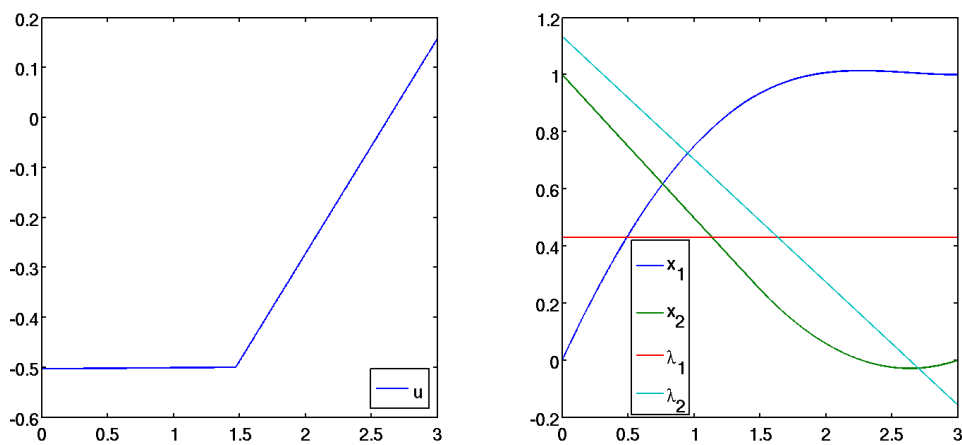
Wir sehen, dass sich die Werte für $S \geq 90$ nicht mehr signifikant verändern und wir somit vermuten, dass die Funktionswerte für $S \rightarrow \infty$ konvergieren.

| | relativer Fehler |
|----------------------|-----------------------|
| $\sigma_1^0(100, 3)$ | $1.640 \cdot 10^{-3}$ |
| $\sigma_2^0(100, 3)$ | $1.525 \cdot 10^{-3}$ |
| $\sigma_1^e(100, 3)$ | $1.640 \cdot 10^{-3}$ |
| $\sigma_2^e(100, 3)$ | $2.467 \cdot 10^{-3}$ |

Tabelle 5.3.1.: Relativer Fehler für $S = 100$

Die Anfangs- und Endwerte werden mit einer akzeptablen Genauigkeit berechnet und sind somit als Startwerte für das Mehrfachschießverfahren geeignet.

Als Nächstes vergleichen wir die Lösung des unbeschränkten Problems (5.3.2) für $S = 100$ mit der Lösung des beschränkten Problems (5.3.1). Zuerst betrachten wir die numerische Lösung mittels indirektem Verfahren.

Abbildung 5.3.2.: Lösung des unbeschränkten Problems (5.3.2) für $S = 100$

Wir erkennen, dass die ursprüngliche Beschränkung der Steuerung für $t \in [0, 1.5]$ aktiv ist. Die Funktionswerte der Steuerung sind dabei für $T = 0$ am kleinsten (rund -0.503). Somit wird die Beschränkung gut eingehalten.

Die numerische Lösung des beschränkten Optimalsteuerungsproblem lautet:

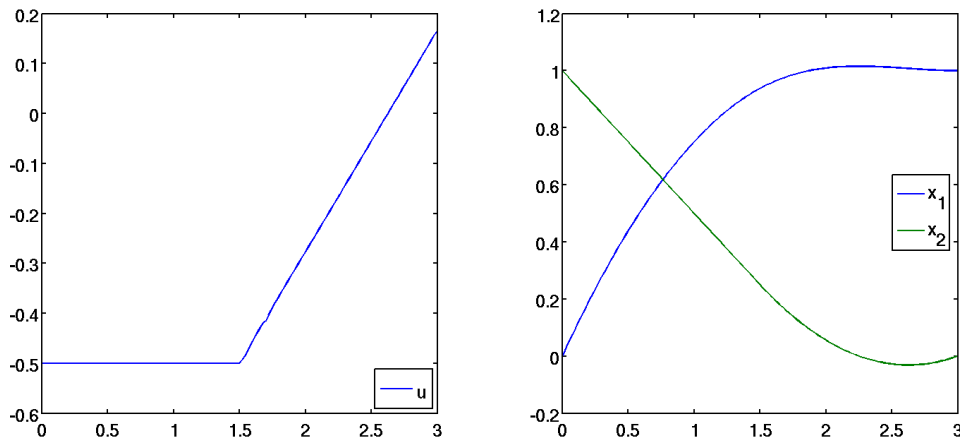


Abbildung 5.3.3.: Lösung des beschränkten Problems (5.3.1)

Wie man deutlich sieht, sind die beiden Lösung sowohl für die Steuerung als auch für die Zustände nahezu identisch. Dies zeigt, dass unser Ansatz auch für beschränkte Optimalsteuerungsprobleme benutzt werden kann.

Wir untersuchen nun, welchen Einfluss S auf die Approximation des beschränkten Problems durch ein unbeschränktes Problem hat. Dazu vergleichen wir die optimale Steuerung von Problem (5.3.1) mit der optimalen Steuerung vom unbeschränkten Problem (5.3.2) für verschiedene S in der L_2 -Norm, wobei wir hierfür das Mehrfachschießverfahren benutzen. Dabei ist im folgenden Bild zu beachten, dass beide Achsen logarithmisch skaliert sind.

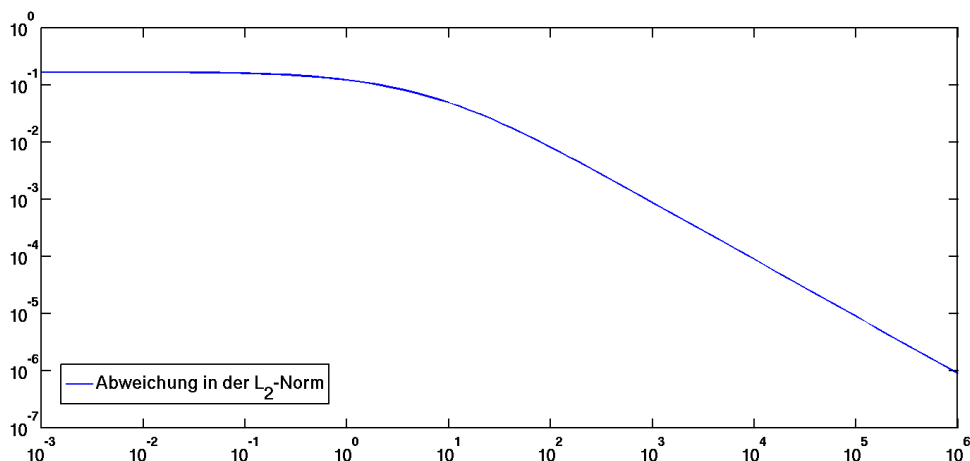


Abbildung 5.3.4.: Abweichung der Steuerungen in der L_2 -Norm für $S \in [10^{-3}, 10^6]$

Wir sehen, dass die optimalen Steuerungen für $S < 1$ noch stark voneinander differieren. Für größere S nähert sich die optimale Steuerung von (5.3.2) der optimalen Steuerung von (5.3.1) immer weiter an. Ferner scheint der Fehler für $S > 10$ linear in Abhängigkeit von $1/s$ zu verschwinden.

5.3.2. Biegung einer Stange

In diesem Beispiel soll die Krümmung einer biegsamen Stange untersucht werden. Diese ist in den Punkten $(0, 0)$ und $(0, 1)$ fest eingespannt, wobei der Einspannwinkel jeweils 45° beträgt. Es handelt sich dabei um das sogenannte „Minimum-Energy-Problem“, wobei wir die Bieghöhe x_1 beschränken:

$$\begin{aligned} \min_u J(u) &= \min_u \int_0^1 \frac{u^2(t)}{\left(\sqrt{1+x_2^2(t)}\right)^5} dt \quad x_1(\cdot) \leq 0.1 \\ \dot{x}(t) &= \begin{pmatrix} x_2(t) \\ u(t) \end{pmatrix} \\ x(0) &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} = x_0 \\ x(1) &= \begin{pmatrix} 0 \\ -1 \end{pmatrix} = x_e. \end{aligned} \tag{5.3.3}$$

Die Herleitung für dieses Optimalsteuerungsproblem kann in [8, S. 14-16] nachvollzogen werden. Wir benutzen den Ansatz aus dem vorigen Beispiel und approximieren das beschränkte Problem (5.3.3) mit dem unbeschränkten Problem

$$\begin{aligned} \min_u J(u) &= \min_u \int_0^1 \frac{u^2(t)}{\left(\sqrt{1+x_2^2(t)}\right)^5} + S\mathcal{P}(x_1(t), 0.1) dt \quad S \geq 0 \\ \dot{x}(t) &= \begin{pmatrix} x_2(t) \\ u(t) \end{pmatrix} \\ x(0) &= x_0 \\ x(1) &= x_e \end{aligned} \tag{5.3.4}$$

mit der Straffunktion

$$\mathcal{P}(v, c) = \begin{cases} 0 & v \leq c \\ (v - c)^2 & \text{sonst} \end{cases}$$

$$\mathcal{P}_v(v, c) = \begin{cases} 0 & v \leq c \\ 2(v - c) & \text{sonst} \end{cases} .$$

Wir stellen die zugehörige Hamilton-Funktion auf und erhalten durch Anwenden der notwendigen Bedingungen aus Satz 2.4.7 die optimale Steuerung

$$\hat{u}(x, \lambda) = -\frac{\lambda_2 \left(\sqrt{1+x_2^2} \right)^5}{2}.$$

Im Gegensatz zum vorigen Beispiel hängt die Steuerung hier nicht explizit von S ab. Trotzdem taucht der Parameter in den Adjungiertengleichungen auf.

$$\begin{aligned} \dot{x} &= \begin{pmatrix} x_2 \\ \hat{u}(x, \lambda) \end{pmatrix} & x(0) &= x_0, \quad x(1) = x_e \\ \dot{\lambda} &= \begin{pmatrix} -S \mathcal{P}_{x_1}(x_1, 0.1) \\ \frac{5\hat{u}^2 x_2}{(\sqrt{1+x_2^2})^7} - \lambda_1 \end{pmatrix}. \end{aligned} \quad (5.3.5)$$

Analog zu Beispiel 5.3.1 ergibt sich ein lineares System in σ_S^0 und σ_S^e , d.h.

$$\begin{aligned} 0 &= \phi_{1,S}(S, T, x_0, \sigma^0) + V_2 \sigma_S^0 \\ \sigma_S^e &= \phi_{2,S}(S, T, x_0, \sigma^0) + V_4 \sigma_S^0. \end{aligned}$$

Die Funktionswerte von σ^0 und σ^e für $S = 0$, d.h. es gibt keine Beschränkung an x_1 , ermitteln wir über das Mehrfachschießverfahren. Wir erhalten die folgende Lösung:

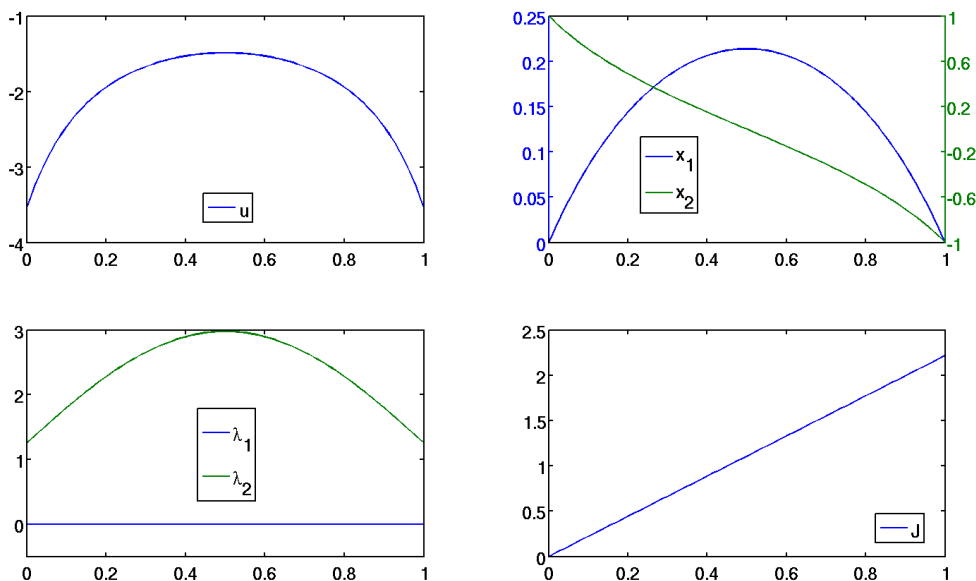
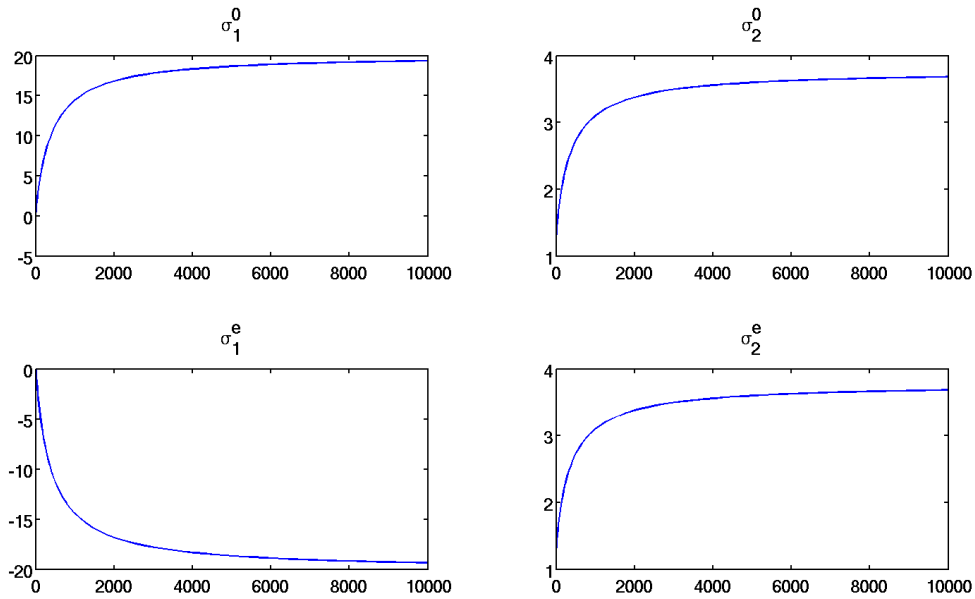


Abbildung 5.3.5.: Lösung von (5.3.4) für $S = 0$

Wir sehen am Verlauf der Zielfunktion, dass die Krümmung nahezu konstant ist. Weiterhin biegt sich die Stange bei $x_1 = 0.5$ auf eine Höhe von knapp 0.214. Somit sollte die Straffunktion für $S > 0$ in diesem Bereich aktiv werden. Wir schauen uns zunächst die numerische Lösung von σ^0 und σ^e mittels unseres Homotopie-Ansatzes an:

Abbildung 5.3.6.: $\sigma^0(S, 1)$ und $\sigma^e(S, 1)$ für $S \in [0, 10000]$

Im Gegensatz zum vorigen Beispiel müssen wir S deutlich größer wählen, um eine Konvergenz der Funktionswerte festzustellen. Bevor wir untersuchen, ob durch die Straffunktion die Beschränkung an x_1 eingehalten wird, vergleichen wir die Ergebnisse unseres Ansatzes für $S = 10^4$ mit den Werten aus dem Mehrfachschießverfahren.

| | relativer Fehler |
|------------------------|-----------------------|
| $\sigma_1^0(10000, 1)$ | $2.295 \cdot 10^{-5}$ |
| $\sigma_2^0(10000, 1)$ | $1.463 \cdot 10^{-5}$ |
| $\sigma_1^e(10000, 1)$ | $8.286 \cdot 10^{-4}$ |
| $\sigma_2^e(10000, 1)$ | $2.962 \cdot 10^{-4}$ |

Tabelle 5.3.2.: Relativer Fehler für $S = 10^4$

Der relative Fehler ist kleiner als im ersten, beschränkten Beispiel und somit sind unsere Ergebnisse gut genug, um sie als Startwerte für ein indirektes Verfahren zu benutzen.

Wir benötigen noch die Lösung des beschränkten Problems (5.3.3) um Aussagen treffen zu können, wie gut das unbeschränkte Problem (5.3.4) die ursprüngliche Aufgabe approximiert.

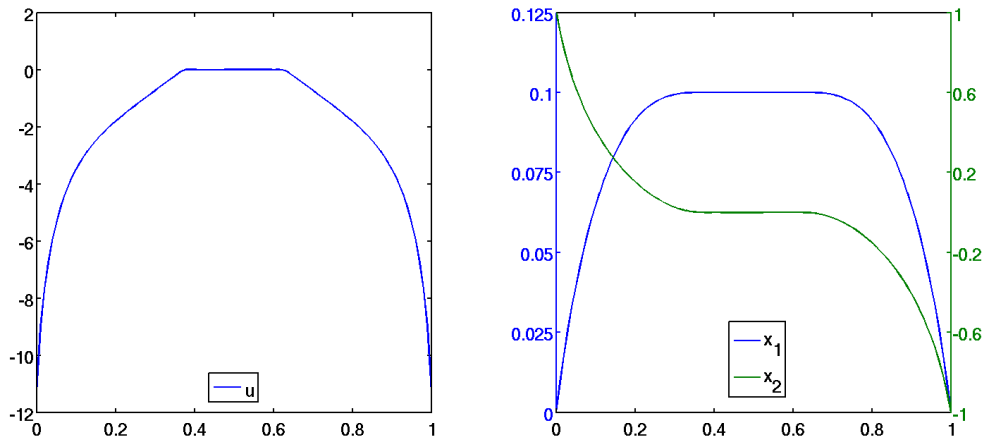
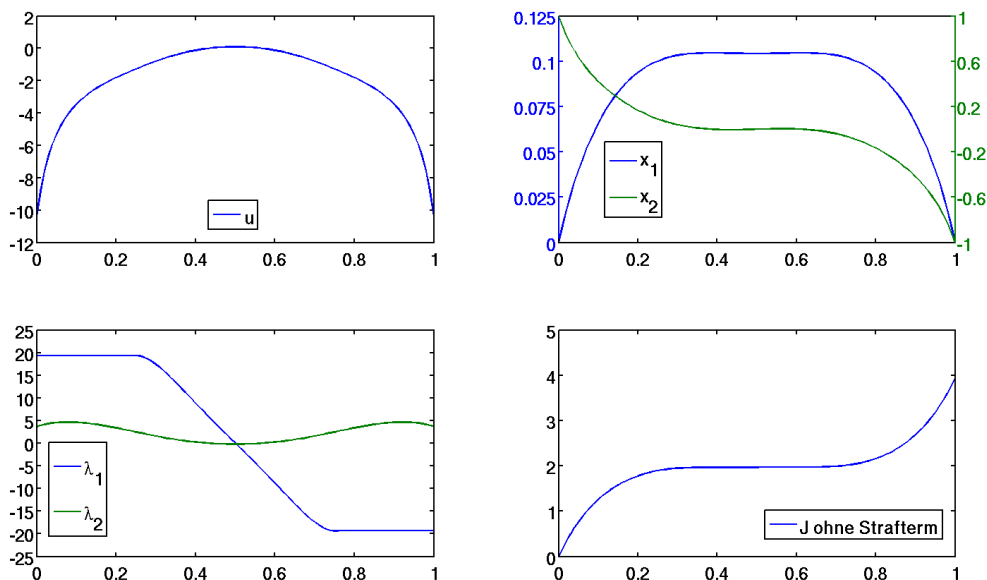


Abbildung 5.3.7.: Lösung von (5.3.3) mittels direktem Verfahren

Diese numerische Lösung vergleichen wir mit den Ergebnissen des Mehrfachschießverfahrens für $S = 10000$.

Abbildung 5.3.8.: Lösung von (5.3.4) für $S = 10000$

Wir erkennen, dass die Beschränkung für x_1 gut eingehalten wird, wobei sie maximal um etwa 0.0046 überschritten wird. Dabei können wir an λ_1 erkennen, wann die Straffunktion aktiv ist, d.h. in diesem Fall für $t \in [0.25, 0.75]$. Wird die Restriktion eingehalten, so ist die Straffunktion konstant Null und dementsprechend ist λ_1 konstant. Ist die Straffunktion aktiv, so ist auch deren Ableitung ungleich Null und der Anstieg von $\lambda_1 < 0$. Dies folgt aus den Adjungiertengleichungen (5.3.5).

Ferner können wir anhand der Steuerung erkennen, dass S noch größer gewählt werden muss, da sie im Vergleich zur optimalen Steuerung vom beschränkten Problem zu glatt ist, insbesondere für $t \in [0.38, 0.62]$. Auch die Werte des Zielfunktional ohne Strafterm im Integranden sind ein Indiz dafür. Der Wert beträgt für das unbeschränkte Problem 3.9407, während das beschränkte Problem einen Zielfunktionalwert von 4.1086 liefert.

Als Nächstes untersuchen wir, wie S die maximale relative Überschreitung der Beschränkung an x_1 beeinflusst. Für $S = 10000$ wissen wir, dass die Beschränkung um etwa 4.6 % überschritten wird und wir deshalb das S für eine sehr gute Approximation noch größer wählen müssen. Dabei ist in der folgenden Abbildung zu beachten, dass die Skalierung beider Achsen logarithmisch gewählt wurde.

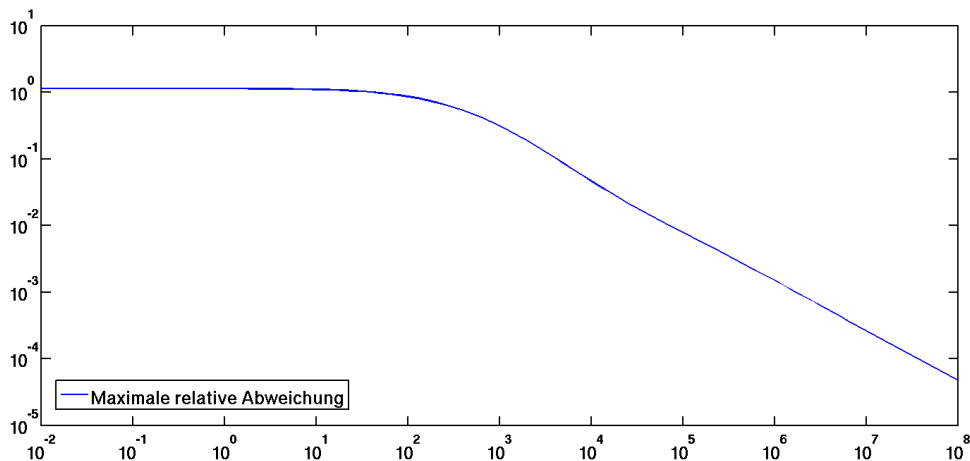
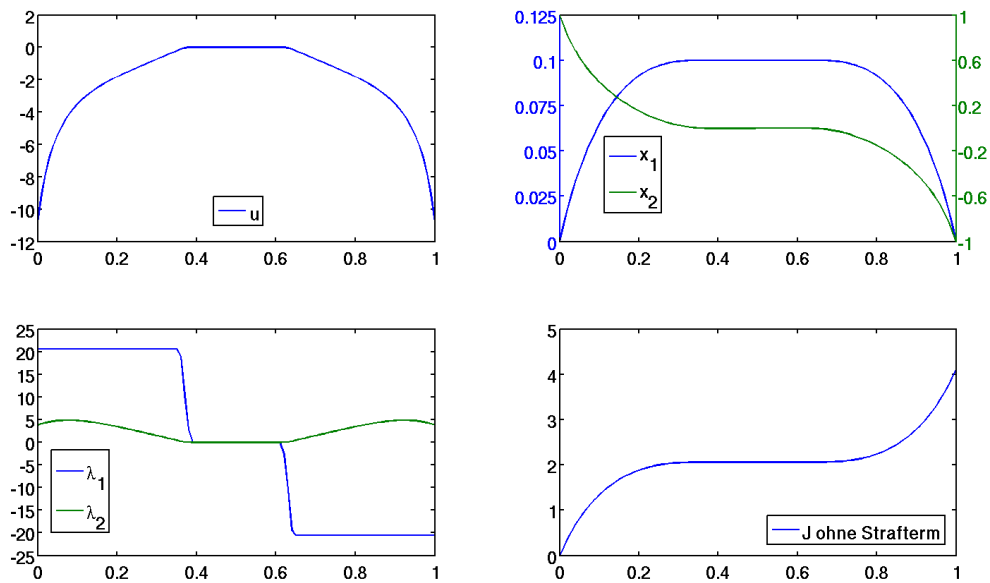


Abbildung 5.3.9.: Maximale relative Abweichung für $S \in [10^{-2}, 10^8]$

Wir sehen, dass sich der relative Fehler erst für $S > 100$ wesentlich ändert. Insbesondere ist wie im vorigen Beispiel eine lineare Abhängigkeit zwischen $1/s$ und der maximalen relativen Abweichung für hinreichend große S festzustellen. Für $S > 2 \cdot 10^6$ ist der relative Fehler kleiner als 0.001 und somit für eine gute Approximation geeignet. Dementsprechend ist unsere ursprüngliche Wahl $S = 10000$ zu konservativ, was wir auch anhand der Steuerung festgestellt haben.

Abschließend schauen wir uns die Ergebnisse des indirekten Verfahrens an, wenn wir S sehr groß wählen, z.B. $S = 10^8$:

Abbildung 5.3.10.: Lösung von (5.3.4) für $S = 10^8$

Die Steuerung entspricht jetzt im Wesentlichen der optimalen Steuerung aus dem beschränkten Problem (5.3.4), wobei sich der Verlauf von x_1 und x_2 im Vergleich zur Lösung für $S = 10000$ nicht verändert hat. λ_1 , welches man wie vorher diskutiert als Schaltverlauf für die Straffunktion \mathcal{P} betrachten kann, hat hier ein anderes Verhalten als bei $S = 10000$. Die Straffunktion war vorher für $t \in [0.25, 0.75]$ aktiv, während sie hier ausschließlich für $t \in [0.35, 0.40] \cup [0.60, 0.65]$ nichtverschwindende Werte annimmt.

6. Zusammenfassung und Ausblick

6.1. Zusammenfassung

In der Dissertation haben wir uns mit dem numerischen Lösen von unbeschränkten Optimalsteuerungsproblemen beschäftigt, wobei die zugehörige Zustandsgleichung eine gewöhnliche Differentialgleichung ist. Als Basis dafür diente der Homotopie-Ansatz von Vicente Costanza [5, 6], welcher die notwendigen Bedingungen für ein lokales Extremum ausnutzte. Ziel der Arbeit war es, diesen Ansatz zu erweitern und auf komplexe Beispiele, wie z.B. das Apollo-Problem [26, S. 491-497], anzuwenden.

Dazu haben wir in Kapitel 2 Optimalsteuerungsprobleme definiert und Resultate aus der Funktionalanalysis zitiert, die wir benötigen, um notwendige Bedingungen für ein unbeschränktes Optimalsteuerungsproblem herzuleiten. Die zentrale Idee dabei ist, dass wir ein äquivalentes, infinites Optimierungsproblem aufstellen und für dieses die notwendigen Bedingungen herleiten und beweisen. Die erhaltenen Resultate haben wir dann auf unbeschränkte Optimalsteuerungsprobleme übertragen.

Dies führte auf die sogenannten Adjungiertengleichungen und die Transversalitätsbedingungen, die zusammen mit der Zustandsgleichung und den gegebenen Randbedingungen ein Randwertproblem für die optimalen Zustände und Adjungierten bilden.

In Kapitel 3 haben wir uns mit Costanzas Ansatz beschäftigt. Dabei stellte sich im Verlauf der Arbeit heraus, dass dieser nicht allgemeingültig ist und nur auf spezielle Fälle angewendet werden kann. Wir haben den ursprünglichen Ansatz neu hergeleitet und an den kritischen Stellen, die in Abschnitt 3.4 untersucht wurden, angepasst, so dass dieser beispielunabhängig benutzt werden kann.

Die Idee dabei ist, dass wir eine Schar von Optimalsteuerungsproblemen betrachten, die von der Endzeit T und einem Parameter S abhängt, der in den Randbedingungen oder der Zielfunktion auftauchen kann. Dementsprechend hängen die Anfangs- und Endwerte der Zustände und der Adjungierten von diesen Parametern ab. Betrachten wir die Anfangs- und Endwerte als Funktionen, die von S und T abhängen, so können wir durch Differenzieren der Flussfunktion ein System von gewöhnlichen Differentialgleichungen aufstellen, mit denen diese Funktionen bestimmt werden können. Mit Hilfe der berechneten Funktionswerte brauchen wir anstatt des ursprünglichen Randwertproblems nur ein Anfangswertproblem lösen, welches numerisch einfacher zu handhaben ist.

In Abschnitt 3.3 haben wir den Ansatz so erweitert, dass der Parameter S auch im Integranden oder der Zustandsgleichung auftauchen darf. Dies kann genutzt werden, um Optimalsteuerungsprobleme mit Beschränkungen zu approximieren.

Dabei entsteht ein weiteres System von gewöhnlichen Differentialgleichungen, welches man simultan mit den anderen Systemen lösen kann. Diese Erweiterung taucht in den Arbeiten von Costanza nicht auf.

Das folgende Kapitel 4 behandelt die numerische Umsetzung unseres neuen Ansatzes. Zum Lösen der gewöhnlichen Differentialgleichungssysteme mit gegebenen Anfangswerten benutzen wir ein in MATLAB implementiertes, explizites Runge-Kutta-Verfahren mit Schrittweitensteuerung. Ein wichtiger Punkt dabei war die Approximation der Jacobi-Matrix der Zustands- und Adjungiertengleichungen mit Hilfe von Complex step differentiation. Deren Eigenschaften und Vorteile haben wir in Abschnitt 4.1.2 untersucht. Weiterhin haben wir direkte und indirekte Verfahren genannt, die man zum Lösen von Optimalsteuerungsproblemen benutzen kann, um die Genauigkeit unseres Ansatzes zu überprüfen.

In Kapitel 5 haben wir unseren Ansatz an verschiedenen Beispielen getestet. Dabei haben wir zuerst unbeschränkte Optimalsteuerungsprobleme betrachtet, die alle sehr gut gelöst wurden.

Als zentrales, komplexes Beispiel haben wir das Apollo-Problem gewählt, dessen numerische Lösung effizient und mit hoher Genauigkeit berechnet wurde. Wir haben dieses Beispiel auch benutzt, um verschiedene Anfangswertlöser und Möglichkeiten zur Approximation der Jacobi-Matrix zu vergleichen. Dabei stellte sich heraus, dass die in Abschnitt 4.1 behandelten Verfahren die besten Ergebnisse liefern.

In Abschnitt 5.3 haben wir Beispiele für beschränkte Optimalsteuerungsprobleme betrachtet. Diese haben wir mit unbeschränkten Optimalsteuerungsproblemen approximiert, wobei wir in dem Integranden eine Straffunktion eingeführt haben, die mit dem Parameter S gewichtet wurde. Somit konnten wir unter Anwendung unseres erweiterten Ansatzes die ursprünglichen Probleme gut approximieren und für hinreichend große S waren die Lösungen der unbeschränkten und beschränkten Probleme im numerischen Sinne identisch. Dabei unterschied sich in den Beispielen, wie groß das S gewählt werden muss, um eine gute Näherung zu erhalten.

6.2. Ausblick

Unser Homotopie-Ansatz funktioniert auch für komplizierte Aufgaben wie z.B. dem Apollo-Problem, welches wir in Kapitel 5 untersucht haben. Allerdings setzt er auch eine hohe Glattheit der Flussfunktion voraus. So muss sie bzgl. der Anfangswerte stetig partiell differenzierbar sein und für den Ansatz aus Abschnitt 3.3 benötigen wir sogar zweimal stetige partielle Differenzierbarkeit bzgl. S und t . Mit Hilfe von [8, S. 144-147] lässt sich zumindest die stetige bzw. Lipschitz-stetige Abhängigkeit der Flussfunktion von S und den Anfangswerten zeigen. Auch können wir auf Grund von Definition 3.1.5 voraussetzen, dass der Mayer-Term φ und die Randbedingungen ψ bzgl. der Anfangs- und Endwerte zweimal stetig differenzierbar und ferner wegen Abschnitt 3.2 auch einmal stetig differenzierbar bzgl. S sind. Es sollte aber weiter untersucht werden, welche Voraussetzungen wir an das Optimalsteuerungsproblem stellen müssen, damit die erforderlichen Bedingungen an die Flussfunktion erfüllt sind.

Weiterhin haben wir noch nicht geklärt, unter welchen Voraussetzungen das lineare, implizite System (4.1.1) als lineares Differentialgleichungssystem aufgefasst werden kann, d.h. wann die Matrixfunktion invertierbar ist. Unter Nutzung von [14, S. 219-228] wissen wir, dass die enthaltene Funktion V symplektisch und somit auch invertierbar ist. Auch sieht man sofort aus der Struktur der Matrixfunktion, dass $D_{x_0, x_e} \psi$ vollen Rang besitzen muss. Da ψ und φ ebenfalls in der Jacobi-Matrix von \mathcal{K} bzgl. der Anfangs- und Endwerte auftauchen, lässt sich abschließend sagen, dass diese wesentlich für die Invertierbarkeit sind.

Costanzas Ansatz nutzte aus, dass V symplektisch ist. Diese Eigenschaft haben wir in unserem Ansatz nicht verwendet. Es kann aber sinnvoll sein, dies zu benutzen, um die Inverse der Matrixfunktion im reduzierten System der Form (4.1.7) direkt zu berechnen, anstatt ein lineares Gleichungssystem in jedem Zeitschritt zu lösen, da man die Inverse einer symplektischen Matrix explizit formulieren kann. Dies setzt natürlich voraus, dass das Beispiel hinreichend einfach ist. Insbesondere bei Gleichungssystemen großer Dimension können wir dadurch Rechenzeit sparen. Auch die parallele Berechnung von \mathcal{A} aus Abschnitt 4.1.2 kann für hochdimensionale Probleme sinnvoll sein.

Gleichzeitig hat unsere allgemeinere Formulierung den Vorteil, dass sie nicht nur auf Randwertprobleme, die aus den notwendigen Bedingungen von Optimalsteuerungsproblemen entstehen, angewendet werden kann, sondern anscheinend auch auf andere Randwertprobleme.

Wir betrachten dazu beispielhaft das folgende Problem aus [28]:

$$\dot{x}(t) = x_5(t) \begin{pmatrix} x_2(t) \\ -\frac{D}{m}v(t)x_2(t) \\ x_4(t) \\ -g - \frac{D}{m}v(t)x_4(t) \\ 0 \end{pmatrix} = f(x(t)) \quad t \in [0, 1] \quad (6.2.1)$$

$$v(t) = \sqrt{x_2^2(t) + x_4^2(t)}$$

unter

$$\begin{aligned} x_1(0) &= 0 \\ x_2(0) &= v_0 \cos \alpha & \alpha &\in \left[0, \frac{\pi}{2}\right] \\ x_3(0) &= 0.9 \\ x_4(0) &= v_0 \sin \alpha \\ x_3(1) &= 0. \end{aligned}$$

Die folgende Tabelle erklärt die verwendeten Konstanten und Variablen:

| Name | Wert | Einheit | Bedeutung |
|----------|-----------------|-------------------|--|
| D | $\rho CA/2$ | kg/m | Luftwiderstand |
| p | 1.2 | kg/m ³ | Luftdichte |
| C | 0.5 | - | Luftwiderstandskoeffizient |
| A | $\pi(0.0366)^2$ | m ² | Fläche |
| m | 0.145 | kg | Masse des Baseballs |
| v_0 | 50 | m/s | Anfangsgeschwindigkeit |
| α | - | rad | Schlagwinkel bzgl. x_1 -Achse |
| g | 9.81 | m/s ² | Erdgravitation |
| v | - | m/s | Absolute Geschwindigkeit |
| x_1 | - | m | Weite |
| x_2 | - | m/s | Geschwindigkeit bzgl. x_1 -Achse |
| x_3 | - | m | Höhe |
| x_4 | - | m/s | Geschwindigkeit bzgl. x_3 -Achse |
| x_5 | - | s | Zeit, die der Baseball in der Luft ist |

Tabelle 6.2.1.: Konstanten und Variablen des Baseball-Problems

Die Schar von Randwertproblemen modelliert die Flugbahn eines Baseballs, welcher auf einer Höhe von 90 cm getroffen wird und dabei eine Anfangsgeschwindigkeit von 50 m/s bekommt, wobei zusätzlich der Luftwiderstand berücksichtigt wird. Die Randbedingungen erzwingen, dass der Ball für $t = 1$ den Boden berührt. Dabei ist der Winkel α , mit dem der Ball getroffen wird, variabel.

Wir können unseren Ansatz auf dieses Problem anwenden, wobei wir dazu einige Definitionen aus Kapitel 3 anpassen müssen.

Definition 6.2.1. Die Funktion $\phi : \mathbb{R} \times [0, T] \times \mathcal{O}_x \rightarrow \mathbb{R}^n$, welche das Anfangswertproblem

$$\begin{aligned} \frac{\partial}{\partial t} \phi(S, t, x_0) &= f(\phi(S, t, x_0)) & \forall t \in [0, T] \\ &= \mathcal{X}(\phi(S, t, x_0)) \\ \phi(S, 0, x_0) &= x_0 & \forall x_0 \in \mathcal{O}_x \end{aligned}$$

löst, heißt *Flussfunktion*. Dabei sei $\mathcal{O}_x \subseteq \mathbb{R}^n$ die Menge aller zulässigen Anfangswerte x_0 .

Definition 6.2.2. Die Funktion $V : \mathbb{R} \times [0, T] \times \mathcal{O}_x \rightarrow \mathbb{R}^{n \times n}$ ist definiert durch

$$V(S, t, x_0) := D_{x_0} \phi(S, t, x_0).$$

Wir setzen $S := \alpha$ und für unser Beispiel (6.2.1) gilt

$$\phi(S, 0, \rho^0(S, 1)) = \begin{pmatrix} 0 \\ v_0 \cos S \\ 0.9 \\ v_0 \sin S \\ \rho_5^0(S, 1) \end{pmatrix} \quad \phi(S, 1, \rho^0(S, 1)) = \begin{pmatrix} \rho_1^e(S, 1) \\ \rho_2^e(S, 1) \\ 0 \\ \rho_4^e(S, 1) \\ \rho_5^e(S, 1) \end{pmatrix}.$$

Wir gehen wie in Abschnitt 3.2 vor, d.h. wir leiten die Flussfunktion zum Zeitpunkt 1 total nach S ab, und erhalten

$$\begin{pmatrix} \rho_{1,S}^e(S, 1) \\ \rho_{2,S}^e(S, 1) \\ 0 \\ \rho_{4,S}^e(S, 1) \\ \rho_{5,S}^e(S, 1) \end{pmatrix} = V(S, 1, \rho^0(S, 1)) \begin{pmatrix} 0 \\ -v_0 \sin S \\ 0 \\ v_0 \cos S \\ \rho_{5,S}^0(S, 1) \end{pmatrix}.$$

Zusammen mit der Matrixgleichung (3.1.8) für unser angepasstes V und dem Differentialgleichungssystem (6.2.1) können wir die Anfangs- und Endwerte in Abhängigkeit von S bestimmen, wobei wir die Funktionswerte für $S = 0$ numerisch mittels Mehrfachschießverfahren ermitteln. Dabei ist in der folgenden Darstellung zu beachten, dass wir S bzw. α im Gradmaß angeben.

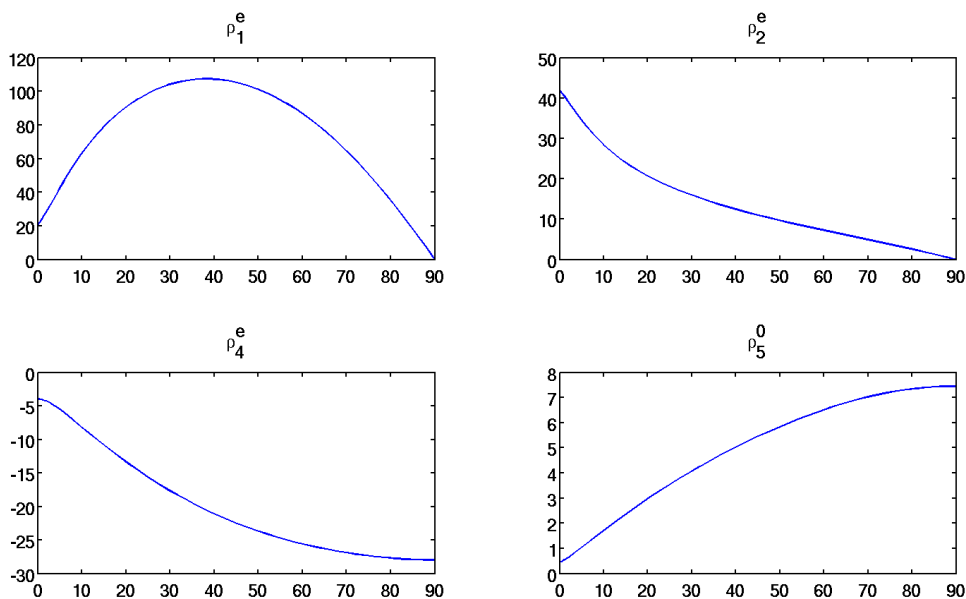


Abbildung 6.2.1.: $\rho_1^e(S, 1)$, $\rho_2^e(S, 1)$, $\rho_4^e(S, 1)$ und $\rho_5^0(S, 1)$ für $S \in \left[0, 90^\circ \frac{\pi}{180^\circ}\right]$

Wir erkennen, dass der Baseball für $\alpha \approx 38.2^\circ$ am weitesten fliegt – und zwar 107.4 Meter. Wird der Ball so getroffen, dass er senkrecht nach oben fliegt, dann ist er mit 7.5 Sekunden am längsten in der Luft. Diese Ergebnisse konnten mittels *fmincon* bestätigt werden. Wir überprüfen zusätzlich die Güte der Approximation mit dem Mehrfachschießverfahren für $\alpha = 35^\circ$.

| | relativer Fehler |
|------------------|-----------------------|
| $\rho_1^e(S, 1)$ | $6.065 \cdot 10^{-8}$ |
| $\rho_2^e(S, 1)$ | $3.131 \cdot 10^{-6}$ |
| $\rho_4^e(S, 1)$ | $6.747 \cdot 10^{-7}$ |
| $\rho_5^0(S, 1)$ | $7.824 \cdot 10^{-8}$ |
| $\rho_5^e(S, 1)$ | $7.824 \cdot 10^{-8}$ |

Tabelle 6.2.2.: Relativer Fehler für $S = 35^\circ \frac{\pi}{180^\circ}$

Die Anfangs- und Endwerte werden mit einer sehr hohen Genauigkeit berechnet und zeigen, dass unser Ansatz anscheinend auch auf allgemeine Randwertprobleme angewendet werden kann. Insbesondere auch dort, wo die Matrixfunktion V nicht notwendigerweise symplektisch ist.

Bei den beschränkten Beispielen haben wir nur Fälle betrachtet, in denen genau eine Komponente Restriktionen einhalten musste. Wir können unsere Straffunktion auch auf mehrdimensionale Beschränkungen anwenden. Sei dazu das folgende restringierte Optimalsteuerungsproblem gegeben

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^T L(x(t), u(t)) dt + \varphi(x(0), x(T)) \quad T \geq 0 \\ \dot{x}(t) &= f(x(t), u(t)) \quad t \in [0, T] \\ \psi(x(0), x(T)) &= 0_{n_\psi} \\ c(x(t), u(t)) &\leq 0_{n_c}, \quad t \in [0, T] \end{aligned} \quad (6.2.2)$$

wobei die Relationszeichen komponentenweise zu verstehen sind. Das bedeutet insbesondere, dass wir n_c Beschränkungen der Form $c_i(x(t), u(t)) \leq 0$ für $t \in [0, T]$ und $i = 1, \dots, n_c$ gegeben haben. Dann definieren wir uns die zugehörigen n_c Straffunktionen

$$\mathcal{P}_i(x(t), u(t)) := \begin{cases} 0 & c_i(x(t), u(t)) \leq 0 \\ c_i^2(x(t), u(t)) & \text{sonst} \end{cases}.$$

Damit können wir das beschränkte Optimalsteuerungsproblem (6.2.2) durch die folgende, unbeschränkte Variationsaufgabe approximieren:

$$\begin{aligned} \min_u \quad J(u) &= \min_u \int_0^T L(x(t), u(t)) + S \sum_{i=1}^{n_c} \mathcal{P}_i(x(t), u(t)) dt + \varphi(x(0), x(T)) \quad S, T \geq 0 \\ \dot{x}(t) &= f(x(t), u(t)) \quad t \in [0, T] \\ \psi(x(0), x(T)) &= 0_{n_\psi}. \end{aligned}$$

Wir können unseren Ansatz auf dieses Problem anwenden und erhalten für $S > 0$ eine Approximation für die Lösung von (6.2.2). Dabei bleibt aber noch zu zeigen, dass, wie in unseren Beispielen in Abschnitt 5.3, die Lösung des unbeschränkten Problems für $S \rightarrow \infty$ gegen die Lösung des beschränkten Optimalsteuerungsproblems konvergiert.

Der Parameter S im Integranden kann nicht nur zur Näherung von beschränkten Problemen benutzt werden, sondern ist auch sinnvoll, wenn wir unsere Aufgabe variieren möchten. So kann man z.B. beim Apollo-Problem (5.2.1) die Luftdichtekonstante $p_0 =: S$ verändern, um zu untersuchen wie sich die Trajektorie der Kapsel in Abhängigkeit von S verändert.

Eine weitere interessante Frage ist, ob man diesen Ansatz auch auf unbeschränkte Probleme der optimalen Steuerung mit partiellen Differentialgleichungen erweitern kann. Prinzipiell scheint dies möglich, da hier ebenfalls das Minimumprinzip [27, S. 178-180] und Adjungiertengleichungen aufgestellt werden können. Allerdings sind dies dann auch partielle Differentialgleichungen und es ist zumindest bei elliptischen Problemen fraglich, wie der Ansatz dort weitere Randbedingungen berechnen soll. So ist in [27, S. 51-53] ein elliptisches Problem angegeben, wo die Randbedingungen für die Zustände als auch die Adjungierten vorgegeben sind.

Bei instationären Problemen kann unsere Homotopie-Methode sinnvoll sein, da man dort entweder zeitliche Anfangs- oder Endbedingungen gegeben hat [27, S. 125-130] und wir durch den Ansatz die fehlenden End- bzw. Anfangsbedingungen ausrechnen können. Somit hätten wir dann reine Anfangs- oder Endwertprobleme, die numerisch einfacher zu handhaben sind. Dabei ist zu erwarten, dass die Differentialgleichungssysteme für die Anfangs- und Endwerte ebenfalls von partieller Natur sind.

Abbildungsverzeichnis

| | | |
|---------|--|----|
| 5.1.1. | $\sigma^0(S, 100)$ und $\sigma^e(S, 100)$ für $S \in [0, 1]$ | 48 |
| 5.1.2. | Lösung von Problem (5.1.1) bzw. (5.1.2) für $S = 1$ | 48 |
| 5.1.3. | $\rho^0(S, T)$ und $\sigma^e(S, T)$ für $S, T \in [0, 1]$ | 50 |
| 5.1.4. | Lösung von Problem (5.1.3) für $S = T = 1$ | 50 |
| 5.1.5. | Zeile 1: $\rho_1^e(S, T)$, $\rho_2^e(S, T)$ und $\hat{\mu}_3(S, T)$, Zeile 2: $\sigma_1^0(S, T)$, $\sigma_2^0(S, T)$ und $\sigma_2^e(S, T)$ für $(S, T) \in [-5, 5] \times [10^{-6}, 2]$ | 53 |
| 5.1.6. | $\hat{\mu}_3(S, T_0)$ für $S \in [-5, 5]$ und $T_0 = 10^{-6}$ | 53 |
| 5.1.7. | Optimale Zustände und Steuerungen für $T = 2$ und $S \in \{-5, 0, 5\}$ | 54 |
| 5.2.1. | Kenngrößen der Trajektorie aus [26, S. 492] | 56 |
| 5.2.2. | $\rho^e(S, T)$ und $\sigma^0(S, T)$ für $S \in [1, 200]$ und $T \in [0, 200]$ | 59 |
| 5.2.3. | $\rho^e(1, T)$ und $\sigma^0(1, T)$ für $T \in [0, 200]$ | 60 |
| 5.2.4. | $\rho^e(S, 200)$ und $\sigma^0(S, 200)$ für $S \in [1, 200]$ | 61 |
| 5.2.5. | Lösung von Problem (5.2.2) für $S = 200$ und $T = 200$ | 62 |
| 5.2.6. | Lösung des Apollo-Problems (5.2.4) | 63 |
| 5.3.1. | $\sigma^0(S, 3)$ und $\sigma^e(S, 3)$ für $S \in [0, 100]$ | 67 |
| 5.3.2. | Lösung des unbeschränkten Problems (5.3.2) für $S = 100$ | 68 |
| 5.3.3. | Lösung des beschränkten Problems (5.3.1) | 69 |
| 5.3.4. | Abweichung der Steuerungen in der L_2 -Norm für $S \in [10^{-3}, 10^6]$ | 69 |
| 5.3.5. | Lösung von (5.3.4) für $S = 0$ | 71 |
| 5.3.6. | $\sigma^0(S, 1)$ und $\sigma^e(S, 1)$ für $S \in [0, 10000]$ | 72 |
| 5.3.7. | Lösung von (5.3.3) mittels direktem Verfahren | 73 |
| 5.3.8. | Lösung von (5.3.4) für $S = 10000$ | 73 |
| 5.3.9. | Maximale relative Abweichung für $S \in [10^{-2}, 10^8]$ | 74 |
| 5.3.10. | Lösung von (5.3.4) für $S = 10^8$ | 75 |
| 6.2.1. | $\rho_1^e(S, 1)$, $\rho_2^e(S, 1)$, $\rho_4^e(S, 1)$ und $\rho_5^0(S, 1)$ für $S \in [0, 90^\circ \frac{\pi}{180^\circ}]$ | 81 |

Tabellenverzeichnis

| | |
|--|----|
| 5.1.1. Relativer Fehler für $S = 1$ | 47 |
| 5.1.2. Relativer Fehler für $S = T = 1$ | 51 |
| 5.1.3. Relativer bzw. absoluter Fehler für $S \in \{-5, 0, 5\}$ und $T = 2$ | 54 |
| 5.2.1. Konstanten und Variablen des Apollo-Problems | 57 |
| 5.2.2. Relativer Fehler für $S_0 = 1$ und $T = 200$ | 60 |
| 5.2.3. Relativer Fehler für $S = 200$ und $T = 200$ | 61 |
| 5.2.4. Laufzeiten für Lösung des Apollo-Problems | 64 |
| 5.2.5. Vergleich der Laufzeiten bei verschiedener Berechnung von \mathcal{A} | 64 |
| 5.2.6. Vergleich der Laufzeiten von Einschritt- und Mehrschrittverfahren | 65 |
| 5.2.7. Vergleich der relativen Fehler von Einschritt- und Mehrschrittverfahren | 65 |
| 5.3.1. Relativer Fehler für $S = 100$ | 68 |
| 5.3.2. Relativer Fehler für $S = 10^4$ | 72 |
| 6.2.1. Konstanten und Variablen des Baseball-Problems | 79 |
| 6.2.2. Relativer Fehler für $S = 35^\circ \frac{\pi}{180^\circ}$ | 81 |

Literaturverzeichnis

- [1] J.F. Bonnans and A. Shapiro. *A Perturbation Analysis of Optimization Problems*. Springer Series in Operation Research, Springer, New York, 2000.
- [2] R.H. Byrd, J.C. Gilbert, and J. Nocedal. *An Interior Point Algorithm for Large-Scale Nonlinear Programming*, pages 877–900. SIAM Journal on Optimization, Vol. 9, No. 4, 1999.
- [3] M. Chyba, H. Sussmann, H. Maurer, and G. Vossen. *Underwater vehicles: The minimum time problem*, pages 1370–1375. 43rd IEEE Conference on Decision and Control, Vol. 2, 2004.
- [4] T.F. Coleman and Y. Li. *An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds*, pages 418–445. SIAM Journal on Optimization, Vol. 6, 1996.
- [5] V. Costanza. *Finding initial costates in finite-horizon nonlinear-quadratic optimal control problems*, pages 225–242. Journal of Optimization Theory and Applications, Vol. 29, No. 3, John Wiley & Sons, 2008.
- [6] V. Costanza, P.S. Rivadeneira, and R.D. Spies. *Equations for the Missing Boundary Values in the Hamiltonian Formulation of Optimal Control Problems*, pages 26–46. Journal of Optimization Theory and Applications, Vol. 149, No. 1, Springer Science, 2011.
- [7] J.R. Dormand and P.J. Prince. *A family of embedded Runge-Kutta formulae*, pages 19–26. Journal of Computational and Applied Mathematics, Vol. 6, No. 1, 1980.
- [8] M. Gerdts. *Optimale Steuerung*. Vorlesungsskript, WS 2009/2010.
- [9] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, London, 1981.
- [10] A. Griewank and A. Walther. *Evaluation Derivatives: Principles and Techniques of Algorithmic Differentiation*. SIAM, Philadelphia, 2008, 2nd edition.
- [11] H. Hermes and J.P. LaSalle. *Functional Analysis and Time Optimal Control*, page 36. Mathematics in Science and Engineering, Vol. 56, Academic Press, New York, 1969.
- [12] M.R. Hestenes. *Calculus of variations and optimal control theory*. John Wiley & Sons, New York, 1966.

- [13] S. Hübbers. *Johann Bernoulli und das Brachistochronenproblem*. Zulassungsarbeit für das Lehramt an Realschulen, Universität Bayreuth, 1998.
- [14] A. Katok and B. Hasselblatt. *Introduction to the Modern Theory of Dynamical Systems*. Cambridge University Press, 1999, 2nd edition.
- [15] J. Kierzenka and L.F. Shampine. *A BVP Solver based on Residual Control and the MATLAB PSE*, pages 299–316. ACM TOMS, Vol. 27, No. 3, 2001.
- [16] S. Kurcyusz. *On the Existence and Nonexistence of Lagrange Multipliers in Banach Spaces*. Journal of Optimization Theory and Applications, Vol. 20, Nr. 1, 1976.
- [17] F. Lempio. *Tangentialmannigfaltigkeiten und infinite Optimierung*. Habilitationsschrift, Universität Hamburg, 1972.
- [18] L.A. Ljusternik and W.I. Sobolew. *Elemente der Funktionalanalysis*. Verlag Harri Deutsch, Zürich-Frankfurt/Main-Thun, 1976.
- [19] J.L. Lunkins and J.D. Turner. *Optimal spacecraft rotational maneuvers*. Elsevier, 1986.
- [20] J.N. Lyness and C.B. Moler. *Numerical Differentiation of Analytic Functions*, pages 202–210. SIAM Journal on Numerical Analysis, Vol. 4, No. 2, SIAM, 1967.
- [21] H.S. Nik, S. Effati, S.S. Motsa, and M. Shirazian. *Spectral homotopy analysis method and its convergence for solving a class of nonlinear optimal control problems*, pages 171–194. Numerical Algorithms, Vol. 65, No. 1, 2014.
- [22] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer Series in Operations Research, Springer Verlag, 2006, 2nd edition.
- [23] L.S. Pontryagin, V. Boltyanskij, R. Gamkrelidze, and E. Mishchenko. *Mathematische Theorie optimaler Prozesse*. Oldenbourg, München, 1964.
- [24] L.F. Shampine and M.K. Gordon. *Computer Solution of Ordinary Differential Equations: the Initial Value Problem*. W.H. Freeman, SanFrancisco, 1975.
- [25] W. Squire and G. Trapp. *Using complex variables to estimate derivatives of real functions*, pages 110–112. SIAM Review, Vol. 40, No. 1, SIAM, 1998.
- [26] J. Stoer and R. Burlisch. *Introduction to Numerical Analysis*. Springer-Verlag New York Inc., 1980.
- [27] F. Tröltzsch. *Optimale Steuerung partieller Differentialgleichungen: Theorie, Verfahren and Anwendungen*. Vieweg+Teubner, 2009, 2. Auflage.
- [28] A. Van Heuvelen and P. D’Alessandris. *Projectile Motion with Air Resistance*. Addison Wesley Publishing,
<http://wps.aw.com/wps/media/objects/877/898586/topics/topic01.pdf>,
Letzter Aufruf: 17.03.2015.

- [29] D. Werner. *Funktionalanalysis*. Springer, Berlin-Heidelberg-New York, 1995.
- [30] J. Zowe and S. Kurcyusz. *Regularity and Stability of the Mathematical Programming Problem in Banach Spaces*, pages 49–62. Applied Mathematics and Optimization, Vol. 5, 1979.

A. Veröffentlichungen

C. Lass, *Minimal number of clues for Sudokus*, pages 143-151, Central European Journal of Computer Science, Vol. 2, No. 2, 2012

C. Lass, *On correctness of first and second order fast marching method*, pages 154-166, Central European Journal of Computer Science, Vol. 1, No. 2, 2011

C. Lass, *Untersuchung von numerischen Methoden zur Formerkennung*, Diplomarbeit, Ernst-Moritz-Arndt-Universität Greifswald, 2010

B. Danksagung

Ich bedanke mich bei meinem Betreuer Herrn Professor Kugelmann, der mir dieses Thema angeboten hat und für Fragen als auch Diskussionen offen war. Insbesondere sind dadurch die schönen Ergebnisse aus Abschnitt 3.3 ermöglicht worden.

Ferner bedanke ich mich bei Herrn Dr. Peter Nestler und Herrn Professor Kugelmann für die Übung und Vorlesung zu Numerik I. Mir wurde dadurch bereits in einer frühen Phase meines Studiums klar, dass ich mich in dem Fachgebiet der Numerik spezialisieren werde. Dies führte auch zu meiner Diplomarbeit mit dem Thema „Untersuchung von numerischen Methoden zur Formerkennung“ und letztendlich auch zu meiner Einstellung als wissenschaftlicher Mitarbeiter bei Herrn Professor Kugelmann.

Weiterhin möchte ich mich bei meinem Kollegen Paul Ziemann für die vielen fruchtbaren Diskussionen bedanken. Insbesondere war seine Idee einer quadratischen Straffunktion, wie sie in Abschnitt 5.3 benutzt wird, deutlich einfacher und effektiver als meine ursprüngliche Idee, die den *Cosinus Hyperbolicus* beinhaltete.

Auch die Korrekturleser sollen nicht unerwähnt bleiben. Ich bedanke mich bei Herrn Professor Kugelmann, Paul Ziemann und meinen Eltern. Ich möchte dabei meine Eltern hervorheben, die mir mit ihrer finanziellen Unterstützung das Mathematik-Studium überhaupt erst möglich gemacht haben.