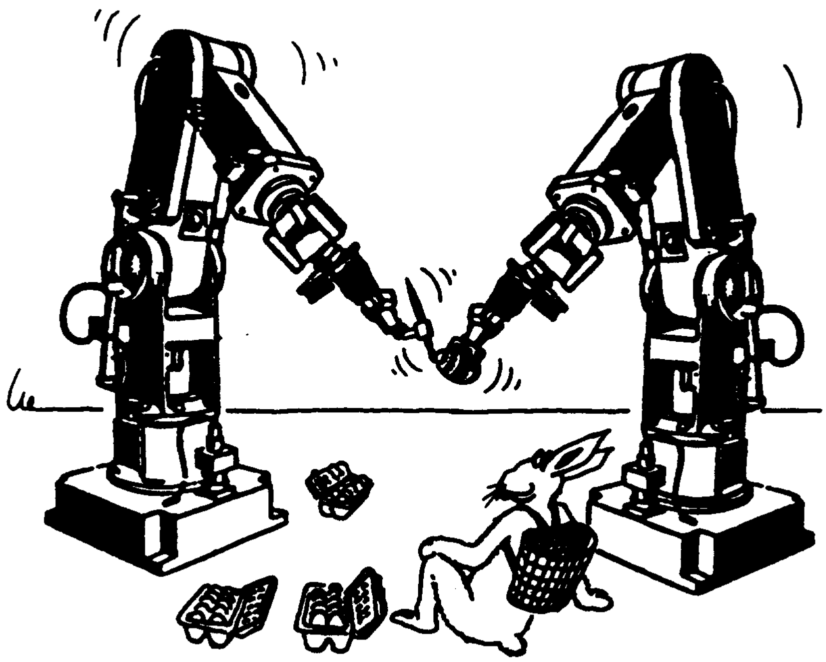
Numerische Berechnung optimaler Steuerungen für Industrieroboter



Diplomarbeit
von
Marc Steinbach

Bonn, im Dezember 1987

– Neufassung: Dezember 1997 –

Inhalt

0. Einleitung

Der Begriff „Roboter“ hat im Lauf der Geschichte verschiedene Bedeutungen erhalten. Er geht zurück auf den slawischen Stamm *robot-* oder *rabot-*, welcher recht gut dem deutschen Stamm *arbeit-* entspricht, im engeren Sinne aber auch Frondienst bedeutet (also die Arbeit der Leibeigenen).

Allgemein geläufig wurde der Begriff Roboter wohl in der ersten Hälfte dieses Jahrhunderts durch die Science-fiction-Literatur. In diesem Zusammenhang bezeichnet er einen Maschinenmenschen, der i.a. mit mehr oder weniger Intelligenz ausgestattet ist. So formulierte denn auch Isaac Asimov um 1950 seine berühmten Roboter-Gesetze, die das Verhalten der Roboter gegenüber Menschen und untereinander regeln sollen. In den Gesetzen kommt deutlich die Auffassung des Roboters als Diener des Menschen zum Ausdruck.

Beim modernen Roboter, dem in der industriellen Fertigung eingesetzten Manipulator, sind nur noch die wichtigsten Teile des Maschinenmenschen vorhanden: Hand, Arm, und „Intelligenz“ im Sinne von Lernvermögen. Auch der Industrieroboter soll dem Menschen gewissermaßen als „Handlanger“ dienen.

Im folgenden wird der Begriff Roboter stets in der letztgenannten Bedeutung benutzt, allerdings oft in einer abstrahierten Form, nämlich als ein spezielles mechanisches Gebilde (Mehrkörpersystem).

Die vorliegende Arbeit befaßt sich mit einem jungen und noch relativ kleinen, aber dennoch wichtigen Zweig der Robotik: mit der *praktischen* Berechnung *optimaler Steuerungen* für Industrieroboter.

In der umfangreichen Literatur zur Robotik, die zum größten Teil in diversen Fachzeitschriften erscheint, findet man hauptsächlich Beiträge aus den Ingenieurwissenschaften. Zum Problemkreis „Steuerung“ sind beispielsweise folgende Themen typisch:

- Planung von Bewegungsabläufen und Realisierung in der Praxis (z.B. digitale Programmsteuerung),
- Sensorgestützte feedback-Regelungen,
- Simulation von Bewegungsabläufen mit Digitalrechnern auf der Grundlage mathematischer Modelle.

Die weitaus größte Zahl der Veröffentlichungen behandelt dabei vorwiegend *technische* Fragen. Im Vordergrund stehen unmittelbar anwendungsorientierte Kriterien, wie z.B. die möglichst einfache Realisierbarkeit eines Bewegungsablaufs auf dem gegebenen Manipulator.

Optimale Steuerungen im mathematischen Sinne wurden dagegen bisher recht selten erforscht. Abgesehen von vereinzelten früheren Arbeiten befassen sich erst seit den letzten fünf bis zehn Jahren einige Autoren mit diesem Problem.

Die mathematisch ausgerichteten Untersuchungen legen meist praxisbezogene Ansätze für eine möglichst gute Steuerung zugrunde, die dann nach einfachen heuristischen Prinzipien optimiert wird. Typisch dafür sind etwa die Arbeiten von Kiriazov/Marinov [18], [26], [19] aus den Jahren 1983/84.

Im Rahmen des von der Deutschen Forschungsgemeinschaft geförderten Sonderforschungsbereich 72 „Approximation und Optimierung“ werden seit 1983 von einer Bonner Arbeitsgruppe optimale Roboter-Steuerungen systematisch erforscht, wobei auch diese Arbeit entstanden ist. Der Optimierung liegt das einfache Prinzip zugrunde, daß gering belastete Motoren über physikalische Wechselwirkungen die stark belasteten Motoren unterstützen können. Dazu ist allerdings ein genaues physikalisches Verständnis der *dynamischen Effekte* (z.B. Fliehkraft, Coriolis-Kraft) erforderlich, wie es in solchem Umfang erstmals von unserer Arbeitsgruppe eingebracht wurde.

Eine Möglichkeit, zu numerischen Ergebnissen zu gelangen, besteht darin, daß man die Steuerungen durch Treppenfunktionen (oder stückweise polynomial) approximiert, so daß sie ohne genaue Kenntnis der optimalen Struktur berechnet werden können (direkter Ansatz, Bock/Plitt [9]). So erhält man auf effiziente Weise relativ gute *suboptimale Lösungen* — je nach Genauigkeit der Approximation.

Schon bei diesem Verfahren ist aber die physikalische Analyse des Problems notwendig, um eine näherungsweise Vorstellung vom optimalen Bewegungsablauf (und damit brauchbare Startdaten für die iterative Lösung) zu erhalten.

Mit dem direkten Ansatz wurden im Laufe der Forschungsarbeiten eine Reihe verschiedenartiger zwei- und dreidimensionaler Bewegungen für unterschiedliche Roboter-Typen berechnet. Ein Teil der Ergebnisse ist in der Diplomarbeit von KONZELMANN [21] veröffentlicht.

Trotz sehr guter, auch praktisch verwertbarer Resultate bleibt es unbefriedigend, daß mit dem direkten Ansatz aufgrund der a priori gewählten Diskretisierung weder singuläre Steuerungen exakt berechenbar sind, noch die notwendigen Bedingungen einer optimalen Steuerung überprüft werden können.
Zu diesem Zweck muß man das Steuerungsproblem nach dem Maximumprinzip in ein assoziiertes Mehrpunktrandwertproblem transformieren, in dem die Steuerung eliminiert ist und ohne vorgewählte Diskretisierung berechnet werden kann (indirekter Ansatz, BOCK [5]).
Dieses Verfahren ist wesentlich aufwendiger und komplizierter, sowohl in Bezug auf die analytische Arbeit als auch auf die Numerik. Bei zeitoptimalen Roboter-Steuerungen, wie sie hier behandelt werden sollen, entstehen durch das Auftreten singulärer Steuerungen zusätzliche Schwierigkeiten. Wie sich herausstellen wird, ergeben sich schon in scheinbar einfachen Fällen unerwartete, außergewöhnlich komplizierte Lösungsstrukturen.
In der vorliegenden Arbeit wurde meines Wissens erstmals eine ausführliche numerische Untersuchung nichttrivialer optimaler Steuerungen für einen Industrieroboter mit dem indirekten Ansatz durchgeführt.

Die Arbeit gliedert sich in drei Kapitel, einen Anhang und eine Zusammenfassung.

Das erste Kapitel stellt die allgemeinen Grundlagen der Theorie der optimalen Steuerung zusammen. Im Anschluß an die abstrakte Definition des allgemeinen Steuerungsproblems wird gezeigt, wie sich die meisten gängigen Zielfunktionale auf das MAYER-Zielfunktional zurückführen lassen.
Es folgt eine kurze Erläuterung des direkten Ansatzes zur Lösung eines Steuerungsproblems. Im Hauptteil des Kapitels wird dargelegt, wie im Rahmen des indirekten Ansatzes die optimale Steuerung aus dem Maximumprinzip ermittelt werden kann. Besondere Berücksichtigung finden dabei die für Roboter wesentlichen linearen und singulären Steuerungen; auch eine kurze Beschreibung der damit verwandten Behandlung von Zustandsbeschränkungen ist vorhanden.
Schließlich wird ein Mehrzielverfahren vorgestellt, das die numerische Lösung des assoziierten Mehrpunktrandwertproblems gestattet, welches beim indirekten Ansatz aus dem Steuerungsproblem ensteht.

Das zweite Kapitel stellt zunächst die mechanische Theorie der Mehrkörpersysteme im LAGRANGE-Formalismus dar, um darauf aufbauend ein allgemeines mathematisches Modell für Roboter zu entwikkeln. Für das durch ein Zielfunktional und Randbedingungen ergänzte Modell wird dann die allgemeine Steuerungstheorie aus dem ersten Kapitel spezialisiert.
Schließlich wird der Polarroboter vorgestellt und mit Hilfe der zuvor erarbeiteten Theorie untersucht. Besonders ausführlich werden die Herleitung des konkreten mathematischen Modells und die Untersuchung der singulären Steuerungen behandelt.

Im dritten Kapitel wird eine Homotopieklasse von Steuerungsproblemen für den Polarroboter formuliert. Das assoziierte Mehrpunktrandwertproblem wird einer einfachen Transformation unterworfen, so daß es numerisch lösbar ist.
Eine mit dem direkten Ansatz gewonnene suboptimale Lösung wird dazu benutzt, eine erste optimale Lösung als Start der Homotopie zu erzeugen. Abschließend werden die einzelnen Lösungen dem Verlauf der Homotopie folgend beschrieben, wobei die strukturellen Daten in Tabellenform und graphisch dargestellt sind.

Im Anhang findet man eine Beschreibung und ein Listing der problemabhängigen Programmteile für die Optimierung.
Schließlich sind dort für ausgewählte Homotopieparameter die kompletten Lösungen graphisch dargestellt, so daß man die Homotopie anschaulich verfolgen kann. Diese Plots beanspruchen gut 40 % der gesamten Seitenzahl der Arbeit und konnten aus Gründen der Übersichtlichkeit nicht in den letzten Abschnitt des dritten Kapitels aufgenommen werden.

Die Zusammenfassung erörtert die wesentlichen Ergebnisse der Arbeit im Hinblick auf ihren theoretischen wie praktischen Nutzen und gibt einen Ausblick auf die zu erwartende weitere Entwicklung von Forschung und Anwendungen.

Die numerischen Rechnungen wurden auf der IBM 4361 des Instituts für angewandte Mathematik sowie auf den Rechnern IBM 3081 und IBM 4381 des regionalen Rechenzentrums der Universität Bonn ausgeführt.

Herrn Dr. H. G. Bock und Herrn Prof. Dr. J. Frehse möchte ich für die Aufgabenstellung und Förderung der Arbeit danken.

Herrn Dr. H. G. Bock und Herrn Dr. J. Schlöder darf ich besonders herzlich danken für ihre vielen wertvollen Anregungen und die unermüdliche Unterstützung während der Arbeit, außerdem Herrn Prof. Dr. R. Longman, der wesentlich an der mathematischen Modellierung verschiedener Roboter beteiligt war, und der durch sehr interessante Gespräche ebenfalls zum Gelingen der vorliegenden Arbeit beitrug.

Schließlich danke ich Herrn Erik Liebermann für die freundliche Genehmigung, seinen Cartoon als Titelbild zu veröffentlichen.

1. Theoretische Grundlagen der optimalen Steuerung

In diesem Kapitel wird zunächst das abstrakte Steuerungsproblem in seiner allgemeinen Form mit Parametern und Beschränkungen vorgestellt. Anschließend werden der sogenannte direkte und der indirekte Lösungsansatz beschrieben. Für den indirekten Ansatz werden Methoden zur Berechnung der optimalen Steuerung aus dem Maximumprinzip dargestellt.

Im Hinblick auf die folgenden Teile der Arbeit wird an geeigneten Stellen auch auf Aspekte der numerischen Realisierung hingewiesen. Schließlich wird ein Mehrzielalgorithmus zur Lösung von Steuerungsproblemen nach dem indirekten Ansatz vorgestellt.

1.1 Das allgemeine abstrakte Steuerungsproblem

Wir betrachten einen dynamischen Prozeß, der zur Zeit t_s startet und zur Zeit t_e endet. Beide Zeitpunkte sind möglicherweise *unbekannt* und sollen im Rahmen der folgenden Aufgabe *optimal* gewählt werden: Gesucht sind eine *Steuerung* u auf dem Intervall $I := [t_s, t_e]$ und ein *Parametervektor* p, so daß die zugehörige Lösung x der *Prozeß*- oder *Zustandsdifferentialgleichung*

$$\dot{x}(t) - f(t, x(t), u(t), p) = 0 \tag{1.1.1}$$

das *Zielfunktional* minimiert:

$$r_1(t_s, x(t_s), t_e, x(t_e), p) = \min \tag{1.1.2}$$

Dabei sind zusätzlich die (klassischen) *Randbedingungen*

$$r_2(t_s, x(t_s), t_e, x(t_e), p) = 0, \tag{1.1.3}$$

die *Ungleichungsbedingungen*

$$r_3(t_s, x(t_s), t_e, x(t_e), p) \geq 0 \tag{1.1.4}$$

und die *Zustandsbeschränkungen*

$$g(t, x(t), u(t), p) \leq 0 \tag{1.1.5}$$

(komponentenweise) zu erfüllen. Das Paar (u, p) heißt *zulässig*, wenn die zugehörige *Zustandsvariable* x die Nebenbedingungen (1.1.3–1.1.5) erfüllt. (u, p) und x heißen *optimal*, wenn außerdem das Zielfunktional (1.1.2) minimiert wird.

In dieser Arbeit werden die folgenden Regularitätsvoraussetzungen und Bezeichnungen benutzt:

$$\begin{aligned}
r_i &\in C^2(\mathbb{R} \times \mathbb{R}^{n_d} \times \mathbb{R} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_p}, \mathbb{R}^{n_i}) \qquad (i \in \{1,2,3\}),\\
f &\in C^2(\mathbb{R} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p}, \mathbb{R}^{n_d}),\\
g &\in C(\mathbb{R} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_p}, \mathbb{R}^{n_g}),\\
u &\in L^\infty(I, U), \quad U \subseteq \mathbb{R}^{n_u},\\
x &\in H^{1,\infty}(I, \mathbb{R}^{n_d}),\\
p &\in \mathbb{R}^{n_p}.
\end{aligned} \tag{1.1.6}$$

Die Minimierungsbedingung (1.1.2) ist dabei stets *einstufig*, d.h. $n_1 = 1$.

Bemerkungen und Erläuterungen:

• Das hier vorgestellte Steuerungsproblem umfaßt im Prinzip alle in der Praxis vorkommenden Fälle, es wird jedoch fast nie in so allgemeiner Form benötigt. Einige häufige Spezialfälle werden im folgenden kurz angesprochen.

• Die Voraussetzung $u \in L^\infty(I, U)$ wird für die theoretische Untersuchung benötigt, z.B. in Existenzbeweisen. Es folgt dann $x \in H^{1,\infty}(I, \mathbb{R}^{n_d})$. Für numerische Zwecke ist dies nicht hinreichend, vielmehr muß gefordert werden, daß *u stückweise stetig* ist. Entsprechend ist die Zustandsvariable x dann stetig und stückweise stetig differenzierbar. In eventuellen Sprungstellen seien alle zeitabhängigen Funktionen o.B.d.A. *linksseitig stetig.*

• Als *Steuerbereich* kommt selbstverständlich nicht jede beliebige Teilmenge $U \subseteq \mathbb{R}^{n_u}$ in Frage. Die wichtigsten in der Praxis auftretenden Fälle werden in Abschnitt (1.3.2) beschrieben; U besitzt dort stets eine sehr spezielle und relativ einfache Struktur.
Hier sei jedoch darauf hingewiesen, daß U nicht notwendig zusammenhängend sein muß. Es lassen sich daher z.B. auch Prozesse mit Gangschaltung behandeln (siehe hierzu BOCK/LONGMAN [7], [8], KRÄMER-EIS [23]).

• In reinen Steuerungsproblemen kommen Parameter fast ausschließlich vor, wenn t_s und/oder t_e unbekannt sind. In diesem Fall muß das Steuerungsproblem für die *numerische* Formulierung auf $[0, 1]$ (oder ein anderes festes Intervall) transformiert werden. Setzt man z.B. $t(\tau) := t_s + (t_e - t_s)\tau$ und $y(\tau) := x(t(\tau))$ mit $\tau \in [0, 1]$, so lautet die neue rechte Seite $\dot{y}(\tau) = \dot{x}(t(\tau))(t_e - t_s)$. Start- und Endzeit des ursprünglichen Problems treten darin explizit auf und müssen als zusätzliche Parameter behandelt werden. „Echte“ freie Parameter spielen aber beim *Entwurf* von Maschinen eine wichtige Rolle (*optimal design*). Es ist z.B. sinnvoll, für eine oft zu wiederholende Bewegung zusätzlich zur eigentlichen Steuerung das Übersetzungsverhältnis eines Getriebes oder die Länge eines Roboterarms zu optimieren.

• Die Randbedingungen (1.1.3) treten oft *separiert* auf als Anfangsbedingungen $x(t_s) - x_s = 0$ und Endbedingungen $x_i(t_e) - x_{ie} = 0$ für gewisse Komponenten x_i (im Startpunkt t_s sind meist *alle* Komponenten vorgeschrieben). Häufig kommen auch Randbedingungen der Form $x_i(t_e) - x_i(t_s) = 0$ vor (periodische Komponenten).

• In der Praxis sind auch Ungleichungsbedingungen der allgemeinen Form (1.1.4) selten. Oft treten sie nur im Endpunkt t_e auf, oder als *Parameterrestriktionen* $p_i \geq 0$.

• Zustandsbeschränkungen kommen in praxisbezogenen Problemen fast immer vor und sind daher besonders wichtig. Sie beschreiben z.B. Mindestanforderungen, Höchstbelastungen, Sicherheitstoleranzen oder „Hindernisse“.

Statt (1.1.2) kann man auch ein allgemeineres Zielfunktional betrachten:

$$\begin{aligned} \gamma_1 L_1(t_s, x(t_s), t_e, x(t_e), p) + \gamma_2 \int_{t_s}^{t_e} L_2(t, x(t), u(t), p)\, dt & \\ + \gamma_3 \max_{t \in I} L_3(t, x(t), u(t), p) \quad &= \min . \end{aligned} \tag{1.1.7}$$

Das sogenannte MAYER-Zielfunktional (1.1.2) erhält man daraus für $\gamma_2 = \gamma_3 = 0$. In den Fällen $\gamma_1 = \gamma_3 = 0$, $\gamma_1 = \gamma_2 = 0$ bzw. $\gamma_3 = 0$ spricht man von LAGRANGE-, TSCHEBYSCHEFF- bzw. BOLZA-Form. Das Steuerungsproblem mit (1.1.7) ist aber nur scheinbar allgemeiner als das mit (1.1.2). Führt man L_2 als Ableitung einer zusätzlichen Zustandsvariablen mit Startwert Null ein, so geht das LAGRANGE-Zielfunktional in eine Endbedingung über. Entsprechend läßt sich das TSCHEBYSCHEFF-Zielfunktional durch Hinzunahme eines weiteren Parameters und einer neuen Zustandsbeschränkung transformieren. So erhält man mit

$$\begin{aligned} x^* &:= (x^\top, x^*_{n_d+1})^\top, \\ p^* &:= (p^\top, p^*_{n_p+1})^\top, \\ f^*(t, x^*, u, p^*) &:= (f(t, x, u, p)^\top, L_2(t, x, u, p))^\top, \\ g^*(t, x^*, u, p^*) &:= (g(t, x, u, p)^\top, L_3(t, x, u, p) - p^*_{n_p+1})^\top, \\ r_1^*(t_s, x_s^*, t_e, x_e^*, p^*) &:= \gamma_1 L_1(t_s, x_s, t_e, x_e, p) + \gamma_2 x^*_{e\, n_d+1} + \gamma_3 p^*_{n_p+1}, \\ r_2^*(t_s, x_s^*, t_e, x_e^*, p^*) &:= (r_2(t_s, x_s, t_e, x_e, p)^\top, x^*_{s\, n_d+1})^\top \end{aligned} \tag{1.1.8}$$

ein (theoretisch) äquivalentes Steuerungsproblem in der Form (1.1.1–1.1.5), denn das Zielfunktional r_1^* ist offensichtlich vom MAYER-Typ. Im folgenden soll stets diese Formulierung verwendet werden, da sie für numerische Zwecke am besten geeignet ist.

Es sei jedoch ausdrücklich darauf hingewiesen, daß die obige Transformation nicht unbedingt eine Vereinfachung des Problems bedeutet. Besonders im TSCHEBYSCHEFF-Fall treten spezifische Schwierigkeiten auf, die auch in MAYER-Formulierung eine spezielle Behandlung erfordern (siehe z.B. BOCK [5]).

1.2 Der direkte Lösungsansatz

Für die Behandlung mit dem *direkten Ansatz* schränkt man die Menge der erlaubten Steuerungen künstlich auf einen (geeignet zu wählenden) endlichdimensionalen Funktionenraum ein. Im einfachsten Fall unterteilt man dazu das Intervall I und hält die Steuerung u auf jedem Teilintervall konstant. Für die meisten Anwendungen reicht dies völlig aus.

Der „unendlichdimensionale Parameter“ u wird auf diese Weise durch einen endlichdimensionalen approximiert; das *infinite* Steuerungsproblem wird in ein *finites*, beschränktes Optimierungsproblem überführt.

Vorteil dieses Ansatzes ist, daß für die numerische Formulierung keine genaue Analyse oder umständliche Transformation des ursprünglichen Steuerungsproblems erforderlich ist.

Durch die gewählte Diskretisierung der Steuerung wird jedoch von vornherein ein (systematischer) Fehler eingeschleppt, die berechneten Lösungen sind nur suboptimal. Außerdem lassen sich Zustandsbeschränkungen nur näherungsweise behandeln.

Einen besonders effizienten und stabilen (Mehrziel-)Algorithmus haben BOCK/PLITT [9] beschrieben. Mit diesem Verfahren (MUSCOD) werden von KONZELMANN [21] Steuerungen für (Industrie-)Roboter nach dem direkten Ansatz berechnet. Dort findet man auch grundlegende Ergebnisse zur Existenz von Lösungen des Roboter-Steuerungsproblems.

1.3 Der indirekte Lösungsansatz

In dieser Arbeit soll zur Lösung des Steuerungsproblems aus Abschnitt (1.1) der *indirekte Ansatz* benutzt werden. Er beruht auf der Tatsache, daß sich mit Hilfe der sogenannten *adjungierten Variablen* notwendige Bedingungen für die optimale Steuerung formulieren lassen, so daß diese *punktweise* aus dem Maximumprinzip berechnet werden kann. Die Existenz einer Lösung wird dabei vorausgesetzt.

Von FREHSE [14], [15] wird unter sehr allgemeinen Voraussetzungen ein Existenzbeweis gegeben, wobei außerdem die Aussagen des nun folgenden Maximumprinzips auf elegante Weise hergeleitet werden.

1.3.1 Das Maximumprinzip

Grundlegend für die folgenden Ausführungen ist der auf PONTRJAGIN [29] zurückgehende

Satz (1.3.1.1): Das Problem (1.1.1–1.1.3) ohne Parameter p besitze eine Lösung

$$(u, x) \colon I \to U \times \mathbb{R}^{n_d}. \tag{1.3.1.2}$$

Dann existieren LAGRANGE-*Multiplikatoren*

$$\alpha := (\alpha_1, \alpha_2^\top)^\top \in \mathbb{R}_0^+ \times \mathbb{R}^{n_2} \tag{1.3.1.3}$$

und *adjungierte Variablen*

$$\lambda \in H^{1,\infty}(I, \mathbb{R}^{n_d} \setminus \{0\}), \tag{1.3.1.4}$$

so daß mit der HAMILTON-*Funktion*

$$H(t, x, \lambda, u) := \lambda^\top f(t, x, u) \tag{1.3.1.5}$$

$$\tilde{H}(t) := H(t, x(t), \lambda(t), u(t)) \tag{1.3.1.6}$$

und dem *erweiterten Zielfunktional*

$$G(t_s, x_s, t_e, x_e) := \alpha_1 r_1(t_s, x_s, t_e, x_e) + \alpha_2^\top r_2(t_s, x_s, t_e, x_e) \tag{1.3.1.7}$$

die folgenden *notwendigen Bedingungen* erfüllt sind: λ genügt auf I den *adjungierten Differentialgleichungen*

$$\begin{aligned} \dot{\lambda}(t)^\top &= -H_x(t, x(t), \lambda(t), u(t)) \\ &= -\lambda(t)^\top f_x(t, x(t), u(t)), \end{aligned} \tag{1.3.1.8}$$

die *Transversalitätsbedingungen*

$$\lambda(t_s)^\top = \quad G_{x_s}(t_s, x(t_s), t_e, x(t_e)) \tag{1.3.1.9}$$

$$\lambda(t_e)^\top = -G_{x_e}(t_s, x(t_s), t_e, x(t_e)) \tag{1.3.1.10}$$

$$\tilde{H}(t_s) \quad = -G_{t_s}(t_s, x(t_s), t_e, x(t_e)) \tag{1.3.1.11}$$

$$\tilde{H}(t_e) \quad = \quad G_{t_e}(t_s, x(t_s), t_e, x(t_e)) \tag{1.3.1.12}$$

sind erfüllt, und auf I gilt punktweise (fast überall) das *Maximumprinzip*:

$$\tilde{H}(t) = \max_{v \in U} H(t, x(t), \lambda(t), v). \tag{1.3.1.13}$$

u und x heißen *extremal*, wenn sie die notwendigen Bedingungen (1.3.1.8–1.3.1.13) erfüllen.

Beweis: Einen Beweis für das Maximumprinzip in dieser Form findet man in FLEMING/RISHEL [13]. Er beruht auf einem Variationsansatz und ist aufgrund technischer Schwierigkeiten sehr lang. ∎

Bemerkungen und Erläuterungen:

• Die Differentialgleichungen für Zustands- und adjungierte Variablen weisen eine ganz spezielle Struktur auf, die man *symplektisch* nennt:

$$\begin{aligned} \dot{x} &= \quad H_\lambda(t, x, \lambda, u, p), \\ \dot{\lambda} &= -H_x(t, x, \lambda, u, p). \end{aligned} \tag{1.3.1.14}$$

Dieser Begriff stammt ebenso wie die Bezeichnung HAMILTON-Funktion aus der klassischen Mechanik, wo die *Ortsvariablen* q und die *kanonisch konjugierten Impulsvariablen* p „denselben" Differentialgleichungen genügen:

$$\begin{aligned} \dot{q} &= \quad H_p(p, q, t), \\ \dot{p} &= -H_q(p, q, t). \end{aligned} \tag{1.3.1.15}$$

Die physikalische Theorie dazu findet man in ARNOLD [1], das mathematische Hintergrundwissen in KLINGENBERG [20].

• α und λ sind durch die adjungierten Differentialgleichungen und die Transversalitätsbedingungen nur bis auf einen gemeinsamen, aber beliebigen Faktor bestimmt. Die Bedingung $\alpha_1 \geq 0$ legt dessen Vorzeichen fest, die Bedingung $\lambda \neq 0$ schließt den Faktor Null aus (für $\alpha = 0$ und $\lambda = 0$ sind (1.3.1.8–1.3.1.13) selbstverständlich erfüllt!). Ändert man das Vorzeichen bei α_1 oder in allen Transversalitätsbedingungen, so erhält man anstelle der Maximierungsbedingung (1.3.1.13) eine (ansonsten gleichlautende) Minimierungsbedingung.

Für $\alpha_1 = 0$ sind die notwendigen Bedingungen völlig unabhängig vom Zielfunktional. Alle zulässigen Lösungen sind dann extremal und bereits durch die Zustandsdifferentialgleichung mit Randbedingungen vollständig bestimmt. Das Steuerungsproblem ist *entartet* (zum reinen Randwertproblem).

Andernfalls ist $\alpha_1 > 0$ ein reiner Skalierungsfaktor, man setzt dann normalerweise $\alpha_1 := 1$.

• Die Transversalitätsbedingungen (1.3.1.9/1.3.1.10) legen gerade die von den Randbedingungen (1.1.3) noch offengelassenen Freiheitsgrade fest. Im allgemeinen Fall hat man zusammen genau $2n_d + n_2$ Bedingungen für die $2n_d + n_2$ Variablen x, λ und α_2. Oft hat jedoch die Ableitung $\partial r_2 / \partial(t_s, x_s, t_e, x_e)$ Vollrang. Dann läßt sich α_2 eliminieren und man erhält genau $2n_d - n_2$ „echte" Transversalitätsbedingungen. Sind z.B. die Randbedingungen separiert und ist eine Komponente $x_i(t_e)$ frei, so lautet die zugehörige Transversalitätsbedingung $\lambda_i(t_e) = 0$. Ist umgekehrt $x_i(t_e) = x_{ie}$ vorgeschrieben, dann ist $\lambda_i(t_e)$ frei. Entsprechendes gilt im Startpunkt t_s. Analog werden durch (1.3.1.11) und (1.3.1.12) gegebenenfalls t_s und t_e festgelegt.

Das Steuerungsproblem wird somit in ein reines Randwertproblem mit gleich vielen Variablen und Randbedingungen transformiert. Die Steuerung u wird punktweise berechnet, also „ohne Diskretisierungsfehler“, denn die einzige Diskretisierung findet bei der numerischen Integration der Differentialgleichungen statt, sie ist dem Verlauf der Lösung angepaßt und i.a. für alle kontinuierlichen Variablen gleich. Ein weiterer Vorzug des indirekten Ansatzes ist die Möglichkeit, Feedback-Steuerungen mitzuberechnen (siehe KRÄMER-EIS [22], [23], KRÄMER-EIS/BOCK [24], [25]).

Diesen Vorteilen stehen auch Nachteile gegenüber: Die Dimension des Differentialgleichungssystems wird verdoppelt, gerade die adjungierten Differentialgleichungen sind oft sehr instabil. Weiter kann die Beschaffung guter Startdaten für die adjungierten Variablen problematisch sein. Schließlich sind vor Beginn der numerischen Rechnungen meist umfangreiche (problemabhängige) analytische Vorarbeiten notwendig, z.B., um Informationen über die *Schaltstruktur* der Lösung zu gewinnen (also die Reihenfolge von freien Bögen, Randbögen und singulären Bögen; vgl. Abschnitte 1.4 und 3.3).

Ob sich der Mehraufwand gegenüber dem direkten Ansatz lohnt, wird man im Einzelfall entscheiden müssen. Häufig ist es aber auch sinnvoll, *beide* Methoden einzusetzen. Bei besonders schwierigen Problemen wird z.B. eine suboptimale Lösung benötigt, um brauchbare Startdaten für den indirekten Ansatz zu gewinnen. In Abschnitt (3.3) wird diese Strategie am Beispiel der Roboter-Steuerung näher beschrieben.

Es soll noch kurz erläutert werden, wie auch Ungleichungsbedingungen und Parameter im Maximumprinzip berücksichtigt werden können. Die Ungleichungsbedingungen (1.1.4) koppelt man einfach mit weiteren LAGRANGE-Multiplikatoren an das erweiterte Zielfunktional an:

$$\begin{aligned} G(t_s,x_s,t_e,x_e) := \alpha_1 r_1(t_s,x_s,t_e,x_e) &+ \alpha_2^\top r_2(t_s,x_s,t_e,x_e) \\ &+ \alpha_3^\top r_3(t_s,x_s,t_e,x_e). \end{aligned} \qquad (1.3.1.16)$$

Die Komponenten von α_3 erfüllen dann die *Vorzeichenbedingungen*

$$\alpha_{3i} \geq 0, \quad \alpha_{3i} r_{3i}(t_s,x(t_s),t_e,x(t_e)) = 0 \qquad (i \in \{1,\dots,n_3\}). \qquad (1.3.1.17)$$

Hat man n_3' *aktive* Bedingungen ($r_{3i} = 0$), dann sind die zugehörigen n_3' Komponenten $\alpha_{3i} \geq 0$ und treten in den Transversalitätsbedingungen auf. Wegen $\alpha_{3i} r_{3i} = 0$ sind die übrigen $n_3 - n_3'$ Komponenten als $\alpha_{3i} = 0$ festgelegt und damit eliminiert.

Die Parameter faßt man *formal* als zusätzliche Zustandsvariablen mit trivialer Differentialgleichung auf ($\dot p(t) = 0$) und führt eine formale Randbedingung ein ($p(t_s) = p$ oder $p(t_e) = p$). Für die zugehörigen adjungierten Variablen $\eta \in H^{1,\infty}(I, \mathbb{R}^{n_p})$ mit $(\lambda(t)^\top, \eta(t)^\top) \neq 0 \ \forall t \in I$ lauten dann die Differentialgleichungen

$$\begin{aligned} \dot\eta(t) &= -H_p(t,x(t),\lambda(t),u(t),p(t)) \\ &= -\lambda(t)^\top f_p(t,x(t),u(t),p(t)) \end{aligned} \qquad (1.3.1.18)$$

und die Transversalitätsbedingungen

$$\eta(t_s)^\top = \quad G_p(t_s,x(t_s),t_e,x(t_e),p(t_s)), \qquad (1.3.1.19a)$$
$$\eta(t_e)^\top = \quad 0, \qquad (1.3.1.20a)$$

bzw.

$$\eta(t_s)^\top = \quad 0, \qquad (1.3.1.19b)$$
$$\eta(t_e)^\top = -G_p(t_s,x(t_s),t_e,x(t_e),p(t_e)). \qquad (1.3.1.20b)$$

Zusammenfassend folgt daraus in beiden Fällen die *Stationaritätsbedingung*

$$\int_{t_s}^{t_e} H_p(t,x(t),\lambda(t),u(t),p)\,dt = G_p(t_s,x(t_s),t_e,x(t_e),p). \qquad (1.3.1.21)$$

Nach einer Transformation auf $I = [0,1]$ erhält man als Stationaritätsbedingung für die Parameter t_s, t_e gerade die Transversalitätsbedingungen (1.3.1.11/1.3.1.12) des ursprünglichen Steuerungsproblems!

Die notwendigen Bedingungen (1.3.1.8–1.3.1.13) der oben angegebenen „Grundversion“ des Maximumprinzips bleiben also formal unverändert. Es werden lediglich neue Komponenten zu α und λ hinzugefügt, die *zusätzliche* notwendige Bedingungen (1.3.1.17/1.3.1.21) erfüllen sowie in H bzw. G berücksichtigt werden müssen.

1.3.2 Berechnung der optimalen Steuerung

Wie schon in Abschnitt (1.3.1) erwähnt, wird mit Hilfe des Maximumprinzips die Steuerung u zugunsten der adjungierten Variablen λ eliminiert, d.h. punktweise aus den übrigen Variablen berechnet. Dazu sei zunächst vorausgesetzt, daß

$$\max_{v\in U} H(t, x(t), \lambda(t), v, p) \tag{1.3.2.1}$$

für alle $t \in I$ existiert; andernfalls ist das Optimierungsproblem nicht sinnvoll gestellt. Weiter sei die Eindeutigkeit der Maximalstelle vorausgesetzt, und damit

$$u(t) := \arg\max_{v\in U} H(t, x(t), \lambda(t), v, p). \tag{1.3.2.2}$$

Liegt z.B. $u(t)$ im Innern des Steuerbereichs U, so gelten

$$H_u(t, x(t), \lambda(t), u(t), p) = 0 \tag{1.3.2.3}$$

und die notwendige Bedingung von Legendre-Clebsch:

$$H_{uu}(t, x(t), \lambda(t), u(t), p) \leq 0. \tag{1.3.2.4}$$

H_{uu} ist also längs der Lösungstrajektorie *negativ semidefinit*. Hinreichend für die Existenz und Eindeutigkeit der Maximalstelle ist bekanntlich die Bedingung $H_{uu} < 0$. Falls sie erfüllt ist, läßt sich $u(t)$ durch Auflösen von (1.3.2.3) wie gewünscht punktweise berechnen, und zwar in der Form

$$u(t) = u^*(t, x(t), \lambda(t), p). \tag{1.3.2.5}$$

Nach dem Satz über implizite Funktionen ist u^* stetig differenzierbar. Da x und λ absolutstetig sind, gilt letzteres auch für u.

Auf diese (vom theoretischen Standpunkt gesehen) einfache Weise kann man die Steuerung z.B. berechnen, wenn der Steuerbereich U der ganze Raum $\mathbb{R}^{n_u}$ oder eine offene Teilmenge davon ist. In der Praxis kommen Probleme recht häufig in dieser Form vor, oder sie lassen sich entsprechend umformulieren.

Für unbeschränktes U ist die Steuerung normalerweise nichtlinear, d.h. f ist nichtlinear bezüglich u; andernfalls existiert das Maximum (1.3.2.1) i.a. nicht.

Falls U abgeschlossen ist oder Teile seines Randes enthält, ist der Sachverhalt wesentlich komplizierter. Denn um die Differentialgleichungen für x und λ über ein Intervall numerisch integrieren zu können, muß die rechte Seite nach t (total) l-mal stetig differenzierbar sein. In Abhängigkeit vom verwendeten Integrator liegt dabei l zwischen zwei (Euler-Verfahren) und sechzehn (bei Verfahren mit extrem hoher Ordnung). Setzt man voraus, daß die zu U gehörenden Teile von ∂U hinreichend glatte Mannigfaltigkeiten mit hinreichend glatten Rändern sind, so kann die obige Differenzierbarkeitsvoraussetzung in den folgenden Fällen verletzt sein:

- u erreicht ∂U aus dem Innern heraus
- u wechselt von einem Randstück auf ein anderes (Sprung möglich!)
- u verläßt ∂U ins Innere hinein

An solchen *Schaltstellen* t_i muß die numerische Integration unterbrochen werden, die rechte Seite wird im obigen Sinne „umgeschaltet".

Für die Numerik ist es daher unerläßlich, das Intervall I durch $t_s < t_1 < \cdots < t_k < t_e$ in endlich viele Teilintervalle zu zerlegen und die Schaltstellen t_i als (einfache, isolierte) Nullstellen sogenannter *Schaltfunktionen*

$$q_i(t_i) := Q_i(t_i, x(t_i), \lambda(t_i), p) \stackrel{!}{=} 0 \tag{1.3.2.6}$$

(implizit) zu charakterisieren, so daß sie während der Integration berechnet werden können. Im einfachsten Fall sind die Schaltstellen $t_i = \hat{t}_i$ explizit bekannt, d.h.

$$q_i(t) = Q_i(t, x(t), \lambda(t), p) = t - \hat{t}_i. \tag{1.3.2.7}$$

Ein nicht offener Steuerbereich U ist meist kompakt, spezieller

- ein konvexes Polyeder U_P oder
- ein (achsenparalleler) Quader U_Q.

Solche Steuerbereiche treten meist bei Problemen mit linearer Steuerung auf. Teilweise abgeschlossene Steuerbereiche oder solche mit „krummen" Kanten und Rändern gibt es in praktischen Problemen seltener.

Gerade bei linearen Problemen kommt es häufig vor, daß die Steuerung auf einem ganzen Intervall $I_S = [t_a, t_b] \subseteq I$ durch das Maximumprinzip (1.3.1.13) nicht eindeutig bestimmt ist. Sie heißt dann *singulär* auf I_S.

Ohne genauere Angaben zum Steuerbereich lassen sich nur wenig relevante Aussagen über Randsteuerungen $u(t) \in \partial U$ und singuläre Steuerungen machen. Für die Steuerung von Robotern und viele andere praktische Probleme ist jedoch die Theorie linearer und insbesondere singulärer Steuerungen von grundlegender Bedeutung. Im folgenden werden deshalb in Anlehnung an die Darstellung in MAURER [27] die wesentlichen Resultate für diesen Fall zusammengestellt. Sei also f linear bzgl. der Steuerung u,

$$f(t,x,u,p) = f^1(t,x,p)u + f^2(t,x,p), \tag{1.3.2.8}$$

und außerdem $U = U_P$ ein kompaktes konvexes Polyeder der Dimension n_u. U_P besitze N Seiten (der Dimension $n_u - 1$). Sei v_i ein Punkt der Seite i und $w_i \neq 0$ ein zu ihr orthogonaler Vektor, der nach außen zeigt. Dann kann man U_P beschreiben als

$$U_P = \{u \in I\!R^{n_u} \mid \forall i \in \{1,\dots,N\}\colon\ \langle u - v_i, w_i\rangle \leq 0\}. \tag{1.3.2.9}$$

Im Spezialfall eines Quaders

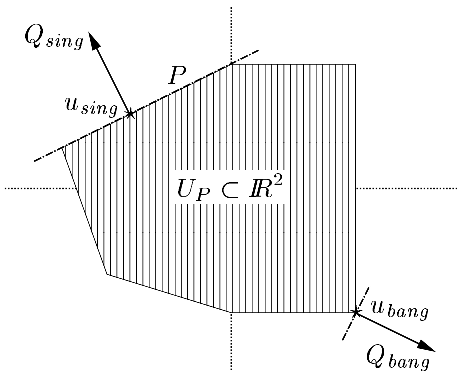
$$U_Q = \left[u_1^-, u_1^+\right] \times \cdots \times \left[u_{n_u}^-, u_{n_u}^+\right] \tag{1.3.2.10}$$

läßt sich, ohne die Struktur von (1.3.2.8) zu zerstören, durch eine affin-lineare Transformation stets $u_i^{\pm} = \pm 1$ erreichen. Gegebenenfalls werde daher o.B.d.A. stets ein solcher *Würfel* betrachtet.

Aus (1.3.2.8) folgt zunächst

$$\begin{aligned} H(t,x,\lambda,u,p) &= \lambda^\top f^1(t,x,p)u + \lambda^\top f^2(t,x,p) \\ &= H^1(t,x,\lambda,p)u + H^2(t,x,\lambda,p), &\quad (1.3.2.11)\\ H_u(t,x,\lambda,u,p) &= H^1(t,x,\lambda,p), &\quad (1.3.2.12)\\ H_{uu}(t,x,\lambda,u,p) &= 0. &\quad (1.3.2.13)\end{aligned}$$

Mit den Bedingungen (1.3.2.3/1.3.2.4) kann man nichts anfangen: H_u und H_{uu} sind beide unabhängig von u, (1.3.2.4) ist stets erfüllt, (1.3.2.3) dagegen längs der optimalen Lösung i.a. nicht (höchstens „zufällig"). Die Steuerung $u(t)$ liegt also stets auf dem Rand (außer für $H^1(t,x,\lambda,p) = 0$), meist sogar in einer Ecke von U_P. Aus der geometrischen Bedeutung der Maximierungsbedingung ist dies auch anschaulich unmittelbar klar (Abbildung 1.3.2.14).



Steuerpolyeder mit Schaltvektoren und Steuerungen (singulär/bang-bang). Abbildung (1.3.2.14)

Denn $\tilde{H}$ nimmt wegen (1.3.2.11) sein Maximum genau für die Werte u an, deren Projektion auf den *Schaltvektor* $\tilde{Q}(t, x, \lambda, p) := H^1(t, x, \lambda, p)^\top$ maximal ist:

$$\begin{aligned} \tilde{H}(t) &= \left\langle \tilde{Q}(t, x(t), \lambda(t), p), u(t) \right\rangle + H^2(t, x(t), \lambda(t), p) \\ &=: \langle \tilde{q}(t), u(t) \rangle + \tilde{H}^2(t). \end{aligned} \tag{1.3.2.15}$$

Sind $\{v_k\}$ die Ecken von U_P, so erhält man als Schaltfunktionen die Projektionen des Schaltvektors $\tilde{q}$ auf die Kanten von U_P:

$$q_{kl}(t) = \langle \tilde{q}(t), v_k - v_l \rangle \,. \tag{1.3.2.16}$$

Für einen Quader $U = U_Q$ sind die Kanten $v_k - v_l$ Vielfache $\beta_i e_i$ der kanonischen Basisvektoren. Dann kann man sogar o.B.d.A. $q_i := \tilde{q}_i$ setzen. Aufgrund dieser einfachen Zusammenhänge bezeichnet man für $U = U_P$ außer q und Q auch $\tilde{q}$ und $\tilde{Q}$ als Schaltvektoren.

Je nachdem, ob singuläre Intervalle auftreten oder nicht, lassen sich die möglichen Steuerungen in die folgenden zwei Klassen einteilen:

1. bang-bang-Steuerungen:
Längs einer Lösungstrajektorie besitzt H fast überall eine eindeutige Maximalstelle $u(t)$, die in einer Ecke v_k des Polyeders U_P liegt. In den Schaltstellen ist die Steuerung nicht eindeutig bestimmt, dort springt u von einer Ecke zu einer anderen — daher die Bezeichnung *bang-bang-Steuerung*. Zur praktischen Berechnung von u bietet sich der *Simplex-Algorithmus* an. Man beachte, daß u zwischen den Sprüngen konstant ist, also jeweils nur einmal berechnet werden muß.
Im Fall des Quaders (bzw. Würfels) $U = U_Q$ kann man jede Steuerungskomponente getrennt behandeln. H wird dann maximal für

$$\begin{aligned} u_i(t) &= u_i^*(t, x(t), \lambda(t), p) \\ &:= \operatorname{sign} H_i^1(t, x(t), \lambda(t), p) \\ &= \operatorname{sign} q_i(t), \end{aligned} \tag{1.3.2.17}$$

$$u(t) = \operatorname{sign} q(t). \tag{1.3.2.18}$$

In den Nullstellen von q_i (sie sind alle isoliert!) springt die Steuerung u_i von -1 nach $+1$ oder umgekehrt.

2. singuläre Steuerungen:
Es gibt (mindestens) ein ganzes Intervall $I_S \subseteq I$, auf dem die Steuerung durch das Maximumprinzip nicht eindeutig bestimmt ist. Es werde o.B.d.A. angenommen, daß im Innern von I_S keine Schaltstelle liegt (andernfalls zerlege man I_S in mehrere singuläre Intervalle mit dieser Eigenschaft). Falls $q = 0$ auf I_S, ist H völlig unabhängig von u und nimmt für alle $u \in U_P$ den gleichen (maximalen) Wert an. Die Steuerung heißt dann *totalsingulär*, nur in diesem Fall kann sie im Innern liegen.
Andernfalls gibt es ein Seitenpolyeder $P \subseteq \partial U_P$ der Dimension $d := \dim P \in \{1, \ldots, n_u - 1\}$, auf dem H für alle $t \in I_S$ sein Maximum annimmt. P ist konvex, der Schaltvektor q steht senkrecht auf P und zeigt nach außen (siehe Abbildung 1.3.2.14).

Sei $J_P \subseteq \{1, \ldots, N\}$ die Indexmenge der $(n_u - 1)$-dimensionalen Seiten von U_P, deren Schnitt P ist, und sei $v \in P$. Dann ist

$$P = \{u \in U_P \mid \forall j \in J_P \colon \ \langle u - v, w_j \rangle = 0\} \tag{1.3.2.19}$$

und auf I_S gilt wegen $\tilde{q} \perp P$ für alle zeitlichen Ableitungen

$$\left(\frac{d}{dt}\right)^n \langle u - v, \tilde{q}(t) \rangle = 0 \qquad \forall u, v \in P. \tag{1.3.2.20}$$

Wählt man d linear unabhängige Differenzen $u - v$ (z.B. geeignete Kanten $v_k - v_l$), so erhält man d Gleichungen für $\tilde{q} = \tilde{Q}$, sowie deren Ableitungen. Daraus lassen sich i.a. Informationen über die singulären Steuerungen gewinnen, wenn man für die Ableitungen von x und λ die bekannten Beziehungen (1.3.1.14) einsetzt. f sei dazu entsprechend oft differenzierbar.

Betrachtet man wieder den Spezialfall eines Quaders U_Q, so läßt sich wie in (1.3.2.17) jede Komponente getrennt behandeln. Sei also u_i singulär für ein $i \in \{1, \dots, n_u\}$, d.h. $q_i(t) = 0$ auf I_S. Es gilt

$$q_i^{(n)}(t) = \left(\frac{d}{dt}\right)^n Q_i(t, x(t), \lambda(t), p) \stackrel{!}{=} 0 \qquad \forall n \in I\!N,\ \forall t \in I_S. \tag{1.3.2.21}$$

Sei $\tilde{u} := (u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_{n_u})^\top$. Ist dann $\tilde{u}(t) = \tilde{u}^*(t, x(t), \lambda(t), p)$ aus dem Maximumprinzip auf I_S bekannt (unabhängig von u_i!), so hat die n-te Ableitung von q_i die Gestalt

$$q_i^{(n)}(t) = a_n(t, x(t), \lambda(t), p) + b_n(t, x(t), \lambda(t), p) u_i(t). \tag{1.3.2.22}$$

Sei nun k die kleinste Zahl mit $b_k(t, x(t), u(t), p) \neq 0$ auf I_S. Ist $k < \infty$, so läßt sich u_i berechnen als

$$u_i(t) = u_i^*(t, x(t), \lambda(t), p) := -\frac{a_k(t, x(t), \lambda(t), p)}{b_k(t, x(t), \lambda(t), p)}. \tag{1.3.2.23}$$

Nach Kelley [17] ist $k =: 2\kappa_i$ stets gerade (oder ∞). Der Beweis ist technisch und relativ einfach. Tieferer Grund für diese „Geradzahligkeit“ ist die symplektische Struktur der Differentialgleichungen.

Definition (1.3.2.24): $\kappa_i \in I\!N \cup \{\infty\}$ heißt *Ordnung* der singulären Steuerung u_i auf I_S.

Auf einem singulären Intervall I_S für u_i ist die Legendre-Clebsch-Bedingung nach (1.3.2.4) stets erfüllt. In Kelley [17] wird eine schärfere notwendige Bedingung für die Optimalität von u_i bewiesen, die sogenannte *verallgemeinerte* Legendre-Clebsch-*Bedingung*: Im Innern von I_S gilt

$$\begin{aligned} 0 &\stackrel{!}{\leq} (-1)^{\kappa_i+1} \frac{\partial}{\partial u_i} \frac{d^{2\kappa_i}}{dt^{2\kappa_i}} \frac{\partial}{\partial u_i} H(t, x(t), \lambda(t), u(t), p) \\ &= (-1)^{\kappa_i+1}\, b_{2\kappa_i}(t, x(t), \lambda(t), p, \tilde{u}(t)) \end{aligned} \tag{1.3.2.25}$$

mit $b_{2\kappa_i}$ aus (1.3.2.22).

Die Ergebnisse dieses Abschnitts haben noch eine wichtige Konsequenz: Da sich u auf singulären Intervallen in einem Polyeder bewegt, auf dem $\tilde{q}$ senkrecht steht, ist auch $\dot{u}(t) \perp \tilde{q}(t)$. Wegen $\tilde{q}(t) = H^1(t, x(t), \lambda(t), p)^\top = H_u(t, x(t), \lambda(t), p)^\top$ folgt daraus $H_u \dot{u} = 0$. Dies gilt sogar für alle $t \in I$ mit Ausnahme der Schaltstellen, denn für eine bang-bang-Steuerung ist u ohnehin (stückweise) konstant. Damit erhält man

$$\begin{aligned} \frac{d\tilde{H}}{dt} &= \frac{\partial H}{\partial t} + \frac{\partial H}{\partial x}\dot{x} + \frac{\partial H}{\partial \lambda}\dot{\lambda} + \frac{\partial H}{\partial u}\dot{u} \\ &= \frac{\partial H}{\partial t} - \dot{\lambda}^\top \dot{x} + \dot{x}^\top \dot{\lambda} + 0 \\ &= \frac{\partial H}{\partial t}. \end{aligned} \tag{1.3.2.26}$$

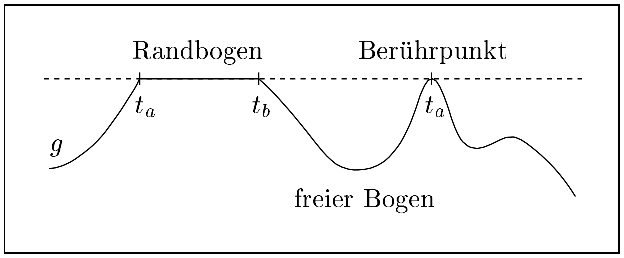
Ist also die Differentialgleichung der Zustandsvariablen autonom, so ist auch H nicht explizit von t abhängig und damit längs einer Lösung zeitlich konstant:

$$\frac{\partial f}{\partial t}(t, x, u, p) = 0 \quad \Rightarrow \quad \frac{d\tilde{H}}{dt}(t) = \frac{\partial H}{\partial t}(t, x(t), \lambda(t), u(t), p) \stackrel{!}{=} 0. \tag{1.3.2.27}$$

Dieses Resultat wurde zwar nur für lineare Steuerungen mit Polyedern als Steuerbereich hergeleitet, es gilt jedoch allgemein.

1.3.3 Zustandsbeschränkungen

Die Berücksichtigung von Zustandsbeschränkungen steht in engem Zusammenhang mit singulären Steuerungen und soll hier der Vollständigkeit halber kurz beschrieben werden. Da in dieser Arbeit jedoch keine Zustandsbeschränkungen benötigt werden, wird zur Vereinfachung nur der Fall einer skalaren Beschränkung behandelt, d.h. $n_g := 1$.



Möglicher zeitlicher Verlauf einer Zustandsbeschränkung. Abbildung (1.3.3.1)

Wir betrachten eine Lösung von (1.1.1–1.1.5). Ein Intervall, auf dem die Beschränkung *inaktiv* ist (also $g(t, x(t), u(t), p) < 0$), heißt *freier Bogen.* Intervalle $I_R := [t_a, t_b]$, auf denen die Beschränkung *aktiv* ist ($g(t, x(t), u(t), p) = 0$), nennt man *Randbögen.* Letztere können zu *Berührpunkten* t_a entarten (Abbildung 1.3.3.1). *Aufsprungpunkt* t_a und *Absprungpunkt* t_b eines Randbogens sind Schaltstellen, g kann gewissermaßen als zugehörige „Schaltfunktion“ angesehen werden.

Zunächst eine Bemerkung zur praktischen Vorgehensweise. Hier hat sich die folgende *Homotopiemethode* vorzüglich bewährt (Bock [3], [5]): Im ersten Schritt berechnet man eine Lösung (x_0, u_0, p_0) des Problems (1.1.1–1.1.4), also *ohne* Zustandsbeschränkung. Sei

$$\epsilon_0 := \max_{t \in I} g(t, x_0(t), u_0(t), p_0) \tag{1.3.3.2}$$

das Maximum von g längs dieser Lösung, und $\epsilon_0 > \epsilon_1 > \cdots > \epsilon_l := 0$. Für $k \in \{0, \ldots, l\}$ sei weiter

$$g_k(t, x, u, p) := g(t, x, u, p) - \epsilon_k. \tag{1.3.3.3}$$

Dann ist (x_0, u_0, p_0) Lösung des durch g_0 beschränkten Steuerungsproblems. Für $k \in \{1, \ldots, l\}$ löst man nun schrittweise das Steuerungsproblem mit der verschärften Zustandsbeschränkung g_k, wobei x_{k-1}, p_{k-1}, λ_{k-1} als Startdaten verwendet werden (vgl. Abschnitt 1.4).

Die Lösung $(x, u, p) := (x_l, u_l, p_l)$ besitze endlich viele freie Bögen und Randbögen bzw. Berührpunkte. Auf einem Randbogen I_R verschwindet g identisch, also auch alle zeitlichen Ableitungen:

$$\left(\frac{d}{dt}\right)^n g(t, x(t), u(t), p) = 0 \qquad \forall n \in \mathbb{N},\ \forall t \in I_R. \tag{1.3.3.4}$$

Man leitet nun g formal so oft ab, bis die Steuerung u explizit in einer Abeitung auftaucht:

$$\begin{aligned} \left(\frac{d}{dt}\right)^n g(t, x, u, p) &= g_t^{(n-1)}(t, x, p) + g_x^{(n-1)}(t, x, p) \cdot f(t, x, u, p) \\ &=: \begin{cases} g^{(n)}(t, x, p) & (n < \kappa) \\ g^{(n)}(t, x, u, p) & (n = \kappa) \end{cases} \end{aligned} \tag{1.3.3.5}$$

mit

$$g(t, x, u, p) =: g^{(0)}(t, x, [u,]\, p). \tag{1.3.3.6}$$

Die entsprechenden Differenzierbarkeitseigenschaften für f und g seien nachträglich vorausgesetzt.

Definition (1.3.3.7): $\kappa \in \mathbb{N} \cup \{\infty\}$ heißt *Ordnung* der Zustandsbeschränkung g auf I_R.

In einfachen Fällen kann man hinreichende Bedingungen für $\kappa < \infty$ angeben. Bei linearen Steuerungsproblemen ist z.B. $\kappa \leq n_d$, wenn das System *steuerbar* ist (Bock [2]).

Um notwendige Bedingungen zu formulieren, koppelt man nach Bryson/Denham/Dreyfus [10] $g^{(\kappa)}$ mit einem neuen Lagrange-Multiplikator μ an die Hamilton-Funktion an:

$$\begin{aligned} H(t, x, \lambda, \mu, u, p) &:= \lambda^\top f(t, x, u, p) - \mu g^{(\kappa)}(t, x, u, p), \\ \tilde{H}(t) &:= H(t, x(t), \lambda(t), \mu(t), u(t), p). \end{aligned} \tag{1.3.3.8}$$

Dann existieren $\{\nu_0, \ldots, \nu_{\kappa-1}\} \subset I\!R_0^+$, mit denen zusätzlich zu den notwendigen Bedingungen aus Abschnitt (1.3) gilt:

$$\begin{aligned} \mu(t) &= 0 \qquad \text{für } g < 0 \text{ (freie Bögen)}, \\ \mu(t) &\geq 0 \qquad \text{für } g = 0 \text{ (Randbögen)}. \end{aligned} \tag{1.3.3.9}$$

λ ist stetig außer in t_a:

$$\lambda(t_a+) = \lambda(t_a-) + \begin{cases} \nu_0 g_x^{(0)}(t_a, x(t_a), p) & \text{bei Berührpunkten,} \\ \sum\limits_{n=0}^{\kappa-1} \nu_n g_x^{(n)}(t_a, x(t_a), p) & \text{bei Randbögen.} \end{cases} \tag{1.3.3.10}$$

$\tilde{H}$ ist stetig außer in t_a:

$$\tilde{H}(t_a+) = \tilde{H}(t_a-) - \begin{cases} \nu_0 g_t^{(0)}(t_a, x(t_a), p) & \text{bei Berührpunkten,} \\ \sum\limits_{n=0}^{\kappa-1} \nu_n g_t^{(n)}(t_a, x(t_a), p) & \text{bei Randbögen.} \end{cases} \tag{1.3.3.11}$$

Beim Aufsprung auf Randbögen mit $\kappa = 0$ sind also sowohl λ als auch $\tilde{H}$ stetig, bei autonomen Problemen ist $\tilde{H}$ sogar überall konstant! Auch hier bleiben die notwendigen Bedingungen formal unverändert, sie werden um die Sprungbedingungen (1.3.3.10/1.3.3.11) ergänzt.

Auf freien Bögen ist $\mu = 0$, die Steuerung wird berechnet wie in Abschnitt (1.3.2) beschrieben. Auf Randbögen (mit $\kappa < \infty$!) berechnet man die Steuerung durch Auflösen von $g^{(\kappa)} = 0$ (1.3.3.5) nach u. Sie liegt dann meist im Innern des Steuerbereichs, so daß man μ aus der Bedingung $H_u = 0$ bestimmen kann. Für eine skalare Steuerung ($n_u = 1$) ist einfach

$$\mu(t) = \frac{\lambda(t)^\top f_u(t, x(t), u(t), p)}{g_u^{(\kappa)}(t, x(t), u(t), p)}. \tag{1.3.3.12}$$

Der Aufsprungpunkt t_a wird implizit aus der Sprungbedingung (1.3.3.11) bestimmt, die Stetigkeit von $\tilde{H}$ in t_b legt den Absprungpunkt fest.

„Echte“ Berührpunkte ($\nu_0 > 0$) können nach Maurer [28] nur bei Beschränkungen der Ordnung $\kappa > 1$ auftreten. Für geradzahliges κ gehen diese in Randbögen über, wenn man die Beschränkung verschärft ($\epsilon \searrow 0$). Für ungeradzahliges κ hingegen können keine Randbögen auftreten, sondern nur Berührpunkte.

Zum Schluß sei noch ein für Theorie und Numerik sehr wichtiger Satz von Bock [2] zitiert, der eine wesentliche Verschärfung von (1.3.3.9) darstellt:

Satz (1.3.3.13): μ ist κ-mal stetig differenzierbar, mit $\mu_n(t) := (-1)^n \mu^{(n)}(t)$ für $n \in \{0, \ldots, \kappa\}$ gilt:

$$\mu_n \geq 0 \text{ auf } I_R \qquad (n \in \{0, \ldots, \kappa\}), \tag{1.3.3.14}$$

$$\mu_n(t_a) = \nu_{\kappa-n-1}, \qquad \mu_n(t_b) = 0 \qquad (n \in \{0, \ldots, \kappa-2\}), \tag{1.3.3.15}$$

$$\mu_{\kappa-1}(t_a) \geq \nu_0, \qquad \mu_{\kappa-1}(t_b) \geq 0. \tag{1.3.3.16}$$

Für $\kappa = 1$ gilt in beiden Ungleichungen (1.3.3.16) die Gleichheit. ∎

Genauere Informationen zur Behandlung zustandsbeschränkter Steuerungsprobleme findet man z.B. in Bock [5], Krämer-Eis [23] und Maurer [28].

1.4 Ein Mehrzielverfahren zur Lösung von Randwertproblemen

In diesem Abschnitt soll beschrieben werden, wie das in ein Randwertproblem transformierte Steuerungsproblem numerisch behandelt werden kann. Die mathematische Analyse und die praktische Erfahrung zeigen, daß dazu der sehr robuste und effiziente *Mehrzielansatz* (Bulirsch [11], Bock [3], [6]) besonders gut geeignet ist. Hier wurde der von BOCK [2], [5] entwickelte und im Programmpaket OPCON realisierte Mehrzielalgorithmus verwendet. Man kann damit sehr allgemeine überbestimmte Mehrpunktrandwertprobleme mit Schalt- und Sprungbedingungen lösen: Im Intervall I seien k Schaltstellen

$$t_s =: t_0 < t_1 < \cdots < t_k < t_{k+1} := t_e \tag{1.4.1}$$

als Nullstellen zugehöriger Schaltfunktionen gegeben:

$$Q_i(t_i, z(t_i), p) = 0 \qquad (i \in \{1, \ldots, k\}). \tag{1.4.2}$$

Gesucht sind ein Parametervektor $p \in \mathbb{R}^{N_p}$ und eine linksseitig stetige, stückweise stetig differenzierbare *Trajektorie* $z\colon I \to \mathbb{R}^{N_d}$, die die Differentialgleichung mit *Schaltbedingungen*

$$\begin{aligned} \dot z(t) &= f(t, z(t), p, Q(t, z(t), p)) && \text{auf } [t_0, t_{k+1}] \\ &=: f_i(t, z(t), p) && \text{auf }]t_i, t_{i+1}] \qquad (i \in \{0, \ldots, k\}) \end{aligned} \tag{1.4.3}$$

löst, die *Sprungbedingungen*

$$z(t_i+) = z(t_i) + d_i(t_i, z(t_i), p) \qquad (i \in \{1, \ldots, k\}) \tag{1.4.4}$$

sowie die folgenden *Minimierungs-*, *Gleichungs-* und *Ungleichungsbedingungen* erfüllt:

$$\|r_1(z(t_0), \ldots, z(t_{k+1}), p)\|_2 = \min, \tag{1.4.5}$$

$$r_2(z(t_0), \ldots, z(t_{k+1}), p) \quad = 0, \tag{1.4.6}$$

$$r_3(z(t_0), \ldots, z(t_{k+1}), p) \quad \geq 0. \tag{1.4.7}$$

Hierbei sei

$$\begin{aligned} Q_i &\in C^2(I \times \mathbb{R}^{N_d} \times \mathbb{R}^{N_p}, \mathbb{R}), \\ d_i &\in C^2(I \times \mathbb{R}^{N_d} \times \mathbb{R}^{N_p}, \mathbb{R}^{N_d}), \\ f_i &\in C^2([t_i, t_{i+1}] \times \mathbb{R}^{N_d} \times \mathbb{R}^{N_p}, \mathbb{R}^{N_d}), \\ r_i &\in C^2(\underbrace{\mathbb{R}^{N_d} \times \cdots \times \mathbb{R}^{N_d}}_{k+2} \times \mathbb{R}^{N_p}, \mathbb{R}^{N_i}). \end{aligned} \tag{1.4.8}$$

Das Steuerungsproblem ohne Zustandsbeschränkung (1.1.5) liege o.B.d.A. in MAYER-Form vor und sei auf $I := [0, 1]$ transformiert, so daß t_s und t_e in (1.1.1–1.1.4) als Parameter auftreten. Es läßt sich dann folgendermaßen mit OPCON behandeln:

Im Vektor z sind die Zustandsvariablen x und die adjungierten Variablen λ, η zusammengefaßt. Die rechte Seite f enthält die entsprechenden Differentialgleichungen, ihre Anzahl ist $N_d = 2n_d + n_p$. p enthält den ursprünglichen Parametervektor sowie α_2 und α_3, also $N_p = n_p + n_2 + n_3$ Komponenten.

Die Schaltstellen und -funktionen t_i, Q_i sind wie in Abschnitt (1.3.2) definiert. Zwecks einfacher Formulierung wird in diesem Kapitel angenommen, daß jede Schaltstelle t_i eine „eigene“ Schaltfunktion Q_i besitzt. Tatsächlich kann jede Schaltfunktion mehrere Nullstellen haben, wie z.B. bei der Roboter-Steuerung. Außerdem ist die Anzahl k der Schaltstellen möglicherweise unbekannt und wird erst im Verlauf der numerischen Rechnung mitgeliefert.

Die Gleichungsbedingungen (1.4.6) fassen die klassischen Randbedingungen (1.1.3) und die Transversalitätsbedingungen (1.3.1.9–1.3.1.12) sowie (1.3.1.19/1.3.1.20) zusammen. Ihre Gesamtzahl ist (maximal) $N_2 = 2n_d + 2n_p + n_2 + n_3$. Wie schon in Abschnitt (1.3.1) bemerkt, fallen durch Elimination von α_2 oft n_2 Bedingungen weg.

Bei reinen Steuerungsproblemen ist die Unterscheidung (1.4.5/1.4.6) im Prinzip hinfällig, denn im Maximumprinzip (1.3.1.1) treten nur Gleichungs(rand)bedingungen auf. Aus numerischen Gründen empfiehlt es sich jedoch manchmal, diese als *Ausgleichsbedingungen* der Form (1.4.5) aufzufassen.

r_3 umfaßt in der Form (1.4.7) die ursprünglichen Ungleichungsbedingungen (1.1.4) und die Vorzeichenbedingungen (1.3.1.17). Zusammen hat man $N_3 = 2n_3$ Ungleichungsbedingungen, von denen genau die Hälfte aktiv ist.

Eine genaue Übersicht aller Variablen und Bedingungen (außer den Schaltbedingungen), die bei der Formulierung des Mehrpunktrandwertproblems für das allgemeine Steuerungsproblem mit dem Zielfunktional (1.1.7) entstehen können, ist in Tabelle (1.4.9) zusammengestellt.

Steuerungsproblem mit	gegeben	zusätzlich (Satz 1.3.1.1, ...)	N_d	N_p	N_2	N_3
Zustandsvariablen	n_d Zust.var. x	n_d adj. Var. λ, n_2 Mult. α_2	$2n_d$	n_2		
und Randbedingungen	n_2 Randbed. r_2	$2n_d$ Transv.bed. für λ			$2n_d + n_2$	
Parametern	n_p Parameter p	n_p adjungierte Variablen η	n_p	n_p		
		$2n_p$ Transv.bed. für η			$2n_p$	
Ungleichungs-		n_3 Multiplikatoren α_3		n_3		
bedingungen	n_3 Ungl.bed. r_3	n_3 Gl.- und Ungl.bed. für α_3			n_3	$2n_3$
freier Startzeit	1 Zeitpunkt t_s			1		
		1 Transv.bedingung für $\tilde{H}$			1	
freier Endzeit	1 Zeitpunkt t_e			1		
		1 Transv.bedingung für $\tilde{H}$			1	
LAGRANGE-	1 Zust.var. x	1 adj. Variable λ (konstant!)	1	1		
Zielfunktional	1 Anf.bed. r_2	1 Gleichungsbedingung für λ			2	
TSCHEBYSCHEFF-	1 Parameter p	1 adjungierte Variable η	1	1		
Zielfunktional		2 Transv.bedingungen für η			2	
Randbogen		κ Multiplikatoren ν		κ		
		κ Gleichungsbed. für ν			κ	

Variablen und Bedingungen des assoziierten Mehrpunktrandwertproblems. Tabelle (1.4.9)

Es soll noch kurz auf die weiteren Möglichkeiten von OPCON eingegangen werden. Sie werden hauptsächlich in Aufgaben der *Parameteridentifizierung* benötigt (BOCK [6]), wie sie u.a. bei der *Modellanpassung* auftreten. Um einen Prozeß (z.B. eine chemische Reaktion oder eine Roboterbewegung) quantitativ richtig beschreiben zu können, muß man oft ungenau bekannte Konstanten (Parameter) der Zustandsdifferentialgleichung *experimentell* ermitteln. In den *Meßpunkten* t_i werden zu diesem Zweck geeignete *Meßdaten* erhoben. Die Parameter sollen nun so bestimmt werden, daß das Modell den Prozeß möglichst gut reproduziert. Mathematisch bedeutet dies, die (geeignet gewichteten) Abweichungen der theoretischen Werte von den Meßwerten gemäß (1.4.5) zu minimieren, wobei in den Meßpunkten zusätzliche Bedingungen (1.4.6/1.4.7) berücksichtigt werden können.

Mit den Sprungbedingungen (1.4.4) schließlich erfaßt man Unstetigkeiten der adjungierten Variablen, wie sie nach (1.3.3.10) bei Zustandsbeschränkungen auftreten können. In manchen Problemen kommen jedoch auch Sprünge der Zustandsvariablen vor. So werden beispielsweise in chemischen Reaktionen durch Hinzugießen einer Spezies plötzliche Konzentrationsänderungen bewirkt, ebenso treten beim Abtrennen von Stufen einer Trägerrakete Massensprünge auf.

Für die numerische Behandlung mit der Mehrzielmethode wird das Intervall I durch ein Gitter Γ_m mit $m \geq 2$ *Stützstellen* τ_j in $m-1$ *Mehrzielintervalle* $I_j := [\tau_j, \tau_{j+1}]$ unterteilt:

$$\Gamma_m: \; t_s = \tau_1 < \cdots < \tau_m = t_e. \tag{1.4.10}$$

Die *diskrete Trajektorie*

$$(s_1, \ldots, s_m) := (z(\tau_1), \ldots, z(\tau_m)) \tag{1.4.11}$$

bildet zusammen mit den Parametern p den Vektor $S := (s_1, \ldots, s_m, p)$. Berechnet man zu gegebenem S auf jedem Teilintervall I_j die Lösung $z(t; \tau_j, s_j, p)$ des Anfangswertproblems

$$z(\tau_j) = s_j; \qquad \dot{z}(t) = f(t, z(t), p, Q(t, z(t), p)) \tag{1.4.12}$$

mit Schalt- und Sprungbedingungen

$$\begin{aligned} Q_i(t_i, z(t_i), p) &= 0 \\ [3pt] z(t_i+) - z(t_i) - d_i(t_i, z(t_i), p) &= 0 \end{aligned} \qquad (i \in \{1, \ldots, k\}), \tag{1.4.13}$$

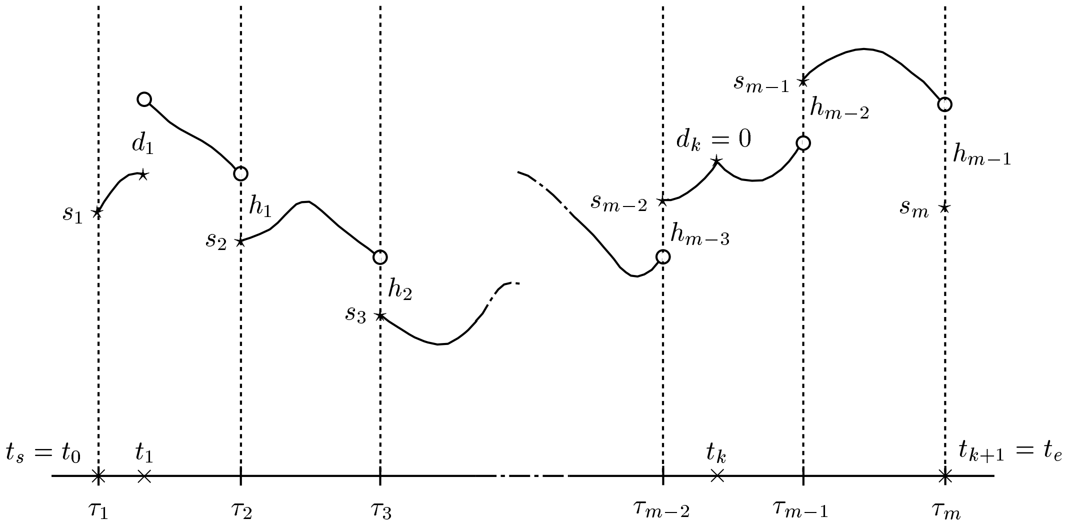
so erhält man eine auf dem ganzen Intervall definierte, in den Stützstellen τ_j aber i.a. *unstetige* Lösung der Differentialgleichung. Sei $j(i)$ der Intervallindex zu t_i (d.h. $t_i \in I_{j(i)}$), und

$$\begin{aligned} R_i(S) &:= r_i(z(t_1;\tau_{j(1)},s_{j(1)},p),\dots,z(t_k;\tau_{j(k)},s_{j(k)},p),p) && (i \in \{1,2,3\}),\\ h_j(S) &:= z(\tau_{j+1};\tau_j,s_j,p) - s_{j+1} && (j \in \{1,\dots,m-1\}). \end{aligned} \tag{1.4.14}$$

Damit läßt sich das Problem (1.4.3–1.4.7) diskretisieren, d.h. äquivalent umformen in ein *finites Ausgleichsproblem mit Beschränkungen:*

$$\|R_1(S)\|_2 = \min, \tag{1.4.15}$$
$$R_2(S) \quad = 0, \tag{1.4.16}$$
$$R_3(S) \quad \geq 0, \tag{1.4.17}$$
$$h_j(S) \quad = 0 \qquad (j \in \{1,\dots,m-1\}). \tag{1.4.18}$$

Abbildung (1.4.19) zeigt eine fiktive Starttrajektorie für das Mehrzielverfahren und veranschaulicht die zuvor beschriebenen Zusammenhänge.



Mögliche Starttrajektorie für das Mehrzielverfahren. Abbildung (1.4.19)

Das Ausgleichsproblem (1.4.15–1.4.18) wird mit einem *verallgemeinerten* GAUSS-NEWTON-*Verfahren* iterativ gelöst. Die *Anschlußbedingungen* (1.4.18) sorgen dabei für die Stetigkeit der Lösung in den Stützstellen τ_j. Die eigentliche Diskretisierung findet während der Integration der Anfangswertprobleme (1.4.12) statt, d.h. bei der Berechnung von $z(\tau_{j+1};\tau_j,s_j,p)$ und $z(t_i;\tau_{j(i)},s_{j(i)},p)$. Die Lösung ist somit weitgehend unabhängig vom gewählten Gitter Γ, nämlich bis auf Diskretisierungs- und Rundungsfehler.

Der *Mehrzieloperator* $M := (R_1^\top, R_2^\top, R_3^\top, h_1^\top, \dots, h_{m-1}^\top)^\top$ ist durch (1.4.14) *numerisch* gegeben und hat extrem komplizierte Differenzierbarkeitseigenschaften, besonders dann, wenn in der Trajektorie z Sprünge auftreten.

Da die Anfangswertprobleme (1.4.12) bei Problemen aus der optimalen Steuerung meistens schlecht konditioniert sind, *muß* man i.a. ein Mehrzielverfahren ($m > 2$) benutzen. Das *Einfachschießverfahren* ($m = 2$) ist entweder ineffizient oder liefert überhaupt kein Ergebnis.

Bei dem in dieser Arbeit gerechneten Roboter-Steuerungsproblem treten auch für relativ feine Gitter Γ_m mit 25 Teilintervallen schwierige Stabilitätsverhältnisse auf. Zur Lösung des *linearisierten* Ausgleichsproblems wurde daher ein vom Verfasser implementierter *mehrstufiger Löser* nach BOCK ([6], Kap. 4) verwendet, der im Unterschied zur konventionellen LR-Zerlegung mit einer orthogonalen Zerlegung des linearisierten Systems arbeitet.

2. Steuerungstheorie für Roboter

In diesem Kapitel werden im Rahmen der Mechanik der Mehrkörpersysteme zunächst die allgemeinen Differentialgleichungen für Roboter hergeleitet. Für eine einfache Klasse von Zielfunktionalen werden danach das Steuerungsproblem und das assoziierte Mehrpunktrandwertproblem formuliert. Die Untersuchung der Schaltfunktionen liefert die optimale Steuerung. Man erhält das wichtige Resultat, daß diese bei Robotern nicht totalsingulär sein kann.

Schließlich wird die Theorie auf den Polarroboter als konkretes Beispiel angewandt. Im Anschluß an die Herleitung der Differentialgleichungen des dreidimensionalen Roboters folgt eine genaue Analyse der Dynamik für den Spezialfall einer rein horizontalen Bewegung. Insbesondere wird eine ausführliche Analyse der singulären Steuerungen durchgeführt.

2.1 Roboter als Mehrkörpersysteme

Im vorliegenden Abschnitt werden die wichtigsten Begriffe aus der Mechanik der Mehrkörpersysteme definiert. Für einen einfachen Fall wird der Zusammenhang zwischen LAGRANGE-Formalismus und NEWTONscher Mechanik hergestellt. Die Differentialgleichungen für Roboter kann man sodann unmittelbar in sehr allgemeiner Form aufstellen, indem man Roboter als spezielle Mehrkörpersysteme auffaßt. Für eine ausführliche Darstellung der physikalischen Theorie sei auf ARNOLD [1] verwiesen.

Wir betrachten zunächst ein *n-Teilchensystem* im $\mathbb{R}^3$, d.h. n *Punktmassen* m_i $(m_i > 0)$, die sich im $\mathbb{R}^3$ unter dem Einfluß gewisser Kräfte F_i bewegen. Jede Punktmasse definiert eine Abbildung $\xi_i\colon \mathbb{R} \to \mathbb{R}^3$, die den Ort $\xi_i(t)$ in Abhängigkeit von der Zeit t angibt. Die Bewegung des gesamten Systems läßt sich zusammenfassend als Kurve $\xi\colon \mathbb{R} \to \mathbb{R}^N$, $N := 3n$ im *Konfigurationsraum* $\mathbb{R}^N$ beschreiben.

Sei nun $K \subseteq \mathbb{R}^N$ eine k-dimensionale Untermannigfaltigkeit des Konfigurationsraums mit beliebigen *lokalen Koordinatensystemen*

$$\begin{aligned} \phi\colon W \subseteq \mathbb{R}^k &\to K \subseteq \mathbb{R}^N, \\ x &\mapsto \phi(x) = \xi. \end{aligned} \tag{2.1.1}$$

Ist die Bewegung des n-Teilchensystems auf K eingeschränkt, so spricht man von einem *System mit $N-k$ (holonomen) Zwangsbedingungen*, K heißt *Konfigurationsmannigfaltigkeit* des Systems, k ist die Anzahl seiner *Freiheitsgrade*.

Die Zwangsbedingungen (und damit auch K) sind häufig explizit gegeben in der Form $\Phi(\xi) = 0$ für $\xi \in K$. Dabei ist $\Phi\colon \mathbb{R}^N \to \mathbb{R}^{N-k}$ eine differenzierbare Abbildung mit $\operatorname{Rang} D\Phi = N - k$.

Ein *starrer Körper* ist ein System von n Punktmassen, in dem je zwei Punkte konstanten Abstand $\rho_{ij} := \|\xi_i - \xi_j\|_2$ haben. Es handelt sich also um ein System mit speziellen Zwangsbedingungen. Ein starrer Körper kann sich beliebig um seinen im Raum $\mathbb{R}^3$ frei beweglichen Schwerpunkt drehen. Seine Konfigurationsmannigfaltigkeit ist daher isomorph zum sechsdimensionalen direkten Produkt $\mathbb{R}^3 \times SO(3)$, sofern nicht alle n Punkte auf einer Geraden liegen.

Ein *Mehrkörpersystem* schließlich ist ein physikalisches System, das aus Punktmassen und starren Körpern besteht, welche untereinander wechselwirken und ggf. äußeren Kräften ausgesetzt sind. Die Konfigurationsmannigfaltigkeit eines Mehrkörpersystems ist eine durch weitere Zwangsbedingungen definierte Untermannigfaltigkeit des direkten Produkts der einzelnen Konfigurationsmannigfaltigkeiten.

Die einzelnen Teile eines Industrieroboters lassen sich in guter Näherung als starre Körper auffassen, die durch Gelenke verbunden sind und auf diese Weise wechselwirken. Punktmassen sind dabei die einzelnen Atome. Mit Antriebsmotoren kann man äußere Kräfte auf den Roboter ausüben und so den Bewegungsablauf steuern.

Im folgenden soll nun der LAGRANGE-Formalismus für Mehrkörpersysteme rekapituliert werden. Die *kinetische Energie* ist

$$T(\dot{x}, x) := \sum_{i=1}^{n} \frac{m_i}{2} \|\dot{\xi}_i\|_2^2 = \frac{1}{2} \dot{x}^\top A(x) \dot{x} \tag{2.1.2}$$

mit der positiv definiten *Trägheitsmatrix*

$$A(x) := D\phi(x)^\top \begin{pmatrix} \overline{m}_1 & & 0 \\ & \ddots & \\ 0 & & \overline{m}_n \end{pmatrix} D\phi(x) \tag{2.1.3}$$

(vgl. 2.1.1). Dabei sei $\overline{m}_i := m_i E_3$ das m_i-fache der (3×3)-Einheitsmatrix.

Sei weiter $V(x,t)$ die *potentielle Energie* des Mehrkörpersystems. Damit definiert man die für alles weitere grundlegende LAGRANGE-*Funktion*

$$L(\dot{x}, x, t) := T(\dot{x}, x) - V(x, t). \tag{2.1.4}$$

An dieser Stelle sei darauf hingewiesen, daß in der Praxis meist nur die Koordinaten x mit zugehöriger Trägheitsmatrix A und potentieller Energie V bekannt sind. Dagegen sind die einzelnen Punktmassen m_i (z.B. Atome) unbekannt. Aufgrund ihrer immensen Anzahl $n \approx 10^{27}$ wäre es aber sowieso nicht möglich, mit dem Konfigurationsraum $\mathbb{R}^N$ und den Koordinatensystemen ϕ zu hantieren. Diese sind hauptsächlich für theoretische Untersuchungen von Nutzen. Für die tatsächliche mathematische Beschreibung eines Mehrkörpersystems stellt man mit Hilfe *makroskopischer* Größen die LAGRANGE-Funktion (2.1.4) auf, die mit T und V alle wesentlichen Informationen enthält.

Für alle Bewegungen mit Startpunkt $x(t_s) = x_s$ und Endpunkt $x(t_e) = x_e$ definiert man nun weiter das *Wirkungsintegral*

$$S(x) := \int_{t_s}^{t_e} L(\dot{x}(t), x(t), t)\, dt. \tag{2.1.5}$$

In einem geeigneten Funktionenraum ist S FRÉCHET-differenzierbar. Die tatsächliche Bewegung des Mehrkörpersystems kann damit *global* und *koordinatenunabhängig* charakterisiert werden durch

$$DS(x) = \frac{\delta}{\delta x} \int_{t_s}^{t_e} L(\dot{x}(t), x(t), t)\, dt \overset{!}{=} 0. \tag{2.1.6}$$

S ist also in x stationär, i.a. sogar minimal. Man bezeichnet daher (2.1.6) als HAMILTON-*Prinzip der kleinsten Wirkung.*

Wertet man die Variationsableitung in (2.1.6) aus, so erhält man als äquivalente *lokale* Beschreibung des Mehrkörpersystems die ebenfalls *koordinatenunabhängige* EULER-LAGRANGE-*Gleichung*

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x} = 0, \tag{2.1.7}$$

die für die praktische und insbesondere numerische Behandlung benötigt wird.

In diesem allgemeinen Formalismus ist die klassische NEWTONsche Mechanik als Spezialfall enthalten. An einem einfachen Beispiel wird dies sofort deutlich. Für eine einzelne Punktmasse m in $K = \mathbb{R}^3$ (mit $\xi = x$) gilt nämlich

$$L(\dot{x}, x, t) = \frac{m}{2} \|\dot{x}\|_2^2 - V(x, t). \tag{2.1.8}$$

Auf m wirkt die Kraft $F = -V_x^\top = L_x^\top$, der Impuls ist $p = m\dot{x} = L_{\dot{x}}^\top$. Die EULER-LAGRANGE-Gleichung (2.1.7) ist also identisch mit der klassischen NEWTONschen Bewegungsgleichung

$$m\ddot{x} = F. \tag{2.1.9}$$

Man nennt deshalb die x_i *verallgemeinerte Koordinaten*, $\dot{x}_i$ *verallgemeinerte Geschwindigkeiten*, $\partial L / \partial \dot{x}_i$ *verallgemeinerte Impulse* und $\partial L / \partial x_i$ *verallgemeinerte Kräfte.*

Im folgenden betrachten wir einen Roboter mit n-dimensionaler Konfigurationsmannigfaltigkeit K. Man muß davon ausgehen, daß jede Koordinate x_i von einer (verallgemeinerten) äußeren Kraft $\bar{u}_i$ beeinflußt wird. Das Potential ist dann

$$\begin{aligned} V_{\text{tot}}(x,t) &= V_{\text{int}}(x) + V_{\text{ext}}(x,t) \\ &= V(x) - \bar{u}(t)^\top x. \end{aligned} \tag{2.1.10}$$

Der rein ortsabhängige Anteil V_{int} ist das eigentliche Potential des Roboters, V_{ext} berücksichtigt die äußeren Einflüsse, welche von jetzt an explizit behandelt werden sollen. Dazu zerlegt man die LAGRANGE-Funktion entsprechend:

$$\begin{aligned} L_{\text{tot}}(\dot{x},x,t) &= L_{\text{int}}(\dot{x},x) + L_{\text{ext}}(x,t) \\ &= L(\dot{x},x) + \bar{u}(t)^\top x. \end{aligned} \tag{2.1.11}$$

Aus der *autonomen* LAGRANGE-Funktion

$$L(\dot{x},x) = T(\dot{x},x) - V_{\text{int}}(x) = \frac{1}{2}\dot{x}^\top A(x)\dot{x} - V(x) \tag{2.1.12}$$

des *antriebslosen* Roboters erhält man dann die modifizierte EULER-LAGRANGE-Gleichung

$$\begin{aligned} \bar{u} &= \frac{d}{dt}L_{\dot{x}}(\dot{x},x)^\top - L_x(\dot{x},x)^\top \\ &= \frac{d}{dt}(A(x)\dot{x}) - (T_x(\dot{x},x) - V_x(x))^\top \\ &= A(x)\ddot{x} + (A(x)\dot{x})_x\dot{x} - \left(\frac{1}{2}(\dot{x}^\top A(x)\dot{x})_x - V_x(x)\right)^\top \end{aligned} \tag{2.1.13}$$

mit den Komponenten

$$\begin{aligned} \bar{u}_i &= \sum_{j=1}^{n} A_{ij}\ddot{x}_j + \sum_{j,k=1}^{n} \frac{\partial(A_{ij}\dot{x}_j)}{\partial x_k}\dot{x}_k - \frac{1}{2}\sum_{j,k=1}^{n}\frac{\partial(\dot{x}_j A_{jk}\dot{x}_k)}{\partial x_i} + \frac{\partial V}{\partial x_i} \\ &= \sum_{j=1}^{n}\left[A_{ij}\ddot{x}_j + \sum_{k=1}^{n}\left(\frac{\partial A_{ij}}{\partial x_k} - \frac{1}{2}\frac{\partial A_{jk}}{\partial x_i}\right)\dot{x}_j\dot{x}_k\right] + \frac{\partial V}{\partial x_i}. \end{aligned}$$

Als verallgemeinerte Koordinaten werden meist *Längen* und *Winkel* verwendet, die ein *globales* Koordinatensystem von K bilden. Die entsprechenden verallgemeinerten Geschwindigkeiten, Impulse, Kräfte, und Steuergrößen sowie die Diagonalelemente der Trägheitsmatrix sind in Tabelle (2.1.14) zusammengestellt.

x_i	$\dot{x}_i$	$\partial L/\partial \dot{x}_i$	$\partial L/\partial x_i,\ u_i$	A_{ii}
Länge	Geschwindigkeit	Impuls	Kraft	Masse
Winkel	Winkelgeschwindigkeit	Drehimpuls	Drehmoment	Trägheitsmoment

Typische verallgemeinerte Koordinaten bei Robotern. Tabelle (2.1.14)

Es werde nun vorausgesetzt, daß die äußeren Kräfte $\bar{u}_i$ gerade die Steuerungen sind, der Roboter also für jeden Freiheitsgrad einen eigenen Antrieb besitzt. Steuerbereich sei der nichtentartete Quader $[-\hat{u}_1,+\hat{u}_1]\times\cdots\times[-\hat{u}_n,+\hat{u}_n]$. Mit $D := \operatorname{diag}(\hat{u}_1,\ldots,\hat{u}_n)$ und $u := D^{-1}\bar{u}$ erhält man für u den transformierten Steuerbereich $U := [-1,+1]^n$. Damit nehmen die Robotergleichungen folgende Form an:

$$A(x)\ddot{x} = T_x(\dot{x},x)^\top - V_x(x)^\top - (A(x)\dot{x})_x\dot{x} + Du \tag{2.1.15}$$

$$=: B(\dot{x},x) + Du, \tag{2.1.16}$$

$$\ddot{x} = A(x)^{-1}(B(\dot{x},x) + Du). \tag{2.1.17}$$

Die spezielle Struktur der Funktion $B\colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ ist für die weiteren Ausführungen irrelevant. Daher können in B Korrekturterme aufgenommen werden, um ggf. Abweichungen vom idealisierten Mehrkörpermodell zu berücksichtigen. Auch Reibungseffekte (sie sind geschwindigkeitsabhängig!) lassen sich auf diese Weise behandeln.

2.2 Formulierung des Steuerungsproblems für den Roboter

Sei $\epsilon := \operatorname{diag}(\epsilon_1, \dots, \epsilon_n)$ eine positiv semidefinite Diagonalmatrix, $\langle x, y\rangle_\epsilon := \langle x, \epsilon y\rangle = x^\top \epsilon y$ das durch ϵ definierte Semiskalarprodukt und $\|x\|_\epsilon := \sqrt{\langle x, x\rangle_\epsilon}$ die zugehörige Seminorm. Wir betrachten das Zielfunktional

$$\int_{t_s}^{t_e} \left(1 + \frac{1}{2}\|u(t)\|_\epsilon^2\right) dt = \min \tag{2.2.1}$$

vom LAGRANGE-Typ. Der Term $\int \|u\|_\epsilon^2$ hat keine unmittelbare physikalische Bedeutung, steht jedoch in engem Zusammenhang mit der vom Antrieb verbrauchten Energie. Durch geeignete Wahl der ϵ_i kann man den Einfluß jeder einzelnen Steuerung auf das Zielfunktional separat gewichten. Von besonderem Interesse ist der Spezialfall $\epsilon = 0$, der das MAYER-Zielfunktional $t_e - t_s = \min$ für eine *rein zeitoptimale* Steuerung liefert.

Man führt nun zunächst $z_1 := \dot{x}$ und $z_2 := x$ als neue Zustandsvariablen ein, um aus (2.1.17) ein System erster Ordnung zu erhalten. Mit der zusätzlichen Variablen z_3 transformiert man sodann das Zielfunktional (2.2.1) auf MAYER-Form:

$$z_3(t_e) = \min\,. \tag{2.2.2}$$

Das Differentialgleichungssystem lautet damit

$$\begin{aligned} \dot{z}_1 &= A(z_2)^{-1}(B(z_1, z_2) + Du) \\ \dot{z}_2 &= z_1 \\ \dot{z}_3 &= 1 + \frac{1}{2}\|u\|_\epsilon^2 \end{aligned} \tag{2.2.3}$$

und hat die Dimension $n_d := 2n + 1$.

Im folgenden werde davon ausgegangen, daß der Roboter aus einer bekannten Startposition $(x_s, \dot{x}_s)$ ein vorgegebenes Ziel $(x_e, \dot{x}_e)$ ansteuern soll, wie es in der Praxis häufig vorkommt. Da sowohl die Differentialgleichungen (2.2.3) als auch das Zielfunktional (2.2.2) und die Randbedingungen autonom sind, d.h.

$$\frac{\partial f}{\partial t} = 0, \qquad \frac{\partial r_1}{\partial t_s} = 0 = \frac{\partial r_1}{\partial t_e}, \qquad \frac{\partial r_2}{\partial t_s} = 0 = \frac{\partial r_2}{\partial t_e}, \tag{2.2.4}$$

ist das Steuerungsproblem invariant gegenüber Zeittranslationen. Durch die Minimierungsbedingung (2.2.1) bzw. (2.2.2) ist nur die Differenz $T := t_e - t_s$ bestimmt. Man setzt deshalb formal $t_s := 0$ und erhält mit den Randbedingungen

$$\begin{array}{ll} t_s \quad = 0 & \\ z_1(t_s) - \dot{x}_s = 0 & z_1(t_e) - \dot{x}_e = 0 \\ z_2(t_s) - x_s = 0 & z_2(t_e) - x_e = 0 \\ z_3(t_s) \quad = 0 & \end{array} \tag{2.2.5}$$

das komplette Steuerungsproblem in der Form (1.1.1–1.1.3). Ungleichungsbedingungen werden hier nicht benötigt. Zustandsbeschränkungen lassen sich erst formulieren, wenn man ein konkretes Problem zugrundelegt, denn sie sind vom verwendeten Robotertyp abhängig, ggf. auch von dessen Umgebung sowie vom gerade auszuführenden Arbeitsgang.

Das Steuerungsproblem kann jetzt in das assoziierte Mehrpunktrandwertproblem transformiert werden. Aus der HAMILTON-Funktion

$$\begin{aligned} H(t, z, \lambda, u) &= \lambda_1^\top \dot{z}_1 + \lambda_2^\top \dot{z}_2 + \lambda_3 \dot{z}_3 \\ &= \lambda_1^\top A(z_2)^{-1}(B(z_1, z_2) + Du) + \lambda_2^\top z_1 + \lambda_3\left(1 + \frac{1}{2}\|u\|_\epsilon^2\right) \end{aligned} \tag{2.2.6}$$

leitet man die adjungierten Differentialgleichungen ab:

$$\begin{aligned}
\dot{\lambda}_1^\top &= -\lambda_1^\top \left[A(z_2)^{-1}(B(z_1,z_2)+Du)\right]_{z_1} - \lambda_2^\top [z_1]_{z_1} - \lambda_3 \Big[1+\frac{1}{2}\|u\|_\epsilon^2\Big]_{z_1} \\
&= -\lambda_1^\top A(z_2)^{-1} B_{z_1}(z_1,z_2) - \lambda_2^\top, \qquad (2.2.7)
\end{aligned}$$

$$\begin{aligned}
\dot{\lambda}_2^\top &= -\lambda_1^\top \left[A(z_2)^{-1}(B(z_1,z_2)+Du)\right]_{z_2} - \lambda_2^\top [z_1]_{z_2} - \lambda_3 \Big[1+\frac{1}{2}\|u\|_\epsilon^2\Big]_{z_2} \\
&= -\lambda_1^\top \left[A(z_2)^{-1} B_{z_2}(z_1,z_2) - A(z_2)^{-1} A_{z_2}(z_2) A(z_2)^{-1}(B(z_1,z_2)+Du)\right] \\
&= -\lambda_1^\top A(z_2)^{-1}[B_{z_2}(z_1,z_2) - A_{z_2}(z_2)\dot{z}_1], \qquad (2.2.8)
\end{aligned}$$

$$\begin{aligned}
\dot{\lambda}_3 &= -\lambda_1^\top \left[A(z_2)^{-1}(B(z_1,z_2)+Du)\right]_{z_3} - \lambda_2^\top [z_1]_{z_3} - \lambda_3 \Big[1+\frac{1}{2}\|u\|_\epsilon^2\Big]_{z_3} \\
&= 0. \qquad (2.2.9)
\end{aligned}$$

Das erweiterte Zielfunktional hat die Form

$$\begin{aligned}
G(t_s,z_s,t_e,z_e) = \alpha_s t_s &+ \alpha_{1s}^\top (z_{1s} - \dot{x}_s) + \alpha_{2s}^\top (z_{2s} - x_s) + \alpha_{3s} z_{3s} \\
&+ \alpha_{1e}^\top (z_{1e} - \dot{x}_e) + \alpha_{2e}^\top (z_{2e} - x_e) + \alpha_{3e} z_{3e}.
\end{aligned} \qquad (2.2.10)$$

Dem Term $\alpha_1 r_1$ aus (1.3.1.7) entspricht hier der letzte Summand $\alpha_{3e} z_{3e}$, der das Zielfunktional (2.2.2) berücksichtigt. Die übrigen Summanden koppeln die Randbedingungen (2.2.5) an, sie entsprechen zusammen dem Term $\alpha_2^\top r_2$ aus (1.3.1.7). Die Transversalitätsbedingungen lauten damit:

$$\begin{aligned}
\lambda_1(t_s) &= \alpha_{1s} & \lambda_1(t_e) &= -\alpha_{1e} \\
\lambda_2(t_s) &= \alpha_{2s} & \lambda_2(t_e) &= -\alpha_{2e} \\
\lambda_3(t_s) &= \alpha_{3s} & \lambda_3(t_e) &= -\alpha_{3e} \le 0\,! \\
\tilde{H}(t_s) &= \alpha_s & \tilde{H}(t_e) &= \quad 0
\end{aligned} \qquad (2.2.11)$$

Offensichtlich existieren α's, die diese „Bedingungen“ erfüllen, denn außer (2.2.11) wird nichts gefordert. Als einzige echte Bedingung bleibt also $\tilde{H}(t_e) = 0$; diese ersetzt gerade die Minimierungsbedingung $z_3(t_e) = \min$. Es werde nun vorausgesetzt, daß das Randwertproblem nicht entartet ist, also $\alpha_{3e} \neq 0$. Andernfalls ist nichts zu optimieren, beispielsweise, weil der Antrieb so schwach ist, daß das vorgegebene Ziel $(x_e, \dot{x}_e)$ gerade noch erreicht werden kann. Die Voraussetzung $\alpha_{3e} \neq 0$ ist also physikalisch sinnvoll. Die noch offene Skalierung von α und λ werde dann o.B.d.A. durch $\alpha_{3e} := 1 > 0$ festgelegt. Wegen (2.2.9) und (2.2.11) ist $\lambda_3 = -1$ konstant und kann aus der Hamilton-Funktion eliminiert werden:

$$H(t,z,\lambda,u) = \lambda_1^\top A(z_2)^{-1}(B(z_1,z_2)+Du) + \lambda_2^\top z_1 - \Big(1+\frac{1}{2}\|u\|_\epsilon^2\Big). \qquad (2.2.12)$$

Bemerkung (2.2.13): Wegen der Autonomie des Systems (2.2.3) ist H zeitlich konstant (vgl. 1.3.2.27), verschwindet also nach (2.2.11) identisch: $\tilde{H}(t) = 0\ \forall t \in I\,!$

In der hier gewählten Formulierung liegt das Randwertproblem in einer sehr klaren Form vor. Selbstverständlich kann man aber auch statt (2.2.1) das Bolza-Zielfunktional

$$t_e - t_s + \frac{1}{2}\int_{t_s}^{t_e} \|u(t)\|_\epsilon^2\, dt = \min \qquad (2.2.14)$$

mit dem Zeitterm in Mayer-Form und dem Energieterm in Lagrange-Form verwenden. Beide Formulierungen sind mathematisch weitgehend äquivalent, d.h., sie besitzen dieselben Lösungen. Die Transformation (1.1.8) führt aber im letzteren Fall zu folgenden Unterschieden: Statt (2.2.2) erhält man die Minimierungsbedingung

$$t_e - t_s + z_{3e} = \min, \qquad (2.2.15)$$

die Differentialgleichung für z_3 ist statt (2.2.3)

$$\dot{z}_3(t) = \frac{1}{2}\|u(t)\|_\epsilon^2. \qquad (2.2.16)$$

Anstelle der HAMILTON-Funktion (2.2.6) erhält man

$$\begin{aligned} H(t,z,\lambda,u) &= \lambda_1^\top \dot z_1 + \lambda_2^\top \dot z_2 + \lambda_3 \dot z_3 \\ &= \lambda_1^\top A(z_2)^{-1}(B(z_1,z_2)+Du) + \lambda_2^\top z_1 + \frac{1}{2}\lambda_3 \|u\|_\epsilon^2, \end{aligned} \tag{2.2.17}$$

das erweiterte Zielfunktional (2.2.10) lautet nun

$$\begin{aligned} G(t_s,z_s,t_e,z_e) = \alpha_s t_s &+ \alpha_{1s}^\top(z_{1s}-\dot x_s) + \alpha_{2s}^\top(z_{2s}-x_s) + \alpha_{3s} z_{3s} \\ &+ \alpha_{1e}^\top(z_{1e}-\dot x_e) + \alpha_{2e}^\top(z_{2e}-x_e) + \alpha_{3e}(t_e - t_s + z_{3e}). \end{aligned} \tag{2.2.18}$$

Schließlich ändern sich noch die Randbedingungen (2.2.11) für $\tilde H$:

$$\tilde H(t_s) = \alpha_s - \alpha_{3e}, \qquad \tilde H(t_e) = -\alpha_{3e}. \tag{2.2.19}$$

Man kann sie zusammenfassen zu

$$\tilde H(t_s) - \tilde H(t_e) = \alpha_s. \tag{2.2.20}$$

Auch hier ist $\tilde H$ konstant, verschwindet aber i.a. nicht.

2.3 Berechnung der optimalen Steuerung aus dem Maximumprinzip

In den meisten Fällen ist es relativ einfach, durch Maximierung der HAMILTON-Funktion die optimale Steuerung für den Roboter zu bestimmen. Aus (2.2.12) erhält man

$$\begin{aligned} H_u(z,\lambda,u) &= \lambda_1^\top A(z_2)^{-1} D - u^\top \epsilon \\ &=: Q(z,\lambda)^\top - u^\top \epsilon, \end{aligned} \tag{2.3.1}$$

$$H_{uu}(z,\lambda,u) = -\epsilon. \tag{2.3.2}$$

Die HAMILTON-Funktion ist also negativ semidefinit bzgl. u. Für jede einzelne Steuerungskomponente u_i ist H *entweder* negativ definit ($\epsilon_i > 0$) *oder* linear ($\epsilon_i = 0$). Im linearen Fall ist $Q_i = \tilde Q_i$ eine Komponente des in (1.3.2.15) definierten Schaltvektors, kann also unmittelbar als Schaltfunktion für u_i benutzt werden. Mit der üblichen Abkürzung $q(t) := Q(z(t),\lambda(t))$ liefert die Maximierung der HAMILTON-Funktion:

$$u_i(t) := \begin{cases} -1, & \text{falls } q_i(t) \in \,]-\infty, -\epsilon_i[, \\ \epsilon_i^{-1} q_i(t), & \text{falls } q_i(t) \in [-\epsilon_i, +\epsilon_i] \text{ und } \epsilon_i \neq 0, \\ +1, & \text{falls } q_i(t) \in \,]+\epsilon_i, +\infty[. \end{cases} \tag{2.3.3}$$

Mit den $2n$ eigentlichen Schaltfunktionen

$$\begin{aligned} Q_i^\pm(z,\lambda) &:= Q_i(z,\lambda) \pm \epsilon_i \\ q_i^\pm(t) &:= Q_i^\pm(z(t),\lambda(t)) \end{aligned} \qquad (i \in \{1,\dots,n\}) \tag{2.3.4}$$

lautet die Formulierung

$$u_i(t) = \begin{cases} -1, & \text{falls } q_i^+(t) < 0, \\ \epsilon_i^{-1} q_i(t), & \text{falls } q_i^-(t) \le 0 \le q_i^+(t) \text{ und } \epsilon_i \neq 0, \\ +1, & \text{falls } 0 < q_i^-(t). \end{cases} \tag{2.3.5}$$

Die Steuerungen $u_i(t)$ für $\epsilon_i > 0$ sind damit vollständig bestimmt. Sie sind in jedem Falle stetig und darüberhinaus entweder konstant ($|u_i| = 1$) oder genauso oft differenzierbar wie q_i ($u_i = q_i/\epsilon_i$). In den Nullstellen von $q_i^\pm$ treten i.a. „Knicke" auf, an anderen Stellen ist dies nicht möglich. Singuläre Steuerungen sind ausgeschlossen.

Für $\epsilon_i = 0$ ist $H_{u_i}(z,\lambda,u) = Q_i(z,\lambda)$ und $Q_i^+ = Q_i^- = Q_i$. Nach (2.3.3/2.3.5) erhält man eine bang-bang-Steuerung $u_i(t) = \operatorname{sign} q_i(t)$ mit den isolierten Nullstellen von q_i als Schaltstellen. In diesen Punkten ist u_i unstetig, ansonsten konstant.

Falls q_i auf einem kompletten Intervall $I_S = [t_a, t_b]$ verschwindet, ist die Steuerung u_i dort singulär. Dies ist der einzige Fall, über den (2.3.3/2.3.5) nichts aussagen; zur Berechnung von u_i muß man dann die Methoden aus Abschnitt (1.3.2) heranziehen.

Im Extremfall $\epsilon = 0$ kann im Prinzip jede Steuerungskomponente singulär sein. Der folgende Satz zeigt jedoch, daß zu jedem Zeitpunkt mindestens eine nichtsinguläre Steuerung existiert. Physikalisch ausgedrückt: mindestens ein Antrieb läuft auf vollen Touren.

Satz (2.3.6): Eine zeitoptimale Roboter-Steuerung kann nicht totalsingulär sein.

Beweis: Alle Steuerungen u_i seien gleichzeitig singulär. Auf I_S gilt dann

$$\lambda_1(t)^\top A(z_2(t))^{-1} D = q(t)^\top \overset{!}{=} 0, \tag{2.3.7}$$

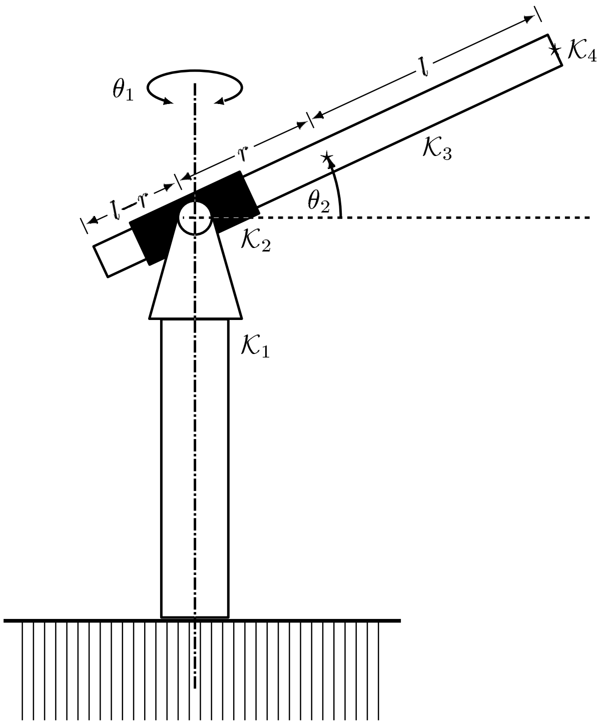
wegen der Regularität von $A(z_2(t))^{-1}D$ also auch $\lambda_1(t) = 0$. Es folgt weiter $\dot{\lambda}_1(t) = 0$. Setzt man beides in die adjungierte Differentialgleichung (2.2.7) ein, so ergibt dies unmittelbar $\lambda_2(t) = 0$. Damit liefert (2.2.6) die HAMILTON-Funktion

$$\tilde{H}(t) = \lambda_3(t). \tag{2.3.8}$$

Diese verschwindet nach Bemerkung (2.2.13) auf ganz I, also insbesondere auf I_S; es folgt $\lambda_3(t) = 0$. Damit ist aber $\lambda|_{I_S} = 0$, eine notwendige Bedingung für die Optimalität von u ist verletzt. ∎

2.4 Der Polarroboter

Die bisher gewonnenen theoretischen Ergebnisse werden jetzt auf einen gängigen Robotertyp angewandt, den sogenannten *Polarroboter* (Abbildung 2.4.1).



Skizze des Polarroboters. Abbildung (2.4.1)

Er besteht im wesentlichen aus einem ausfahrbaren Manipulatorarm mit Greifhand, der in einer Führungsmanschette kippbar auf einem drehbaren Stativ gelagert ist. Die beiden Achsen des Roboters schneiden sich im *Drehpunkt*. Die Greifhand mit Last an der Spitze des Manipulatorarms werde als Massenpunkt betrachtet und kurz als „Last" bezeichnet. Das Mehrkörpersystem als vereinfachtes Modell des Polarroboters besteht dann aus drei starren Körpern und einer Punktmasse, nämlich

- dem ortsfesten, drehbaren Stativ $\mathcal{K}_1$ der Masse m_1,
- der kippbaren Führungsmanschette $\mathcal{K}_2$ mit Masse m_2,
- dem ausfahrbaren Manipulatorarm $\mathcal{K}_3$ der Masse $m_3 = m_B$,
- der punktförmigen Last $\mathcal{K}_4$ mit Masse $m_4 = m_L$.

Die Geometrie des Polarroboters bestimmt eindeutig den *Arbeitsbereich* (also alle Punkte im Raum, die die Greifhand erreichen kann) und damit auch die Konfigurationsmannigfaltigkeit. Außerdem wird dadurch die Benutzung spezieller Koordinaten nahegelegt, die im folgenden beschrieben werden.

Sei zunächst $2l$ die Länge des Manipulatorarms und $\bar{r} \in [0, 2l]$ der Abstand der Last vom Drehpunkt. Aus Symmetriegründen benutzen wir als Koordinate nicht $\bar{r}$, sondern den Abstand $r := \bar{r} - l \in [-l, +l]$ des Armmittelpunkts vom Drehpunkt.

Der *Horizontalwinkel* θ_1 beschreibt die Drehung des gesamten Roboters um die vertikale Achse; er ist keinen Beschränkungen unterworfen. Dies hat gegenüber der Normierung auf ein festes Intervall (z.B. $[0, 2\pi[$ oder $]-\pi, +\pi]$) zwei Vorteile: erstens springt θ_1 nicht beim Erreichen des Intervallrandes, ist also stetig, und zweitens wird für jede Bewegung des Roboters automatisch die Anzahl der Drehungen um die vertikale Achse mitgezählt.

Der *Vertikalwinkel* θ_2 schließlich gibt die Neigung des Manipulatorarms gegenüber der Horizontalen an. Diese ist beschränkt durch das Stativ, eventuell auch durch Boden und/oder Decke des Raums, in dem der Roboter steht. θ_2 liege im Intervall $[\theta_2^-, \theta_2^+] \subset]-\pi/2, +\pi/2[$.

Als natürliche Zustandsbeschränkungen für den Polarroboter ergeben sich:

$$\begin{aligned} g_1(r) &:= r^2 - l^2 \leq 0, \\ g_2(\theta_2) &:= \theta_2^- - \theta_2 \leq 0, \\ g_3(\theta_2) &:= \theta_2 - \theta_2^+ \leq 0. \end{aligned} \tag{2.4.2}$$

Die Konfigurationsmannigfaltigkeit kann man identifizieren mit dem (globalen) Koordinatenbereich:

$$K = [-l, +l] \times I\!R \times [\theta_2^-, \theta_2^+] \subset I\!R^3. \tag{2.4.3}$$

Sie ist dreidimensional und *berandet* durch die Zustandsbeschränkungen (2.4.2).

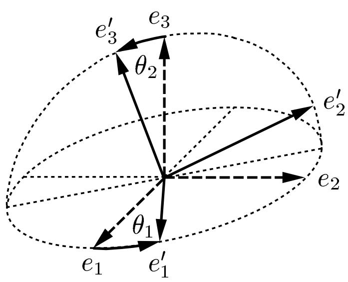
Die Koordinaten (r, θ_1, θ_2) werden als *Polarkoordinaten* bezeichnet; ihnen verdankt der Polarroboter seinen Namen.

Der Arbeitsbereich hat die Form einer Kugel vom Radius l um den Drehpunkt, aus der ein vertikaler Doppelkegel (mit Spitze im Drehpunkt) herausgeschnitten ist. Befindet sich die Greifhand im Mittelpunkt ($r = 0$), so kommt es auf ihre Orientierung an, die durch die Winkel θ_1 und θ_2 berücksichtigt wird. Diese Information fehlt bei einer reinen Ortsangabe ebenso wie die Umdrehungszahl. Daher kann man den Arbeitsbereich nicht unmittelbar mit der Konfigurationsmannigfaltigkeit identifizieren.

Um die Drehbewegung der Roboterteile $\mathcal{K}_1$–$\mathcal{K}_4$ zu beschreiben, betrachten wir zusätzlich zu den bereits erläuterten Polarkoordinaten noch zwei kartesische Koordinatensysteme, deren gemeinsamer Ursprung im Drehpunkt liegt.

Das erste Koordinatensystem $B := [e_1, e_2, e_3]$ ist *ortsfest*. Der Basisvektor e_3 zeigt in Richtung der vertikalen Achse nach oben, e_1 als Bezugsrichtung für θ_1 kann in der horizontalen Ebene beliebig gewählt werden.

Das zweite Koordinatensystem $B' := [e_1', e_2', e_3']$ ist *körperfest*, dreht sich also mit dem Roboterarm gegenüber B. e_1' hat die Richtung der Kippachse für den Manipulatorarm, e_2' zeigt in Richtung des Arms, so daß sich dessen Mittelpunkt am Ort re_2' und die Last am Ort $\bar{r}e_2' = (r + l)e_2'$ befinden. Die Lage der Koordinatensysteme zueinander zeigt Abbildung (2.4.4).



Ortsfestes Koordinatensystem (e_i) und körperfestes Koordinatensystem (e_i'). Abbildung (2.4.4)

Die Basis B kann man durch eine Drehung $R \in SO(3)$ in die Basis B' überführen. R läßt sich darstellen als Komposition einer Rotation um die Achse e_1 mit Drehwinkel θ_2

$$R(e_1,\theta_2) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_2 & -\sin\theta_2 \\ 0 & \sin\theta_2 & \cos\theta_2 \end{pmatrix} \tag{2.4.5}$$

sowie einer Rotation um e_3 mit Drehwinkel θ_1

$$R(e_3,\theta_1) = \begin{pmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.4.6}$$

und hat die Gestalt

$$R = R(e_3,\theta_1)R(e_1,\theta_2) = \begin{pmatrix} \cos\theta_1 & -\sin\theta_1\cos\theta_2 & \sin\theta_1\sin\theta_2 \\ \sin\theta_1 & \cos\theta_1\cos\theta_2 & -\cos\theta_1\sin\theta_2 \\ 0 & \sin\theta_2 & \cos\theta_2 \end{pmatrix}. \tag{2.4.7}$$

Alle Matrizen sind dabei bezüglich der ortsfesten Basis B dargestellt, den Vektor $e_i' = Re_i$ findet man also in der i-ten Spalte von R. In den Zeilen sind entsprechend die Basisvektoren $e_i = R^\top e_i'$ bezüglich der körperfesten Basis B' dargestellt.

Aus der zweiten Spalte der Drehmatrix R erhält man z.B. die zeitliche Ableitung

$$\dot{e}_2' = \begin{pmatrix} -\dot\theta_1\cos\theta_1\cos\theta_2 + \dot\theta_2\sin\theta_1\sin\theta_2 \\ -\dot\theta_1\sin\theta_1\cos\theta_2 - \dot\theta_2\cos\theta_1\sin\theta_2 \\ \dot\theta_2\cos\theta_2 \end{pmatrix} \tag{2.4.8}$$

in ortsfesten Koordinaten. Die entsprechende Darstellung in körperfesten Koordinaten lautet

$$R^\top \begin{pmatrix} -\dot\theta_1\cos\theta_1\cos\theta_2 + \dot\theta_2\sin\theta_1\sin\theta_2 \\ -\dot\theta_1\sin\theta_1\cos\theta_2 - \dot\theta_2\cos\theta_1\sin\theta_2 \\ \dot\theta_2\cos\theta_2 \end{pmatrix} = \begin{pmatrix} -\dot\theta_1\cos\theta_2 \\ 0 \\ \dot\theta_2 \end{pmatrix} = -\dot\theta_1\cos\theta_2 e_1' + \dot\theta_2 e_3'. \tag{2.4.9}$$

Damit ist die Geschwindigkeit des Armmittelpunkts

$$\frac{d}{dt}(re_2') = \dot r e_2' + r\dot e_2' = \underbrace{\dot r e_2'}_{\text{Radial-}} + \underbrace{r\big(-\dot\theta_1\cos\theta_2 e_1' + \dot\theta_2 e_3'\big)}_{\text{Tangentialkomponente}}, \tag{2.4.10}$$

und analog die Geschwindigkeit der Last:

$$\frac{d}{dt}\big((r+l)e_2'\big) = \dot r e_2' + (r+l)\big(-\dot\theta_1\cos\theta_2 e_1' + \dot\theta_2 e_3'\big). \tag{2.4.11}$$

Bevor wir mit der Aufstellung der LAGRANGE-Funktion aus kinetischer und potentieller Energie fortfahren, werden noch einige physikalische Begriffe benötigt.

Für eine Drehbewegung mit Winkelgeschwindigkeit $\dot\theta$ um eine Achse e ($\|e\|_2 = 1$) definiert man den *Vektor der Winkelgeschwindigkeit*

$$\omega := \dot\theta e. \tag{2.4.12}$$

Überlagern sich mehrere Drehbewegungen, so kann man deren vektorielle Winkelgeschwindigkeiten einfach addieren, sofern sich alle Achsen in einem gemeinsamen Drehpunkt schneiden. Der Polarroboter z.B. dreht sich mit $\omega_1 := \dot\theta_1 e_3 = \dot\theta_1(\sin\theta_2 e_2' + \cos\theta_2 e_3')$ um die vertikale Achse, sein Manipulatorarm rotiert mit der Winkelgeschwindigkeit $\omega_2 := \dot\theta_2 e_1'$ um die horizontale Achse. Die resultierende Winkelgeschwindigkeit des Roboterarms (genauer: des Vektors e_2') ist damit

$$\omega := \omega_1 + \omega_2 = \dot\theta_2 e_1' + \dot\theta_1\sin\theta_2 e_2' + \dot\theta_1\cos\theta_2 e_3'. \tag{2.4.13}$$

Die Winkelgeschwindigkeit ω eines starren Körpers $\mathcal{K}$ benötigt man, um dessen Rotationsenergie zu berechnen. Dazu betrachtet man $\mathcal{K} \subset \mathbb{R}^3$ *makroskopisch* als Kompaktum, das *kontinuierlich* mit Masse der ortsabhängigen *Dichte* ρ gefüllt ist, und definiert den *Trägheitstensor* I mit den Komponenten

$$I_{jk} := \int_{\mathcal{K}} \rho(y) \left(\delta_{jk}\|y\|_2^2 - y_j y_k\right)\, d^3y \qquad (j,k \in \{1,2,3\}). \tag{2.4.14}$$

Die Rotationsenergie von $\mathcal{K}$ ist für $\omega = \dot\theta e$

$$\begin{aligned} T := \frac{1}{2}\omega^\top I \omega &= \frac{1}{2}\int_{\mathcal{K}} \rho(y) \left(\|y\|_2^2\|\omega\|_2^2 - \langle y,\omega\rangle^2\right)\, d^3y \\ &= \frac{\dot\theta^2}{2}\int_{\mathcal{K}} \rho(y) \left(\|y\|_2^2 - \langle y,e\rangle^2\right)\, d^3y, \end{aligned} \tag{2.4.15}$$

wobei $\|y\|_2^2 - \langle y,e\rangle^2$ das Quadrat des Abstands von y zur Drehachse angibt.

I ist stets symmetrisch und positiv definit, die sechs verschiedenen Komponenten enthalten sowohl Informationen über die Masseverteilung ρ als auch über Position und Orientierung von $\mathcal{K}$ im Raum. Aufgrund der Symmetrie besitzt I eine Orthonormalbasis von Eigenvektoren, diese heißen *Hauptträgheitsachsen*. Die zugehörigen Eigenwerte I_j nennt man *Hauptträgheitsmomente*.

Für einen bezüglich der Koordinatenachsen symmetrischen Körper konstanter Dichte ρ ist I diagonal. Der Quader $[-l_1,+l_1] \times [-l_2,+l_2] \times [-l_3,+l_3]$ mit der Masse $m = 8l_1l_2l_3\rho$ hat beispielsweise das Hauptträgheitsmoment

$$\begin{aligned} I_1 &= \int_{-l_1}^{+l_1}\int_{-l_2}^{+l_2}\int_{-l_3}^{+l_3} \rho\left(y_2^2 + y_3^2\right)\, dy_1\, dy_2\, dy_3 \\ &= 2l_1\rho \int_{-l_2}^{+l_2} \left[y_2^2 y_3 + \frac{y_3^3}{3}\right]_{y_3=-l_3}^{y_3=+l_3} dy_2 \\ &= 8l_1l_2l_3\rho\frac{l_2^2+l_3^2}{3} = m\frac{l_2^2+l_3^2}{3} \end{aligned} \tag{2.4.16}$$

und allgemein

$$I_j = m\frac{l_1^2+l_2^2+l_3^2-l_j^2}{3}. \tag{2.4.17}$$

Nach diesen Vorbereitungen kann die kinetische Energie des Polarroboters berechnet werden. Sie setzt sich additiv zusammen aus den vier Anteilen der Körper $\mathcal{K}_1$–$\mathcal{K}_4$.

1) Die kinetische Energie des Stativs $\mathcal{K}_1$
Das Stativ rotiert mit der Winkelgeschwindigkeit $\omega_1 = \dot\theta_1 e_3$ um die vertikale Achse e_3, die als Symmetrieachse durch dessen Schwerpunkt verläuft. Mit dem Trägheitstensor $I^1 = \operatorname{diag}(I_1^1, I_2^1, I_3^1)$ ist seine kinetische Energie also

$$\begin{aligned} T_1(\dot\theta_1) &= T_{1,\mathrm{rot}}(\dot\theta_1) \\ &= \frac{1}{2}\begin{pmatrix} 0 & 0 & \dot\theta_1 \end{pmatrix}\begin{pmatrix} I_1^1 & 0 & 0 \\ 0 & I_2^1 & 0 \\ 0 & 0 & I_3^1 \end{pmatrix}\begin{pmatrix} 0 \\ 0 \\ \dot\theta_1 \end{pmatrix} \\ &= \frac{1}{2}I_3^1\dot\theta_1^2. \end{aligned} \tag{2.4.18}$$

Das Stativ $\mathcal{K}_1$ besteht im wesentlichen aus einem Rohr der Dichte ρ_1 mit Höhe h, Innenradius R_i und Außenradius R_a $(0 \le R_i < R_a)$. Sein Trägheitsmoment ist also näherungsweise

$$\begin{aligned} I_3^1 &= \int_0^h\int_0^{2\pi}\int_{R_i}^{R_a} \rho_1 R^2\, R dR\, d\phi\, dz \\ &= 2\pi h\rho_1 \int_{R_i}^{R_a} R^3\, dR \\ &= \pi h\rho_1\frac{R_a^4 - R_i^4}{2} = m_1\frac{R_a^2+R_i^2}{2} \end{aligned} \tag{2.4.19}$$

mit der Masse $m_1 = \pi h(R_a^2 - R_i^2)\rho_1$.

2) Die kinetische Energie der Führungsmanschette $\mathcal{K}_2$
Die Führungsmanschette rotiert mit $\omega = \omega_1 + \omega_2$ um ihren Schwerpunkt (vgl. 2.4.13). Sie hat die Hauptträgheitsachsen e_1', e_2' und e_3', ihre kinetische Energie ist

$$\begin{aligned} T_2(\theta_2, \dot\theta_1, \dot\theta_2) &= T_{2,\mathrm{rot}}(\theta_2, \dot\theta_1, \dot\theta_2) \\ &= \frac{1}{2}\begin{pmatrix} \dot\theta_2 & \dot\theta_1 \sin\theta_2 & \dot\theta_1\cos\theta_2 \end{pmatrix} \begin{pmatrix} I_1^2 & 0 & 0 \\ 0 & I_2^2 & 0 \\ 0 & 0 & I_3^2 \end{pmatrix} \begin{pmatrix} \dot\theta_2 \\ \dot\theta_1 \sin\theta_2 \\ \dot\theta_1 \cos\theta_2 \end{pmatrix} \\ &= \frac{1}{2}\left[I_1^2 \dot\theta_2^2 + \left(I_2^2 \sin^2\theta_2 + I_3^2 \cos^2\theta_2 \right) \dot\theta_1^2 \right]. \end{aligned} \tag{2.4.20}$$

Die Führungsmanschette habe die Länge $2l_2$ und einen quadratischen Querschnitt der Breite und Höhe $2C$. Die Aussparung zur Führung des Manipulatorarms sei ebenfalls quadratisch, mit Breite und Höhe $2c$. Mit den Massen $m_a := 8l_2C^2\rho_2$ des äußeren Vollquaders und $m_i := 8l_2c^2\rho_2$ des herausgeschnittenen Quaders erhält man nach (2.4.17) die Hauptträgheitsmomente

$$I_1^2 = m_a \frac{l_2^2 + C^2}{3} - m_i \frac{l_2^2 + c^2}{3} = I_3^2, \qquad I_2^2 = m_a \frac{2C^2}{3} - m_i \frac{2c^2}{3}. \tag{2.4.21}$$

3) Die kinetische Energie des Manipulatorarms $\mathcal{K}_3$
Der Manipulatorarm rotiert ebenfalls mit der (vektoriellen) Winkelgeschwindigkeit ω um den Koordinatenursprung (also i.a. *nicht* um seinen eigenen Schwerpunkt!) und bewegt sich zusätzlich mit der Geschwindigkeit $\dot r$ radial nach außen. Nach dem Satz von STEINER ist seine kinetische Energie die Summe der folgenden beiden Terme:
Energie der Rotation des Manipulatorarms mit Winkelgeschwindigkeit ω um den *eigenen* Schwerpunkt:

$$\begin{aligned} T_{3,\mathrm{rot}}(\theta_2, \dot\theta_1, \dot\theta_2) &= \frac{1}{2}\begin{pmatrix} \dot\theta_2 & \dot\theta_1 \sin\theta_2 & \dot\theta_1\cos\theta_2 \end{pmatrix} \begin{pmatrix} I_1^3 & 0 & 0 \\ 0 & I_2^3 & 0 \\ 0 & 0 & I_3^3 \end{pmatrix} \begin{pmatrix} \dot\theta_2 \\ \dot\theta_1 \sin\theta_2 \\ \dot\theta_1 \cos\theta_2 \end{pmatrix} \\ &= \frac{1}{2}\left[I_1^3 \dot\theta_2^2 + \left(I_2^3 \sin^2\theta_2 + I_3^3 \cos^2\theta_2 \right) \dot\theta_1^2 \right]. \end{aligned} \tag{2.4.22}$$

Translationsenergie einer Punktmasse $m_3 = m_B$ im Armschwerpunkt re_2' (vgl. 2.4.10):

$$\begin{aligned} T_{3,\mathrm{trans}}(r, \theta_2, \dot r, \dot\theta_1, \dot\theta_2) &= \frac{m_B}{2} \left\| \frac{d}{dt}(re_2') \right\|_2^2 \\ &= \frac{m_B}{2}\left[\dot r^2 + \left(\dot\theta_1^2 \cos^2\theta_2 + \dot\theta_2^2 \right) r^2 \right]. \end{aligned} \tag{2.4.23}$$

Diese Zerlegung entspricht (trotz zusätzlicher Zwangsbedingungen!) genau der Darstellung der Konfigurationsmannigfaltigkeit von $\mathcal{K}_3$ als direktes Produkt $\mathbb{R}^3 \times SO(3)$, wie sie bei der Definition eines starren Körpers angegeben wurde. $\mathcal{K}_3$ besitzt insgesamt die kinetische Energie

$$\begin{aligned} T_3(r, \theta_2, \dot r, \dot\theta_1, \dot\theta_2) &= T_{3,\mathrm{rot}}(\theta_2, \dot\theta_1, \dot\theta_2) + T_{3,\mathrm{trans}}(r, \theta_2, \dot r, \dot\theta_1, \dot\theta_2) \\ &= \frac{1}{2}\left[I_1^3 \dot\theta_2^2 + \left(I_2^3 \sin^2\theta_2 + I_3^3 \cos^2\theta_2 \right) \dot\theta_1^2 \right] + \frac{m_B}{2}\left[\dot r^2 + \left(\dot\theta_1^2 \cos^2\theta_2 + \dot\theta_2^2 \right) r^2 \right]. \end{aligned} \tag{2.4.24}$$

Der Manipulatorarm hat die Länge $2l$ und einen quadratischen Querschnitt mit Breite und Höhe $2c$, seine Masse ist $m_3 = 8lc^2\rho_3 = m_B$. Die Hauptträgheitsmomente sind also nach (2.4.17)

$$I_1^3 = m_B \frac{l^2 + c^2}{3} = I_3^3, \qquad I_2^3 = m_B \frac{2c^2}{3}. \tag{2.4.25}$$

4) Die kinetische Energie der Last $\mathcal{K}_4$
Mit (2.4.11) erhält man für die Punktmasse $m_4 = m_L$ einfach

$$\begin{aligned} T_4(r, \theta_2, \dot r, \dot\theta_1, \dot\theta_2) &= \frac{m_L}{2} \left\| \frac{d}{dt}\left((r + l)e_2'\right) \right\|_2^2 \\ &= \frac{m_L}{2}\left[\dot r^2 + \left(\dot\theta_1^2 \cos^2\theta_2 + \dot\theta_2^2 \right)(r + l)^2 \right]. \end{aligned} \tag{2.4.26}$$

Alles zusammen ergibt

$$
\begin{aligned}
T(r,\theta_2,\dot r,\dot\theta_1,\dot\theta_2) &= T_1(\dot\theta_1) + T_2(\theta_2,\dot\theta_1,\dot\theta_2) + T_3(r,\theta_2,\dot r,\dot\theta_1,\dot\theta_2) + T_4(r,\theta_2,\dot r,\dot\theta_1,\dot\theta_2)\\
&= \frac{1}{2} I_3^1 \dot\theta_1^2 + \frac{1}{2}\left[I_1^2\dot\theta_2^2 + \left(I_2^2\sin^2\theta_2 + I_3^2\cos^2\theta_2\right)\dot\theta_1^2\right]\\
&\quad + \frac{1}{2}\left[I_1^3\dot\theta_2^2 + \left(I_2^3\sin^2\theta_2 + I_3^3\cos^2\theta_2\right)\dot\theta_1^2\right]\\
&\quad + \frac{m_B}{2}\left[\dot r^2 + \left(\dot\theta_1^2\cos^2\theta_2 + \dot\theta_2^2\right)r^2\right]\\
&\quad + \frac{m_L}{2}\left[\dot r^2 + \left(\dot\theta_1^2\cos^2\theta_2 + \dot\theta_2^2\right)(r+l)^2\right].
\end{aligned}
\tag{2.4.27}
$$

Setzt man $I^{23} := I^2 + I^3$ und $m_{LB} := m_L + m_B$, so folgt

$$
\begin{aligned}
T(r,\theta_2,\dot r,\dot\theta_1,\dot\theta_2) = \frac{1}{2}\Big\{ m_{LB}\dot r^2 + \left[I_3^1 + I_2^{23}\sin^2\theta_2 + \left(I_3^{23} + m_B r^2 + m_L(r+l)^2\right)\cos^2\theta_2\right]\dot\theta_1^2\\
+ \left(I_1^{23} + m_B r^2 + m_L(r+l)^2\right)\dot\theta_2^2\Big\}.
\end{aligned}
\tag{2.4.28}
$$

Mit $a_0(r) := I_1^{23} - I_2^{23} + m_B r^2 + m_L(r+l)^2$ gilt schließlich wegen $I_1^{23} = I_3^{23}$

$$
\begin{aligned}
T(r,\theta_2,\dot r,\dot\theta_1,\dot\theta_2) &= \frac{1}{2}\left\{ m_{LB}\dot r^2 + \left(I_3^1 + I_2^{23} + a_0(r)\cos^2\theta_2\right)\dot\theta_1^2 + \left(I_2^{23} + a_0(r)\right)\dot\theta_2^2\right\}\\
&= \frac{1}{2}\begin{pmatrix}\dot r & \dot\theta_1 & \dot\theta_2\end{pmatrix}\begin{pmatrix} m_{LB} & 0 & 0\\ 0 & I_3^1 + I_2^{23} + a_0(r)\cos^2\theta_2 & 0\\ 0 & 0 & I_2^{23} + a_0(r)\end{pmatrix}\begin{pmatrix}\dot r\\ \dot\theta_1\\ \dot\theta_2\end{pmatrix}.
\end{aligned}
\tag{2.4.29}
$$

Die Trägheitsmatrix ist also in Polarkoordinaten diagonal!

Wir betrachten nun ein *homogenes* Gravitationsfeld mit Gravitationsbeschleunigung g, wie es z.B. auf der Erdoberfläche gegeben ist. Das Nullniveau der potentiellen Energie V liege im Drehpunkt des Roboters. Ihre Berechnung ist sehr einfach, da V nur von der Höhe der Roboterteile über dem Nullniveau abhängig ist, nicht aber von deren Geschwindigkeit. Zudem können starre Körper so behandelt werden, als ob ihre gesamte Masse im Schwerpunkt konzentriert wäre. Für die Einzelteile $\mathcal{K}_1$–$\mathcal{K}_4$ erhält man:

$$V_1 = \text{const} \tag{2.4.30}$$

(da der Schwerpunkt von $\mathcal{K}_1$ ortsfest ist),

$$V_2 = 0 \tag{2.4.31}$$

(da der Schwerpunkt von $\mathcal{K}_2$ stets im Nullniveau liegt),

$$V_3(r,\theta_2) = m_B g r\sin\theta_2 \tag{2.4.32}$$

und

$$V_4(r,\theta_2) = m_L g(r+l)\sin\theta_2. \tag{2.4.33}$$

Die Konstante V_1 tritt in der Euler-Lagrange-Gleichung nicht auf und kann daher von vornherein unberücksichtigt bleiben:

$$
\begin{aligned}
V(r,\theta_2) &= V_3(r,\theta_2) + V_4(r,\theta_2)\\
&= (m_{LB}r + m_L l)g\sin\theta_2 =: \frac{1}{2}s(r)g\sin\theta_2.
\end{aligned}
\tag{2.4.34}
$$

Der Schwerpunkt von Manipulatorarm und Last zusammen hat vom Drehpunkt gerade den Abstand $s(r)/2m_{LB} = r + m_L l/m_{LB}$. s gibt deshalb an, wie gut der belastete Manipulatorarm ausbalanciert ist. Für $r := -m_L l/m_{LB}$ ist er im Gleichgewicht ($s(r) = 0$), die potentielle Energie ist $V(r,\theta_2) = 0$. Wegen $a_0'(r) = s(r) = 0$ und $a_0''(r) = 2m_{LB} > 0$ ist dann außerdem der radiusabhängige Anteil $a_0(r)$ der Trägheitsmomente in (2.4.29) strikt minimal.

Aufgrund der Drehsymmetrie bezüglich der vertikalen Achse kommt der Winkel θ_1 weder in T noch in V vor, also auch nicht in den Differentialgleichungen des Polarroboters. Wohl aber tritt θ_1 in den Randbedingungen auf und muß daher in der numerischen Rechnung mitgeführt werden.

Mit (2.4.29) und (2.4.34) erhält man nun die LAGRANGE-Funktion (2.1.12) des Polarroboters

$$\begin{aligned} L(r,\theta_2,\dot r,\dot\theta_1,\dot\theta_2) &= T(r,\theta_2,\dot r,\dot\theta_1,\dot\theta_2) - V(r,\theta_2) \\ &= \frac{1}{2}\Big[m_{LB}\dot r^2 + \big(I_3^1 + I_2^{23} + a_0(r)\cos^2\theta_2\big)\,\dot\theta_1^2 + \big(I_2^{23} + a_0(r)\big)\,\dot\theta_2^2 - s(r)g\sin\theta_2\Big] \end{aligned} \tag{2.4.35}$$

und hieraus die drei Komponenten (2.1) der EULER-LAGRANGE-Gleichung:

1) Längs des Manipulatorarms wirkt die Kraft

$$\begin{aligned} F &= \frac{d}{dt}\frac{\partial L}{\partial \dot r} - \frac{\partial L}{\partial r} \\ &= m_{LB}\ddot r - \frac{1}{2}\Big[s(r)\big(\dot\theta_1^2\cos^2\theta_2 + \dot\theta_2^2\big) - 2m_{LB}g\sin\theta_2\Big] \\ &= m_{LB}\ddot r - \frac{1}{2}s(r)\big(\dot\theta_1^2\cos^2\theta_2 + \dot\theta_2^2\big) + m_{LB}g\sin\theta_2. \end{aligned} \tag{2.4.36}$$

2) Auf die vertikale Achse wirkt das Drehmoment

$$\begin{aligned} M_1 &= \frac{d}{dt}\frac{\partial L}{\partial \dot\theta_1} - \frac{\partial L}{\partial \theta_1} \\ &= \frac{d}{dt}\Big[\big(I_3^1 + I_2^{23} + a_0(r)\cos^2\theta_2\big)\,\dot\theta_1\Big] - 0 \\ &= \big(I_3^1 + I_2^{23} + a_0(r)\cos^2\theta_2\big)\,\ddot\theta_1 + s(r)\dot r\dot\theta_1\cos^2\theta_2 - 2a_0(r)\dot\theta_1\dot\theta_2\sin\theta_2\cos\theta_2. \end{aligned} \tag{2.4.37}$$

3) Auf die horizontale Achse wirkt das Drehmoment

$$\begin{aligned} M_2 &= \frac{d}{dt}\frac{\partial L}{\partial \dot\theta_2} - \frac{\partial L}{\partial \theta_2} \\ &= \frac{d}{dt}\Big[a_0(r)\dot\theta_2\Big] - \frac{1}{2}\Big[-2a_0(r)\dot\theta_1^2\sin\theta_2\cos\theta_2 - s(r)g\cos\theta_2\Big] \\ &= a_0(r)\ddot\theta_2 + s(r)\dot r\dot\theta_2 + a_0(r)\dot\theta_1^2\sin\theta_2\cos\theta_2 + \frac{1}{2}s(r)g\cos\theta_2. \end{aligned} \tag{2.4.38}$$

Diese ausführliche Herleitung soll hauptsächlich das prinzipielle Vorgehen beim Aufstellen der Differentialgleichungen eines Roboters verdeutlichen. Wie schon am Ende von Abschnitt (2.1) bemerkt, könnte man an dieser Stelle Reibungsterme und ähnliches in das mathematische Modell aufnehmen. In der Praxis müßte man außerdem dessen quantitative Brauchbarkeit überprüfen und gegebenenfalls verbessern, z.B. durch Korrektur der Werte für die Trägheitsmomente (vgl. Abschnitt 1.4).

Ziel dieser Arbeit ist jedoch nicht primär die Entwicklung eines praxisreifen Modells, deshalb werden die idealisierte LAGRANGE-Funktion (2.4.35) und die daraus gewonnenen Differentialgleichungen (2.4.36–2.4.38) unverändert als Grundlage für die folgenden Ausführungen verwendet.

2.5 Der Polarroboter in zwei Dimensionen

Wir schränken jetzt die Bewegungsfreiheit des Polarroboters auf die horizontale Ebene ein ($\theta_2 = 0$), d.h. die Führungsmanschette wird horizontal ausgerichtet und am Stativ „festgeschweißt“.

Mit $a_1(r) := I_3^1 + I_2^{23} + a_0(r)$ gilt dann $a_1'(r) = a_0'(r) = s(r)$, und die LAGRANGE-Funktion des Polarroboters lautet

$$\begin{aligned} L(r,\dot r,\dot\theta_1) &= \frac{1}{2}\Big(m_{LB}\dot r^2 + \big(I_3^1 + I_2^{23} + a_0(r)\big)\dot\theta_1^2\Big) \\ &= \frac{1}{2}\Big(m_{LB}\dot r^2 + a_1(r)\dot\theta_1^2\Big). \end{aligned} \tag{2.5.1}$$

Die Gleichungen (2.4.36/2.4.37) für Kraft und Drehmoment vereinfachen sich zu

$$F = m_{LB}\ddot r - \frac{1}{2}s(r)\dot\theta_1^2, \tag{2.5.2}$$

$$M_1 = a_1(r)\ddot\theta_1 + s(r)\dot r\dot\theta_1. \tag{2.5.3}$$

Mit den Bezeichnungen

$$\begin{aligned} z_1 &:= \dot r & z_3 &:= r & u_1 &:= F \\ z_2 &:= \dot\theta_1 & z_4 &:= \theta_1 & u_2 &:= M_1 \end{aligned} \tag{2.5.4}$$

sowie

$$\begin{aligned} s(z_3) &:= 2(m_{LB} z_3 + m_L l) \\ a_1(z_3) &:= I_3^1 + I_1^{23} + m_B z_3^2 + m_L (z_3 + l)^2 \end{aligned} \tag{2.5.5}$$

erhält man dann das komplette Differentialgleichungssystem

$$\begin{aligned} \ddot r &= \dot z_1 = (u_1 + \tfrac{1}{2} s z_2^2)/m_{LB} & \dot\lambda_1 &= \lambda_2 s z_2/a_1 - \lambda_3 \\ \ddot\theta_1 &= \dot z_2 = (u_2 - s z_1 z_2)/a_1 & \dot\lambda_2 &= -\lambda_1 s z_2/m_{LB} + \lambda_2 s z_1/a_1 - \lambda_4 \\ \dot r &= \dot z_3 = z_1 & \dot\lambda_3 &= -\lambda_1 s z_2^2 + \lambda_2 \underbrace{(2 m_{LB} z_1 z_2 + s \dot z_2)}_{(s z_2)^{\cdot}}/a_1 \\ \dot\theta_1 &= \dot z_4 = z_2 & \dot\lambda_4 &= 0 \\ \dot z_5 &= 1 + \tfrac{1}{2}(\epsilon_1 u_1^2 + \epsilon_2 u_2^2) & \dot\lambda_5 &= 0 \end{aligned} \tag{2.5.6}$$

und die HAMILTON-Funktion

$$\begin{aligned} H(z,\lambda,u) = {} & \lambda_1 \Big(u_1 + \frac{1}{2} s z_2^2\Big)\Big/ m_{LB} + \lambda_3 z_1 \\ & + \lambda_2 (u_2 - s z_1 z_2)/a_1 + \lambda_4 z_2 - \Big(1 + \frac{1}{2}\|u\|_\epsilon^2\Big). \end{aligned} \tag{2.5.7}$$

In dieser Formulierung sind außer den adjungierten Variablen λ alle in den Differentialgleichungen auftretenden Größen unmittelbar physikalisch verständlich. Zudem läßt sie sich fast sofort in den zentralen Programmteil für die rechte Seite umsetzen: man braucht nur s und a_1 nach (2.5.5) zu berechnen und u_1, u_2 nach dem Maximumprinzip zu bestimmen; alle erforderlichen Variablen stehen dann zur Verfügung.

Allerdings können bisher nur bang-bang-Steuerungen berechnet werden, es fehlt noch eine Untersuchung der singulären Steuerungen. Zu diesem Zweck wird das Differentialgleichungssystem (2.5.6) etwas umformuliert. Die physikalische Bedeutung von s als gemeinsamer „Schwerpunkt“ von Arm und Last läßt es nämlich sinnvoll erscheinen, s als Zustandsvariable einzuführen:

$$\begin{aligned} x_1 &:= 2 m_{LB} \dot r & x_3 &:= s & u_1 &:= 2F \\ x_2 &:= \dot\theta_1 & x_4 &:= \theta_1 & u_2 &:= 2 m_{LB} M_1 \end{aligned} \tag{2.5.8}$$

Es sei darauf hingewiesen, daß in diesen *Schwerpunktskoordinaten* (wie auch in 2.5.4) die Steuerungen u_i nicht auf $[-1,+1]$ normiert sind, die Matrix D also nicht benötigt wird. Mit

$$\begin{aligned} a(x_3) &:= \left[\frac{x_3^2}{2} + 2 m_{LB}\left(\frac{m_L m_B}{m_L + m_B} l^2 + I_3^1 + I_1^{23}\right)\right]^{-1} \\ &= 2/\big[x_3^2 + 4 l^2 m_L m_B + 4\big(I_3^1 + I_1^{23}\big) m_{LB}\big] \end{aligned} \tag{2.5.9}$$

gelten die Beziehungen

$$1/a(x_3) = 2 m_{LB} a_1(r), \tag{2.5.10}$$

$$da(x_3)/dx_3 =: a'(x_3) = -x_3 a(x_3)^2, \tag{2.5.11}$$

$$da(x_3(t))/dt =: \dot a(x_3) = -x_1 x_3 a(x_3)^2, \tag{2.5.12}$$

und aus (2.5.2/2.5.3) erhält man in der neuen Notation weiter

$$\begin{aligned} u_1 &= 2 m_{LB} \ddot r - s(r) \dot\theta_1^2 \\ &= \dot x_1 - x_3 x_2^2, \end{aligned} \tag{2.5.13}$$

$$\begin{aligned} u_2 &= 2 m_{LB} a_1(r) \ddot\theta_1 + 2 m_{LB} s(r) \dot r \dot\theta_1 \\ &= \dot x_2 / a(x_3) + x_1 x_3 x_2. \end{aligned} \tag{2.5.14}$$

Das Differentialgleichungssystem (2.5.6) nimmt damit eine etwas einfachere Gestalt an:

$$\dot{x}_1 = u_1 + x_2^2 x_3 \qquad \dot{\lambda}_1 = \lambda_2 x_2 x_3 a(x_3) - \lambda_3 \tag{2.5.15}$$

$$\dot{x}_2 = (u_2 - x_1 x_2 x_3) a(x_3) \qquad \dot{\lambda}_2 = \lambda_2 x_1 x_3 a(x_3) - \lambda_4 - 2\lambda_1 x_2 x_3 \tag{2.5.16}$$

$$\dot{x}_3 = x_1 \qquad \dot{\lambda}_3 = \lambda_2 (\underbrace{x_1 x_2 + x_3 \dot{x}_2}_{(x_2 x_3)^{\cdot}}) a(x_3) - \lambda_1 x_2^2 \tag{2.5.17}$$

$$\dot{x}_4 = x_2 \qquad \dot{\lambda}_4 = 0 \tag{2.5.18}$$

$$\dot{x}_5 = 1 + \frac{1}{2}(\epsilon_1 u_1^2 + \epsilon_2 u_2^2) \qquad \dot{\lambda}_5 = 0 \tag{2.5.19}$$

Verweise auf die Zustandsdifferentialgleichungen werden z.B. mit (2.5.18a), solche auf die adjungierten Differentialgleichungen z.B. mit (2.5.15b) vorgenommen. Zum Schluß sei noch die HAMILTON-Funktion in Schwerpunktskoordinaten angegeben:

$$H(x, \lambda, u) = \lambda_1(u_1 + x_2^2 x_3) + \lambda_2(u_2 - x_1 x_2 x_3) a(x_3) + \lambda_3 x_1 + \lambda_4 x_2 - \Big(1 + \frac{1}{2}\|u\|_\epsilon^2\Big). \tag{2.5.20}$$

2.6 Singuläre Steuerungen für den Polarroboter

In diesem Abschnitt sollen die singulären Steuerungen für den in (2.4) eingeführten Polarroboter untersucht werden, die bei rein *zeitoptimaler* Steuerung auftreten können. Sei also jetzt $\epsilon = 0$. Jede der Steuerungen u_i kann dann auf einem Intervall $I_S := [t_a, t_b]$ singulär werden. Nach Satz (2.3.6) gibt es jedoch stets mindestens eine Steuerungskomponente, die auf dem Rand liegt (bang-bang). Im folgenden werde o.B.d.A. die Formulierung in Schwerpunktskoordinaten (2.5.8) verwendet. Soweit nichts anderes gesagt wird, beziehen sich alle Aussagen auf das Intervall I_S.

2.6.1 Die Radialsteuerung u_1

Zunächst sei die Steuerung u_1 singulär. In der HAMILTON-Funktion (2.5.20) fällt wegen $\epsilon = 0$ der Term $\|u\|_\epsilon^2$ weg:

$$H(x, \lambda, u) = \lambda_1(u_1 + x_2^2 x_3) + \lambda_2(u_2 - x_1 x_2 x_3) a(x_3) + \lambda_3 x_1 + \lambda_4 x_2 - 1. \tag{2.6.1.1}$$

Aus ihr leitet man sofort die Schaltfunktion q_1 für u_1 ab (vgl. 2.3.1):

$$\frac{\partial H}{\partial u_1}(x, \lambda, u) = \lambda_1. \tag{2.6.1.2}$$

Nach Voraussetzung ist u_1 singulär, also verschwinden $q_1 = \lambda_1$ sowie alle zeitlichen Ableitungen. Aus (2.5.15–2.5.19) folgt daher mit (2.5.12):

$$0 \overset{!}{=} H_{u_1} = \lambda_1, \tag{2.6.1.3}$$

$$0 \overset{!}{=} (H_{u_1})^{\cdot} = \dot{\lambda}_1 = \lambda_2 x_2 x_3 a - \lambda_3, \tag{2.6.1.4}$$

$$\begin{aligned} 0 \overset{!}{=} (H_{u_1})^{\cdot\cdot} &= \ddot{\lambda}_1 \\ &= (\lambda_2 x_1 x_3 a - \lambda_4) x_2 x_3 a + \lambda_2 (x_2 x_3)^{\cdot} a - \lambda_2 x_2 x_3 x_1 x_3 a^2 - \lambda_2 (x_2 x_3)^{\cdot} a \\ &= -\lambda_4 x_2 x_3 a. \end{aligned} \tag{2.6.1.5}$$

Der Faktor $a(x_3)$ ist stets positiv. Auf I_S ist daher die mit (2.6.1.5) äquivalente Gleichung

$$\boxed{x_2 x_3 \lambda_4 = 0} \tag{2.6.1.6}$$

erfüllt. Anstatt nun H_{u_1} weiter abzuleiten, wird zunächst die durch (2.6.1.6) unmittelbar verfügbare Information ausgewertet. Dazu formulieren wir den

Satz (2.6.1.7): $f_1, \ldots, f_n\colon [t_a, t_b] \to I\!R \setminus \{0\}$ seien differenzierbare Funktionen, $f\colon [t_a, t_b] \to I\!R$ löse die Differentialgleichung

$$f' = \pm f \sum_{\nu=1}^{n} \frac{f'_\nu}{f_\nu}. \tag{2.6.1.8}$$

Dann gilt

$$f = f(t_a) \prod_{\nu=1}^{n} \left(\frac{f_\nu}{f_\nu(t_a)}\right)^{\pm 1}. \tag{2.6.1.9}$$

Beweis: Dies folgt unmittelbar aus der Lösungsformel

$$f(t) = f(t_0) \exp\left(\pm \int_{t_0}^{t} \sum_{\nu=1}^{n} \frac{f'_\nu(\tau)}{f_\nu(\tau)}\, d\tau\right) \qquad \forall t, t_0 \in [t_a, t_b] \tag{2.6.1.10}$$

für die *lineare* Differentialgleichung (2.6.1.8). ∎

Wir unterscheiden nun drei Fälle, in denen die Gleichung (2.6.1.6) erfüllt ist:

- *Fall 1: $x_2 = 0$, x_3, λ_4 beliebig*

Es folgt $\dot{x}_2 = 0$, nach (2.5.16a) also $u_2 a = 0$. Damit ist $u_2 = 0$ ebenfalls singulär. Dies steht aber im Widerspruch zu Satz (2.3.6). Fall 1 ist daher bei optimaler Steuerung ausgeschlossen.

- *Fall 2: $x_2 \neq 0$, $x_3 = 0$, λ_4 beliebig*

Es folgt $\dot{x}_3 = 0$, nach (2.5.17a) also $x_1 = 0$ und weiter $\dot{x}_1 = 0$. Mit (2.5.15a) erhält man schließlich $u_1 = 0$. Die (singuläre) Steuerung u_1 ist in diesem Fall eindeutig bestimmt: sie ist nicht aktiv. Physikalisch bedeutet die Bedingung $x_3 = 0$, daß der Manipulatorarm im Gleichgewicht ist; das minimale Trägheitsmoment ermöglicht dann eine rasche Drehbewegung (man beachte $x_3 = s$ und vergleiche die Ausführungen nach 2.4.34)!

- *Fall 3: $x_2 x_3 \neq 0$, $\lambda_4 = 0$*

Wegen $\dot{\lambda}_1 = 0$, $a(x_3) > 0$ und $x_2 x_3 \neq 0$ folgt aus (2.5.15b)

$$\lambda_2 = \frac{\lambda_3}{x_2 x_3 a}. \tag{2.6.1.11}$$

Aus (2.5.17b) folgt weiter wegen $\lambda_1 = 0$

$$\dot{\lambda}_3 = \lambda_3 \frac{(x_2 x_3)^{\cdot}}{x_2 x_3}. \tag{2.6.1.12}$$

Nach Satz (2.6.1.7) gilt also

$$\lambda_3 = \frac{\lambda_3(t_a)}{(x_2 x_3)(t_a)} x_2 x_3, \tag{2.6.1.13}$$

und nach (2.6.1.11) erhält man

$$\lambda_2 = \frac{\lambda_3(t_a)}{(x_2 x_3)(t_a)} \frac{1}{a}. \tag{2.6.1.14}$$

Aus der Beziehung

$$0 \stackrel{!}{=} H(x, \lambda, u) = \frac{\lambda_3(t_a)}{(x_2 x_3)(t_a)} \left(\frac{1}{a}(u_2 - x_1 x_2 x_3) a + x_2 x_3 x_1\right) - 1 = \frac{\lambda_3(t_a)}{(x_2 x_3)(t_a)} u_2 - 1 \tag{2.6.1.15}$$

folgt schließlich

$$\frac{\lambda_3(t_a)}{(x_2 x_3)(t_a)} = \frac{1}{u_2}, \tag{2.6.1.16}$$

und die Gleichungen (2.6.1.13/2.6.1.14) vereinfachen sich zu

$$\lambda_2 = \frac{1}{u_2 a}, \qquad \lambda_3 = \frac{x_2 x_3}{u_2}. \tag{2.6.1.17}$$

Damit sind zwar die adjungierten Differentialgleichungen analytisch gelöst, was die Dimension des Systems (2.5.15–2.5.19) auf vier reduziert, zu u_1 hat man aber keine weiteren Informationen erhalten. Auch die höheren (formalen) Ableitungen der Schaltfunktion λ_1 ergeben nichts Neues, denn sie enthalten alle $\lambda_4 = 0$ als Faktor! Also ist die Steuerung u_1 von der Ordnung $\kappa_1 = \infty$, eine verallgemeinerte Legendre-Clebsch-Bedingung kann man nicht formulieren.

Die Bedingung $\lambda_4|_{I_S} = 0$ ist jedoch wegen $\lambda_4 = \text{const}$ äquivalent zu $\lambda_4(t_e) = 0$. Diese zusätzliche Transversalitätsbedingung(!) kann nur erfüllt sein, wenn die entsprechende Endbedingung für $x_4 = \theta_1$ überflüssig ist. Das Randwertproblem hat dann dieselbe Lösung, wenn man die genannte Endbedingung wegläßt; es ist *entartet*. Man beachte, daß dieser Typ von Entartung nicht dem in Abschnitt (1.3.1) besprochenen entspricht. Tatsächlich ist letzterer im Fall 3 sogar ausgeschlossen, denn andernfalls wäre $\lambda_5 = 0$ (statt -1), und aus (2.6.1.15) würde mit (2.6.1.13) und (2.6.1.14) folgen, daß alle adjungierten Variablen identisch verschwinden.

Da Fall 1 ausgeschlossen ist und Fall 3 höchstens weiter untersucht werden kann, wenn spezielle Randbedingungen vorgegeben sind, sei nun $x_2 \neq 0$, $x_3 = 0$ und λ_4 beliebig (Fall 2). Aus (2.6.1.5) folgt dann weiter mit (2.5.15–2.5.19) sowie (2.5.12)

$$\begin{aligned}
0 &\stackrel{!}{=} (H_{u_1})^{\dddot{}} = \dddot{\lambda}_1 \\
&= -\lambda_4 \left((u_2 - x_1x_2x_3)ax_3a + x_2x_1a - x_2x_3x_1x_3a^2\right) \\
&= -\lambda_4 a \left(u_2x_3a + (1-2x_3^2a)x_1x_2\right), \qquad (2.6.1.18)
\end{aligned}$$

$$\begin{aligned}
0 &\stackrel{!}{=} (H_{u_1})^{\ddddot{}} = \ddddot{\lambda}_1 \\
&= -\lambda_4\dot{a}\big(u_2\underline{x}_3a + (1-2x_3^2a)\underline{x}_1x_2\big) - \lambda_4a\big(u_2\underline{x}_1a + u_2\underline{x}_3\dot{a} \\
&\qquad + (1-2x_3^2a)^{\dot{}}\,\underline{x}_1x_2 + (1-2\underline{x}_3^2a)(u_1x_2 + x_2^3\underline{x}_3 + \underline{x}_1\dot{x}_2)\big) \\
&= -\lambda_4au_1x_2 \qquad (2.6.1.19)
\end{aligned}$$

wegen $x_1 = x_3 = 0$ (es genügt, die unterstrichenen Terme zu betrachten). Also hat u_1 im Fall $x_3 = 0$ die Ordnung $\kappa_1 = 2$. Die verallgemeinerte Legendre-Clebsch-Bedingung lautet damit

$$0 \stackrel{!}{\leq} (-1)^3 \frac{\partial}{\partial u_1}\frac{d^4}{dt^4}\frac{\partial}{\partial u_1} H(x,\lambda,u) = \lambda_4x_2a. \qquad (2.6.1.20)$$

Sie ist äquivalent zu

$$\boxed{\lambda_4x_2 \geq 0} \qquad (2.6.1.21)$$

und bestimmt das Vorzeichen der Konstanten λ_4 in Abhängigkeit von der Drehrichtung des Roboters, die auf I_S durch $\operatorname{sign} x_2 \neq 0$ eindeutig gegeben ist. Umgekehrt folgt daraus, daß die Drehrichtung auf *jedem* singulären Bogen dieselbe ist und insbesondere dort nicht wechselt!

Übrigens ist das komplette Differentialgleichungssystem auf I_S analytisch lösbar. Der Vollständigkeit halber sei die Lösung hier angegeben:

$$\begin{aligned}
&\dot{x}_1(t) = 0 && x_1(t) = 0 \\
&\dot{x}_2(t) = u_2a(0) && x_2(t) = x_2(t_a) + (t-t_a)u_2a(0) \\
&\dot{x}_3(t) = 0 && x_3(t) = 0 \\
&\dot{x}_4(t) = x_2(t) && x_4(t) = x_4(t_a) + (t-t_a)x_2(t_a) + \tfrac{1}{2}(t-t_a)^2u_2a(0) \\
&\dot{x}_5(t) = 1 && x_5(t) = t \\
&\dot{\lambda}_1(t) = 0 && \lambda_1(t) = 0 \\
&\dot{\lambda}_2(t) = -\lambda_4 && \lambda_2(t) = \lambda_2(t_a) - (t-t_a)\lambda_4 \\
&\dot{\lambda}_3(t) = 0 && \lambda_3(t) = 0 \\
&\dot{\lambda}_4(t) = 0 && \lambda_4(t) = \text{const} \\
&\dot{\lambda}_5(t) = 0 && \lambda_5(t) = -1
\end{aligned} \qquad (2.6.1.22)$$

Die HAMILTON-Funktion

$$\begin{aligned} H(x,\lambda,u) &= \big(\lambda_2(t_a) - (t-t_a)\lambda_4\big)u_2a(0) + \lambda_4\big(x_2(t_a) + (t-t_a)u_2a(0)\big) - 1 \\ &= \lambda_2(t_a)u_2a(0) + \lambda_4 x_2(t_a) - 1 \end{aligned} \tag{2.6.1.23}$$

ist offenbar konstant, und da sie nach Bemerkung (2.2.13) sogar verschwindet, gilt

$$\lambda_4 = \frac{1 - \lambda_2(t_a)u_2a(0)}{x_2(t_a)}. \tag{2.6.1.24}$$

2.6.2 Die Tangentialsteuerung u_2

Sei jetzt die Steuerung u_2 singulär auf I_S. Aus der HAMILTON-Funktion (2.6.1.1) erhält man

$$\frac{\partial H}{\partial u_2}(x,\lambda,u) = a\lambda_2. \tag{2.6.2.1}$$

Als Schaltfunktion wird man der Einfachheit halber $q_2 := \lambda_2$ definieren. Sie hat in Bezug auf die Nullstellen dieselben Eigenschaften wie $a\lambda_2$. Analog zu (2.6.1.3–2.6.1.5) erhält man

$$0 \overset{!}{=} H_{u_2} = a\lambda_2, \tag{2.6.2.2}$$

$$\begin{aligned} 0 \overset{!}{=} (H_{u_2})^{\cdot} &= a\dot{\lambda}_2 \\ &= a(-2\lambda_1 x_2 x_3 - \lambda_4), \end{aligned} \tag{2.6.2.3}$$

$$\begin{aligned} 0 \overset{!}{=} (H_{u_2})^{\cdot\cdot} &= a\ddot{\lambda}_2 \\ &= -2a(-\lambda_3 x_2 x_3 + \lambda_1(u_2 - x_1x_2x_3)ax_3 + \lambda_1 x_2 x_1) \\ &= 2a(x_2(\lambda_3 x_3 - \lambda_1 x_1) - (u_2 - x_1x_2x_3)\lambda_1 x_3 a). \end{aligned} \tag{2.6.2.4}$$

Wegen $\lambda_1 x_3 \neq 0$ (dies wird weiter unten bewiesen!) hat u_2 also die Ordnung $\kappa_2 = 1$, es gilt

$$\frac{\partial}{\partial u_2}(H_{u_2})^{\cdot\cdot} = -2a^2\lambda_1 x_3. \tag{2.6.2.5}$$

Die verallgemeinerte LEGENDRE-CLEBSCH-Bedingung für u_2 lautet damit

$$0 \overset{!}{\leq} (-1)^2 \frac{\partial}{\partial u_2}\frac{d^2}{dt^2}\frac{\partial}{\partial u_2}H(x,\lambda,u) = -2a^2\lambda_1 x_3. \tag{2.6.2.6}$$

Wegen $a > 0$ und $u_1 = \hat{u}_1 \operatorname{sign}\lambda_1$ läßt sie sich vereinfachen zu

$$\boxed{u_1 x_3 \leq 0} \tag{2.6.2.7}$$

Die verallgemeinerte LEGENDRE-CLEBSCH-Bedingung besitzt auch eine physikalische Interpretation: u_2 kann nur dann singulär sein, wenn u_1 den Manipulatorarm „nach innen zieht“ ($x_3 = s \to 0$). Da u_2 stets eine Fliehkraft erzeugt, den Arm also „nach außen drückt“, wird u_1 dadurch behindert. Oder umgekehrt ausgedrückt: Der Rotationsmotor läuft mit voller Kraft, wenn die von ihm erzeugte Fliehkraft die Radialbewegung unterstützen kann.

Die (singuläre) Steuerung u_2 läßt sich aus der Bedingung (2.6.2.4) unmittelbar herleiten. Sie ist durch

$$\begin{aligned} u_2 &= \frac{x_2}{\lambda_1 x_3 a}(\lambda_3 x_3 - \lambda_1 x_1) + x_1x_2x_3 \\ &= \frac{x_2}{a}\left(\frac{\lambda_3}{\lambda_1} - \frac{x_1}{x_3}\right) + x_1x_2x_3 \\ &= -\frac{x_2}{a}\left(\frac{\dot{\lambda}_1}{\lambda_1} + \frac{\dot{x}_3}{x_3}\right) + x_1x_2x_3 \end{aligned} \tag{2.6.2.8}$$

stets eindeutig bestimmt. Setzt man diesen Ausdruck in die Differentialgleichung (2.5.16a) für x_2 ein, so ergibt dies

$$\dot{x}_2 = -x_2\left(\frac{\dot{\lambda}_1}{\lambda_1} + \frac{\dot{x}_3}{x_3}\right). \tag{2.6.2.9}$$

Nach Satz (2.6.1.7) folgt daraus mit (2.5.16b) wegen $\lambda_2 = \dot{\lambda}_2 = 0$

$$x_2 = \frac{(\lambda_1 x_2 x_3)(t_a)}{\lambda_1 x_3} = -\frac{\lambda_4}{2\lambda_1 x_3}, \tag{2.6.2.10}$$

und man erhält im wesentlichen die beiden folgenden Differentialgleichungen *zweiter* Ordnung:

$$\ddot{x}_3 = u_1 + \frac{\lambda_4^2}{4\lambda_1^2 x_3}, \tag{2.6.2.11}$$

$$\ddot{\lambda}_1 = \qquad \frac{\lambda_4^2}{4\lambda_1 x_3^2}. \tag{2.6.2.12}$$

Jetzt muß noch genau untersucht werden, ob die Division durch λ_1 und x_3 in (2.6.2.8) überhaupt zulässig war. Zunächst ist klar, daß keine der beiden Variablen auf einem ganzen Intervall verschwinden kann, solange u_2 singulär ist. Nach den Ergebnissen des letzten Abschnitts wäre dann nämlich auch u_1 singulär: ein Widerspruch zu Satz (2.3.6). Nun sind λ_1 und x_3 beide absolutstetig. Also ist die Menge

$$\mathcal{N} := \left(\lambda_1^{-1}(\{0\}) \cup x_3^{-1}(\{0\})\right) \cap I_S \tag{2.6.2.13}$$

ihrer Nullstellen eine *Nullmenge*. Sei $\mathcal{N} \neq \emptyset$, $t_1 \in \mathcal{N}$, $I_l := [t_a, t_1[$ und $I_r :=]t_1, t_b]$. Aus der Differentialgleichung (2.5.16b) folgt dann

$$0 = \dot{\lambda}_2(t_1) \stackrel{!}{=} -\lambda_4. \tag{2.6.2.14}$$

Mit (2.6.2.10) erhält man hieraus

$$x_2(t) = 0 \qquad \forall t \in I_S \setminus \mathcal{N}, \tag{2.6.2.15}$$

und mit (2.6.2.8) schließlich

$$u_2(t) = 0 \qquad \forall t \in I_S \setminus \mathcal{N}. \tag{2.6.2.16}$$

Da x_2 absolutstetig sein muß, definiert man $x_2|_{\mathcal{N}} := 0$, womit (2.6.2.15) auf ganz I_S gilt. Mit u_2 verfährt man aus praktischen Gründen genauso, obwohl die Werte auf der Nullmenge $\mathcal{N}$ theoretisch beliebig sind. Damit gilt auch (2.6.2.16) auf ganz I_S.

Da u_1 sowohl auf I_l als auch auf I_r konstant ist und höchstens in t_1 springt, erhält man auf I_S folgende Differentialgleichungen und Lösungen:

$$\begin{aligned}
\dot{x}_1(t) &= u_1(t) & x_1(t) &= x_1(t_1) + (t-t_1)u_1(t) \\
\dot{x}_2(t) &= 0 & x_2(t) &= 0 \\
\dot{x}_3(t) &= x_1(t) & x_3(t) &= x_3(t_1) + (t-t_1)x_1(t_1) + \tfrac{1}{2}(t-t_1)^2 u_1(t) \\
\dot{x}_4(t) &= 0 & x_4(t) &= x_4(t_1) \\
\dot{\lambda}_1(t) &= -\lambda_3(t) & \lambda_1(t) &= \lambda_1(t_1) - (t-t_1)\lambda_3(t_1) \\
\dot{\lambda}_2(t) &= 0 & \lambda_2(t) &= 0 \\
\dot{\lambda}_3(t) &= 0 & \lambda_3(t) &= \lambda_3(t_1) \\
\dot{\lambda}_4(t) &= 0 & \lambda_4(t) &= 0
\end{aligned} \tag{2.6.2.17}$$

Aus der linearen Gleichung für λ_1 erkennt man sofort, daß λ_1 höchstens eine Nullstelle t_1 in I_S besitzt. Die Gleichung für x_3 ist quadratisch in t und hat auf I_l, I_r verschiedene Koeffizienten für t^2. Sie besitzt also im Falle $x_3(t_1) = 0$ bis zu zwei weitere Lösungen in I_S, wenn die Gleichung

$$t - t_1 = \frac{2x_1(t_1)}{u_1(t)} \tag{2.6.2.18}$$

auf I_l oder I_r lösbar ist. Eine Untersuchung aller Fälle (einschließlich $t_1 \in \partial I_S$, d.h. $I_l = \emptyset$ oder $I_r = \emptyset$) zeigt jedoch, daß aufgrund der verallgemeinerten LEGENDRE-CLEBSCH-Bedingung der Bruch auf I_l stets positiv und auf I_r stets negativ ist, so daß in keinem Fall eine weitere Lösung existiert. Damit besitzt auch x_3 höchstens eine Nullstelle auf I_S!

Im Falle $\lambda_1(t_1) = 0$ gilt

$$\tilde{H}(t_1) = \lambda_3(t_1)\dot{x}_3(t_1) - 1 = 0, \tag{2.6.2.19}$$

also

$$-\dot{\lambda}_1(t_1) = \lambda_3(t_1) \neq 0, \tag{2.6.2.20}$$

$$x_1(t_1) = \dot{x}_3(t_1) \neq 0. \tag{2.6.2.21}$$

t_1 ist dann *einfache* Nullstelle von λ_1 und Schaltstelle von u_1. Weiter besteht die einfache Beziehung

$$\lambda_3(t_1) = \frac{1}{x_1(t_1)}. \tag{2.6.2.22}$$

Da λ_3 auf I_S konstant ist (vgl. 2.6.2.17), läßt sich mit (2.6.2.22) und der verallgemeinerten LEGENDRE-CLEBSCH-Bedingung zeigen, daß λ_1 und x_3 keine *verschiedenen* Nullstellen auf I_S haben können. Ist andererseits $t_1 \in \mathring{I}_S$, so liefert die verallgemeinerte LEGENDRE-CLEBSCH-Bedingung die Information, daß t_1 gemeinsame Nullstelle von λ_1 und x_3 sein *muß*. Diese Ergebnisse lassen sich zusammenfassen in dem folgenden

Lemma (2.6.2.23): Auf einem singulären Bogen $I_S = [t_a, t_b]$ von u_2 ist die Nullstellenmenge $\mathcal{N}$ aus (2.6.2.13) *entweder* leer *oder* sie enthält genau ein Element:

$$\mathcal{N} \in \{\emptyset, \{t_1\}\}. \tag{2.6.2.24}$$

Für $\mathcal{N} = \{t_1\}$ liegt einer der drei folgenden Fälle vor:
- $t_1 \in \partial I_S$, entweder $x_3(t_1) = 0$ *oder* $\lambda_1(t_1) = 0$,
- $t_1 \in \partial I_S$, $x_3(t_1) = 0$ *und* $\lambda_1(t_1) = 0$,
- $t_1 \in \mathring{I}_S$, $x_3(t_1) = 0$ *und* $\lambda_1(t_1) = 0$.

Die Steuerung u_1 ist stets auf I_l und auf I_r konstant, in den beiden ersten Fällen also auf ganz I_S. Die Steuerung u_2 sowie die Zustandsvariablen und die adjungierten Variablen sind gegeben durch (2.6.2.16) und (2.6.2.17), mit entsprechenden Vereinfachungen für $\lambda_1(t_1) = 0$ bzw. $x_3(t_1) = 0$. ∎

3. Numerische Ergebnisse

Dieses Kapitel behandelt die numerische Berechnung zeitoptimaler Steuerungen für den Polarroboter. Zunächst wird eine von einem Homotopieparameter abhängige Klasse von Steuerungsproblemen ausgewählt und genau formuliert.

Sodann werden Aspekte der numerischen Realisierung behandelt. Es folgt eine Erläuterung des Vorgehens beim Ermitteln der optimalen Schaltstruktur und bei der Beschaffung geeigneter Startdaten für die iterative Lösung nach dem indirekten Ansatz.

Im abschließenden Hauptteil des Kapitels werden für geeignet gewählte Werte des Homotopieparameters die Lösungen beschrieben, die im Anhang graphisch dargestellt sind. Dabei ergeben sich verschiedene Schaltstrukturen, die am Schluß zusammenfassend dargestellt werden.

3.1 Wahl und Formulierung einer Problemklasse

Im folgenden sollen *zeitoptimale* Steuerungen für einen Polarroboter berechnet werden. Im industriellen Bereich sind zeitoptimale Lösungen von Interesse, weil sie eine maximale Produktionsrate ermöglichen, aus mathematischer Sicht sind sie u.a. interessant, weil singuläre Steuerungen auftreten ($\epsilon = 0!$).

Zur exakten Problemformulierung gehört zunächst eine Spezifizierung des Robotermodells, d.h. der physikalischen Konstanten in der LAGRANGE-Funktion (2.4.35). Wir verwenden die folgenden Werte:

$$\begin{aligned} g &:= 9.81\,\mathrm{ms}^{-2} & m_B &:= 40\,\mathrm{kg} \\ l &:= 0.75\,\mathrm{m} & m_L &:= 0 \\ I_1^{23} &:= 18.5\,\mathrm{kgm}^2 & I_3^{23} &:= I_1^{23} \\ I_2^{23} &:= 0.12\,\mathrm{kgm}^2 & I_3^{1} &:= 0 \end{aligned} \tag{3.1.1}$$

Diese Daten sind dem idealisierten Modell eines typischen Leichtbauroboters entnommen und wurden bereits in einigen Untersuchungen verwendet. Ihre Genauigkeit ist völlig hinreichend, um die grundsätzlichen Schwierigkeiten bei der Roboter-Steuerung zu untersuchen, denn die Struktur der Lösungen wird von kleinen Änderungen der Werte nicht beeinflußt. Die Daten wurden daher beibehalten, so daß die Vergleichbarkeit der hier gewonnenen Resultate mit früheren Ergebnissen gewährleistet ist.

Wir betrachten wieder die zweidimensionale Einschränkung des Roboters auf die horizontale Ebene ($\theta_2 = 0$) und lassen das Stativ unberücksichtigt ($I_3^1 = 0$). Es sei noch darauf hingewiesen, daß der Roboter *unbelastet* ist ($m_L = 0$), und daß die *Normfallbeschleunigung* g für den rein horizontalen Fall keine Rolle spielt.

Der Manipulatorarm sei zu Beginn der Bewegung voll ausgefahren ($r = l$) und befinde sich in Ruhe. Dann soll der Arm eingefahren werden bis zu einem vorgegebenen Endradius $r_e \in [-l, +l]$ und dort wieder zur Ruhe kommen. Eine effektive Drehung finde dabei nicht statt, o.B.d.A. ist der Winkel in Start und Ziel also $\theta_1 = 0$. Da der Winkel *nicht* modulo 2π gemessen wird, ist durch diese Forderung jede Bewegung mit einer oder mehr vollen Umdrehungen ausgeschlossen, obwohl der Polarroboter physikalisch denselben Endzustand erreichen würde. KONZELMANN [21] untersucht auch Fälle, in denen eine Bewegung mit Volldrehung schneller ist als jede Bewegung ohne effektive Drehung.

Randbögen und Berührpunkte treten für das hier formulierte Problem nicht auf, denn die einzige Zustandsbeschränkung

$$r^2 - l^2 \leq 0 \tag{3.1.2}$$

ist immer nur im Startpunkt aktiv (für $r_e = -l$ auch im Endpunkt). Sie braucht daher numerisch nicht berücksichtigt zu werden, so daß sich das Randwertproblem genau wie in den Abschnitten (2.2) und (2.6) formulieren läßt. Die Startzeit sei o.B.d.A. $t_s := 0$, die unbekannte Endzeit sei $t_e =: T$. Zu den Differentialgleichungen (2.5.6) lauten dann die Rand- und Transversalitätsbedingungen:

$$\begin{aligned} \dot r(0) &= 0 & \dot r(T) &= 0 \\ \dot\theta_1(0) &= 0 & \dot\theta_1(T) &= 0 \\ r(0) &= l & r(T) &= r_e \\ \theta_1(0) &= 0 & \theta_1(T) &= 0 \\ & & \tilde H(T) &= 0 \end{aligned} \tag{3.1.3}$$

Die Differentialgleichung $\dot z_5 = 1 + \frac{1}{2}\|u\|_\epsilon^2$ mit Randbedingung $z_5(0) = 0$, die durch das LAGRANGE-Zielfunktional (2.2.1) entstanden ist, besitzt wegen $\epsilon = 0$ die Lösung $z_5(t) = t$. Daher ist die Differentialgleichung überflüssig und wird numerisch nicht berücksichtigt.

Homotopieparameter unserer Problemklasse ist der noch nicht festgelegte *Endradius* $r_e \in [-l, +l]$. Im wesentlichen wird im Rest der Arbeit die Abhängigkeit der Lösungen des Randwertproblems von r_e untersucht.

Nun müssen nur noch die Maximalwerte für Kraft bzw. Drehmoment der beiden Antriebe festgelegt werden. Diese seien

$$\begin{aligned} \hat u_1 &:= 5\,\mathrm{kgms}^{-2} & \hat u_2 &:= 300\,\mathrm{kgm}^2\mathrm{s}^{-2} \\ &= 5\,\mathrm{N}, & &= 300\,\mathrm{Nm}. \end{aligned} \tag{3.1.4}$$

Der Radialantrieb u_1 wurde im Vergleich zum Tangentialantrieb u_2 relativ schwach gewählt, damit das charakteristische Zusammenwirken beider Motoren stärker ausgeprägt wird.

3.2 Aspekte der numerischen Realisierung

Das im letzten Abschnitt formulierte Randwertproblem mit den Differentialgleichungen (2.5.6) und den Randbedingungen (3.1.3) ergibt sich im wesentlichen zwangsläufig aus dem zugrundeliegenden Steuerungsproblem, es werde hier als *natürliches* Randwertproblem bezeichnet. Hingegen gibt es viele Möglichkeiten, ein auf dem Rechner lösbares *numerisches* Mehrpunktrandwertproblem zu erhalten; sie unterscheiden sich hauptsächlich in der Behandlung der Schaltvorschriften und in der Transformation auf ein festes Intervall.

Die Transformation wird beschrieben durch einen mindestens stückweise stetig differenzierbaren, überall links- und rechtsseitig differenzierbaren Homöomorphismus

$$\begin{aligned} \phi\colon [0,1] =: I^* &\to I = [0,T], \\ t^* &\mapsto t. \end{aligned} \tag{3.2.1}$$

Setzt man wie in Abschnitt (1.4) $z := (x^\top, \lambda^\top)^\top$ und definiert

$$z^*(t^*) := z(t), \qquad p^* := p, \tag{3.2.2}$$

so lautet die transformierte rechte Seite (1.4.3)

$$f_i^*(t^*, z^*(t^*), p^*) = f_i(t, z(t), p) \cdot \phi'(t^*), \tag{3.2.3}$$

worunter gegebenenfalls der links- oder rechtsseitige Grenzwert zu verstehen ist. Für die numerischen Schaltfunktionen Q_i^* fordert man

$$Q_i^*(t^*, z^*(t^*), p^*) = Q_i(t, z(t), p). \tag{3.2.4}$$

Im einfachsten Fall behandelt man nur die freie Endzeit als Parameter $p := T$ und arbeitet mit sogenannten *festen* Stützstellen. Dazu wählt man ein beliebiges Mehrzielgitter

$$\Gamma_m^*\colon\ 0 = \tau_1^* < \cdots < \tau_m^* = 1 \tag{3.2.5}$$

und die lineare Transformation

$$\phi\colon I^* \to I, \quad t^* \mapsto t := Tt^*. \tag{3.2.6}$$

So entsteht (bezogen auf die Endzeit T) eine feste Unterteilung des Intervalls I durch die natürlichen Stützstellen $\tau_j = T\tau_j^*$. Die natürlichen Randbedingungen (3.1.3) werden für die Numerik mit einem * versehen und in 0 und 1 (statt 0 und T) ausgewertet, die freie Endzeit $p^* = p$ ist dann durch die einzige Transversalitätsbedingung $\tilde{H}^*(1) = 0$ festgelegt.

Wir wenden uns jetzt der Behandlung der Schaltvorschriften zu. Die natürlichen Schaltfunktionen

$$Q_i(t, z, p) = \lambda_i \qquad (i \in \{1, 2\}) \tag{3.2.7}$$

für u_1 und u_2 wurden schon in Abschnitt (2.6) angegeben.

Falls nur bang-bang-Bögen auftreten, werden die Steuerungen automatisch bei der Berechnung der Schaltstellen mitgeliefert. Sie lassen sich also prinzipiell während des Programmablaufs bestimmen, denn zwischen zwei benachbarten Schaltstellen ist

$$u_i(t) = \hat{u}_i \operatorname{sign} \lambda_i(t) \tag{3.2.8}$$

eindeutig definiert. Jede Schaltfunktion kann hierbei auch mehrere Nullstellen haben, es muß weder deren Anzahl bekannt sein noch werden sonstige a priori-Informationen zur Schaltstruktur benötigt. In schwierigen Situationen ist die Verwendung der natürlichen Schaltfunktionen allerdings nicht empfehlenswert, weil nach Bock [3], [5] die Schaltstellen sehr empfindlich auf Störungen — vor allem der adjungierten Variablen — reagieren. Numerisch weit günstiger ist dann die weiter unten beschriebene Technik mit Zeitschaltern oder variablen Stützstellen.

Besonders kompliziert wird die Situation beim Auftreten singulärer Steuerungen. Anders als bei reinen bang-bang-Steuerungen kann man nämlich die natürlichen Schaltfunktionen selbst *nicht* benutzen, um während der Integration die Schaltstruktur zu ermitteln. Dies wird im folgenden erläutert.

Die singuläre Steuerung ist (in 1.3.2.23) gerade so definiert, daß $q_i^{(2\kappa_i)} = 0$ auf I_S. Der Aufsprungpunkt t_a des singulären Bogens I_S ist also nach (1.3.2.21) festgelegt durch die $2\kappa_i$ Bedingungen

$$q_i^{(n)}(t_a+) = \lambda_i^{(n)}(t_a+) \overset{!}{=} 0 \qquad (n \in \{0, \ldots, 2\kappa_i - 1\}), \tag{3.2.9}$$

so daß er als spezieller Schaltpunkt *theoretisch* während des Programmablaufs erkannt werden kann (die *numerische* Brauchbarkeit von (3.2.9) wird weiter unten erörtert). Für den Absprungpunkt t_b hingegen gibt es keine ähnlichen Bedingungen. Einerseits kann wegen $\lambda_i|_{I_S} = 0$ jeder Punkt $t \in I_S$ Schaltstelle und damit Absprungpunkt sein. Andererseits könnte man prinzipiell solange singulär steuern, bis

- die singuläre Steuerung an den Rand des Steuerbereichs stößt,
- die verallgemeinerte Legendre-Clebsch-Bedingung verletzt ist,
- eine Zustandsbeschränkung verletzt ist oder
- die Endbedingungen erfüllt sind.

Der letzte Fall ist eher unwahrscheinlich, meist wird eines der drei ersten Ereignisse die maximal mögliche Länge des singulären Intervalls begrenzen. Es ist aber auch möglich, daß keiner der Fälle auftritt. Soll z.B. der Polarroboter von einer sehr niedrigen auf eine sehr hohe Drehzahl beschleunigen, so ist die folgende Strategie optimal: Zunächst wird die Gleichgewichtsposition des Manipulatorarms angesteuert ($s = 0$), so daß die Drehzahl besonders effektiv erhöht werden kann. Ist die Enddrehzahl nahezu erreicht, wird die Gleichgewichtslage wieder verlassen, um das vorgegebene Ziel zu erreichen. In der mittleren Phase läuft der Drehmotor mit voller Kraft $u_2 = \hat{u}_2$, die Radialsteuerung ist $u_1 = 0$, also singulär. Mit dieser Steuerung kann der Roboter theoretisch beliebig lange laufen, ohne daß einer der vier Fälle eintritt.

Das Beispiel ist zwar aus praktischer Sicht belanglos, zeigt aber gut die wesentliche Schwierigkeit: Der Absprungpunkt t_b ist nicht wie t_a durch lokale Eigenschaften der Lösungstrajektorie — und damit der Differentialgleichungen — sondern ausschließlich durch die Randbedingungen festgelegt. Dies bedeutet, daß der Verlauf der Lösung für $t > t_b$ den Wert t_b unmittelbar beeinflußt!

Zu einem beliebigen Zeitpunkt t_0 sind aber bei der numerischen Integration der Differentialgleichungen außer den lokalen Informationen in t_0 nur für $t < t_0$ Informationen zum Lösungsverlauf verfügbar.

Daher muß die Schaltstruktur *a priori* bekannt sein, um eine optimale Steuerung mit einem oder mehreren singulären Bögen numerisch berechnen zu können.

Zudem sind aber die Bedingungen (3.2.9) *numerisch instabil*, so daß man stattdessen andere (äquivalente) Bedingungen heranziehen muß. Im Fall $\kappa_i = 1$ ersetzt man beispielsweise die „Anfangsbedingungen"

$$q_i(t_a+) = 0, \quad \dot{q}_i(t_a+) = 0 \tag{3.2.10}$$

durch die (wegen $\ddot{q}_i|_{I_S} = 0$) äquivalenten, aber wesentlich stabileren „Randbedingungen"

$$q_i(t_a+) = 0, \quad q_i(t_b-) = 0. \tag{3.2.11}$$

Analog formuliert man für $\kappa_i > 1$ in t_a *und* t_b insgesamt $2\kappa_i$ geeignete Bedingungen. Auf diese Weise ist zugleich das Problem gelöst, den Absprungpunkt t_b zu bestimmen.

Auch für diese Technik muß die Schaltstruktur a priori bekannt sein, was von jetzt an vorausgesetzt werde. Es seien also gegeben:

- die Anzahl k der Schaltstellen,
- für jede Schaltstelle t_i die zugehörige Schaltfunktion $Q_i(t,z,p) = \lambda_{\sigma(i)} \in \{\lambda_1, \lambda_2\}$,
- für jedes Teilintervall $]t_i, t_{i+1}]$ ($i \in \{0, \dots, k\}$) die Steuerung u (ggf. in Abhängigkeit von t, z, p), womit dann auch die rechte Seite $f_i(t,z,p)$ bekannt ist.

Unter diesen Voraussetzungen können die numerischen Rechnungen mit sogenannten *variablen Stützstellen* ([3], [5]) durchgeführt werden, wie es auch in der vorliegenden Arbeit gehandhabt wurde. Dazu behandelt man alle Schaltstellen, die Endzeit T sowie die konstante adjungierte Variable λ_4 als Parameter:

$$\begin{aligned} p_i &:= t_i \qquad (i \in \{1, \dots, k\}), \\ p_{k+1} &:= T, \\ p_{k+2} &:= \lambda_4. \end{aligned} \tag{3.2.12}$$

Das Intervall I^* wird durch k beliebige Zeitpunkte t_i^* unterteilt:

$$0 =: t_0^* < t_1^* < \cdots < t_k^* < t_{k+1}^* := 1. \tag{3.2.13}$$

Mit der formalen Definition $p_0 := 0$ läßt sich die Transformation ϕ dann einfach darstellen als

$$\phi(t^*) := \sum_{i=0}^{k} \chi_{]t_i^*, t_{i+1}^*]}(t^*) \left(\frac{t_{i+1}^* - t^*}{t_{i+1}^* - t_i^*} p_i + \frac{t^* - t_i^*}{t_{i+1}^* - t_i^*} p_{i+1} \right). \tag{3.2.14}$$

Die charakteristische Funktion $\chi_{]t_i^*, t_{i+1}^*]}$ ist höchstens für einen Summanden ungleich Null, daher werden die Intervalle $[t_i^*, t_{i+1}^*]$ *linear* auf $[p_i, p_{i+1}]$ abgebildet. Die festen numerischen Schaltstellen t_i^* sind einfach die Nullstellen der *Zeitschalter*

$$Q_i^*(t^*, z^*, p^*) := t^* - t_i^*. \tag{3.2.15}$$

Ihre Bilder sind die natürlichen Schaltstellen $t_i = p_i$, die als Parameter mitoptimiert werden und im Mehrpunktrandwertproblem durch die *zusätzlichen inneren Punktbedingungen*

$$\lambda_{\sigma(i)}^*(t_i^*) = 0 \qquad (\Leftrightarrow Q_i(t,z,p) = 0) \tag{3.2.16}$$

festgelegt sind. Es sei jedoch ausdrücklich darauf hingewiesen, daß diese Bedingungen Auf- und Absprungpunkte nur für singuläre Bögen der Ordnung eins eindeutig definieren; bei höheren Ordnungen müßten in den entsprechenden Schaltstellen weitere Bedingungen gefordert werden. Wie sich noch herausstellen wird, kommen solche aber in unserem Fall nicht vor.

Die Mehrzielstützstellen τ_j^* schließlich werden geeignet zwischen den t_i^* verteilt. Dadurch passen sich die natürlichen Stützstellen automatisch zwischen den Schaltstellen t_i ein, sind also variabel.

Dieses Verfahren ist bei bekannter Schaltstruktur wegen seiner numerischen Vorteile grundsätzlich zu empfehlen, auch wenn keine singulären Bögen (oder Zustandsbeschränkungen) auftreten.

Zunächst wird die numerisch aufwendige *Schaltpunktsuche* wesentlich vereinfacht, da OPCON die Nullstellen linearer Schaltfunktionen auf Anhieb exakt ermittelt. Dadurch spart man bei jedem Schaltpunkt einen Zwischenschritt (HERMITE-Interpolation von q_i^*) und die anschließende iterative Approximation von t_i^* mittels NEWTON-Verfahren. Da für jede Auswertung von q_i^* die Differentialgleichungen über ein gewisses Intervall integriert werden müssen, ist die Zeitersparnis nicht unerheblich. Zudem arbeiten die einfachen Zeitschalter (3.2.15) mit maximaler Genauigkeit und vermeiden zusätzliche Instabilitäten.

Ein weiterer Vorzug ist die Tatsache, daß sich die Reihenfolge der Stütz- und Schaltstellen im Laufe der GAUSS-NEWTON-Iteration nicht ändert. Bei festen Stützstellen kann es nämlich passieren, daß eine Schaltstelle über eine Stützstelle wandert. Dies bewirkt fast immer eine Konvergenzverschlechterung und führt oft sogar zur Divergenz. Vor allem bei relativ dichten Mehrzielgittern und bei Homotopien spielt dieser Aspekt eine wesentliche Rolle, denn in beiden Fällen ist die Wahrscheinlichkeit sehr groß, daß feste Stützstellen von Schaltstellen übersprungen werden.

Zum Schluß sei noch erwähnt, daß die natürlichen Schaltzeiten $t_i = p_i$ *explizit* berechnet werden. Das bedeutet, daß sie als Bestandteile der Lösung in der Ausgabe von OPCON erscheinen und nicht als implizit während des Programmablaufs anfallende Daten umständlich gespeichert werden müssen. Tatsächlich sind die genauen Werte der Schaltzeiten zumindest in dieser Arbeit ebenso interessant wie die Zustandsvariablen und die adjungierten Variablen. Die Formulierung mit variablen Stützstellen entspricht somit genau der Aufgabenstellung.

3.3 Ermittlung der Schaltstruktur und Beschaffung der Starttrajektorie

Da bei unserer Problemklasse in jedem Fall singuläre Steuerungen auftreten, muß (zumindest einmal zu Beginn der Homotopie) die optimale Schaltstruktur ermittelt werden. Außerdem stellt sich das Problem, brauchbare Startdaten zur iterativen Lösung des Ausgleichsproblems (1.4.15–1.4.18) zu finden. Allgemein ist es für beide Aufgaben hilfreich, sich auf eine mit dem direkten Ansatz erzeugte Lösung des Steuerungsproblems zu stützen. Diese Methode wurde auch in der vorliegenden Arbeit mit Erfolg angewandt.

Zunächst liefert eine suboptimale Lösung bei genügend feiner Unterteilung des Prozeßintervalls wertvolle Informationen zur Schaltstruktur der optimalen Lösung. So läßt sich z.B. die ungefähre Lage von Schaltstellen ermitteln, oft kann man auch singuläre Bögen von bang-bang-Bögen unterscheiden.

Allerdings ist die strukturelle Interpretation einer suboptimalen Lösung nicht eindeutig, und a priori ist nicht klar, wie zuverlässig die aus ihr gewonnenen Informationen sind. Daher ist i.a. eine genaue theoretische Analyse erforderlich, gegebenenfalls muß man mehrere strukturell verschiedene Lösungsansätze numerisch überprüfen, um eine Extremale zu finden, die alle notwendigen Bedingungen erfüllt.

Schließlich eignet sich die suboptimale Lösung meist sehr gut als Starttrajektorie für den indirekten Ansatz, wenn man jeweils das gleiche Gitter (mit festen Stützstellen!) wählt. Besonders wichtig ist hierbei die im Programmpaket MUSCOD [9] realisierte Möglichkeit, mit Hilfe des direkten Ansatzes Näherungen der adjungierten Variablen in den Stützstellen zu berechnen. Dies ist sehr hilfreich, weil die adjungierten Variablen im Gegensatz zu den Zustandsvariablen nur abstrakte Hilfsgrößen sind, von deren Verlauf man selten eine anschauliche Vorstellung hat. Aus diesem Grund sind brauchbare Schätzungen anders nur schwierig zu gewinnen.

Wir wenden uns jetzt wieder dem konkreten Randwertproblem aus Abschnitt (3.1) zu und verwenden zur weiteren Analyse die Schwerpunktskoordinaten (2.5.8). Auch in dieser Formulierung sind alle Randwerte Null, bis auf

$$x_3(0) = s(l) \quad \text{und} \quad x_3(T) = s(r_e). \tag{3.3.1}$$

Zunächst untersuchen wir die einfachste *zulässige* Steuerung. Bei ihr ist nur der Radialantrieb aktiv:

$$u_1(t) := \begin{cases} -\hat{u}_1 & \text{für } t \in [0, T/2], \\ +\hat{u}_1 & \text{für } t \in]T/2, T], \end{cases} \tag{3.3.2}$$

$$u_2(t) := 0. \tag{3.3.3}$$

Durch Konstruktion adjungierter Variablen wird gezeigt, daß diese Steuerung auch *extremal* im Sinne des Maximumprinzips (1.3.1.1) ist.

Wegen (3.3.3) gilt $x_2(t) = 0$ und $x_4(t) = 0$. Weiter muß $\lambda_2(t) = 0$ sein, denn $u_2 = 0$ ist singulär. Mit (2.5.16b) folgt nun $\lambda_4(t) = 0$. Das Randwertproblem vereinfacht sich dadurch erheblich, so daß es leicht zu lösen ist. Man erhält sofort:

$$\frac{T}{2} = \sqrt{\frac{s(l) - s(r_e)}{\hat{u}_1}} \tag{3.3.4}$$

sowie eine komplette *extremale* Lösung:

$$\begin{aligned}
x_1(t) &= \begin{cases} -t\hat{u}_1 & \text{für } t \in [0, T/2] \\ -(T-t)\hat{u}_1 & \text{für } t \in]T/2, T] \end{cases} & \lambda_1(t) &= \left(\frac{t}{T/2} - 1\right)\frac{1}{\hat{u}_1} \\
x_2(t) &= 0 & \lambda_2(t) &= 0 \\
x_3(t) &= \begin{cases} s(l) - \frac{1}{2}t^2\hat{u}_1 & \text{für } t \in [0, T/2] \\ s(r_e) + \frac{1}{2}(T-t)^2\hat{u}_1 & \text{für } t \in]T/2, T] \end{cases} & \lambda_3(t) &= \frac{1}{T/2}\frac{1}{\hat{u}_1} \\
x_4(t) &= 0 & \lambda_4(t) &= 0
\end{aligned} \tag{3.3.5}$$

Die notwendigen Bedingungen des Maximumprinzips verifiziert man unmittelbar. Weiter sieht man, daß die Schaltbedingungen erfüllt sind, und daß die Formel (2.6.2.8) für die singuläre Steuerung u_2 mit (3.3.3) konsistent ist.

Zu überprüfen bleibt noch die verallgemeinerte LEGENDRE-CLEBSCH-Bedingung (2.6.2.7):

$$u_1 x_3 \leq 0. \tag{3.3.6}$$

Wegen $m_L = 0$ gilt $x_3 = 2m_{LB}r$, und (3.3.6) ist äquivalent mit

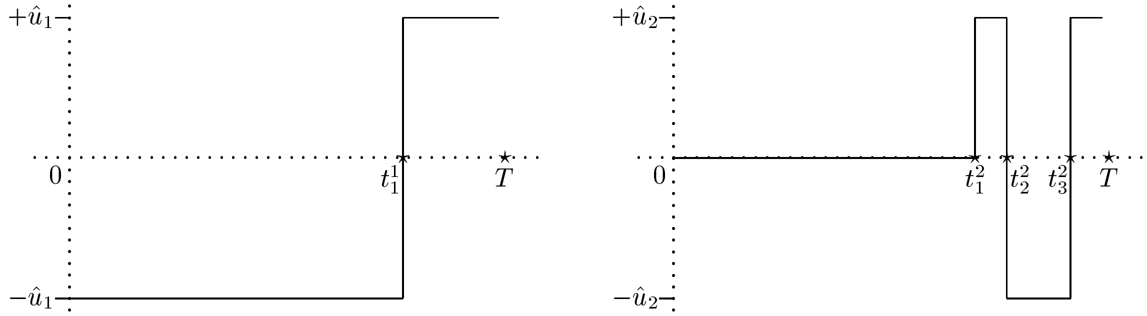
$$\begin{aligned} r(t) &\geq 0 \quad \text{für } t \in [0, T/2], \\ r(t) &\leq 0 \quad \text{für } t \in]T/2, T]. \end{aligned} \tag{3.3.7}$$

Deshalb kann die Lösung (3.3.5) nur für $r_e = -l$ optimal sein. Das trifft auch offensichtlich zu, denn jede Aktivität des Tangentialantriebs würde in diesem symmetrischen Fall die Bewegung nur verzögern.

Es sei noch darauf hingewiesen, daß die Schaltstelle $t_1 = T/2$ von u_1 im Innern des singulären Bogens $I_S = [0, T]$ der Tangentialsteuerung u_2 liegt; man vergleiche dazu Lemma (2.6.2.23).

Die hier konstruierte Lösung (3.3.5) wird in Abschnitt (3.4) als Referenzlösung verwendet. Sie werde von jetzt an als *Grundlösung* bezeichnet, ihre Endzeit gemäß (3.3.4) sei $\bar{T}$.

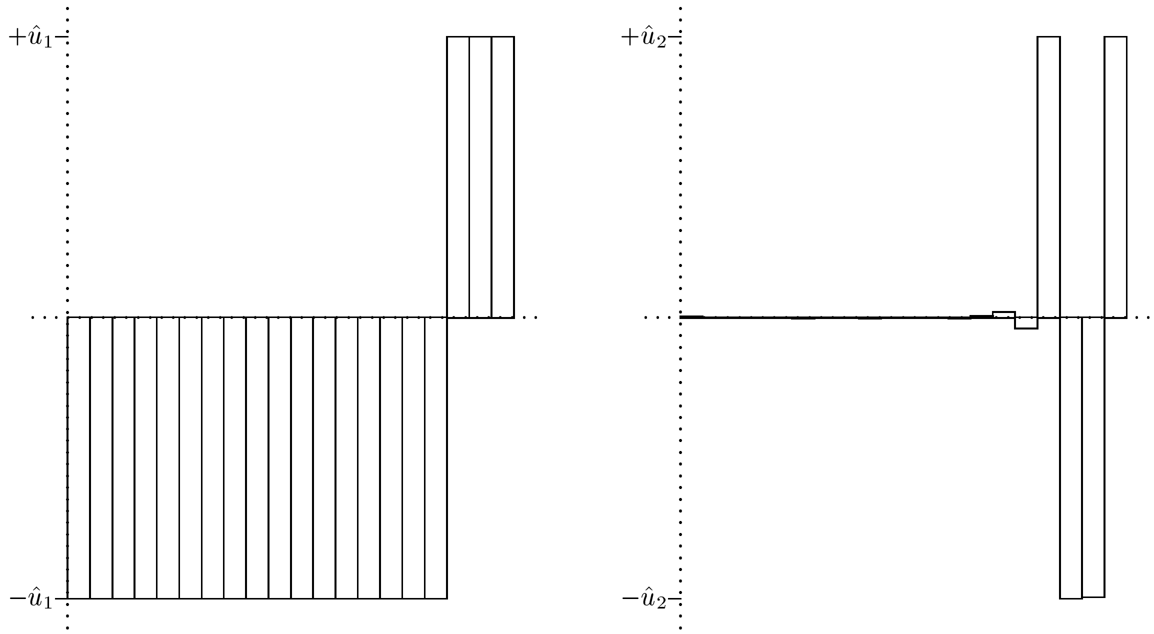
Das Steuerungsproblem muß jetzt nur noch für $r_e \in]-l, +l[$ gelöst werden (für $r_e = +l$ ist nichts zu optimieren). Der Radialantrieb wird in jedem Falle den Manipulatorarm zunächst nach innen beschleunigen, um ihn in einer zweiten Phase der Bewegung wieder abzubremsen. Die erste Bewegungsphase kann der Rotationsmotor nicht unterstützen, weil er stets eine Fliehkraft erzeugt. Durch einen „Schlenker" zur Seite kann er aber die Bremskraft in der zweiten Phase verstärken. Dazu muß der Schlenker (zumindest im wesentlichen) vor dem Nulldurchgang des Armschwerpunkts vonstatten gehen, weil danach die Fliehkraft wieder in die falsche Richtung wirkt. Aus diesem Grund betrachten wir zunächst nur Endradien $r_e \in]0, l[$, bei denen kein Nulldurchgang stattfindet und der Schlenker bis zum Ende der Bewegung dauert. Diesen rein physikalischen Überlegungen zufolge könnte die Schaltstruktur z.B. wie in Abbildung (3.3.8) aussehen.



Möglicher Kandidat für die optimale Schaltstruktur bei $r_e \in]0, l[$. Abbildung (3.3.8)

Aufgrund der Rotationssymmetrie des Polarroboters ist die *Richtung* des Schlenkers unwesentlich, man darf also ohne weiteres die Steuerung u_2 durch $-u_2$ ersetzen. Über die Reihenfolge der Schaltstellen t_1^1, t_1^2, t_2^2 läßt sich nichts Genaues sagen außer der Bedingung $t_1^2 \leq t_1^1$, die aus der verallgemeinerten LEGENDRE-CLEBSCH-Bedingung (3.3.6) folgt.

Es lohnt sich jetzt, eine mit dem direkten Ansatz erzeugte suboptimale Lösung als Informationsquelle heranzuziehen. Dazu wählen wir willkürlich $r_e := l/3 = 0.25\,\mathrm{m}$ und unterteilen $I = [0, T]$ äquidistant in zwanzig Teilintervalle, auf denen u jeweils konstant ist. Das auf diese Weise diskretisierte Randwertproblem wurde mit dem Programmpaket MUSCOD gelöst (siehe KONZELMANN [21]). Die Steuerungen zeigt Abbildung (3.3.9).



Suboptimale Steuerung bei 20 Teilintervallen, $r_e = 25\,\mathrm{cm}$. Abbildung (3.3.9)

Ein Vergleich mit Abbildung (3.3.8) bestätigt offensichtlich die vorherigen Überlegungen. Auffällig sind jedoch die von Null abweichenden Werte der Steuerung u_2 während der Beschleunigungsphase. Sie deuten darauf hin, daß der Tangentialantrieb schon von Anfang an aktiv ist, was sich auch als richtig herausstellen wird. Dieses Verhalten läßt sich folgendermaßen erklären: Die Bremswirkung des Schlenkers wird ein wenig vermindert durch die Endbedingung $\theta_1(T) = 0$, welche den Bewegungsablauf beeinflußt. Deshalb ist es sinnvoll, den Arm durch geringfügige Korrekturen während der Startphase in eine geeignete Ausgangsposition für den Schlenker zu bringen, so daß dieser seine Bremskraft voll entfalten kann. Insgesamt ergibt sich ein Zeitgewinn, obwohl die Beschleunigung am Anfang leicht behindert wird.

Was man der Abbildung (3.3.9) in diesem Maßstab nicht entnehmen kann, wohl aber den numerischen Daten, ist die Tatsache, daß der Wert der suboptimalen Steuerung u_2 im ersten Teilintervall mindestens viermal so groß ist wie in den folgenden dreizehn Intervallen. Dies läßt sich als Hinweis auf einen kurzen „bang" vor Beginn des singulären Bogens interpretieren.

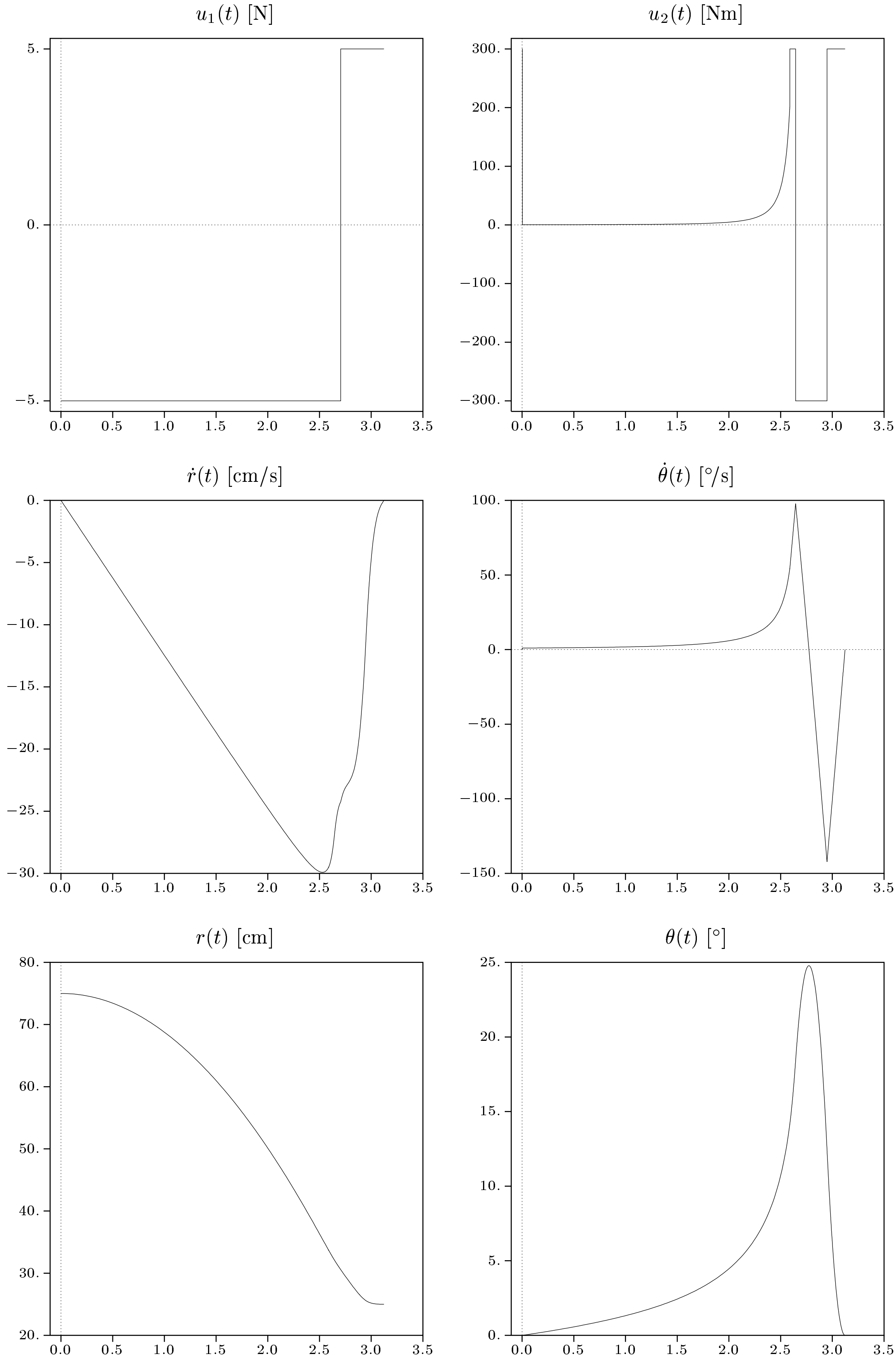
Tatsächlich ist es sehr unwahrscheinlich, daß ausgerechnet am Anfang der Bewegung die Aufsprungbedingungen (3.2.9) für einen singulären Bogen erfüllt sind. Dieser Sonderfall trifft nur für einen speziellen Endradius r_e zu, bei dem die Steuerung u_2 von Anfang an singulär und exakt Null ist, also die Schaltstruktur (3.3.8) hat. In diesem Fall wird die Bremswirkung des Schlenkers *nicht* behindert von der Endbedingung $x_4(T) = \theta_1(T) = 0$. Die Bedingung ist dann überflüssig und es gilt $\lambda_4(T) = 0$ (siehe Abschnitt 2.6.1, Fall 3). Mit dieser Transversalitätsbedingung anstelle der Endbedingung $r(T) = r_e$ kann der entsprechende Endradius r_e exakt ermittelt werden.

Jetzt muß „nur noch" eine optimale Lösung für $r_e = 0.25\,\mathrm{m}$ berechnet werden. Die Hauptschwierigkeit dabei ist, wie sich herausstellte, gute Startwerte für die Schaltzeiten zu finden. Bei den ersten Versuchen trat der im letzten Abschnitt beschriebene Effekt auf, daß die Schaltstellen (mehrmals) während der Iteration ihre Reihenfolge wechselten oder über Stützstellen wanderten. Dies führte stets zur Divergenz. Um die suboptimale Lösung verwenden zu können, durften aber keine neuen Stützstellen eingeführt und insbesondere keine variablen Stützstellen benutzt werden. Deshalb wurden für den indirekten Ansatz alle ungeradzahligen Stützstellen entfernt, also ein Gitter mit nur zehn Teilintervallen verwendet. Aus der MUSCOD-Lösung wurde dann zuerst eine Zwischenlösung mit der vereinfachten Schaltstruktur (3.3.8) berechnet, wobei zusätzlich die Bedingung $t_1^1 = t_2^2$ zu erfüllen war. Auf diese Weise mußten statt fünf nur drei Schaltstellen identifiziert werden.

In einem zweiten Schritt konnte dann aus dieser Zwischenlösung die wirklich optimale Lösung mit fünf Schaltstellen berechnet werden. Das Vorzeichen des „bang" von u_2 am Anfang sowie die Reihenfolge der Schaltstellen t_1^1, t_2^2 waren durch Ausprobieren zu ermitteln. Es stellte sich heraus, daß u_2 zwischen 0 und t_0^2 (dies sei der Schaltpunkt nach dem Startbang) denselben Wert hat wie auf $[t_1^2, t_2^2]$, und daß $t_2^2 < t_1^1$ ist. In den Plots auf der folgenden Seite kann man dies alles gut erkennen. Allerdings ist der Startbang mit etwa 2.44 ms Dauer so kurz, daß er optisch nur als senkrechte Linie erscheint. Ausführlichere Erläuterungen zu den Plots findet man im nächsten Abschnitt.

Zur Übertragung der Lösung auf ein Gitter mit variablen Stützstellen integriert man einfach die Differentialgleichungen mit bekanntem Anfangswert $z(0)$ und bekannter Steuerung u über das ganze Intervall (*single shooting*). Die so gewonnenen Werte $z(\tau_j)$ in den Stützstellen sind sehr genau. Sie dienen als Startdaten für das Mehrzielverfahren, welches nach wenigen Iterationen konvergiert. Diese Methode, nicht belegte Mehrzielknoten mit Startwerten zu versehen, ist in OPCON bereits eingebaut. Man muß dazu nur die bereits belegten Stützstellen — hier die erste und letzte — in einem Indexvektor spezifizieren.

$r_e = 25\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen



3.4 Die r_e-Homotopie

In diesem Abschnitt werden die Lösungen des Steuerungsproblems für verschiedene ausgewählte Endradien r_e beschrieben. Am Schluß folgt eine Zusammenfassung, die insbesondere die Abhängigkeit der Schaltstruktur von r_e erörtert.

Als Startwerte für die GAUSS-NEWTON-Iteration werden jeweils die Lösungsdaten eines benachbarten Problems verwendet. Auf diese Weise tastet man sich, von der Startlösung für $r_e = 25\,\text{cm}$ ausgehend, zu größeren und kleineren Werten vor.

Für jeden Endradius sind im Anhang auf je zwei Seiten die folgenden Daten geplottet:

- *Seite 1:* Radial- und Tangentialsteuerung, Radial- und Winkelgeschwindigkeit sowie Radius und Winkel in Abhängigkeit von der Zeit. Alle Zeiten sind in Sekunden gemessen, die Winkel aus Gründen der Anschaulichkeit in Altgrad. Für Ortsangaben wurde die Einheit Zentimeter gewählt. In den einzelnen Bildern sind die Einheiten jeweils angegeben.

- *Seite 2:* Hier sind die adjungierten Variablen gegen die Zeit aufgetragen, außerdem die Bahnen des Armschwerpunkts und der Last. Da die HAMILTON-Funktion dimensionslos ist, besitzt jede adjungierte Variable eine physikalische Dimension, die der Vollständigkeit halber angegeben ist.

Die Bahnen von Schwerpunkt und Last sind definiert durch

$$x = r\cos\theta_1, \qquad y = r\sin\theta_1, \tag{3.4.1}$$
$$\bar{x} = \bar{r}\cos\theta_1, \qquad \bar{y} = \bar{r}\sin\theta_1, \tag{3.4.2}$$

mit $\bar{r} := r + l$ (vgl. Abschnitt 2.4). Man beachte, daß in den Plots die Maßstäbe der beiden Achsen verschieden sind, so daß die Bahnen verzerrt wiedergegeben werden.

Alle geplotteten Daten wurden aus den *diskreten* OPCON-Lösungen mit dem Integrator VOAS berechnet (SEDGWICK [30]), und zwar in doppelter Genauigkeit (`DOUBLE PRECISION, REAL*8`) mit einem relativen Fehler unter 10^{-7}. Die Graphiken selbst wurden mit dem genormten Graphischen Kern-System GKS und dem Zusatz-Programmpaket BIZEPS von BECHLARS/BUHTZ/HECHT erzeugt.[1] Die dabei verwendete einfache(!) Genauigkeit von etwa sieben Dezimalstellen wurde also voll ausgenutzt.

Vor der Beschreibung der einzelnen Lösungen soll noch allgemein etwas zu deren Differenzierbarkeitseigenschaften gesagt werden.

Die Radialsteuerung u_1 besitzt keine singulären Bögen (siehe Abschnitt 3.3), sie springt also in jedem eigenen(!) Schaltpunkt. Hingegen besitzt u_2 sowohl singuläre Bögen als auch bang-bang-Bögen. Nach KELLEY [17] ist aber eine Steuerung beim Übergang von einem singulären Bogen zu einem bang-bang-Bogen *genau dann* stetig, wenn ihre Ordnung gerade ist. Also springt auch die Steuerung u_2 in jedem eigenen Schaltpunkt, denn sie ist von der Ordnung $\kappa_2 = 1$.

Mit diesen Informationen findet man, daß die einzelnen Variablen so oft stetig differenzierbar sind, wie in Tabelle (3.4.3) angegeben. Für u_2 wird unterschieden zwischen Schaltstellen sowie bang-bang-Bögen und singulären Bögen, für u_1 treten nur die beiden ersten Fälle auf. Die Zahl -1 bezeichnet einen Sprung, ν bezeichnet die Vielfachheit einer eventuellen Nullstelle von x_3 im Schaltpunkt von u_2. Es gilt stets $\nu \in \{0, 1, 2\}$.

Differenzierbarkeit von:		u_1	x_1	x_3	λ_1	λ_3	u_2	x_2	x_4	λ_2	λ_4
u_1: bang	u_2: bang	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
	u_2 schaltet	∞	$\nu+1$	$\nu+2$	≥ 2	≥ 1	-1	0	1	$\nu+1$	∞
	u_2: singulär	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
u_1 schaltet	u_2: bang	-1	0	1	≥ 3	≥ 1	∞	≥ 1	≥ 2	2	∞
	u_2 schaltet	-1	0	1	≥ 2	≥ 1	-1	0	1	2	∞
	u_2: singulär	-1	0	1	∞	∞	∞	∞	∞	∞	∞

Differenzierbarkeit der Steuerungen, Zustands- und adjungierten Variablen. Tabelle (3.4.3)

[1] Das gilt für die Originalversion der Arbeit. In dieser Neufassung wurden die Graphiken mit *gnuplot* erzeugt.

r_e	$t_1 = t_0^2$	t_2	$t_3 = t_1^2$	$t_4 = t_2^2$	$t_5 = t_1^1$	$t_6 = t_3^2$	t_7	t_8	T	$\bar{T}$
74.9990	.000496	$*$	0.00641	0.00775	0.00895	0.01431	$*$	$*$	0.01788	0.01790
74.9900	.001529	$*$	0.02051	0.02469	0.02847	0.04523	$*$	$*$	0.05643	0.05657
74.9000	.003946	$*$	0.07101	0.08238	0.09403	0.14223	$*$	$*$	0.17527	0.17889
74.0000	.005714	$*$	0.28504	0.30426	0.33531	0.43852	$*$	$*$	0.51611	0.56569
72.5000	.005580	$*$	0.49309	0.51314	0.55555	0.68266	$*$	$*$	0.78303	0.89443
70.0000	.005221	$*$	0.73720	0.75706	0.80891	0.95393	$*$	$*$	1.07251	1.26491
64.0000	.004590	$*$	1.15017	1.16990	1.23217	1.39715	$*$	$*$	1.53594	1.87617
55.0000	.003945	$*$	1.59829	1.62003	1.68815	1.86874	$*$	$*$	2.02080	2.52982
48.0000	.003550	$*$	1.87976	1.90488	1.97422	2.16457	$*$	$*$	2.32235	2.93939
40.0000	.003153	$*$	2.15775	2.18915	2.25772	2.46062	$*$	$*$	2.62348	3.34664
32.0000	.002779	$*$	2.40093	2.44239	2.50791	2.72823	$*$	$*$	2.89653	3.70945
25.0000	.002442	$*$	2.58937	2.64477	2.70484	2.94823	$*$	$*$	3.12298	4.00000
18.0000	.002055	$*$	2.75575	2.83346	2.88306	3.16302	$*$	$*$	3.34734	4.27083
12.0000	.001631	$*$	2.87945	2.98825	3.02033	3.35186	$*$	$*$	3.54885	4.48999
6.5025	.001089	$*$	2.97535	3.13118	3.13118	3.54451	$*$	$*$	3.76066	4.68180
3.0000	.000591	$*$	3.02662	3.23208	3.19186	3.69889	$*$	$*$	3.93675	4.80000
0.4535	.000000	$*$	3.05811	3.32935	3.22613	3.86998	$*$	$*$	4.13936	4.88415
0.0000	.000180	$*$	3.06278	3.35457	3.22906	3.91787	$*$	$*$	4.19705	4.89898
−0.3666	.000387	$*$	3.06595	3.38067	3.22839	3.96776	4.22161	4.22161	4.25709	4.91094
−1.0000	.000346	$*$	3.05812	3.35262	3.21439	3.89653	4.13603	4.34082	4.35280	4.93153
−3.0000	.000293	$*$	3.04581	3.31182	3.19159	3.80153	4.02124	4.54835	4.55167	4.99600
−5.0000	.000270	$*$	3.03942	3.28997	3.17830	3.74983	3.95658	4.69729	4.69913	5.05964
−10.0000	.000245	$*$	3.03316	3.25973	3.15955	3.67162	3.85567	4.98398	4.98483	5.21536
−15.0000	.000234	$*$	3.03428	3.24470	3.15091	3.62245	3.78910	5.21313	5.21368	5.36656
−20.0000	.000231	$*$	3.04038	3.23816	3.14829	3.58732	3.73843	5.41144	5.41186	5.51362
−25.0000	.000233	$*$	3.05077	3.23819	3.15031	3.56132	3.69721	5.58968	5.59004	5.65685
−32.0000	.000247	$*$	3.07266	3.24906	3.16000	3.53699	3.64965	5.81592	5.81624	5.85150
−40.0000	.000270	$*$	3.11054	3.28032	3.18112	3.52501	3.60398	6.05103	6.05134	6.06630
−45.0000	.000255	$*$	3.14218	3.30759	3.20064	3.52182	3.57786	6.18867	6.18894	6.19677
−48.0000	.000225	$*$	3.16353	3.32314	3.21510	3.51850	3.56314	6.26853	6.26876	6.27375
−50.0000	.000199	$*$	3.17871	3.33271	3.22602	3.51545	3.55382	6.32078	6.32098	6.32456
−50.6432	.000190	3.18378	3.18378	3.33569	3.22977	3.51435	3.55091	6.33742	6.33762	6.34081
−52.0000	.000171	3.19321	3.19480	3.34185	3.23806	3.51187	3.54490	6.37231	6.37248	6.37495
−54.0000	.000142	3.20851	3.21197	3.35079	3.25133	3.50789	3.53633	6.42317	6.42331	6.42495
−55.0000	.000128	3.21686	3.22102	3.35528	3.25846	3.50580	3.53216	6.44836	6.44850	6.44981
−56.0000	.000114	3.22567	3.23040	3.35980	3.26593	3.50365	3.52807	6.47342	6.47353	6.47457
−58.0000	.000088	3.24471	3.25023	3.36910	3.28196	3.49925	3.52008	6.52310	6.52319	6.52380
−60.0000	.000064	3.26567	3.27156	3.37884	3.29948	3.49475	3.51234	6.57227	6.57233	6.57267

Alle Schaltzeiten und die Endzeit $\bar{T}$ der Grundlösung im Verlauf der Homotopie. Tabelle (3.4.4)

Nun zu den einzelnen Lösungen. Da sich die Schaltstruktur auf dem Intervall $]25\,\text{cm}, 75\,\text{cm}[$ nicht ändert, beginnen wir mit der Darstellung der Homotopie nicht beim eigentlichen Startwert $r_e = 25\,\text{cm}$, sondern beim höchsten Wert, und ziehen den Manipulatorarm dann immer weiter nach innen. Dies hat den zusätzlichen Vorteil, daß die Schaltstruktur am Anfang deutlicher sichtbar ist.

Die Schaltzeiten sämtlicher Lösungen sind in Tabelle (3.4.4) verzeichnet. Zum Vergleich ist jeweils auch die Endzeit $\bar{T}$ der Grundlösung (3.3.5) angegeben. Die Tabellen (3.4.5) und (3.4.6) enthalten die gleichen Daten, aber normiert auf die optimale Endzeit T bzw. auf $\bar{T}$.

Wie üblich sind die Schaltstellen in aufsteigender Reihenfolge numeriert. Man beachte aber, daß t_1^1 und t_2^2 bei $r_e = 6.5025\,\text{cm}$ die Reihenfolge tauschen. Ab $r_e = 3\,\text{cm}$ gilt also $\phi(t_4^*) = t_5 < t_4 = \phi(t_5^*)$.

r_e	t_1/T	t_2/T	t_3/T	t_4/T	t_5/T	t_6/T	t_7/T	t_8/T	$\bar{T}/T$
74.9990	.027722	$*$	.358223	.433379	.500463	.800335	$*$	$*$	1.00109
74.9900	.027087	$*$	.363500	.437463	.504515	.801566	$*$	$*$	1.00253
74.9000	.022517	$*$	.405136	.470015	.536489	.811498	$*$	$*$	1.02064
74.0000	.011072	$*$	.552294	.589529	.649691	.849670	$*$	$*$	1.09607
72.5000	.007126	$*$	.629716	.655333	.709491	.871822	$*$	$*$	1.14227
70.0000	.004868	$*$	.687363	.705878	.754219	.889444	$*$	$*$	1.17940
64.0000	.002989	$*$	.748839	.761689	.802229	.909642	$*$	$*$	1.22151
55.0000	.001952	$*$	.790918	.801679	.835386	.924754	$*$	$*$	1.25189
48.0000	.001529	$*$	.809419	.820237	.850097	.932059	$*$	$*$	1.26569
40.0000	.001202	$*$	.822478	.834445	.860580	.937921	$*$	$*$	1.27565
32.0000	.000960	$*$	.828898	.843213	.865830	.941894	$*$	$*$	1.28065
25.0000	.000782	$*$	.829135	.846873	.866107	.944044	$*$	$*$	1.28083
18.0000	.000614	$*$	.823267	.846482	.861299	.944936	$*$	$*$	1.27589
12.0000	.000459	$*$	.811374	.842033	.851072	.944490	$*$	$*$	1.26519
6.5025	.000290	$*$	.791176	.832615	.832615	.942523	$*$	$*$	1.24494
3.0000	.000150	$*$	.768811	.821002	.810784	.939578	$*$	$*$	1.21928
0.4535	.000000	$*$	.738787	.804315	.779379	.934922	$*$	$*$	1.17993
0.0000	.000043	$*$	.729746	.799270	.769365	.933482	$*$	$*$	1.16724
-0.3666	.000091	$*$	.720199	.794129	.758356	.932036	.991668	.991668	1.15359
-1.0000	.000079	$*$	.702564	.770220	.738464	.895176	.950199	.997247	1.13296
-3.0000	.000064	$*$	.669163	.727606	.701192	.835195	.883464	.999271	1.09762
-5.0000	.000058	$*$	.646805	.700123	.676358	.797984	.841982	.999608	1.07672
-10.0000	.000049	$*$	.608477	.653930	.633832	.736559	.773481	.999830	1.04625
-15.0000	.000045	$*$	.581984	.622344	.604353	.694797	.726761	.999894	1.02932
-20.0000	.000043	$*$	.561800	.598346	.581739	.662862	.690785	.999922	1.01880
-25.0000	.000042	$*$	.545751	.579278	.563557	.637083	.661393	.999936	1.01195
-32.0000	.000042	$*$	.528289	.558619	.543307	.608122	.627493	.999945	1.00606
-40.0000	.000045	$*$	.514025	.542081	.525689	.582517	.595567	.999949	1.00247
-45.0000	.000041	$*$	.507708	.534436	.517154	.569050	.578106	.999955	1.00127
-48.0000	.000036	$*$	.504650	.530110	.512876	.561275	.568396	.999962	1.00080
-50.0000	.000032	$*$	.502883	.527246	.510367	.556156	.562226	.999967	1.00056
-50.6432	.000030	.502362	.502362	.526331	.509618	.554522	.560291	.999969	1.00050
-52.0000	.000027	.501094	.501343	.524418	.508132	.551099	.556282	.999972	1.00039
-54.0000	.000022	.499511	.500049	.521661	.506177	.546119	.550546	.999977	1.00026
-55.0000	.000020	.498854	.499499	.520319	.505306	.543662	.547750	.999980	1.00020
-56.0000	.000018	.498286	.499016	.519007	.504506	.541227	.544999	.999982	1.00016
-58.0000	.000013	.497412	.498257	.516481	.503122	.536432	.539626	.999986	1.00009
-60.0000	.000010	.496882	.497777	.514101	.502026	.531737	.534412	.999990	1.00005

Alle Schaltzeiten und die Vergleichszeit $\bar{T}$, jeweils normiert auf $T = 1$. Tabelle (3.4.5)

Auf die wenig anschaulichen numerischen Daten konnte nicht verzichtet werden. In einigen Fällen lassen sich nämlich wichtige Details in den graphischen Darstellungen optisch nicht auflösen, so daß die exakte Schaltstruktur nur noch den Tabellen (3.4.4–3.4.6) zu entnehmen ist.

1) Homotopie von 75 cm *bis* 0.4535 cm.

Für die extremen Werte $r_e = 74.999$ cm und $r_e = 74.99$ cm wurde aus technischen Gründen ganz auf Plots verzichtet: fünf bis sechs Dezimalstellen einiger Variablen sind hier zeitlich konstant, so daß eine spezielle Aufbereitung der Daten nötig gewesen wäre. Dieser Zusatzaufwand erschien unangemessen hoch, da zur nächsten Lösung ohnehin keine strukturellen Unterschiede bestehen.

r_e	$t_1/\bar{T}$	$t_2/\bar{T}$	$t_3/\bar{T}$	$t_4/\bar{T}$	$t_5/\bar{T}$	$t_6/\bar{T}$	$t_7/\bar{T}$	$t_8/\bar{T}$	$T/\bar{T}$
74.9990	.027692	*	.357834	.432908	.499919	.799465	*	*	.998913
74.9900	.027018	*	.362581	.436357	.503239	.799540	*	*	.997472
74.9000	.022061	*	.396942	.460509	.525639	.795086	*	*	.979775
74.0000	.010101	*	.503888	.537859	.592748	.775199	*	*	.912354
72.5000	.006238	*	.551286	.573713	.621125	.763238	*	*	.875451
70.0000	.004127	*	.582809	.598508	.639496	.754152	*	*	.847891
64.0000	.002447	*	.613042	.623561	.656750	.744684	*	*	.818656
55.0000	.001559	*	.631778	.640374	.667299	.738685	*	*	.798791
48.0000	.001208	*	.639506	.648053	.671644	.736401	*	*	.790080
40.0000	.000942	*	.644753	.654134	.674622	.735250	*	*	.783915
32.0000	.000749	*	.647247	.658425	.676086	.735481	*	*	.780853
25.0000	.000611	*	.647343	.661192	.676209	.737057	*	*	.780745
18.0000	.000481	*	.645249	.663445	.675058	.740610	*	*	.783767
12.0000	.000363	*	.641304	.665537	.672681	.746518	*	*	.790392
6.5025	.000233	*	.635514	.668799	.668799	.757083	*	*	.803252
3.0000	.000123	*	.630546	.673350	.664970	.770601	*	*	.820157
0.4535	.000000	*	.626129	.681665	.660531	.792355	*	*	.847510
0.0000	.000037	*	.625187	.684750	.659129	.799731	*	*	.856718
−0.3666	.000079	*	.624310	.688397	.657387	.807943	.859635	.859635	.866858
−1.0000	.000070	*	.620116	.679833	.651803	.790125	.838690	.880217	.882647
−3.0000	.000059	*	.609650	.662895	.638830	.760915	.804891	.910399	.911063
−5.0000	.000053	*	.600719	.650237	.628166	.741125	.781989	.928384	.928748
−10.0000	.000047	*	.581581	.625024	.605816	.704001	.739292	.955635	.955798
−15.0000	.000044	*	.565405	.604615	.587137	.675004	.706057	.971409	.971513
−20.0000	.000042	*	.551431	.587303	.571003	.650628	.678036	.981467	.981544
−25.0000	.000041	*	.539305	.572436	.556901	.629559	.653581	.988126	.988189
−32.0000	.000042	*	.525106	.555254	.540033	.604458	.623713	.993921	.993975
−40.0000	.000045	*	.512757	.540744	.524393	.581080	.594098	.997483	.997533
−45.0000	.000041	*	.507067	.533761	.516501	.568331	.577375	.998692	.998736
−48.0000	.000036	*	.504249	.529689	.512468	.560828	.567944	.999167	.999205
−50.0000	.000031	*	.502599	.526948	.510078	.555842	.561909	.999403	.999435
−50.6432	.000030	.502109	.502109	.526067	.509362	.554244	.560009	.999467	.999498
−52.0000	.000027	.500900	.501149	.524215	.507936	.550886	.556066	.999585	.999613
−54.0000	.000022	.499383	.499922	.521528	.506048	.545980	.550405	.999722	.999745
−55.0000	.000020	.498752	.499398	.520213	.505203	.543551	.547639	.999777	.999797
−56.0000	.000018	.498206	.498937	.518924	.504425	.541141	.544912	.999822	.999840
−58.0000	.000013	.497365	.498210	.516432	.503075	.536381	.539575	.999892	.999906
−60.0000	.000010	.496856	.497752	.514075	.502000	.531710	.534385	.999939	.999949

Alle Schaltzeiten und optimale Endzeit T, jeweils normiert auf $\bar{T} = 1$. Tabelle (3.4.6)

Wir betrachten also jetzt die Plots für $r_e = 74.9\,\mathrm{cm}$. Deutlich ist die in Abschnitt (3.3) ermittelte Schaltstruktur zu erkennen, die Reihenfolge der Schaltstellen ist $t_0^2 < t_1^2 < t_2^2 < t_1^1 < t_3^2$. Auch der Zusammenhang der Schaltfunktionen λ_1, λ_2 mit den Steuerungen ist klar erkennbar. Man beachte hierbei die Knicke und Sprünge, insbesondere an den Randpunkten des singulären Bogens.

Da die gesamte Bewegung sehr kurz ist, bringt der Schlenker keine wesentliche Zeitersparnis: Radius und Radialgeschwindigkeit haben fast denselben Verlauf wie bei der Grundlösung. Der Einfluß der Tangentialsteuerung u_2 ist nur sichtbar als leichte Nichtlinearität der Radialgeschwindigkeit. Trotzdem ist die Bremsphase ($\approx 81\,\mathrm{ms}$) deutlich kürzer als die Beschleunigungsphase ($\approx 94\,\mathrm{ms}$), der relative Zeitgewinn $1 - T/\bar{T}$ gegenüber der Grundlösung beträgt immerhin schon gut $2\,\%$ (vgl. Tabelle 3.4.6).

Es sei noch darauf hingewiesen, daß der Plot für λ_3 im unteren Bereich ziemlich „krakelig“ aussieht und sogar Sprünge zu haben scheint. Dies ist einfach darauf zurückzuführen, daß λ_3 hier bis zur fünften Dezimalstelle konstant ist. Die zwei restlichen Dezimalen werden in der Zeichnung über ca. einen Zentimeter ausgebreitet, so daß die Maschinengenauigkeit (`REAL*4`!) sichtbar wird.

Es folgen die Plots der Lösungen für $r_e \in \{74\,\text{cm}, 70\,\text{cm}, 55\,\text{cm}, 40\,\text{cm}, 25\,\text{cm}\}$. Man achte insbesondere auf die Entwicklung der Variablen λ_2, u_2, und $\dot{\theta}_1$ im Verlauf der Homotopie. Die positiven Bögen von λ_2 vor und nach der Nullinie des singulären Bogens flachen rasch ab und sind bald nicht mehr zu erkennen. Der Verlauf der Steuerung u_2 zeigt, daß der singuläre Bogen immer breiter wird und daß u_2 dort gegen Null konvergiert. Die letzten Schaltstellen rücken dichter zusammen, der Startbang wird immer kürzer. Für $r_e < 70\,\text{cm}$ ist er in der Zeichnung (als senkrechter Strich) nur noch sichtbar, weil der Zeitnullpunkt ein wenig vom linken Rand abgesetzt wurde. Übrigens ist der Quotient $|\lambda_4/t_1| \approx 87$ während der ganzen Homotopie annähernd konstant. Man kann deshalb die Entwicklung von t_1 am besten an der adjungierten Variablen λ_4 verfolgen.

Das gerade beschriebene Verhalten der Steuerung u_2 bewirkt, daß ihr (unerwüschter) bremsender Einfluß während der Beschleunigungsphase von u_1 mit fallendem Endradius immer geringer wird. Andererseits wird die erwünschte Bremswirkung des Schlenkers immer stärker, so daß die Bremsphase von u_1 später beginnen kann. Diese Effekte zeigen sich am deutlichsten im Verhalten der Radial- und Winkelgeschwindigkeit sowie am Schaltpunkt von u_1, außerdem in der Kurve für den Radius r. Bei $r_e = 25\,\text{cm}$ beträgt die Zeitersparnis durch den Schlenker nahezu 22 %.

Wir fahren fort mit den Lösungen für $r_e \in \{12\,\text{cm}, 6.5025\,\text{cm}, 3\,\text{cm}, 0.4535\,\text{cm}\}$. Hier steigt λ_2 rechts vom singulären Bogen wieder deutlich an. Dessen relative Breite geht zurück, wobei die letzten Schaltstellen etwas auseinanderrücken, so daß die Dauer des Schlenkers wächst. Die Schaltzeit t_1 hingegen strebt gegen Null, was an λ_4 deutlich sichtbar ist. Zugleich geht die Steuerung u_2 auf dem singulären Bogen gegen Null. Bei $r_e = 0.4535\,\text{cm}$ schließlich liegt die Schaltstruktur (3.3.8) vor: der Schaltpunkt t_1 und der Startbang sind ganz verschwunden, es gilt $u_2 = 0$ auf $[0, t_3]$ sowie $\lambda_4 = 0$ und (formal) $t_1 = 0$. Die optimale Startposition für den Schlenker liegt genau bei diesem Endradius auf der x-Achse ($\theta_1 = 0$).

Derartige *Grenzfälle* mit veränderter Schaltstruktur kann man mit sogenannten *Grenzfallalgorithmen* behandeln (Bock [2]). Dazu löst man ein entsprechend verändertes Randwertproblem, welches den jeweiligen Sonderfall eindeutig charakterisiert. In der genannten Situation beispielsweise fällt die Schaltstelle t_1 weg. Die Endbedingung $r(T) = r_e$ wird durch $\lambda_4(T) = 0$ ersetzt, wie schon in Abschnitt (3.3) erwähnt. Auf diese Weise konnten eine korrekte Lösung und der zunächst *unbekannte* Endradius ermittelt werden.

Ein weiterer Grenzfall liegt bei $r_e = 6.5025\,\text{cm}$ vor, wo $t_4 = t_2^2$ und $t_5 = t_1^1$ die Reihenfolge tauschen. Hier gestattet die innere Punktbedingung $t_4 = t_5$ statt $r(T) = r_e$ eine exakte Behandlung des Problems.

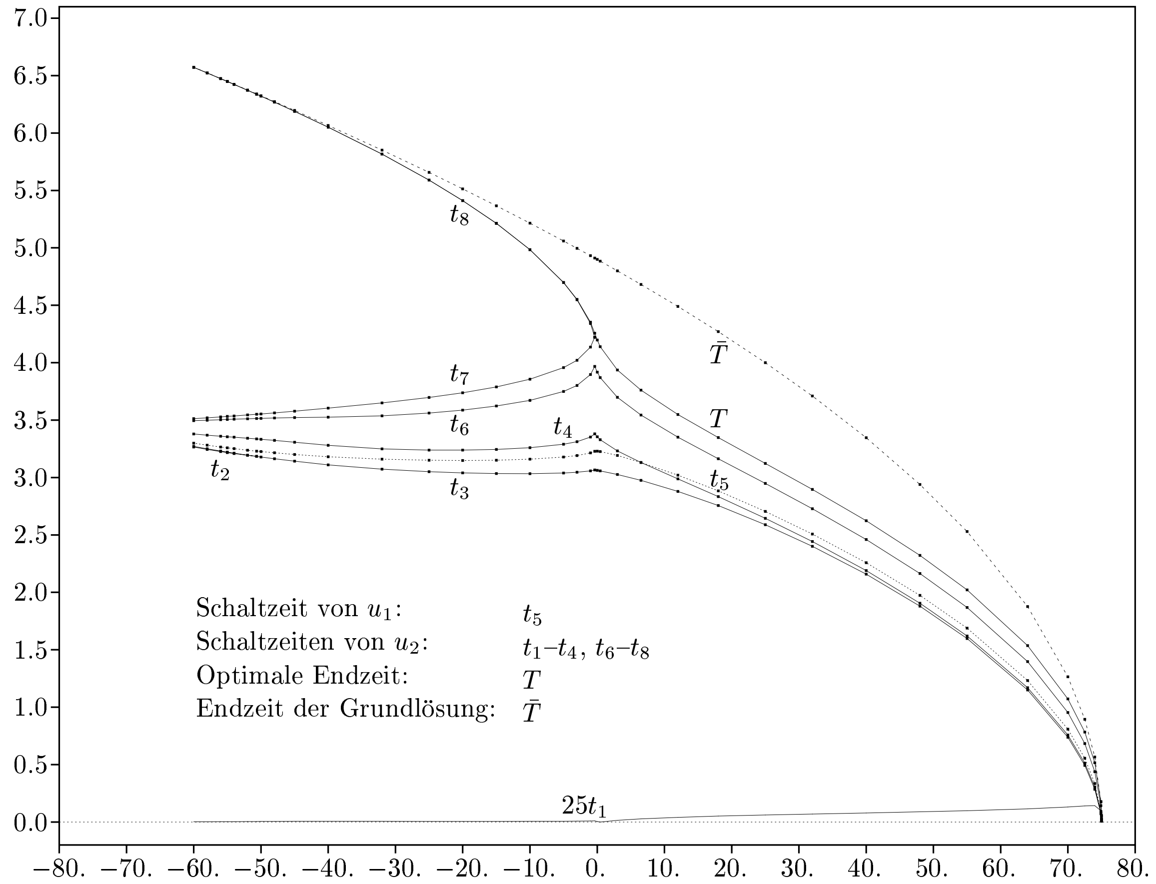
Obwohl der Schlenker die Beschleunigungsphase im gerade beschriebenen Bereich immer weniger stört, geht die Zeitersparnis insgesamt doch zurück: bei $r_e = 0.4535\,\text{cm}$ beträgt sie nur noch gut 15 %. Die Erklärung für den merklichen Rückgang der Bremswirkung ist einfach: der Schlenker findet bei kleineren Radien statt als zuvor, so daß die erzielte Fliehkraft (der Term $\frac{1}{2}sz_2^2$ in 2.5.6) abnimmt.

2) Homotopie von $0.4535\,\text{cm}$ *bis* $-0.3666\,\text{cm}$.

Beim nächsten Fall wird der Arm genau in die Symmetrieposition $r_e = 0$ gefahren. Hier ist der Startbang von u_2 negativ, ebenso die singuläre Steuerung, die Winkelgeschwindigkeit und der Winkel auf $[t_1, t_3]$. Der Armschwerpunkt wird entsprechend vor Beginn des Schlenkers etwas nach „links“ bewegt, was an den beiden Bahnkurven deutlich zu erkennen ist.

Die adjungierte Variable λ_4 ist jetzt ebenfalls negativ, denn der Quotient λ_4/t_1 hat sein Vorzeichen bei $0.4535\,\text{cm}$ sprunghaft gewechselt. Dies entspricht dem Vorzeichenwechsel von $u_2(0)$ an derselben Stelle.

Rechnet man nun einfach weiter, so stellt man bald fest, daß die Schaltfunktion λ_2 nach t_6 zwei neue Nullstellen hat, zwischen denen sie im negativen Bereich verläuft. Auf $[t_6, T]$ gilt jedoch $u_2 = +\hat{u}_2 > 0$, die Lösung des Mehrpunktrandwertproblems ist also mit der vorgegebenen Schaltstruktur inkonsistent. Wie schon zuvor angedeutet, sollte für $r_e < 0$ der Schlenker tatsächlich nicht bis zum Ende der Bewegung



Schaltzeiten t_i und Endzeiten über Endradius r_e, komplette Homotopie. Abbildung (3.4.7)

dauern, weil sonst der Bremsvorgang behindert würde. Deshalb ist zu erwarten, daß zwischen t_6 und T ein neuer singulärer Bogen auftritt, dem ganz zum Schluß wieder ein kurzer bang folgt. Denkbar, aber aus physikalischen Gründen weniger wahrscheinlich, wäre auch ein Sprung der Steuerung u_2 von $+\hat{u}_2$ nach $-\hat{u}_2$ und eventuell zurück.

Auf jeden Fall müssen ein oder zwei neue Schaltstellen eingefügt werden. Auch hier findet man die neue Schaltstruktur wieder durch Ausprobieren: das Intervall $[t_6, T]$ wird durch zwei zusätzliche Schaltstellen t_7, t_8 zerlegt. Zwischen diesen ist u_2 singulär (und positiv), auf $[t_6, t_7]$ und $[t_8, T]$ gilt wie zuvor $u_2 = +\hat{u}_2$.

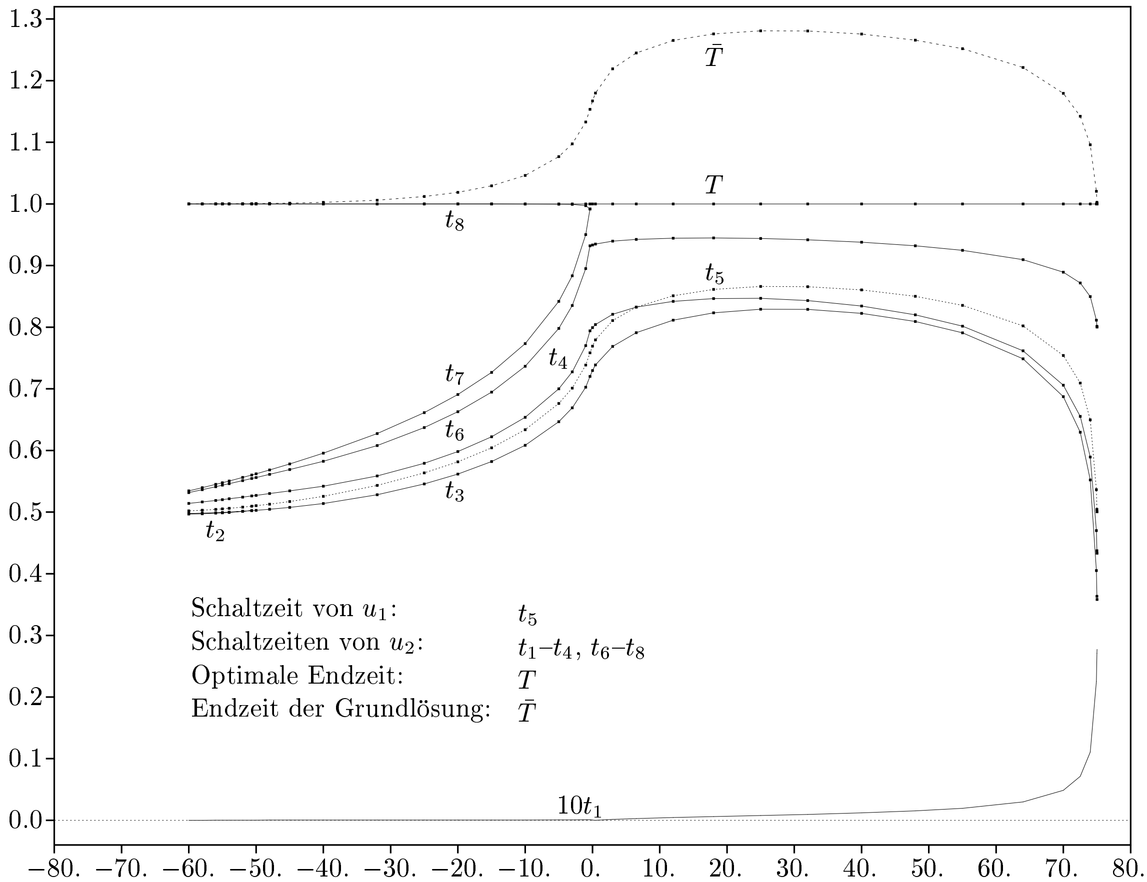
Erst nachdem eine Lösung mit dieser Schaltstruktur als optimal erkannt war, konnte der Grenzfall $r_e = -0.3666\,\text{cm}$ bestimmt werden, bei dem der singuläre Bogen gerade auftaucht ($t_7 = t_8$). Hier besitzt λ_2 eine doppelte Nullstelle. Die Bedingung $r(T) = r_e$ ist also durch die innere Punktbedingung $\dot{\lambda}_2(t_8) = 0$ zu ersetzen, t_7 fällt weg.

In der Darstellung von λ_2 ist deutlich zu sehen, wie die Nullinie knapp vor dem rechten Rand von oben berührt wird. Die Steuerung u_2 weist in t_8 einen senkrechten Strich auf, dessen unteres Ende den notwendigen Sprung beim Übergang zum singulären Bogen anzeigt. Es liegt sozusagen ein zum Punkt entarteter singulärer Bogen vor mit $u_2(t_8-) = \hat{u}_2 = u_2(t_8+)$ und $u_2(t_8) < \hat{u}_2$.

3) Homotopie von $-3.666\,\text{cm}$ bis $-50.6432\,\text{cm}$.

Von hier an bleibt die Schaltstruktur wieder in einem großen Bereich konstant. Das Verhalten der Lösungen läßt sich an den Darstellungen für $r_e \in \{-1\,\text{cm}, -5\,\text{cm}, -15\,\text{cm}, -32\,\text{cm}, -45\,\text{cm}, -50\,\text{cm}\}$ gut verfolgen. Der neue singuläre Bogen wird rasch breiter, der bang danach immer kürzer; er ist im Plot bei $r_e = -1\,\text{cm}$ gerade noch als solcher zu erkennen, danach nur noch als senkrechter Strich.

Auf dem größten Teil der beiden singulären Bögen ist die Steuerung ab $r_e = -15\,\text{cm}$ nicht mehr von Null zu unterscheiden, nur in unmittelbarer Nähe des Schlenkers bewegt sie sich steil auf den benachbarten bang zu. Dieses Verhalten wird immer ausgeprägter, während der Schlenker immer mehr



Schaltzeiten t_i und Endzeiten (normiert auf $T = 1$) über Endradius r_e. Abbildung (3.4.8)

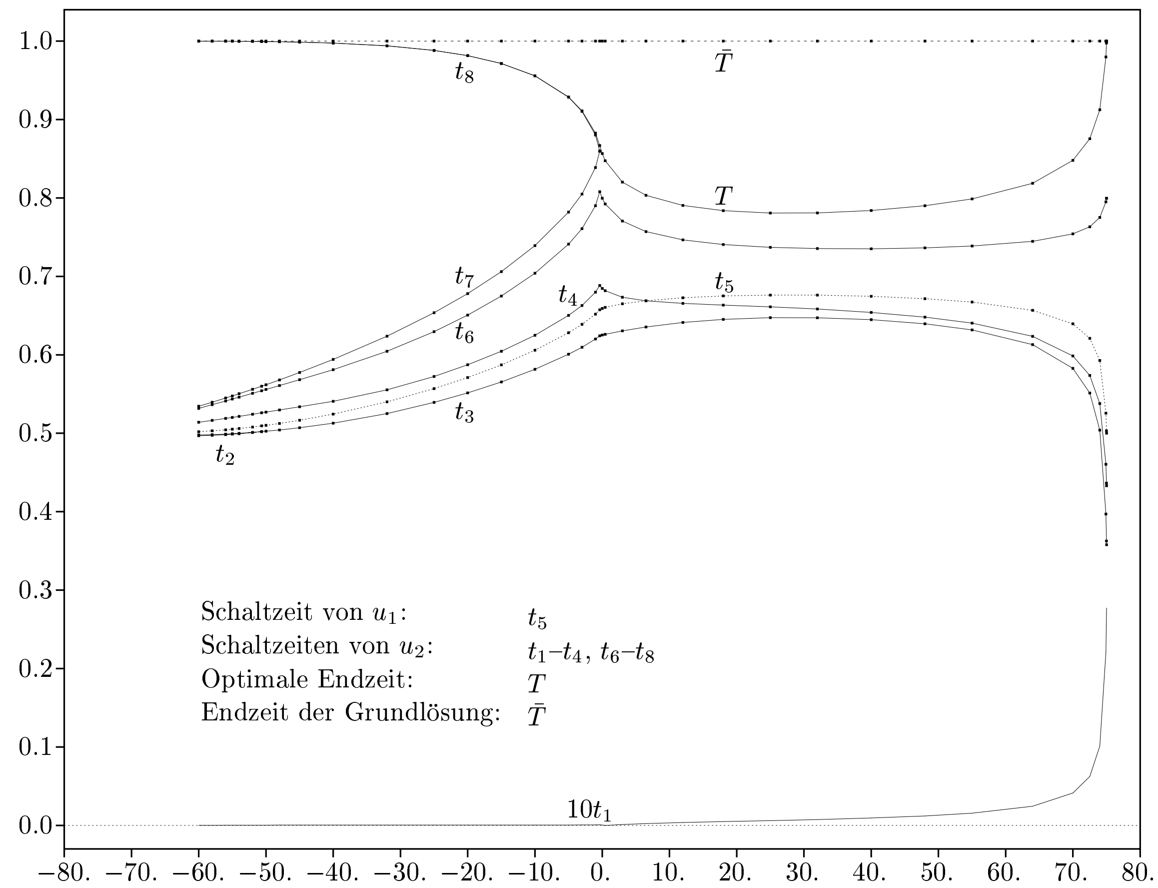
zur Mitte des Intervalls rückt und seine Dauer von etwa 1.2 s auf etwa 0.5 s zurückgeht. Die erzielte Zeitersparnis wird ebenfalls rasch geringer. Betrug sie bei $r_e = -0.3666\,\text{cm}$ noch über 13 %, so ist es bei $r_e = -25\,\text{cm}$ gerade noch gut 1 %, bei $r_e = -50\,\text{cm}$ sogar nur noch 0.06 %. Trotzdem ist die zusätzliche Bremswirkung zu Beginn der Bremsphase noch deutlich zu erkennen an der größeren Steigung der Radialgeschwindigkeit kurz nach dem Knick.

Besonders auffällig ist die Entstehung einer tiefen Spitze der Tangentialsteuerung am Ende des ersten singulären Intervalls, die bei $r_e = -50\,\text{cm}$ bis $-290\,\text{Nm}$ reicht, also schon fast den unteren Rand des Steuerbereichs trifft. Dies läßt eine erneute Änderung der Schaltstruktur erwarten. Tatsächlich würde andernfalls schon für $r_e = -51\,\text{cm}$ die „singuläre“ Steuerung den Steuerbereich verlassen. Zwischen dem singulären Bogen und dem Sprung nach $+\hat{u}_2$ wird deswegen ein neuer bang eingefügt, auf dem u_2 den Wert $-\hat{u}_2$ annimmt. In der Schaltstruktur findet somit eine *Verzweigung* statt, sobald die Spitze den unteren Rand des Steuerbereichs erreicht.

Die Plots für $r_e = -50.6432\,\text{cm}$ zeigen diesen Grenzfall. Dabei liegt die eigenartige Situation vor, daß die singuläre Steuerung den Rand des Steuerbereichs erreicht, aber aufgrund der Unstetigkeitsbedingung sofort zum anderen Rand springt. Der neue bang entsteht dadurch, daß die Tangentialsteuerung für $r_e < -50.6432\,\text{cm}$ schon kurze Zeit vor Erreichen des unteren Randes nach $-\hat{u}_2$ *springt* und wenig später nach $+\hat{u}_2$. Der obengenannte Grenzfall ist also durch die Bedingung festgelegt, daß die singuläre Steuerung den Wert $-\hat{u}_2$ annimmt: $u_2^{\text{sing}}(t_3) = -\hat{u}_2$ anstelle von $r(T) = r_e$.

4) Homotopie von $-50.6432\,\text{cm}$ bis $-60\,\text{cm}$.

Auf den letzten sechs Seiten sind die Lösungen für $r_e \in \{-52\,\text{cm}, -56\,\text{cm}, -60\,\text{cm}\}$ zu sehen. Die Auflösung der Plots ist aber zu niedrig, um den extrem kurzen neuen bang noch sichtbar zu machen. Überhaupt sind die Verhältnisse jetzt so extrem, daß man die wesentlichen Strukturmerkmale nur noch mit viel Phantasie den graphischen Darstellungen entnehmen kann.



Schaltzeiten t_i und Endzeiten (normiert auf $\bar{T} = 1$) über Endradius r_e. Abbildung (3.4.9)

Auch numerisch treten nun erhebliche Probleme auf. Mit 25 Mehrzielintervallen war für $r_e < -60$ cm keine Lösung mehr mit ausreichender Genauigkeit berechenbar. Man beachte dabei, daß die optimale Lösung bei $r_e = -60$ cm nur um 0.005 % schneller ist als die Grundlösung! Die Abbruchgenauigkeit muß also schon bei mindestens sechs Dezimalstellen liegen, um hier einigermaßen verläßlich einen Unterschied feststellen zu können. Ein Weiterziehen der Homotopie wäre also nur noch mit erheblichem numerischem Aufwand möglich.

Nun ist aber die Schaltstruktur für $r_e < -50.6432$ cm *symmetrisch*: Dem negativen bang am Anfang steht ein positiver bang am Ende gegenüber, dem negativen singulären Bogen danach entspricht ein positiver singulärer Bogen vor dem Schlußbang usw. Diese Symmetrie läßt es als sehr unwahrscheinlich erscheinen, daß sich die Schaltstruktur vor dem Endpunkt der Homotopie bei $r_e = -75$ cm (mit bekannter Lösung!) nochmals ändert. Damit kann das in Abschnitt (3.1) gestellte Problem im wesentlichen als gelöst betrachtet werden. Angesichts des Kosten-Nutzen-Verhältnisses weiterer Rechnungen wurde deshalb die Homotopie an dieser Stelle abgebrochen.

Eine anschauliche Vorstellung der Schaltzeiten im Verlauf der kompletten Homotopie vermitteln die Abbildungen (3.4.7–3.4.9). Alle Zeitpunkte der Tabellen (3.4.4–3.4.6) sind in ihnen über dem jeweiligen Endradius r_e aufgetragen und zu Polygonzügen verbunden. Für die Schaltzeiten von u_2 wurden die Verbindungsstrecken durchgezeichnet, während die Nullinie, die Schaltstelle t_5 von u_1 und $\bar{T}$ als Vergleichswerte gestrichelt dargestellt sind. Damit man auch den Verlauf von t_1 einigermaßen erkennen kann, wurde diese Schaltzeit in Abbildung (3.4.7) mit dem Faktor 25 multipliziert, in den Abbildungen (3.4.8/3.4.9) mit dem Faktor 10. Besonders auffällig tritt das Auftauchen des zweiten singulären Bogens bei $r_e = -0.3666$ cm hervor, auch der Wechsel von t_4 und t_5 bei $r_e = 6.5025$ cm ist deutlich sichtbar. Die „Nullstelle“ von t_1 bei $r_e = 0.4535$ cm ist nur in Abbildung (3.4.7) erkennbar, die Bifurkation bei $r_e = -50.6432$ cm kann man in allen drei Abbildungen höchstens noch erahnen. Sehr schön ist dagegen die Zeitersparnis durch den Schlenker ablesbar, vor allem in Abbildung (3.4.9).

Insgesamt haben wir im Bereich $r_e \in [-60\,\text{cm}, 75\,\text{cm}[$ fünf verschiedene Schaltstrukturen mit vier Grenzstrukturen gefunden. Letztere sind jeweils durch charakteristische Bedingungen gekennzeichnet. Die zugehörigen Endradien kann man berechnen, indem man die Endbedingung $r(T) = r_e$ durch die jeweilige spezielle Bedingung ersetzt.

In Tabelle (3.4.10) sind alle neun Fälle zusammengestellt. '+' und '−' geben darin einen positiven bzw. negativen bang der Tangentialsteuerung u_2 an, s_σ bezeichnet einen singulären Bogen mit Vorzeichen σ. Die beiden eingeklammerten Einträge in den Grenzfällen 6 und 8 beziehen sich auf Bögen der Länge Null. '++' und '−−' geben denjenigen bang von u_2 an, auf dem die Schaltstelle t_5 der Radialsteuerung u_1 liegt. Für jeden der vier Grenzfälle sind der Endradius r_e und die jeweilige charakteristische Bedingung eingetragen. Die letzten beiden Spalten der Tabelle schließlich beinhalten die Anzahl der Schaltstellen von u_2 sowie ihre Gesamtzahl.

Nr.	r_e	$0 \quad t_1 \quad t_2 \quad t_3 \quad t_4 \quad t_6 \quad t_7 \quad t_8 \quad T$								Bed.	$\sharp\{t_i^2\}$	$\sharp\{t_i\}$
9		$-$	s_-	$-$	$++$	$-$	$+$	s_+	$+$		7	8
8	-50.6432	$-$	s_-	$(-)$	$++$	$-$	$+$	s_+	$+$	$u_2^{\text{sing}}(t_3) = -\hat{u}_2$	$7-1$	7
7		$-$	s_-		$++$	$-$	$+$	s_+	$+$		6	7
6	-0.3666	$-$	s_-		$++$	$-$	$+$	(s_+)	$+$	$\dot{\lambda}_2(t_8) = 0$	$6-1$	6
5		$-$	s_-		$++$	$-$	$+$				4	5
4	0.4535	s_0			$++$	$-$	$+$			$\lambda_4 = 0$	3	4
3		$+$	s_+		$++$	$-$	$+$				4	5
2	6.5025	$+$	s_+		$+$	$-$	$+$			$t_4 = t_5$	$4-1$	4
1		$+$	s_+		$+$	$--$	$+$				4	5

Die verschiedenen Schaltstrukturen im Homotopieverlauf. Tabelle (3.4.10)

4. Anhang

Nachdem die Homotopie mit verschiedenen Programmen (entsprechend den verschiedenen Schaltstrukturen) zum erstenmal durchlaufen war, wurde das auf den folgenden Seiten abgedruckte Programm geschrieben, welches sämtliche Fälle einheitlich behandelt. Dazu wurde im wesentlichen die Strukturtabelle (3.4.10) in einer Matrix `MSW` gespeichert.

Die Anzahl der Mehrzielintervalle, die von zehn bei $r_e = 25\,\mathrm{cm}$ (vgl. Abschnitt 3.3) bis auf 25 bei $r_e = -60\,\mathrm{cm}$ (vgl. Abschnitt 3.4) gestiegen war, wurde einheitlich auf 25 gesetzt. Die acht numerischen Schaltstellen sitzen genau in der Mitte der Intervalle 2, 8, 10, 12, 14, 16, 18 und 24, so daß I^* in insgesamt 33 Teilintervalle zerlegt ist.

Der Einfachheit halber wird stets mit allen acht Schaltzeiten gerechnet. Schaltstellen, die im Mehrpunktrandwertproblem nicht vorkommen, werden jeweils durch künstliche Bedingungen zwischen den benachbarten Zeiten „eingeklemmt“. Für die Wahl der richtigen Bedingungen sorgt der Strukturindex `ISTR`, der in Tabelle (3.4.10) in der ersten Spalte steht.

Für $\texttt{ISTR} \leq 6$ existiert der zweite singuläre Bogen nicht, die inneren Punktbedingungen für t_7 und t_8 werden deshalb ersetzt durch

$$t_7 = t_8 \quad (\texttt{ISTR} \leq 6) \qquad \text{und} \qquad t_8 = 0.99T \quad (\texttt{ISTR} < 6).$$

Die innere Punktbedingung für t_2 wird entsprechend für $\texttt{ISTR} \leq 8$ ersetzt durch

$$t_2 = t_3.$$

Im Grenzfall $\texttt{ISTR} = 4$ existiert der Startbang nicht. Anstelle der inneren Punktbedingung für t_1 wird hier gefordert

$$t_1 = 4 \cdot 10^{-4}.$$

Diese Bedingungen sind so gewählt, daß im Verlauf der Homotopie der Übergang zu den jeweiligen inneren Punktbedingungen und umgekehrt möglichst nahtlos erfolgen kann.

Zur Lösung eines Randwertproblems mit OPCON (der Unterprogrammname lautet OPTI) sind mindestens die folgenden vier problemabhängigen Prozeduren erforderlich:

• *Das Hauptprogramm*
muß alle organisatorischen Aufgaben erfüllen, die zur Ansteuerung des universell einsetzbaren Programmpakets OPCON benötigt werden. Dazu gehört auf jeden Fall die Dimensionierung des Mehrpunktrandwertproblems, die Definition des Mehrzielgitters Γ^* und die Bereitstellung einer (diskreten) Starttrajektorie, außerdem die Angabe der gewünschten Abbruchgenauigkeit und der Maximalzahl von GAUSS-NEWTON-Iterationen. Weiter müssen übergeben werden ein workspace array, die Namen der drei anderen problemabhängigen Prozeduren sowie der Name eines Unterprogramms für die numerische Integration.

Über `COMMON`-Blöcke kann man schließlich auf diverse Steuerparameter zugreifen. Auf diese Weise lassen sich zusätzliche Kenntnisse des Problems an OPCON weitergeben, z.B. um in Sonderfällen unnötigen Aufwand zu vermeiden oder um die Konvergenz zu verbessern.

• *Das Unterprogramm für die rechte Seite* (hier `FROB`)
berechnet gemäß (3.2.3) in einem gegebenen Zeitpunkt $t^* \in I^*$ die rechte Seite $f_i^*(t^*, z^*(t^*), p^*)$.

• *Das Unterprogramm für die Randbedingungen* (hier `RROB`)
wertet die Randbedingungen aus und berücksichtigt eventuelle innere Punktbedingungen. Letztere werden in der folgenden Prozedur berechnet und mit dem `COMMON`-Block `/MPBVP/` übergeben.

• *Das Unterprogramm für die Schaltfunktionen* (hier `SROB`)
berechnet zu einem gegebenen Zeitpunkt die Schaltfunktionen und ihre zeitlichen Ableitungen (diese können wahlweise auch automatisch erzeugt werden); gegebenenfalls wertet es auch innere Punktbedingungen aus, um sie an `RROB` zu übergeben.

Diese vier Routinen sind hier noch durch den `BLOCK DATA PRHOM` ergänzt, der die wichtigsten globalen Variablen für die Homotopie initialisiert.

```
C ******************************************************************** PRHOM001
C     Global data for horizontal polar robot                           PRHOM002
C     Pulling arm in, homotopy with respect to final radius            PRHOM003
C ******************************************************************** PRHOM004
      BLOCK DATA PRHOM                                                 PRHOM005
C     Moments of inertia, arm length & masses of robot                 PRHOM006
      COMMON   /INERT/ I323, I31, L, MB, ML, MLB                       PRHOM007
      REAL*8   I323/18.5D0/, I31/0D0/, L/75D-2/                        PRHOM008
      REAL*8   MB/40D0/, ML/0D0/, MLB/40D0/                            PRHOM009
C     Boundary values, maximal control values & structural information PRHOM010
      COMMON   /BOUND/ XA, XB  /CNTRL/ D, ISTR  /STRUC/ FI, JI, IQ, MSW PRHOM011
      REAL*8   XA(4)/2*0D0,75D-2,0D0/, XB(4)/4*0D0/, D(2)/5D0,300D0/   PRHOM012
C     Length & number of subintervals between each two switching points PRHOM013
      REAL*8   FI(33)/2*6D-2,7*24D-2,15*8D-2,7*24D-2,2*6D-2/           PRHOM014
      INTEGER  JI(33)/2*1,7*2,3*3,3*4,3*5,3*6,3*7,7*8,2*9/             PRHOM015
C     Permutation of switching pts, structure index & structure matrix PRHOM016
      INTEGER        IQ(9)/1,2,3,5,4,6,7,8,9/, ISTR/5/, MSW(9,9)       PRHOM017
     @ /3,2,2,3,0,0,3,3,3, 3,2,2,3,0,0,3,3,3, 3,2,2,3,3,0,3,3,3,       PRHOM018
     @  1,1,1,3,3,0,3,3,3, 0,2,2,3,3,0,3,3,3, 0,2,2,3,3,0,3,2,3,       PRHOM019
     @  0,2,2,3,3,0,3,2,3, 0,2,0,3,3,0,3,2,3, 0,2,0,3,3,0,3,2,3/       PRHOM020
      END                                                              PRHOM021
C ******************************************************************** PRHOM022
C     Main program for horizontal polar robot                          PRHOM023
C ******************************************************************** PRHOM024
      PARAMETER (MN=26, MT=34, NV=17, ND=7, NB=17, NSW=8)              PRHOM025
      PARAMETER (IPROV=56000, ITMAX=16, KPR=2)                         PRHOM026
      REAL*8    T(MT), Z(NV,MT), W(IPROV), ACC/1D-9/                   PRHOM027
      EXTERNAL  FROB, RROB, SROB, STIFF                                PRHOM028
C     Common variables for OPTI fine tuning                            PRHOM029
      COMMON   /OPCON/ IQDIFF, ID1, DUM1, KS  /BOUND/ XA, XB           PRHOM030
      COMMON   /STEST/ ISTIFF, ID2, DUM2      /CNTRL/ D, ISTR          PRHOM031
      COMMON   /SWITCH/ NSWIX                 /STRUC/ FI, JI, IQ, MSW  PRHOM032
      INTEGER  IQDIFF, ID1, KS, ISTIFF, ID2, ISTR                      PRHOM033
      INTEGER  NSWIX(50), JI(33), IQ(9), MSW(9,9), I, J, K, KFLAG      PRHOM034
      REAL*8   DUM1(11), DUM2(52), XA(4), XB(4), D(2), FI(33)          PRHOM035
C                                                                      PRHOM036
C     MN = number of multiple shooting nodes                           PRHOM037
C     MT            nodes & switching points                           PRHOM038
C     NV            variables                                          PRHOM039
C     ND            differential equations                             PRHOM040
C     NB            boundary conditions                                PRHOM041
C     NSW           switching functions                                PRHOM042
C                                                                      PRHOM043
C     IPROV = size of OPTI workspace array                             PRHOM044
C     ITMAX = maximal number of iterations                             PRHOM045
C     KPR   = OPTI print parameter                                     PRHOM046
C     KFLAG = OPTI return code                                         PRHOM047
C                                                                      PRHOM048
C     T(MT)     = multiple shooting nodes & switching points           PRHOM049
C     Z(NV,MT) = discrete trajectory (including parameters)            PRHOM050
C     W(IPROV) = OPTI workspace array                                  PRHOM051
C     ACC      = required accuracy                                     PRHOM052
C                                                                      PRHOM053
C     FROB = SUBROUTINE for right hand side                            PRHOM054
C     RROB                  boundary conditions                        PRHOM055
C     SROB                  switching functions                        PRHOM056
C     STIFF                 numerical integration                      PRHOM057
C                                                                      PRHOM058
```

```
C     Read problem specification from console
      WRITE(6,600) XB(3), ISTR, ACC
  600 FORMAT(' Enter final radius (default:',D11.3,'),',
     @       ' structure index (default:',I3,')'/
     @       ' & required accuracy (default:',1PD8.1,'), please.')
      READ(5,*,END=10) XB(3), ISTR, ACC
C     Read initial trajectory from disk file
   10 READ(13,500) K, T(1), (Z(I,1),I=1,NV)
  500 FORMAT(/I2,D19.8,(T22,3D17.8))
      DO 30 J = 2, MN
   20    READ(13,500) K, T(J), (Z(I,J),I=1,ND)
         IF (K.NE.0) GOTO 20
   30 CONTINUE
C     Set switch & jump index vector (0 = time switch without jump)
      DO 40 I = 1, NSW
         NSWIX(I) = 0
   40 CONTINUE
C     Set scale factors for transformation on unit interval (cf FROB)
      DO 50 I = 1, 33
         FI(I) = 1/FI(I)
   50 CONTINUE
C     Exchange T4 and T5 if necessary
      IF (ISTR.LE.2) THEN
         IQ(4) = 4
         IQ(5) = 5
      END IF
C     Set some OPTI fine tuning values to improve convergence
      DUM1( 9) = 1D-12
      DUM1(10) = 1D-08
C     Disable automatic differentiation of switching functions
      IQDIFF = 0
C     Choose numerical integrator (1 = DIFSYS, 2 = METANB)
      ISTIFF = 1
C
      CALL OPTI ( FROB, RROB, SROB, STIFF, T, Z, NV, ND,
     @ NB, MN, MT, NSW, ACC, ITMAX, KPR, KFLAG, W, IPROV )
C
C     Save structural information for plot routine
      WRITE(6,'(20I4)') ISTR, (IQ(I),I=1,9)
      STOP
      END
```

```
C ******************************************************************** PRHOM100
C     Right hand side of robot differential equations                  PRHOM101
C ******************************************************************** PRHOM102
      SUBROUTINE FROB ( T, Z, DZ )                                     PRHOM103
C                                                                      PRHOM104
C     Contents of state vector  Z  are                                 PRHOM105
C     Z(1- 2)  velocity vector  X1                                     PRHOM106
C     Z(3- 4)  position vector  X2                                     PRHOM107
C     Z(5- 6)  velocity adjoint vector  LAM1                           PRHOM108
C     Z(7- 8)  position adjoint vector  LAM2                           PRHOM109
C     Z(9-16)  switching point  T1-T8                                  PRHOM110
C     Z(  17)  (free) end time  TT                                     PRHOM111
C                                                                      PRHOM112
C     Counts for procedure calls & OPTI global variables               PRHOM113
      COMMON   /COUNT/ ICF, ID1  // J, M, JU                           PRHOM114
      INTEGER  ICF, ID1(4), J, M, JU(18,80)                            PRHOM115
      COMMON   /INERT/ I323, I31, L, MB, ML, MLB                       PRHOM116
      REAL*8   I323, I31, L, MB, ML, MLB                               PRHOM117
      COMMON   /CNTRL/ D, ISTR  /STRUC/ FI, JI, IQ, MSW                PRHOM118
      INTEGER  ISTR, JI(33), IQ(9), MSW(9,9), I                        PRHOM119
      REAL*8   D(2), FI(33), T, Z(17), DZ(7), A1, S, U1, U2, FCT       PRHOM120
C                                                                      PRHOM121
C     Count FROB calls                                                 PRHOM122
      ICF = ICF+1                                                      PRHOM123
C     Compute  S, A1  and optimal control  U                           PRHOM124
      S  = (MLB*Z(3)+ML*L)*2                                           PRHOM125
      A1 = I31+I323+ML*L*L+(S/2+ML*L)*Z(3)                             PRHOM126
      U1 = D(1)                                                        PRHOM127
      IF (JU(5,J).LT.0) U1 = -U1                                       PRHOM128
      GOTO (10,20,30) MSW(JI(J),ISTR)                                  PRHOM129
C     Negative bang (MSW=0)                                            PRHOM130
      U2 = -D(2)                                                       PRHOM131
      GOTO 40                                                          PRHOM132
C     Zero singular control (MSW=1)                                    PRHOM133
   10 U2 = 0                                                           PRHOM134
      GOTO 40                                                          PRHOM135
C     Nonzero singular control (MSW=2)                                 PRHOM136
   20 U2 = Z(2)*(Z(7)/Z(5)*A1+(S-2*MLB*A1/S)*Z(1))                     PRHOM137
      GOTO 40                                                          PRHOM138
C     Positive bang (MSW=3)                                            PRHOM139
   30 U2 = D(2)                                                        PRHOM140
C     Compute right hand side  DZ                                      PRHOM141
   40 DZ(1) = (U1+S*Z(2)**2/2)/MLB                                     PRHOM142
      DZ(2) = (U2-S*Z(1)*Z(2))/A1                                      PRHOM143
      DZ(3) = Z(1)                                                     PRHOM144
      DZ(4) = Z(2)                                                     PRHOM145
      DZ(5) =                  Z(6)*Z(2)*S/A1-Z(7)                     PRHOM146
      DZ(6) = -Z(5)*Z(2)*S/MLB+Z(6)*Z(1)*S/A1-Z(8)                     PRHOM147
      DZ(7) = -Z(5)*Z(2)**2   +Z(6)*(2*MLB*Z(1)*Z(2)+DZ(2)*S)/A1       PRHOM148
C     Transformation of right hand side on unit interval               PRHOM149
      FCT = FI(J)*Z(IQ(JI(J))+8)                                       PRHOM150
      IF (JI(J).GT.1) FCT = FCT-FI(J)*Z(IQ(JI(J)-1)+8)                 PRHOM151
      DO 50 I = 1, 7                                                   PRHOM152
         DZ(I) = FCT*DZ(I)                                             PRHOM153
   50 CONTINUE                                                         PRHOM154
      RETURN                                                           PRHOM155
      END                                                              PRHOM156
```

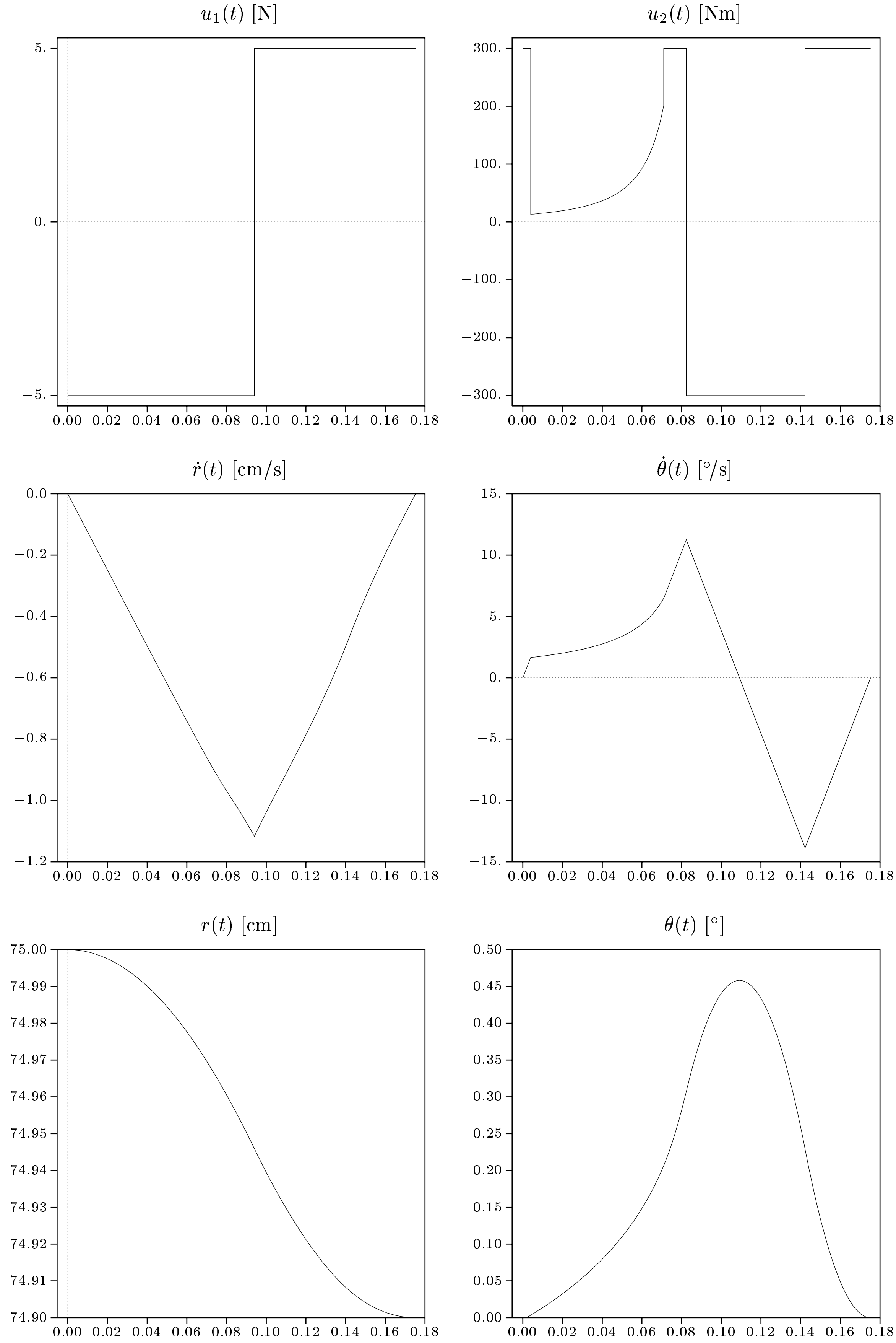
```
C **********************************************************************
C     Boundary & interior point conditions
C **********************************************************************
      SUBROUTINE RROB ( ZA, ZB, W, NFIX )
C
      COMMON   /MPBVP/ ND1, ZI, ZIS  /COUNT/ ID1, ICR, ID2  // J, M, JU
      COMMON   /BOUND/ XA, XB  /CNTRL/ D, ISTR  /STRUC/ FI, JI, IQ, MSW
      INTEGER  ID1(2), ICR, ID2(2), ND1(2), J, M, JU(18,80)
      INTEGER  ISTR, JI(33), IQ(9), MSW(9,9), NFIX(17), I
      REAL*8   ZI(200), ZIS(200), XA(4), XB(4), D(2), FI(33)
      REAL*8   ZA(17), ZB(17), W(17), T/0D0/, DZ(7)
C
C     ZA, ZB   =  start and final values
C     ZI, ZIS  =  interior  point values (cf SROB)
C
C     Count RROB calls
      ICR = ICR+1
C     Save common variable  J  and compute  DZ(0,Z(0))
      I = J
      J = 1
      CALL FROB ( T, ZA, DZ )
      J = I
C     Boundary conditions for position and velocity,
C     interior point conditions Q(T1)=...=Q(T8) = 0
C     & HAMILTON function (as minimizing condition)
      DO 10 I = 1, 4
         W(I   ) = ZA(I)-XA(I)
         W(I+ 4) = ZB(I)-XB(I)
         W(I+ 8) = ZI(I)
         W(I+12) = ZI(I+4)
   10 CONTINUE
      W(17) = (ZA(5)*DZ(1)+ZA(6)*DZ(2)
     @        +ZA(7)*DZ(3)+ZA(8)*DZ(4))/(ZA(9)*FI(1))-1
C     Set condition index vector (-1 = lsq, 0 = eq, 1 = pos)
      DO 20 I = 1, 17
         NFIX(I) = 0
   20 CONTINUE
C     Change conditions if necessary
      GOTO (70,60,70,50,70,40,90,30,100) ISTR
C     Structure change (ISTR=8, T2=T3, U2SING(T3)=-U2MAX)
   30 W(7) = ZI(9)
      GOTO 90
C     Structure change (ISTR=6, T7=T8, DLAM2(T8)=0)
   40 W(7) = ZI(9)
      GOTO 80
C     Structure change (ISTR=4, T1=0, LAM4=0)
   50 W(7) = ZA(8)
      W(9) = ZA(9)-4D-4
      GOTO 70
C     Structure change (ISTR=2, T4=T5)
   60 W(7) = ZA(12)-ZA(13)
C     Auxiliary condition for T8
   70 W(16) = ZA(16)-ZA(17)*99D-2
C     Auxiliary condition for T7
   80 W(15) = ZA(15)-ZA(16)
C     Auxiliary condition for T2
   90 W(10) = ZA(10)-ZA(11)
  100 RETURN
      END
```

```
C ******************************************************************
C     Switching functions, derivatives & interior point conditions
C ******************************************************************
      SUBROUTINE SROB ( T, Z, NSW, Q, DQ, L, LS, LD )
C
      COMMON   /COUNT/ ID1, ICS, ID2  /MPBVP/ ND1, ZI, ZIS
      COMMON   /STRUC/ FI, JI, IQ, MSW  /CNTRL/ D, ISTR
      COMMON   /INERT/ I323, I31, EL, MB, ML, MLB
      INTEGER  ID1, ICS, ID2(3), ND1(2), JI(33), IQ(9), MSW(9,9), ISTR
      INTEGER  NSW, L, LS, LD, I
      REAL*8   ZI(200), ZIS(200), FI(33), D(2), S, A1
      REAL*8   I323, I31, EL, MB, ML, MLB, T, Z(17), Q(NSW), DQ(NSW)
C     Numerical switching points on unit interval
      REAL*8   TI(8)/6D-2,30D-2,38D-2,46D-2,54D-2,62D-2,70D-2,94D-2/
C
C     Numerical time is T=T(L)  (cf MAIN)
C     If LD=-1, T(L) is a switching point TI(LS)
C     If LD=-2, T(L) is a grid point, Z=Z(T(L)+)
C     If LD=-3, T(L) is a grid point, Z=Z(T(L)-)
C
C     Count SROB calls
      ICS = ICS+1
C     Evaluate time switches  Q(I)  and derivatives  DQ(I)
      DO 10 I = 1, NSW
         Q(I) = T-TI(IQ(I))
        DQ(I) = 1
   10 CONTINUE
      IF (LD.NE.-1) GOTO 20
C     Evaluate q1 = lambda1 (at interior point T5) or
C     evaluate q2 = lambda2 (at other interior points)
      IF (LS.EQ.5) ZI(LS) = Z(5)
      IF (LS.NE.5) ZI(LS) = Z(6)
C     Evaluate interior point conditions for special cases
      S  = (MLB*Z(3)+ML*EL)*2
      A1 = I31+I323+ML*EL*EL+(S/2+ML*EL)*Z(3)
      IF (LS.EQ.2 .AND. ISTR.EQ.8)
     @   ZI(9) = Z(2)*(Z(7)/Z(5)*A1+(S-2*MLB*A1/S)*Z(1))+D(2)
      IF (LS.EQ.8 .AND. ISTR.EQ.6)
     @   ZI(9) = -Z(5)*Z(2)*S/MLB+Z(6)*Z(1)*S/A1-Z(8)
C     Monitoring the use of switching functions
   20 CALL MONIT ( NSW, Q, DQ, L, LS, LD )
      RETURN
      END
```

Alle diskreten Lösungen wurden mit diesen Prozeduren berechnet, wobei eine Abbruchgenauigkeit von mindestens 10^{-8}, meist sogar 10^{-9} erreicht wurde. Als Integrator kam dabei DIFSYS in einer auf Hussels [16] zurückgehenden Variante zum Einsatz.

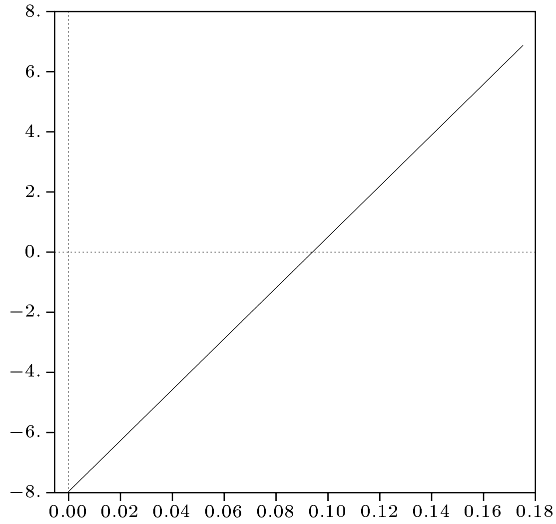
Die nun folgenden graphischen Darstellungen der Lösungen sind in Abschnitt (3.4) beschrieben.

$r_e = 74.9\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

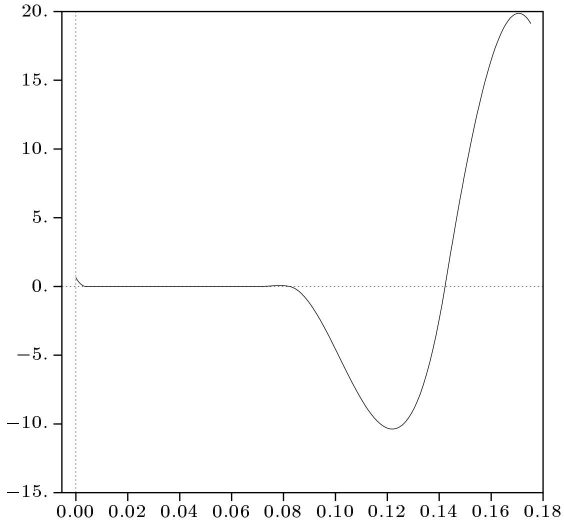


$r_e = 74.9\,\text{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

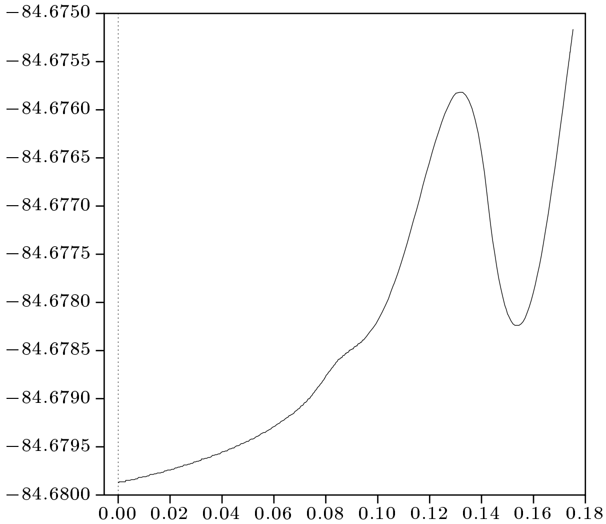
$\lambda_1(t)$ [s²/m]



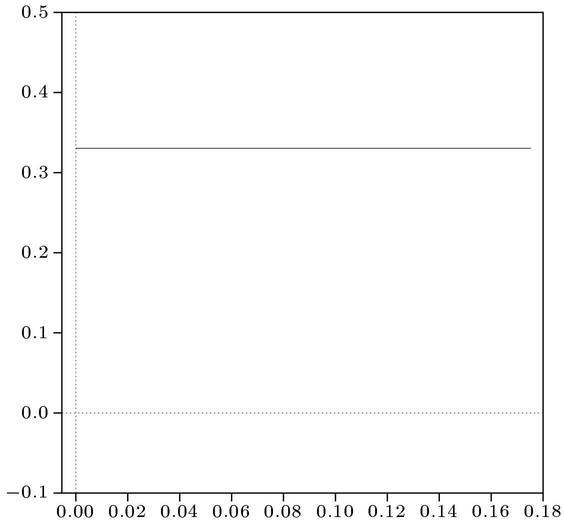
$\lambda_2(t)$ [10^{-3}s²]



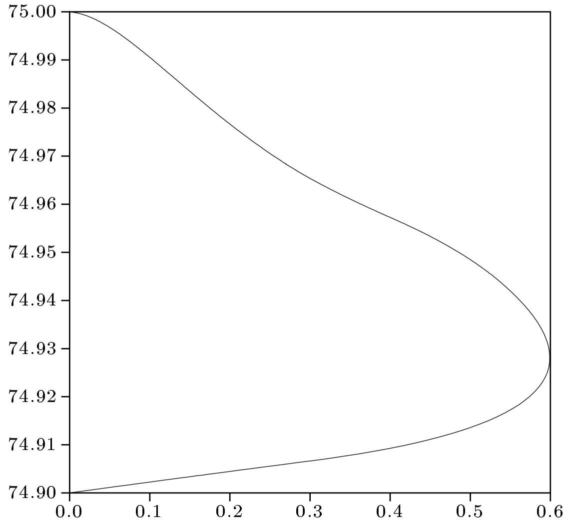
$\lambda_3(t)$ [s/m]



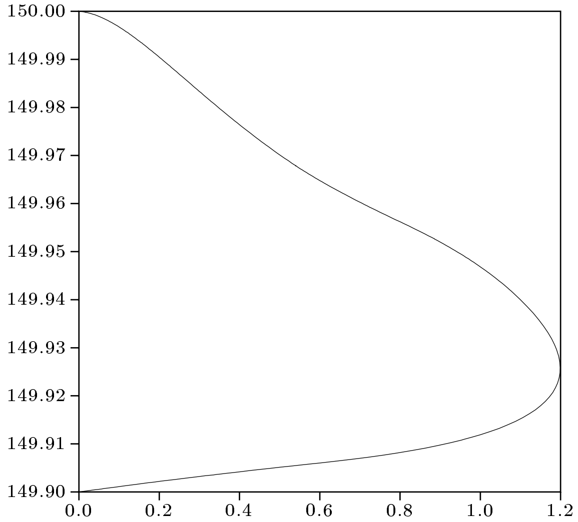
$\lambda_4(t)$ [s]



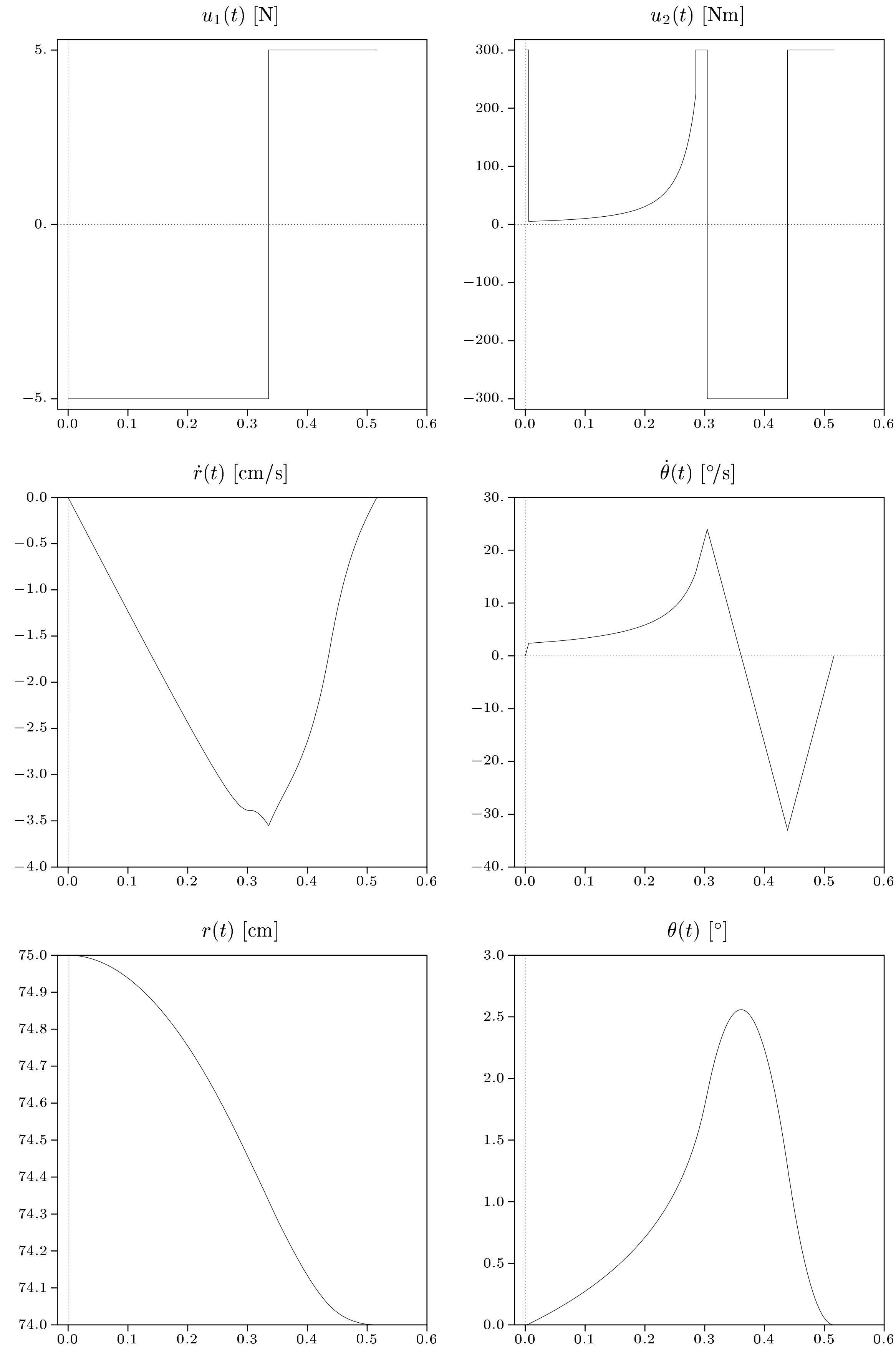
$x(y)$ [cm]



$\bar{x}(\bar{y})$ [cm]

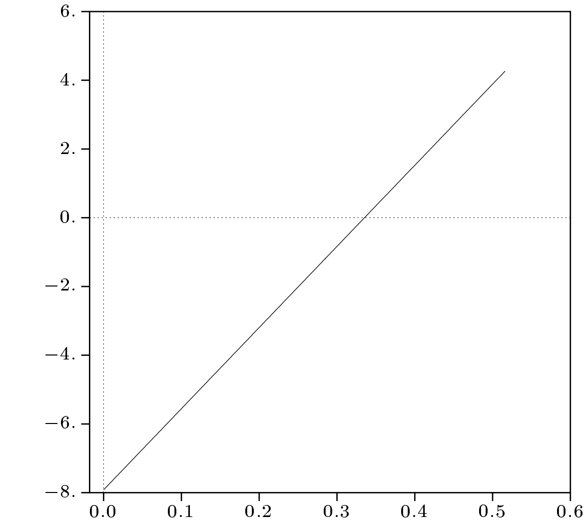


$r_e = 74\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen

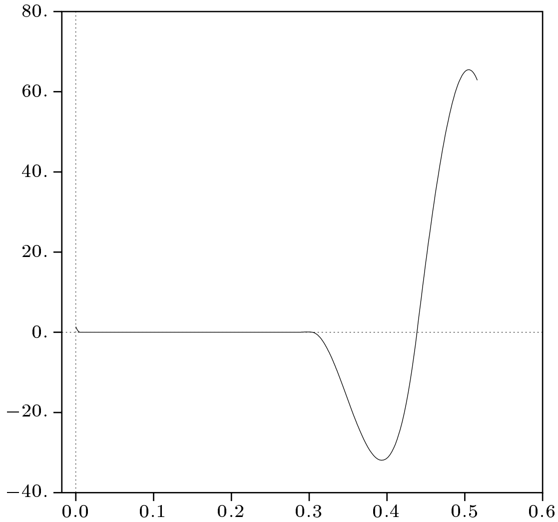


$r_e = 74$ cm: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

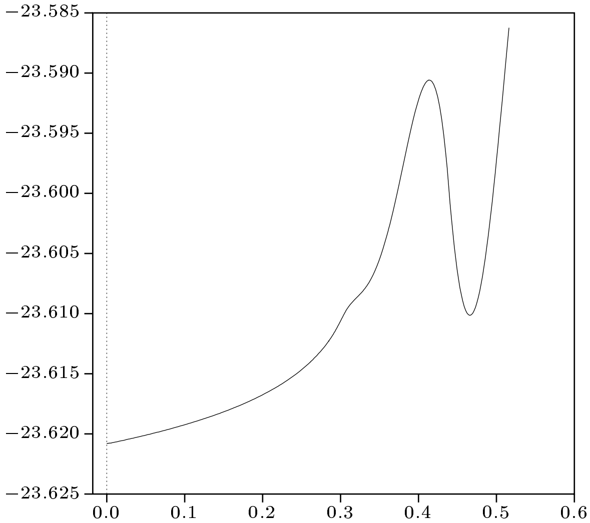
$\lambda_1(t)$ [s²/m]



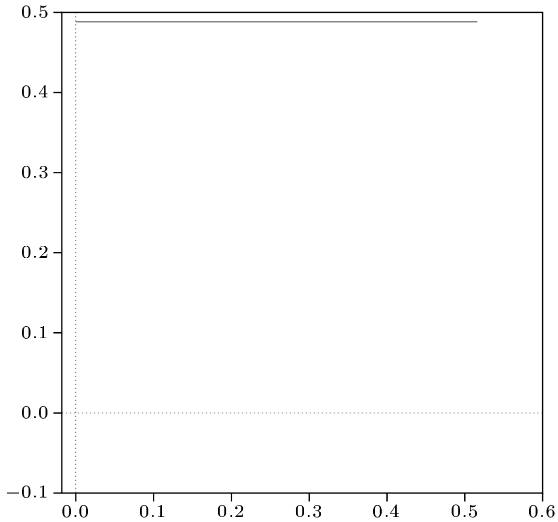
$\lambda_2(t)$ $[10^{-3}\mathrm{s}^2]$



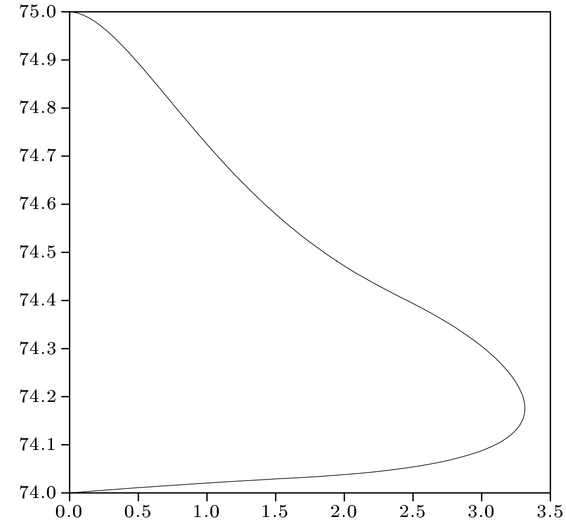
$\lambda_3(t)$ [s/m]



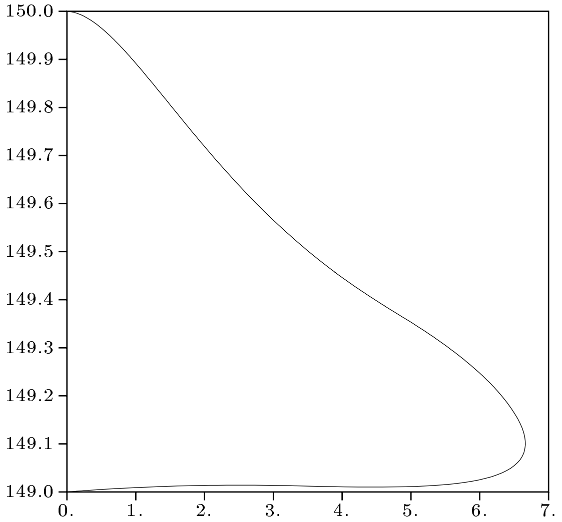
$\lambda_4(t)$ [s]



$x(y)$ [cm]

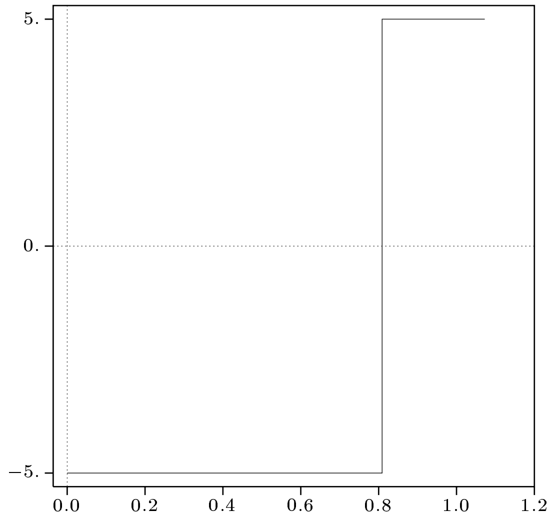


$\bar{x}(\bar{y})$ [cm]

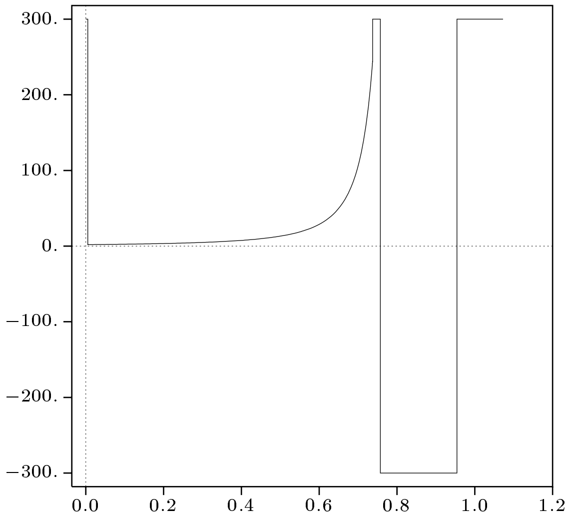


$r_e = 70\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen

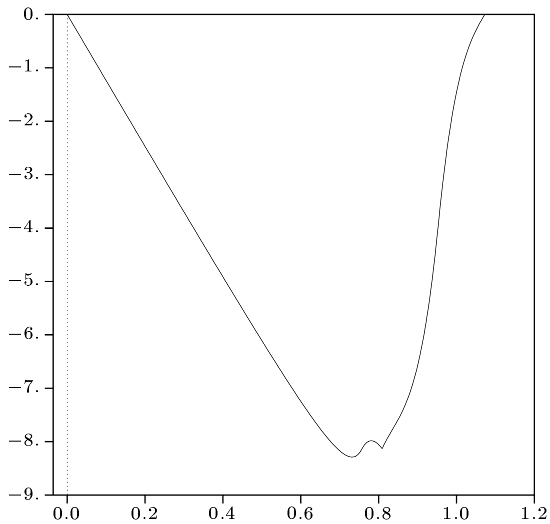
$u_1(t)$ [N]



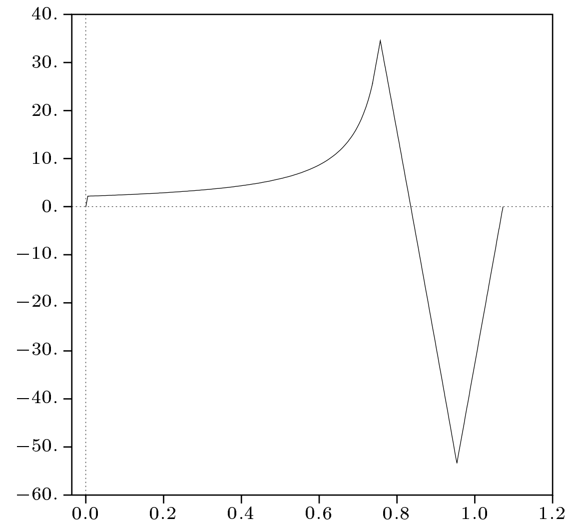
$u_2(t)$ [Nm]



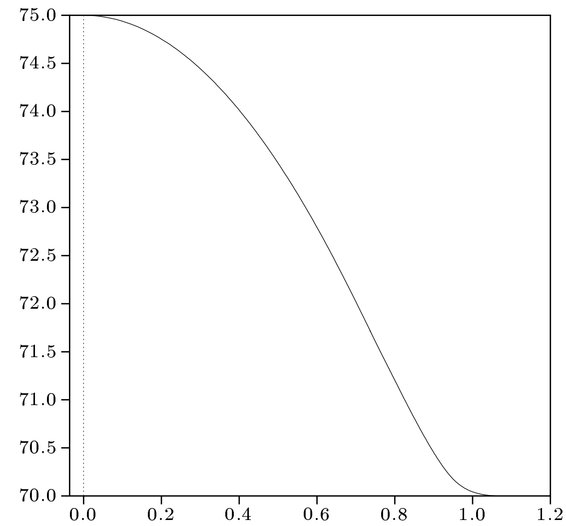
$\dot{r}(t)$ [cm/s]



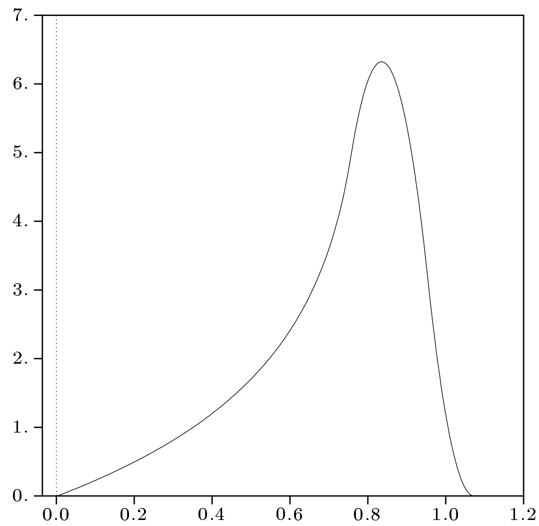
$\dot{\theta}(t)$ [°/s]



$r(t)$ [cm]

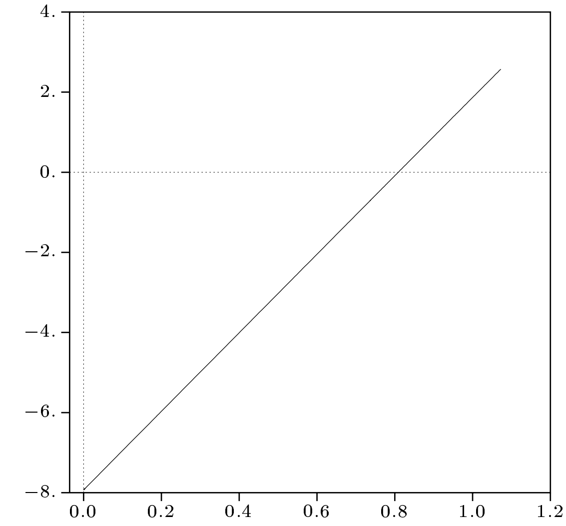


$\theta(t)$ [°]

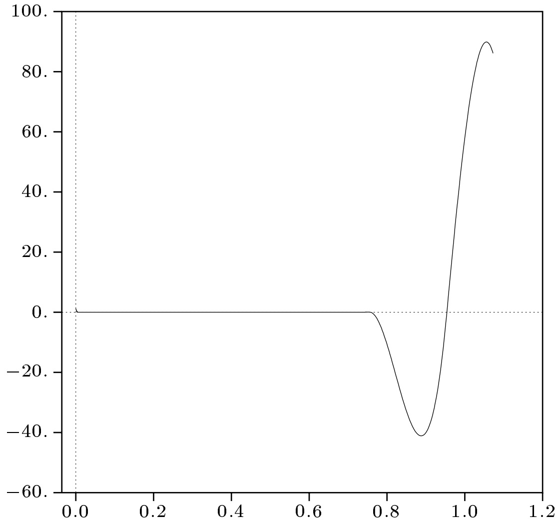


$r_e = 70\,\text{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

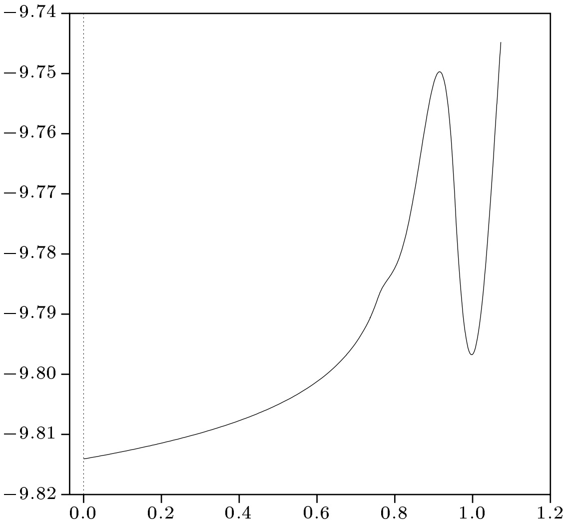
$\lambda_1(t)$ [s²/m]



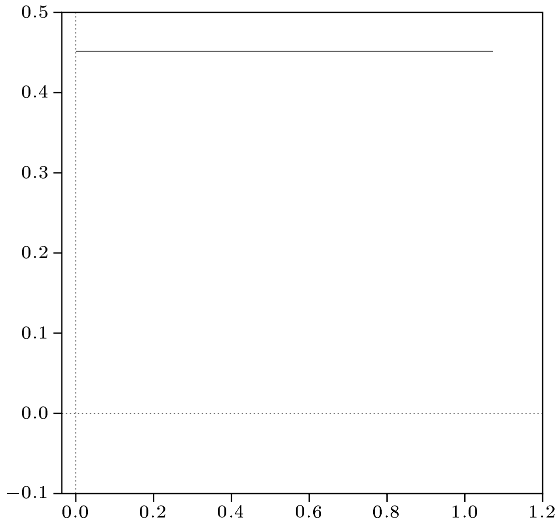
$\lambda_2(t)$ [10^{-3}s²]



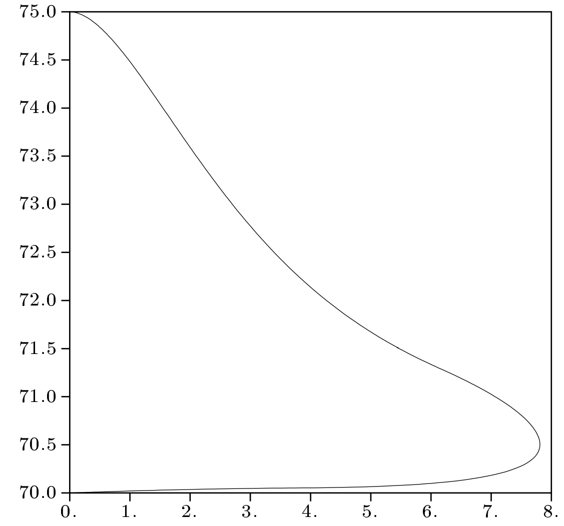
$\lambda_3(t)$ [s/m]



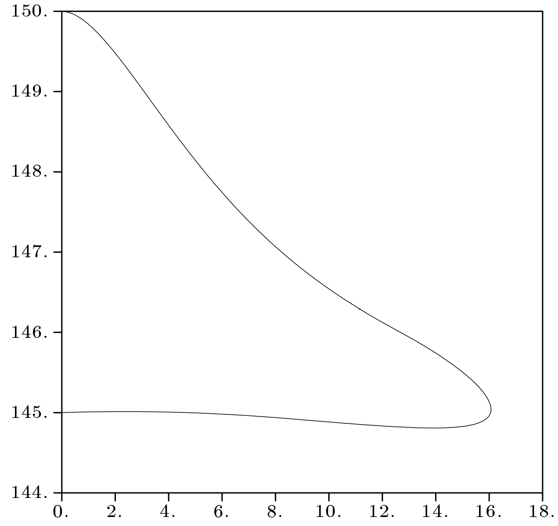
$\lambda_4(t)$ [s]



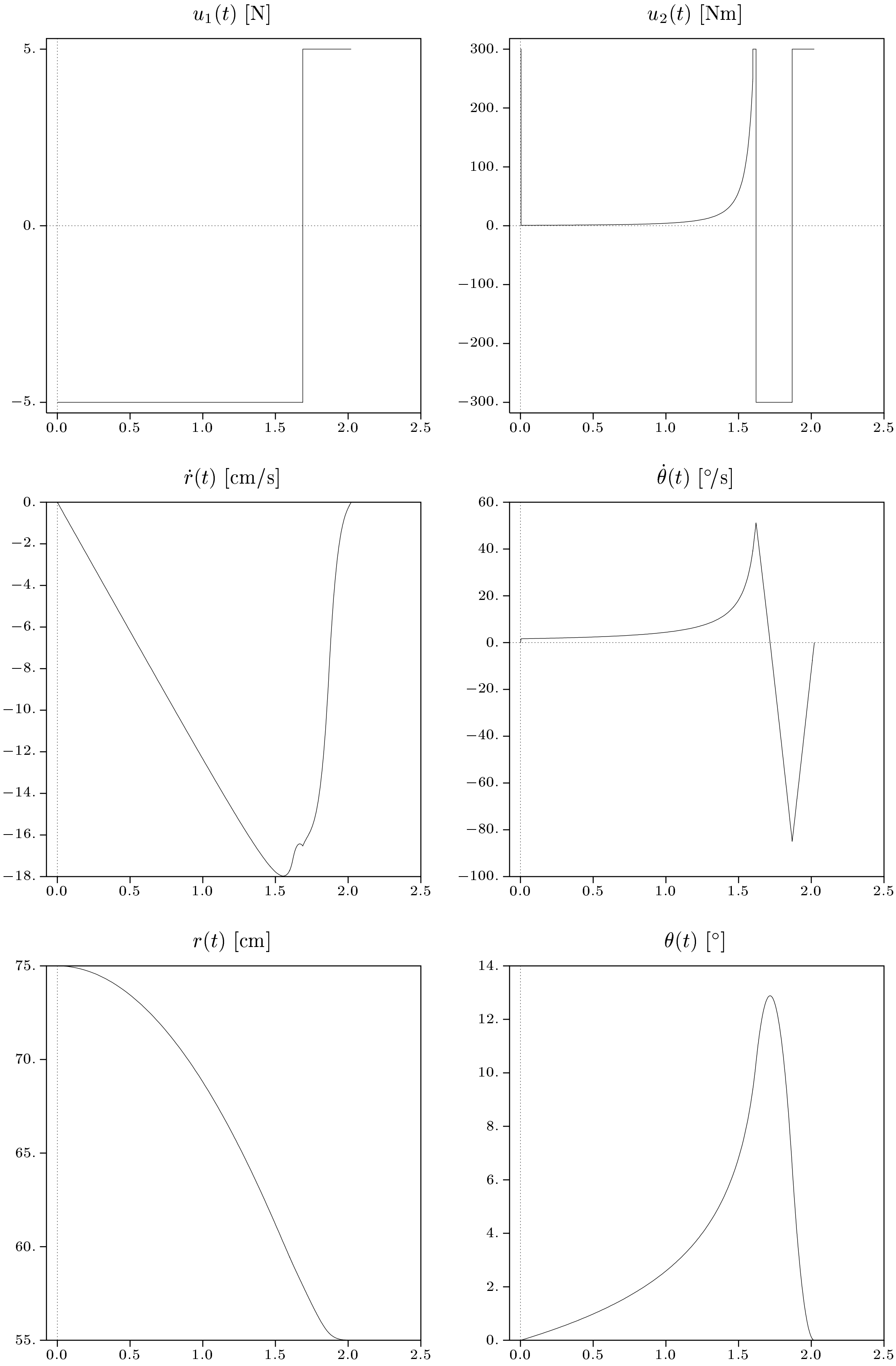
$x(y)$ [cm]



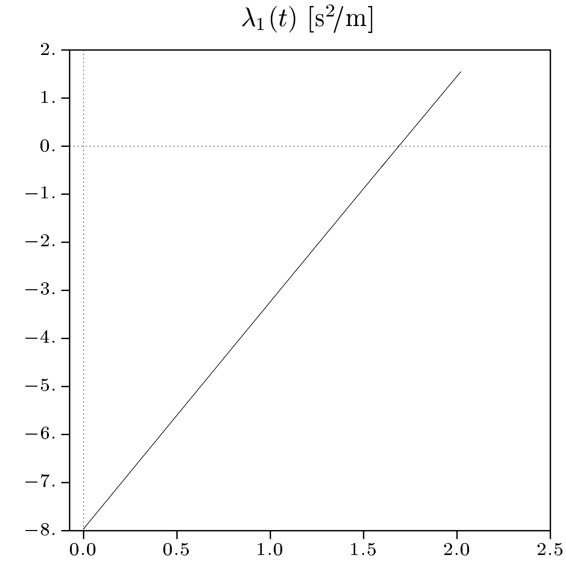
$\bar{x}(\bar{y})$ [cm]

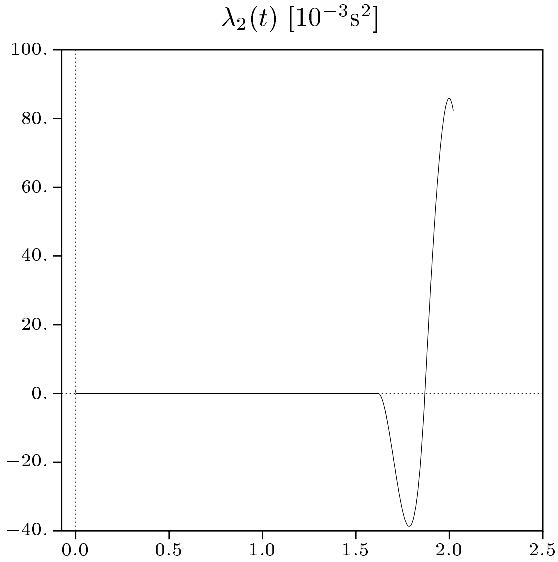


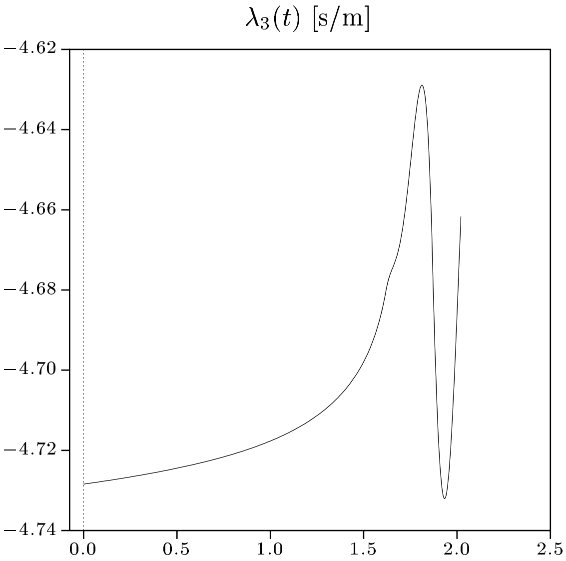
$r_e = 55\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

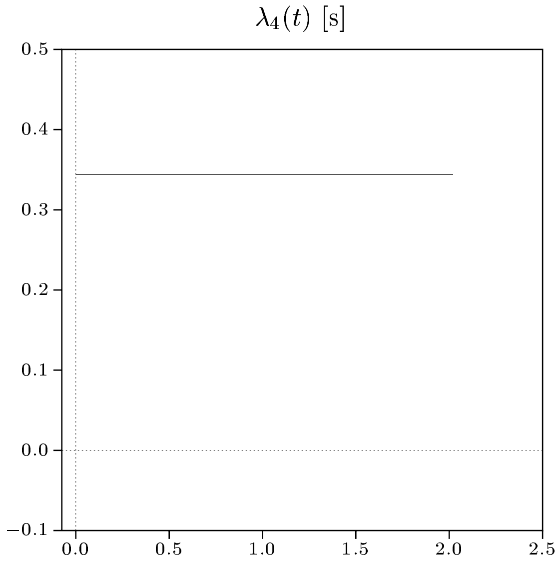


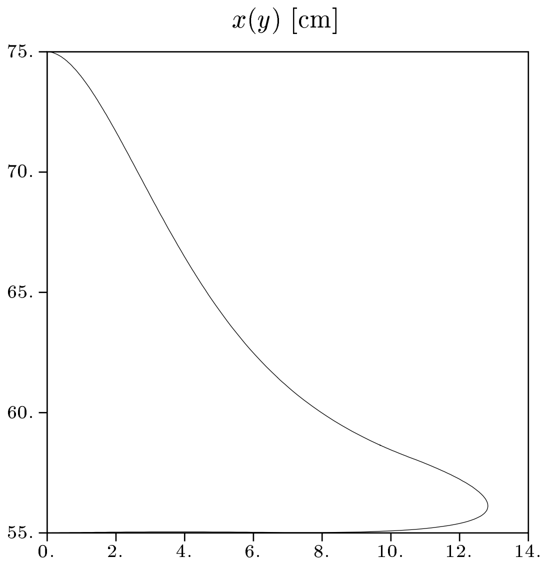
$r_e = 55\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

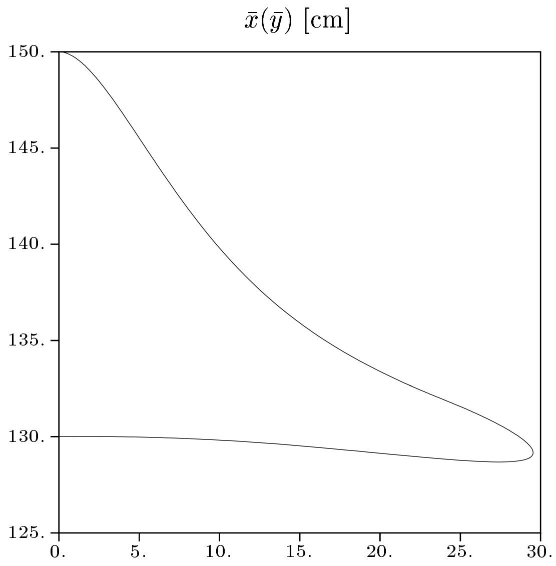






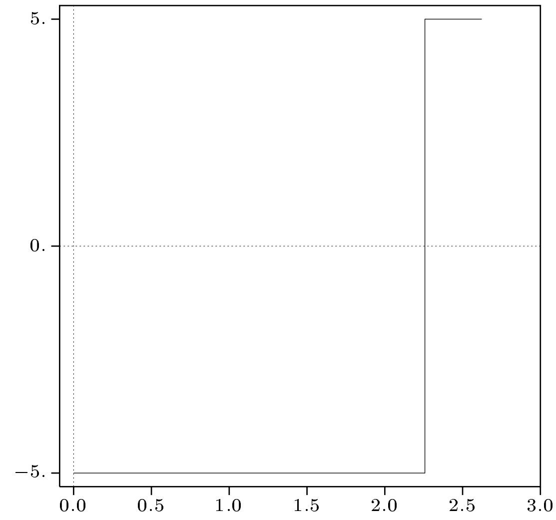




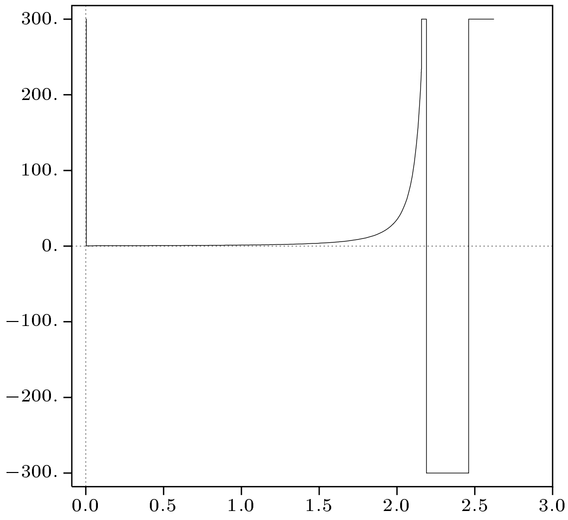


$r_e = 40\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen

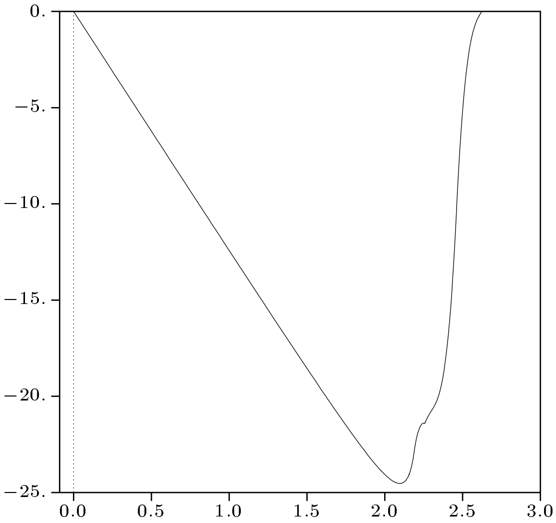
$u_1(t)$ [N]



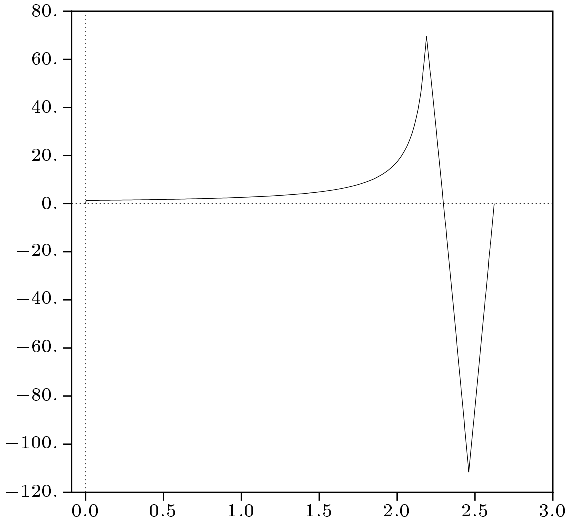
$u_2(t)$ [Nm]



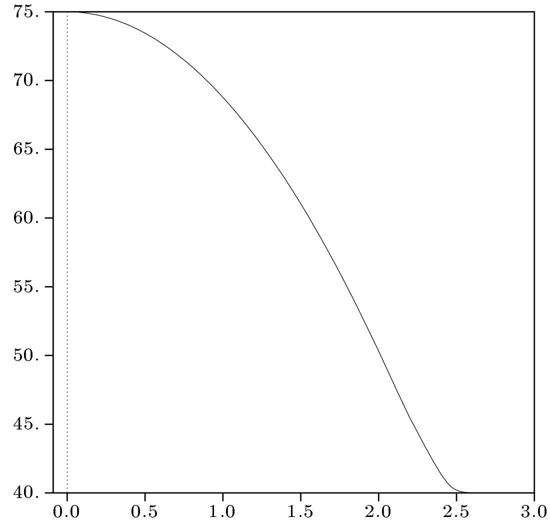
$\dot{r}(t)$ [cm/s]



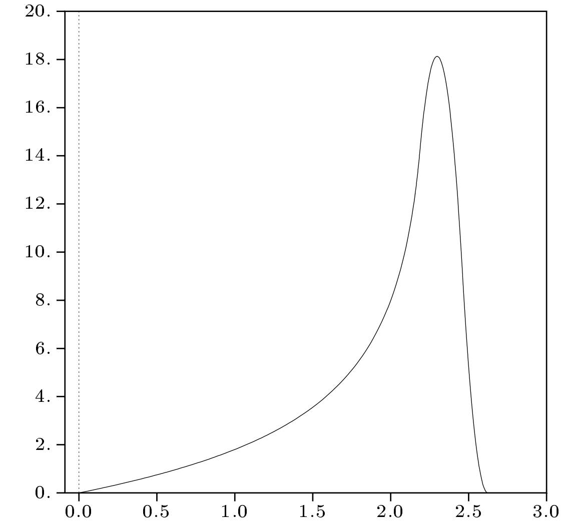
$\dot{\theta}(t)$ [°/s]



$r(t)$ [cm]

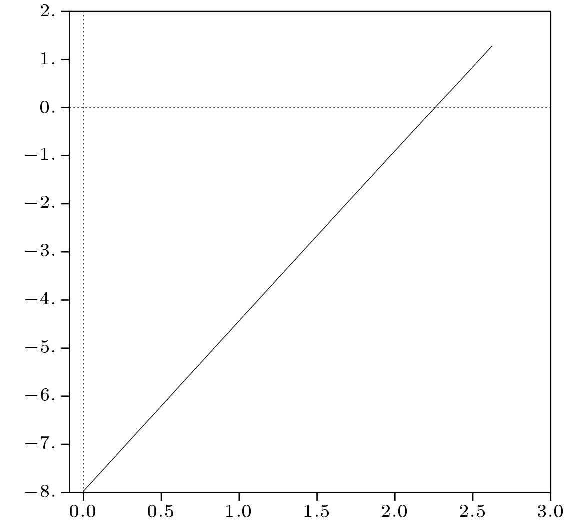


$\theta(t)$ [°]

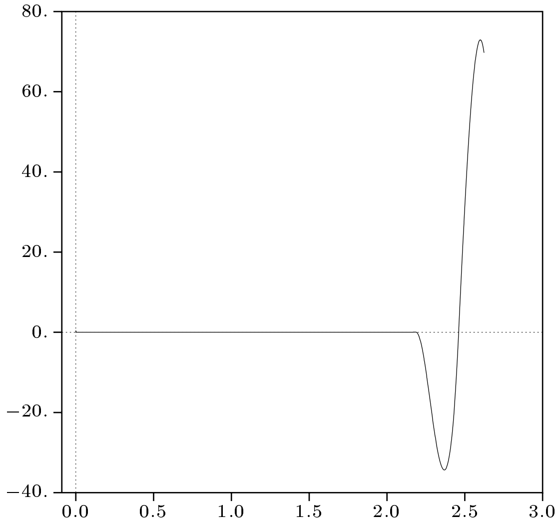


$r_e = 40\,$cm: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

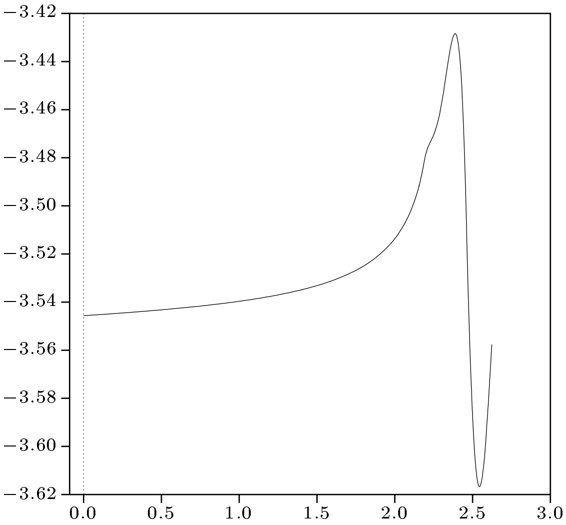
$\lambda_1(t)$ [s²/m]



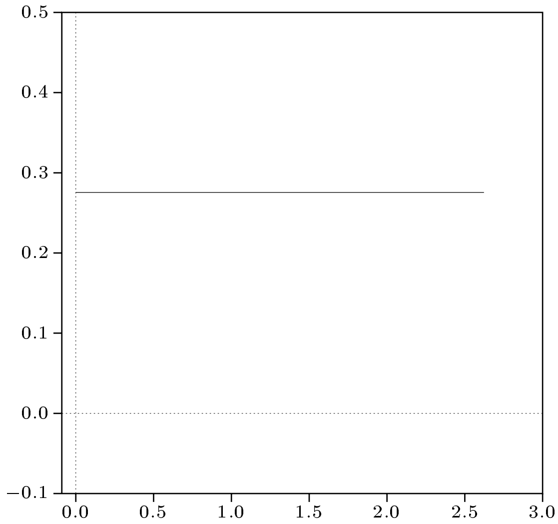
$\lambda_2(t)$ [10^{-3}s²]



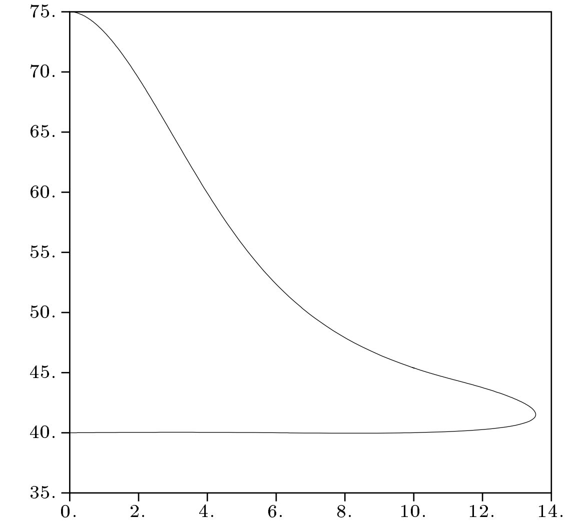
$\lambda_3(t)$ [s/m]



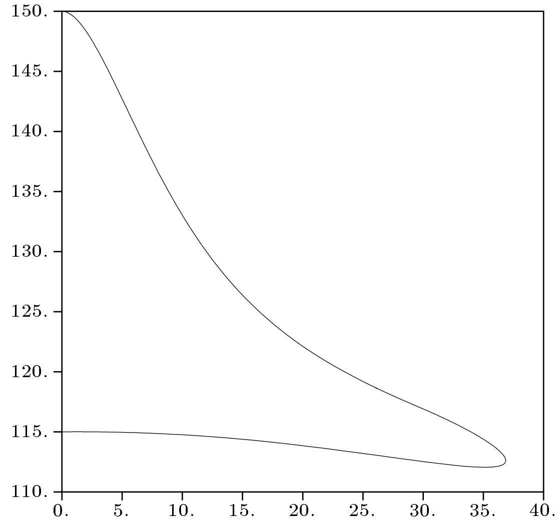
$\lambda_4(t)$ [s]



$x(y)$ [cm]

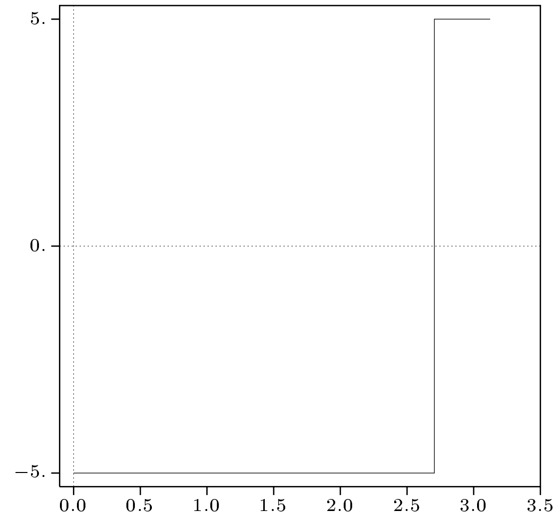


$\bar{x}(\bar{y})$ [cm]

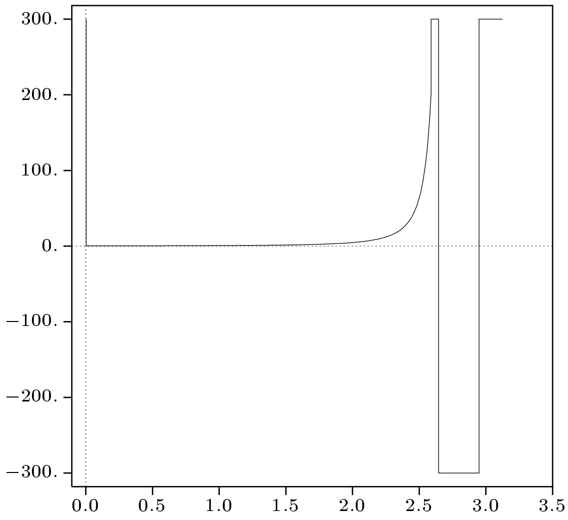


$r_e = 25\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

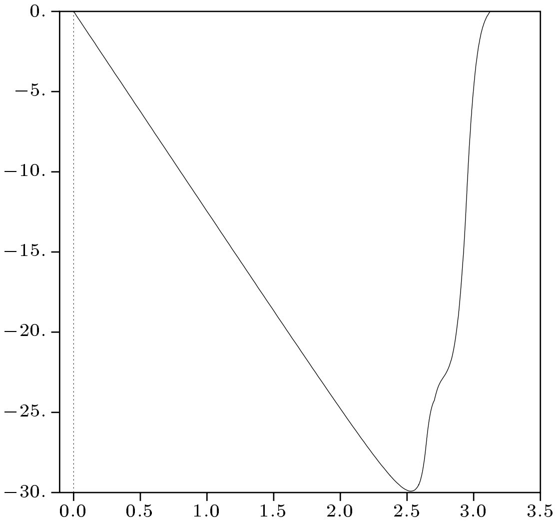
$u_1(t)$ [N]



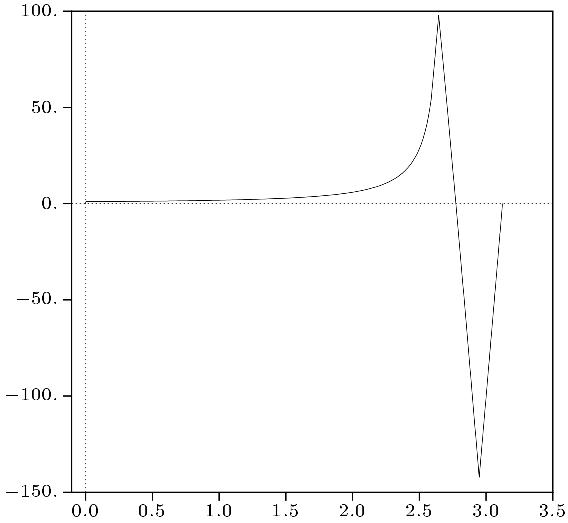
$u_2(t)$ [Nm]



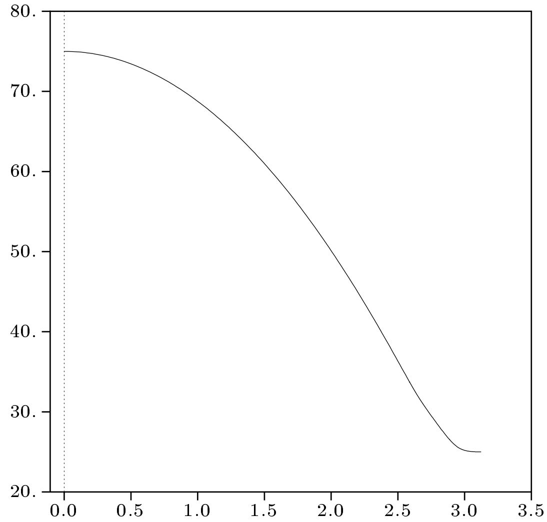
$\dot{r}(t)$ [cm/s]



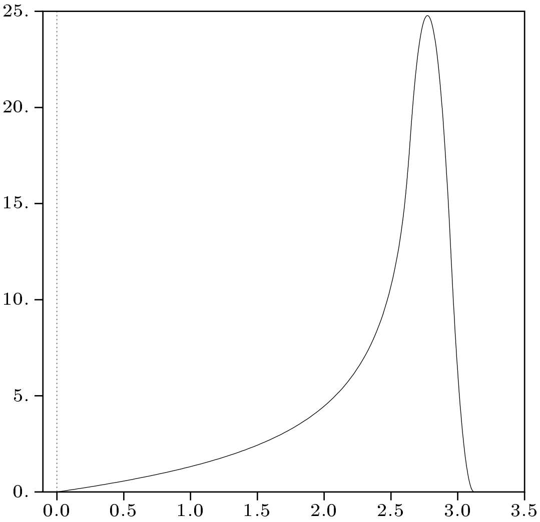
$\dot{\theta}(t)$ [°/s]



$r(t)$ [cm]

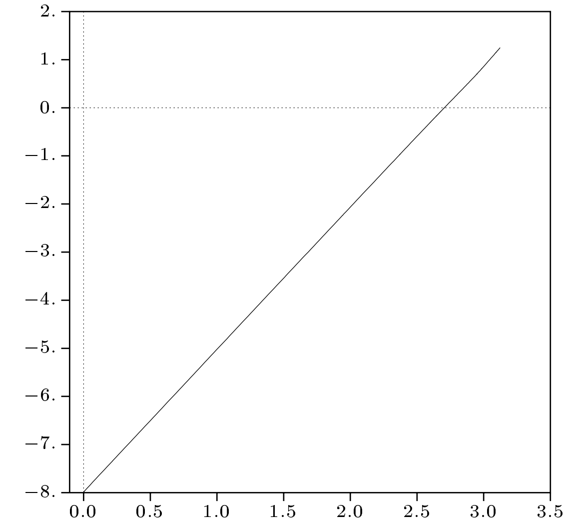


$\theta(t)$ [°]

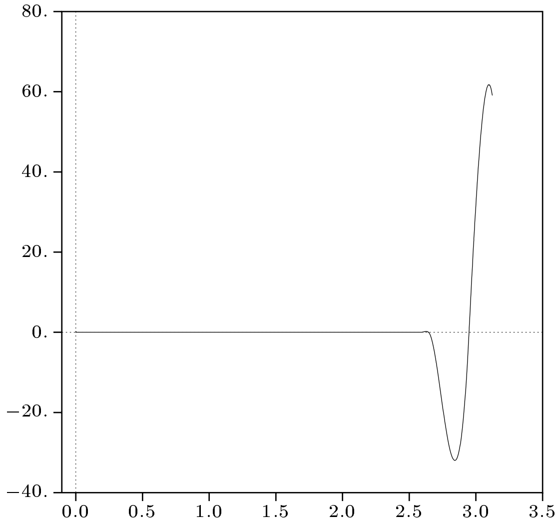


$r_e = 25\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

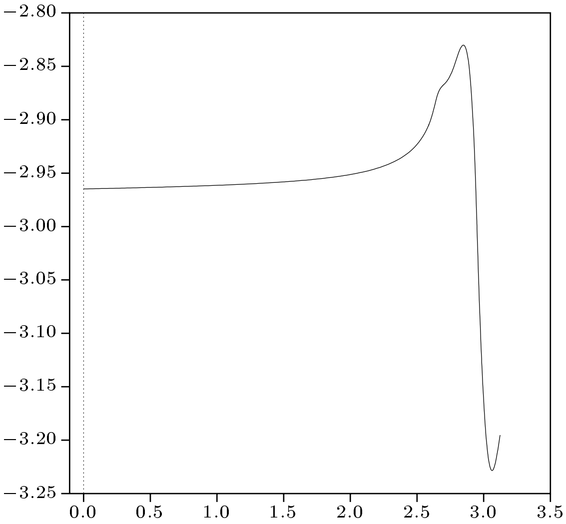
$\lambda_1(t)$ [$\mathrm{s^2/m}$]



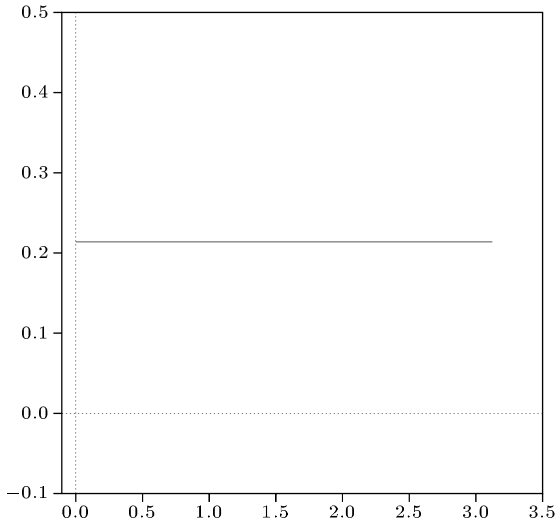
$\lambda_2(t)$ [$10^{-3}\mathrm{s^2}$]



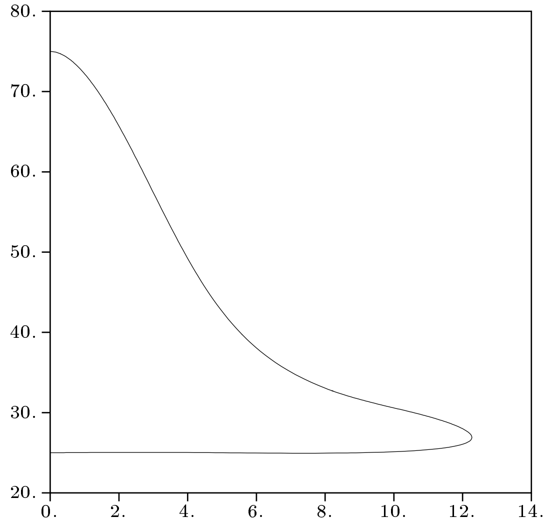
$\lambda_3(t)$ [s/m]



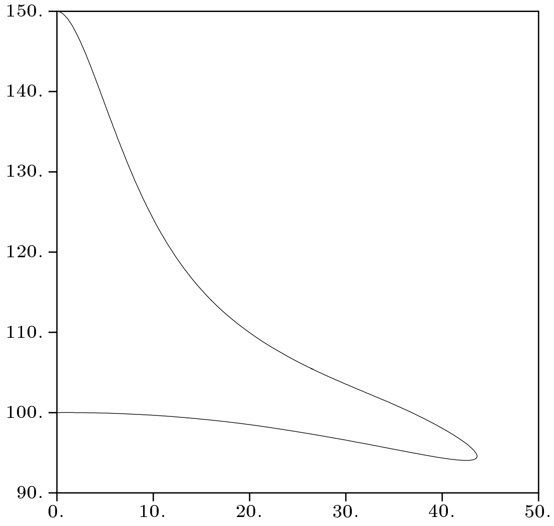
$\lambda_4(t)$ [s]



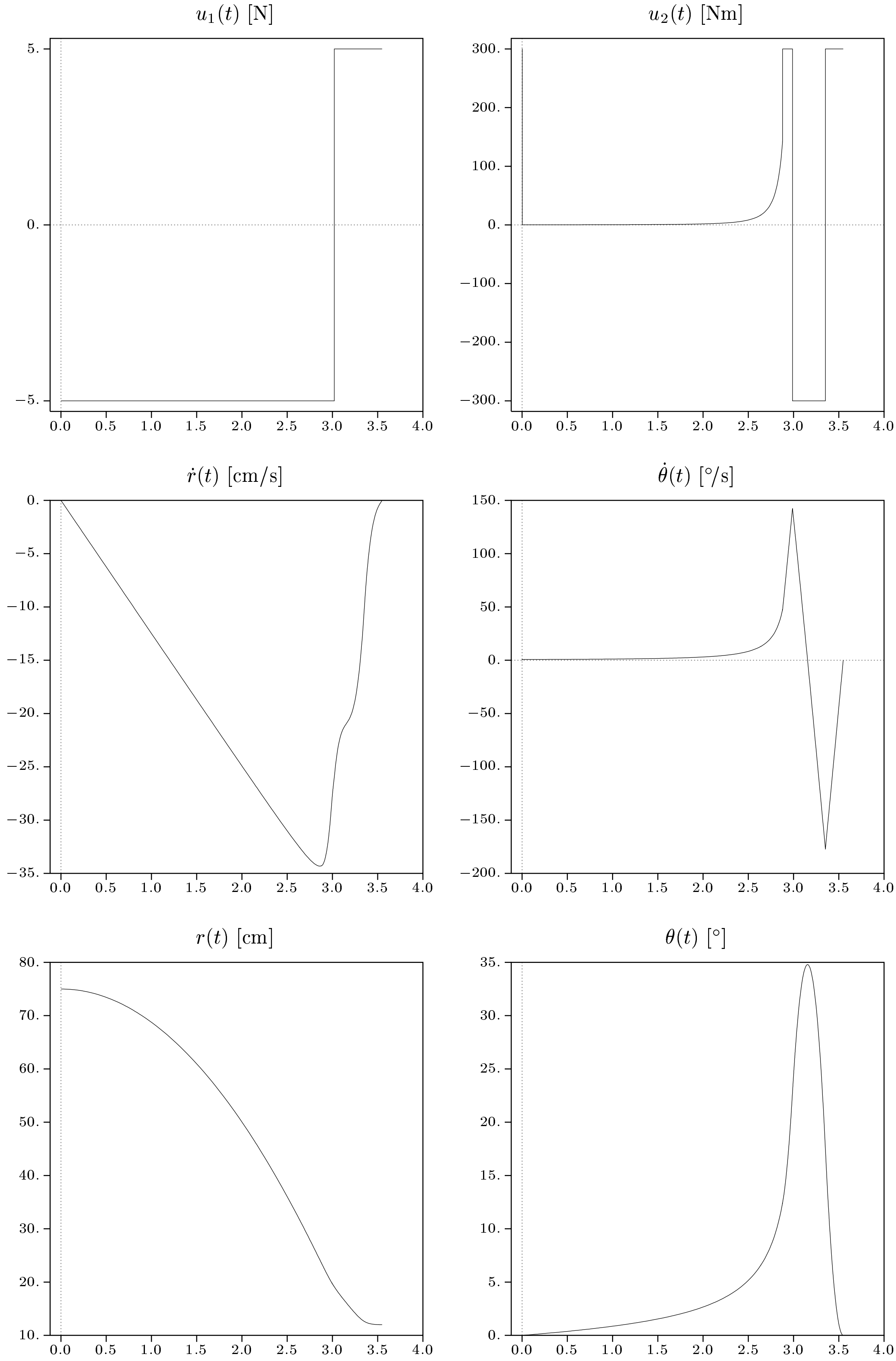
$x(y)$ [cm]



$\bar{x}(\bar{y})$ [cm]

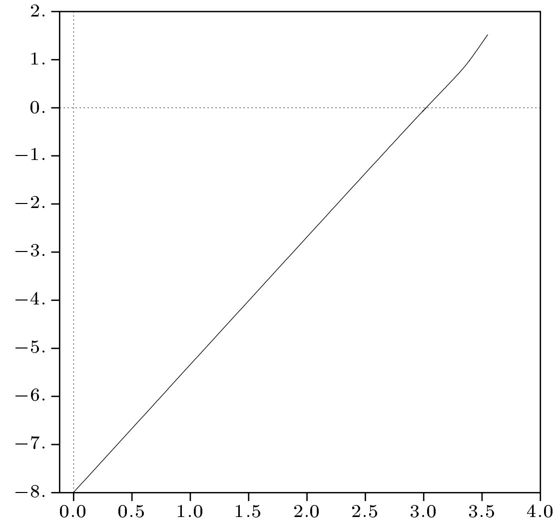


$r_e = 12\,$cm: optimale Steuerungen und Zustandsvariablen

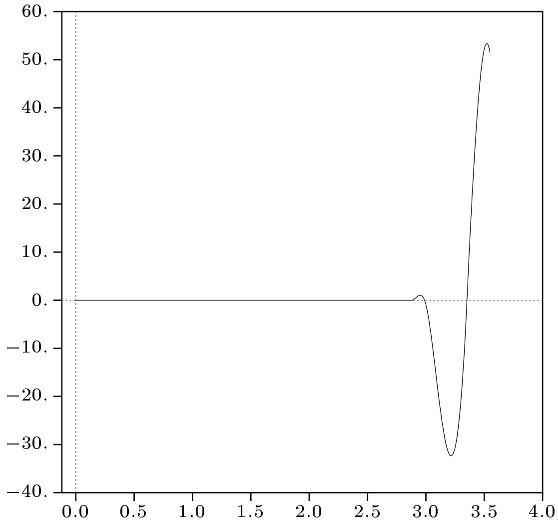


$r_e = 12\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

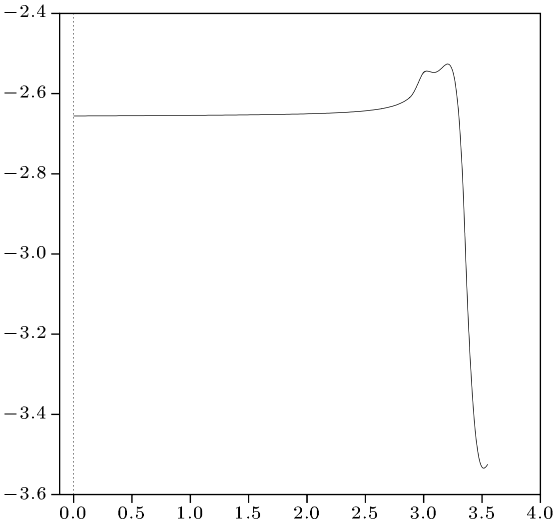
$\lambda_1(t)$ [s²/m]



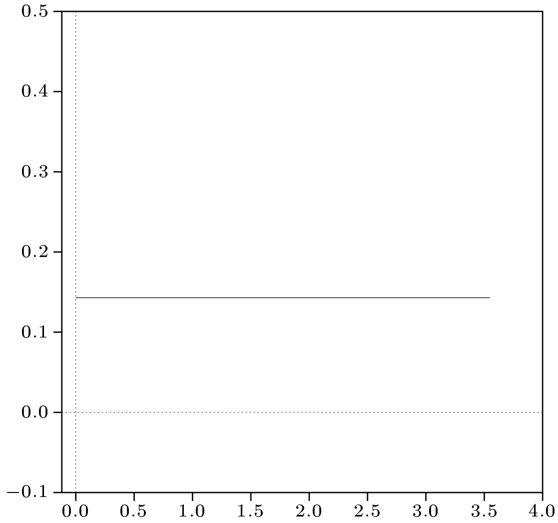
$\lambda_2(t)$ $[10^{-3}\mathrm{s}^2]$



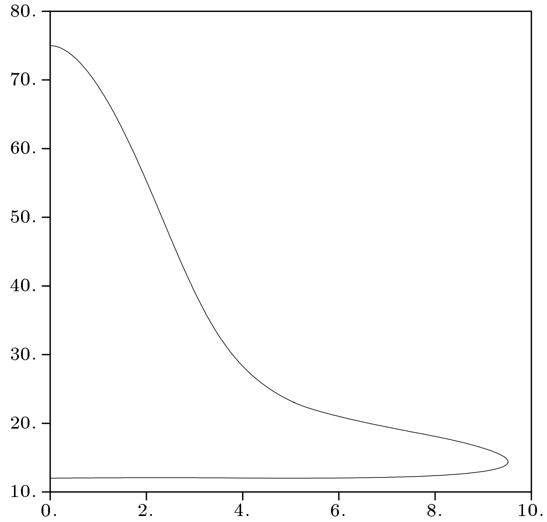
$\lambda_3(t)$ [s/m]



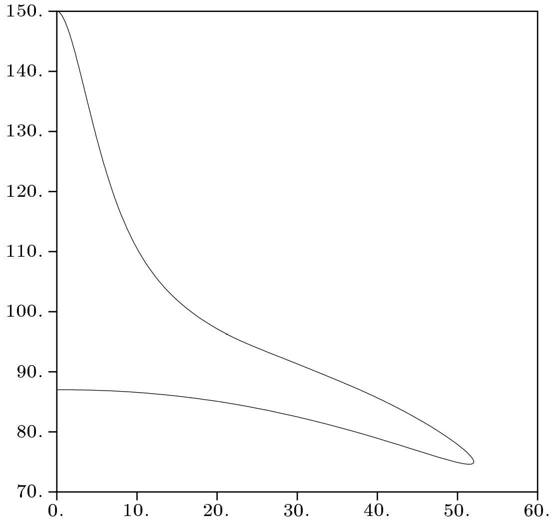
$\lambda_4(t)$ [s]



$x(y)$ [cm]

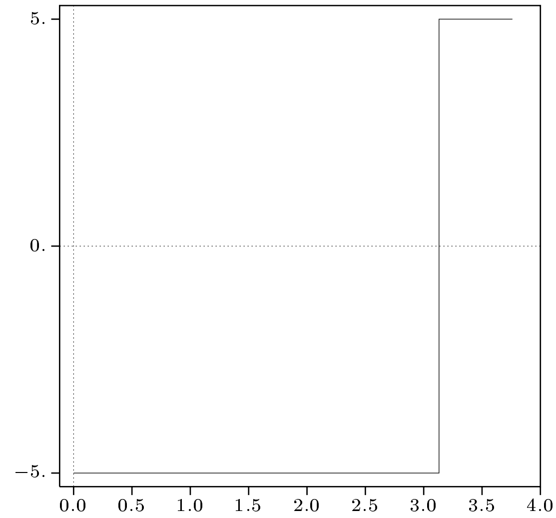


$\bar{x}(\bar{y})$ [cm]

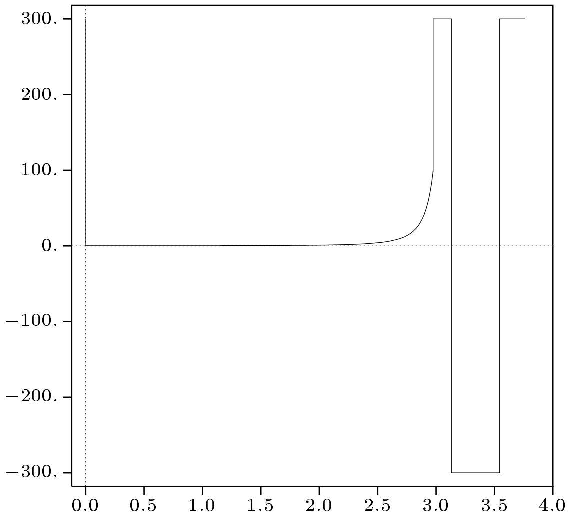


$r_e = 6.5025\,$cm: optimale Steuerungen und Zustandsvariablen

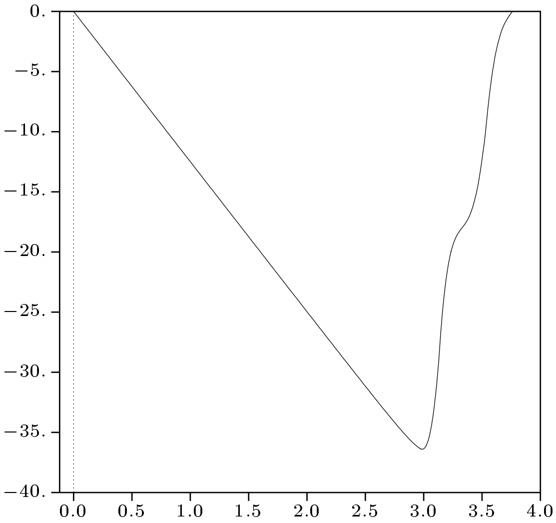
$u_1(t)$ [N]



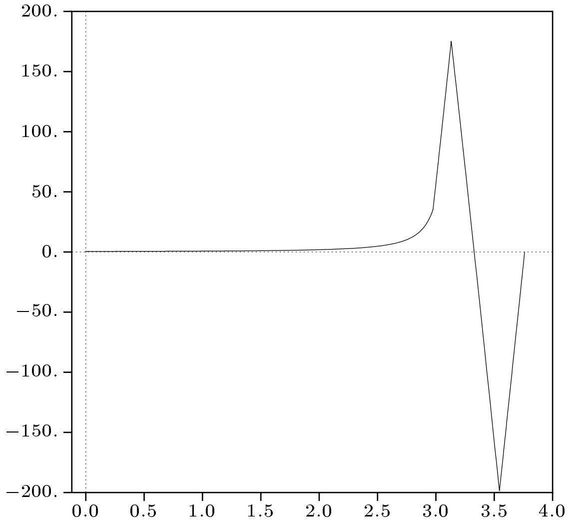
$u_2(t)$ [Nm]



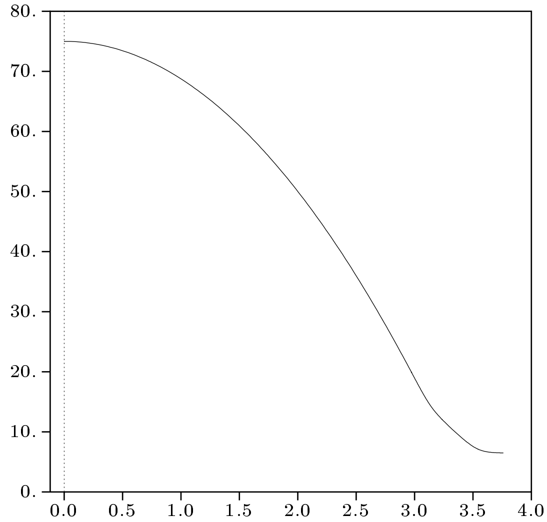
$\dot{r}(t)$ [cm/s]



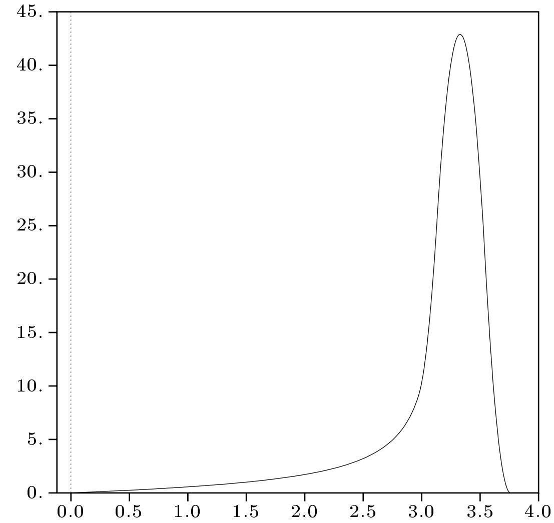
$\dot{\theta}(t)$ [°/s]



$r(t)$ [cm]

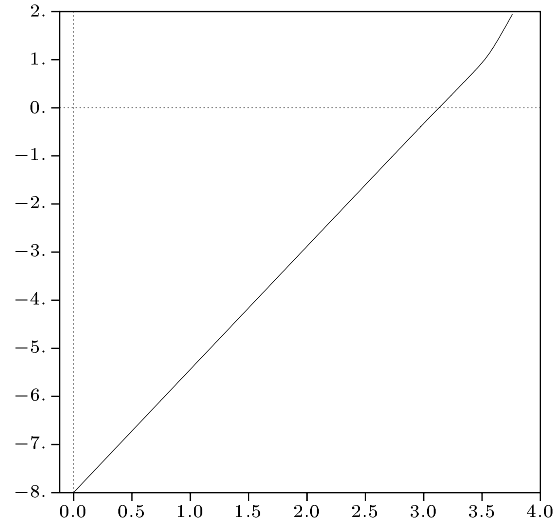


$\theta(t)$ [°]

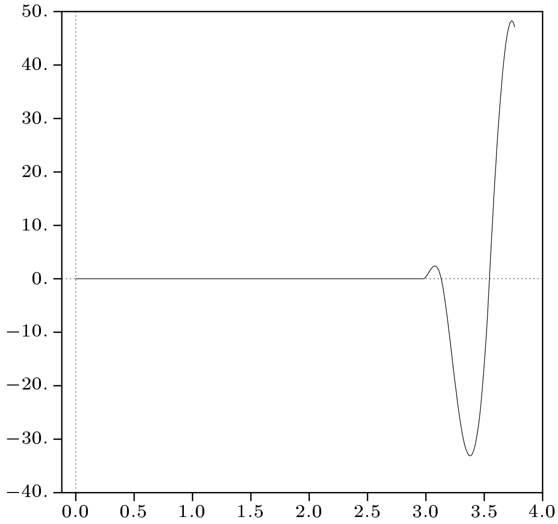


$r_e = 6.5025\,\text{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

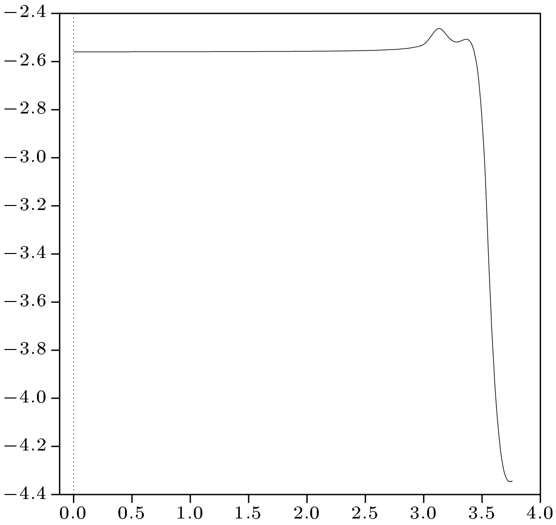
$\lambda_1(t)$ [s²/m]



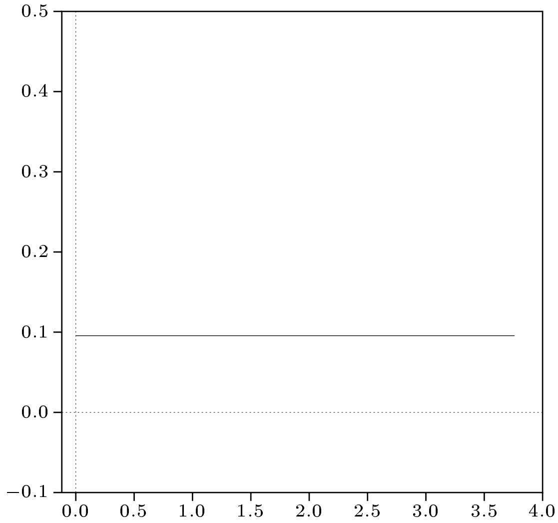
$\lambda_2(t)$ [10^{-3}s²]



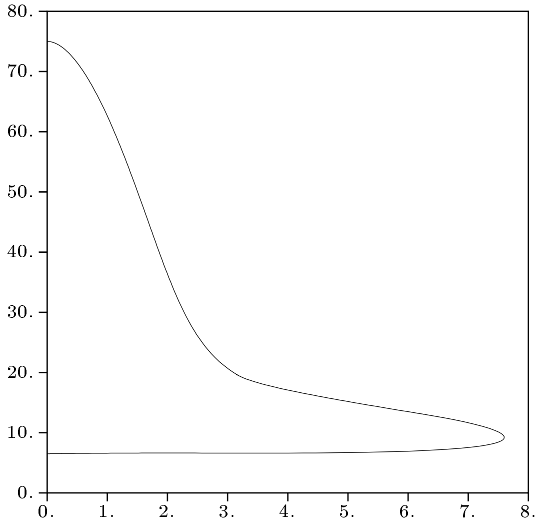
$\lambda_3(t)$ [s/m]



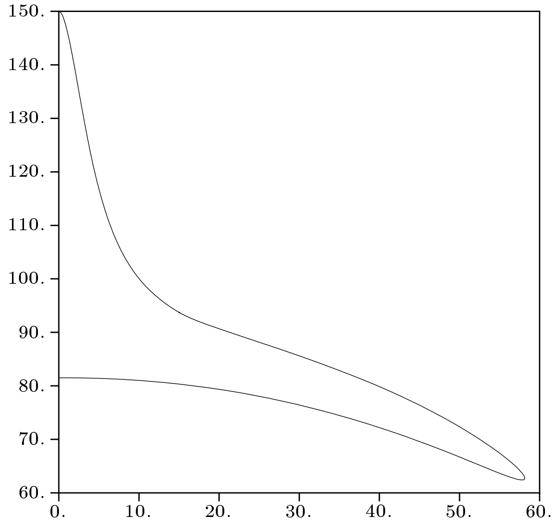
$\lambda_4(t)$ [s]



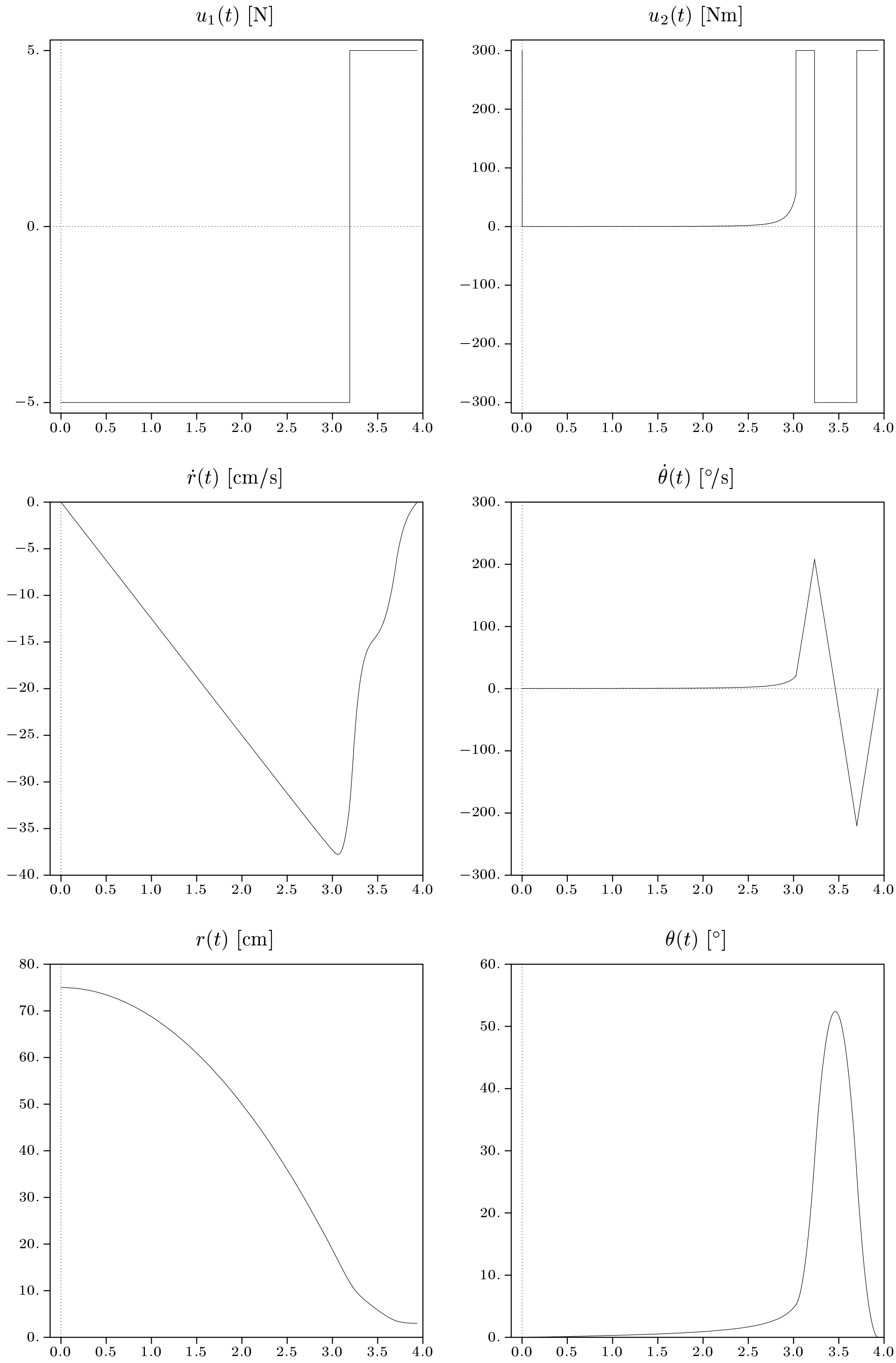
$x(y)$ [cm]



$\bar{x}(\bar{y})$ [cm]

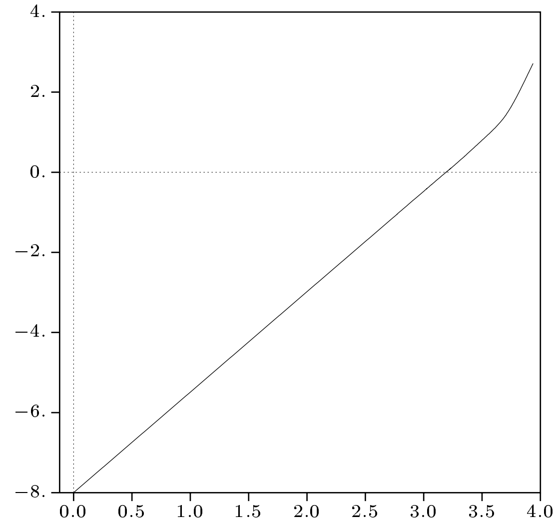


$r_e = 3\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

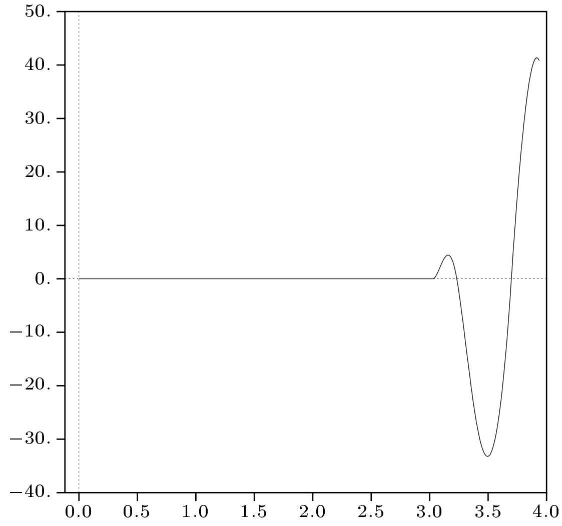


$r_e = 3\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

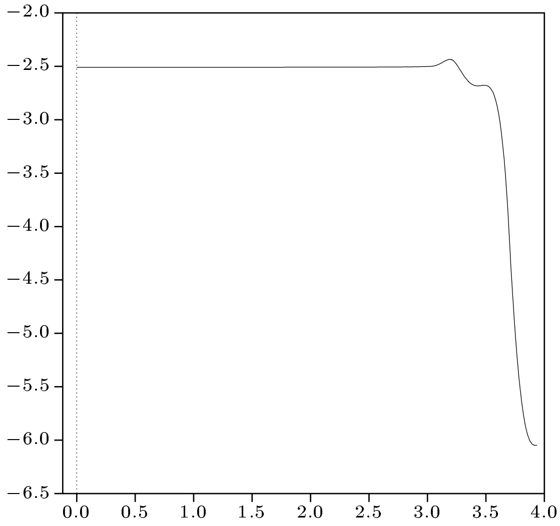
$\lambda_1(t)$ [s^2/m]



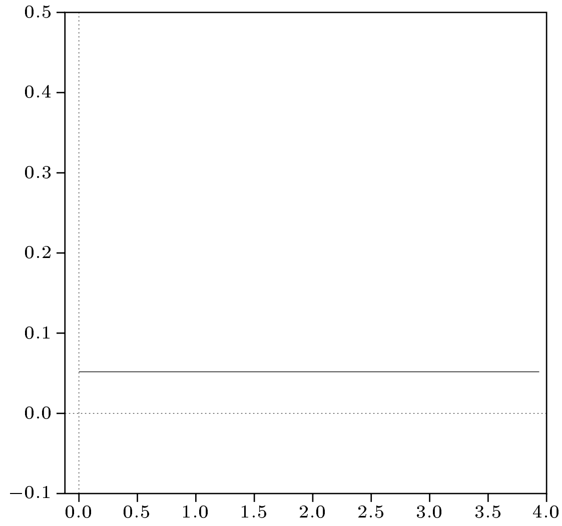
$\lambda_2(t)$ [$10^{-3}\mathrm{s}^2$]



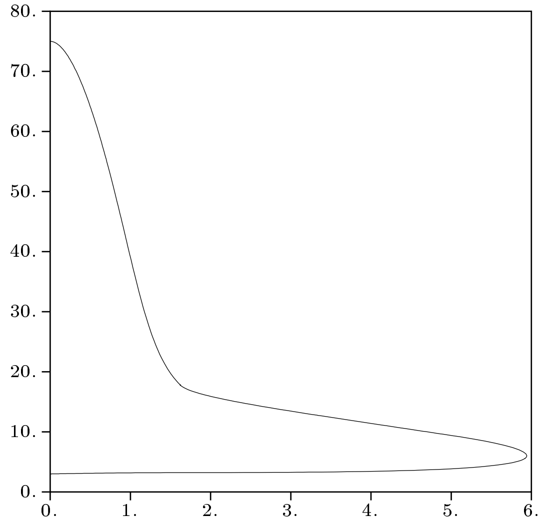
$\lambda_3(t)$ [s/m]



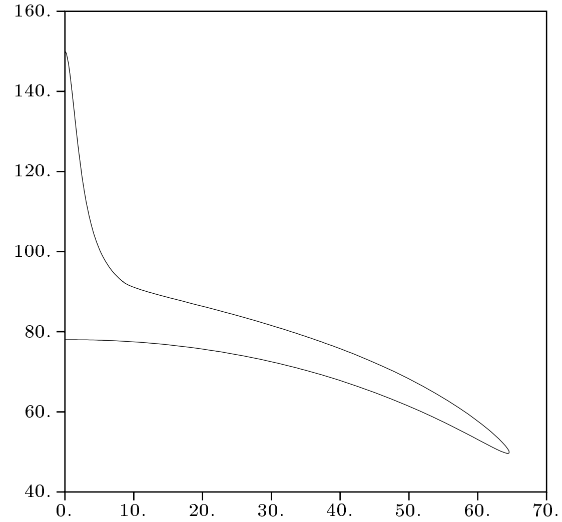
$\lambda_4(t)$ [s]



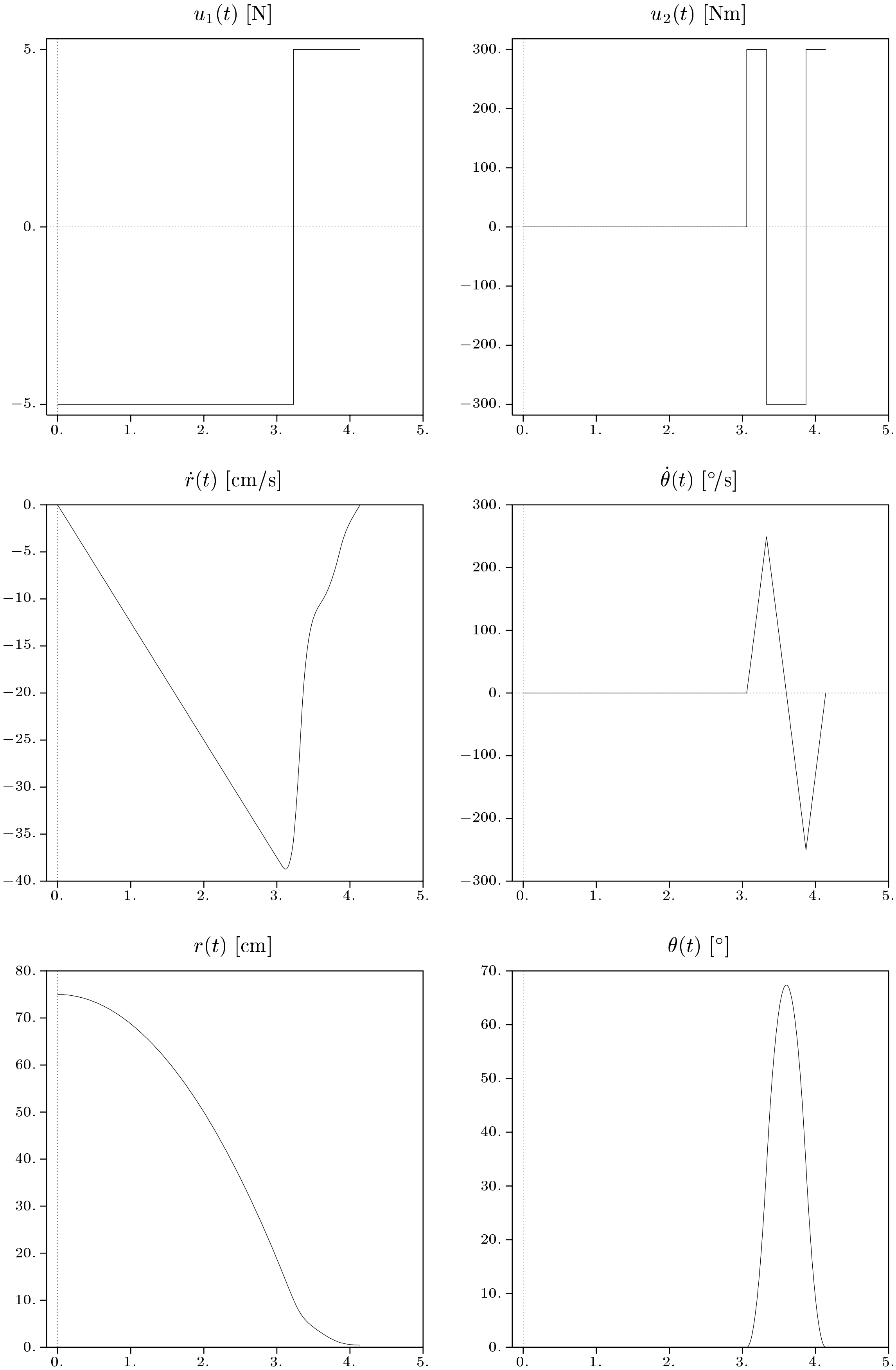
$x(y)$ [cm]



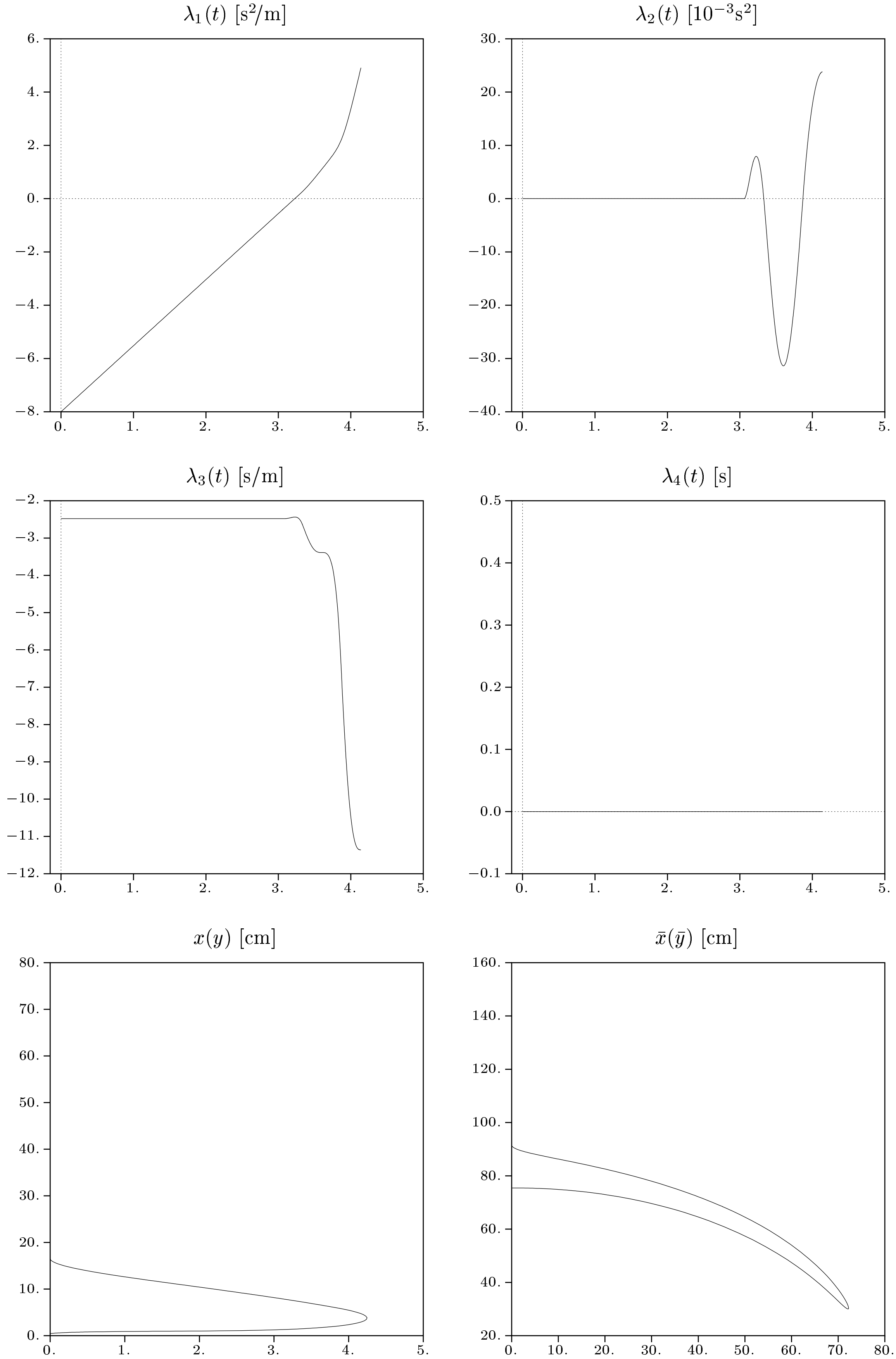
$\bar{x}(\bar{y})$ [cm]



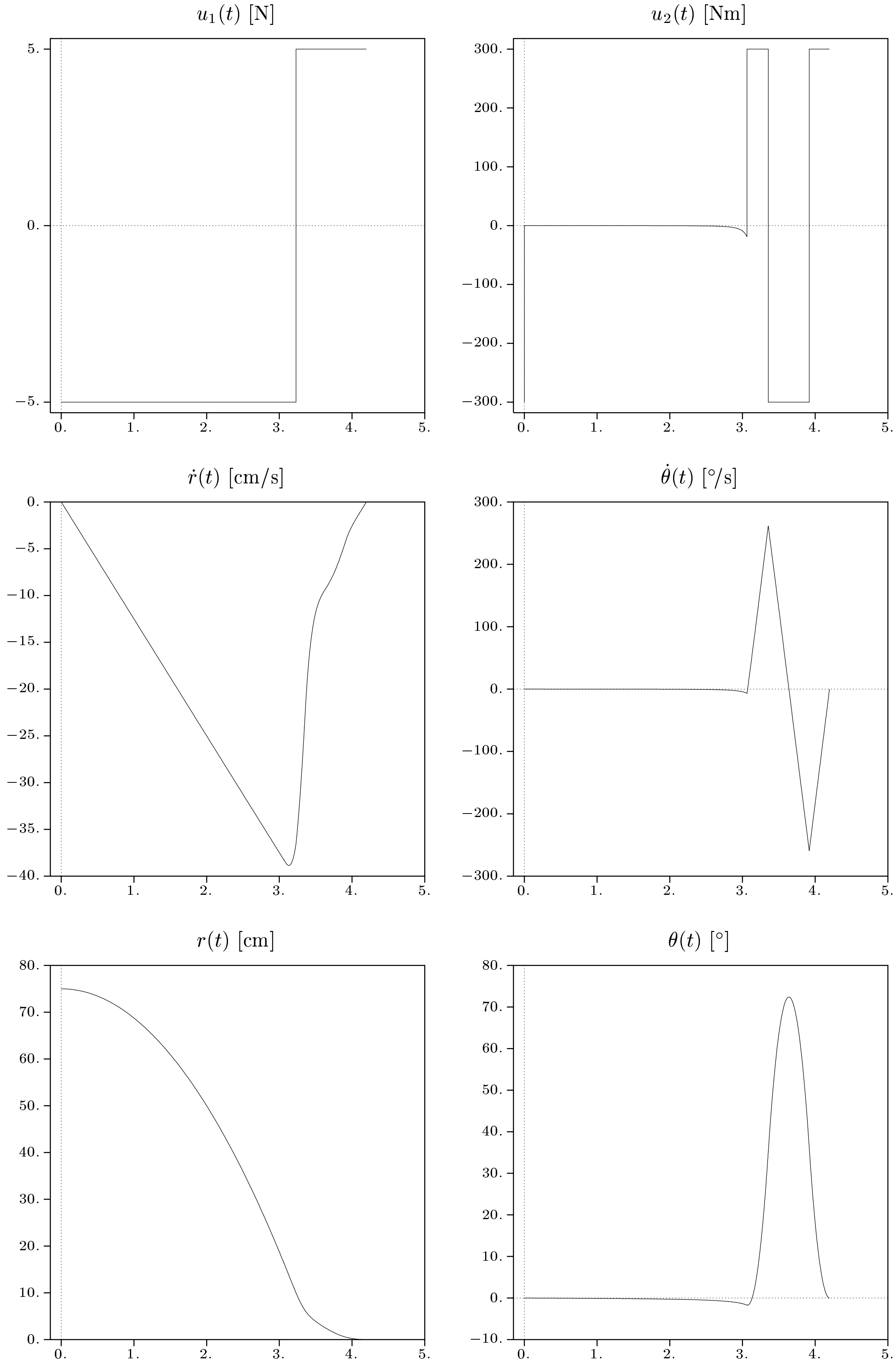
$r_e = 0.4535\,$cm: optimale Steuerungen und Zustandsvariablen



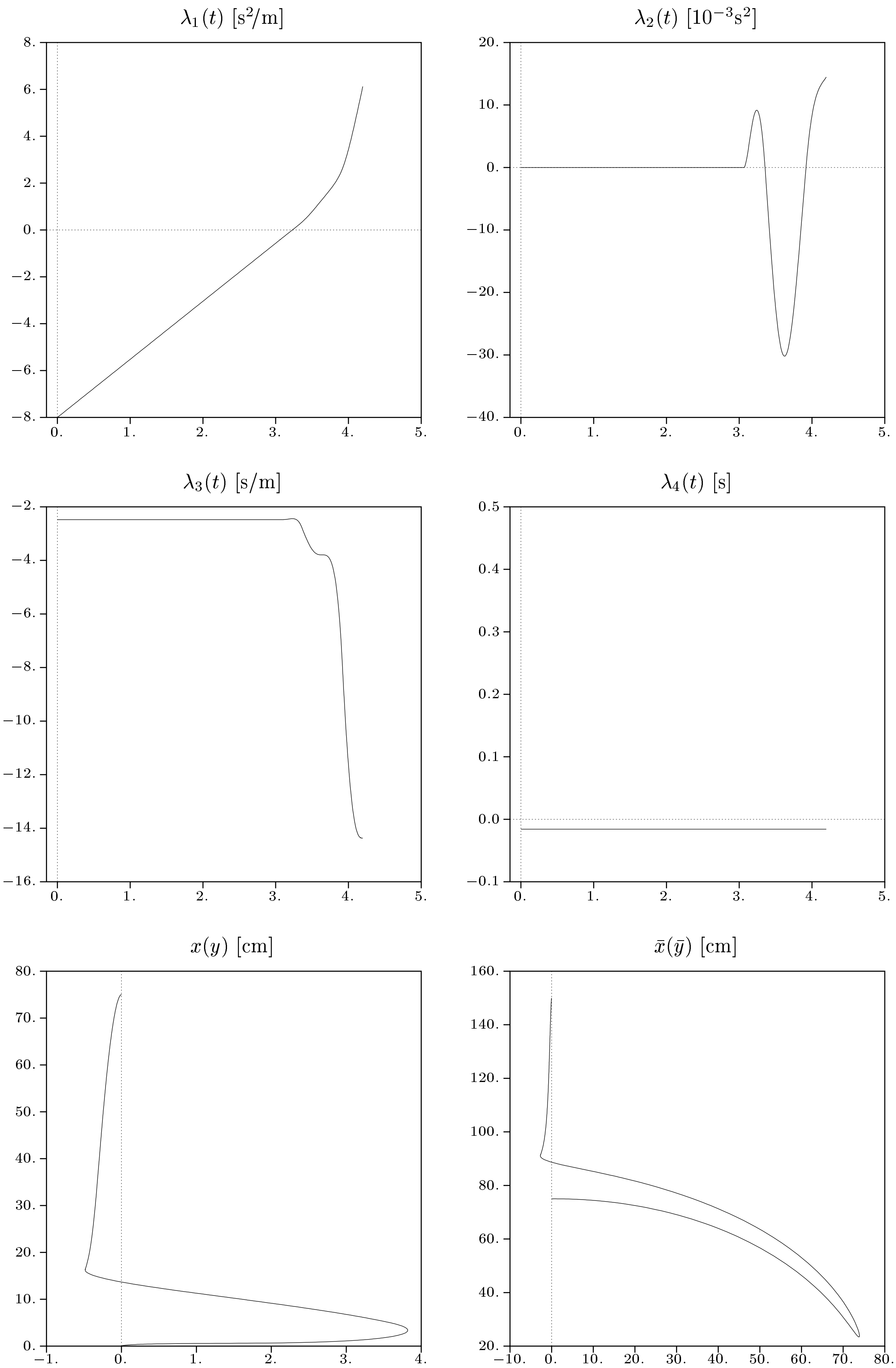
$r_e = 0.4535\,\text{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last



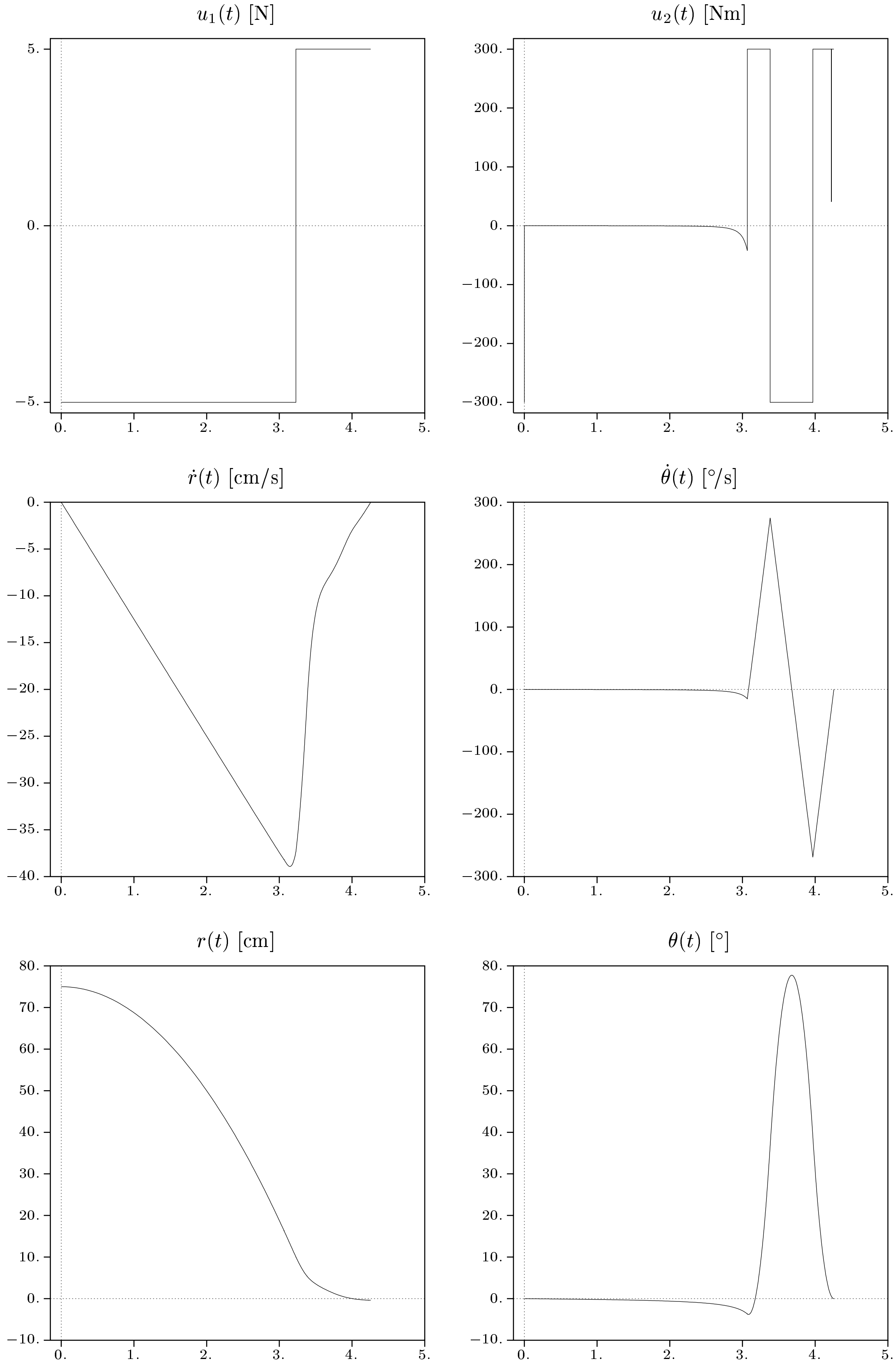
$r_e = 0\,\text{cm}$: optimale Steuerungen und Zustandsvariablen



$r_e = 0\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

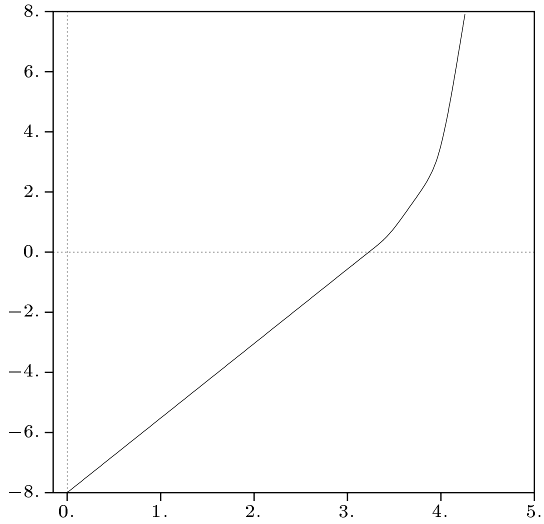


$r_e = -0.3666$ cm: optimale Steuerungen und Zustandsvariablen

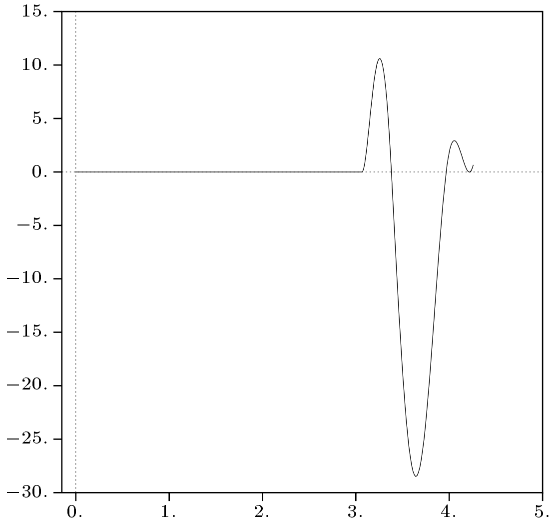


$r_e = -0.3666\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

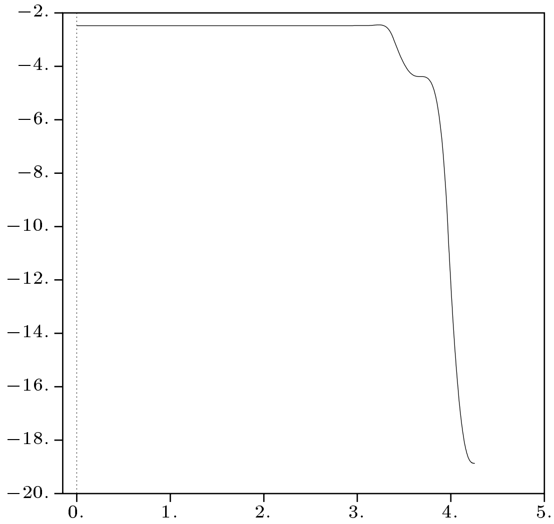
$\lambda_1(t)$ [s²/m]



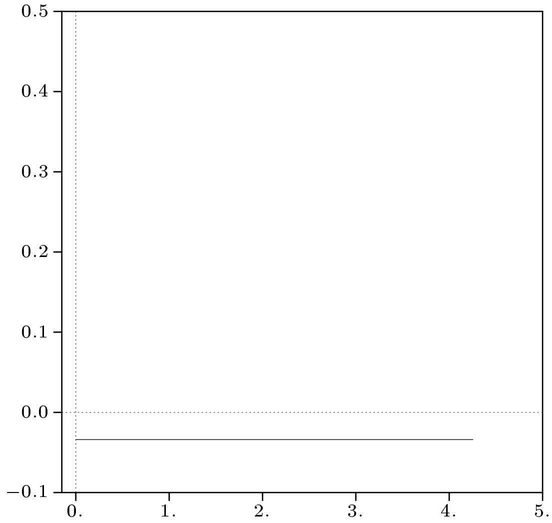
$\lambda_2(t)$ [10^{-3}s²]



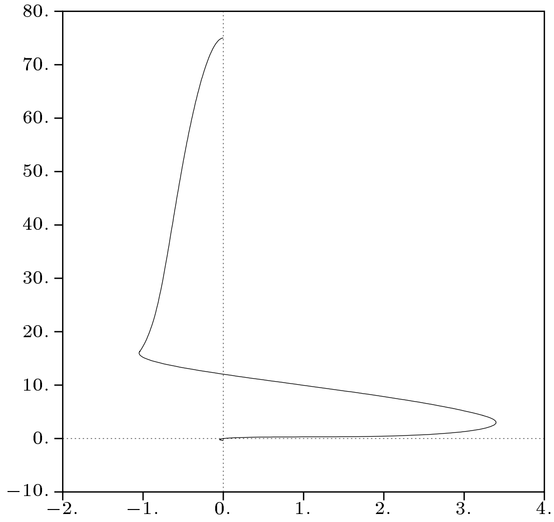
$\lambda_3(t)$ [s/m]



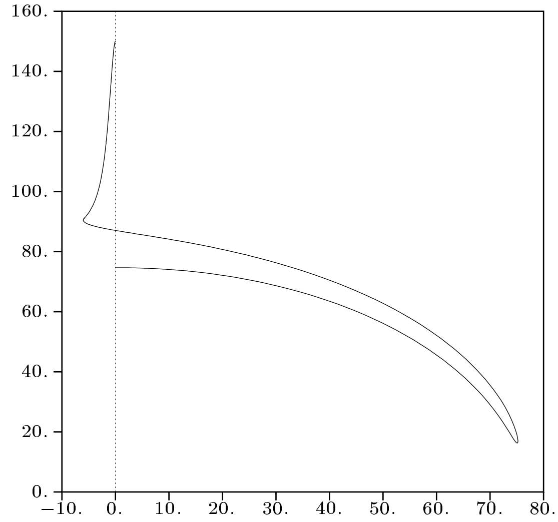
$\lambda_4(t)$ [s]



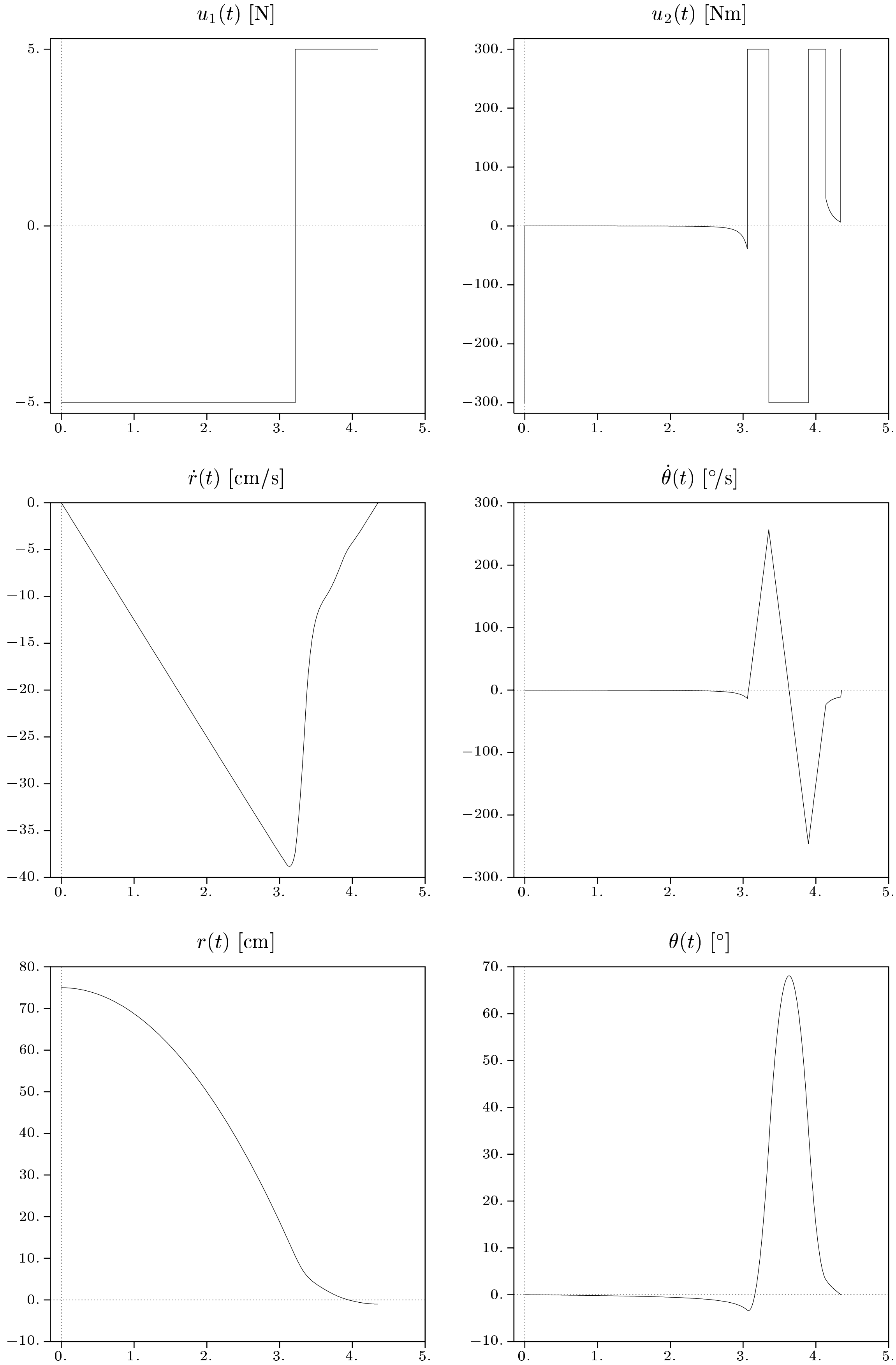
$x(y)$ [cm]



$\bar{x}(\bar{y})$ [cm]

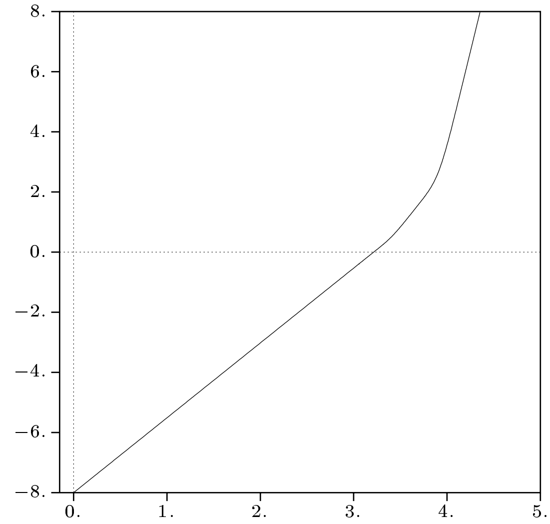


$r_e = -1\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

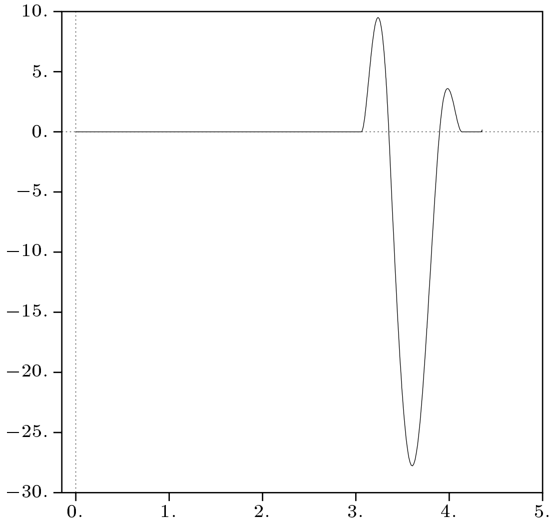


$r_e = -1\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

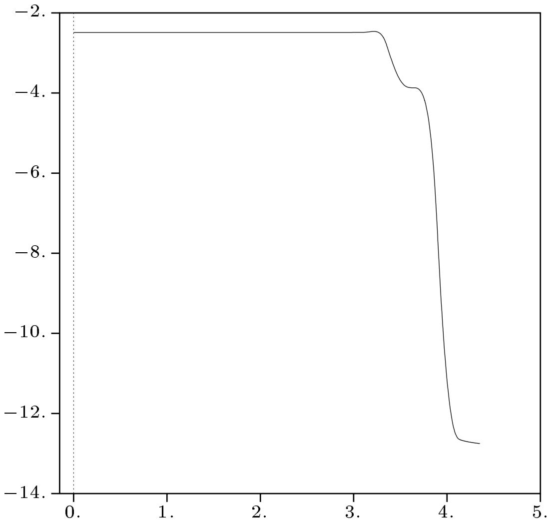
$\lambda_1(t)$ [s²/m]



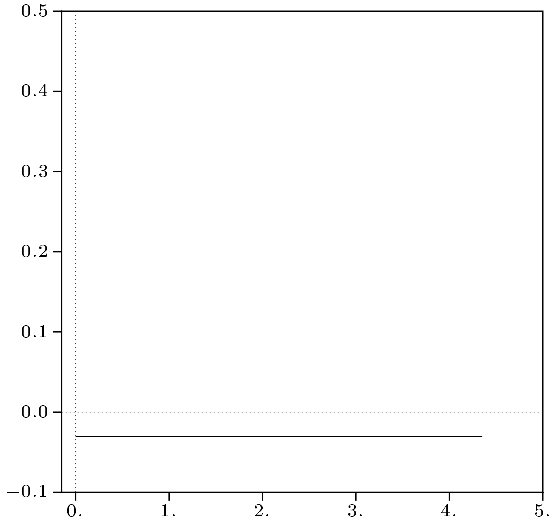
$\lambda_2(t)$ [10^{-3}s²]



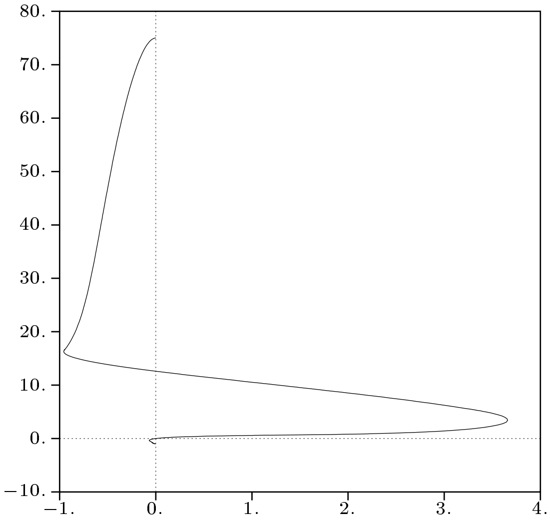
$\lambda_3(t)$ [s/m]



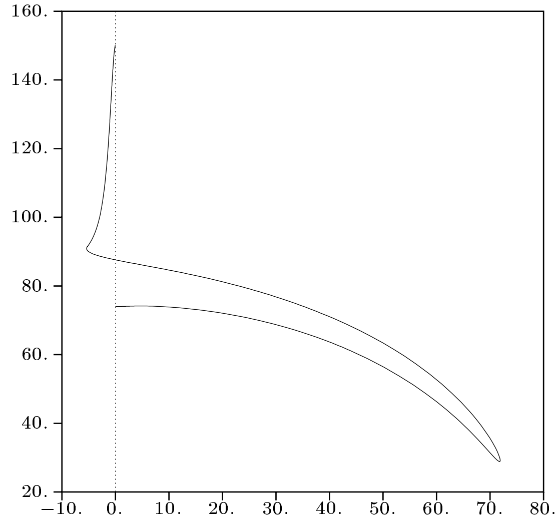
$\lambda_4(t)$ [s]



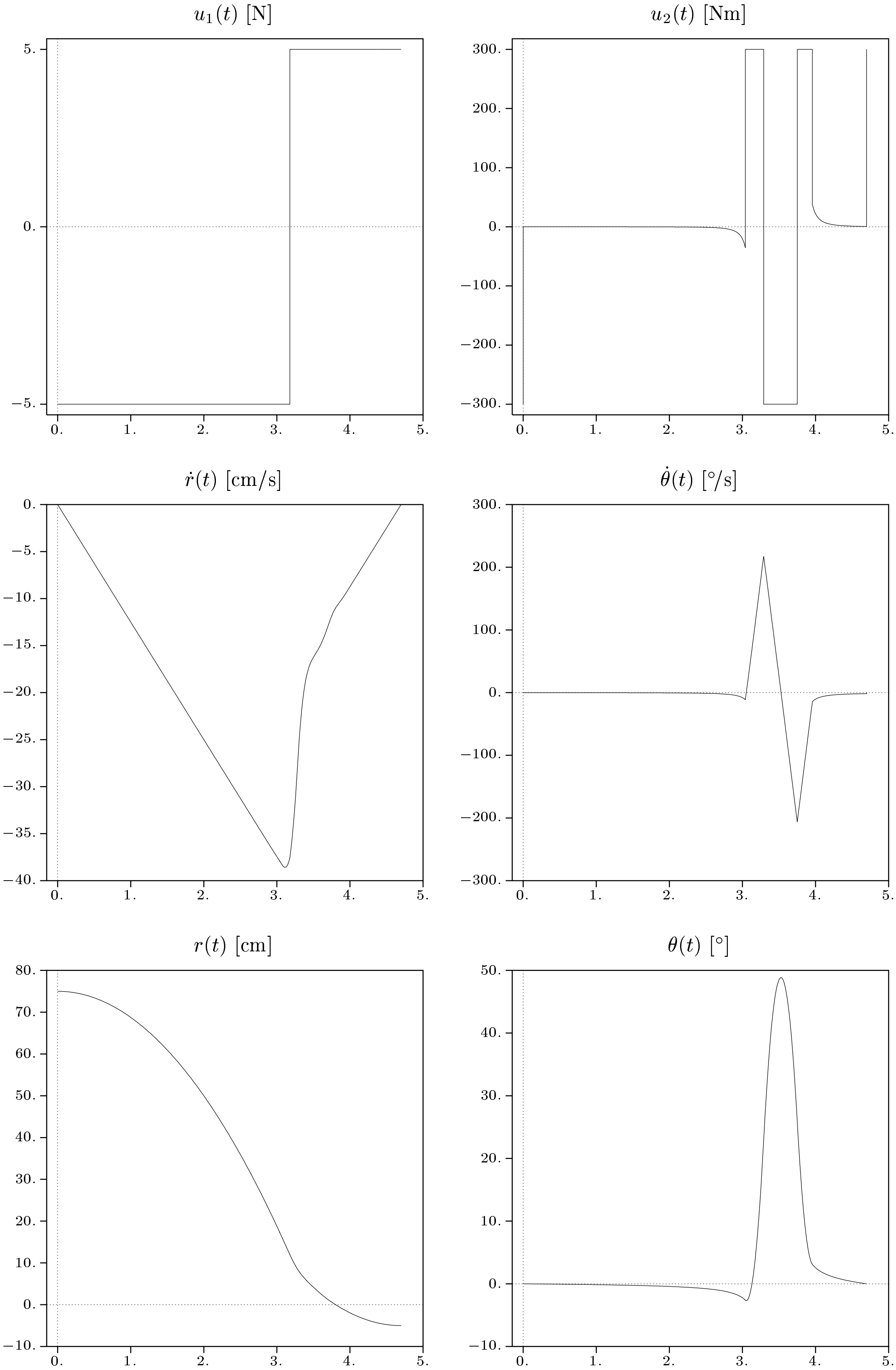
$x(y)$ [cm]



$\bar{x}(\bar{y})$ [cm]

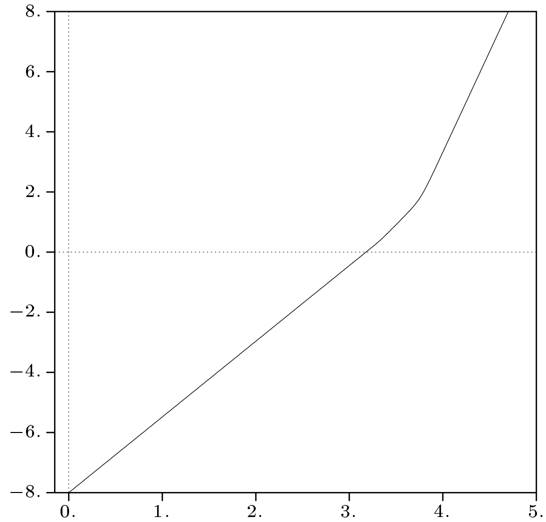


$r_e = -5\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen

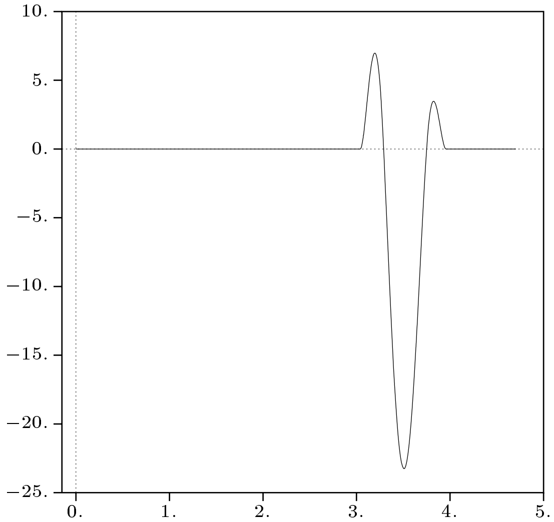


$r_e = -5\,\text{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

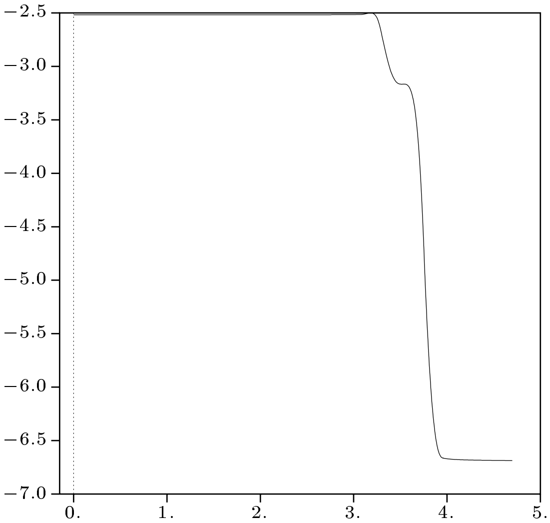
$\lambda_1(t)$ [s²/m]



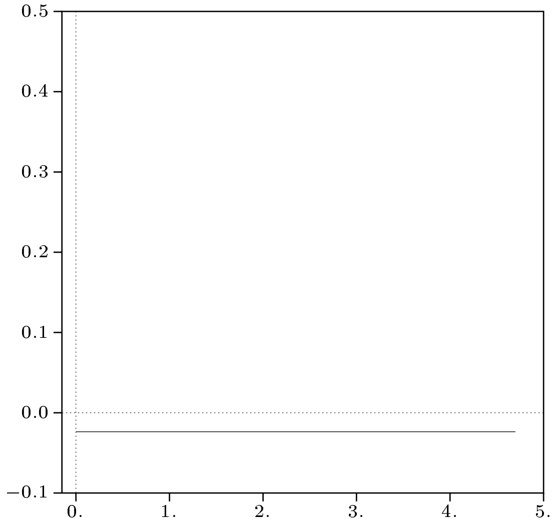
$\lambda_2(t)$ [10^{-3}s²]



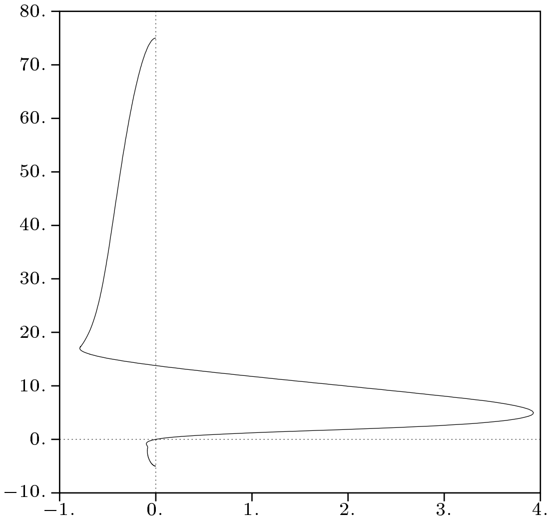
$\lambda_3(t)$ [s/m]



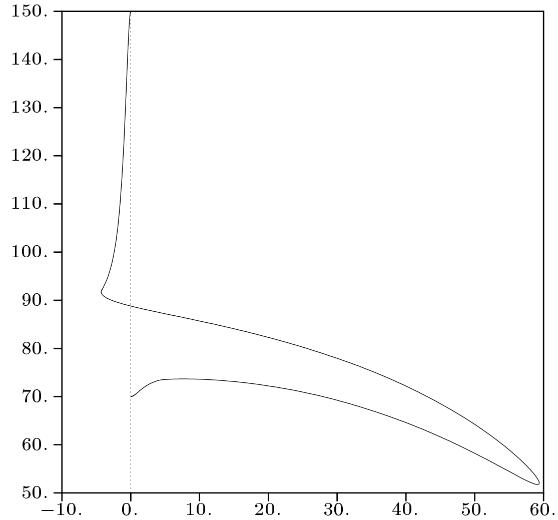
$\lambda_4(t)$ [s]



$x(y)$ [cm]

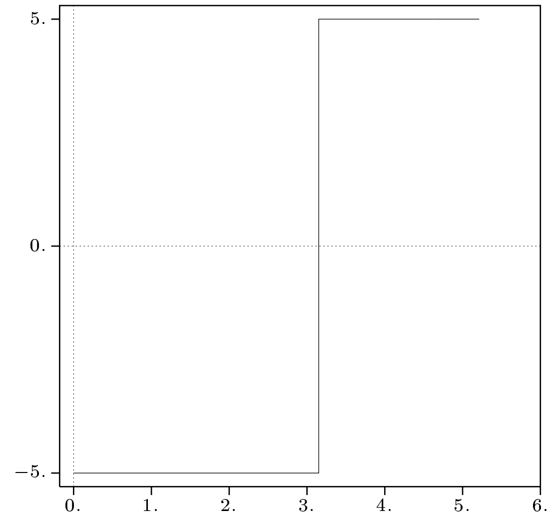


$\bar{x}(\bar{y})$ [cm]

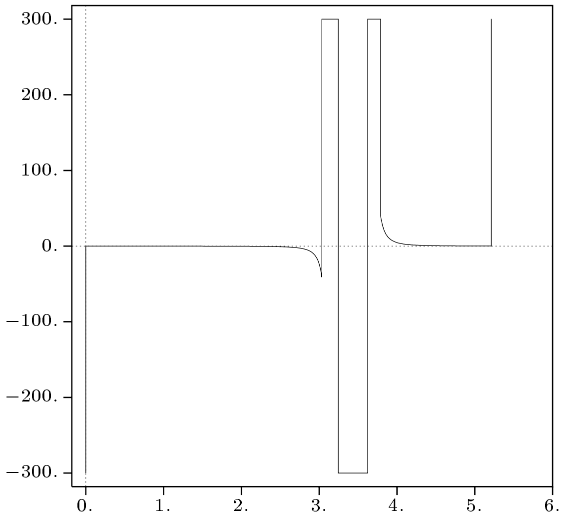


$r_e = -15\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen

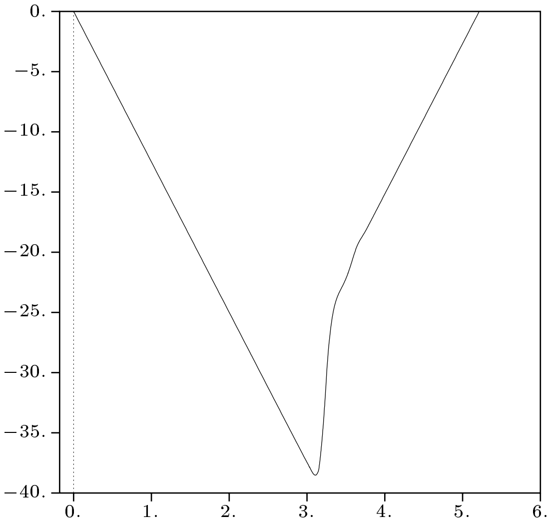
$u_1(t)$ [N]



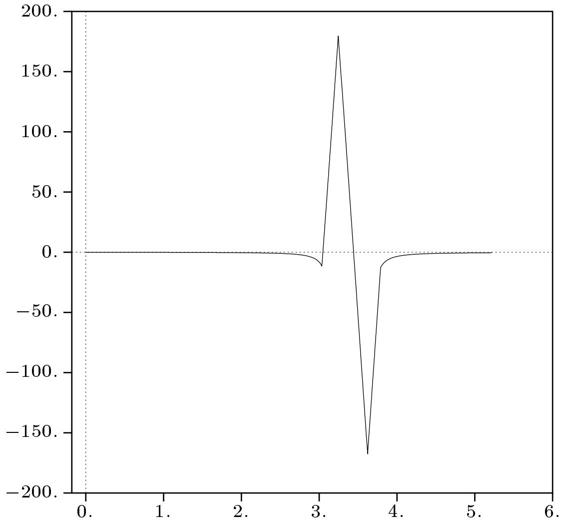
$u_2(t)$ [Nm]



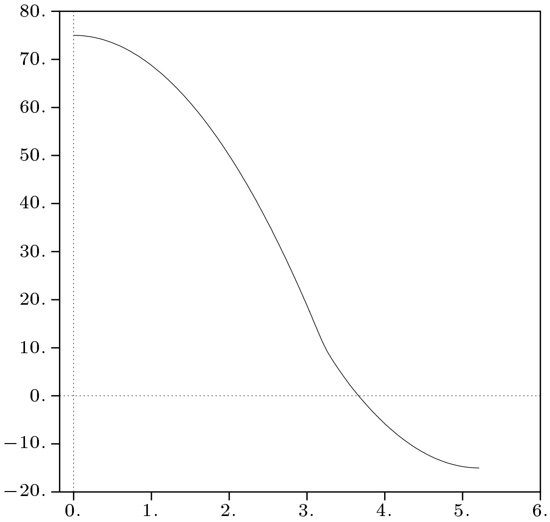
$\dot{r}(t)$ [cm/s]



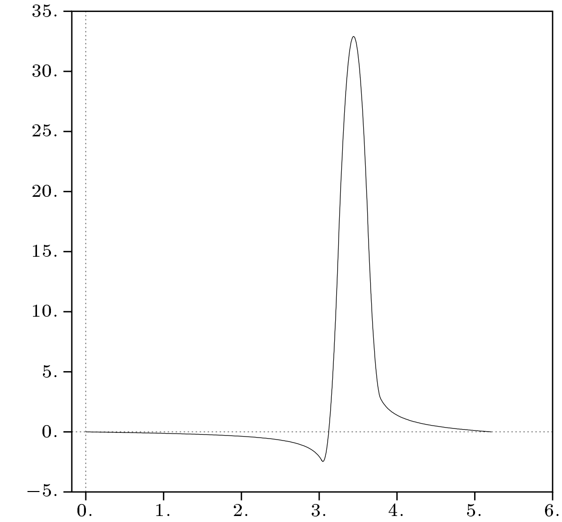
$\dot{\theta}(t)$ [°/s]



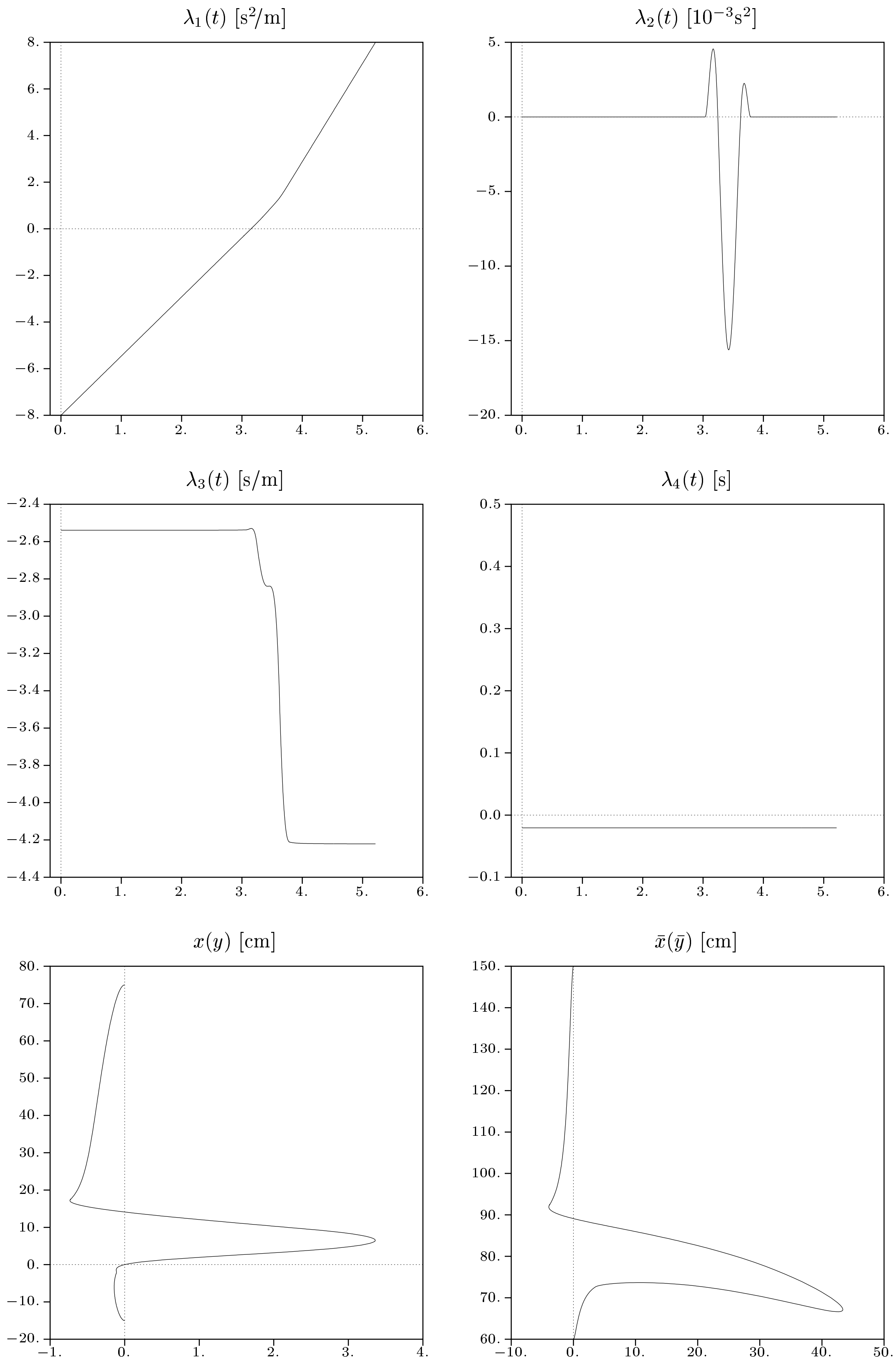
$r(t)$ [cm]



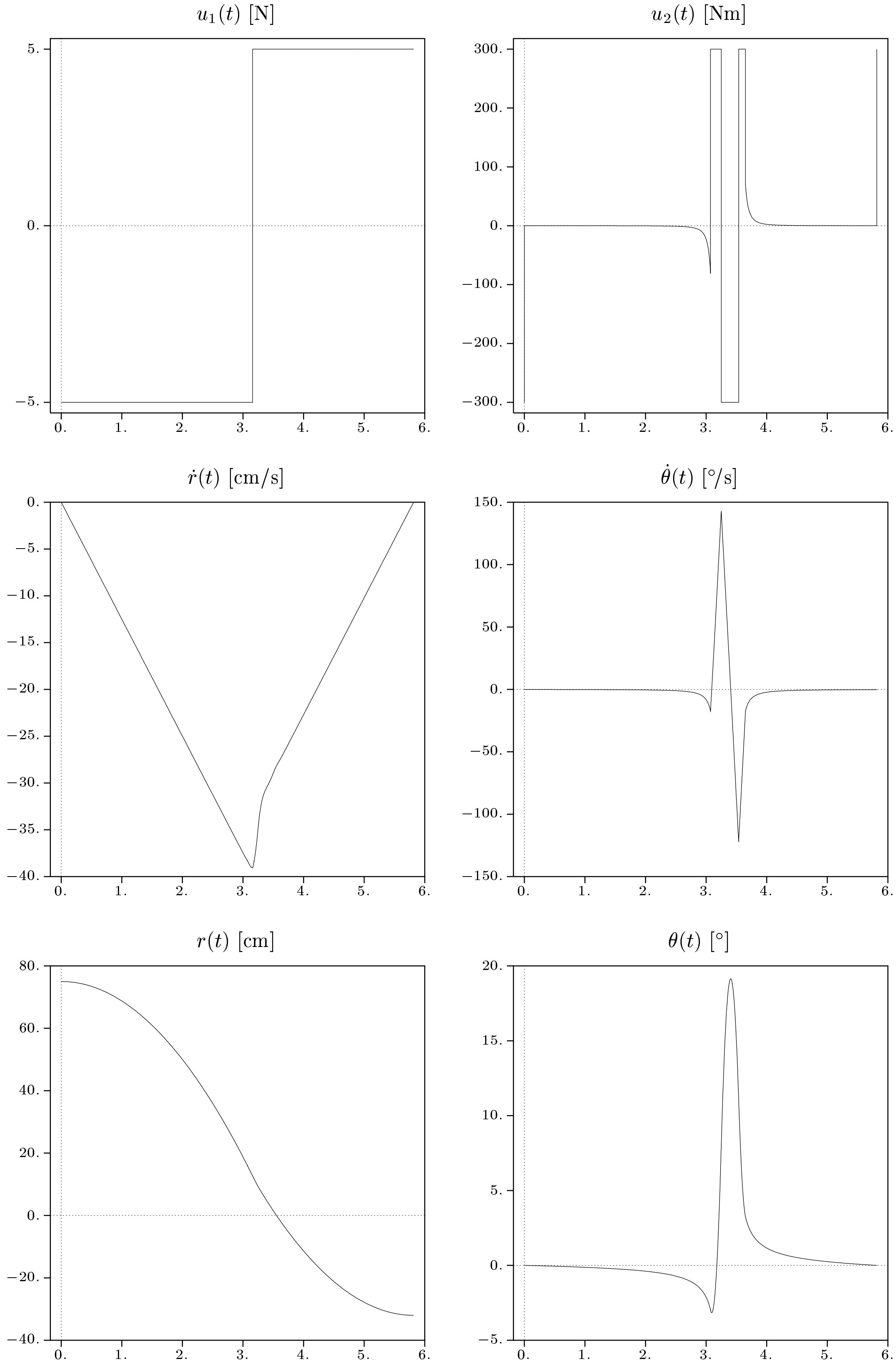
$\theta(t)$ [°]



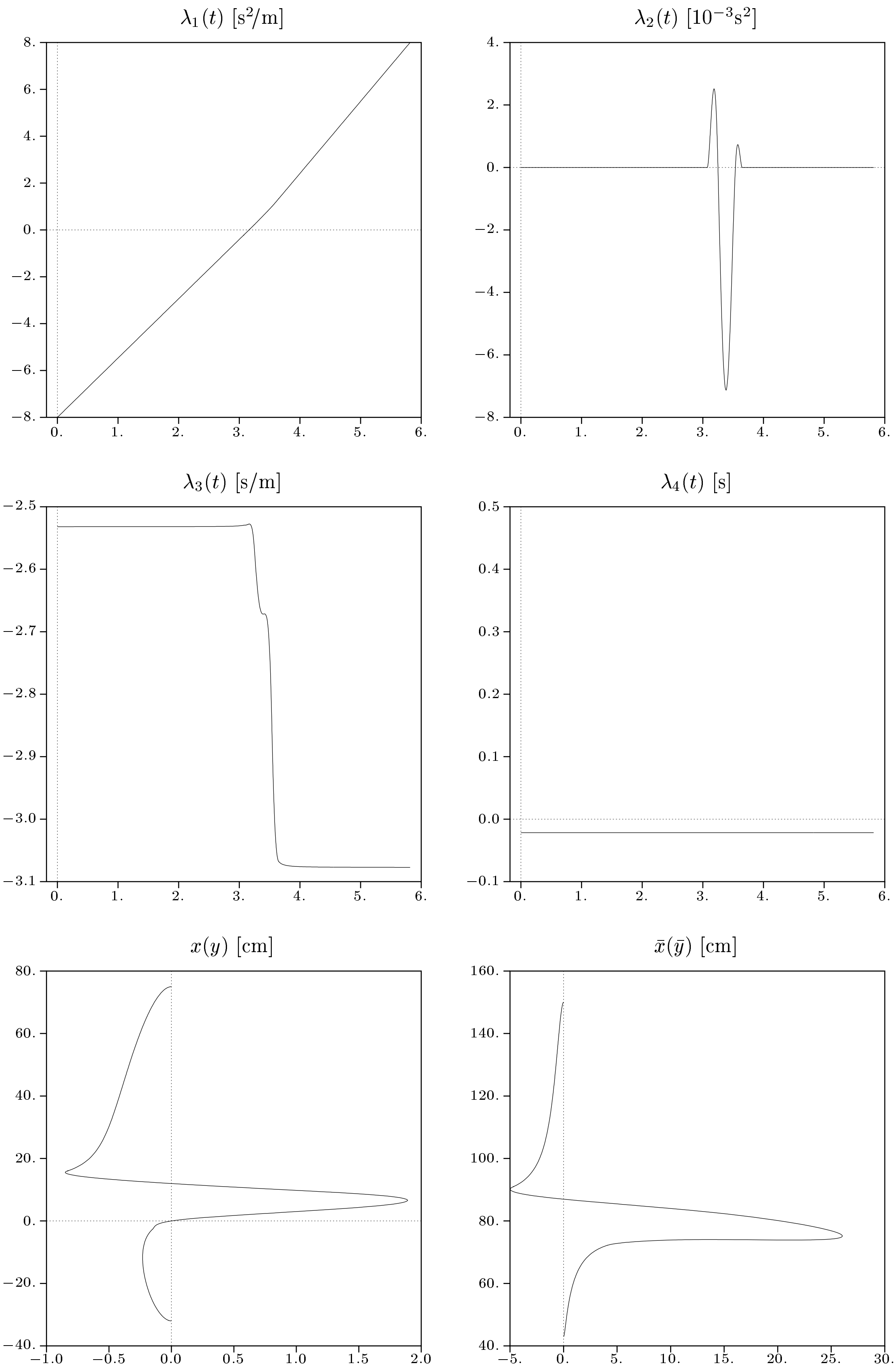
$r_e = -15\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last



$r_e = -32\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen

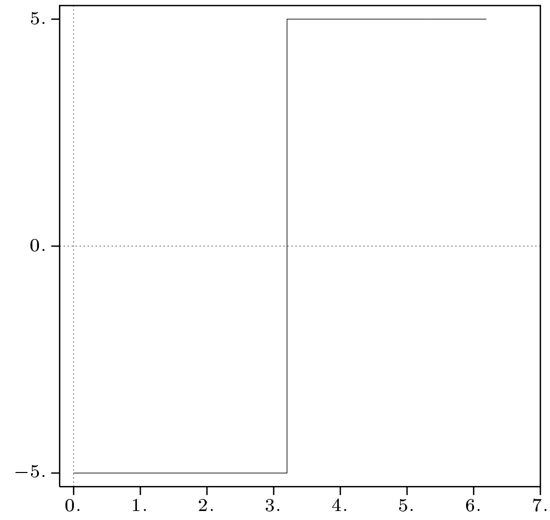


$r_e = -32\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

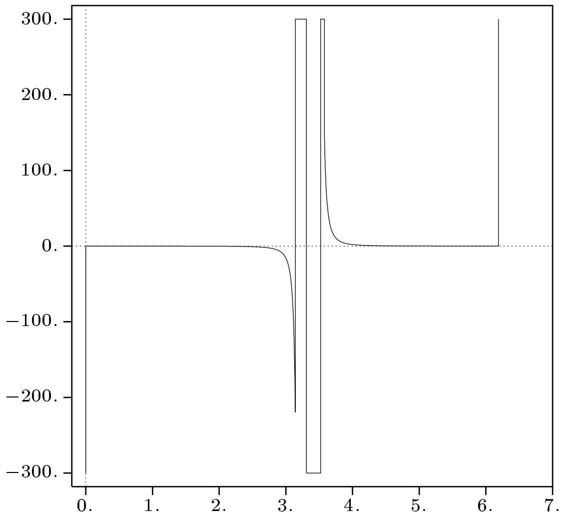


$r_e = -45\,\mathrm{cm}$: optimale Steuerungen und Zustandsvariablen

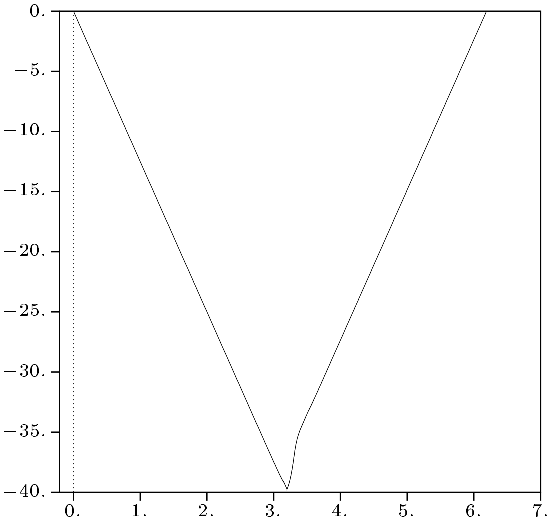
$u_1(t)$ [N]



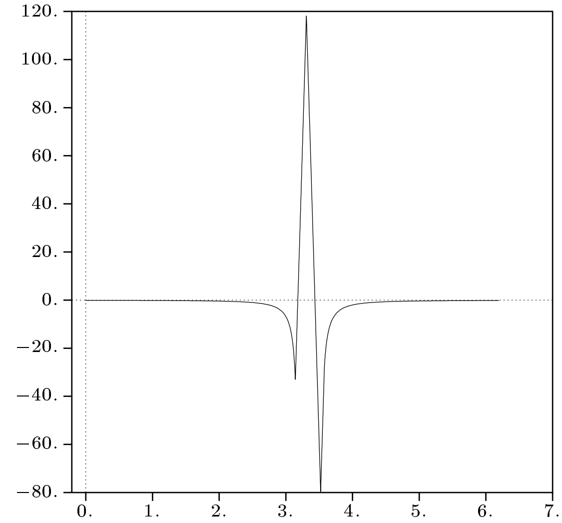
$u_2(t)$ [Nm]



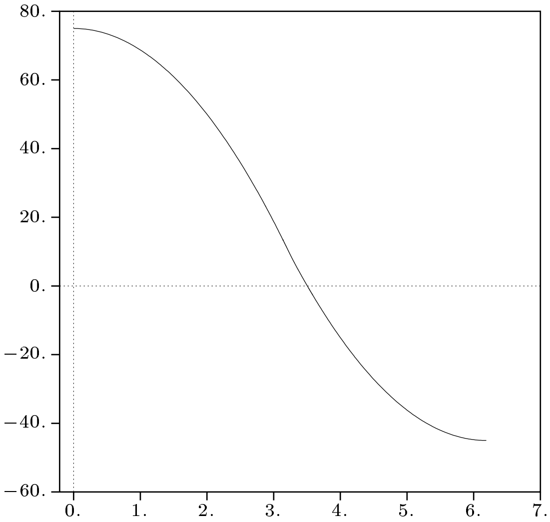
$\dot{r}(t)$ [cm/s]



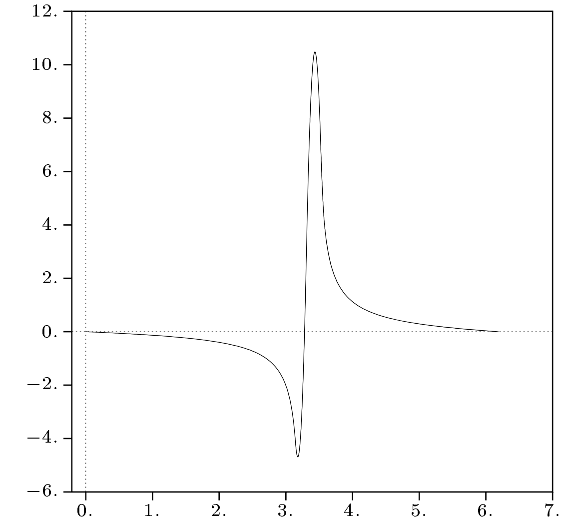
$\dot{\theta}(t)$ [°/s]



$r(t)$ [cm]

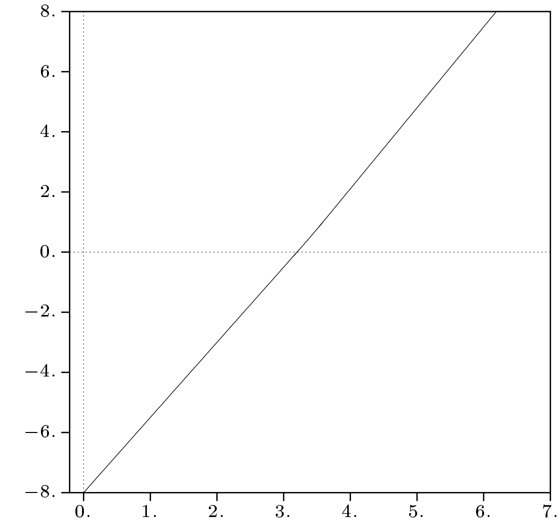


$\theta(t)$ [°]

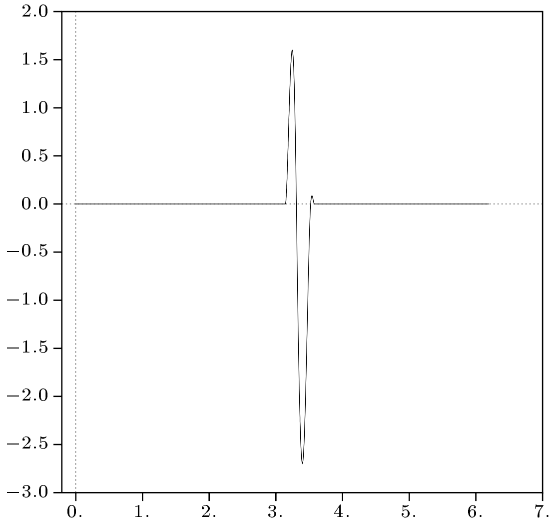


$r_e = -45\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

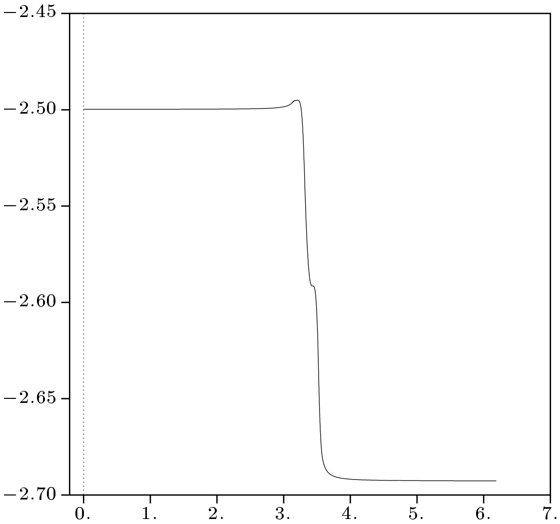
$\lambda_1(t)$ [s²/m]



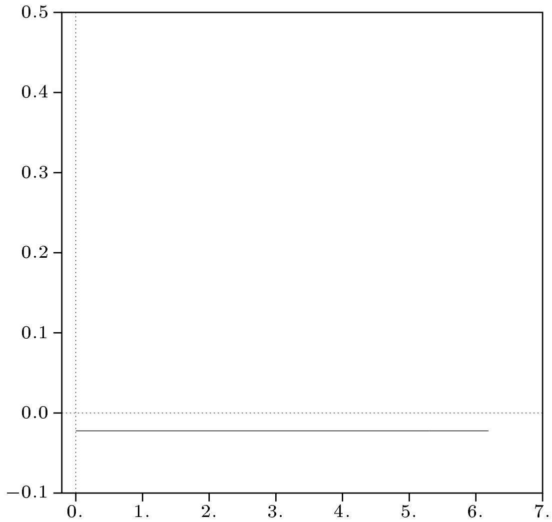
$\lambda_2(t)$ $[10^{-3}\mathrm{s}^2]$



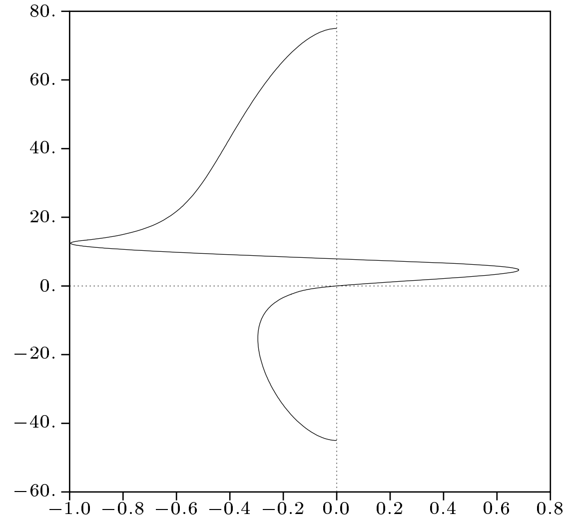
$\lambda_3(t)$ [s/m]



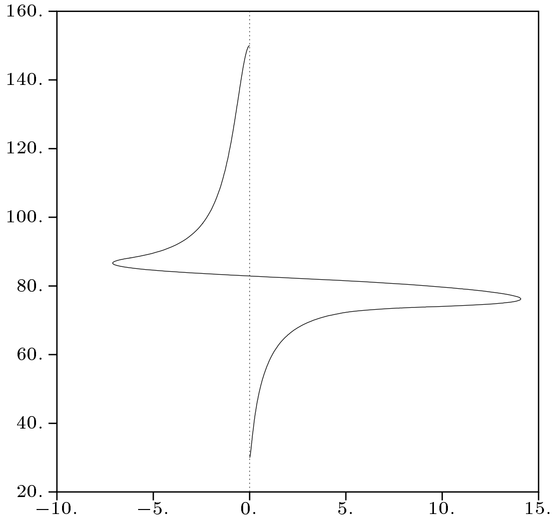
$\lambda_4(t)$ [s]



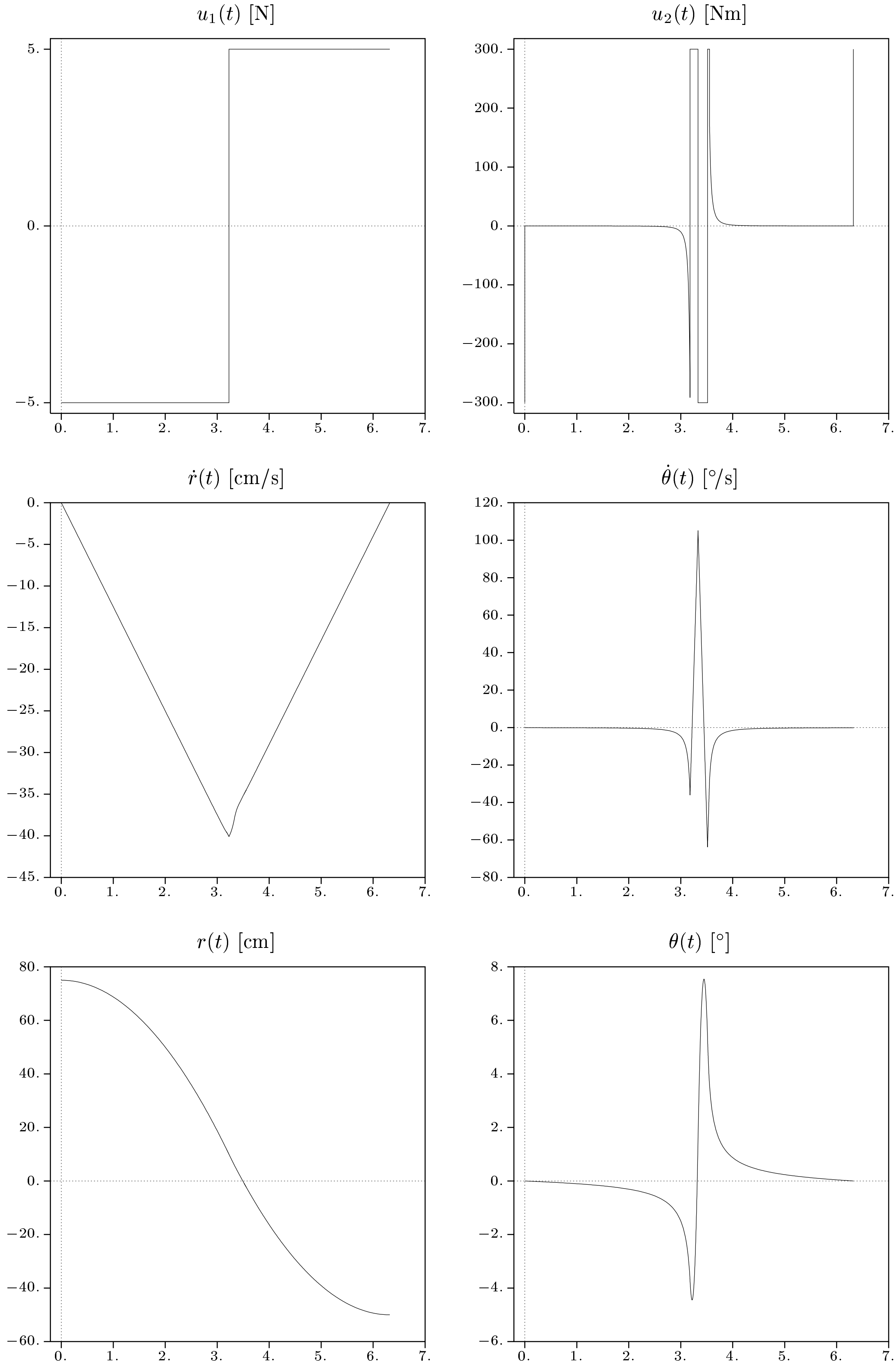
$x(y)$ [cm]



$\bar{x}(\bar{y})$ [cm]

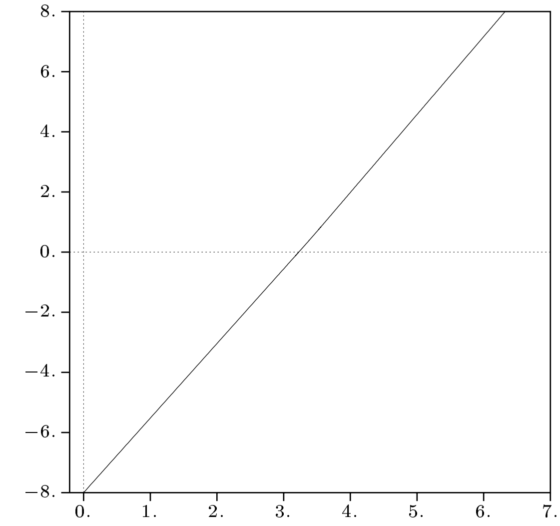


$r_e = -50\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

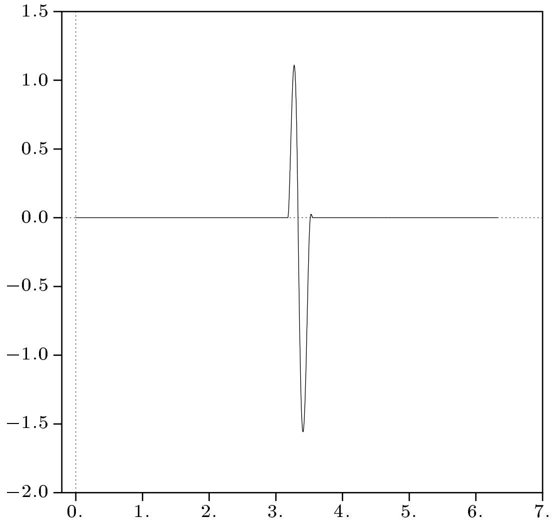


$r_e = -50\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

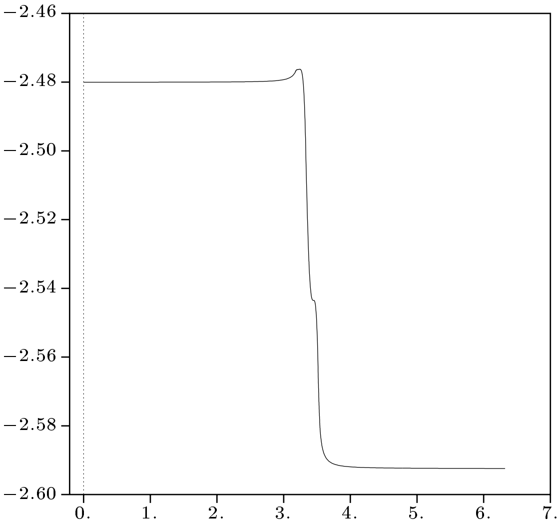
$\lambda_1(t)$ [$\mathrm{s^2/m}$]



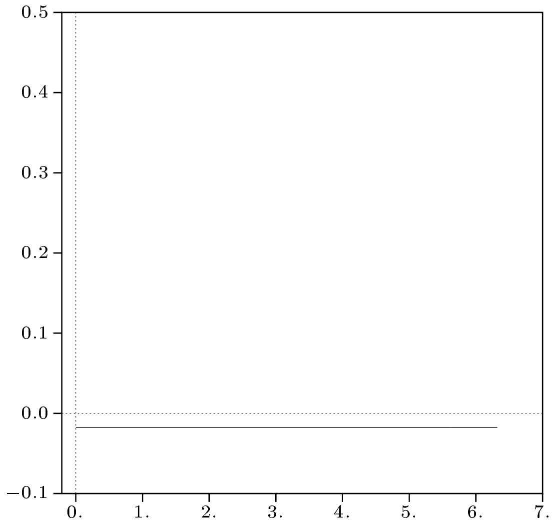
$\lambda_2(t)$ [$10^{-3}\mathrm{s}^2$]



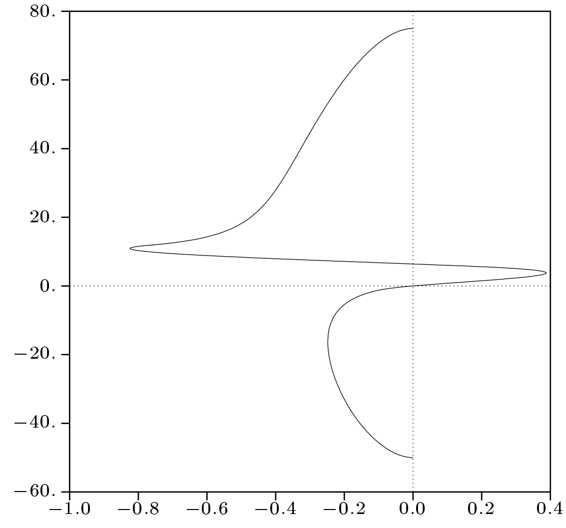
$\lambda_3(t)$ [s/m]



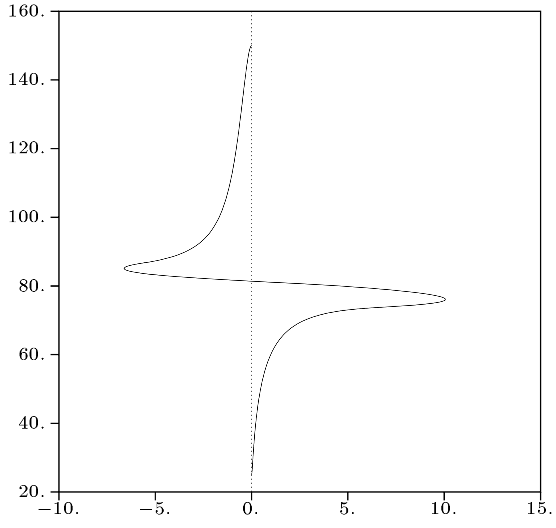
$\lambda_4(t)$ [s]



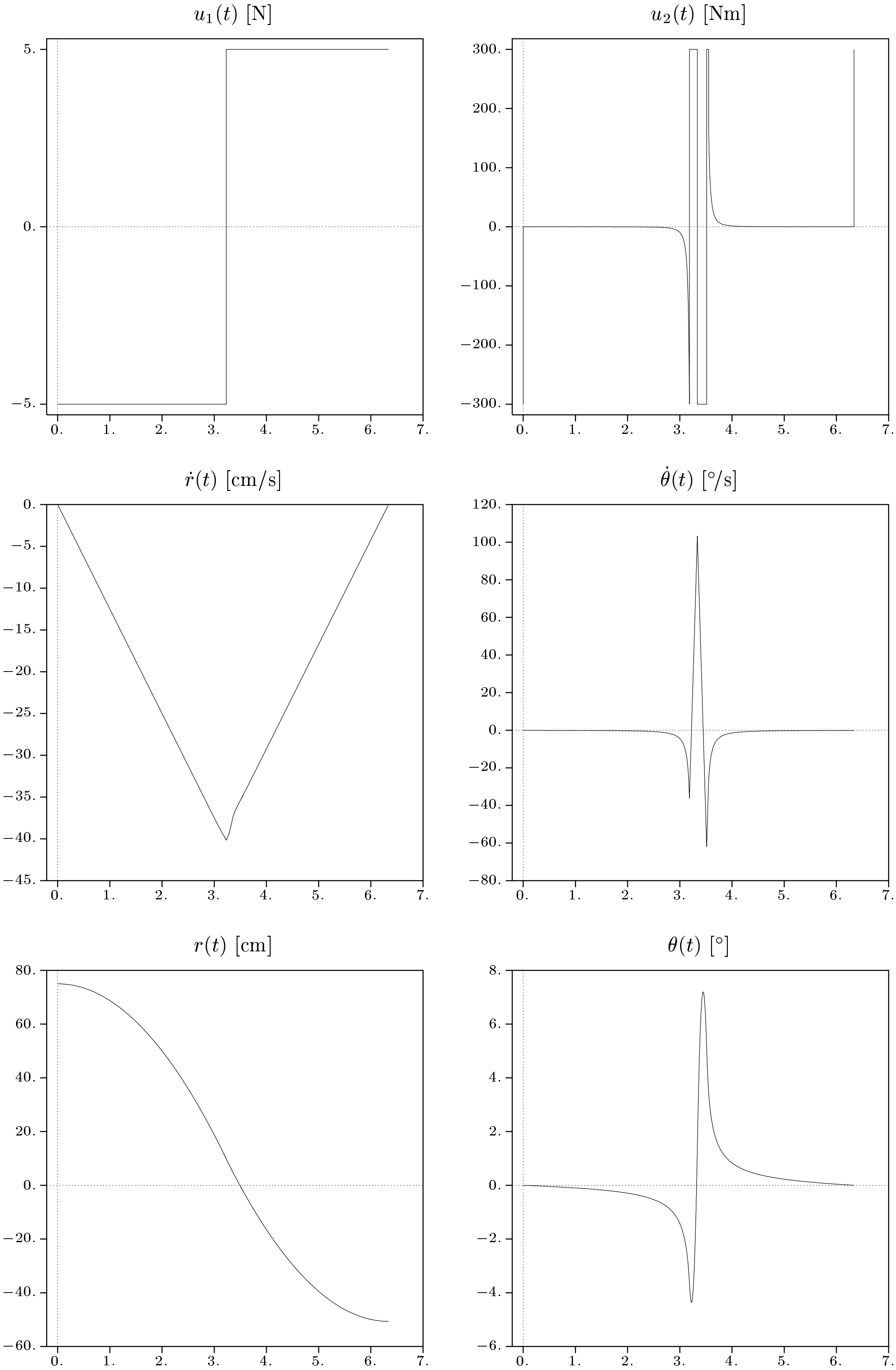
$x(y)$ [cm]



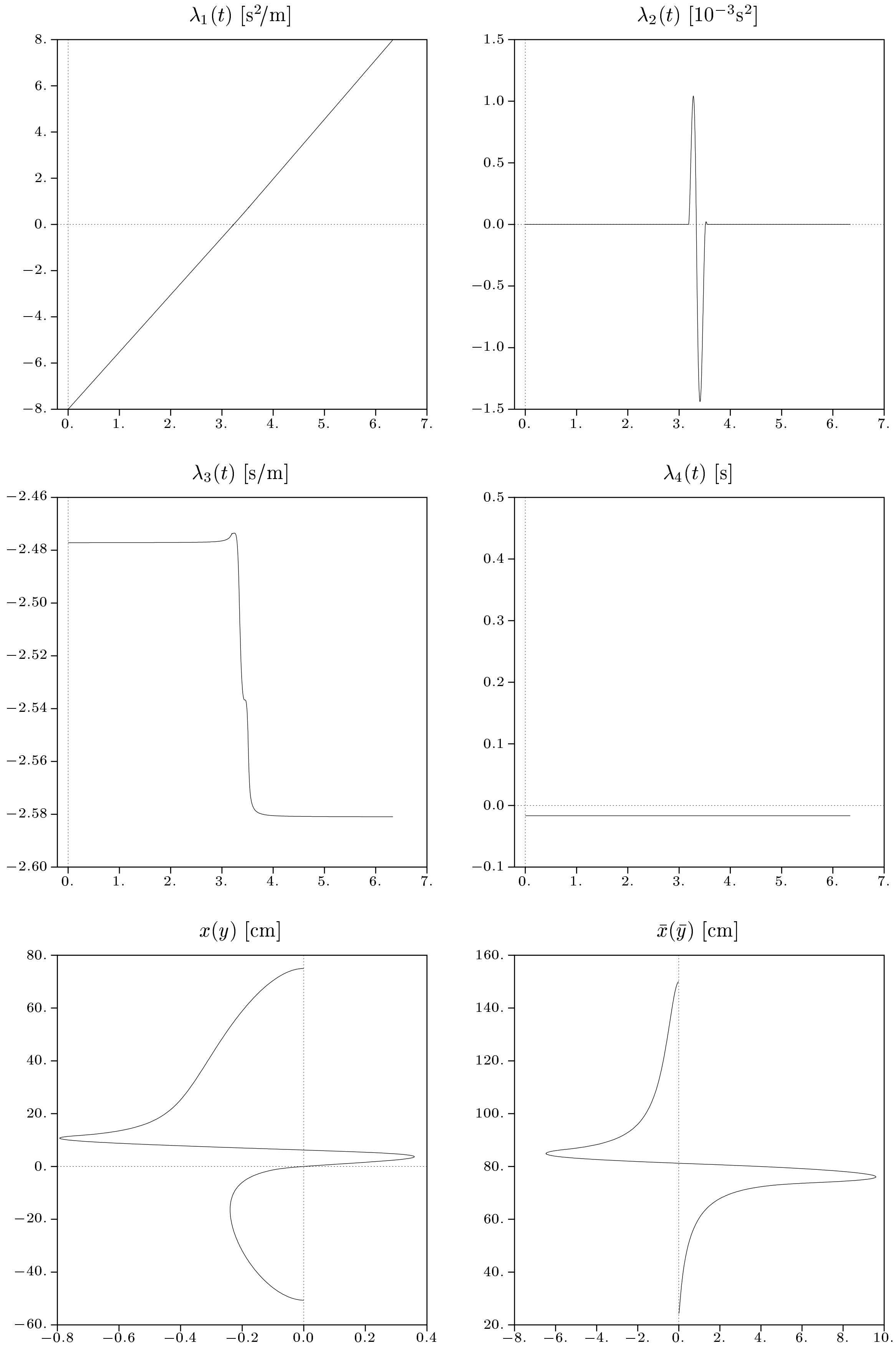
$\bar{x}(\bar{y})$ [cm]



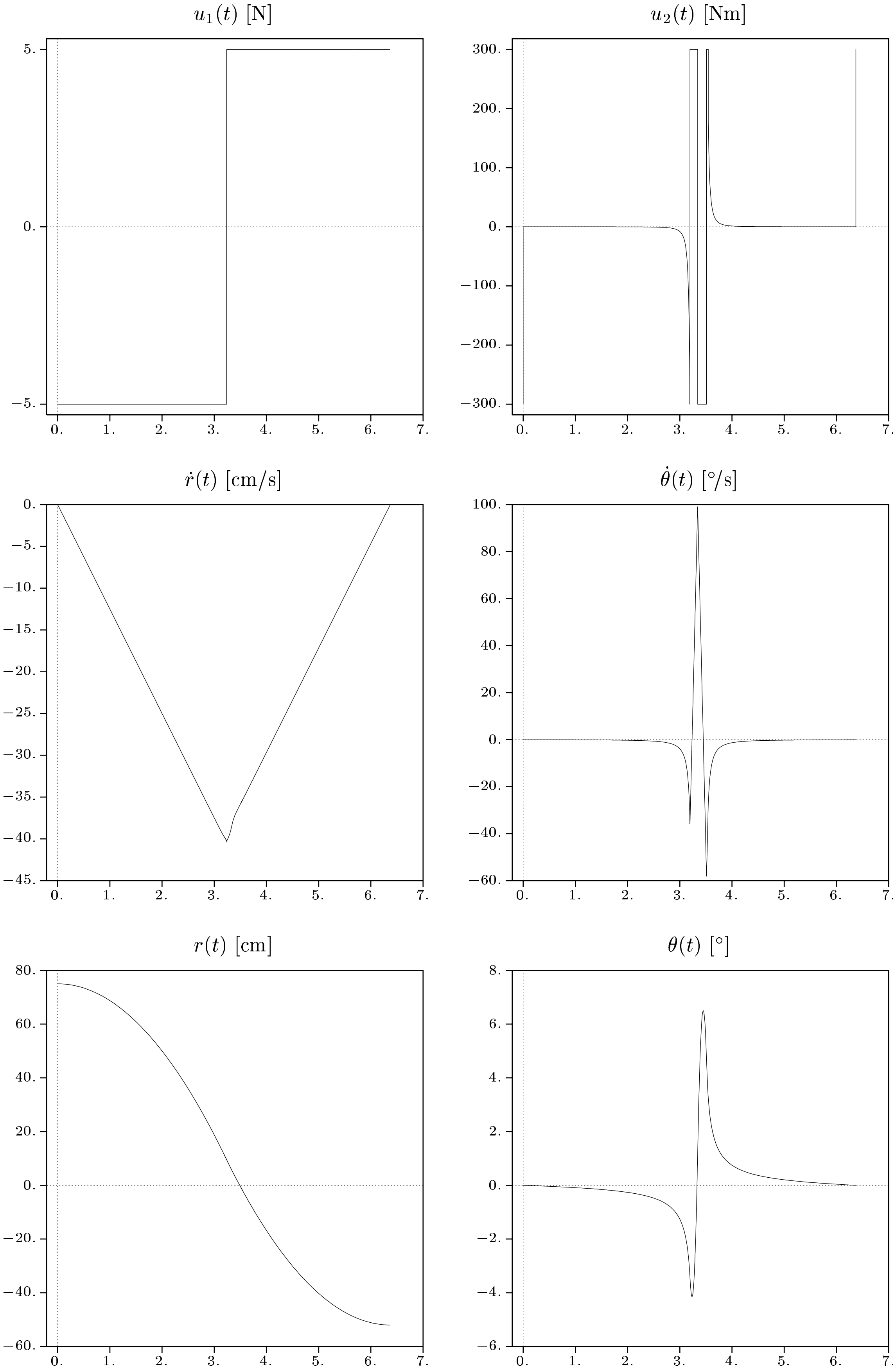
$r_e = -50.6432\,\text{cm}$: optimale Steuerungen und Zustandsvariablen



$r_e = -50.6432\,\text{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

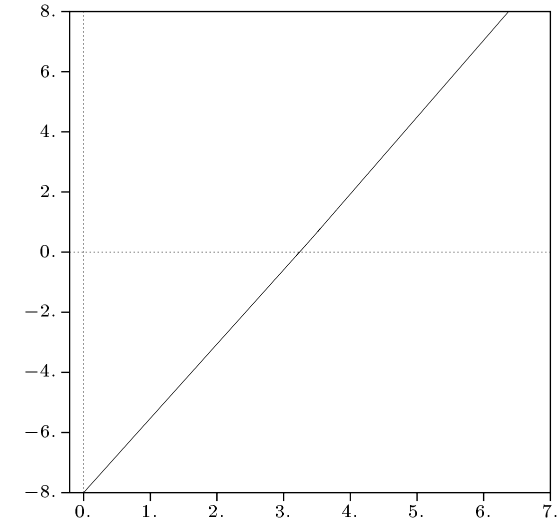


$r_e = -52\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

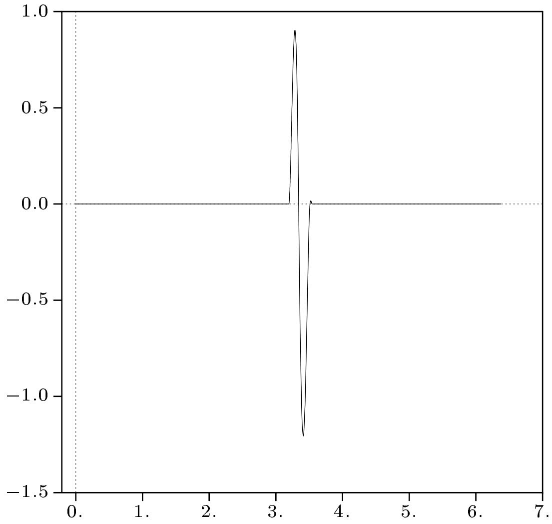


$r_e = -52\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

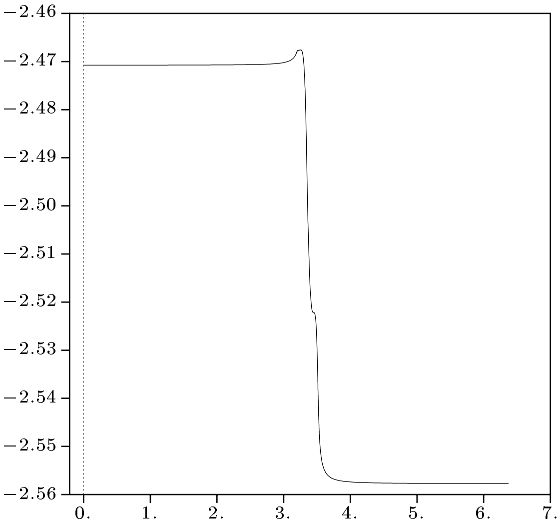
$\lambda_1(t)$ [s²/m]



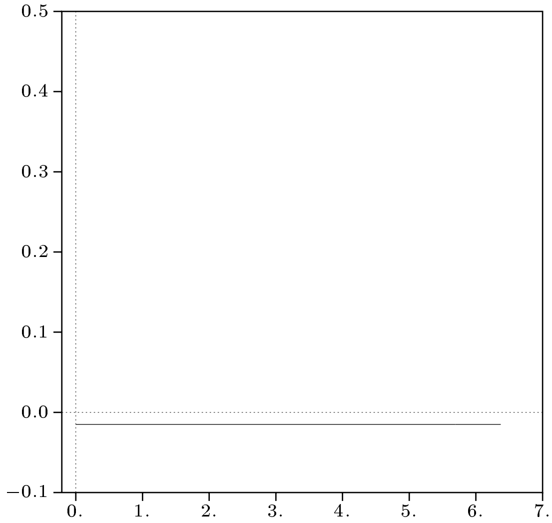
$\lambda_2(t)$ [10^{-3}s²]



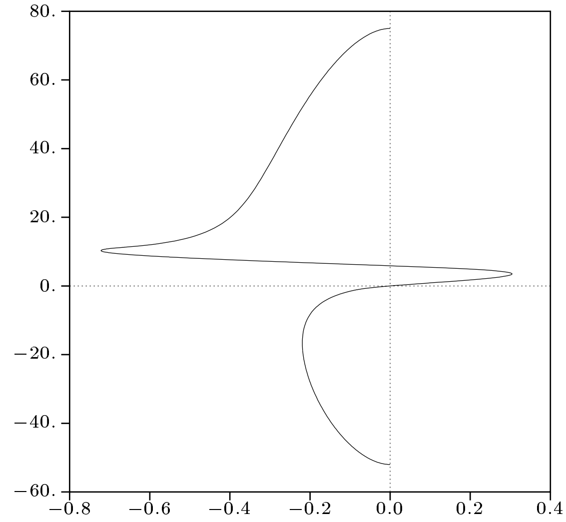
$\lambda_3(t)$ [s/m]



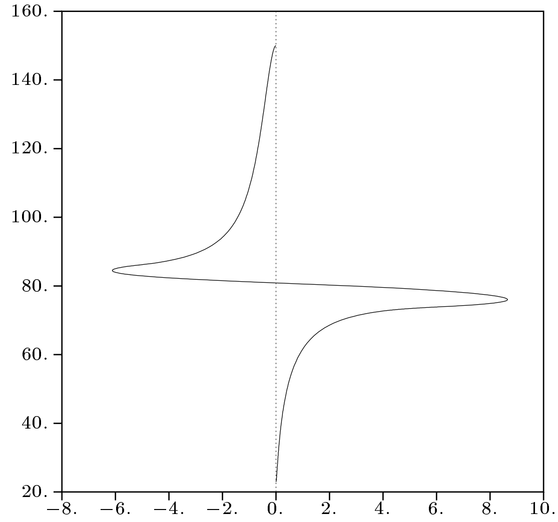
$\lambda_4(t)$ [s]



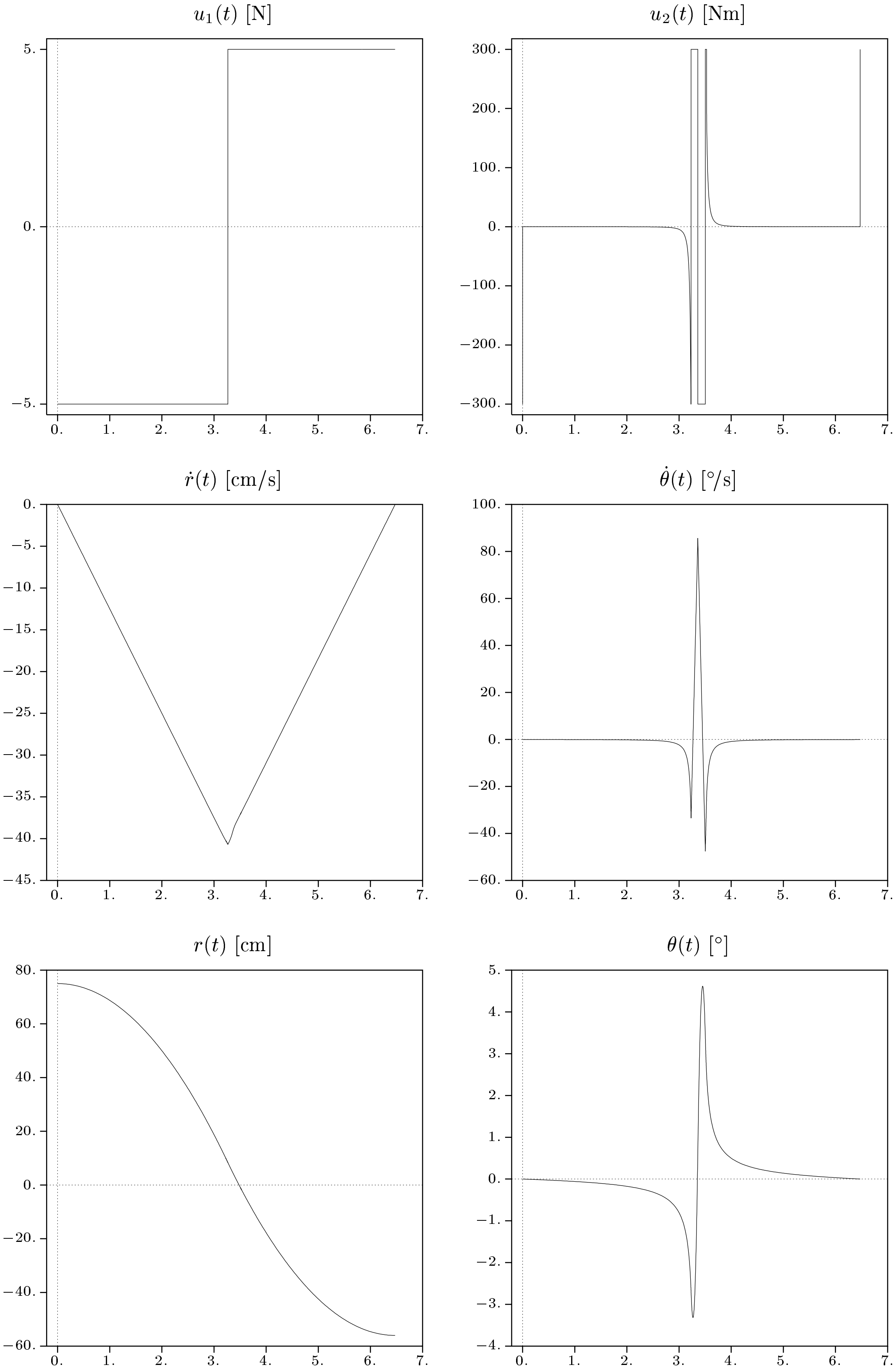
$x(y)$ [cm]



$\bar{x}(\bar{y})$ [cm]

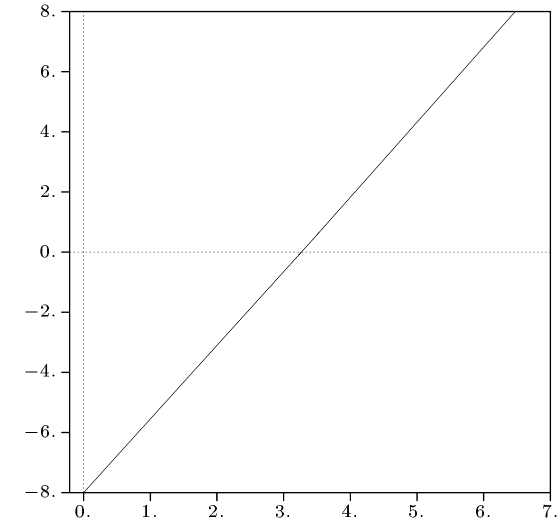


$r_e = -56\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

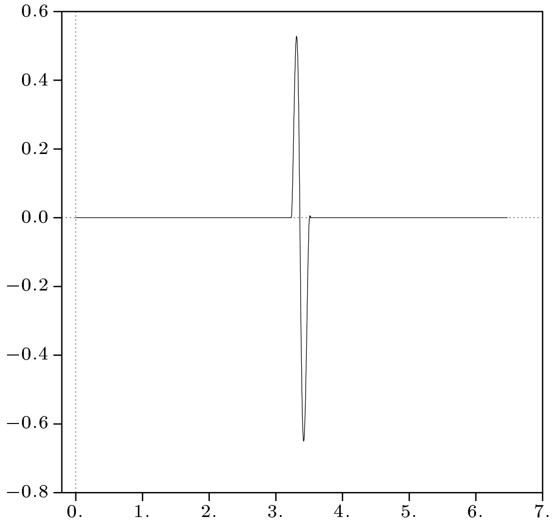


$r_e = -56\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last

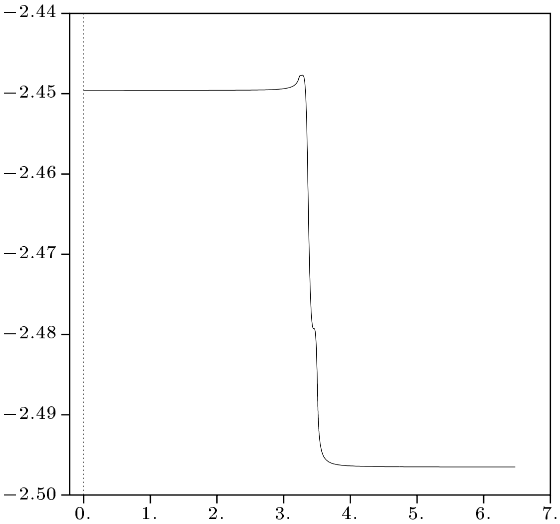
$\lambda_1(t)$ [s^2/m]



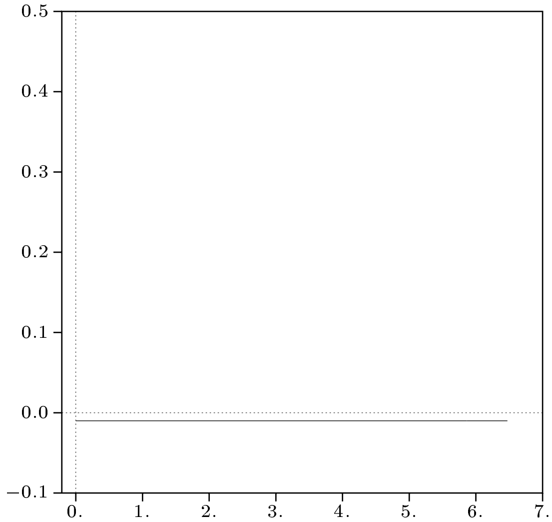
$\lambda_2(t)$ [$10^{-3}\mathrm{s}^2$]



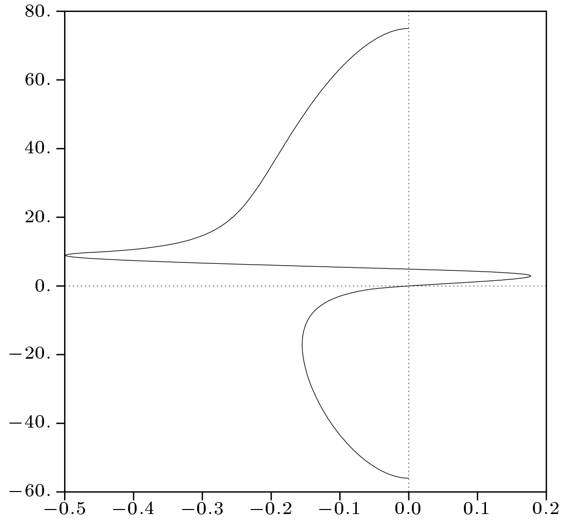
$\lambda_3(t)$ [s/m]



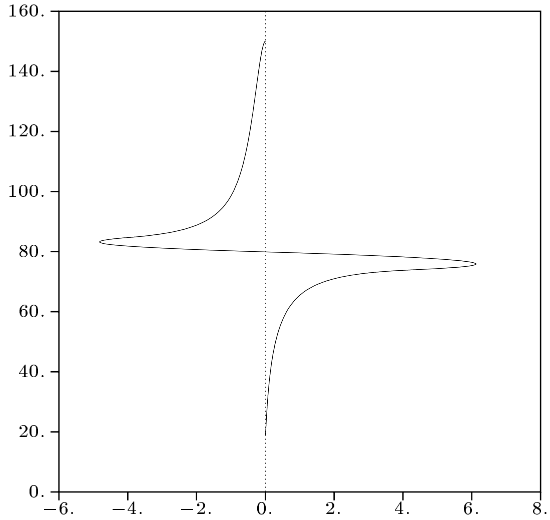
$\lambda_4(t)$ [s]



$x(y)$ [cm]

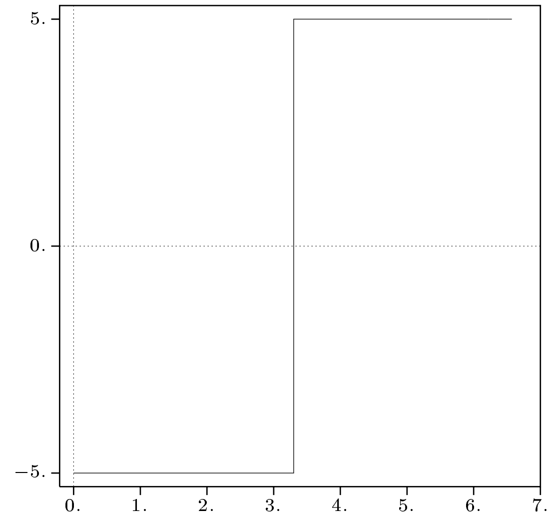


$\bar{x}(\bar{y})$ [cm]

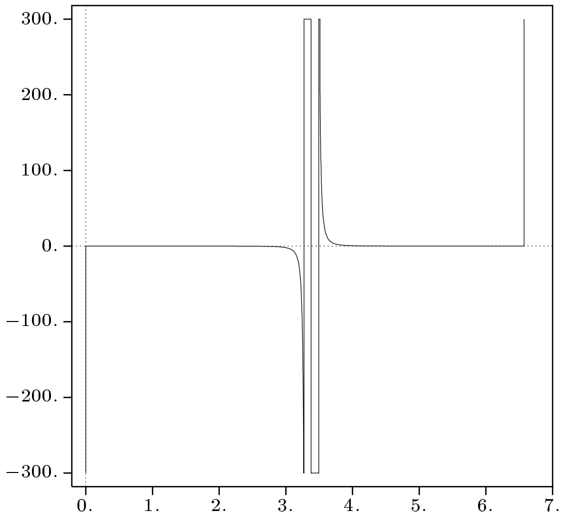


$r_e = -60\,\text{cm}$: optimale Steuerungen und Zustandsvariablen

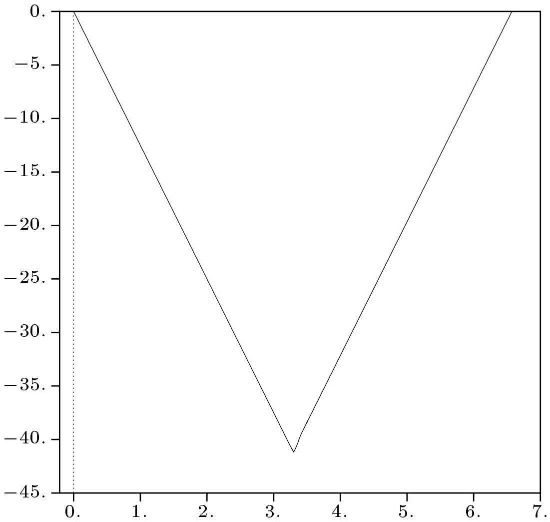
$u_1(t)$ [N]



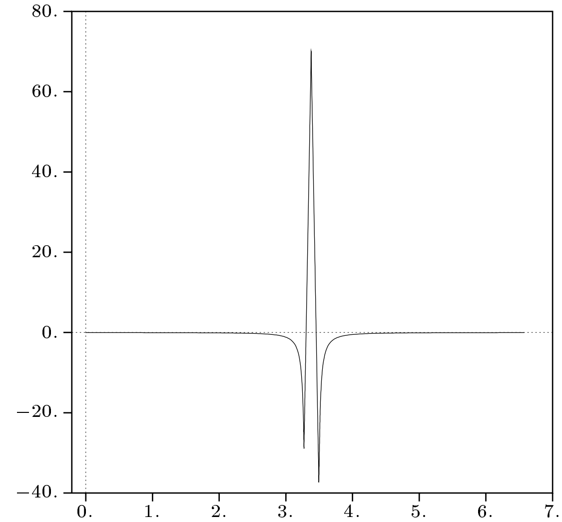
$u_2(t)$ [Nm]



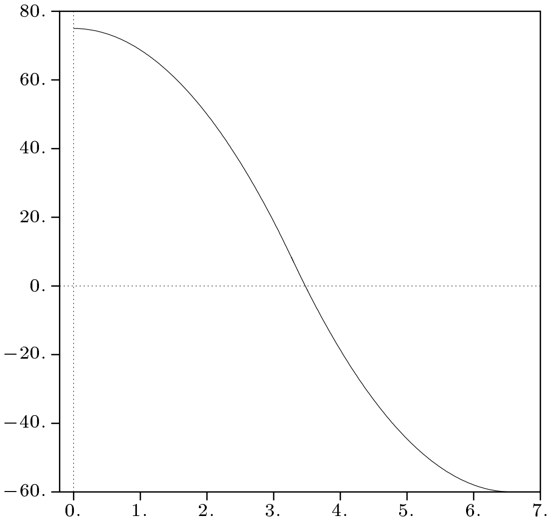
$\dot{r}(t)$ [cm/s]



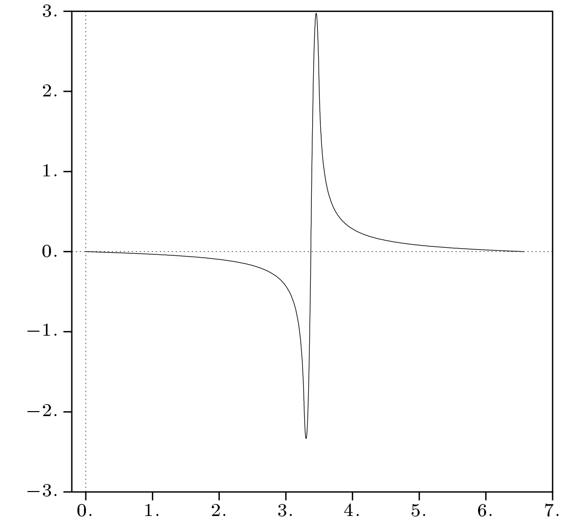
$\dot{\theta}(t)$ [°/s]



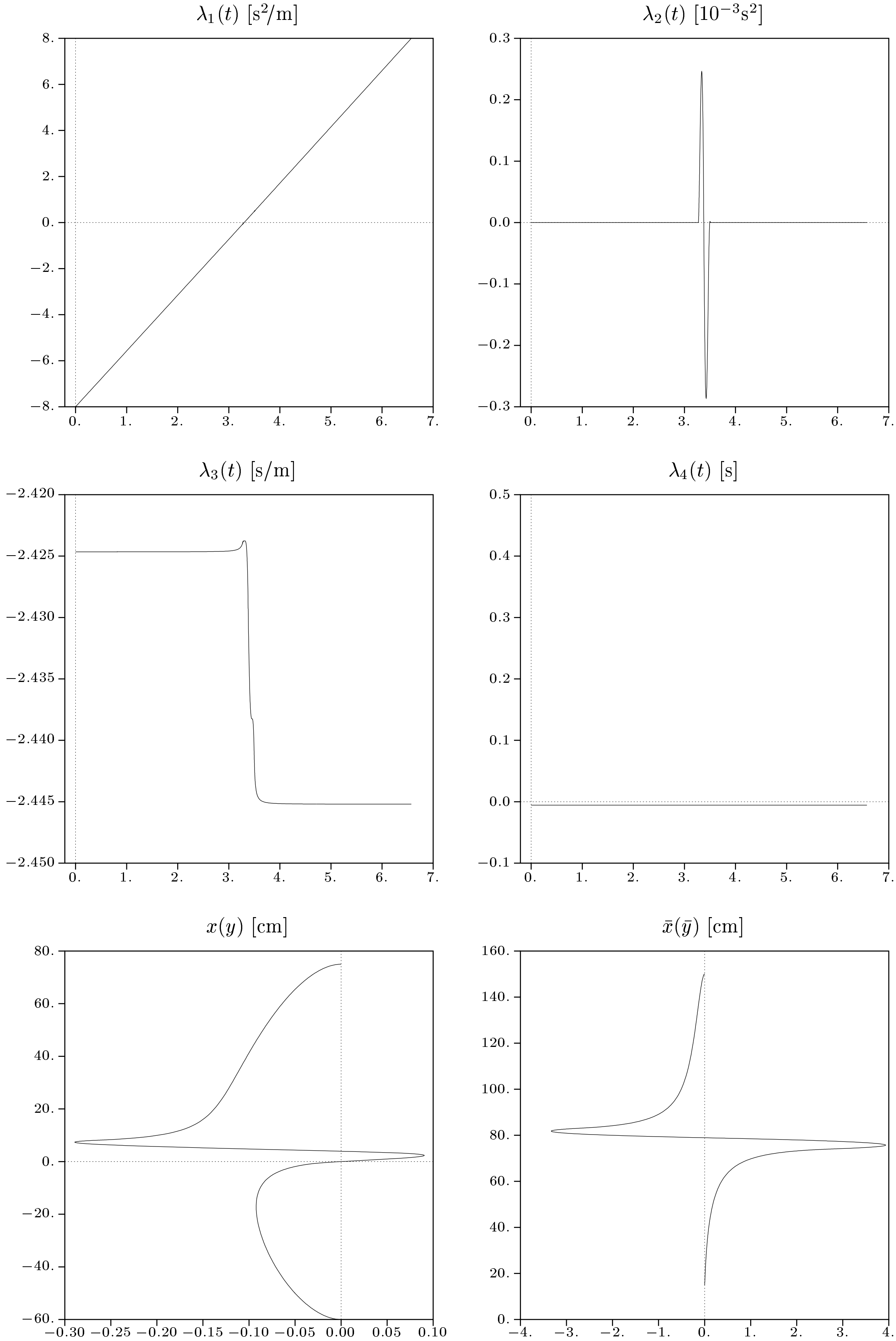
$r(t)$ [cm]



$\theta(t)$ [°]



$r_e = -60\,\mathrm{cm}$: adjungierte Variablen, Bahnen des Schwerpunkts und der Last



5. Zusammenfassung und Ausblick

Die Ergebnisse dieser Arbeit zeigen, daß schon bei sehr einfachen Bewegungen von Robotertypen mit nur wenigen Freiheitsgraden die optimalen Steuerungen äußerst kompliziert sein können. Obwohl sich für eine große Klasse von Robotertypen im allgemeinen Rahmen der Mehrkörpermechanik auf relativ einfache Weise ein mathematisches Modell entwickeln läßt, ist für jeden einzelnen Typ eine genaue Analyse des jeweiligen Modells erforderlich. Diese Analyse muß von vornherein das gewählte Zielfunktional berücksichtigen, und kann insbesondere beim Auftreten singulärer Steuerungen sehr umfangreich sein.

Die Schaltstruktur der optimalen Lösung hängt nicht nur von der *Art* der Randbedingungen eines Steuerungsproblems ab, sondern auch von den konkreten Zahlenwerten. Weiter läßt sich die optimale Schaltstruktur eines gegebenen Problems i.a. nicht vollständig auf analytischem Wege bestimmen, so daß in der Praxis fast immer umfangreiche numerische Untersuchungen erforderlich sind.

Nicht zuletzt kommt die Beschaffung geeigneter Startdaten für den indirekten Ansatz einer näherungsweisen Lösung des Problems gleich. So ist z.B. die in der vorliegenden Arbeit verwendete suboptimale Lösung für $r_e = 25\,\mathrm{cm}$ nur um $1.5\,\mathrm{ms}$ oder etwa $0.5\,\%$ langsamer als die optimale.

Die Berechnung einer optimalen Lösung mit dem indirekten Ansatz ist also mit einem in jeder Hinsicht immensen Aufwand verbunden.

All diese Schwierigkeiten lassen den praktischen Nutzen optimaler Steuerungen auf den ersten Blick zweifelhaft erscheinen, zumal die mathematischen Lösungen aufgrund technischer Beschränkungen für einen existierenden Manipulator nur selten in vollem Umfang realisierbar sein werden.

Dennoch gibt es Bereiche, in denen sich auch ein so hoher Planungs- und Entwicklungsaufwand lohnen wird, ganz abgesehen von den theoretischen Erkenntnissen, die die Untersuchung optimaler Steuerungen liefert.

So wurde in dieser Arbeit qualitativ und quantitativ gezeigt, wie ein Robotermotor den anderen durch geschicktes Ausnutzen der physikalischen Wechselwirkungen unterstützen kann. Obwohl im untersuchten Beispiel der grobe Bewegungsablauf schon aus physikalischen Überlegungen bekannt und mit dem direkten Ansatz berechnet worden war, ergaben sich doch überraschende neue Erkenntnisse, was die „Feinstruktur" betrifft. Insbesondere die Abhängigkeit der Schaltstruktur von den gegebenen Randwerten sowie die recht häufig auftretenden Übergangs- bzw. Grenzstrukturen sind hochinteressant und sehr aufschlußreich.

In der praktischen Anwendung sollte man auf jeden Fall Gebrauch von dem genannten allgemeinen Optimierungsprinzip machen, also eine möglichst gute Zusammenarbeit der einzelnen Robotermotoren anstreben. Vor allem bei weiträumigen dreidimensionalen Bewegungen sind dadurch beträchtliche Einsparungen zu erwarten. Wie weit man die Optimierung treibt, hängt natürlich von den gegebenen Möglichkeiten ab und ist nicht zuletzt eine Kostenfrage. Eine gute suboptimale Lösung sollte in den meisten Fällen die praktischen Anforderungen erfüllen.

Die Steuerungsoptimierung mit dem indirekten Ansatz ist aus den genannten Gründen sehr aufwendig und numerisch nur mit den besten Codes zu bewerkstelligen. Sie wird daher wohl vor allem in besonderen Anwendungsgebieten zum Einsatz kommen, wie z.B. in der Raumfahrttechnik (Zusammenbau einer Raumstation) oder etwa im medizinischen Bereich (Steuerung von Arm- und Beinprothesen).

Sinnvoll erscheint aber auch ein Einsatz im großtechnischen Maßstab, wo häufig schon kleine bis mittlere Verbesserungen entscheidend sind. Man denke beispielsweise an eine komplette Fließbandstraße mit Robotern. Hier kann man zunächst für jeden Arbeitsgang eine *zeitoptimale* Steuerung berechnen. Sodann wird der Zeittakt so festgelegt, daß der langsamste Roboter voll ausgelastet ist. Die übrigen Roboter können jetzt *energieoptimal* gesteuert werden, so daß — bei maximalem Gesamtarbeitstempo — alle Geräte die zur Verfügung stehende Zeit möglichst gut ausnutzen. Selbst wenn dadurch die Produktionsrate nur um 3 bis 5 % gesteigert und der Energieverbrauch um 2 bis 3 % gesenkt werden können, ist dies ein respektables Ergebnis.

Die Kenntnis optimaler Lösungen liefert aber auch wertvolle Hinweise, wie die technischen Möglichkeiten am besten auszunutzen sind. Außerdem lassen sich mit ihnen systematisch Referenzbahnen erzeugen, um die Güte einer tatsächlich realisierten Lösung mit dem theoretischen Optimum zu vergleichen.
Dies läßt sich nutzbringend berücksichtigen beim Entwurf von Spezialrobotern, die nur wenige, schon im Planungsstadium festliegende Bewegungsabläufe ausführen sollen. In solchen Fällen kann z.B. die Dimensionierung des Roboters so vorgenommen werden, daß die vorgesehenen Bewegungen bei einer relativ einfachen, technisch realisierbaren Schaltstruktur möglichst schnell oder energiesparend ablaufen können.

Zusammenfassend ist also zu erwarten, daß die Resultate der mathematischen Steuerungstheorie schon in naher Zukunft beträchtlichen Einfluß auf die Entwicklung und den industriellen Einsatz von Robotern nehmen werden.

6. Literaturverzeichnis

[1] ARNOLD, V. I.:
Mathematical methods of classical mechanics. Springer Verlag, New York 1978.

[2] BOCK, H. G.:
Numerische Optimierung zustandsbeschränkter parameterabhängiger Prozesse mit linear auftretenden Steuerungen unter Anwendung der Mehrzielmethode. Diplomarbeit, Köln 1974.

[3] BOCK, H. G.:
Zur numerischen Behandlung zustandsbeschränkter Steuerungsprobleme mit Mehrzielmethode und Homotopieverfahren. ZAMM **57**, 1977.

[4] BOCK, H. G.:
Numerical Solution of Nonlinear Multipoint Boundary Value Problems with Applications to Optimal Control. ZAMM **58**, 1978.

[5] BOCK, H. G.:
Numerische Behandlung von zustandsbeschränkten und Chebychef-Steuerungsproblemen. Carl-Cranz-Gesellschaft, 1981.

[6] BOCK, H. G.:
Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen. Dissertation, Bonner mathematische Schriften **183**, Bonn 1987.

[7] BOCK, H. G., R. W. LONGMAN:
Optimal control of velocity profiles for minimization of energy consumption in the New York subway system. Proceedings of the 2nd IFAC-Workshop on control applications of nonlinear programming and optimization, Oberpfaffenhofen 1980.

[8] BOCK, H. G., R. W. LONGMAN:
Computation of optimal controls on disjoint control sets for minimum energy subway operation. Proceedings of the American Astronomical Society. Symposium on Engineering Science and Mechanics, Taiwan 1982.

[9] BOCK, H. G., K. J. PLITT:
A multiple shooting algorithm for direct solution of optimal control processes. Proceedings of the 9th IFAC-World Congress, Budapest 1984.

[10] BRYSON, A. E., W. F. DENHAM, S. E. DREYFUS:
Optimal programming problems with inequality constraints I. AIAA Journal **1**, 1963.

[11] BULIRSCH, R.:
Die Mehrzielmethode zur numerischen Lösung von nichtlinearen Randwertproblemen und Aufgaben der optimalen Steuerung. Carl-Cranz-Gesellschaft, Technischer Report 1971.

[12] BULIRSCH, R., J. STOER:
Einführung in die numerische Mathematik II. Heidelberger Taschenbücher **114**, 1972.

[13] FLEMING, W., R. RISHEL:
Deterministic and stochastic optimal control. Springer Verlag, Berlin 1975.

[14] FREHSE, J.:
Existence of Optimal Controls I. Methods of Operations Research, Bonn 1978.

[15] FREHSE, J.:
Ein Existenzsatz für optimale Steuerungen. SFB 72, Preprint No. **199**, Bonn 1979.

[16] HUSSELS, H. G.:
Schrittweitensteuerung bei der Integration gewöhnlicher Differentialgleichungen mit Extrapolationsverfahren. Diplomarbeit, Köln 1973.

[17] KELLEY, H. J., R. E. KOPP, H. G. MOYER:
Singular extremals. In "Topics in optimization" (G. Leitmann, editor), Chapter 3. Academic Press, New York 1967.

[18] KIRIAZOV, P. K., P. A. MARINOV:
Control Synthesis of Manipulator Dynamics in Handling Operations. Theoretical and Applied Mechanics, Publishing House of the Bulgarian Academy of Sciences, Year XIV, No. **2**, Sofia 1983.

[19] KIRIAZOV, P. K., P. A. MARINOV:
A Method for Time-Optimal Control of Dynamically Constrained Manipulators. International Symposium on Theory and Practice of Robots & Manipulators, Udine (Italy) 1984.

[20] KLINGENBERG, W.:
Riemannian geometry. De Gruyter studies in mathematics; **1**, Berlin 1982.

[21] KONZELMANN, J.:
Berechnung zeitoptimaler Steuerungen von Industrierobotern unter Verwendung eines direkten Mehrzielansatzes. Diplomarbeit, Bonn 1988.

[22] KRÄMER-EIS, P.:
Numerische Berechnung optimaler Feedback-Steuerungen bei nichtlinearen Prozessen. Diplomarbeit, Bonn 1980.

[23] KRÄMER-EIS, P.:
Ein Mehrzielverfahren zur numerischen Berechnung optimaler Feedback-Steuerungen bei beschränkten nichtlinearen Problemen. Dissertation, Bonner mathematische Schriften **166**, Bonn 1985.

[24] KRÄMER-EIS, P., H. G. BOCK:
An efficient algorithm for approximate computation of feedback control laws in nonlinear processes. ZAMM **61**, 1981.

[25] KRÄMER-EIS, P., H. G. BOCK:
A multiple shooting method for numerical computation of open and closed loop controls in nonlinear systems. Proceedings of the 9th IFAC-World Congress, Budapest 1984.

[26] MARINOV, P. A., P. K. KIRIAZOV:
Synthesis of Time-Optimal Control for Manipulator Dynamics. Theoretical and Applied Mechanics, Publishing House of the Bulgarian Academy of Sciences, Year XV, No. **1**, Sofia 1984.

[27] MAURER, H.:
Zur numerischen Berechnung optimaler singulärer Steuerungen. Dissertation, Köln 1972.

[28] MAURER, H.:
Optimale Steuerprozesse mit Zustandsbeschränkungen. Habilitationsschrift, Würzburg 1976.

[29] PONTRJAGIN, L. S., V. G. BOLTJANSKIJ, R. V. GAMKRELIDZE, E. F. MISCENKO:
Mathematische Theorie optimaler Prozesse. Verlag R. Oldenburg, München 1964.

[30] SEDGWICK, A.:
An effective variable order variable step Adams method. University of Toronto, Deptartment of Computer Science: Technical Report No. **53**, 1973.

7. Anmerkungen zur Neufassung

Die Originalarbeit war in einer alten $\mathcal{AMS}$-TEX-Version gesetzt, mit Graphiken in einem nicht standardisierten Dateiformat. Da beides nicht mehr unterstützt wird, habe ich diese Neufassung angefertigt. Hier sind die Abweichungen von der Originalversion beschrieben.

Formale Änderungen

Die Neufassung ist gesetzt in LaTeX 2_ε mit den Makropaketen `amsmath`, `babel`, `german` und `graphicx`, wodurch geringfügige Änderungen von Layout, Zeichensätzen sowie Zeilen- und Seitenumbrüchen entstehen. Die deutsche Silbentrennung und Anführungszeichen („ “ statt “ ”) sind ebenfalls verbessert.

Die Strukturplots auf den Seiten 52–54 sowie die Trajektorienplots auf den Seiten 46 und 62–105 wurden mit *gnuplot* neu angefertigt, die Abbildungen auf den Seiten 10, 13, 17, 24 und 25 mit *IRIS Showcase*. Die beiden Abbildungen auf Seite 44 sind in der `picture`-Umgebung von LaTeX gesetzt.

Das Erscheinungsbild der Originalarbeit wurde weitgehend beibehalten.

Fehlerkorrekturen

Neben einigen Tippfehlern wurden folgende inhaltlichen Ungenauigkeiten und Fehler korrigiert:

Seite	Position	Korrektur
18	Seitenmitte	„Ort $\xi_i(t)$“ statt „$\xi(t)$“
22	nach (2.2.10)	„Summand $\alpha_{3e}z_{3e}$“ statt „$\alpha_3 z_{3e}$“
24	Satz (2.3.6)	„Eine zeitoptimale“ statt „optimale“
	in (2.3.8)	$\epsilon = 0$ eingesetzt, Normquadrat entfällt
36	nach (2.6.13)	„eine Nullmenge“ statt „diskret in I_S“
41	in (3.2.14)	Addition statt Subtraktion in der großen Klammer
55	Tab. (3.4.10)	„$u_2^{\text{sing}}(t_3) = -\hat{u}_2$“ statt „$= 0$“.
60	`PRHOM196`	„`U2SING(T3)=-U2MAX`“ statt „`=0`“ (Kommentarzeile)
61	`PRHOM252`	„`+D(2)`“ am Zeilenende angefügt
106	vorletzter Absatz	„Die Steuerungsoptimierung mit dem indirekten“ statt „mit dem direkten“

Die ursprüngliche Formulierung von Satz (2.3.6) auf S. 24 war zwar korrekt, aber verwirrend: wie zuvor ausdrücklich bemerkt, geht es um den rein zeitoptimalen Fall ($\epsilon = 0$). Ebenso ist „diskret“ auf S. 36 richtig, was aber erst später bewiesen wird. Die identischen Fehler auf S. 55, 60 und 61 sind wohl beim Zusammenfassen aller Schaltstrukturen in einem Programm entstanden: wie der Verlauf von u_2 auf S. 98 zeigt, wurden die numerischen Rechnungen tatsächlich mit der korrekten Schaltbedingung $u_2^{\text{sing}}(t_3) = -\hat{u}_2$ durchgeführt, die auch auf S. 53 so angegeben ist.

Elektronische Version

Diese Neufassung der Arbeit ist als PostScript-Datei verfügbar unter

`http://www.zib.de/steinbach/papers/dipl.ps.gz`

Eventuelle zukünftige Änderungen meiner homepage-Adresse sollten in der *Na-Net Whitepages Database* der *netlib* (`http://www.netlib.org/`) zu finden sein.