

# 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:	30. September 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, 30. September 2022

Konstantin Wrede

## **Kurzfassung**

An dieser Stelle fügen Sie bitte eine deutsche Kurzfassung ein.

## **Abstract**

Please insert the English abstract here.

# Inhaltsverzeichnis

Verzeichnis der Formelzeichen	VII
Abbildungsverzeichnis	VIII
Tabellenverzeichnis	1
<b>1 Einleitung</b>	<b>2</b>
<b>2 System- und Problembeschreibung</b>	<b>3</b>
2.1 Demonstratorsystem . . . . .	3
2.2 Problembeschreibung und Zielsetzung . . . . .	4
<b>3 Analytische Modellbildung</b>	<b>6</b>
3.1 Allgemeine Modellannahmen . . . . .	6
3.2 Modellierung mittels Lagrange-Formalismus . . . . .	7
3.2.1 Lagrange-Gleichungen erster Art . . . . .	7
3.2.2 Lagrange-Gleichungen zweiter Art . . . . .	8
3.3 Generierung und Simulation von DAE-Systemen . . . . .	8
3.4 Analytisches Modell Einzelkran . . . . .	10
3.5 Analytisches Modell Doppelkran . . . . .	12
3.5.1 Ansatz über ein ODE-System . . . . .	12
3.5.2 Ansatz über ein DAE-System . . . . .	14
3.6 Systemidentifikation . . . . .	17
<b>4 Flachheitsanalyse</b>	<b>19</b>
4.1 Definition differenzieller Flachheit . . . . .	19
4.2 Flachheitsanalyse von MIMO-Systemen . . . . .	19
4.3 Anwendung Flachheitsanalyse am Einzelkran . . . . .	21
4.4 Anwendung Flachheitsanalyse am Doppelkran . . . . .	22
<b>5 Steuerungs- und Regelungsentwurf</b>	<b>25</b>
5.1 Regelung zur Stabilisierung von Ruhelagen . . . . .	25
5.2 Trajektorienplanung für den flachen Ausgang . . . . .	25
5.3 Trajektorienfolge Regelung . . . . .	26
5.3.1 Vektorieller relativer Grad . . . . .	26
5.3.2 Statische Rückführung . . . . .	26

5.3.3	Zustandsrückführung aus Fehlerdynamik . . . . .	28
5.3.4	Dynamische Erweiterung . . . . .	28
5.3.5	Quasi-statische Rückführungen . . . . .	31
5.3.6	Exact feedforward linearization . . . . .	34
<b>6</b>	<b>Fazit und Ausblick</b>	<b>39</b>
6.1	Fazit . . . . .	39
6.2	Ausblick . . . . .	39
	<b>Literatur</b>	<b>40</b>

# Verzeichnis der Formelzeichen

# Abbildungsverzeichnis

1	CAD-Zeichnung des Doppelkransystems . . . . .	3
2	Doppelkran-Demonstratorsystem mit Laufkatzen und Last . . . . .	4
3	Kurzbeschreibung für Abbildungsverzeichnis . . . . .	10
4	Kurzbeschreibung für Abbildungsverzeichnis . . . . .	12
5	Kurzbeschreibung für Abbildungsverzeichnis . . . . .	15
6	Kurzbeschreibung für Abbildungsverzeichnis . . . . .	30
7	Kurzbeschreibung für Abbildungsverzeichnis . . . . .	35
8	Kurzbeschreibung für Abbildungsverzeichnis . . . . .	37
9	Kurzbeschreibung für Abbildungsverzeichnis . . . . .	38



# Tabellenverzeichnis

1	Physikalische Parameter des Doppelkrandemonstratorsystems. . . . .	18
2	Relative Grade der Ausgänge und explizites Auftreten der Eingänge. . .	27
3	Auftreten der Eingänge $\tau_j$ bei Ordnung $k$ der Ausgangsableitung $y_i$ . . .	31

# Kapitel 1

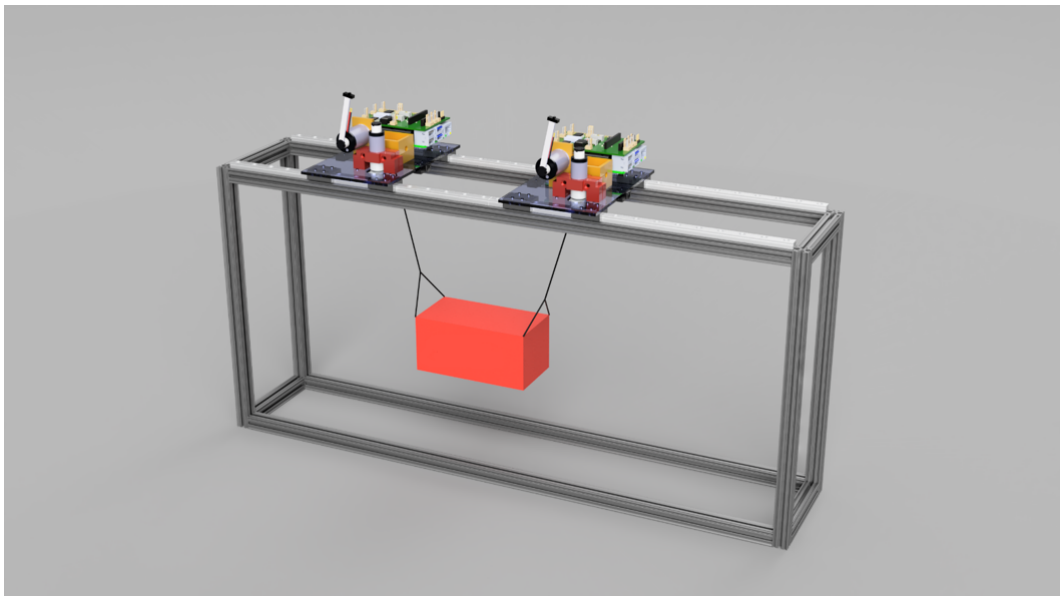
## Einleitung

- ganz am Ende

# Kapitel 2

## System- und Problembeschreibung

### 2.1 Demonstratorsystem



**Abbildung 1** – CAD-Zeichnung des Doppelkran-Demonstratorsystems ohne Verkabelung.

Im Rahmen vorangegangener studentischer Arbeiten ist am Fraunhofer-Institut für Integrierte Schaltungen IIS, Institutsteil Entwicklung Adaptiver Systeme EAS [5] in Dresden ein Demonstratorsystem entwickelt worden. Langfristig soll dieses Vorteile von Regelungsstrategien auf verteilten Recheneinheiten gegenüber einer zentralen Messgrößenverarbeitung und Stellgrößenberechnung untersuchen. Das Konzept des Demonstrators ist in [Abbildung 1](#) dargestellt.

Das Demonstratorsystems besteht aus zwei Brückenkränen, die eine gemeinsame Last in einer festen Ebene anheben. Die Kräne befinden sich auf Schienen und verfügen jeweils

über einen Raspberry Pi 4B als Hauptrecheneinheit sowie einen STM32-Mikrocontroller für Motoransteuerungen und Messungen. Beide Raspberry Pis können über eine LAN-Verbindung miteinander kommunizieren.

Beide Kräne sind auf Schienen in horizontaler Richtung sowie die Seillängen mit jeweils einem DC-Motor aktuiert. Auf den STM32-Mikrocontroller ist bereits eine unterlagerte Strom- beziehungsweise Kraftregelung für diese implementiert. Messungen der Seillängen und Kranpositionen auf der Schiene erfolgen mittels Inkrementalgebern nach einem anfänglichen Kalibrierungsvorgang. Die Seilwinkel werden mittels mitschwingender Potentiometer bestimmt. Abbildung 2 zeigt eine Photographie der realen Anordnung des Demonstrators.

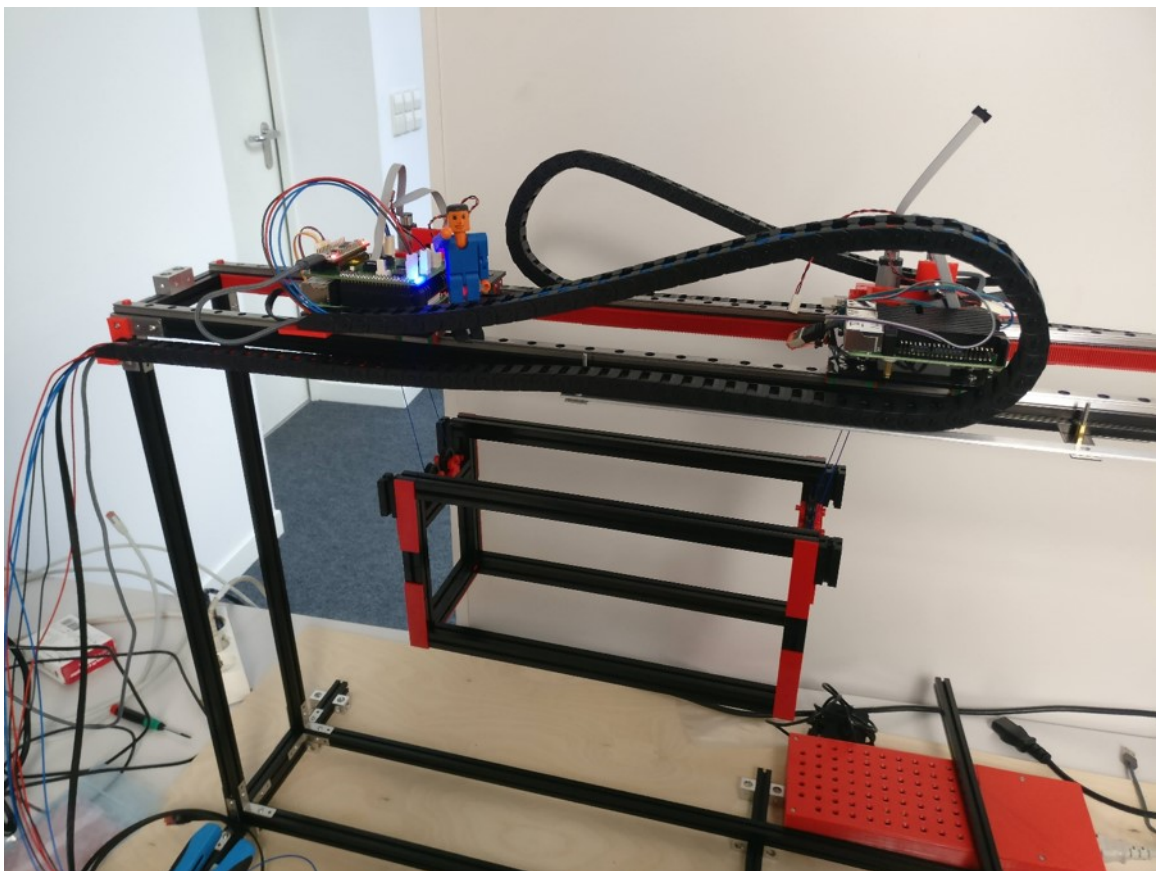


Abbildung 2 – Doppelkran-Demonstratorsystem mit Laufkatzen und Last.

## 2.2 Problembeschreibung und Zielsetzung

Bei der Bewegung von Containern in Häfen ist ein ruckarmer und gegenüber der Horizontalen stabiler Transport notwendig. Ziel der Studienarbeit ist es deshalb, bezüglich

des vorhandenen Demonstratorsystems eine zentrale Referenzregelstrategie zu entwerfen. Damit soll unter Vorgabe von Sollposen eine Planung von Trajektorien der Last in der Ebene und Folgeregelung zur Überführung dieser zwischen verschiedenen Ruhelagen ermöglicht werden. Diese Überlegungen sollen auf Basis einer Modellierung des Krans als Mehrkörpersystems geschehen.

# Kapitel 3

## Analytische Modellbildung

### 3.1 Allgemeine Modellannahmen

In der folgenden Modellierung wird das Demonstratorsystem nur planar betrachtet, also die gesamte 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 dieser nur in einer Ebene zulässt. Die Seile werden aufgrund der geringen Dicke und auch im Vergleich zu den Laufkatzen und der Last als masselos angenommen sowie nicht mit einem Trägheitsmoment behaftet. Die Last wird trotz ihrer Aussparungen mit einer homogenen Masseverteilung modelliert. Für eine exakte Beschreibung wären CAD-Daten oder eine Demontage mit Vermessung der Teilkomponenten notwendig. Es wird aus Gründen der Einfachheit auf die Abbildung dissipativer Kräfte verzichtet. Zudem können diese bei einer funktionierenden unterlagerten Regelung als kompensiert angenommen werden.

Die Modellbildung erfolgt in mehreren Stufen. Es ist zunächst problemlos möglich mittels der Lagrange-Gleichungen erster oder auch zweiter Art den Einzelkran darzustellen. Dabei hat die Modellierung zweiter Art den Vorteil, dass Stellkräfte einfacher abgebildet werden können und sich ein typischerweise effizient zu simulierendes gewöhnliches Differenzialgleichungssystem (DGL-System, engl. ODE system) ergibt. Ausgehend von den Erkenntnissen und aufgrund geringer Komplexität der überschaubaren Terme des Einzelkrans, kann die prinzipielle Methodik verifiziert werden. Daran anschließend wird eine Erweiterung auf ein Doppelkransystem in Analogie zum realen Demonstrator vorgenommen. Wegen der an beiden Laufkatzen befestigten Last liegt zunächst alternativ zum vorherigen Vorgehen die Nutzung der Lagrange-Gleichungen erster Art, bei der mittels algebraischer Zwangsbedingungen ein Differenzial-algebraisches Gleichungssystem (DAE-System, von engl. differential algebraic equations) zur Simulation generiert wird, nahe. Allerdings ist dabei eine 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 physischen Demonstrator sperrt das Schneckengetriebe der Seilwinde im stromfreien Fall die Hubaktuierung.

Auch bei einem Doppelkransystem ist stattdessen eine Modellierung mit den Lagrange-Gleichungen zweiter Art möglich. Es folgt nun ein kurzer theoretischer Überblick zu beiden erwähnten Arten des Lagrange-Formalismus.

## 3.2 Modellierung mittels Lagrange-Formalismus

Die Dynamik mechanischer Systeme lässt sich über Differenzialgleichungen, den so genannten Lagrange-Gleichungen beschreiben. Dabei wird eine Menge aus  $n$  auftretenden und zeitlich veränderlichen Koordinaten als Konfigurationskoordinaten oder Systemgrößen  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n)^T$  bezeichnet. Die zeitlichen Änderungsraten dieser sind die (Konfigurations-)Geschwindigkeiten  $\dot{\boldsymbol{\theta}}$ . Anfänglich wurde der Literatur [8, S.10] folgend eine Unterteilung der Systemgrößen in "aktive" und "passive" Koordinaten vorgenommen. Im Verlauf dieser Arbeit ist bemerkt worden, dass die Aufgliederung  $\boldsymbol{\theta} = (\mathbf{q}, \mathbf{p})^T$  in die direkt aktuierten Koordinaten  $\mathbf{q}$  und nicht direkt aktuierten Koordinaten  $\mathbf{p}$  eine treffendere Bezeichnung darstellt. [8, 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:

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

Eine stark automatisierte Durchführung dieses Formalismus ist unter Nutzung des Python-Pakets symbttools [10] möglich.

### 3.2.1 Lagrange-Gleichungen erster Art

Mit den Lagrange-Gleichungen erster Art können Probleme mit Zwangsbedingungen und -kräften dargestellt werden:

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

Die sich auf die jeweilige Koordinate  $\theta_i$  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 [2, S. 49].

Es können  $m$  holonomen Zwangsbedingungen  $g_1(\boldsymbol{\theta}) = \dots = g_m(\boldsymbol{\theta}) = 0$  eingeführt werden, aus denen die Zwangskraft  $\tilde{Q}_i$  in Richtung der Koordinate  $\theta_i$  folgt:

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

In dieser Beziehung bezeichnet  $\lambda_j$  den jeweiligen Lagrange-Multiplikator.

### 3.2.2 Lagrange-Gleichungen zweiter Art

Die Lagrange-Gleichungen zweiter Art beschreiben bezüglich des Vorhergehenden den Spezialfall ohne Zwangsbedingungen, also  $m = 0$ . Zwangskräfte müssen dabei nicht explizit bestimmt werden. Die Bewegungsgleichungen können folgendermaßen aus der Lagrange-Funktion abgeleitet werden:

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

Zur Bestimmung der Komponenten  $Q_i$  der verallgemeinerten Kraft wird das Prinzip der virtuellen Arbeit herangezogen [11]:

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

Dabei entspricht bei einem System von  $l$  (massebehafteten) Teilchen  $\mathbf{r}_k$  dem Richtungsvektor zum  $k$ -ten Partikel,  $\mathbf{F}_k$  der jeweils entlang dieses Richtungsvektoren angewandten Stellkraft,  $\delta \mathbf{r}_k$  der virtuellen Verschiebung des Partikels im Raum und  $\delta \theta_i$  der virtuellen Verschiebung der Koordinate  $\theta_i$ , welche der Beziehung

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

genügt.

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 Nebenbedingungen) ergänzt werden. Diese Nebenbedingungen können zur Darstellung



von Zwangsbedingungen der Lagrange-Gleichungen erster Art genutzt werden. DAE-Systeme lassen sich typischerweise in einer semi-expliziten Form darstellen:

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

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{z}, t), \quad (3.10)$$

wobei  $\mathbf{x}$  dem Zustand,  $\mathbf{z}$  den algebraischen Variablen (weitere Systemgrößen, die in den Systemgleichungen nicht abgeleitet vorkommen),  $\mathbf{u}$  dem Systemeingang sowie  $t$  der Zeit entspricht. [7, S.137]

Eine Möglichkeit zu Klassifikation von DAE-Systemen ist der differenzielle Index  $i_d$ . Dieser entspricht der minimale Anzahl an Differenziation  $\frac{d}{dt}$  der AGL  $\mathbf{g}$  (Zwangsbedingungen), damit unter Einbeziehung der DGL ein explizites DGL-System aus dem DAE-System entsteht. Ein gewöhnliches DGL-System besitzt also den differenzielle Index  $i_d = 0$ . Die Differenziation der AGL mit dem Resultat eines DAE-Systems mit kleinerem Index wird als Indexreduktion bezeichnet. [7, S.139]

Für die Simulation von DAE-Systemen ist die numerische Integration dieser Gleichungssysteme notwendig. Die in dieser Arbeit untersuchten mechanischen Systeme sind im Allgemeinen solche mit starrer Kopplung als Zwangsbedingungen vom Index  $i_d = 3$  und können über folgende Bewegungsgleichungen beschrieben werden:

$$\mathbf{0} = \mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{K}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{G}(\boldsymbol{\theta})\boldsymbol{\lambda} - \mathbf{B}(\boldsymbol{\theta})\boldsymbol{\tau} \quad (3.11)$$

$$\mathbf{0} = \mathbf{g}(\boldsymbol{\theta}). \quad (3.12)$$

[7, S.240]

Zur Integration davon gibt es verschiedene Möglichkeiten [6, Kap. 8]:

- Indexreduktion auf Index  $i_d = 2$  und anschließende Integration über ein implizites<sup>1</sup> Verfahren
- Indexreduktion auf Index  $i_d = 1$  und anschließende Integration über ein explizites<sup>2</sup> 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 `symtools` [10] enthaltene Bibliothek `modeltools` führt die Reduktion von Index-3-Systemen auf Index-1-Systeme durch. Zur numerischen Berechnung kann daraufhin der Solver `ODASSL` des Python-Pakets `assimulo` [1] verwendet werden.

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

<sup>1</sup>Auf 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.

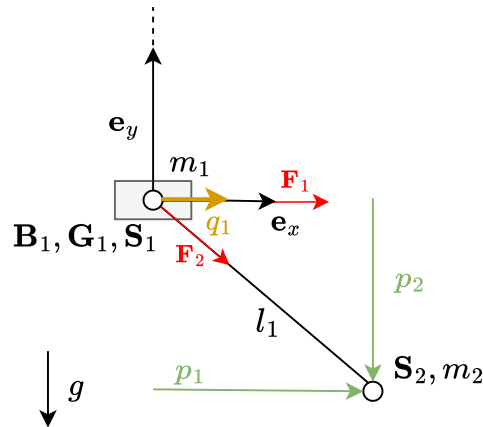
<sup>2</sup>Zur Berechnung des Näherungswertes von  $\mathbf{x}(t_{i+1})$  an der Stelle  $t_{i+1}$  wird einzig der zuletzt berechnete Näherungswert von  $\mathbf{x}(t_i)$  an der Stelle  $t_i$  benötigt.

mehr frei gewählt werden, da  $\mathbf{z}(0)$  nicht mehr allein aus AGL bestimmbar ist. Es ist 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 es andernfalls zu inkonsistenten Anfangswerten kommt, folgen daraus Simulationsfehler in den ersten Schritten oder sogar vollständig falsche Ergebnisse. [7, S.207]

### 3.4 Analytisches Modell Einzelkran

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [14, flatness\_notebooks/ODE\_flatness\_analysis\_single\_crane.ipynb] zu finden.

Mittels der Lagrange-Gleichungen zweiter Art wird im Folgenden ein ODE-Modell des Einzelkrans erzeugt. Dabei wird entsprechend der Erläuterungen aus Abschnitt 3.2.2 vorgegangen. Diese Modellierung hat den Vorteil, dass durch die generalisierte Kraft 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** – Langbeschreibung, die gerne auch über mehrer Zeilen gehen darf.

Als direkt aktuierte Koordinate wird nur die  $x$ -Verschiebung  $q_1$  der Laufkatzenposition ( $\mathbf{B}_1 = \mathbf{G}_1 = \mathbf{S}_1$ ) ausgewählt, als nicht direkt aktuierte Koordinaten 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:

$$l_1 = \sqrt{(p_1 - q_1)^2 + p_2^2} \quad (3.13)$$

ausgedrückt werden kann.

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

$$\mathbf{S}_1 = \begin{pmatrix} q_1 \\ 0 \end{pmatrix}, \quad \mathbf{S}_2 = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}. \quad (3.14)$$

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

$$T = \frac{m_1}{2} \dot{\mathbf{S}}_1^T \dot{\mathbf{S}}_1 + \frac{m_2}{2} \dot{\mathbf{S}}_2^T \dot{\mathbf{S}}_2 = \frac{m_1 \dot{q}_1^2}{2} + \frac{m_2 \dot{p}_1^2}{2} + \frac{m_2 \dot{p}_2^2}{2} \quad (3.15)$$

$$V = m_2 g \mathbf{S}_2^T \mathbf{e}_y = m_2 g p_2. \quad (3.16)$$

Die Systemgleichungen können durch die Lagrange-Gleichungen zweiter Art zunächst in Abhängigkeit der Komponenten der generalisierten Kraft  $Q_i$  bestimmt werden:

$$-Q_1 + m_2 \ddot{p}_1 = 0 \quad (3.17a)$$

$$-Q_2 + g m_2 + m_2 \ddot{p}_2 = 0 \quad (3.17b)$$

$$-Q_3 + m_1 \ddot{q}_1 = 0. \quad (3.17c)$$

Die Eingangskomponenten  $\tau_1, \tau_2$  des Systems entsprechen den unterlagert geregelten Kräften der Motoren der Laufkatze bzw. der Seilwinde. Sie können über Stellkräfte vektoriell dargestellt werden:

$$\mathbf{F}_1 = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix}, \quad \mathbf{F}_2 = \begin{pmatrix} \frac{\tau_2(p_1 - q_1)}{l_1} \\ \frac{p_2 \tau_2}{l_1} \end{pmatrix}. \quad (3.18)$$

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:

$$\mathbf{Q} = \begin{pmatrix} \frac{\tau_2(p_1 - q_1)}{l_1} \\ \frac{p_2 \tau_2}{l_1} \\ \tau_1 - \frac{\tau_2(p_1 - q_1)}{l_1} \end{pmatrix}. \quad (3.19)$$

Durch einsetzen dieser lässt sich ein abschließender Satz an Systemgleichungen des Einzelkrans bilden:

$$m_2 \ddot{p}_1 - \frac{\tau_2(p_1 - q_1)}{l_1} = 0 \quad (3.20a)$$

$$g m_2 + m_2 \ddot{p}_2 - \frac{p_2 \tau_2}{l_1} = 0 \quad (3.20b)$$

$$m_1 \ddot{q}_1 - \tau_1 + \frac{\tau_2(p_1 - q_1)}{l_1} = 0. \quad (3.20c)$$



Die Position der Massenschwerpunkte und Gelenke kann somit wie folgt durch die Koordinaten beschrieben werden:

$$\mathbf{s}_1 = \begin{pmatrix} q_1 \\ 0 \end{pmatrix}, \mathbf{s}_2 = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}, \mathbf{s}_3 = \begin{pmatrix} l_0 + q_2 \\ 0 \end{pmatrix}, \mathbf{G}_3 = \begin{pmatrix} p_1 - s_2 \cos(p_3) \\ p_2 - s_2 \sin(p_3) \end{pmatrix}, \mathbf{G}_4 = \begin{pmatrix} p_1 + s_2 \cos(p_3) \\ p_2 + s_2 \sin(p_3) \end{pmatrix}. \quad (3.23)$$

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

$$\begin{aligned} T &= \frac{m_1}{2} \dot{\mathbf{S}}_1^T \dot{\mathbf{S}}_1 + \frac{m_2}{2} \dot{\mathbf{S}}_2^T \dot{\mathbf{S}}_2 + \frac{J_2}{2} \dot{p}_3^2 + \frac{m_3}{2} \dot{\mathbf{S}}_3^T \dot{\mathbf{S}}_3 \\ &= \frac{J_2 \dot{p}_3^2}{2} + \frac{m_1 \dot{q}_1^2}{2} + \frac{m_2 \dot{p}_1^2}{2} + \frac{m_2 \dot{p}_2^2}{2} + \frac{m_3 \dot{q}_2^2}{2} \end{aligned} \quad (3.24)$$

$$V = m_2 g \mathbf{S}_2^T \mathbf{e}_y = m_2 g p_2. \quad (3.25)$$

Die Systemgleichungen können durch die Lagrange-Gleichungen zweiter Art zunächst in Abhängigkeit der Komponenten der generalisierten Kraft  $Q_i$  bestimmt werden:

$$-Q_1 + m_2 \ddot{p}_1 = 0 \quad (3.26a)$$

$$-Q_2 + g m_2 + m_2 \ddot{p}_2 = 0 \quad (3.26b)$$

$$J_2 \ddot{p}_3 - Q_3 = 0 \quad (3.26c)$$

$$-Q_4 + m_1 \ddot{q}_1 = 0 \quad (3.26d)$$

$$-Q_5 + m_3 \ddot{q}_2 = 0. \quad (3.26e)$$

Die Komponenten des Systemeingangs  $\boldsymbol{\tau}$  können über Stellkräfte vektoriell dargestellt werden:

$$\mathbf{F}_1 = \begin{pmatrix} \tau_1 \\ 0 \end{pmatrix}, \mathbf{F}_2 = \begin{pmatrix} \tau_2 \\ 0 \end{pmatrix}, \mathbf{F}_3 = \begin{pmatrix} \frac{\tau_3(p_1 - q_1 - s_2 \cos(p_3))}{\frac{\tau_3(p_2 - s_2 \sin(p_3))}{l_1}} \\ \frac{\tau_3(p_2 - s_2 \sin(p_3))}{l_1} \end{pmatrix}, \mathbf{F}_4 = \begin{pmatrix} \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 \cos(p_3))}{\frac{\tau_4(p_2 + s_2 \sin(p_3))}{l_2}} \\ \frac{\tau_4(p_2 + s_2 \sin(p_3))}{l_2} \end{pmatrix}. \quad (3.27)$$

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

$$\mathbf{Q} = \begin{pmatrix} \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 c_{p_3})}{l_2} + \frac{\tau_3(p_1 - q_1 - s_2 c_{p_3})}{l_1} \\ \frac{\tau_4(p_2 + s_2 s_{p_3})}{l_2} + \frac{\tau_3(p_2 - s_2 s_{p_3})}{l_1} \\ \frac{s_2 \tau_4(p_2 + s_2 s_{p_3}) c_{p_3}}{l_2} - \frac{s_2 \tau_4(-l_0 + p_1 - q_2 + s_2 c_{p_3}) s_{p_3}}{l_2} - \frac{s_2 \tau_3(p_2 - s_2 s_{p_3}) c_{p_3}}{l_1} + \frac{s_2 \tau_3(p_1 - q_1 - s_2 c_{p_3}) s_{p_3}}{l_1} \\ \tau_1 - \frac{\tau_3(p_1 - q_1 - s_2 c_{p_3})}{l_1} \\ \tau_2 - \frac{\tau_4(-l_0 + p_1 - q_2 + s_2 c_{p_3})}{l_2} \end{pmatrix}. \quad (3.28)$$

Durch Einsetzen dieser lässt sich ein abschließender Satz an Systemgleichungen des

Doppelkrans bilden:

$$m_2 \ddot{p}_1 - \frac{\tau_4 (-l_0 + p_1 - q_2 + s_2 \cos(p_3))}{l_2} - \frac{\tau_3 (p_1 - q_1 - s_2 \cos(p_3))}{l_1} = 0 \quad (3.29a)$$

$$gm_2 + m_2 \ddot{p}_2 - \frac{\tau_4 (p_2 + s_2 \sin(p_3))}{l_2} - \frac{\tau_3 (p_2 s_2 \sin(p_3))}{l_1} = 0 \quad (3.29b)$$

$$J_2 \ddot{p}_3 - \frac{s_2 \tau_4 (p_2 + s_2 \sin(p_3)) \cos(p_3)}{l_2} + \frac{s_2 \tau_4 (-l_0 + p_1 - q_2 + s_2 \cos(p_3)) \sin(p_3)}{l_2} \\ + \frac{s_2 \tau_3 (p_2 - s_2 \sin(p_3)) \cos(p_3)}{l_1} - \frac{s_2 \tau_3 (p_1 - q_1 - s_2 \cos(p_3)) \sin(p_3)}{l_1} = 0 \quad (3.29c)$$

$$m_1 \ddot{q}_1 - \tau_1 + \frac{\tau_3 (p_1 - q_1 - s_2 \cos(p_3))}{l_1} = 0 \quad (3.29d)$$

$$m_3 \ddot{q}_2 - \tau_2 + \frac{\tau_4 (-l_0 + p_1 - q_2 + s_2 \cos(p_3))}{l_2} = 0. \quad (3.29e)$$

Alternativ kann das Doppelkransystem auch eingangsaffin im Zustandsraum beschrieben werden:

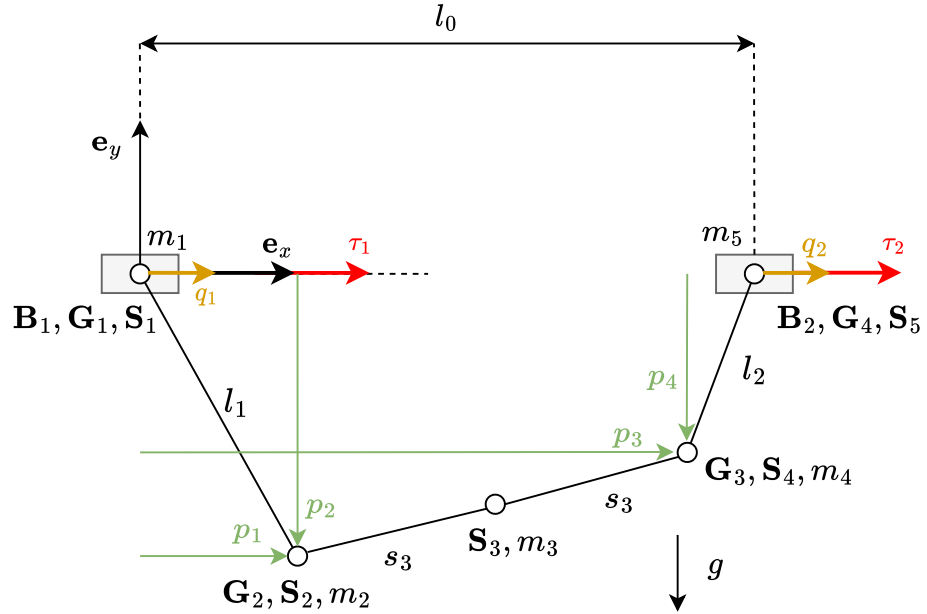
$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\boldsymbol{\tau} \text{ mit} \\ \mathbf{x} &= (p_1, p_2, p_3, q_1, q_2, \dot{p}_1, \dot{p}_2, \dot{p}_3, \dot{q}_1, \dot{q}_2)^T, \\ \mathbf{f}(\mathbf{x}) &= (\dot{p}_1, \dot{p}_2, \dot{p}_3, \dot{q}_1, \dot{q}_2, 0, -g, 0, 0, 0)^T, \\ \mathbf{g}(\mathbf{x}) &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{p_1 - q_1 - s_2 \cos(p_3)}{l_1 m_2} & \frac{-l_0 + p_1 - q_2 + s_2 \cos(p_3)}{l_2 m_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{p_2 - s_2 \sin(p_3)}{l_1 m_2} & \frac{p_2 + s_2 \sin(p_3)}{l_2 m_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{s_2 (p_1 \sin(p_3) - p_2 \cos(p_3) - q_1 \sin(p_3))}{J_2 l_1} & \frac{s_2 (l_0 \sin(p_3) - p_1 \sin(p_3) + p_2 \cos(p_3) + q_2 \sin(p_3))}{J_2 l_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{m_1} & 0 & \frac{-p_1 + q_1 + s_2 \cos(p_3)}{l_1 m_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{m_3} & 0 & \frac{l_0 - p_1 + q_2 - s_2 \cos(p_3)}{l_2 m_3} & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \end{aligned} \quad (3.30)$$

### 3.5.2 Ansatz über ein DAE-System

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [\[14, double\\_crane\\_notebooks/DAE\\_double\\_crane\\_cartesian.ipynb\]](#) zu finden.

Mittels der Lagrange-Gleichungen erster Art wird im Folgenden ein DAE-Modell des Doppelkrans erzeugt. Dabei wird entsprechend der Erläuterungen aus 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 ist trotzdem sinnvoll. Durch die Einprägung von Zwangsbedingungen kann die kinematische Schleife wegen der gemeinsamen Last intuitiv abgebildet werden kann.

Aufgrund der algebraischen Gleichungen würde sich ein Flachheitsnachweis anders als bei den ODE-Systemen gestalten. Es wird zunächst einzig eine horizontale Aktuierung der Laufkatzen durch den Eingang  $\tau = (\tau_1, \tau_2)^T$  betrachtet. Für eine spätere effiziente Simulation mit aktuierten Seilwinden wäre gerade bei konstanten Seillängen eine Umformulierung des Problems mit diesen Längen als Stellgröße sinnvoll. (Quelle: Buch von Rudolf S.21 -> Nextcloud Flachheit/fotos ???) Abbildung 5 zeigt die Konfiguration dieses Doppelkransystems.



**Abbildung 5** – Langbeschreibung, die gerne auch über mehrer Zeilen gehen darf.

Als direkt aktuierte Koordinate werden nur die  $x$ -Verschiebungen  $q_1$  und  $q_2$  der Laufkatzenpositionen ( $\mathbf{B}_1 = \mathbf{G}_1 = \mathbf{S}_1$ ,  $\mathbf{B}_2 = \mathbf{G}_4 = \mathbf{S}_5$ ) 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  $\mathbf{G}_3 (= \mathbf{S}_4)$ .

Für konstante Seillängen können folgende Zwangsbedingungen formuliert werden:

$$l_1 - \sqrt{p_2^2 + (p_1 - q_1)^2} = 0 \quad (3.31)$$

$$l_2 - \sqrt{p_4^2 + (-l_0 + p_3 - q_2)^2} = 0. \quad (3.32)$$

Die konstante Lastlänge wird durch

$$2s_3 - \sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2} \quad (3.33)$$

ausgedrückt.

Die Position der Massenschwerpunkte und Gelenke kann somit wie folgt durch die Koordinaten beschrieben werden:

$$\begin{aligned}\mathbf{S}_1 &= \begin{pmatrix} q_1 \\ 0 \end{pmatrix}, \quad \mathbf{S}_2 = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix}, \quad \mathbf{S}_3 = \frac{1}{2}(\mathbf{S}_2 + \mathbf{S}_4) = \frac{1}{2} \begin{pmatrix} p_1 + p_3 \\ p_2 + p_4 \end{pmatrix}, \\ \mathbf{S}_4 &= \begin{pmatrix} p_3 \\ p_4 \end{pmatrix}, \quad \mathbf{S}_5 = \begin{pmatrix} l_0 + q_2 \\ 0 \end{pmatrix}.\end{aligned}\quad (3.34)$$

Es ist notwendig, dass alle abgeleiteten Koordinaten in der kinetischen Energie  $T$  vorhanden sind, damit die Massenmatrix  $\mathbf{M}(\boldsymbol{\theta})$  in Gleichung (3.11) regulär ist [8, S. 7]. Dies wäre bei einer Modellierung der Last mit homogener Masseverteilung und Trägheitsmoment nicht der Fall. Deshalb wird die Last durch ein System aus drei Massen ( $m_2, m_3, m_4$ ) beschrieben, das sich simulativ äquivalent verhält und folgenden Gleichungen genügt:

$$m_{\text{ges}} = m_2 + m_3 + m_4 \quad (3.35)$$

$$m_2 = m_4 \quad (3.36)$$

$$J = m_2 s_3^2 + m_4 s_3^2 = 2m_2 s_3^2 \quad (3.37)$$

$$\Rightarrow m_2 = m_4 = \frac{J}{2s_3^2} \quad (3.38)$$

$$\Rightarrow m_3 = m_{\text{ges}} - m_2 - m_4. \quad (3.39)$$

Die Größen  $m_{\text{ges}}$  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:

$$\begin{aligned}T &= \frac{m_1}{2} \dot{\mathbf{S}}_1^T \dot{\mathbf{S}}_1 + \frac{m_2}{2} \dot{\mathbf{S}}_2^T \dot{\mathbf{S}}_2 + \frac{m_3}{2} \dot{\mathbf{S}}_3^T \dot{\mathbf{S}}_3 + \frac{m_4}{2} \dot{\mathbf{S}}_4^T \dot{\mathbf{S}}_4 + \frac{m_5}{2} \dot{\mathbf{S}}_5^T \dot{\mathbf{S}}_5 \\ &= \frac{m_1 \dot{q}_1^2}{2} + \frac{m_2 \dot{p}_1^2}{2} + \frac{m_2 \dot{p}_2^2}{2} + \frac{m_3 \cdot 0.25 (\dot{p}_1 + \dot{p}_3)^2}{2} + \frac{m_3 \cdot 0.25 (\dot{p}_2 + \dot{p}_4)^2}{2} \\ &\quad + \frac{m_4 \dot{p}_3^2}{2} + \frac{m_4 \dot{p}_4^2}{2} + \frac{m_5 \dot{q}_2^2}{2}\end{aligned}\quad (3.40)$$

$$\begin{aligned}V &= (m_2 g \mathbf{S}_2^T + m_3 g \mathbf{S}_3^T + m_4 g \mathbf{S}_4^T) \mathbf{e}_y \\ &= g (m_2 p_2 + m_3 (0.5 p_2 + 0.5 p_4) + m_4 p_4).\end{aligned}\quad (3.41)$$

Unter Nutzung der Lagrange-Multiplikatoren  $\lambda_j$  lassen sich die Systemgleichungen



ausdrücken:

$$\begin{aligned}
& -\frac{\lambda_1 p_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} + \frac{\lambda_1 q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} - \frac{\lambda_2 p_1}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} \\
& + \frac{\lambda_2 p_3}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} + m_2 \ddot{p}_1 + 0.25 m_3 \ddot{p}_1 + 0.25 m_3 \ddot{p}_3 = 0
\end{aligned} \tag{3.42a}$$

$$\begin{aligned}
& g m_2 + 0.5 g m_3 - \frac{\lambda_1 p_2}{\sqrt{p_2^2 + (p_1 - q_1)^2}} - \frac{\lambda_2 p_2 + \lambda_2 p_4}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} \\
& + m_2 \ddot{p}_2 + 0.25 m_3 \ddot{p}_2 + 0.25 m_3 \ddot{p}_4 = 0
\end{aligned} \tag{3.42b}$$

$$\begin{aligned}
& \frac{\lambda_2 (p_1 - p_3)}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} + \frac{\lambda_3 (l_0 - p_3 + q_2)}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} \\
& + \frac{m_3 (0.5 \ddot{p}_1 + 0.5 \ddot{p}_3)}{2} + m_4 \ddot{p}_3 = 0
\end{aligned} \tag{3.42c}$$

$$\begin{aligned}
& 0.5 g m_3 + g m_4 + \frac{\lambda_2 p_2}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} - \frac{\lambda_2 p_4}{\sqrt{(p_1 - p_3)^2 + (p_2 - p_4)^2}} \\
& - \frac{\lambda_3 p_4}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} + 0.25 m_3 \ddot{p}_2 + 0.25 m_3 \ddot{p}_4 + m_4 \ddot{p}_4 = 0
\end{aligned} \tag{3.42d}$$

$$\begin{aligned}
& \frac{\lambda_1 (p_1 - q_1) + \sqrt{p_2^2 + (p_1 - q_1)^2} (m_1 \ddot{q}_1 - \tau_1)}{\sqrt{p_2^2 + (p_1 - q_1)^2}} = 0 \\
& - \frac{\lambda_3 (l_0 - p_3 + q_2) + \sqrt{p_4^2 + (l_0 - p_3 + q_2)^2} (m_5 \ddot{q}_2 - \tau_2)}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} = 0.
\end{aligned} \tag{3.42e}$$

$$\begin{aligned}
& \frac{\lambda_1 (p_1 - q_1) + \sqrt{p_2^2 + (p_1 - q_1)^2} (m_1 \ddot{q}_1 - \tau_1)}{\sqrt{p_2^2 + (p_1 - q_1)^2}} = 0 \\
& - \frac{\lambda_3 (l_0 - p_3 + q_2) + \sqrt{p_4^2 + (l_0 - p_3 + q_2)^2} (m_5 \ddot{q}_2 - \tau_2)}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} = 0.
\end{aligned} \tag{3.42f}$$

$$\begin{aligned}
& - \frac{\lambda_3 (l_0 - p_3 + q_2) + \sqrt{p_4^2 + (l_0 - p_3 + q_2)^2} (m_5 \ddot{q}_2 - \tau_2)}{\sqrt{p_4^2 + (l_0 - p_3 + q_2)^2}} = 0.
\end{aligned} \tag{3.42g}$$

### 3.6 Systemidentifikation

Die meisten in den Modellen vorkommenden Systemparameter sind geometrischer Natur oder Massen und lassen sich durch Messen mit einem Lineal oder einer Wage bestimmen. Nur für das Trägheitsmoment der Last  $J$  ist eine direkte experimentelle Bestimmung (z.B. durch einen Pendelversuch) zu aufwendig. Daher wird dieser Parameter, unter der Annahme eines Quaders mit homogener Masseverteilung und Rotationsachse durch seinen Mittelpunkt, die in die Ebene zeigt, berechnet [12]:

$$J = \frac{1}{12} m_2 ((2s_2)^2 + h^2). \tag{3.43}$$

Dabei entspricht  $h$  der Höhe der Last, welche zu  $h = 90$  mm bestimmt wurde. 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
Initialer Laufkatzenabstand	$l_0$	0,3 m
Trägheitsmoment Last	$J_2$	0,00455 kgm <sup>2</sup>

# Kapitel 4

## Flachheitsanalyse

### 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  $\mathbf{y} := (y_1, \dots, y_m)^T$  sowie glatte Funktionen  $\Psi, \theta$  existieren, so dass:

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

$$\mathbf{u} = \theta(\mathbf{y}, \dot{\mathbf{y}}, \dots, \mathbf{y}^{(n_u)}) \text{ mit } n_u < \infty. \quad (4.2)$$

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. [9, S. 185]

### 4.2 Flachheitsanalyse von MIMO-Systemen

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 [3] verwiesen.

Es sei ein nichtlineares Mehrgrößensystem der Form aus Abschnitt 4.1 gegeben. Dessen Eingang  $\mathbf{u}$  kommt in affiner Form vor und könne mittels der Systemgleichungen eliminiert werden, so dass ein autonomes System aus  $p = n - m$  Gleichungen folgt. Diese Gleichungen werden wiederum zur Elimination  $p$  übriger Zustandskomponenten genutzt, so dass sich ein flacher Ausgang  $\mathbf{y}$  der Dimension  $n - p = m$  ergibt.

Zur Auswahl einer geeigneten Systemgleichung und Eingangskomponente für die Elimination bietet sich die Bildung der Jacobi-Matrix  $\mathbf{J}_i$  der **zu diesem Zeitpunkt noch  $i$  verbleibenden Systemgleichungen** bezüglich des Eingangs  $\mathbf{u}_{m-(n-i)}$  an. 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:

$$\begin{pmatrix} * & \cdots & * & 0 & * & \cdots & * \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ * & \cdots & * & 0 & * & \cdots & * \\ * & \cdots & * & \varepsilon & * & \cdots & * \\ * & \cdots & * & 0 & * & \cdots & * \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ * & \cdots & * & 0 & * & \cdots & * \end{pmatrix}. \quad (4.3)$$

Falls keine solcher Spalten zu finden ist, kann eine Transformation  $\mathbf{T}_i$  der Systemgleichungen durchgeführt werden, welche mittels linkem Orthokomplement  $\mathbf{J}_i^{L\perp}$  und linker Pseudoinverser  $\mathbf{J}_i^{L+}$  der Jacobi-Matrix das System in eine geeignete Form überführt [3, Abschnitt 2.1.2]:

$$\mathbf{T}_i \mathbf{J}_i = \begin{pmatrix} \mathbf{I}_{m-(n-i)} \\ \mathbf{0}_{(n-m) \times (m-(n-i))} \end{pmatrix}. \quad (4.4)$$

Wobei  $\mathbf{T}_i$  über die Eigenschaften von  $\mathbf{J}_i^{L+}$  und  $\mathbf{J}_i^{L\perp}$  folgendermaßen gebildet werden kann:

$$\mathbf{J}_i^{L+} \mathbf{J}_i = \mathbf{I}_i \quad (4.5)$$

$$\mathbf{J}_i^{L\perp} \mathbf{J}_i = \mathbf{0}_{(n-m) \times (m-(n-i))} \quad (4.6)$$

$$\Rightarrow \mathbf{T}_i = \begin{pmatrix} \mathbf{J}_i^{L+} \\ \mathbf{J}_i^{L\perp} \end{pmatrix}. \quad (4.7)$$

Nachdem die Eingangskomponenten aus den Systemgleichungen eliminiert wurden, kann die Behandlung der anderen noch vorhandenen Systemgrößen analog durch Bildung der Jacobi-Matrix bezüglich dieser vorgenommen werden. Der flache Ausgang  $\mathbf{y}$  besteht aus der Menge an Systemgrößen, welche im Idealfall nach Elimination aller Gleichungen übrig ist. Der Flachheitsnachweis für ein System ist erbracht, wenn mittels der Zusammenhänge aus den Systemgleichungen entsprechend Abschnitt 4.1 alle Systemgrößen (insbesondere auch der Systemeingang) allein durch  $(\mathbf{y}, \dot{\mathbf{y}}, \ddot{\mathbf{y}}, \dots)$  parametrisiert ist.

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

Anm. bez. Parametrisierung vs Parametrierung: Ich hatte mal gelesen, dass ersteres für die symbolische Darstellung mit Parametern genutzt wird, zweiteres für die Belegung der Parameter durch Werte. Ich finde allerdings gerade keine Quelle dafür ...

### 4.3 Anwendung Flachheitsanalyse am Einzelkran

**BITTE ABSCHNITT NOCH EINMAL KOMPLETT LESEN.**

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [\[14, flatness\\_notebooks/ODE\\_flatness\\_analysis\\_single\\_crane.ipynb\]](#) zu finden.

Aus der analytischer Modellbildung des Einzelkransystems folgen die 3 Systemgleichungen (3.20a), (3.20b) und (3.20c), welche als Grundlage des Flachheitsnachweises dienen.

Zunächst wird die Jacobi-Matrix dieser Gleichungen bezüglich  $\tau$  gebildet:

$$\mathbf{J}_3 = \begin{pmatrix} 0 & -\frac{p_1 - q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \\ 0 & -\frac{p_2}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \\ -1 & \frac{p_1 - q_1}{\sqrt{p_2^2 + (p_1 - q_1)^2}} \end{pmatrix}. \quad (4.8)$$

Dabei ist zu erkennen, dass in der ersten Spalte der Jacobi-Matrix die Eingangskomponente  $\tau_1$  isoliert vorkommt, also durch die korrespondierende dritte Systemgleichung (3.20c) bestimmt werden kann. Dementsprechend kann die letzte Zeile von  $\mathbf{J}_3$  eliminiert werden. Eine Parametrisierung  $\tau_1 = \text{func}(p_1, p_2, q_1, \ddot{q}_1, \tau_2)$  ist möglich.

Die verbleibenden ersten beiden Systemgleichungen (3.20a) und (3.20b) enthalten jeweils die Eingangskomponente  $\tau_2$ . So kann ohne Einschränkung auf diese Wahl die zweiten Gleichung (3.20b) zur Elimination von  $\tau_2$  genutzt werden. Eine Parametrisierung  $\tau_2 = \text{func}(p_1, p_2, \ddot{p}_2, q_1)$  erfolgt.

Die letzte verbleibende Systemgleichung (3.20a) weist nach Substitution des zuvor eliminierten  $\tau_1$  und  $\tau_2$  nur noch die Systemgrößen bzw. Ableitungen  $p_1, \ddot{p}_1, p_2, \ddot{p}_2, q_1$  auf. Daher bietet sich die Elimination der allein algebraisch auftretenden Größe  $q_1$  an. Daher folgt als Kandidat für einen flachen Ausgang  $\mathbf{y} = (p_1, p_2)^T$ . Mit diesem ist nun die konkrete Parametrisierung aller Systemgrößen durchzuführen.

Eine erste Parametrisierung von  $\tau_2$  ergibt sich zu:

$$\tau_2 = \frac{m_2 (g + \ddot{p}_2) \sqrt{p_2^2 + (p_1 - q_1)^2}}{p_2}. \quad (4.9)$$

Es ist zu bemerken, dass dabei die Koordinate  $q_1$  neben den Komponenten des flachen Ausgangs und seinen Ableitungen enthalten ist. Durch Einsetzen dieser vorläufigen Darstellung von  $\tau_2$  in Gleichung (3.20a) ergibt eine flache Parametrisierung von  $q_1$ :

$$q_1 = \frac{p_1 (g + \ddot{p}_2) - p_2 \ddot{p}_1}{g + \ddot{p}_2}. \quad (4.10)$$

Durch Einsetzen davon in Gleichung erfolgt eine flache Parametrisierung von  $\tau_2$ :

$$\tau_2 = \frac{m_2 \sqrt{\frac{p_2^2 (g + \ddot{p}_2)^2 + (-gp_1 - p_1 \ddot{p}_2 + p_1 (g + \ddot{p}_2) + p_2 \ddot{p}_1)^2}{(g + \ddot{p}_2)^2}} (g + \ddot{p}_2)}{p_2}. \quad (4.11)$$

Um abschließend die zuerst eliminierte Eingangskomponente  $\tau_1$  ebenso auszudrücken, wird nach Gleichung (3.20c) die zweite zeitliche Ableitung  $\ddot{q}_1$  benötigt:

$$\begin{aligned} \ddot{q}_1 = & \frac{-2 \ddot{p}_2 (g + \ddot{p}_2) (p_1 \ddot{p}_2 - p_2 \ddot{p}_1 - \ddot{p}_1 \dot{p}_2 + \dot{p}_1 (g + \ddot{p}_2))}{(g + \ddot{p}_2)^3} \\ & + \frac{(g + \ddot{p}_2)^2 (p_1 \ddot{\ddot{p}}_2 - p_2 \ddot{\ddot{p}}_1 - 2 \ddot{p}_1 \dot{p}_2 + 2 \ddot{p}_2 \dot{p}_1 - \ddot{p}_1 \ddot{p}_2 + \ddot{p}_1 (g + \ddot{p}_2))}{(g + \ddot{p}_2)^3} \\ & - \frac{(p_1 (g + \ddot{p}_2) - p_2 \ddot{p}_1) (\ddot{\ddot{p}}_2 (g + \ddot{p}_2) - 2 \ddot{\ddot{p}}_2^2)}{(g + \ddot{p}_2)^3}. \end{aligned} \quad (4.12)$$

Daraus folgt:

$$\begin{aligned} \tau_1 = & \frac{-m_1 2 \ddot{p}_2 (g + \ddot{p}_2) (p_1 \ddot{p}_2 - p_2 \ddot{p}_1 - \ddot{p}_1 \dot{p}_2 + \dot{p}_1 (g + \ddot{p}_2))}{(g + \ddot{p}_2)^3} \\ & + \frac{-m_1 (g + \ddot{p}_2)^2 (-p_1 \ddot{\ddot{p}}_2 + p_2 \ddot{\ddot{p}}_1 + 2 \ddot{p}_1 \dot{p}_2 - 2 \ddot{p}_2 \dot{p}_1 + \ddot{p}_1 \ddot{p}_2 - \ddot{p}_1 (g + \ddot{p}_2))}{(g + \ddot{p}_2)^3} \\ & + \frac{-m_1 (p_1 (g + \ddot{p}_2) - p_2 \ddot{p}_1) (\ddot{\ddot{p}}_2 (g + \ddot{p}_2) - 2 \ddot{\ddot{p}}_2^2)}{(g + \ddot{p}_2)^3} + \frac{m_2 \ddot{p}_1 (g + \ddot{p}_2)^3}{(g + \ddot{p}_2)^3} \end{aligned} \quad (4.13)$$

Die Parametrisierung aller Systemgrößen durch  $\mathbf{y} = (p_1, p_2)^T$  zeigt, dass es sich bei diesen um einen flachen Ausgang handelt.

## 4.4 Anwendung Flachheitsanalyse am Doppelkran

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [14, flatness\_notebooks/ODE\_flatness\_analysis.ipynb] zu finden.

Aus der analytischer Modellbildung des Doppelkransystems mit dem Lagrange-Formalismus zweiter Art folgen die 5 Systemgleichungen (3.29a) bis (3.29e), welche als Grundlage des Flachheitsnachweises dienen.

Zunächst wird die Jacobi-Matrix dieser Gleichungen bezüglich  $\boldsymbol{\tau}$  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 jeweils isoliert vorkommen. Es ergeben sich zur Bestimmung der Größen  $\tau_1$  und  $\tau_2$  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  $\tau_3, \tau_4$  gibt es in der darauf bezogenen Jacobimatrix  $\mathbf{J}_3$  zu den ersten drei Systemgleichungen (3.29a), (3.29b) und (3.29c) 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:

$$\mathbf{J}_3^{L+} \mathbf{J}_3 = \mathbf{I}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (4.14)$$

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

$$\mathbf{J}_3^{L\perp} \mathbf{J}_3 = \mathbf{0}_{1 \times 2} = \begin{pmatrix} 0 & 0 \end{pmatrix}, \quad (4.15)$$

so dass eine Transformation  $\mathbf{T}_3$  der übrigen Systemgleichungen erneut Spalten einer korrespondierenden Jacobi-Matrix impliziert mit jeweils nur einem konstanten nicht-Null-Eintrag:

$$\mathbf{T}_3 \mathbf{J}_3 = \begin{pmatrix} \mathbf{J}_3^{L+} \\ \mathbf{J}_3^{L\perp} \end{pmatrix} \mathbf{J}_3 = \begin{pmatrix} \mathbf{I}_2 \\ \mathbf{0}_{1 \times 2} \end{pmatrix} = \mathbf{I}_{3 \times 2}. \quad (4.16)$$

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

$$\mathbf{J}_3^{L+} = \left( \begin{pmatrix} J_{3,(1,1)} & J_{3,(1,2)} \\ J_{3,(2,1)} & J_{3,(2,2)} \end{pmatrix}^{-1} \mathbf{0}_{2 \times 1} \right), \mathbf{J}_3^{L\perp} = \begin{pmatrix} J_{3,(2,1)}J_{3,(3,2)} - J_{3,(2,2)}J_{3,(3,1)} \\ -J_{3,(1,1)}J_{3,(3,2)} + J_{3,(1,2)}J_{3,(3,1)} \\ J_{3,(1,1)}J_{3,(2,2)} - J_{3,(1,2)}J_{3,(2,1)} \end{pmatrix}^T. \quad (4.17)$$

Aus der Multiplikation der somit gefundenen Transformationsmatrix  $\mathbf{T}_3$  mit den übrigen 3 Systemgleichungen lassen sich entsprechend der Anforderungen an ihre Konstruktion anschließend die noch übrigen Eingangskomponenten  $\tau_3$  und  $\tau_4$  sowie die beiden transformierten Systemgleichungen eliminieren. Eine vorläufige Parametrisierung  $\tau_3 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, q_1, q_2)$  und  $\tau_4 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, q_1, q_2)$  ist möglich.

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

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

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

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:

$$q_2 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1) \quad (4.19a)$$

$$\tau_1 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1, \ddot{q}_1) \quad (4.19b)$$

$$\tau_2 = \text{func}(p_1, \dot{p}_1, \ddot{p}_1, p_1^{(3)}, p_1^{(4)}, p_2, \dot{p}_2, \ddot{p}_2, p_2^{(3)}, p_2^{(4)}, p_3, \dot{p}_3, \ddot{p}_3, p_3^{(3)}, p_3^{(4)}, q_1, \dot{q}_1, \ddot{q}_1) \quad (4.19c)$$

$$\tau_3 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1) \quad (4.19d)$$

$$\tau_4 = \text{func}(p_1, \ddot{p}_1, p_2, \ddot{p}_2, p_3, \ddot{p}_3, q_1). \quad (4.19e)$$



# Kapitel 5

## Steuerungs- und Regelungsentwurf

### 5.1 Regelung zur Stabilisierung von Ruhelagen

- notwendig???

### 5.2 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. An den real vorhandenen Versuchsstand gibt es bisher keine formal spezifizierten Anforderungen bezüglich Grenzwerten von etwa Beschleunigungen oder Geschwindigkeiten, die bei der Abfahrt von Trajektorien auftreten dürfen.

Aufgrund der einfachen Einprägung 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 der Ruhelage  $(t_0, y_{i,0})$  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 [9, 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  $\alpha_i = 2N_i - 1$ , wobei  $N_i$  der Anzahl der Randbedingungen des jeweiligen Ausgangs entspricht:

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

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 `symtools` verwendet.

## 5.3 Trajektorienfolgeregelung

### 5.3.1 Vektorieller relativer Grad

Es wird ein Mehrgrößensystem mit  $m$  Ein- und Ausgangskomponenten  $u_1, \dots, u_m$  und  $y_1, \dots, y_m$  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  $\mathbf{f} : M \rightarrow \mathbb{R}^n$ ,  $\mathbf{g} : M \rightarrow \mathbb{R}^n$  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  $\mathbf{p} \in M$  den vektoriellen relativen Grad  $\mathbf{r} = (r_1, \dots, r_m)^T$ , falls:

1.  $L_{\mathbf{g}_j} L_{\mathbf{f}}^k h_i(\mathbf{x}) = 0$  für alle  $\mathbf{x}$  aus einer Umgebung von  $\mathbf{p}$  sowie für alle  $i, j \in \{1, \dots, m\}$  und  $k \in \{0, \dots, r_i - 2\}$  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  $\mathbf{x} = \mathbf{p}$  regulär ist.

[9, S. 194]

### 5.3.2 Statische Rückführung

Ziel dieses Ansatzes ist es, 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

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

unter Nutzung eines virtuellen Eingangs  $\mathbf{v} = (v_1, \dots, v_m)^T$  zu kompensieren. Die Zusammensetzung dieses virtuellen Eingangs ist in Abschnitt 5.3.3 erklärt. [9, 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 ein Eingang explizit auftritt. Hierbei werden die Zeitableitungen des Ausgangs durch Lie-Ableitungen entlang des Vektorfelds des Zustandsraummodells des Systems  $\delta := \dot{\mathbf{x}} = \mathbf{f} + \mathbf{g}\tau$ , welches in Gleichung (3.30) bestimmt wurde, rekursiv erzeugt:

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

In der 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.

$i$	1	2	3	4
$r_i$	2	2	2	2
explizites Auftreten von $\tau_j$ bei $y_i^{(r_i)}$	$\tau_3, \tau_4$	$\tau_3, \tau_4$	$\tau_3, \tau_4$	$\tau_1, \tau_3$
$y_j^{(k)}$ mit minimalem $k$ bis erstes Auftreten von $\tau_i$	$y_4^{(2)}$	$y_{1,2,3}^{(4)}$	$y_{1,2,3,4}^{(2)}$	$y_{1,2,3}^{(2)}$

Das Fehlen von  $\tau_2$  bei all diesen Ausgangsableitungen bedeutet für die Entkopplungsmatrix  $\Lambda$  gemäß Gleichung (5.4) eine Singularität, weil die ihre Spalte nur Nulleinträge enthält. 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  $\tau_j$  in den Komponenten der Ausgangsableitung  $y_i^{(r_i)}$  genau dann nicht explizit auftritt, also die  $j$ -te Spalte von  $\Lambda$  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 [9, S. 201]:

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

Analog gilt dies für den alternativen flachen Ausgang  $\tilde{\mathbf{y}} = (p_1, p_2, p_3, q_2)^T$  aufgrund des Fehlens von  $\tau_1$ , 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 können zur Regelung Ansätze mit einer dynamischen Erweiterung oder quasi-statischen Zustandsrückführung verfolgt werden.

### 5.3.3 Zustandsrückführung aus Fehlerdynamik

Nach [9, S. 195] kann eine lineare Fehlerdynamik bezüglich des Trajektorienfolgefehlers  $e_i := y_i - y_{i,\text{ref}}$  (mit  $y_{i,\text{ref}} = \text{func}(t)$  und nicht wie in der Literatur an dieser Stelle beschrieben bezüglich einer Abweichung von einem Festwert  $y_{i,\text{ref}} = \text{const.}$ ) angesetzt werden und damit 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.

### 5.3.4 Dynamische Erweiterung

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [14, flatness\_notebooks/ODE\_flatness\_trajectory\_control\_simulation\_dyn.ipynb] zu finden.

Die Entkopplungsmatrix  $\mathbf{\Lambda}$  hat eine Nullspalte, weil  $\tau_2$  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  $\tau_1$ ,  $\tau_3$  und  $\tau_4$  in allen Ausgangsableitungen frühzeitig zur Ableitungsordnung 2 auftreten. Tabelle 2 kann entnommen werden, dass  $\tau_2$  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  $\tau_2$  um jeweils zwei Integratoren vorzunehmen. [9, S. 200]

So treten die neuen Eingangskomponenten ebenso wie  $\tau_2$  das erste mal bei der Ableitungsordnung 4 explizit auf. Der neue vektorielle relative Grad folgt mit  $r_i = 2$  für

$i = 1, \dots, 4$ . 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  $\tau_2$  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 [9, 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 stets  $r_i = 4$  für  $i = 1, \dots, 4$ . Dieser relative Grad scheint wohldefiniert zu sein, da Bedingung 1 aus Abschnitt 5.3.1 erfüllt ist und für prototypische Trajektorien die Inverse der Entkopplungsmatrix stets existiert,  $\mathbf{\Lambda}$  also regulär ist. Dementsprechend ist eine Zustandsrückführung nach Gleichung (5.6) in Abschnitt 5.3.2 möglich. Dabei handelt es sich um eine dynamische Rückführung. Prototypische Trajektorien beim Übergang zwischen zwei Ruhelagen mit einem auszuregelnden Anfangsfehler sind in Abbildung 6 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 können.

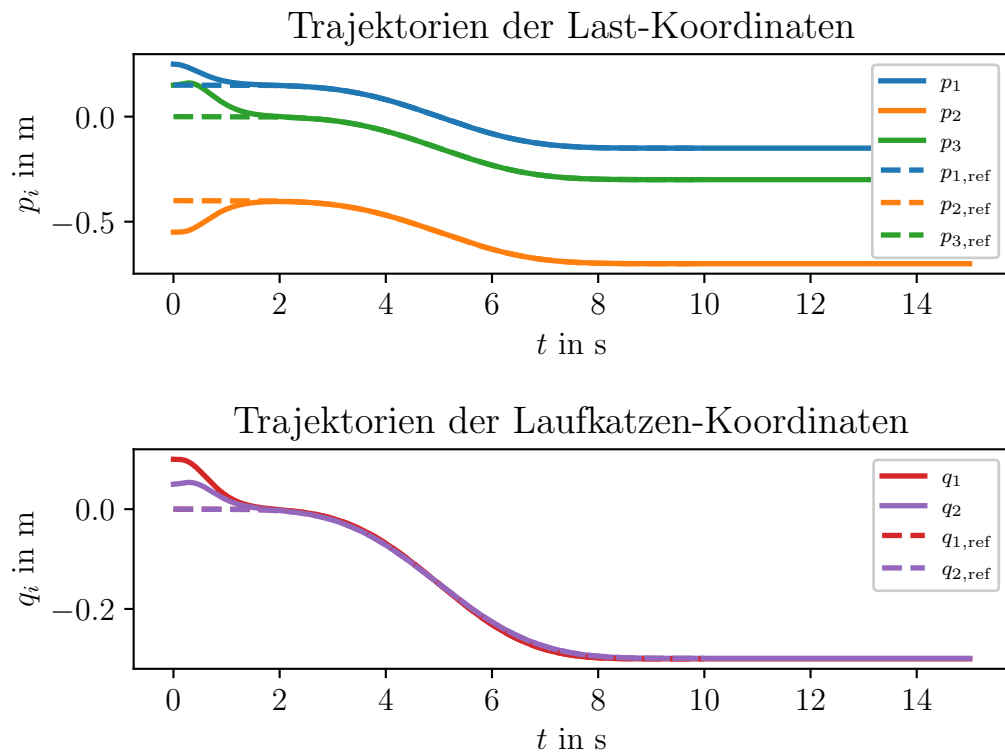
Auf die konkrete symbolische Angabe der Stellgesetze des Systemeingangs wird aus Praktikabilitätsgründen verzichtet und stattdessen auf das zu Beginn dieses Abschnitts 5.3.4 erwähnte Jupyter-Notebook verwiesen. Übersichtlicher ist an dieser Stelle eine Angabe von funktionalen Abhängigkeiten:

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

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

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

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



**Abbildung 6** – Langbeschreibung, die gerne auch über mehrer Zeilen gehen darf.

### Anpassung der Trajektorienplanung

Durch die Einführung von jeweils zwei Integratoren nach den neuen Stellgrößen  $\beta_1$ ,  $\beta_3$  und  $\beta_4$  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 5.2 kann erneut analog durchgeführt werden mit jeweils um vier Ordnungen höheren Polynomen der Ausgangstrajektorien.

### 5.3.5 Quasi-statische Rückführungen

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [14, 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 [9, 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  $\tau_j$  bei Ordnung  $k$  der Ausgangsableitung  $y_i$ .

$k$	$\tau_1$	$\tau_2$	$\tau_3$	$\tau_4$
$y_1^{(k)}$	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  $\tau_3$  tritt das erste mal 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  $\tau_4$  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 [9, S. 206] dargestellten Form, bei der für das Auftreten von  $u_2$  der Ausgang  $y_2$  noch weiter differenziert werden muss.  $\tau_2$  erscheint das erste Mal explizit bei  $r_i = 4$  in  $y_1^{(4)}$ ,  $y_2^{(4)}$  oder  $y_3^{(4)}$ ,  $\tau_1$  zuerst bei  $r_4 = 2$  in  $\ddot{y}_4$ .

Eine Definition der neuen virtuellen Eingänge  $v_i$  ist so möglich:

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

Daraus ist ersichtlich, dass durch ein Gleichungssystem aus  $v_1$ ,  $v_3$  und  $v_4$  in denen die Eingangskomponenten  $\tau_1$ ,  $\tau_3$  und  $\tau_4$  affin auftreten, eine explizite Darstellung dieser möglich ist. Dies steht im Gegensatz zur Situation in [9, S. 207], bei der die relativen Grade der Ausgangskomponenten aufsteigend sind, so dass kein Gleichungssystem gelöst werden muss, sondern sukzessives Einsetzen der vorher berechneten Eingänge und die algebraische Umformung der abgeleiteten Ausgänge genügt.

Die Berechnung von  $\tau_2$  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 von  $\tilde{\delta}$  generiert werden können. Es folgt ein relativ umfangreicher Ausdruck für  $\tau_2$ , der zu diesem Zeitpunkt mehr als einhunderttausend Rechenoperationen enthält.

Nach [9, S. 208] kann eine lineare Fehlerdynamik  $\mathbf{v}$  entsprechend des jeweiligen relativen Grades der Ausgangskomponente zur Stabilisierung des Systems angesetzt werden. Für  $v_1$ ,  $v_3$  und  $v_4$  diese jeweils nur von Zustandskomponenten  $\tilde{\mathbf{x}}$  abhängig,  $v_2$  allerdings auch von  $\ddot{y}_2 = \ddot{p}_2$  und  $y_2^{(3)} = p_2^{(3)}$ , die 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  $\tau_2$  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 Nutzung 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 des Systemeingangs wird aus Praktikabilitätsgründen verzichtet und stattdessen auf das zu Beginn dieses Abschnitts 5.3.5 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 aus numerischen Effekten. Beim Einsetzen einer zuvor ermittelten Ruhelage entstehen Definitionslücken, sowohl Zähler als auch Nenner der Eingangsgrößen  $\tau$  werden Null. Es ist nicht auszuschließen, dass diese Lücken durch algebraische Manipulation hebbar 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 Betrachtung eignen sich die einfacheren Terme der Nenner  $N_{1,3,4}$  von  $\tau_1$ ,  $\tau_3$  und  $\tau_4$ :

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

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

#### 5.3.6 Exact feedforward linearization

Eine Implementierung der in diesem Abschnitt durchgeführten Überlegungen ist unter [\[14, flatness\\_notebooks/ODE\\_flatness\\_trajectory\\_control\\_simulation.ipynb\]](#) zu finden.

Bei der Exact feedforward linearization wird ein Ansatz verfolgt, bei dem statt einer Kompensation der Nichtlinearität mittels Rückführung der gemessenen beziehungsweise simulierten Zustandskomponenten wie bei der exakten Eingangs-Ausgangs-Linearisierung die Referenztrajektorie eingesetzt wird [\[4\]](#).

Heuristisch wird eine Fehlerdynamik der Ordnung zwei für alle Komponenten des Koordinatenvektors angesetzt:

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

Aus der Zusammensetzung des Zustandsvektors  $\mathbf{x} = (\theta, \dot{\theta})^T$  und der eingangsaffinen Zustandsraumdarstellung kann außerdem der Zusammenhang:

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

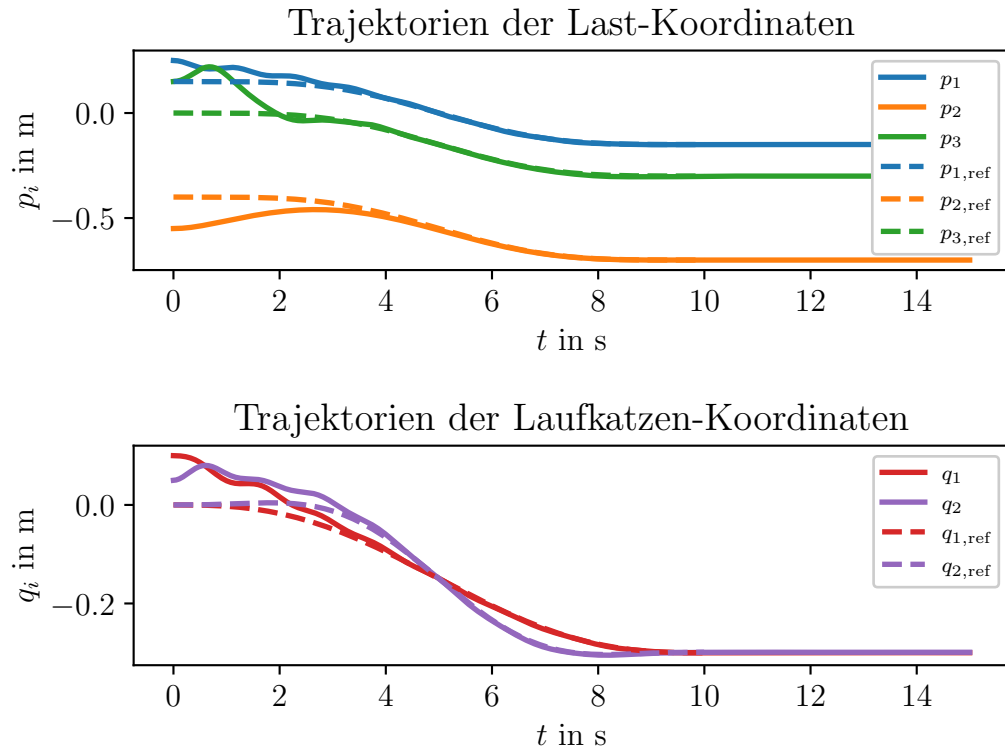
hergestellt werden. Dabei bedeutet die Indizierung  $\bullet_{[i,j]}$ , die Auswahl der Zeilen  $i$  bis  $j$  von  $\bullet$ . Da für den Systemeingang  $\tau \in \mathbb{R}^4$  und für die Eingangsmatrix  $\mathbf{g}_{[6,10]} \in \mathbb{R}^{5 \times 4}$  gilt, kann dieser Zusammenhang über die Bildung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

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

Durch Einsetzen von  $\theta_{\text{ref}}$  in  $\mathbf{g}_{[6,10]}^+(\theta)$  und  $\mathbf{f}_{[6,10]}(\theta)$  wird die Exact feedforward linearization realisiert:

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

Abbildung 7 zeigt prototypische Trajektorien beim Übergang zwischen zwei Ruhelagen mit einem auszuregelnden Anfangsfehler für diesen Regelungsansatz.



**Abbildung 7** – Langbeschreibung, die gerne auch über mehrer Zeilen gehen darf.

### Vereinfachung mittels Ausgangsselektion

Statt der Nutzung einer Pseudo-Inversen wird im Folgenden eine Selektion von vier der fünf Gleichungen der Beschleunigungen  $\ddot{\theta}$  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 4 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 8 zeigt, dass sich für Beispieltrajektorien mit Anfangsfehlern ebenso wie bei den vorherigen Ansätzen ein sehr gutes Folgeverhalten sowie eine stationäre Genauigkeit ergibt.

Auf die konkrete symbolische Angabe der Stellgesetze des Systemeingangs wird aus Praktikabilitätsgründen dennoch verzichtet, da jede Eingangskomponente trotzdem mehr als einhundert Rechenoperationen beinhaltet. Stattdessen wird auf das zu Beginn dieses Abschnitts 5.3.6 erwähnte Jupyter-Notebook verwiesen. Übersichtlicher ist an dieser Stelle eine Angabe von funktionalen Abhängigkeiten:

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

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

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

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

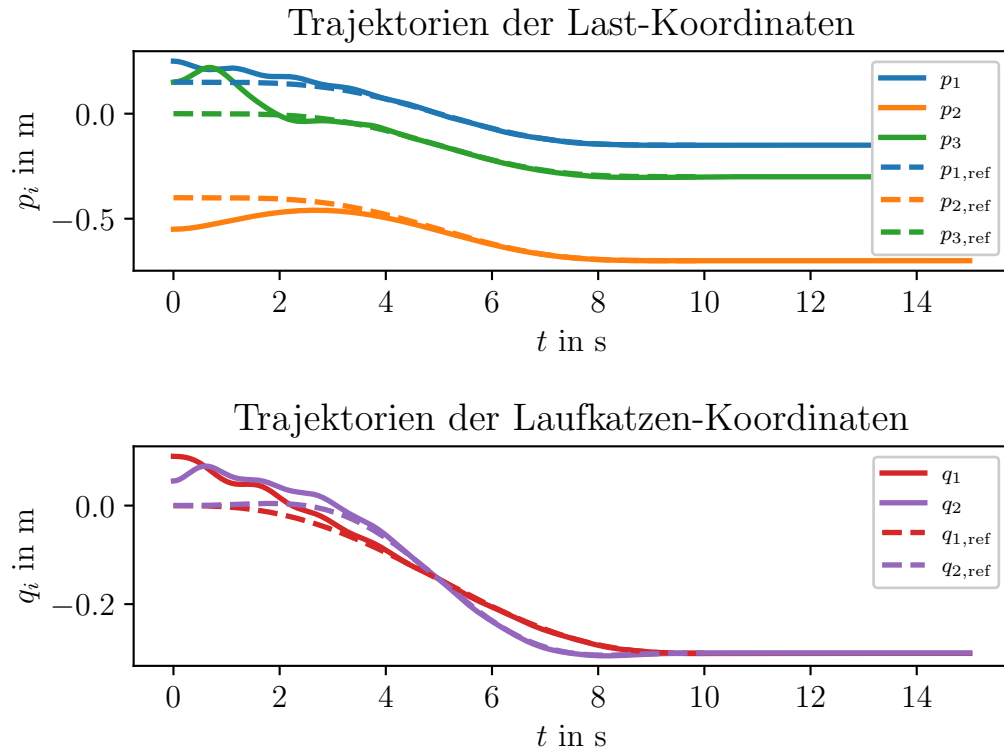
## Stabilitätsbetrachtung

Die Betrachtung einer prototypischen Trajektorien zeigt, dass zu späteren Zeitpunkten der Simulation mit dem zuletzt vorgestellten Regelungsansatz Eigenwerte der Jacobimatrix  $\mathbf{J}_{\dot{\mathbf{e}}}$  einer Fehlerdynamik:

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

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

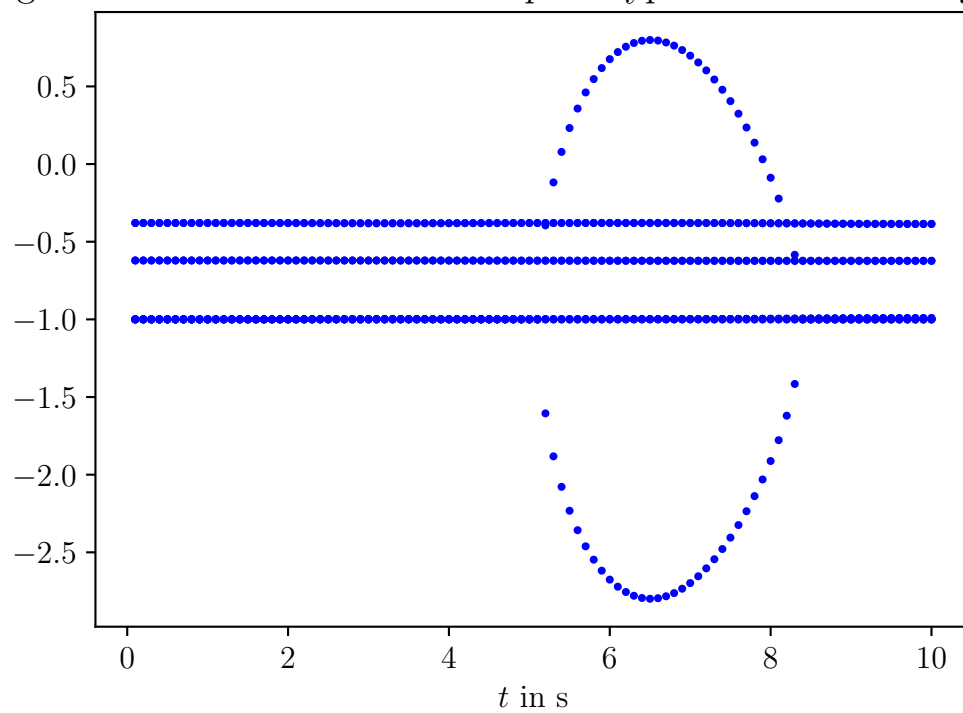
$$\mathbf{J}_{\dot{\mathbf{e}}} = \frac{\partial \dot{\mathbf{e}}}{\partial \mathbf{e}} \quad (5.31)$$



**Abbildung 8** – Langbeschreibung, die gerne auch über mehrer Zeilen gehen darf.

mit positiven Realteilen vorkommen. Abbildung 9 veranschaulicht dies. Es ist möglich, dass aufgrund des bis dahin abgeklungenen Folgefehlers keine weitere Destabilisierung des Systems stattfindet, sondern eine stationäre Genauigkeit dennoch erreicht wird. Bis zum Ende der Simulationen 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 [13, Vorlesung 4]. Eine andere Wahl von Rückführung (keine lineare Dynamik) könnte dieses Problem lösen.

Eigenwerte der Jacobimatrix bei prototypischer Referenztrajektorie



**Abbildung 9** – Langbeschreibung, die gerne auch über mehrer Zeilen gehen darf.

# Kapitel 6

## Fazit und Ausblick

### 6.1 Fazit

bla

### 6.2 Ausblick

blub

# Literatur

- [1] Christian Andersson, Claus Führer und Johan Åkesson. „Assimulo: A unified framework for  $\{\text{ODE}\}$  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] Torsten Fließbach. *Mechanik : Lehrbuch zur Theoretischen Physik I*. Spektrum Akademischer Verlag, 2009. DOI: [https://doi.org/10.1007/978-3-8274-2188-3\\_8](https://doi.org/10.1007/978-3-8274-2188-3_8).
- [3] 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>.
- [4] 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>.
- [5] Home page Fraunhofer IIS/EAS. Fraunhofer-Institut fuer Integrierte Schaltungen IIS, Institutsteil Entwicklung Adaptiver Systeme EAS. Juni 2022. URL: <https://www.eas.iis.fraunhofer.de/>.
- [6] Klaus Janschek. „Skript Elemente der Modellbildung und Simulationstechnik Sommersemester 2020“. In: (2020).
- [7] Klaus Janschek. *Systementwurf mechatronischer Systeme Methoden - Modelle - Konzepte*. Berlin: Springer, 2010. ISBN: 9783540788768. URL: [http://slubdd.de/katalog?TN\\_libero\\_mab2\)500406929](http://slubdd.de/katalog?TN_libero_mab2)500406929).
- [8] 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>.
- [9] Klaus Röbenack. *Nichtlineare Regelungssysteme*. Springer Vieweg Berlin, Heidelberg, 2017. DOI: <https://doi.org/10.1007/978-3-662-44091-9>.
- [10] *symbtools GitHub Repository*. Juni 2022. URL: <https://github.com/TUD-RST/symbtools>.
- [11] Bruce Torby. „Energy Methods“. In: *Advanced Dynamics for Engineers* (1984).



- 
- [12] Trägheitsmoment. 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..>
- [13] Jan Winkler. „Skript Nichtlineare Regelungstechnik 1 Sommersemester 2021“. In: (2021).
- [14] Konstantin Wrede. GitHub Repository zu dieser Studienarbeit. Sep. 2022. URL: <https://github.com/wredsen/SA-Wrede>.