

Technische Universität Dresden

Fakultät Elektrotechnik und Informationstechnik

Institut für Regelungs- und Steuerungstheorie

Studienarbeit

Modellbildung und Reglerentwurf für ein Brückenkransystem

vorgelegt von: Konstantin Wrede
geboren am: 19. Januar 1998 in Sondershausen

Betreuer:	M.Sc. Tommy Wohlfahrt Dipl.-Ing. Fabian Paschke Dr.-Ing. Carsten Knoll
Verantwortlicher Hochschullehrer:	Prof. Dr.-Ing. habil. Dipl.-Math. K. Röbenack
Tag der Einreichung:	XX. Oktober 2022

Bitte ersetzen Sie diese Seite vor dem Binden mit der Aufgabenstellung.

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die von mir am heutigen Tage an der Fakultät Elektrotechnik und Informationstechnik eingereichte Studienarbeit zum Thema

Modellbildung und Reglerentwurf für ein Brückenkransystem

selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Schriften entnommen sind, wurden als solche kenntlich gemacht.

Dresden, XX. Oktober 2022

Konstantin Wrede

Kurzfassung

Am Fraunhofer-Institut für Integrierte Schaltungen IIS wurde ein Versuchsstand entwickelt, der einen miniaturisierten Brückenkran mit zwei unabhängigen Steuerungseinheiten nachbildet. An diesem kann die Leistungsfähigkeit verschiedener Regelungsansätze in Hinblick auf das Maß ihrer Verteiltheit und Kommunikationsmöglichkeiten der Steuerungen untersucht werden. Für Vergleiche wird eine zentrale Referenzregelstrategie benötigt.

Im Rahmen dieser Arbeit wurde zu diesem Zweck ein analytisches Modell des Brückenkrans über den Lagrange-Formalismus hergeleitet. Für das ermittelte nichtlineare Mehrgrößensystem ist die differenzielle Flachheit konstruktiv nachgewiesen worden. Auf Basis des dabei bestimmten flachen Ausgangs sind für eine Referenzregelstrategie Ausgangstrajektorien zur Überführung zwischen Ruhelagen geplant worden. Außerdem ist darüber eine Parametrisierung der Eingangsgrößen des Systems für eine Vorsteuerung möglich. Verschiedene Folgeregelungen wurden für Anfangsfehler in den Zustandskomponenten simulativ geprüft.

Abstract

At the Fraunhofer Institute for Integrated Circuits IIS, a test bed has been developed that represents a miniaturized overhead crane with two independent control units. This system can be used to investigate the performance of different control approaches with regard to their degree of distribution and the communication possibilities of the control units. For comparison, a central reference control strategy has to be designed.

Within the scope of this work, an analytical model of the crane was derived using the Lagrangian formalism. For the determined nonlinear multivariable system, the differential flatness has been constructively proved. Based on the flat output determined in this process, output trajectories for the transfer between equilibriums were planned leading to a reference control strategy. In addition, a parameterization of the system's

input variables for a feedforward control is possible using this flat output. Various trajectory tracking control strategies were examined within simulations with assumed initial errors in the state components.

Inhaltsverzeichnis

Verzeichnis der wichtigsten Abkürzungen und Symbole	VIII
Abbildungsverzeichnis	X
1 Einleitung	1
2 System- und Problembeschreibung	2
2.1 Aufbau des Demonstratorsystems	2
2.2 Problembeschreibung und Zielsetzung	4
3 Analytische Modellbildung	5
3.1 Allgemeine Modellannahmen	5
3.2 Modellierung mittels Lagrange-Formalismus	6
3.2.1 Lagrange-Gleichungen erster Art	6
3.2.2 Lagrange-Gleichungen zweiter Art	7
3.3 Generierung und Simulation von DAE-Systemen	7
3.4 Analytisches Modell des Einzelkrans	9
3.5 Analytisches Modell des Doppelkrans	11
3.5.1 Ansatz über ein DAE-System	11
3.5.2 Ansatz über ein ODE-System	14
3.6 Identifikation der Systemparameter	17
4 Flachheitsanalyse	18
4.1 Definition differenzieller Flachheit	18
4.2 Flachheitsanalyse von Mehrgrößensystemen	19
4.3 Anwendung Flachheitsanalyse am Einzelkran	20
4.4 Anwendung Flachheitsanalyse am Doppelkran	22
5 Steuerungs- und Regelungsentwurf	25
5.1 Trajektorienplanung für den flachen Ausgang	25
5.2 Trajektorienfolge­regelung	26
5.2.1 Vektorieller relativer Grad	26
5.2.2 Statische Rückführung	26
5.2.3 Zustandsrückführung aus Fehlerdynamik	28
5.2.4 Dynamische Erweiterung	28
5.2.5 Quasi-statische Rückführungen	33

5.2.6	Exact feedforward linearization	36
6	Fazit und Ausblick	42
6.1	Fazit	42
6.2	Ausblick	43
	Literatur	44

Verzeichnis der wichtigsten Abkürzungen und Symbole

Die folgenden Tabellen enthalten Abkürzungen sowie Symbole, welche in dieser Arbeit eine herausragende Bedeutung besitzen oder kapitelübergreifend verwendet werden.

Abkürzung	Bedeutung
AGL	algebraische Gleichungen
DAE	differenzial-algebraische Gleichungen, von engl. differential algebraic equations
DGL, ODE	(gewöhnliche) Differenzialgleichung, von engl. ordinary differential equation
LG	Lagrange-Gleichungen
ZB	Zwangsbedingung

Vektor- oder matrixwertige Größen werden durch **fett gedruckte Symbole** dargestellt.

Symbol	Bedeutung
$\boldsymbol{\theta} = (\mathbf{q}, \mathbf{p})^T$	Konfigurationskoordinaten
λ	Lagrange-Multiplikator
Λ	Entkopplungsmatrix
$\boldsymbol{\tau}, \mathbf{u}$	Systemeingang
\mathbf{a}	holonome Zwangsbedingungen
\mathbf{e}	Trajektorienfolgefehler
$\mathbf{f}, \mathbf{g}, \mathbf{h}$	Felder in eingangsaffiner Zustandsraumdarstellung
i_d	differenzieller Index
\mathbf{J}	Jacobi-Matrix
\mathcal{L}	Lagrange-Funktion
L_f	Lie-Ableitung entlang Vektorfeld \mathbf{f}
\mathbf{p}	nicht direkt aktuierte Koordinaten
\mathbf{q}	direkt aktuierte Koordinaten
\mathbf{Q}	generalisierte (verallgemeinerte) Kraft
\mathbf{r}	vektorieller relativer Grad
T	kinetische Energie
\mathbf{v}	virtueller Systemeingang
V	potentielle Energie
\mathbf{x}	Systemzustand
\mathbf{y}	flacher Ausgang

Die Symbolik func stellt einen allgemeinen und nicht weiter spezifizierten funktionalen Zusammenhang dar, welcher sich bei jeder Erwähnung unterscheiden kann.

Abbildungsverzeichnis

1	Konstruktionsmodell des Doppelkransystems	2
2	Doppelkran-Demonstratorsystem mit Laufkatzen und Last	3
3	Planare Geometrie des Einzelkransystems	9
4	Planare Geometrie des Doppellkransystems im DAE-Modell	12
5	Planare Geometrie des Doppellkransystems im ODE-Modell	15
6	Trajektorien Ruhelagenüberführung mit Regelung über dynamische Erweiterung	31
7	Zwiebelschalendiagramm Ruhelagenüberführung mit Regelung über dynamische Erweiterung	32
8	Trajektorien Ruhelagenüberführung mit Regelung über exact feedforward linearization (Pseudoinverse)	38
9	Eigenwerte der Jacobimatrix bei Trajektorienfolge mit Regelung über exact feedforward linearization (Selektionsmatrix)	40
10	Trajektorien Ruhelagenüberführung mit Regelung über exact feedforward linearization (Selektionsmatrix)	41

Kapitel 1

Einleitung

In einer Vielzahl heutiger industrieller Anwendungen wird eine zentrale Organisation zur Steuerung von Geräten im Kontext des Internets der Dinge¹ und deren Kollaboration eingesetzt. Das agile Zusammenarbeiten einzelner Teilsysteme verspricht eine besonders hohe Flexibilität und Anpassungsfähigkeit von Produktionsprozessen. Allerdings sind die IoT-Geräte aufgrund des zentralen Ansatzes auch in hohem Maße von der Verfügbarkeit und Verlässlichkeit ihrer Verbindung zur Steuerung abhängig. Gerade bei mobilen Geräten stellt die drahtlose Kommunikation auf den letzten Metern eine enorme Herausforderung dar.

Im Rahmen vorangegangener studentischer Arbeiten ist am Fraunhofer-Institut für Integrierte Schaltungen IIS, Institutsteil Entwicklung Adaptiver Systeme EAS [7] in Dresden ein Demonstratorsystem entwickelt worden. Mit diesem System sollen perspektivisch Regelungsstrategien auf verteilten Recheneinheiten gegenüber einer zentralen Messgrößenverarbeitung und Stellgrößenberechnung verglichen werden. Außerdem ist es möglich, damit den Einfluss von Störungen und Verzögerungen der (drahtlosen) Kommunikation zwischen den Recheneinheiten zu evaluieren.

Aufgabe dieser Studienarbeit war es, eine analytische Modellbildung des Versuchsstands durchzuführen sowie eine zentrale Referenzregelstrategie ohne Verteilung auf verschiedene Recheneinheiten für das Demonstratorsystem zu entwerfen. Zudem waren die dabei ermittelten Ansätze zur Steuerung und Folgeregelung simulativ zu verifizieren.

Da es sich beim Demonstrator um ein mechanisches Starrkörpersystem handelt, bietet sich eine kinetische Beschreibung des Systems mittels des Lagrange-Formalismus an. Die Systemgleichungen eines solchen nichtlinearen Mehrgrößensystems können auf die Eigenschaft der differentiellen Flachheit untersucht werden. Daraus kann bei erfolgreichem Nachweis eine flachheitsbasierte Trajektorienplanung der Ausgangsgrößen und Parametrisierung der Stellgrößen dadurch erfolgen. Die Korrektur von Abweichungen von den Referenzverläufen ist durch eine ebenfalls flachheitsbasierte Folgeregelung möglich.

¹„Mit dem „Internet der Dinge“ (engl.: Internet of things, IoT) oder „Allesnetz“ wird die Vernetzung von Gegenständen mit dem Internet verstanden, so dass diese Gegenstände über das Internet selbstständig kommunizieren und einen Material- oder Produktionsfluss festlegen.“ [2]

Kapitel 2

System- und Problembeschreibung

2.1 Aufbau des Demonstratorsystems

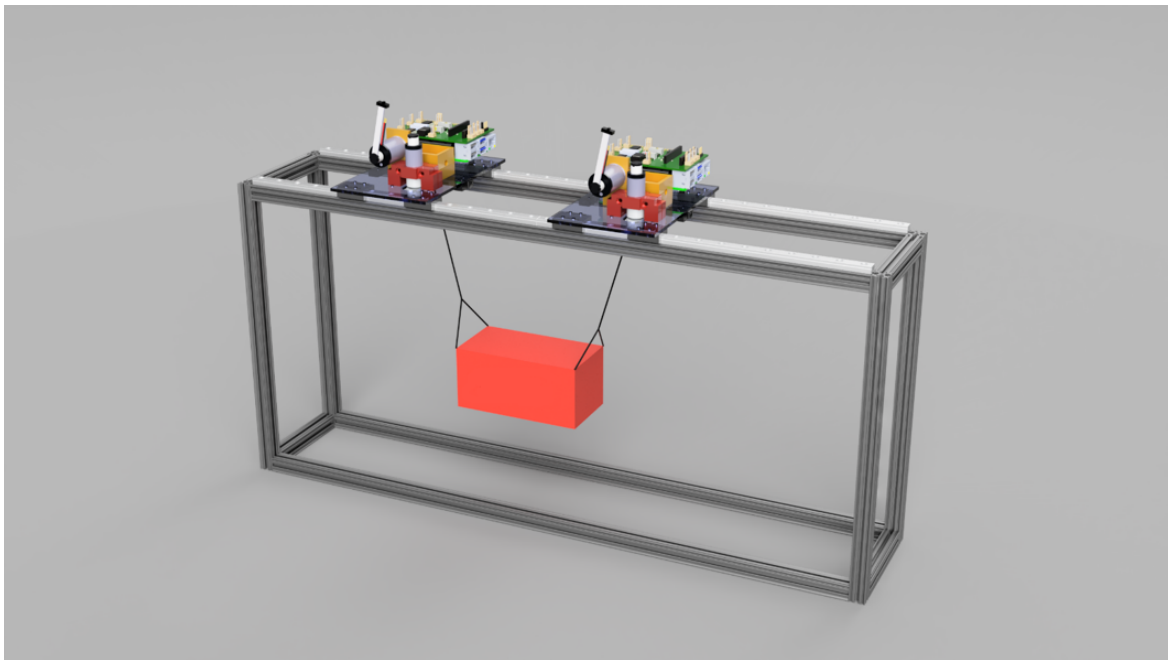


Abbildung 1 – Konstruktionsmodell des Doppelkran-Demonstratorsystems ohne Verkabelung.

Das Demonstratorsystem besteht aus zwei Brückenkränen, die eine gemeinsame Last in der vertikalen Ebene anheben. Abbildung 1 stellt ein Modell des Aufbaus dar, das während des Konstruktionsprozesses entstanden ist. Die Kräne befinden sich auf zwei horizontalen Führungsschienen und verfügen jeweils ¹ über einen Raspberry Pi 4B² als

¹Diese Anordnung erlaubt eine Architektur verteilter Systeme. Im Rahmen dieser Studienarbeit wird allerdings nur ein zentraler Regelungsansatz, also die effektive Nutzung einer Recheneinheit, verfolgt.

²Spezifikationen: 8GB DDR4 RAM, 64-bit ARM64 SoC, 5.0 GHz WLAN, Bluetooth 5.0, BLE, Gigabit

Hauptrecheneinheit sowie einen STM32-Mikrocontroller³ für die Motoransteuerungen und Messungen. Beide Raspberry Pis können über eine LAN-Verbindung miteinander kommunizieren. Die Kräne sind als Laufkatzen entlang der Schienen sowie Seilwinden darauf mit jeweils einem Gleichstrommotor aktuiert. Auf den STM32-Mikrocontrollern ist bereits eine unterlagerte Strom- beziehungsweise Kraftregelung für diese Motoren implementiert. Messungen der Seillängen und Kranpositionen auf der Schiene erfolgen mittels Inkrementalgebern nach einem anfänglichen Kalibrierungsvorgang. Die Seilwinkel zur Horizontalen werden mittels mitschwingender Potentiometer bestimmt. Abbildung 2 zeigt eine Fotografie der realen Anordnung des Demonstrators.

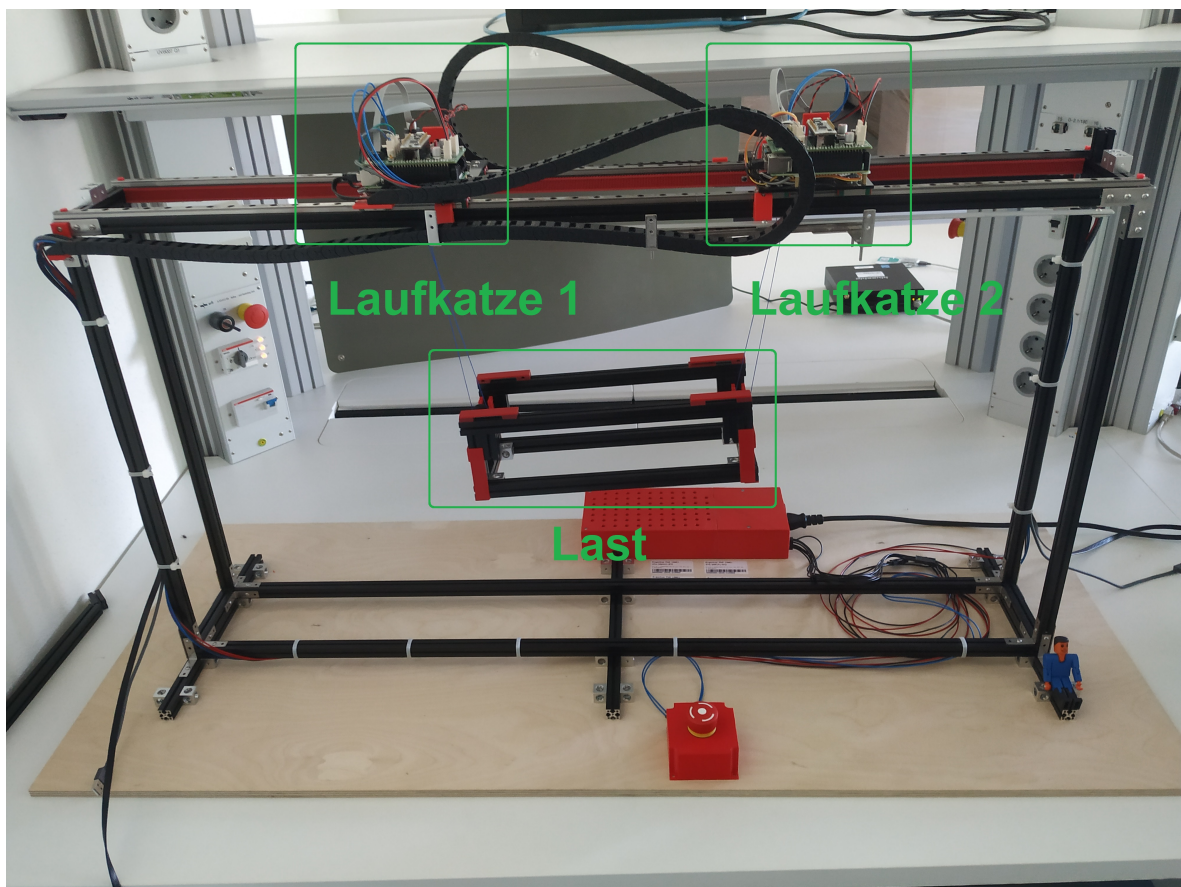


Abbildung 2 – Anordnung des Doppelkran-Demonstratorsystems mit Laufkatzen und Last am Versuchsstand.

Ethernet; Betriebssystem: Ubuntu 20.04 [4]

³Spezifikationen: STM32F303K8 mit 32-bit Arm Cortex-M4, 64 kB Flash-Speicher, 12-bit ADCs [13]

2.2 Problembeschreibung und Zielsetzung

Bei der Bewegung von Containern in Häfen ist ein ruckarmer und gegenüber der Horizontalen stabiler Transport notwendig. Ziel dieser Studienarbeit ist es deshalb, für das vorhandene Demonstratorsystems eine zentrale Referenzregelstrategie zu entwerfen. Damit soll unter Vorgabe von Sollposen eine Planung von Trajektorien der Last in ihrer vertikalen Aufhängungsebene und eine Folgeregelung zur Überführung der Last zwischen verschiedenen Ruhelagen ermöglicht werden. Diese Überlegungen sollen auf Basis einer Modellierung des Krans als Mehrkörpersystem geschehen.

Perspektivisch soll diese zentrale Referenzregelstrategie einerseits zum Vergleich mit zukünftig zu entwickelnden verteilten Regelungsansätzen genutzt werden sowie andererseits als Grundlage für den Einsatz von Methoden des maschinellen Lernens dienen. Dabei ist es von Interesse zu untersuchen, ob etwa durch bestärkendes Lernen unter der Vorgabe von flachheitsbasierten Referenztrajektorien im Lernprozess ein künstliches neuronales Netz generiert werden kann, welches eine vergleichbar gute Regelung im Normalfall bietet, allerdings Vorteile in beispielsweise Robustheit und Adaptivität aufweist.

Kapitel 3

Analytische Modellbildung

3.1 Allgemeine Modellannahmen

In der folgenden Modellierung wird das Demonstratorsystem nur planar betrachtet, also die Bewegung aller Komponenten nur in der vertikalen Ebene berücksichtigt. Diese Annahme ist gerechtfertigt, weil die Laufkatzen nur entlang einer Achse verfahren können und die Lagerung bzw. Aufhängung der Last ein Schwingen senkrecht zur vertikalen Ebene unterdrückt. Die Seile werden aufgrund ihrer geringen Dicke als masselos im Vergleich zu den Laufkatzen sowie der Last angenommen und somit auch nicht mit einem Trägheitsmoment versehen. Die Last wird trotz ihrer Aussparungen (siehe Abbildung 2) mit einer homogenen Masseverteilung modelliert. Für eine genauere Beschreibung wären Konstruktionsdaten oder eine Demontage mit Vermessung der Teilkomponenten notwendig. Zur Reduktion weiterer Komplexität werden dissipative Kräfte im Modell nicht berücksichtigt.

Die Modellbildung erfolgt in mehreren Stufen. Zunächst wird mittels der Lagrange-Gleichungen erster Art ein Einzelkran modelliert. Das Modell liegt dann in Form eines gewöhnlichen Differenzialgleichungssystems (DGL-System, engl. ODE system) vor. Zusätzlich wird ein Ansatz mit Lagrange-Gleichungen zweiter Art gewählt. Diese Methode gibt die geschlossene kinematische Kette der durch die Last verbundenen Laufkatzen intuitiv wider, so dass ein differenzial-algebraisches Gleichungssystem (DAE-System, von engl. differential algebraic equations) entsteht. In der Arbeit hat sich allerdings gezeigt, dass die Lagrange-Gleichungen zweiter Art eine kompliziertere Formulierung der Hubaktuierung sowie aufwändigere Lösungsalgorithmen bedeuten. Um das Prinzipielle Vorgehen zur Herleitung eines DAE-Modells dennoch zu schildern, wird es auf den Einzelkran angewendet. Ausgehend vom Einzelkransystem kann die prinzipielle Methodik der Modellbildung verifiziert werden und ein Transfer der Erkenntnisse auf die Modellierung des Doppelkransystems erfolgen.

3.2 Modellierung mittels Lagrange-Formalismus

Die Dynamik mechanischer Systeme lässt sich über Differenzialgleichungen, den sogenannten Lagrange-Gleichungen (LG) beschreiben. Dabei wird eine Menge aus n zeitlich veränderlichen Koordinaten als Konfigurationskoordinaten oder Systemgrößen $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n)^T$ bezeichnet. Deren zeitliche Änderungsraten werden im Vektor der Konfigurationsgeschwindigkeiten $\dot{\boldsymbol{\theta}}$ zusammengefasst. In [10, S.10] wird eine Unterteilung der Konfigurationskoordinaten $\boldsymbol{\theta} = (\mathbf{q}, \mathbf{p})^T$ in “aktive” Koordinaten \mathbf{q} und “passive” Koordinaten \mathbf{p} vorgenommen. In Anlehnung daran wird in dieser Arbeit eine Aufteilung der Systemgrößen in die direkt aktuierten Koordinaten \mathbf{q} und nicht direkt aktuierten Koordinaten \mathbf{p} durchgeführt. [10, S.7]

Die kinetische Energie eines Systems wird im Folgenden durch die Funktion T sowie die potentielle Energie durch V beschrieben. Die Lagrange-Funktion kann damit folgendermaßen definiert werden:

$$\mathcal{L}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = T(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - V(\boldsymbol{\theta}). \quad (3.1)$$

Eine automatisierte Generierung von Systemgleichungen aus den Energiefunktionen ist unter Nutzung des Python-Pakets symbtools [14] möglich.

3.2.1 Lagrange-Gleichungen erster Art

Mit den LG erster Art können Problemstellungen mit Zwangsbedingungen (ZB) und -kräften abgebildet werden:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i} \right) - \frac{\partial \mathcal{L}}{\partial \theta_i} = \tilde{Q}_i + Q_i, \quad i = 1, \dots, n. \quad (3.2)$$

Die sich auf die jeweilige Koordinate θ_i beziehende Stellkraft $Q_i = f_i - D_i$ entspricht der verallgemeinerten Kraft, welche sich aus der äußeren (Stell-)Kraft f_i sowie internen Reibungskraft D_i zusammensetzt [3, S. 49].

Weiterhin können m holonome ZB $a_1(\boldsymbol{\theta}) = \dots = a_m(\boldsymbol{\theta}) = 0$ eingeführt werden, aus denen die Zwangskraft \tilde{Q}_i in Richtung der Koordinate θ_i folgt:

$$\tilde{Q}_i = \sum_{j=1}^m \lambda_j \frac{\partial a_j}{\partial \theta_i}. \quad (3.3)$$

In dieser Beziehung bezeichnet λ_j den jeweiligen Lagrange-Multiplikator.

3.2.2 Lagrange-Gleichungen zweiter Art

Die LG zweiter Art beschreiben bezüglich der LG erster Art den Spezialfall ohne ZB, also $m = 0$. Zwangskräfte sind dabei nicht explizit zu bestimmen. Als Voraussetzung müssen die Konfigurationskoordinaten unabhängig voneinander sein, es darf also keine ZB geben, die diese verkoppeln. In diesem Zusammenhang spricht man auch von den Minimalkoordinaten $\boldsymbol{\theta}$. Dann können die Bewegungsgleichungen folgendermaßen aus der Lagrange-Funktion abgeleitet werden:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i} \right) - \frac{\partial \mathcal{L}}{\partial \theta_i} = Q_i, \quad i = 1, \dots, n. \quad (3.4)$$

Zur Bestimmung der Komponenten Q_i der verallgemeinerten Kraft wird das Prinzip der virtuellen Arbeit herangezogen [15]:

$$\delta W = \sum_{k=1}^l \mathbf{F}_k \cdot \frac{\partial \mathbf{r}_k}{\partial \theta_1} \delta \theta_1 + \dots + \sum_{k=1}^l \mathbf{F}_k \cdot \frac{\partial \mathbf{r}_k}{\partial \theta_n} \delta \theta_n. \quad (3.5)$$

Dabei entspricht \mathbf{r}_k bei einem System aus l (massebehafteten) Teilchen dem Richtungsvektor zum k -ten Partikel und \mathbf{F}_k der jeweils entlang dieses Richtungsvektors angewandten Stellkraft. Die virtuelle Verschiebung des Partikels im Raum $\delta \mathbf{r}_k$ und die virtuelle Verschiebung $\delta \theta_i$ der Koordinate θ_i genügen der Beziehung:

$$\delta \mathbf{r}_k = \sum_{i=1}^n \frac{\partial \mathbf{r}_k}{\partial \theta_i} \delta \theta_i. \quad (3.6)$$

Die gesamte virtuelle Arbeit des Systems dieser Teilchen kann also ebenso durch

$$\delta W = Q_1 \delta \theta_1 + \dots + Q_n \delta \theta_n = \sum_{k=1}^l \delta \mathbf{r}_k^T \mathbf{F}_k \quad (3.7)$$

dargestellt werden, wobei sich die Komponenten der verallgemeinerten Kraft zu

$$Q_i = \sum_{k=1}^l \left(\frac{\partial \mathbf{r}_k}{\partial \theta_i} \right)^T \mathbf{F}_k = \frac{\partial \delta W}{\partial \delta \theta_i}, \quad i = 1, \dots, n \quad (3.8)$$

ergeben.

3.3 Generierung und Simulation von DAE-Systemen

DAE-Systeme sind ODE-Systeme, welche um algebraische Gleichungen (AGL, auch als Nebenbedingungen bezeichnet) ergänzt werden. Diese Nebenbedingungen können

zur Darstellung von ZB der LG erster Art genutzt werden. DAE-Systeme lassen sich typischerweise in einer semi-expliziten Form darstellen

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{z}, \mathbf{u}, t) \quad (3.9a)$$

$$\mathbf{0} = \mathbf{a}(\mathbf{x}, \mathbf{z}, t), \quad (3.9b)$$

wobei \mathbf{x} dem Systemzustand, \mathbf{z} den algebraischen Variablen (weitere Systemgrößen, die in den Systemgleichungen ohne Ableitung vorkommen), \mathbf{u} dem Systemeingang sowie t der Zeit entspricht. [8, S.137]

Eine Möglichkeit zur Klassifikation von DAE-Systemen ist der differenzielle Index i_d . Dieser entspricht der minimalen Anzahl an Differenziationen $\frac{d}{dt}$ der AGL \mathbf{a} (ZB), damit unter Einbeziehung der DGL ein explizites DGL-System aus dem DAE-System entsteht. Ein gewöhnliches DGL-System besitzt also den differenziellen Index $i_d = 0$. Die Differenziation der AGL mit dem Resultat eines DAE-Systems mit kleinerem Index wird als Indexreduktion bezeichnet. [8, S.139]

Für die Simulation von DAE-Systemen ist die numerische Integration dieser Gleichungssysteme notwendig. Die in dieser Arbeit untersuchten mechanischen Systeme sind solche mit starrer Kopplung als ZB vom Index $i_d = 3$ und können für den Zustandsvektor $\mathbf{x} = (\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})^T$ über folgende Bewegungsgleichungen mittels der Matrizen \mathbf{M} , \mathbf{C} , \mathbf{K} , \mathbf{G} und \mathbf{B} beschrieben werden [8, S.240]:

$$\mathbf{0} = \mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{K}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{G}(\boldsymbol{\theta})\mathbf{z} - \mathbf{B}(\boldsymbol{\theta})\mathbf{u} \quad (3.10a)$$

$$\mathbf{0} = \mathbf{g}(\boldsymbol{\theta}). \quad (3.10b)$$

Zur Integration des Systems der Gleichungen (3.9) beziehungsweise (3.10) gibt es verschiedene Möglichkeiten [9, Kap. 8]:

- Indexreduktion auf Index $i_d = 2$ und anschließende Integration über ein implizites¹ Verfahren
- Indexreduktion auf Index $i_d = 1$ und anschließende Integration über ein explizites² Verfahren mit AGL-Löser oder ein implizites Verfahren
- Indexreduktion auf Index $i_d = 0$ und anschließende Integration über ein explizites oder implizites Verfahren.

Die im Python-Paket `sympytools` [14] enthaltene Bibliothek `modeltools` führt die Reduktion von Systemen mit Index $i_d = 3$ auf $i_d = 1$ durch. Zur numerischen Berechnung kann daraufhin der Solver `ODASSL` des Python-Pakets `assimulo` [1] verwendet werden.

¹Auf der linken und rechten Seite der zu integrierenden Gleichungen sind die gesuchten Näherungswerte von $\mathbf{x}(t_{i+1})$ an der Stelle t_{i+1} enthalten.

²Zur Berechnung des Näherungswertes von $\mathbf{x}(t_{i+1})$ an der Stelle t_{i+1} wird einzig der zuletzt berechnete Näherungswert von $\mathbf{x}(t_i)$ an der Stelle t_i benötigt.

Zur Erfüllung der AGL zu Simulationsbeginn müssen konsistente Anfangswerte $\mathbf{x}(0)$ und $\mathbf{z}(0)$ bestimmt werden. Bei DAE-Systemen mit Index $i_d \geq 2$ kann $\mathbf{x}(0)$ nicht mehr frei gewählt werden, da $\mathbf{z}(0)$ nicht mehr ausschließlich aus AGL bestimmbar ist. Daher ist es notwendig, zusätzliche algebraische Bedingungen an \mathbf{x} und \mathbf{z} aus den DGL abzuleiten. Die Bibliothek `modeltools` berechnet konsistente Anfangswerte mit der Funktion `calc_consistent_init_vals`. Falls sich andernfalls inkonsistente Anfangswerte ergeben, folgen daraus Simulationsfehler in den ersten Schritten oder sogar vollständig falsche Ergebnisse. [8, S.207]

3.4 Analytisches Modell des Einzelkrans

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [18, `flatness_notebooks/ODE_flatness_analysis_single_crane.ipynb`] zu finden.

Mittels der LG zweiter Art wird in diesem Abschnitt ein ODE-Modell des Einzelkrans erzeugt. Dabei wird entsprechend der Erläuterungen aus Abschnitt 3.2.2 vorgegangen. Diese Modellierung hat gegenüber den LG erster Art den Vorteil, dass durch die generalisierte Kraft \mathbf{Q} gerade die Stellkraft im Seil gut abgebildet werden kann sowie eine Vielzahl verschiedener Integrationsverfahren für eine effiziente Lösung des ODE-Systems nutzbar ist. Abbildung 3 zeigt die Konfiguration des Einzelkransystems.



Abbildung 3 – Planare Geometrie des Einzelkransystems mit Systemgrößen, Systemparametern und Kräften.

Als direkt aktuierte Koordinate wird nur die x -Verschiebung q_1 der Laufkatzenposition \mathbf{S}_1 ($= \mathbf{B}_1 = \mathbf{G}_1$) ausgewählt. Die nicht direkt aktuierten Koordinaten sind die x - und y -Auslenkungen (p_1, p_2) der Last \mathbf{S}_2 aus dem Ursprung. Die variable Seillänge wird mit l_1 bezeichnet, die durch die Koordinaten auch mittels

$$l_1 = \sqrt{(p_1 - q_1)^2 + p_2^2} \quad (3.11)$$

ausgedrückt werden kann. Die Position der beiden Massenschwerpunkte kann somit wie folgt durch die Koordinaten beschrieben werden:

$$\mathbf{S}_1 = \begin{pmatrix} q_1 \\ 0 \end{pmatrix}, \quad \mathbf{S}_2 = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}. \quad (3.12)$$

Damit ist es möglich die kinetische und potentielle Energie des Systems zu formulieren:

$$T = \frac{m_1}{2} \dot{\mathbf{S}}_1^T \dot{\mathbf{S}}_1 + \frac{m_2}{2} \dot{\mathbf{S}}_2^T \dot{\mathbf{S}}_2 = \frac{m_1 \dot{q}_1^2}{2} + \frac{m_2 \dot{p}_1^2}{2} + \frac{m_2 \dot{p}_2^2}{2} \quad (3.13)$$

$$V = m_2 g \mathbf{S}_2^T \mathbf{e}_y = m_2 g p_2. \quad (3.14)$$

Die Systemgleichungen können durch die LG zweiter Art (3.4) zunächst in Abhängigkeit der Komponenten der generalisierten Kraft Q_i bestimmt werden:

$$-Q_1 + m_2 \ddot{p}_1 = 0 \quad (3.15a)$$

$$-Q_2 + g m_2 + m_2 \ddot{p}_2 = 0 \quad (3.15b)$$

$$-Q_3 + m_1 \ddot{q}_1 = 0. \quad (3.15c)$$

Die Eingangskomponenten τ_1, τ_2 des Systems entsprechen den unterlagert geregelten Kräften der Motoren der Laufkatze bzw. der Seilwinde. Sie können über Stellkräfte vektoriell dargestellt werden:

$$\mathbf{F}_1 = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix}, \quad \mathbf{F}_2 = \begin{pmatrix} \frac{\tau_2(p_1 - q_1)}{l_1} \\ \frac{p_2 \tau_2}{l_1} \end{pmatrix}. \quad (3.16)$$

Durch die Anwendung des Prinzips der virtuellen Arbeit nach Gleichung (3.7) und (3.8) ist es möglich, die generalisierte Kraft durch diese Stellkräfte bzw. den Systemeingang auszudrücken:

$$\mathbf{Q} = \begin{pmatrix} \frac{\tau_2(p_1 - q_1)}{l_1} \\ \frac{p_2 \tau_2}{l_1} \\ \tau_1 - \frac{\tau_2(p_1 - q_1)}{l_1} \end{pmatrix}. \quad (3.17)$$

Durch Einsetzen dieser generalisierten Kraft in die Gleichungen (3.15) lässt sich ein abschließender Satz an Systemgleichungen des Einzelkrans bilden:

$$m_2 \ddot{p}_1 - \frac{\tau_2(p_1 - q_1)}{l_1} = 0 \quad (3.18a)$$

$$g m_2 + m_2 \ddot{p}_2 - \frac{p_2 \tau_2}{l_1} = 0 \quad (3.18b)$$

$$m_1 \ddot{q}_1 - \tau_1 + \frac{\tau_2(p_1 - q_1)}{l_1} = 0. \quad (3.18c)$$

3.5 Analytisches Modell des Doppelkrans

Aufbauend auf der Modellierung eines Einzelkrans wird in diesem Abschnitt eine Erweiterung des Systems auf ein Doppelkransystem in Analogie zum realen Demonstrator vorgenommen. Aufgrund der an beiden Laufkatzen befestigten Last liegt zunächst alternativ zum vorherigen Vorgehen die Nutzung der LG erster Art nahe. Bei dieser wird mittels algebraischer ZB zur Modellierung der kinematischen Schleife³ ein DAE-System erzeugt. Allerdings ist dabei die Beschreibung der Aktuierung der Seilwinden anspruchsvoller, da bereits für eine konstante Seillänge die bei der Katzfahrt auftretenden dynamischen Seilkräfte kompensiert werden müssen, ohne dass es dafür eine explizite resultierende Gesamtkraft gibt, die zu Null gesetzt werden kann. Beim realen Demonstrator sperrt ein Schneckengetriebe der Seilwinde im stromfreien Fall die Hubaktuierung. Auch bei einem Doppelkransystem ist wieder eine Modellierung mit den LG zweiter Art möglich.

3.5.1 Ansatz über ein DAE-System

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [\[18, double_crane_notebooks/DAE_double_crane_cartesian.ipynb\]](#) zu finden.

Mittels der LG erster Art wird im Folgenden ein DAE-Modell des Doppelkrans erzeugt. Dabei wird entsprechend Abschnitt 3.2.1 vorgegangen. Schließlich ergibt sich hierbei ein komplexerer Satz an Systemgleichungen, die typischerweise weniger effizient zu simulieren sind als bei einem ODE-System. Daher wird dieser Ansatz in weiteren Kapiteln nicht weiter verfolgt. Eine Dokumentation dieser Modellierung wird trotzdem als sinnvoll erachtet. Durch die Einprägung von ZB kann die kinematische Schleife wegen der gemeinsamen Last intuitiv abgebildet werden. Es wird zunächst einzig eine horizontale Aktuierung der Laufkatzen durch den Eingang $\tau = (\tau_1, \tau_2)^T$ betrachtet. Für eine effiziente Simulation mit aktuierten Seilwinden, welche für diesen Ansatz nicht mehr im Rahmen dieser Studienarbeit durchgeführt wird, wäre eine Umformulierung des Problems mit den Seillängen als Stellgröße sinnvoll. [\[12\]](#) Abbildung 4 zeigt die Konfiguration dieses Doppelkransystems.

Als direkt aktuierte Koordinate werden nur die x -Verschiebungen q_1 der Position der linken Laufkatze \mathbf{S}_1 ($= \mathbf{G}_1 = \mathbf{B}_1$) und q_2 der Position der rechten Laufkatze \mathbf{S}_5 ($= \mathbf{G}_4 = \mathbf{B}_2$) ausgewählt, als nicht direkt aktuierte Koordinaten die absolute Position (p_1, p_2) des Gelenks \mathbf{G}_2 ($= \mathbf{S}_2$) sowie (p_3, p_4) des Gelenks \mathbf{G}_3 ($= \mathbf{S}_4$).

³Diese Bezeichnung zielt auf den Vergleich des Systems mit einem Viergelenk aus der Literatur ab, durch die gemeinsame Last wird die Schleife „geschlossen“.

aus drei Massen (m_2, m_3, m_4) beschrieben, das sich unter Einhaltung der folgenden Zusammenhänge simulativ äquivalent verhält:

$$m_{\text{ges}} = m_2 + m_3 + m_4 \quad (3.23a)$$

$$m_2 = m_4 \quad (3.23b)$$

$$J = m_2 s_3^2 + m_4 s_3^2 = 2m_2 s_3^2 \quad (3.23c)$$

$$\Rightarrow m_2 = m_4 = \frac{J}{2s_3^2} \quad (3.23d)$$

$$\Rightarrow m_3 = m_{\text{ges}} - m_2 - m_4. \quad (3.23e)$$

Die Größen m_{ges} und J beziehen sich auf die in Abschnitt 3.6 identifizierten Systemparameter, die so auch bei der Modellbildung des Doppelkrans als ODE-System genutzt wurden.

Somit ist es möglich, die kinetische und potentielle Energie des Systems zu formulieren:

$$\begin{aligned} T &= \frac{m_1}{2} \dot{\mathbf{S}}_1^T \dot{\mathbf{S}}_1 + \frac{m_2}{2} \dot{\mathbf{S}}_2^T \dot{\mathbf{S}}_2 + \frac{m_3}{2} \dot{\mathbf{S}}_3^T \dot{\mathbf{S}}_3 + \frac{m_4}{2} \dot{\mathbf{S}}_4^T \dot{\mathbf{S}}_4 + \frac{m_5}{2} \dot{\mathbf{S}}_5^T \dot{\mathbf{S}}_5 \\ &= \frac{m_1 \dot{q}_1^2}{2} + \frac{m_2 \dot{p}_1^2}{2} + \frac{m_2 \dot{p}_2^2}{2} + \frac{m_3 \cdot 0.25 (\dot{p}_1 + \dot{p}_3)^2}{2} + \frac{m_3 \cdot 0.25 (\dot{p}_2 + \dot{p}_4)^2}{2} \\ &\quad + \frac{m_4 \dot{p}_3^2}{2} + \frac{m_4 \dot{p}_4^2}{2} + \frac{m_5 \dot{q}_2^2}{2} \end{aligned} \quad (3.24)$$

$$\begin{aligned} V &= (m_2 g \mathbf{S}_2^T + m_3 g \mathbf{S}_3^T + m_4 g \mathbf{S}_4^T) \mathbf{e}_y \\ &= g (m_2 p_2 + m_3 (0.5 p_2 + 0.5 p_4) + m_4 p_4). \end{aligned} \quad (3.25)$$

Unter Nutzung der Lagrange-Multiplikatoren λ_j ist es mittels der LG zweiter Art (3.2)

möglich, die Systemgleichungen auszudrücken:

$$\begin{aligned}
& - \frac{\lambda_1 p_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} + \frac{\lambda_1 q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} - \frac{\lambda_2 p_1}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} \\
& + \frac{\lambda_2 p_3}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} + m_2 \ddot{p}_1 + 0.25 m_3 \ddot{p}_1 + 0.25 m_3 \ddot{p}_3 = 0
\end{aligned} \tag{3.26a}$$

$$\begin{aligned}
& g m_2 + 0.5 g m_3 - \frac{\lambda_1 p_2}{\sqrt{p_2^2 + (p_1 - q_1)^2}} - \frac{\lambda_2 p_2 + \lambda_2 p_4}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} \\
& + m_2 \ddot{p}_2 + 0.25 m_3 \ddot{p}_2 + 0.25 m_3 \ddot{p}_4 = 0
\end{aligned} \tag{3.26b}$$

$$\begin{aligned}
& \frac{\lambda_2 (p_1 - p_3)}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} + \frac{\lambda_3 (l_0 - p_3 + q_2)}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} \\
& + \frac{m_3 (0.5 \ddot{p}_1 + 0.5 \ddot{p}_3)}{2} + m_4 \ddot{p}_3 = 0
\end{aligned} \tag{3.26c}$$

$$\begin{aligned}
& 0.5 g m_3 + g m_4 + \frac{\lambda_2 p_2}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} - \frac{\lambda_2 p_4}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} \\
& - \frac{\lambda_3 p_4}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} + 0.25 m_3 \ddot{p}_2 + 0.25 m_3 \ddot{p}_4 + m_4 \ddot{p}_4 = 0
\end{aligned} \tag{3.26e}$$

$$\begin{aligned}
& \frac{\lambda_1 (p_1 - q_1) + \sqrt{p_2^2 + (p_1 - q_1)^2} (m_1 \ddot{q}_1 - \tau_1)}{\sqrt{p_2^2 + (p_1 - q_1)^2}} = 0
\end{aligned} \tag{3.26f}$$

$$\begin{aligned}
& \frac{-\lambda_3 (l_0 - p_3 + q_2) + \sqrt{p_4^2 + (l_0 - p_3 + q_2)^2} (m_5 \ddot{q}_2 - \tau_2)}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} = 0.
\end{aligned} \tag{3.26g}$$

3.5.2 Ansatz über ein ODE-System

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [\[18, flatness_notebooks/ODE_flatness_analysis.ipynb\]](#) zu finden.

Mittels der LG zweiter Art wird im Folgenden ein ODE-Modell des Doppelkrans erzeugt. Dabei wird entsprechend Abschnitt 3.2.2 vorgegangen. Abbildung 5 zeigt die Konfiguration dieses Doppelkransystems.

Als direkt aktuierte Koordinate werden zunächst nur die x -Verschiebungen q_1 der Position der linken Laufkatze \mathbf{S}_1 ($= \mathbf{G}_1 = \mathbf{B}_1$) und q_2 der Position der rechten Laufkatze \mathbf{S}_3 ($= \mathbf{B}_2 = \mathbf{G}_6$) ausgewählt. Die absolute Position (p_1, p_2) des Lastschwerpunkts \mathbf{S}_2 sowie die Orientierung der Last p_3 gegenüber der Horizontalen entsprechen den nicht direkt aktuierten Koordinaten. Die variablen Seillängen werden mit l_1 sowie l_2 bezeichnet,

der Komponenten der generalisierten Kraft Q_i bestimmt werden:

$$-Q_1 + m_2 \ddot{p}_1 = 0 \quad (3.32a)$$

$$-Q_2 + gm_2 + m_2 \ddot{p}_2 = 0 \quad (3.32b)$$

$$J_2 \ddot{p}_3 - Q_3 = 0 \quad (3.32c)$$

$$-Q_4 + m_1 \ddot{q}_1 = 0 \quad (3.32d)$$

$$-Q_5 + m_3 \ddot{q}_2 = 0. \quad (3.32e)$$

Über die Komponenten des Systemeingangs τ und normierte Richtungsvektoren der Führungsschiene bzw. der Seile können die vektoriellen Stellkräfte dargestellt werden:

$$\mathbf{F}_1 = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix}, \mathbf{F}_2 = \begin{pmatrix} \tau_2 \\ 0 \end{pmatrix}, \mathbf{F}_3 = \begin{pmatrix} \frac{\tau_3(p_1 - q_1 - s_2 \cos(p_3))}{l_1} \\ \frac{\tau_3(p_2 - s_2 \sin(p_3))}{l_1} \end{pmatrix}, \mathbf{F}_4 = \begin{pmatrix} \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 \cos(p_3))}{l_2} \\ \frac{\tau_4(p_2 + s_2 \sin(p_3))}{l_2} \end{pmatrix}. \quad (3.33)$$

Durch die Anwendung des Prinzips der virtuellen Arbeit ist es möglich, die generalisierte Kraft durch die Stellkräfte bzw. den Systemeingang auszudrücken (Abkürzungen $s_x := \sin x$ und $c_x := \cos x$):

$$\mathbf{Q} = \begin{pmatrix} \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 c_{p_3})}{l_2} + \frac{\tau_3(p_1 - q_1 - s_2 c_{p_3})}{l_1} \\ \frac{\tau_4(p_2 + s_2 s_{p_3})}{l_2} + \frac{\tau_3(p_2 - s_2 s_{p_3})}{l_1} \\ \frac{s_2 \tau_4(p_2 + s_2 s_{p_3}) c_{p_3}}{l_2} - \frac{s_2 \tau_4(-l_0 + p_1 - q_2 + s_2 c_{p_3}) s_{p_3}}{l_2} - \frac{s_2 \tau_3(p_2 - s_2 s_{p_3}) c_{p_3}}{l_1} + \frac{s_2 \tau_3(p_1 - q_1 - s_2 c_{p_3}) s_{p_3}}{l_1} \\ \tau_1 - \frac{\tau_3(p_1 - q_1 - s_2 c_{p_3})}{l_1} \\ \tau_2 - \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 c_{p_3})}{l_2} \end{pmatrix}. \quad (3.34)$$

Durch Einsetzen dieser generalisierten Kraft in die Gleichungen (3.32) lässt sich ein abschließender Satz an Systemgleichungen des Doppelkrans bilden:

$$m_2 \ddot{p}_1 - \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 \cos(p_3))}{l_2} - \frac{\tau_3(p_1 - q_1 - s_2 \cos(p_3))}{l_1} = 0 \quad (3.35a)$$

$$gm_2 + m_2 \ddot{p}_2 - \frac{\tau_4(p_2 + s_2 \sin(p_3))}{l_2} - \frac{\tau_3(p_2 - s_2 \sin(p_3))}{l_1} = 0 \quad (3.35b)$$

$$J_2 \ddot{p}_3 - \frac{s_2 \tau_4(p_2 + s_2 \sin(p_3)) \cos(p_3)}{l_2} + \frac{s_2 \tau_4(-l_0 + p_1 - q_2 + s_2 \cos(p_3)) \sin(p_3)}{l_2} \\ + \frac{s_2 \tau_3(p_2 - s_2 \sin(p_3)) \cos(p_3)}{l_1} - \frac{s_2 \tau_3(p_1 - q_1 - s_2 \cos(p_3)) \sin(p_3)}{l_1} = 0 \quad (3.35c)$$

$$m_1 \ddot{q}_1 - \tau_1 + \frac{\tau_3(p_1 - q_1 - s_2 \cos(p_3))}{l_1} = 0 \quad (3.35d)$$

$$m_3 \ddot{q}_2 - \tau_2 + \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 \cos(p_3))}{l_2} = 0. \quad (3.35e)$$

Alternativ kann das Doppelkransystem für spätere regelungstechnische Untersuchungen auch eingangsaffin im Zustandsraum beschrieben werden:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\boldsymbol{\tau} \text{ mit} \\ \mathbf{x} &= (p_1, p_2, p_3, q_1, q_2, \dot{p}_1, \dot{p}_2, \dot{p}_3, \dot{q}_1, \dot{q}_2)^T, \\ \mathbf{f}(\mathbf{x}) &= (\dot{p}_1, \dot{p}_2, \dot{p}_3, \dot{q}_1, \dot{q}_2, 0, -g, 0, 0, 0)^T, \\ \mathbf{g}(\mathbf{x}) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{p_1 - q_1 - s_2 \cos(p_3)}{l_1 m_2} & \frac{-l_0 + p_1 - q_2 + s_2 \cos(p_3)}{l_2 m_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{p_2 - s_2 \sin(p_3)}{l_1 m_2} & \frac{p_2 + s_2 \sin(p_3)}{l_2 m_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{s_2(p_1 \sin(p_3) - p_2 \cos(p_3) - q_1 \sin(p_3))}{J_2 l_1} & \frac{s_2(l_0 \sin(p_3) - p_1 \sin(p_3) + p_2 \cos(p_3) + q_2 \sin(p_3))}{J_2 l_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{m_1} & 0 & \frac{-p_1 + q_1 + s_2 \cos(p_3)}{l_1 m_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{m_3} & 0 & \frac{l_0 - p_1 + q_2 - s_2 \cos(p_3)}{l_2 m_3} & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (3.36)\end{aligned}$$

3.6 Identifikation der Systemparameter

Die meisten Systemparameter der Modelle sind Massen oder geometrische Größen und lassen sich durch Messungen mit einem Lineal oder einer Waage empirisch bestimmen. Lediglich für das Trägheitsmoment J der Last ist eine direkte experimentelle Bestimmung, etwa durch einen Pendelversuch, zu aufwendig. Daher wird dieser Parameter unter der Annahme eines Quaders mit homogener Masseverteilung und Rotationsachse durch seinen Mittelpunkt in der vertikalen Aufhängungsebene nach [16] berechnet:

$$J = \frac{1}{12} m_2 ((2s_2)^2 + h^2). \quad (3.37)$$

Dabei entspricht h der Höhe der Last, $2s_2$ der Lastlänge und m_2 der Masse der Last. Die numerische Belegung der Systemparameter ist in Tabelle 1 zusammengefasst.

Tabelle 1 – Physikalische Parameter des Doppelkrandemonstratorsystems.

Parameter	Bezeichnung im ODE-Modell	Wert
Masse Laufkatze 1	m_1	0,45 kg
Masse Laufkatze 2	m_3	0,45 kg
Masse Last	m_2	0,557 kg
Länge Last	$2s_2$	0,15 m
Höhe Last	h	0,09 m
Initialer Laufkatzenabstand	l_0	0,3 m
Trägheitsmoment Last	J_2	0,00455 kgm ²

Kapitel 4

Flachheitsanalyse

Zur Erweiterung des Begriffs der Steuerbarkeit auf nichtlineare Systeme wird die sogenannte (differenzielle) Flachheit als Systemeigenschaft definiert. Gerade für flache Mehrgrößensysteme gestaltet sich die Bestimmung geeigneter Eingangsgrößenverläufe zur Steuerung und auch Folgeregelung aus der Vorgabe nur weniger Referenztrajektorien als sehr praktikabel.

4.1 Definition differenzieller Flachheit

Ein System der Form $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}, \mathbf{u})$ mit $\mathbf{F}, \mathbf{x} \in \mathbb{R}^n$ und $\mathbf{u} \in \mathbb{R}^m$ heißt (differenziell) flach, falls ein m -Tupel $y := (y_1, \dots, y_m)^T$ sowie glatte Funktionen Ψ, θ existieren, so dass

$$\mathbf{x} = \Psi(\mathbf{y}, \dot{\mathbf{y}}, \dots, \mathbf{y}^{(n_x)}) \text{ mit } n_x < \infty \text{ und} \quad (4.1a)$$

$$\mathbf{u} = \theta(\mathbf{y}, \dot{\mathbf{y}}, \dots, \mathbf{y}^{(n_u)}) \text{ mit } n_u < \infty \text{ gilt.} \quad (4.1b)$$

Dabei ist \mathbf{y} ein flacher Ausgang. Aus der Existenz eines flachen Ausgangs folgt, dass die Systemgrößen bestehend aus dem Zustand \mathbf{x} und Eingang \mathbf{u} eindeutig aus dem flachen Ausgang \mathbf{y} und einer endlichen Anzahl dessen Zeitableitungen berechnet werden können, also keine Differenzialgleichungen durch Integration dafür gelöst werden müssen. Eine alternative Formulierung des Flachheitsbegriffs verzichtet auf die Angabe der gleichen Dimension m von Eingang und flachem Ausgang, fordert aber die differenzielle Unabhängigkeit der Komponenten von \mathbf{y} . Die Existenzbedingungen eines flachen Ausgangs sind bei Eingrößensystemen bekannt, allerdings ist der systematische Flachheitsnachweis und die Berechnung eines flachen Ausgangs bei Mehrgrößensystemen im Allgemeinen nicht abschließend gelöst. [11, S. 185]

4.2 Flachheitsanalyse von Mehrgrößensystemen

Im Folgenden wird ein prinzipielles praktisches Vorgehen zur Bestimmung flacher Ausgänge von Mehrgrößensystemen sowie zur Parametrisierung der Systemgrößen anhand solcher flachen Ausgänge skizziert. Für eine mathematisch fundierte und systematische Herangehensweise sei auf den Beitrag [5] und die dort zitierten Quellen verwiesen. Im Unterschied zu diesem Beitrag wird hier aber nicht auf den Gleichungen des sogenannten Tangentialsystems operiert, sondern auf den originalen nichtlinearen Gleichungen¹.

Es sei ein nichtlineares Mehrgrößensystem der Form der Gleichungen (4.1) gegeben. Dieses sei affin bezüglich des Eingangs \mathbf{u} . Der Eingang könne mittels der Systemgleichungen eliminiert werden, so dass ein autonomes System aus $p := n - m$ Gleichungen folgt. Diese Gleichungen werden wiederum zur Elimination von p der übrigen Zustandskomponenten genutzt, wobei jeweils eine Gleichung und eine Systemgröße gemeinsam eliminiert werden. Wenn sich dieses sukzessive Vorgehen bis einschließlich der letzten Gleichung durchführen lässt, verbleiben schlussendlich $n - p = m$ Systemgrößen, die keiner Gleichung mehr genügen müssen. Diese bilden dann den flachen Ausgang \mathbf{y} .

Zur sukzessiven Elimination von Gleichungen und Systemgrößen bietet sich folgendes Vorgehen an: Um Systemgleichungen und Eingangskomponenten, welche mit geringem Aufwand eliminiert werden können zu identifizieren, wird die Jacobi-Matrix \mathbf{J}_i der zu diesem Zeitpunkt noch i verbleibenden Systemgleichungen bezüglich des Eingangs $\mathbf{u}_{m-(n-i)}$ gebildet. Dabei sind die zu den Matrixzeilen korrespondierenden Gleichungen geeignet, in denen nur ein isolierter Spalteneintrag $\varepsilon \neq 0$ steht, also weitere Einträge der Spalte Null sind:

$$\mathbf{J}_i = \begin{pmatrix} * & \cdots & * & 0 & * & \cdots & * \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ * & \cdots & * & 0 & * & \cdots & * \\ * & \cdots & * & \varepsilon & * & \cdots & * \\ * & \cdots & * & 0 & * & \cdots & * \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ * & \cdots & * & 0 & * & \cdots & * \end{pmatrix}. \quad (4.2)$$

Falls keine solche Spalte zu finden ist, kann sie durch eine Transformation der Systemgleichungen durch die Multiplikation mit der Matrix \mathbf{T}_i erzeugt werden:

$$\mathbf{T}_i \mathbf{J}_i = \begin{pmatrix} \mathbf{I}_{m-(n-i)} \\ \mathbf{0}_{(n-m) \times (m-(n-i))} \end{pmatrix}. \quad (4.3)$$

Dabei kann \mathbf{T}_i über das linke Orthokomplement (der Vektorraum aller Zeilen, die orthogonal zu allen Spalten von \mathbf{J}_i sind) \mathbf{J}_i^{\perp} und die linke Pseudoinverse $\mathbf{J}_i^{\perp+}$ der

¹Direkt auf den nichtlinearen Systemgleichungen zu operieren erschwert ggf. die Eliminierung von Größen bzw. ist diese nicht immer möglich. Es muss im Gegensatz zu [5] jedoch keine Integrabilitätsbedingung geprüft werden.

Jacobi-Matrix folgendermaßen gebildet werden [5, Abschnitt 2.1.2]:

$$\begin{aligned} \mathbf{J}_i^{\text{L}+} \mathbf{J}_i &= \mathbf{I}_{m-(n-i)} \\ \mathbf{J}_i^{\text{L}\perp} \mathbf{J}_i &= \mathbf{0}_{(n-m) \times (m-(n-i))} \\ \Rightarrow \mathbf{T}_i &= \begin{pmatrix} \mathbf{J}_i^{\text{L}+} \\ \mathbf{J}_i^{\text{L}\perp} \end{pmatrix}. \end{aligned} \quad (4.4)$$

Nachdem die Eingangskomponenten aus den Systemgleichungen eliminiert wurden, kann die Behandlung der anderen noch vorhandenen Systemgrößen analog durch Bildung der Jacobi-Matrix bezüglich dieser erfolgen. Der flache Ausgang \mathbf{y} besteht aus der Menge an Systemgrößen, welche im Idealfall nach Elimination aller Gleichungen übrig ist. Der konstruktive Flachheitsnachweis für ein System ist erbracht, wenn mittels der Zusammenhänge aus den Systemgleichungen entsprechend der Gleichungen (4.1) alle Systemgrößen (insbesondere auch der Systemeingang) allein durch $(\mathbf{y}, \dot{\mathbf{y}}, \ddot{\mathbf{y}}, \dots)$ parametrisiert sind. Dafür werden die Systemgleichung in umgekehrter Reihenfolge ihrer Eliminierung zur Berechnung der Parametrisierung verwendet. Jede eliminierte Gleichung wird als Definitionsgleichung einer Hilfsgröße auf Basis des flachen Ausgangs (und ggf. seiner Zeitableitungen) aufgefasst, die in den weiteren Gleichungen dann ggf. auch abgeleitet vorkommt.

Bei nichtlinearen Systemen ist es allerdings oft nicht direkt möglich, alle Gleichungen direkt zu eliminieren, so dass Freiheitsgrade bestehen, was eine Schwierigkeit der Flachheitsanalyse im Mehrgrößenfall darstellt. Der skizzierte Algorithmus ist deshalb nur als hinreichend zur Bestimmung eines flachen Ausgangs, aber nicht als notwendig zu betrachten.

4.3 Anwendung Flachheitsanalyse am Einzelkran

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [18, flatness_notebooks/ODE_flatness_analysis_single_crane.ipynb] zu finden.

Die im vorherigen Abschnitt allgemein geschilderte Vorgehensweise wird nun auf das Einzelkransystem aus Abschnitt 3.4 angewendet. Aus der analytischen Modellbildung mit den LG zweiter Art folgen die drei Systemgleichungen (3.18a), (3.18b) und (3.18c), welche als Grundlage des Flachheitsnachweises dienen.

Zunächst wird die Jacobi-Matrix dieser Gleichungen bezüglich $\boldsymbol{\tau} = (\tau_1, \tau_2)^T$ gebildet:

$$\mathbf{J}_3 = \begin{pmatrix} 0 & -\frac{p_1 - q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \\ 0 & -\frac{p_2}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \\ -1 & \frac{p_1 - q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \end{pmatrix}. \quad (4.5)$$

Dabei ist zu erkennen, dass in der ersten Spalte der Jacobi-Matrix die Eingangskomponente τ_1 isoliert vorkommt, also durch die korrespondierende dritte Systemgleichung (3.18c) bestimmt werden kann. Eine Transformation des Systems ist dafür nicht notwendig. Dementsprechend können die letzte Zeile von \mathbf{J}_3 und die Ausgangskomponente τ_1 eliminiert werden.

Im nächsten Schritt wird die Jacobi-Matrix \mathbf{J}_2 der verbliebenen Gleichungen (3.18a) und (3.18b) bezüglich τ_2 berechnet:

$$\mathbf{J}_2 = \begin{pmatrix} -\frac{p_1 - q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \\ -\frac{p_2}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \end{pmatrix}. \quad (4.6)$$

Diese weist keine Spalte mit nur einer Komponente ungleich Null auf, also ist eine Transformation \mathbf{T}_2 des Systems entsprechend der Gleichungen (4.4) durchzuführen. Wegen der nicht-quadratischen Dimension von \mathbf{J}_2 gilt es eine linke Pseudoinverse $\mathbf{J}_2^{L+} \in \mathbb{R}^{1 \times 2}$ zu finden:

$$\mathbf{J}_2^{L+} \mathbf{J}_2 = \mathbf{I}_1 = 1 \quad (4.7)$$

sowie das linke Orthokomplement $\mathbf{J}_2^{L\perp} \in \mathbb{R}^{1 \times 2}$:

$$\mathbf{J}_2^{L\perp} \mathbf{J}_2 = \mathbf{0}_{1 \times 1} = 0. \quad (4.8)$$

Da beide Matrizen nicht eindeutig sind, kann für ihre Bestimmung der folgende Ansatz gewählt werden:

$$\mathbf{J}_2^{L+} = \left((J_{2,(1,1)})^{-1} \mathbf{0}_{1 \times 1} \right), \mathbf{J}_2^{L\perp} = \begin{pmatrix} -J_{2,(2,1)} & J_{2,(1,1)} \end{pmatrix}. \quad (4.9)$$

Durch Einsetzen der Komponenten von \mathbf{J}_2 folgt für die Transformationsmatrix:

$$\mathbf{T}_2 = \begin{pmatrix} \frac{\sqrt{p_1^2 - 2p_1q_1 + p_2^2 + q_1^2}}{-p_1 + q_1} & 0 \\ \frac{p_2}{\sqrt{p_2^2 + (p_1 - q_1)^2}} & \frac{-p_1 + q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \end{pmatrix}. \quad (4.10)$$

Die damit transformierten verbliebenen Systemgleichungen ergeben sich zu:

$$\frac{\left(-m_2 \ddot{p}_1 \sqrt{p_2^2 + (p_1 - q_1)^2} + \tau_2 (p_1 - q_1) \right) \sqrt{p_1^2 - 2p_1q_1 + p_2^2 + q_1^2}}{(p_1 - q_1) \sqrt{p_2^2 + (p_1 - q_1)^2}} = 0 \quad (4.11a)$$

$$g \frac{m_2 (-gp_1 + gq_1 - p_1 \ddot{p}_2 + p_2 \ddot{p}_1 + \ddot{p}_2 q_1)}{\sqrt{p_1^2 - 2p_1q_1 + p_2^2 + q_1^2}} = 0. \quad (4.11b)$$

Per Konstruktion gilt für die dazu bezüglich τ_2 gebildete Jacobi-Matrix $\mathbf{J}'_2 = (1 \ 0)^T$. Die erste transformierte Systemgleichung (4.11a) kann somit zusammen mit τ_2 eliminiert werden. Es verbleibt die Systemgleichung (4.11b), welche folgende Menge an Systemgrößen und deren Ableitungen beinhaltet:

$$M = \{p_1, \ddot{p}_1, p_2, \ddot{p}_2, q_1\}. \quad (4.12)$$

Das rein algebraische auftreten von q_1 legt nahe, dass eine Parametrisierung dieser und aller weiteren Systemgrößen durch einen flachen Ausgang $\mathbf{y} = (p_1, p_2)^T$ möglich ist.

Eine Parametrisierung $\tau_1 = \text{func}(p_1, p_2, q_1, \ddot{q}_1, \tau_2)$ ist daher möglich. Die Symbolik `func` stellt in dieser Arbeit einen allgemeinen und nicht weiter spezifizierten funktionalen Zusammenhang dar, welcher sich bei jeder Erwähnung unterscheiden kann.

4.4 Anwendung Flachheitsanalyse am Doppelkran

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [18, flatness_notebooks/ODE_flatness_analysis.ipynb] zu finden.

Aus der analytischen Modellbildung des Doppelkransystems mit den LG zweiter Art im Unterabschnitt 3.5.2 folgen die fünf Systemgleichungen (3.35a) bis (3.35e), welche als Grundlage des Flachheitsnachweises dienen.

Zunächst wird die Jacobi-Matrix dieser Gleichungen bezüglich $\boldsymbol{\tau} = (\tau_1, \tau_2, \tau_3, \tau_4)^T$ gebildet (Abkürzungen $s_x := \sin x$ und $c_x := \cos x$):

$$\mathbf{J}_5 = \begin{pmatrix} 0 & 0 & -\frac{p_1 - q_1 - s_2 c(p_3)}{l_1} & -\frac{-l_0 + p_1 - q_2 + s_2 c(p_3)}{l_2} \\ 0 & 0 & -\frac{p_2 - s_2 s(p_3)}{l_1} & -\frac{p_2 + s_2 s(p_3)}{l_2} \\ 0 & 0 & \frac{s_2(p_2 - s_2 s(p_3))c(p_3)}{l_1} - \frac{s_2(p_1 - q_1 - s_2 c(p_3))s(p_3)}{l_1} - \frac{s_2(p_2 + s_2 s(p_3))c(p_3)}{l_2} + \frac{s_2(-l_0 + p_1 - q_2 + s_2 c(p_3))s(p_3)}{l_2} \\ -1 & 0 & \frac{p_1 - q_1 - s_2 c(p_3)}{l_1} & 0 \\ 0 & -1 & 0 & \frac{-l_0 + p_1 - q_2 + s_2 c(p_3)}{l_2} \end{pmatrix}.$$

Dabei ist zu erkennen, dass in den ersten beiden Spalten der zu den Systemgleichungen korrespondierenden Jacobi-Matrix zwei Eingangsgrößen τ_1 und τ_2 jeweils isoliert vorkommen. Für deren Bestimmung ergeben sich keine redundanten Gleichungen. Dementsprechend können die letzten beiden Zeilen von \mathbf{J}_5 eliminiert werden. Eine vorläufige Parametrisierung $\tau_1 = \text{func}(p_1, p_3, q_1, \ddot{q}_1, \tau_3)$ und $\tau_2 = \text{func}(p_1, p_3, q_2, \ddot{q}_2, \tau_4)$ ist möglich.

Bei den übrigen Eingangsgrößen τ_3, τ_4 gibt es in der darauf bezogenen Jacobimatrix \mathbf{J}_3 zu den ersten drei Systemgleichungen (3.35a), (3.35b) und (3.35c) keine Spalten mehr,

in denen diese Eingangskomponenten nur einmal vorkommen:

$$\mathbf{J}_3 = \begin{pmatrix} -\frac{p_1 - q_1 - s_2^c(p_3)}{l_1} & -\frac{-l_0 + p_1 - q_2 + s_2^c(p_3)}{l_2} \\ -\frac{p_2 - s_2^s(p_3)}{l_1} & -\frac{p_2 + s_2^s(p_3)}{l_2} \\ \frac{s_2(p_2 - s_2^s(p_3))^c(p_3)}{l_1} - \frac{s_2(p_1 - q_1 - s_2^c(p_3))^s(p_3)}{l_1} & -\frac{s_2(p_2 + s_2^s(p_3))^c(p_3)}{l_2} + \frac{s_2(-l_0 + p_1 - q_2 + s_2^c(p_3))^s(p_3)}{l_2} \end{pmatrix}.$$

Wegen der nicht-quadratischen Dimension von \mathbf{J}_3 gilt es eine linke Pseudoinverse $\mathbf{J}_3^{L+} \in \mathbb{R}^{2 \times 3}$ zu finden:

$$\mathbf{J}_3^{L+} \mathbf{J}_3 = \mathbf{I}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (4.13)$$

sowie das linke Orthokomplement (der Vektorraum aller Zeilen, die orthogonal zu allen Spalten von \mathbf{J}_3 sind) $\mathbf{J}_3^{L\perp} \in \mathbb{R}^{1 \times 3}$:

$$\mathbf{J}_3^{L\perp} \mathbf{J}_3 = \mathbf{0}_{1 \times 2} = \begin{pmatrix} 0 & 0 \end{pmatrix}, \quad (4.14)$$

Damit kann eine Transformationsmatrix \mathbf{T}_3 gebildet werden, die in Multiplikation mit den übrigen Systemgleichungen bzw. der damit korrespondierenden Jacobimatrix erneut Spalten mit jeweils nur einem konstanten nicht-Null-Eintrag zur Folge hat:

$$\mathbf{T}_3 \mathbf{J}_3 = \begin{pmatrix} \mathbf{J}_3^{L+} \\ \mathbf{J}_3^{L\perp} \end{pmatrix} \mathbf{J}_3 = \begin{pmatrix} \mathbf{I}_2 \\ \mathbf{0}_{1 \times 2} \end{pmatrix} = \mathbf{I}_{3 \times 2}. \quad (4.15)$$

Da beide Matrizen nicht eindeutig sind, kann für ihre Bestimmung der folgende Ansatz gewählt werden:

$$\mathbf{J}_3^{L+} = \left(\begin{pmatrix} J_{3,(1,1)} & J_{3,(1,2)} \\ J_{3,(2,1)} & J_{3,(2,2)} \end{pmatrix}^{-1} \mathbf{0}_{2 \times 1} \right), \quad \mathbf{J}_3^{L\perp} = \begin{pmatrix} J_{3,(2,1)} J_{3,(3,2)} - J_{3,(2,2)} J_{3,(3,1)} \\ -J_{3,(1,1)} J_{3,(3,2)} + J_{3,(1,2)} J_{3,(3,1)} \\ J_{3,(1,1)} J_{3,(2,2)} - J_{3,(1,2)} J_{3,(2,1)} \end{pmatrix}^T. \quad (4.16)$$

Aus der Multiplikation der somit gefundenen Transformationsmatrix \mathbf{T}_3 mit den übrigen drei Systemgleichungen lassen sich entsprechend der Anforderungen an ihre Konstruktion anschließend die noch übrigen Eingangskomponenten τ_3 und τ_4 sowie die beiden transformierten Systemgleichungen eliminieren. So ergibt sich eine vorläufige Parametrisierung $\tau_3 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, q_1, q_2)$ und $\tau_4 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, q_1, q_2)$.

Die letzte verbliebene Systemgleichung enthält folgende Menge M an Systemgrößen und deren Ableitungen:

$$M = \{p_1, p_2, p_3, \ddot{p}_1, \ddot{p}_2, \ddot{p}_3, q_1, q_2\}. \quad (4.17)$$

In dieser Gleichung sind sowohl q_1 als auch q_2 rein algebraisch enthalten. Eine dieser beiden Größen kann also ebenso wie der Eingang $\boldsymbol{\tau}$ eliminiert werden. Die übrigen Systemgrößen bilden einen flachen Ausgang $\mathbf{y} = (p_1, p_2, p_3, q_1)^T$ oder alternativ $\tilde{\mathbf{y}} = (p_1, p_2, p_3, q_2)^T$.

Die zur Eliminierung umgeformten Systemgleichungen ermöglichen die Parametrisierung der Systemgrößen durch einen flachen Ausgang, welche konstruktiv entsprechend dieses Flachheitsnachweises bestimmt werden kann. Auf die konkrete Angabe der Parametrisierungen wird aus Praktikabilitätsgründen verzichtet und stattdessen auf das zu Beginn dieses Abschnitts 4.4 erwähnte Jupyter-Notebook verwiesen.

Eine abschließende flache Parametrisierung aller Systemgrößen durch den flachen Ausgang $\mathbf{y} = (p_1, p_2, p_3, q_1)^T$ enthält folgende Zusammenhänge:

$$q_2 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1) \quad (4.18a)$$

$$\tau_1 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1, \ddot{q}_1) \quad (4.18b)$$

$$\tau_2 = \text{func}(p_1, \dot{p}_1, \ddot{p}_1, p_1^{(3)}, p_1^{(4)}, p_2, \dot{p}_2, \ddot{p}_2, p_2^{(3)}, p_2^{(4)}, p_3, \dot{p}_3, \ddot{p}_3, p_3^{(3)}, p_3^{(4)}, q_1, \dot{q}_1, \ddot{q}_1) \quad (4.18c)$$

$$\tau_3 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1) \quad (4.18d)$$

$$\tau_4 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1). \quad (4.18e)$$

Kapitel 5

Steuerungs- und Regelungsentwurf

Auf Grundlage der Flachheitsanalyse des Brückenkransystems ist es im Folgenden möglich, Referenztrajektorien für einen flachen Ausgang aufzustellen, die Eingangsgrößen danach zu parametrisieren und eine Folgerelung dafür zu ermitteln.

5.1 Trajektorienplanung für den flachen Ausgang

Aufgabe der Trajektorienplanung ist es, Zeitverläufe des flachen Ausgangs vorzugeben, anhand derer später die Eingänge parametrisiert werden können, um eine Überführung des Kransystems zwischen zwei Ruhelagen zu ermöglichen. Für den real vorhandenen Versuchsstand gibt es bisher keine formal spezifizierten Grenzwerten für die Beschleunigungen oder die maximalen Geschwindigkeiten seiner Komponenten. **Aus physikalischer beziehungsweise technischer Sicht ist es sinnvoll, eine Planung der Referenztrajektorien so vorzunehmen, dass die Stell- bzw. Eingangsgrößen des Systems stetig verlaufen.**

Aufgrund der einfachen Vorgabe von Randbedingungen und Erfüllung von Differenzierbarkeitsbedingungen ist die Wahl eines polynombasierten Trajektorienansatzes sinnvoll. Aus der Flachheitsanalyse in Abschnitt 4.4 ist ersichtlich, dass für den flachen Ausgang $\mathbf{y} = (y_1, y_2, y_3, y_4)^T = (p_1, p_2, p_3, q_1)^T$ der Eingang $\tau_2 = \text{func}(y_1^{(4)}, y_2^{(4)}, y_3^{(4)}, \ddot{y}_4, \dots)$ die höchsten Ausgangsableitungen aufweist. Damit für die polynomialen Trajektorien außerdem aus den Ruhelagen $(t_0, y_{i,0})$ ein stetig differenzierbarer Übergang der Eingangsgrößenverläufe (ohne Sprünge an den Rändern) in die Ruhelage $(t_e, y_{i,e})$ gewährleistet werden kann, müssen demnach folgende Bedingungen erfüllt sein [11, S. 230]:

$$\begin{aligned} y_i(t_0) &= y_{i,0} & \text{für } i = 1, 2, 3, 4 \\ y_i(t_e) &= y_{i,e} & \text{für } i = 1, 2, 3, 4 \\ \dot{y}_i(t_0) = \ddot{y}_i(t_0) = y_i^{(3)}(t_0) = y_i^{(4)}(t_0) &= 0 & \text{für } i = 1, 2, 3 \\ \dot{y}_i(t_e) = \ddot{y}_i(t_e) = y_i^{(3)}(t_e) = y_i^{(4)}(t_e) &= 0 & \text{für } i = 1, 2, 3 \\ \dot{y}_4(t_0) = \ddot{y}_4(t_0) &= 0 \\ \dot{y}_4(t_e) = \ddot{y}_4(t_e) &= 0. \end{aligned} \tag{5.1}$$

Es ergeben sich Ansatzfunktionen für die Trajektorien des flachen Ausgangs mit jeweiliger Ordnung $2N_i - 1$, wobei N_i der Anzahl der Randbedingungen des jeweiligen Ausgangs entspricht:

$$\begin{aligned} y_i(t) &= a_{i,9}t^9 + a_{i,8}t^8 + \dots + a_{i,0} \quad \text{für } i = 1, 2, 3; \quad t_0 < t < t_e \\ y_4(t) &= a_{4,5}t^5 + a_{4,4}t^4 + \dots + a_{4,0} \quad \text{für } t_0 < t < t_e. \end{aligned} \quad (5.2)$$

Die Koeffizienten der Trajektorien $a_{i,j}$ können durch Einsetzen der Randbedingungen und Lösen des resultierenden linearen Gleichungssystems bestimmt werden. In der Simulation wurde als Implementierung dafür die Funktion `condition_poly` der Bibliothek `sympbtools` verwendet.

5.2 Trajektorienfolgeregelung

5.2.1 Vektorieller relativer Grad

Es wird ein Mehrgrößensystem mit m Eingangskomponenten u_1, \dots, u_m und m Ausgangskomponenten y_1, \dots, y_m betrachtet:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}, \quad \mathbf{y} = \mathbf{h}(\mathbf{x}). \quad (5.3)$$

Dabei gelte für den Zustand $\mathbf{x} \in \mathbb{R}^n$ sowie für die Vektorfelder $\mathbf{f} : M \rightarrow \mathbb{R}^n$, $\mathbf{h} : M \rightarrow \mathbb{R}^m$ und die Matrix $\mathbf{g} = (\mathbf{g}_1, \dots, \mathbf{g}_m) : M \rightarrow \mathbb{R}^{n \times m}$, wobei M eine offene Teilmenge $M \subseteq \mathbb{R}^n$ repräsentiert. Dieses System hat an der Stelle $\gamma \in M$ den vektoriellen relativen Grad $\mathbf{r} = (r_1, \dots, r_m)^T$, falls [11, S. 194]:

1. Die Lie-Ableitungen $L_{\mathbf{g}_j} L_{\mathbf{f}}^k h_i(\mathbf{x}) = 0$ für alle \mathbf{x} aus einer Umgebung von γ sowie für alle $i, j \in \{1, \dots, m\}$ und $k \in \{0, \dots, r - 2\}$ und
2. die sogenannte Entkopplungsmatrix

$$\mathbf{\Lambda} = \begin{pmatrix} L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_1-1} h_1(\mathbf{x}) & \dots & L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_1-1} h_1(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_m-1} h_m(\mathbf{x}) & \dots & L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_m-1} h_m(\mathbf{x}) \end{pmatrix} \quad (5.4)$$

im Punkt $\mathbf{x} = \gamma$ regulär ist.

5.2.2 Statische Rückführung

Durch die Rückkopplung gemessener Zustandskomponenten soll eine Stabilisierung des Systems und Folgeregelung entlang der Referenztrajektorien erreicht werden. Bei der

statischen Rückführung wird die nichtlinearen Eingangs-Ausgangs-Verkopplungen eines Systems mit dem Verhalten

$$\mathbf{y} = \mathbf{\Gamma}(\mathbf{x}) + \mathbf{\Lambda}(\mathbf{x})\mathbf{u} \quad \text{mit} \quad \mathbf{\Gamma}(\mathbf{x}) = (L_{\mathbf{f}}^{r_1} h_1(\mathbf{x}), \dots, L_{\mathbf{f}}^{r_m} h_m(\mathbf{x}))^T \quad (5.5)$$

durch eine statische (Zustands-)Rückführung

$$\mathbf{\Gamma}(\mathbf{x}) + \mathbf{\Lambda}(\mathbf{x})\mathbf{u} \stackrel{!}{=} \mathbf{v} \Rightarrow \mathbf{u} = \mathbf{\Lambda}^{-1}(\mathbf{x}) \cdot (\mathbf{v} - \mathbf{\Gamma}(\mathbf{x})) \quad (5.6)$$

unter Nutzung eines virtuellen Eingangs $\mathbf{v} = (v_1, \dots, v_m)^T$ zu kompensieren. Die Zusammensetzung dieses virtuellen Eingangs ist in Unterabschnitt 5.2.3 erklärt. [11, S. 195]

Bei dem vorliegenden Doppelkransystem kann durch Zeitableitungen der jeweiligen Komponente des flachen Ausgangs y_i die damit korrespondierende Komponente des relativen Grades r_i bestimmt werden. Diese entspricht der Ableitungsordnung, bei der das erste mal eine Komponente des Eingangs $\mathbf{u} := \boldsymbol{\tau}$ explizit auftritt. Hierbei werden die Zeitableitungen des Ausgangs durch Lie-Ableitungen entlang des Vektorfelds des Zustandsraummodells des Systems $\boldsymbol{\delta} := \dot{\mathbf{x}} = \mathbf{f} + \mathbf{g}\boldsymbol{\tau}$, welches in Gleichung (3.36) bestimmt wurde, rekursiv erzeugt:

$$y_i^{(k)} = L_{\boldsymbol{\delta}} y_i^{(k-1)}. \quad (5.7)$$

In Tabelle 2 sind die daraus bestimmten relativen Grade sowie die dabei explizit auftretenden Eingänge aufgelistet.

Tabelle 2 – Relative Grade der Ausgänge und explizites Auftreten der Eingänge.

Index i	1	2	3	4
Komponente des vektoriellen relativen Grades r_i	2	2	2	2
explizites Auftreten von τ_j bei $y_i^{(r_i)}$	τ_3, τ_4	τ_3, τ_4	τ_3, τ_4	τ_1, τ_3
$y_j^{(k)}$ mit minimalem k , bei dem τ_i zuerst auftritt	$y_4^{(2)}$	$y_{1,2,3}^{(4)}$	$y_{1,2,3,4}^{(2)}$	$y_{1,2,3}^{(2)}$

Das Fehlen von τ_2 bei all diesen Ausgangsableitungen bedeutet für die Entkopplungsmatrix $\mathbf{\Lambda}$ gemäß Gleichung (5.4) eine Singularität, weil ihre Spalte nur Nulleinträge enthält:

$$\mathbf{\Lambda} = \begin{pmatrix} 0 & 0 & \frac{p_1 - q_1 - s_2 \cos(p_3)}{m_2 \sqrt{(p_2 - s_2 \sin(p_3))^2 + (-p_1 + q_1 + s_2 \cos(p_3))^2}} & \frac{-l_0 + p_1 - q_2 + s_2 \cos(p_3)}{m_2 \sqrt{(p_2 + s_2 \sin(p_3))^2 + (l_0 - p_1 + q_2 - s_2 \cos(p_3))^2}} \\ 0 & 0 & \frac{p_2 - s_2 \sin(p_3)}{m_2 \sqrt{(p_2 - s_2 \sin(p_3))^2 + (-p_1 + q_1 + s_2 \cos(p_3))^2}} & \frac{p_2 + s_2 \sin(p_3)}{m_2 \sqrt{(p_2 + s_2 \sin(p_3))^2 + (l_0 - p_1 + q_2 - s_2 \cos(p_3))^2}} \\ 0 & 0 & \frac{s_2(p_1 \sin(p_3) - p_2 \cos(p_3) - q_1 \sin(p_3))}{J_2 \sqrt{(p_2 - s_2 \sin(p_3))^2 + (-p_1 + q_1 + s_2 \cos(p_3))^2}} & \frac{s_2(l_0 \sin(p_3) - p_1 \sin(p_3) + p_2 \cos(p_3) + q_2 \sin(p_3))}{J_2 \sqrt{(p_2 + s_2 \sin(p_3))^2 + (l_0 - p_1 + q_2 - s_2 \cos(p_3))^2}} \\ \frac{1}{m_1} & 0 & \frac{-p_1 + q_1 + s_2 \cos(p_3)}{m_1 \sqrt{(p_2 - s_2 \sin(p_3))^2 + (-p_1 + q_1 + s_2 \cos(p_3))^2}} & 0 \end{pmatrix}. \quad (5.8)$$

Der einzige von Null verschiedene Eintrag in der zweiten Spalte \mathbf{g}_2 der Eingangsmatrix \mathbf{g} erfolgt nämlich durch $g_{10,2} = \frac{1}{m_3}$ und die Lie-Ableitungen $L_{\mathbf{f}}y_i = \dot{y}_i$ geben nur die zeitliche Ableitung des flachen Ausgangs $\dot{\mathbf{y}} = (\dot{p}_1, \dot{p}_2, \dot{p}_3, \dot{q}_1)^T$ wieder. So folgt für das relevante Produkt $\frac{\partial}{\partial \dot{q}_2} L_{\mathbf{f}}y_i \cdot g_{10,2} = \frac{\partial}{\partial \dot{q}_2} \dot{y}_i \cdot g_{10,2} = 0$ für $i = 1, \dots, 4$, weil $\dot{q}_2 \notin \dot{\mathbf{y}}$. Allgemeiner gilt, dass ein Eingang τ_j in den Komponenten der Ausgangsableitung $y_i^{(r_i)}$ genau dann nicht explizit auftritt, also die j -te Spalte von $\mathbf{\Lambda}$ eine Nullspalte ist, wenn für alle zugehörigen gemischten Lie-Ableitungen $L_{\mathbf{g}_j} L_{\mathbf{f}}^{r_i-1} y_i = 0$ mit $i = 1, \dots, m$ gilt [11, S. 201]

Analog gilt dies für den alternativen flachen Ausgang $\tilde{\mathbf{y}} = (p_1, p_2, p_3, q_2)^T$ aufgrund des Fehlens von τ_1 , welches eine erste Nullspalte von $\mathbf{\Lambda}$ impliziert.

Daher ist die Entkopplungsmatrix $\mathbf{\Lambda}$ in dieser Form nie regulär und der vektorielle relative Grad \mathbf{r} nie wohldefiniert. Dieses System ist nicht statisch eingangs-ausgangs-linearisierbar. Stattdessen wird zur Regelung des Doppelkransystems ein Ansatz mit einer dynamischen Erweiterung oder quasi-statischen Zustandsrückführung verfolgt.

5.2.3 Zustandsrückführung aus Fehlerdynamik

Nach [11, S. 195] kann eine lineare Fehlerdynamik bezüglich des Trajektorienfolgefehlers $e_i := y_i - y_{i,\text{ref}}$ mit dem Messwert bzw. simuliertem Wert y_i und dem Wert der Referenztrajektorie $y_{i,\text{ref}} = \text{func}(t)$. In der Literatur wird an dieser Stelle lediglich die Abweichung bezüglich eines Festwerts $y_{i,\text{ref}} = \text{const.}$ angesetzt. Damit können auch die Komponenten des virtuellen Eingangs gewählt werden:

$$e_i^{(r_i)} + c_{i,r_i-1} e_i^{(r_i-1)} + \dots + c_{i,1} \dot{e}_i + c_{i,0} e_i = 0 \quad (5.9)$$

$$\Leftrightarrow v_i = y_i^{(r_i)} = y_{i,\text{ref}}^{(r_i)} - c_{i,r_i-1} e_i^{(r_i-1)} - \dots - c_{i,1} \dot{e}_i - c_{i,0} e_i. \quad (5.10)$$

Die Koeffizienten $c_{i,k}$, $k \in 0, 1, \dots, r_i - 1$ sind für eine Stabilisierung des Systems so zu wählen, dass das charakteristische Polynom dieser Gleichung ausschließlich Polstellen mit negativem Realteil aufweist.

5.2.4 Dynamische Erweiterung

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [18, flatness_notebooks/ODE_flatness_trajectory_control_simulation_dyn.ipynb] zu finden.

Die Entkopplungsmatrix $\mathbf{\Lambda}$ hat eine Nullspalte, weil τ_2 in keiner Ableitung der Ausgangskomponenten $y_i^{(r_i)}$ explizit auftritt. Somit gilt $\text{rang } \mathbf{\Lambda} =: k = 3 < m = 4$. Es gilt für eine reguläre Matrix $\tilde{\mathbf{\Lambda}}$ zu verhindern, dass die drei sonstigen Eingangskomponenten τ_1 , τ_3 und τ_4 in allen Ausgangsableitungen frühzeitig zur Ableitungsordnung 2 auftreten.

Tabelle 2 kann entnommen werden, dass τ_2 das erste Mal in einer der Ausgangsableitungen der Ordnung 4 von y_1 , y_2 oder y_3 auftritt. Daher ist eine Ergänzung aller $k = 3$ Eingangskomponenten außer τ_2 um jeweils zwei Integratoren vorzunehmen. [11, S. 200]

So treten die neuen Eingangskomponenten ebenso wie τ_2 das erste mal bei der Ableitungsordnung 4 explizit auf. Die Systemgleichungen werden um die folgenden DGL dynamisch erweitert:

$$\begin{aligned}\dot{\tau}_1 &=: \alpha_1 \\ \dot{\tau}_3 &=: \alpha_3 \\ \dot{\tau}_4 &=: \alpha_4 \\ \dot{\alpha}_1 &=: \beta_1 \\ \dot{\alpha}_3 &=: \beta_3 \\ \dot{\alpha}_4 &=: \beta_4.\end{aligned}\tag{5.11}$$

Daraus folgt die Definition eines neuen Eingangsvektors unter Umsortierung der Komponente τ_2 an die letzte Stelle $\tilde{\tau} = (\beta_1, \beta_3, \beta_4, \tau_2)^T$. Der Zustandsvektor wird auf $\tilde{\mathbf{x}} = (\mathbf{x}, \tau_1, \tau_3, \tau_4, \alpha_1, \alpha_3, \alpha_4)^T$ erweitert. Die Ausgangsableitungen werden erneut gemäß (5.7) rekursiv über Lie-Ableitungen entlang der erweiterten Zustandsgleichungen $\tilde{\delta} = \tilde{\mathbf{f}} + \tilde{\mathbf{g}}\tilde{\tau}$ mit

$$\tilde{\delta} = \begin{pmatrix} \delta \\ \alpha_1 \\ \alpha_3 \\ \alpha_4 \\ \beta_1 \\ \beta_3 \\ \beta_4 \end{pmatrix}, \quad \tilde{\mathbf{f}} = \tilde{\delta}|_{\tilde{\tau}=0}, \quad \tilde{\mathbf{g}} = \frac{\partial \tilde{\delta}}{\partial \tilde{\tau}}\tag{5.12}$$

gebildet. Eine alternative Berechnungsvorschrift zur Bestätigung des vorangegangenen Vorgehens ist für die höchsten Zeitableitungen der Ausgangskomponenten $y_i^{(r_i)}$ gemäß [11, S. 195] möglich:

$$y_i^{(r_i)} = L_{\tilde{\mathbf{f}}}^{r_i} y_i + L_{\tilde{\mathbf{g}}_1} L_{\tilde{\mathbf{f}}}^{r_i-1} y_i \beta_1 + L_{\tilde{\mathbf{g}}_2} L_{\tilde{\mathbf{f}}}^{r_i-1} y_i \beta_3 + L_{\tilde{\mathbf{g}}_3} L_{\tilde{\mathbf{f}}}^{r_i-1} y_i \beta_4 + L_{\tilde{\mathbf{g}}_4} L_{\tilde{\mathbf{f}}}^{r_i-1} y_i \tau_2.\tag{5.13}$$

Für dieses modifizierte System folgt somit $r_i = 4$ für $i = 1, \dots, 4$. Dieser relative Grad ist für eine Auswahl prototypischer Referenztrajektorien wohldefiniert, da Bedingung 1 aus Abschnitt 5.2.1 erfüllt ist und die Inverse der Entkopplungsmatrix stets existiert, $\mathbf{\Lambda}$ also regulär ist. Dementsprechend ist eine Zustandsrückführung nach Gleichung (5.6) möglich. Diese wird als dynamische Rückführung bezeichnet.

Prototypische Trajektorien beim Übergang zwischen zwei Ruhelagen mit einem auszuregelnden Anfangsfehler sind in Abbildung 6 dargestellt. Die Zwiebelchalendiagramme

in Abbildung 7 zeigen eine geometrische Veranschaulichung des selben Sachverhalts. Dabei ist die Bewegung der vier Gelenke sowie der beiden Seile und der Last dargestellt. Für spätere Simulationszeitpunkte sind diese Momentaufnahmen der Konfiguration des Doppelkransystems dunkler dargestellt.

Ein Nachteil der dynamischen Rückführung ist die höhere Trägheit des Systems beim Ausregeln von Abweichungen von der Solltrajektorie, welche durch die zusätzlichen Integratoren vor den Systemeingängen hervorgerufen werden. So ist es notwendig die Pole der Fehlerdynamik für ein schnelles Ausgleichen von Anfangsfehlern relativ weit links in der komplexen Ebene zu platzieren. Zudem ergeben sich sehr lange Simulationsdauern, welche auch aus den relativ komplexen Stellgesetzen mit mehr als 20 000 Operationen je Eingangsgröße folgen.

Auf die konkrete symbolische Angabe der Stellgesetze des Systemeingangs wird aus Praktikabilitätsgründen verzichtet und stattdessen auf das zu Beginn dieses Teilausschnitts 5.2.4 erwähnte Jupyter-Notebook verwiesen. Übersichtlicher ist an dieser Stelle eine Angabe von funktionalen Abhängigkeiten:

$$\beta_1 = \text{func}(\mathbf{x}, \tau_1, \tau_3, \tau_4, \alpha_1, \alpha_3, \alpha_4, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}, \mathbf{y}_{\text{ref}}^{(3)}, \mathbf{y}_{\text{ref}}^{(4)}) \quad (5.14a)$$

$$\beta_3 = \text{func}(\mathbf{x}, \tau_1, \tau_3, \tau_4, \alpha_3, \alpha_4, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}, \mathbf{y}_{\text{ref}}^{(3)}, \mathbf{y}_{\text{ref}}^{(4)}) \quad (5.14b)$$

$$\beta_4 = \text{func}(\mathbf{x}, \tau_1, \tau_3, \tau_4, \alpha_3, \alpha_4, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}, \mathbf{y}_{\text{ref}}^{(3)}, \mathbf{y}_{\text{ref}}^{(4)}) \quad (5.14c)$$

$$\tau_2 = \text{func}(\mathbf{x}, \tau_1, \tau_3, \tau_4, \alpha_3, \alpha_4, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}, \mathbf{y}_{\text{ref}}^{(3)}, \mathbf{y}_{\text{ref}}^{(4)}). \quad (5.14d)$$

Anpassung der Trajektorienplanung

Durch die Einführung von jeweils zwei Integratoren nach den neuen Stellgrößen β_1 , β_3 und β_4 ist an beide Rändern jeder Referenztrajektorie der flachen Ausgangskomponenten y_i auch eine um jeweils zwei Ordnungen höhere Forderung an die stetige Differenzierbarkeit zu stellen. Das Vorgehen nach Abschnitt 5.1 kann erneut analog durchgeführt werden mit jeweils um vier Ordnungen höheren Polynomen der Ausgangstrajektorien in den Gleichungen (5.2).

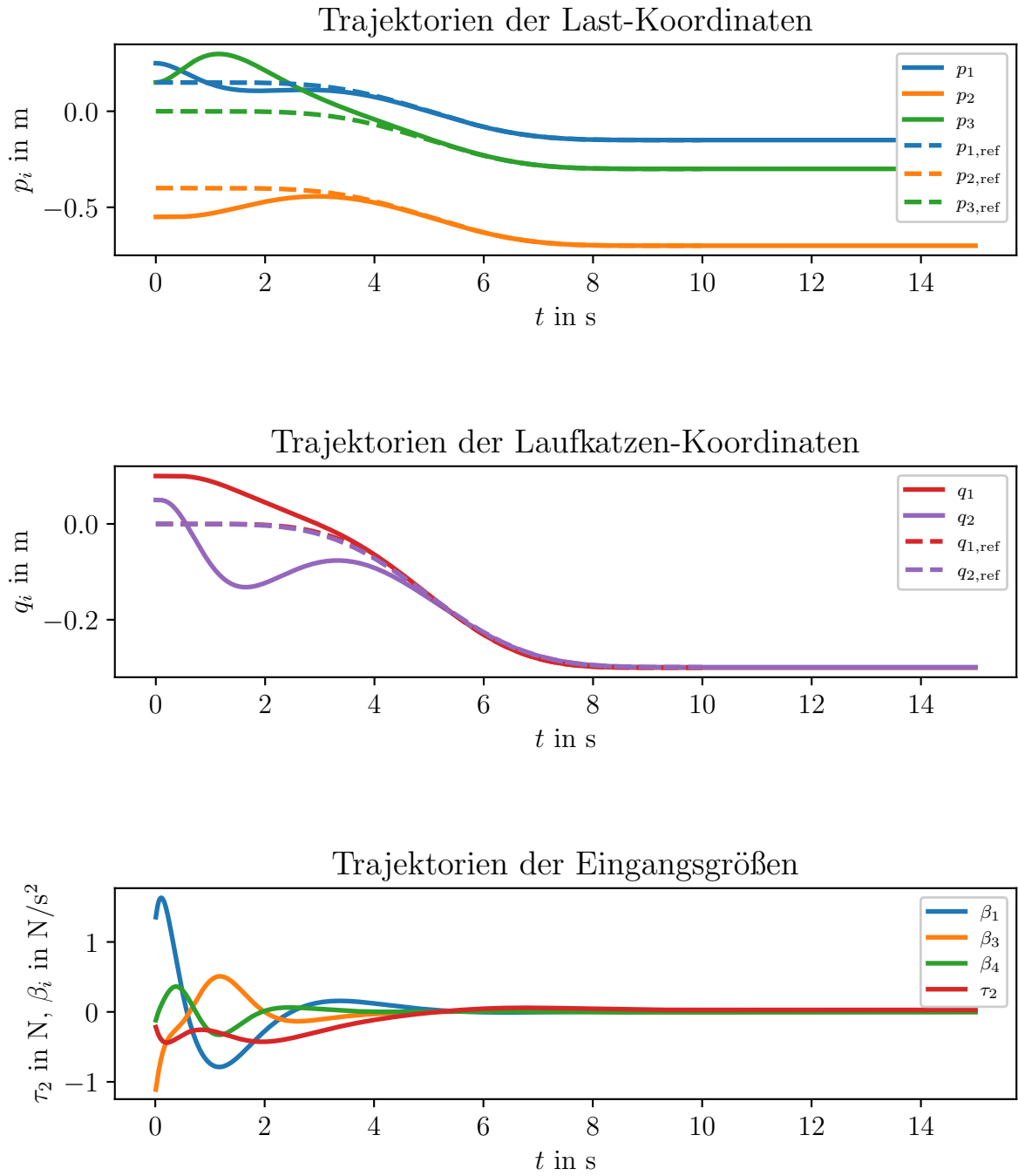


Abbildung 6 – Prototypische Trajektorien des Doppelkransystems bei der Überführung zwischen zwei Ruhelagen unter der Ausregelung von Anfangsfehlern über die dynamische Erweiterung. Koordinatenbezeichnung entsprechend Abbildung 5.

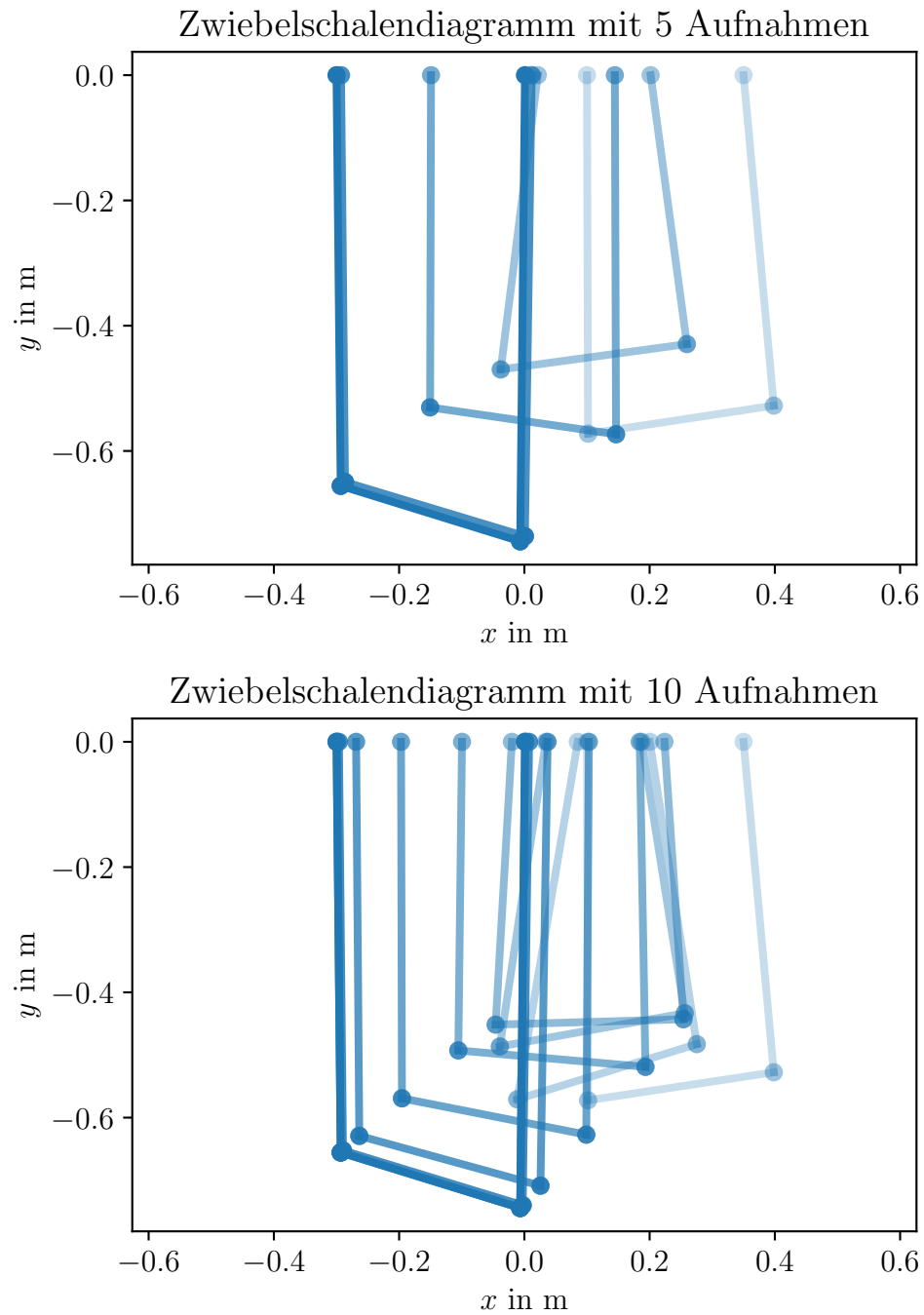


Abbildung 7 – Zwiebelschalendiagramme der Bewegung der vier Gelenke sowie der beiden Seile und der Last für prototypische Trajektorien des Doppelkransystems bei der Überführung zwischen zwei Ruhelagen unter der Ausregelung von Anfangsfehlern über die dynamische Erweiterung.

5.2.5 Quasi-statische Rückführungen

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [18, flatness_notebooks/ODE_flatness_trajectory_control_simulation_qstat.ipynb] zu finden.

Ebenso wie die dynamische Erweiterung kann eine quasi-statische Zustandsrückführung bei einem nicht wohldefinierten relativen vektoriellen Grad im statischen Ansatz zur Stabilisierung des Systems angesetzt werden. Dafür wird das Vorgehen nach [11, S. 206] im Folgenden nachvollzogen und am Doppelkransystem angewendet. Tabelle 3 stellt für jede Eingangskomponente die Ableitungsordnung jeder Ausgangskomponente von \mathbf{y} gegenüber.

Tabelle 3 – Auftreten der Eingänge τ_j bei Ableitungsordnung k der Ausgänge y_i .

k	τ_1	τ_2	τ_3	τ_4
$y_1^{(k)}$	4	4	2	2
$y_2^{(k)}$	4	4	2	2
$y_3^{(k)}$	4	4	2	2
$y_4^{(k)}$	2	>4	2	4
$(y_5^{(k)} = q_2^{(k)})$	>4	2	4	2

Erneut wird o. B. d. A. der flache Ausgang $\mathbf{y} = (y_1, y_2, y_3, y_4)^T = (p_1, p_2, p_3, q_1)^T$ betrachtet. Die zeitlichen Ausgangsableitungen können wieder rekursiv gemäß Gleichung (5.7) mittels Lie-Ableitungen bezüglich des hierfür erweiterten Zustandsvektors $\tilde{\mathbf{x}} = (\mathbf{x}, \boldsymbol{\tau}, \dot{\boldsymbol{\tau}})^T$ entlang des Vektorfelds der Zustandsgleichungen $\tilde{\boldsymbol{\delta}} = \tilde{\mathbf{f}} + \tilde{\mathbf{g}}\tilde{\boldsymbol{\tau}}$ mit

$$\tilde{\boldsymbol{\delta}} = \begin{pmatrix} \mathbf{f} + \mathbf{g}\boldsymbol{\tau} \\ \dot{\boldsymbol{\tau}} \\ \ddot{\boldsymbol{\tau}} \end{pmatrix} \quad (5.15)$$

generiert werden.

Die Eingangskomponente τ_3 tritt zuerst bei $r_1 = 2$ in \ddot{y}_1 sowie auch in allen sonstigen zweiten Ableitungen des flachen Ausgangs auf. Dies gilt nicht für alle Eingangskomponenten, da der relative vektorielle Grad für $r_i = 2$ für $i = 1, \dots, 4$ nicht wohldefiniert ist. Außerdem tritt τ_4 auch bei $r_3 = 2$ in \ddot{y}_1 , \ddot{y}_2 und \ddot{y}_3 , nicht aber in \ddot{y}_4 auf. Dies entspricht nicht der in [11, S. 206] dargestellten Form, bei der für das Auftreten von u_2 der Ausgang y_2 noch weiter differenziert werden muss. Die Eingangsgröße τ_2 erscheint das erste Mal explizit bei $r_i = 4$ in $y_1^{(4)}$, $y_2^{(4)}$ oder $y_3^{(4)}$, τ_1 zuerst bei $r_4 = 2$ in \ddot{y}_4 .

Eine Definition der neuen virtuellen Eingänge v_i ist so möglich:

$$\begin{aligned}
 \ddot{y}_1 &= \ddot{p}_1 =: v_1 = \text{func}(\mathbf{x}, \tau_3, \tau_4) \\
 y_2^{(4)} &= p_2^{(4)} =: v_2 = \text{func}(\mathbf{x}, \tau_1, \tau_2, \tau_3, \tau_4, \dot{\tau}_3, \dot{\tau}_4, \ddot{\tau}_3, \ddot{\tau}_4) \\
 \ddot{y}_3 &= \ddot{p}_3 =: v_3 = \text{func}(\mathbf{x}, \tau_3, \tau_4) \\
 \ddot{y}_4 &= \ddot{q}_1 =: v_4 = \text{func}(\mathbf{x}, \tau_1, \tau_3).
 \end{aligned} \tag{5.16}$$

Daraus ist ersichtlich, dass durch ein Gleichungssystem aus v_1 , v_3 und v_4 , in denen die Eingangskomponenten τ_1 , τ_3 und τ_4 affin auftreten, eine explizite Darstellung dieser Komponenten möglich ist. Dies steht im Gegensatz zur Situation in [11, S. 207], bei der die relativen Grade der Ausgangskomponenten aufsteigend sind. Dabei muss kein Gleichungssystem gelöst werden, sondern es genügt das sukzessives Einsetzen der vorher berechneten Eingänge und die algebraische Umformung der abgeleiteten Ausgänge.

Die Berechnung von τ_2 erfordert einen größeren Aufwand als bei den anderen Eingangskomponenten. Dabei ist die Substitution bisheriger Eingangskomponenten und deren Ableitungen notwendig, welche analog zu Gleichung (5.7) aus den Lie-Ableitungen entlang $\tilde{\delta}$ generiert werden können. Es folgt ein relativ umfangreicher Ausdruck für τ_2 , der zu diesem Zeitpunkt mehr als einhunderttausend Rechenoperationen enthält.

Nach [11, S. 208] können lineare Fehlerdynamiken und daraus \mathbf{v} entsprechend der jeweiligen relativen Grade der Ausgangskomponenten zur Stabilisierung des Systems angesetzt werden. Die Komponenten des virtuellen Eingangs v_1 , v_3 und v_4 sind jeweils nur von den Zustandskomponenten $\tilde{\mathbf{x}}$ abhängig. Die Komponente v_2 hängt allerdings auch von den Ableitungen $\ddot{y}_2 = \ddot{p}_2$ und $y_2^{(3)} = p_2^{(3)}$ ab, welche durch Lie-Ableitungen berechnet werden können:

$$\begin{aligned}
 \ddot{p}_2 &= \ddot{y}_2 = L_{\tilde{\delta}} \dot{y}_2 = L_{\tilde{\delta}} \dot{p}_2 \\
 p_2^{(3)} &= y_2^{(3)} = L_{\tilde{\delta}} \ddot{p}_2.
 \end{aligned} \tag{5.17}$$

Außerdem werden in der Darstellung von τ_2 Ableitungen der Eingangskomponenten v_1 ,

v_3 und v_4 benötigt, die aus deren Fehlerdynamiken zweiter Ordnung folgen:

$$\begin{aligned}
\ddot{e}_i + c_{i,1}\dot{e}_i + c_{i,0}e_i &= 0 \\
e_i^{(3)} + c_{i,1}\ddot{e}_i + c_{i,0}\dot{e}_i &= 0 \\
e_i^{(3)} + c_{i,1}(-c_{i,1}\dot{e}_i - c_{i,0}e_i) + c_{i,0}\dot{e}_i &= 0 \\
e_i^{(3)} - c_{i,1}^2\dot{e}_i + c_{i,0}\dot{e}_i - c_{i,0}c_{i,1}e_i &= 0 \\
e_i^{(3)} + (c_{i,0} - c_{i,1}^2)\dot{e}_i - c_{i,0}c_{i,1}e_i &= 0 \\
e_i^{(4)} + (c_{i,0} - c_{i,1}^2)\ddot{e}_i - c_{i,0}c_{i,1}\dot{e}_i &= 0 \\
e_i^{(4)} + (c_{i,0} - c_{i,1}^2)(-c_{i,1}\dot{e}_i - c_{i,0}e_i) - c_{i,0}c_{i,1}\dot{e}_i &= 0 \\
e_i^{(4)} + (c_{i,1}^3 - 2c_{i,0}c_{i,1})\dot{e}_i + (c_{i,0}c_{i,1}^2 - c_{i,0}^2)e_i &= 0 \\
\Rightarrow \text{für } i = 1, 3, 4 : \dot{v}_i = y_i^{(3)} = y_{i,\text{ref}}^{(3)} - (c_{i,0} - c_{i,1}^2)\dot{e}_i + c_{i,0}c_{i,1}e_i \\
\ddot{v}_i = y_i^{(4)} = y_{i,\text{ref}}^{(4)} - (c_{i,1}^3 - 2c_{i,0}c_{i,1})\dot{e}_i - (c_{i,0}c_{i,1}^2 - c_{i,0}^2)e_i.
\end{aligned} \tag{5.18}$$

Die Komponente v_2 ergibt sich nach einem Ansatz vierter Ordnung unter Einbeziehung des Zusammenhangs (5.17):

$$v_2 = y_2^{(4)} = y_{2,\text{ref}}^{(4)} - c_{2,3}(y_2^{(3)} - y_{2,\text{ref}}^{(3)}) - c_{2,2}(\ddot{y}_2 - \ddot{y}_{2,\text{ref}}) - c_{2,1}(\dot{y}_2 - \dot{y}_{2,\text{ref}}) - c_{2,0}(y_2 - y_{2,\text{ref}}) \tag{5.19}$$

Auf die konkrete symbolische Angabe der Stellgesetze wird aus Praktikabilitätsgründen verzichtet und stattdessen auf das zu Beginn dieses Abschnitts 5.2.5 erwähnte Jupyter-Notebook verwiesen. Übersichtlicher ist an dieser Stelle eine Angabe von funktionalen Abhängigkeiten:

$$\tau_1 = \text{func}(\mathbf{x}, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}) \tag{5.20a}$$

$$\tau_2 = \text{func}(\mathbf{x}, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}, \mathbf{y}_{\text{ref}}^{(3)}, \mathbf{y}_{\text{ref}}^{(4)}) \tag{5.20b}$$

$$\tau_3 = \text{func}(\mathbf{x}, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}) \tag{5.20c}$$

$$\tau_4 = \text{func}(\mathbf{x}, \mathbf{y}_{\text{ref}}, \dot{\mathbf{y}}_{\text{ref}}, \ddot{\mathbf{y}}_{\text{ref}}). \tag{5.20d}$$

Bei der simulativen Untersuchung dieses Ansatzes ergibt sich bereits im reinen Vorsteuerungsfall, also ohne Anfangsfehler oder Störungen des Systems, ein Problem mit Singularitäten in den berechneten Stellgrößen. Wenn im Stellgesetz allerdings nicht alle Zustandskomponenten \mathbf{x} vorkommen, sondern nur solche, welche auch Teil des flachen Ausgangs \mathbf{y} und seiner Ableitungen sind, treten keine Singularitäten auf. Eine Simulation dieses Regelungsansatzes ist daher erschwert und kann erst weiter verfolgt werden, wenn dieses Problem gelöst ist.

Singularitäten in der Nähe von Ruhelagen

In allen Eingangsgrößen treten bei diesem quasi-statischen Rückführungsentwurf in der Nähe von Ruhelagen Singularitäten auf. Diese ergeben sich aufgrund numerischer Effekte. Beim Einsetzen einer zuvor ermittelten Ruhelage entstehen Definitionslücken: Sowohl der Zähler als auch der Nenner der Eingangsgrößen τ werden Null. Es ist nicht auszuschließen, dass diese Lücken durch algebraische Manipulation hebbbar sind. Allerdings ist eine weitere händische Untersuchung dieses Zusammenhangs mit hohem Aufwand verbunden und wird im Rahmen dieser Arbeit nicht weiter verfolgt. Die CAS-Bibliothek SymPy bietet mittels des Aufrufs `simplify` allerdings keine Lösung dieses Problems in den Eingangsgrößen. Für eine weitere Analyse dieses Problems eignen sich die einfacheren Terme des Nenners $N_{1,3,4}$ von τ_1 , τ_3 und τ_4 :

$$\begin{aligned} N_{1,3,4} = & s_2(-4l_0pm_1 \sin(pm_3) + 2l_0pm_2 \cos(pm_3) + 4l_0qm_1 \sin(pm_3) + l_0s_2 \sin(2pm_3) \\ & + 4pm_1^2 \sin(pm_3) - 4pm_1pm_2 \cos(pm_3) - 4pm_1qm_1 \sin(pm_3) - 4pm_1qm_2 \sin(pm_3) \\ & + 2pm_2qm_1 \cos(pm_3) + 2pm_2qm_2 \cos(pm_3) + 4qm_1qm_2 \sin(pm_3) - qm_1s_2 \sin(2pm_3) \\ & + qm_2s_2 \sin(2pm_3)). \end{aligned} \quad (5.21)$$

Des Weiteren ist die Fragestellung von Interesse, weshalb sich solche Lücken bei der Konstruktion der Trajektorie in den Ruhelagen ergeben.

5.2.6 Exact feedforward linearization

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [\[18, flatness_notebooks/ODE_flatness_trajectory_control_simulation.ipynb\]](#) zu finden.

Bei der exact feedforward linearization wird statt einer Kompensation der Nichtlinearität mittels Rückführung der gemessenen beziehungsweise simulierten Zustandskomponenten \mathbf{x} wie bei der exakten Eingangs-Ausgangs-Linearisierung die Referenztrajektorie \mathbf{x}_{ref} eingesetzt [\[6\]](#).

Heuristisch wird eine Fehlerdynamik der Ordnung zwei für alle Komponenten des Koordinatenvektors $\boldsymbol{\theta} = (p_1, p_2, p_3, q_1, q_2)^T$ (nicht nur solchen, die auch Teil des flachen Ausgangs sind!) angesetzt:

$$\ddot{\boldsymbol{\theta}} = \ddot{\boldsymbol{\theta}}_{\text{ref}} - \mathbf{c}_1^T \dot{\mathbf{e}} - \mathbf{c}_0^T \mathbf{e}. \quad (5.22)$$

Aus der Zusammensetzung des Zustandsvektors $\mathbf{x} = (\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})^T$ und der eingangsaffinen Zustandsraumdarstellung in Gleichung [\(5.3\)](#) kann außerdem der Zusammenhang

$$\ddot{\boldsymbol{\theta}} = \mathbf{f}_{[6,10]}(\boldsymbol{\theta}) + \mathbf{g}_{[6,10]}(\boldsymbol{\theta}) \boldsymbol{\tau} \quad (5.23)$$

hergestellt werden. Dabei bedeutet die Indizierung $\bullet_{[i,j]}$, die Auswahl der Zeilen i bis j von \bullet . Da für den Systemeingang $\boldsymbol{\tau} \in \mathbb{R}^4$ und für die Eingangsmatrix $\mathbf{g}_{[6,10]} \in \mathbb{R}^{5 \times 4}$ gilt, kann dieser Zusammenhang über die Bildung einer Pseudo-Inversen $\mathbf{g}_{[6,10]}^+$ nach dem Eingangsvektor aufgelöst werden:

$$\boldsymbol{\tau} = \mathbf{g}_{[6,10]}^+(\boldsymbol{\theta}) (\ddot{\boldsymbol{\theta}}_{\text{ref}} - \mathbf{c}_1 \dot{\boldsymbol{\theta}} - \mathbf{c}_0 \boldsymbol{\theta} - \mathbf{f}_{[6,10]}(\boldsymbol{\theta})). \quad (5.24)$$

Durch Einsetzen der Referenztrajektorien der Koordinaten $\boldsymbol{\theta}_{\text{ref}}$ in $\mathbf{g}_{[6,10]}^+(\boldsymbol{\theta})$ und $\mathbf{f}_{[6,10]}(\boldsymbol{\theta})$ wird die exact feedforward linearization realisiert:

$$\boldsymbol{\tau} = \mathbf{g}_{[6,10]}^+(\boldsymbol{\theta}_{\text{ref}}) (\ddot{\boldsymbol{\theta}}_{\text{ref}} - \mathbf{c}_1 \dot{\boldsymbol{\theta}} - \mathbf{c}_0 \boldsymbol{\theta} - \mathbf{f}_{[6,10]}(\boldsymbol{\theta}_{\text{ref}})). \quad (5.25)$$

Abbildung 8 zeigt prototypische Trajektorien beim Übergang zwischen zwei Ruhelagen mit einem auszuregelnden Anfangsfehler für diesen Regelungsansatz.

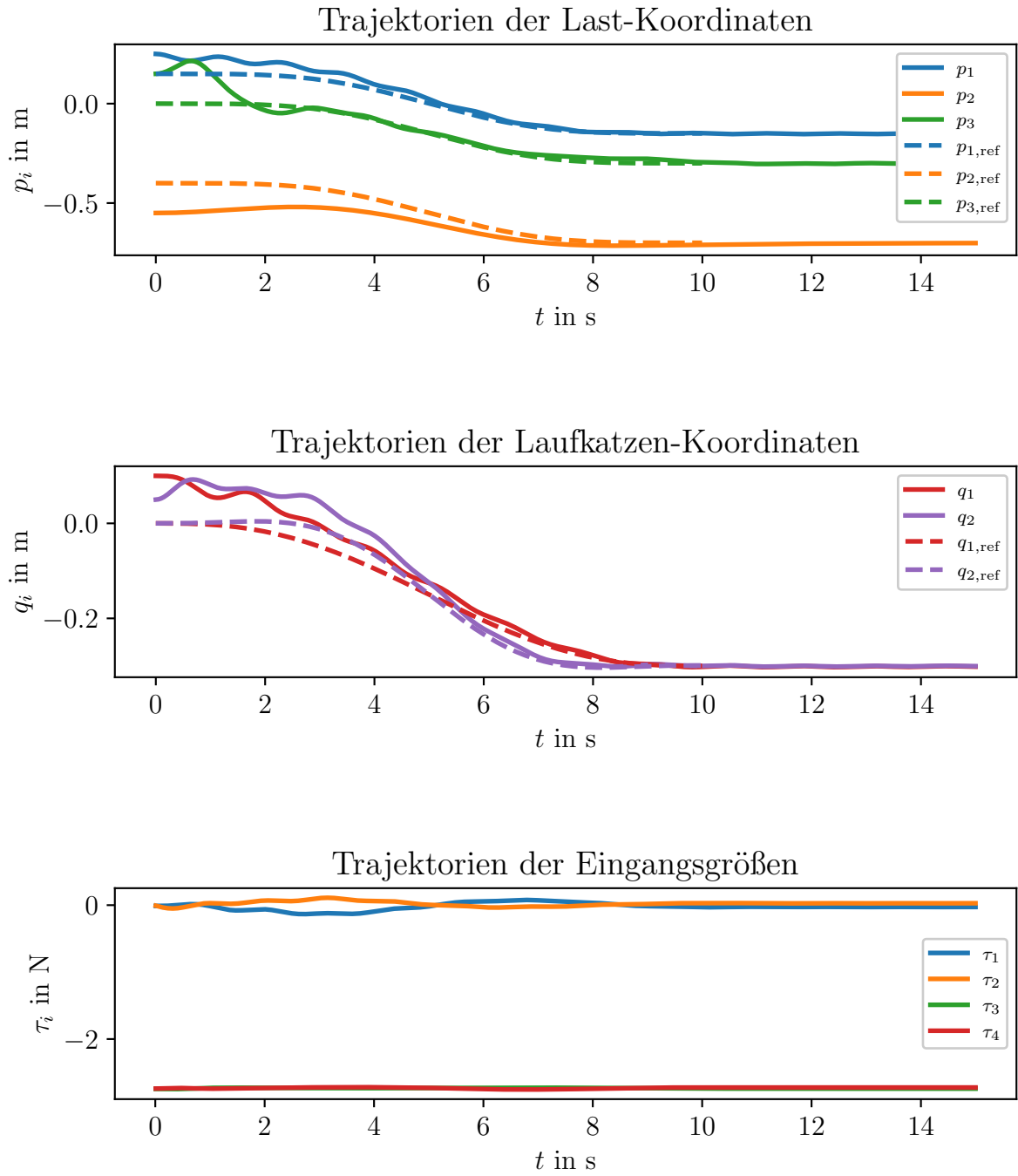


Abbildung 8 – Prototypische Trajektorien des Doppelkransystems bei der Überführung zwischen zwei Ruhelagen unter der Ausregelung von Anfangsfehlern über die exact feedforward linearization mit Pseudoinverser. Koordinatenbezeichnung entsprechend Abbildung 5.

Vereinfachung mittels Ausgangsselektion

Statt der Nutzung einer Pseudo-Inversen wie in Gleichung (5.25) wird im Folgenden eine Selektion von vier der fünf Gleichungen der Beschleunigungen $\ddot{\boldsymbol{\theta}}$ vorgenommen. Symbolisch wird dies über eine Selektionsmatrix \mathbf{S} dargestellt:

$$\mathbf{S} \cdot \ddot{\boldsymbol{\theta}} = \mathbf{S} \cdot (\ddot{\boldsymbol{\theta}}_{\text{ref}} - \mathbf{c}_1^T \dot{\mathbf{e}} - \mathbf{c}_0^T \mathbf{e}) = \mathbf{S} \cdot \mathbf{f}_{[6,10]}(\boldsymbol{\theta}) + \mathbf{S} \cdot \mathbf{g}_{[6,10]}(\boldsymbol{\theta}) \boldsymbol{\tau}. \quad (5.26)$$

Durch die Wahl der letzten vier Gleichungen kann eine direkte Inversion der somit quadratischen Eingangsmatrix $\mathbf{g}_{[7,10]} = \mathbf{S} \cdot \mathbf{g}_{[6,10]}$ erfolgen:

$$\boldsymbol{\tau} = \mathbf{g}_{[7,10]}^{-1}(\boldsymbol{\theta}_{\text{ref}}) \cdot \mathbf{S} \cdot (\ddot{\boldsymbol{\theta}}_{\text{ref}} - \mathbf{c}_1 \dot{\mathbf{e}} - \mathbf{c}_0 \mathbf{e} - \mathbf{f}_{[6,10]}(\boldsymbol{\theta}_{\text{ref}})) \text{ mit } \mathbf{S} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (5.27)$$

Diese Berechnungsvorschrift der Stellgrößen enthält deutlich weniger Operationen als alle zuvor dargestellten Ansätze. Damit eignet sie sich insbesondere für eine spätere Implementierung auf einem Echtzeitsystemen sowie sehr viel kürzeren Simulationszeiten. Abbildung 10 zeigt, dass sich für Beispieltrajektorien mit Anfangsfehlern ebenso wie bei den vorherigen Ansätzen ein sehr gutes Folgeverhalten sowie stationäre Genauigkeit ergibt.

Auf die Darstellung konkreter symbolischer Ausdrücke für die Stellgesetze wird aus Praktikabilitätsgründen dennoch verzichtet, da jede Eingangskomponente trotzdem mehr als einhundert Rechenoperationen beinhaltet. Stattdessen wird auf das zu Beginn dieses Abschnitts 5.2.6 erwähnte Jupyter-Notebook verwiesen. Übersichtlicher ist an dieser Stelle eine Angabe von funktionalen Abhängigkeiten:

$$\tau_1 = \text{func}(\mathbf{x}, \boldsymbol{\theta}_{\text{ref}}, \dot{\boldsymbol{\theta}}_{\text{ref}}, \ddot{\boldsymbol{\theta}}_{\text{ref}}) \quad (5.28a)$$

$$\tau_2 = \text{func}(\mathbf{x}, \boldsymbol{\theta}_{\text{ref}}, \dot{\boldsymbol{\theta}}_{\text{ref}}, \ddot{\boldsymbol{\theta}}_{\text{ref}}) \quad (5.28b)$$

$$\tau_3 = \text{func}(\mathbf{x}, \boldsymbol{\theta}_{\text{ref}}, \dot{\boldsymbol{\theta}}_{\text{ref}}, \ddot{\boldsymbol{\theta}}_{\text{ref}}) \quad (5.28c)$$

$$\tau_4 = \text{func}(\mathbf{x}, \boldsymbol{\theta}_{\text{ref}}, \dot{\boldsymbol{\theta}}_{\text{ref}}, \ddot{\boldsymbol{\theta}}_{\text{ref}}). \quad (5.28d)$$

Stabilitätsbetrachtung

Im Folgenden wird die Simulation einer prototypischen Referenztrajektorie mit dem zuletzt vorgestellten Regelungsansatz betrachtet. Die Eigenwerte der Jacobimatrix $\mathbf{J}_{\dot{\mathbf{e}}}$ einer Fehlerdynamik

$$\mathbf{e} := \mathbf{x} - \mathbf{x}_{\text{ref}} \Leftrightarrow \mathbf{x} = \mathbf{e} + \mathbf{x}_{\text{ref}} \quad (5.29)$$

$$\dot{\mathbf{e}} = \dot{\mathbf{x}} - \dot{\mathbf{x}}_{\text{ref}} = \mathbf{f}(\mathbf{e} + \mathbf{x}_{\text{ref}}) + \mathbf{g}(\mathbf{e} + \mathbf{x}_{\text{ref}}) \boldsymbol{\tau} - \dot{\mathbf{x}}_{\text{ref}} \quad (5.30)$$

$$\mathbf{J}_{\dot{\mathbf{e}}} = \frac{\partial \dot{\mathbf{e}}}{\partial \mathbf{e}} \quad (5.31)$$

sind für eine Stabilitätsbetrachtung von Interesse. Abbildung 9 zeigt, dass diese Eigenwerte in diesem Fall im Zeitraum von 5 s bis 9 s einen positiven Realteil aufweisen. Es ist möglich, dass aufgrund des bis dahin bereits stark abgeklungenen Folgefehlers keine ausreichende Destabilisierung des Systems stattfindet, sondern eine stationäre Genauigkeit dennoch erreicht wird. Bis zum Ende der Simulation weisen alle Eigenwerte erneut einen negativen Realteil auf. Formal kann allerdings bereits am Beispiel einer möglichen Referenztrajektorie mit Ljapunows erster (indirekter) Methode gezeigt werden, dass dieser Regelungsansatz zu einem instabilen System zu gewissen Simulationszeitpunkten führt [17, Vorlesung 4]. Eine andere Wahl der Rückführung, etwa ohne lineare Dynamik, könnte dieses Problem lösen.

Eigenwerte der Jacobimatrix bei prototypischer Referenztrajektorie

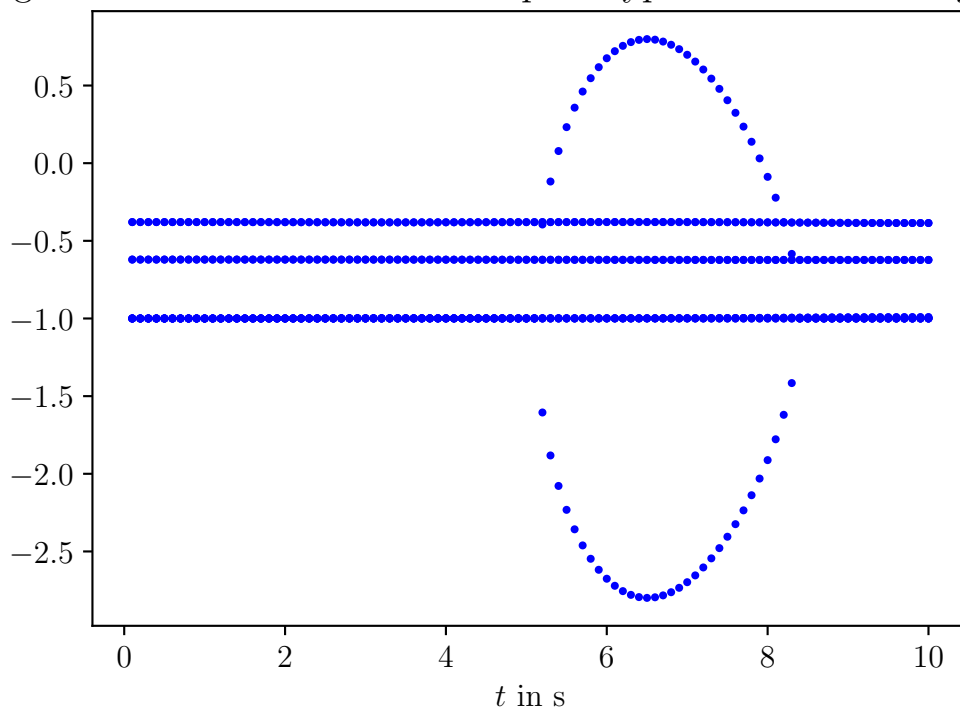


Abbildung 9 – Eigenwerte der Jacobimatrix \mathbf{J}_e bei der Folge einer prototypischen Referenztrajektorie bei der Regelung über exact feedforward linearization mit Selektionsmatrix.

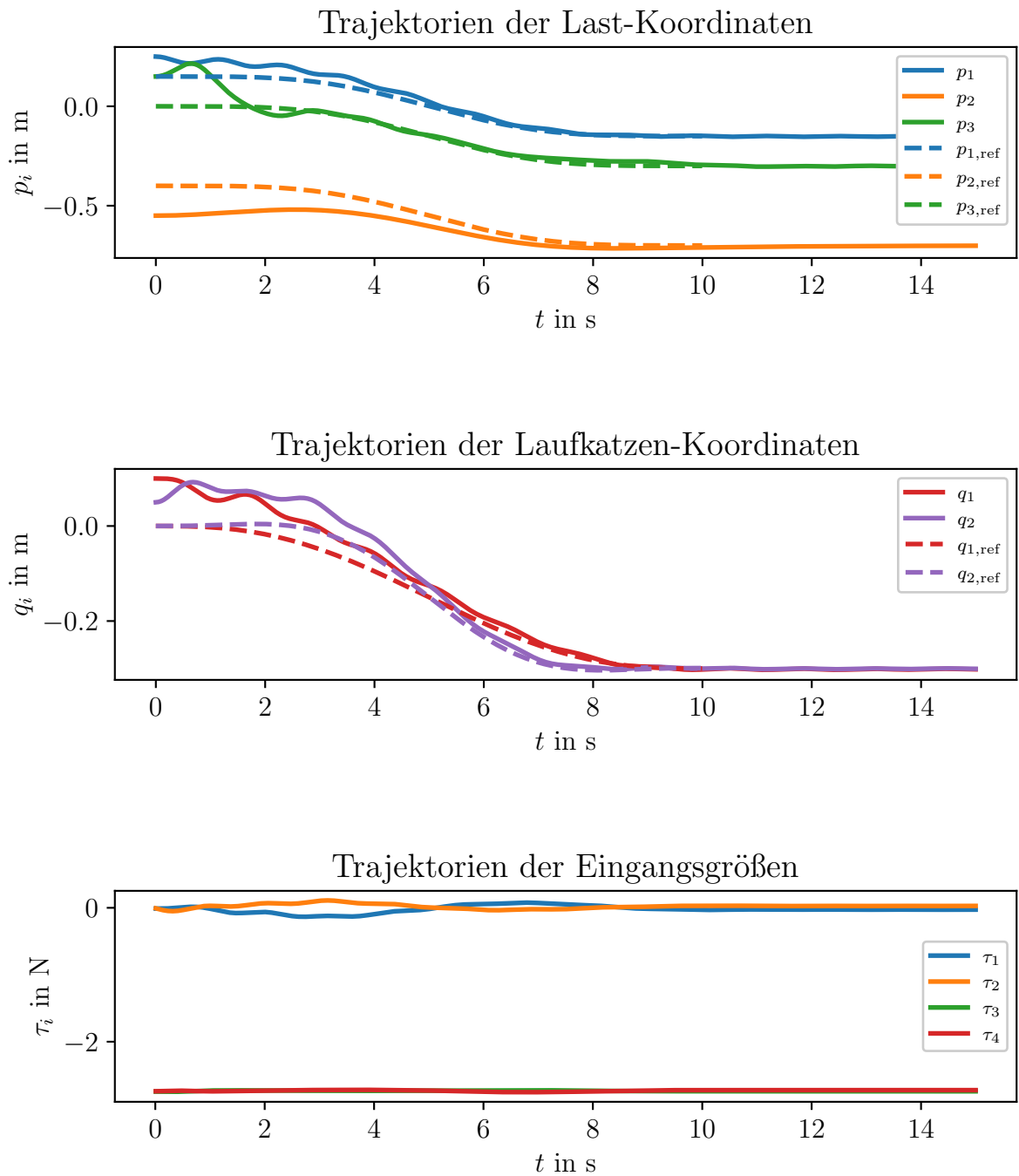


Abbildung 10 – Prototypische Trajektorien des Doppelkransystems bei der Überführung zwischen zwei Ruhelagen unter der Ausregelung von Anfangsfehlern über die exact feedforward linearization mit Selektionsmatrix. Koordinatenbezeichnung entsprechend Abbildung 5.

Kapitel 6

Fazit und Ausblick

6.1 Fazit

Für das zu untersuchende Brückenkransystem wurden analytische nichtlineare Modelle unter Nutzung des Lagrange-Formalismus ermittelt. Diese ergaben sich für die LG erster Art zu DGL- und für die LG zweiter Art zu DAE-Systemen. Für eine bessere Nachvollziehbarkeit der zugrunde liegenden Methodik wurde dieses Vorgehen zunächst anhand eines Einzelkransystems durchgeführt. Es erfolgte eine Identifikation aller zumeist geometrischen Modellparameter.

Die DGL-Modelle von Einzel- und Doppelkransystem wurden auf differenzielle Flachheit untersucht. Dabei wurde eine systematische Vorgehensweise für Mehrgrößensysteme skizziert. Dieses konnte auf Gleichungsebene anhand des Einzelkransystems veranschaulicht werden. Für das Doppelkransystem wurde aus Praktikabilitätsgründen auf eine explizite Darstellung der umfangreichen Ausdrücke in den Ergebnissen verzichtet. Unter Zuhilfenahme von Computeralgebrasystemen war eine Beschreibung dieser Zusammenhänge sowie Darstellung funktionaler Abhängigkeiten möglich.

Auf Grundlage des für das Doppelkransystem bestimmten flachen Ausgangs konnten polynombasierte Trajektorien der Ausgangskomponenten aufgestellt werden. Diese dienen der Überführung des Systems von einer gegebenen Ruhelage in eine andere. Auf Basis der Parametrisierung der Eingangskomponenten durch den flachen Ausgang war es möglich daraus Stellgrößenverläufe für eine Vorsteuerung abzuleiten. Zur Folgeregelung ist eine statische Rückführung aufgrund eines nicht wohldefinierten vektoriellen relativen Grades nicht möglich. Stattdessen konnte durch eine dynamische Rückführung mit linearen Fehlerdynamiken eine Stabilisierung des Systems bei Anfangsfehlern sowie Einschwingen entlang der Solltrajektorie erzielt werden, allerdings in Verbindung mit einem hohen Komplexitätsgrad des Stellgesetzes. Der Ansatz einer quasi-statischen Rückführung mündete aufgrund von Singularitäten in der Nähe von Ruhelagen in numerischen Problemen. Diese konnten im Rahmen dieser Arbeit nicht behoben werden. Mittels der exact feedforward linearization wurde alternativ ein Regelungsansatz verfolgt, aus welchem ein kompaktes Stellgesetz folgt. Dafür wurde eine kleinere Ordnung der

linearen Fehlerdynamiken angenommen sowie alle Komponenten oder auch nur eine heuristische Selektion der Zustandskomponenten mit diesen Fehlerdynamiken versehen. Dieser Ansatz eignete sich während praktischer Untersuchungen ebenso zur Stabilisierung des Systems bei Anfangsfehlern, verstößt aber gegen formale Stabilitätsbedingungen.

Die Durchführung umfangreicher symbolischer Berechnungen wie auch die numerisch simulative Verifikation der ermittelten Steuerungs- und Regelungsgesetze erfolgte über Jupyter-Notebooks in Python. Über eine Robustheit der Regelungsansätze wurde keine Aussage getroffen. Gerade bei nichtlinearen Systemen wie dem in dieser Arbeit untersuchten können bereits geringe Modellabweichungen zur Instabilität bei Regelungsansätzen der exakten Linearisierung führen, woraus hohe Genauigkeitsanforderungen für die Modellbildung folgen.

6.2 Ausblick

Für die Untersuchung der Eignung der ermittelten Regelungsansätze am realen Demonstratorsystem ist eine Implementierung auf der vorhandenen Hardware der Raspberry Pis notwendig. Gerade die Einhaltung von Anforderungen an die Echtzeitfähigkeit bei numerisch umfangreichen Stellgesetzen stellt dabei eine Herausforderung dar.

Die flachheitsbasierte Vorsteuerung kann zuerst allein erprobt werden, so dass auch aus Betrachtung der Abweichung von Ist- und Referenztrajektorien auf die Größenordnung der Modellabweichungen geschlossen werden kann. Die Regelungsansätze sollten daraufhin unter Begrenzung von Stellgrößen und Überwachung des gegebenenfalls instabil werdenden Systems getestet werden.

Neben einer vertieften Auseinandersetzung mit der quasi-statischen Rückführung, bei der Singularitäten in den Stellgesetzen gegebenenfalls durch die Hebung von Definitionslücken vermieden werden könnten, verspricht auch die Nutzung moderner Regelungsansätze aus dem Gebiet des maschinellen Lernens kompaktere sowie robuste Lösungen. Über das sogenannte bestärkende Lernen (englisch reinforcement learning) können sich Agenten für den Übergang zwischen Ruhelagen selbstständig eine Strategie aneignen. Ein Transferlernen aus Stelltrajektorien klassischer Regelungsansätze auf die Agenten wäre zudem denkbar.

Eine Präzisierung des physikalischen Modells durch Einbeziehung von Reibungseffekten könnte vor allem die Effektivität der daraus abgeleiteten Vorsteuerung weiter verbessern. Allerdings ergibt bereits das einfachere aktuelle Modell bei einem flachheitsbasiertem Vorgehen äußerst umfangreiche Terme für die Stellgrößen, so dass die Anpassung zuvor darauf trainierter Agenten über beispielsweise bestärkendes Lernen eine kompaktere Lösung ergeben könnte.

Literatur

- [1] Christian Andersson, Claus Führer und Johan Åkesson. „Assimulo: A unified framework for ODE solvers“. In: Mathematics and Computers in Simulation 116.0 (2015), S. 26–43. ISSN: 0378-4754. DOI: <http://dx.doi.org/10.1016/j.matcom.2015.04.007>.
- [2] Dr. rer. nat. Christine Steinhoff. Aktueller Begriff: Industrie 4.0. Techn. Ber. Wissenschaftliche Dienste des dt. Bundestages, 2016. URL: <https://www.bundestag.de/resource/blob/474528/cae2bfac57f1bf797c8a6e13394b5e70/industrie-4-0-data.pdf>.
- [3] Torsten Fließbach. Mechanik : Lehrbuch zur Theoretischen Physik I. Spektrum Akademischer Verlag, 2009. DOI: https://doi.org/10.1007/978-3-8274-2188-3_8.
- [4] Raspberry Pi Foundation. Technische Spezifikationen Raspberry Pi 4. Sep. 2022. URL: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>.
- [5] Klemens Fritzsche u. a. „Zur systematischen Bestimmung flacher Ausgänge nichtlinearer Mehrgrößensysteme“. In: at - Automatisierungstechnik 64.12 (2016), S. 948–960. DOI: [doi:10.1515/auto-2016-0079](https://doi.org/10.1515/auto-2016-0079). URL: <https://doi.org/10.1515/auto-2016-0079>.
- [6] Veit Hagenmeyer und Emmanuel Delaleau. „Exact feedforward linearization based on differential flatness“. In: International Journal of Control 76.6 (2003), S. 537–556. DOI: [10.1080/0020717031000089570](https://doi.org/10.1080/0020717031000089570). eprint: <https://doi.org/10.1080/0020717031000089570>. URL: <https://doi.org/10.1080/0020717031000089570>.
- [7] Internetpräsenz Fraunhofer IIS/EAS. Fraunhofer-Institut fuer Integrierte Schaltungen IIS, Institutsteil Entwicklung Adaptiver Systeme EAS. Juni 2022. URL: <https://www.eas.iis.fraunhofer.de/>.
- [8] Klaus Janschek. Systementwurf mechatronischer Systeme. Berlin: Springer, 2010. ISBN: 9783540788768. URL: [http://slubdd.de/katalog?TN_libero_mab2\)500406929](http://slubdd.de/katalog?TN_libero_mab2)500406929).
- [9] Klaus Janschek. Vorlesungsskript Elemente der Modellbildung und Simulationstechnik. 2020.
- [10] Carsten Knoll. „Regelungstheoretische Analyse- und Entwurfsansätze fuer unteraktuierte mechanische Systeme“. Diss. TU Dresden, 2016. URL: <https://nbn-resolving.org/urn:nbn:de:bsz:14-qucosa-209765>.

-
- [11] Klaus Röbenack. Nichtlineare Regelungssysteme. Springer Vieweg Berlin, Heidelberg, 2017. DOI: <https://doi.org/10.1007/978-3-662-44091-9>.
 - [12] J. Rudolph. Vorlesungsskript Flache Systeme. 2007.
 - [13] STMicroelectronics. Technische Spezifikationen STM32F303K8. Sep. 2022. URL: <https://www.st.com/en/microcontrollers-microprocessors/stm32f303k8.html>.
 - [14] sybtools GitHub Repository. Juni 2022. URL: <https://github.com/TUD-RST/sybtools>.
 - [15] Bruce Torby. „Energy Methods“. In: Advanced Dynamics for Engineers (1984).
 - [16] Wikipedia. Trägheitsmoment - Wikipedia, Die freie Enzyklopädie. Juni 2022. URL: <https://de.wikipedia.org/wiki/Tr%C3%A4gheitsmoment#:~:text=Ein%20Quader%2C%20der%20um%20eine%20Achse%20durch%20den%20Mittelpunkt%20rotiert%2C%20die%20parallel%20zu%20seinen%20Kanten%20c%20liegt..>
 - [17] Jan Winkler. Vorlesungsskript Nichtlineare Regelungstechnik 1. 2021.
 - [18] Konstantin Wrede. GitHub Repository zu dieser Studienarbeit. Sep. 2022. URL: <https://github.com/wredsen/SA-Wrede>.