



Kollaborative Synchronisation eines 7-Achs-Roboters

DIPLOMARBEIT

Ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Diplom-Ingenieurs (Dipl.-Ing.)

unter der Leitung von

Univ.-Prof. Dr. techn. A. Kugi
Dipl.-Ing. M. Schwegel

eingereicht an der

Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik
Institut für Automatisierungs- und Regelungstechnik

von

Thomas Weingartshofer
Matrikelnummer 01126457
Seyringer Straße 19/1/50
1210 Wien
Österreich

Wien, am 11. November 2018

Danksagung

Ich möchte ganz besonders meinem Betreuer Dipl.-Ing. Michael Schwegel danken, der mir während meiner gesamten Arbeit zur Seite stand. Nicht nur seine Expertise, sondern auch sein kollegialer Umgang waren Anreize für etliche Diskussionen während meiner Arbeit. Außerdem möchte ich mich für die schnellen und hilfreichen Anmerkungen zu diesem Dokument bedanken.

Weiters danke ich Dipl.-Ing. Christian Hartl-Nesic für seine Hilfsbereitschaft und etliche Anmerkungen durch seine große Erfahrung mit Industrierobotern und dem verwendeten optischen Messsystem. Außerdem möchte ich Dr. techn. Tobias Glück für seine Ideen und Bemerkungen zu diesem Projekt danken. Zusätzlich danke ich all meinen Kollegen, die mir durch zahlreiche Gespräche einen anderen Blickwinkel auf gewisse Probleme gaben.

Ich danke speziell Institutsvorstand Univ.-Prof. Dr. techn. Andreas Kugi, dem merkbar die Lehre und Förderung spannender Projekte ein wichtiges Anliegen ist.

Ein ganz spezieller Dank geht an meine Eltern, die mich nicht nur während meiner Studienzeit jederzeit unterstützen. Dank ihrer Unterstützung wurde ich nicht nur immer aufs neue motiviert, ich konnte mich auch voll und ganz auf mein Studium konzentrieren.

Wien, am 11. November 2018

Abstract

Conveyor belts are often used to transport products in automated manufacturing. This work focuses on synchronizing a process sequence of an industry robot with a moving product. Using such a synchronization motion, the process sequence can be performed during transportation on a conveyor belt.

For this task, a mathematical model of a redundant collaborative robot with flexible joints is derived in a first step. Based on this model a control strategy is developed to synchronize the end-effector with a moving object. Additionally to the synchronized movement of the robot a path is planned to represent the process sequence of the robot. A collaborative scenario is investigated which includes a worker who carries the object. The path progress is dynamically monitored in order to stop the process sequence, if the object is moving too fast. Hereby erroneous process sequences are avoided. A fixed camera system is used to measure the position and orientation of the moving object. It is shown that a good accuracy of the process sequence on the moving object is achieved. Finally, the properties of a fixed camera system for this use case are discussed in more detail.

Kurzzusammenfassung

In der automatisierten Fertigung ist die Verwendung von Förderbändern für den Transport von Produkten weitverbreitet. In dieser Arbeit wird der Arbeitsablauf eines Industrieroboters mit einem bewegten Produkt synchronisiert. Somit können die Fertigungsschritte während des Transports durchgeführt werden.

Dafür wird in einem ersten Schritt ein mathematisches Modell eines kollaborativen redundanten Roboters mit flexiblen Gelenken berechnet. Auf Basis dieses Modells wird ein Regelungskonzept entwickelt, das den Endeffektor des Roboters mit dem bewegten Objekt synchronisiert. Zusätzlich zu der synchronisierten Bewegung wird ein Pfad auf dem Objekt geplant, der einem Arbeitsablauf entspricht. Des Weiteren wird ein kollaboratives Szenario untersucht, bei dem eine Person das Objekt führt. Damit bei zu schnellen Bewegungen des Objekts der Arbeitsablauf angehalten werden kann, wird der Pfadfortschritt dynamisch überwacht. Für die Messung der Position und Orientierung des Objekts wird ein fixes Kamerasystem verwendet. Es wird eine gute Genauigkeit des Pfades am bewegten Objekt erzielt. Abschließend werden die Eigenschaften bei der Verwendung von fixen Kamerasystemen für diese Art von Anwendung näher diskutiert.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Terminologie	2
1.2	Literaturstudie	2
1.3	Ziele der Arbeit	3
1.4	Gliederung der Arbeit	4
2	Mathematisches Modell	5
2.1	Kinematik	5
2.1.1	Direkte Kinematik	5
2.1.2	Jacobi-Matrizen	8
2.2	Dynamik	9
2.2.1	Dynamik mit starren Gelenken	9
2.2.2	Dynamik mit flexiblen Gelenken	11
2.3	Modellierung der Gleitreibung	12
2.4	Mathematisches Modell für den Reglerentwurf	15
3	Regelungskonzept	16
3.1	Regelgesetz der inversen Dynamik	16
3.1.1	Stabilität der Dynamik des Positionsfehlers	17
3.1.2	Stabilität der Dynamik des Orientierungsfehlers	18
3.2	Regelung der Nulldynamik	19
4	Pfadplanung	21
4.1	Pfade im körperfesten Koordinatensystem der Tafel	21
4.1.1	Ebene Pfade	21
	Lissajous-Kurve	22
	Kreiskurve	22
4.1.2	Räumliche Pfade	24
4.1.3	Pfade auf gekrümmten Flächen	25
4.2	Messung der Pose der Tafel	27
4.2.1	Transformation der Messergebnisse	28
	Transformation für ebene Tafel	28
	Transformation für zylindrische Tafel	29
	Transformation in Basis-Koordinatensystem des Roboters	29
	Kombination der Transformationen der Messung	31
4.2.2	Filterung	31
	Filterung der Position	31
	Filterung der Orientierung	32

Diskretisierung des Filters	32
Darstellung als homogene Transformation	32
4.3 Bestimmung der gewünschten Pose des Endeffektors	33
4.4 Dynamischer Pfadfortschritt	33
5 Ergebnisse	37
5.1 Linearbewegung der Tafel	37
5.2 Bewegung der Tafel im Raum	43
5.2.1 Lissajous-Kurve auf ebener Tafel	43
5.2.2 Unterbrochene Kreiskurve auf ebener Tafel	48
5.2.3 Lissajous-Kurve auf zylindrischer Tafel	53
5.3 Ergebnisse auf der Tafel	58
6 Zusammenfassung	60
A Anhang	62
A.1 Herleitungen	62
A.1.1 Quaternionenprodukt	62
A.1.2 Winkelgeschwindigkeit aus Quaternionen	62
A.1.3 Ableitung der Lyapunov Funktion	63
A.2 Parameter	64
A.3 Stifthalterung	66

Abbildungsverzeichnis

1.1	Arbeitsablauf des Roboters unter Verwendung der CAD-Modelle der Robotergelenke aus [1].	1
2.1	Koordinatensysteme des Roboters unter Verwendung der CAD-Modelle der Robotergelenke aus [1].	6
2.2	Schematischer Aufbau des Robotergelenks, vergleiche [23].	11
2.3	Ausschnitt der Messung der Gleitreibung für Gelenk 1.	13
2.4	Reibkurve für Gelenk 1.	14
2.5	Reibkurven der Gelenke 2, . . . , 7.	14
4.1	Tafel-Koordinatensystem Σ_l im Mittelpunkt der ebenen Tafel.	22
4.2	Lissajous-Kurve und Kreiskurve im Tafel-Koordinatensystem Σ_l	23
4.3	Pfad zum Abheben des Endeffektors von der Oberfläche der Tafel.	25
4.4	Tafel-Koordinatensystem Σ_l im Mittelpunkt der zylindrischen Tafel.	26
4.5	Projektion der Lissajous-Kurve auf der zylindrischen Tafel.	27
4.6	Transformation zum Tafel-Koordinatensystem Σ_l bei ebener Tafel.	29
4.7	Transformation zum Tafel-Koordinatensystem Σ_l bei zylindrischer Tafel.	30
4.8	Transformationen der Messung unter Verwendung der CAD-Modelle der Robotergelenke aus [1].	30
4.9	Veränderung des Fortschritts des Pfadparameters φ im Bereich 17 s bis 22 s.	35
5.1	Trajektorien bei Linearbewegung der Tafel.	39
5.2	Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 bei linearer Bewegung der Tafel.	40
5.3	Gelenkwinkel bei Linearbewegung der Tafel.	41
5.4	Drehmomente bei Linearbewegung der Tafel.	42
5.5	Pfade beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.	44
5.6	Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.	45
5.7	Gelenkwinkel beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.	46
5.8	Drehmomente beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.	47
5.9	Pfade beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.	49
5.10	Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.	50
5.11	Gelenkwinkel beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.	51
5.12	Drehmomente beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.	52
5.13	Pfade beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.	54

5.14	Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.	55
5.15	Gelenkwinkel beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.	56
5.16	Drehmomente beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.	57
5.17	Gezeichnete Ergebnisse auf der Tafel.	58
A.1	Modell der federgelagerten Stifthalterung.	66
A.2	Bauteilzeichnung der zwei Führungsstäbe für die Stifthalterung, Maße in mm.	67
A.3	Bauteilzeichnung der Stifthalterung, alle unbemaßten Radien R 5, Maße in mm.	68

1 Einleitung

In der automatisierten Fertigung bewegen sich Werkstücke häufig auf Förderbändern, um verschiedene Fertigungsstationen zu erreichen. Um Arbeiten in den Fertigungsstationen durchzuführen, können Industrieroboter verwendet werden. In dieser Arbeit wird der Arbeitsablauf eines Roboters mit einem bewegten Werkstück synchronisiert. Wenn der Arbeitsablauf während einer Bewegung des Werkstücks durchgeführt wird, kann sich dies zeitsparend auf den Fertigungsprozess auswirken.

Beispielhaft wird für diese Arbeit der Endeffektor (ein Stift) eines kollaborativen Roboters mit einer Tafel synchronisiert. Die Tafel wird dabei frei im Arbeitsraum des Roboters bewegt und durch ein Messsystem erfasst. Der Roboter synchronisiert sich mit der Bewegung der Tafel und führt relativ zu dieser einen Arbeitsablauf aus. Um den Pfad auf der Tafel sichtbar zu machen, wird als Endeffektor ein Stift befestigt. Eine schematische Darstellung dieses Aufbaus ist in Abbildung 1.1 gezeigt.

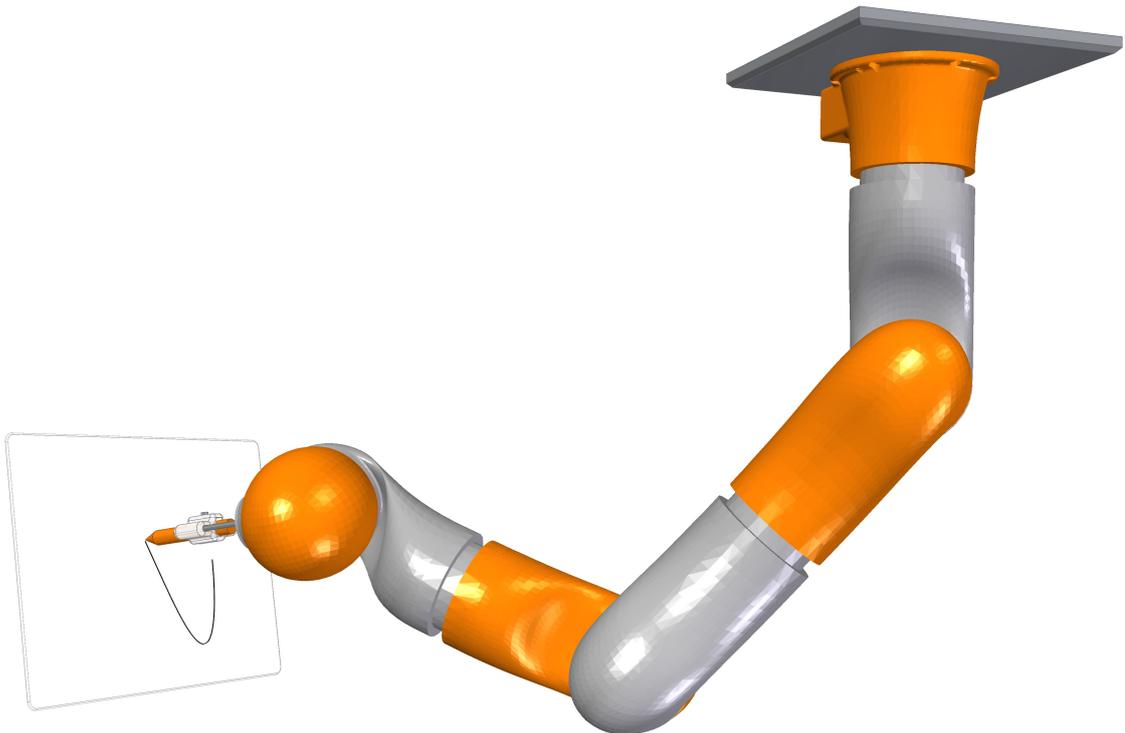


Abbildung 1.1: Arbeitsablauf des Roboters unter Verwendung der CAD-Modelle der Robotergelenke aus [1].

In einem ersten Schritt wird die Bewegung der Tafel durch eine Linearachse durchgeführt. Wird die Tafel im Weiteren durch einen Menschen bewegt, muss dies zusätzlich in der Entwicklung der Regelungsstrategie berücksichtigt werden. Die Präzision mit der der Tafel gefolgt wird, ist von einer genauen Messung der Tafelposition und -orientierung sowie der Absolutgenauigkeit des Roboters abhängig.

1.1 Terminologie

Der verwendete Begriff Synchronisierung und dessen Unterteilung wird in diesem Abschnitt erläutert.

In [2, 3] ist der Begriff der Synchronisierung als gegenseitige zeitliche Übereinstimmung von zwei oder mehreren Prozessen beschrieben. Außerdem ist darin ein kleiner geschichtlicher Überblick zu diesem Thema zu finden.

In [3] wird der Begriff der Synchronisierung in drei Arten unterteilt sowie mathematisch dargestellt. Tritt bei getrennten Systemen ohne Eingang eine Synchronität auf, wird dies als natürliche Synchronisation bezeichnet. Wenn die synchronisierten Systeme jedoch miteinander verbunden sind und es keine äußere Beeinflussung gibt, handelt es sich um Selbstsynchronisierung. Die dritte Art der Synchronisierung wird durch eine externe Beeinflussung beziehungsweise Verbindung erreicht und daher kontrollierte Synchronisierung genannt.

In [3, 4] wird die kontrollierte Synchronisierung weiter in eine interne (gegenseitige) und externe Synchronisierung unterteilt. Bei der internen Synchronisierung sind alle Teilnehmer des Gesamtsystems gleich gestellt und die Interaktion aller Teilnehmer führt zur Synchronität. Ein Beispiel dafür sind kooperative Systeme. Bei der externen Synchronisation hingegen wird der Anteil einiger Teilnehmer höher gewichtet z. B. koordinierte Systeme oder Master-Slave Systeme. Bei Master-Slave Systemen kann die Bewegung des Masters unabhängig von jener des Slaves erfolgen.

In [5] wird für ein Multi-Roboter System die interne und externe Synchronisierungsmethode verglichen. Ein Beispiel für koordinierte Synchronisierung beziehungsweise ein Master-Slave System ist in [6] gezeigt. Ähnlich dazu ist diese Arbeit einzuordnen, da die bewegte Tafel eine willkürliche Bewegung vollziehen kann, mit der der Roboter seine Bewegung synchronisieren muss.

Zusätzlich wird der Begriff kollaborierender Betrieb verwendet. Dieser beschreibt nach [7] einen Zustand, wenn ein Roboter mit einem Menschen im selben Arbeitsraum zusammen tätig ist.

1.2 Literaturstudie

Die Synchronisierung eines Industrieroboters ist ein viel beachtetes Thema in der Wissenschaft.

In [8, 9] wird untersucht, wie sich ein Roboterarm mit einem gleichmäßig bewegten Objekt synchronisieren kann. Dafür wird die Bewegung mittels eines Kalman-Filters geschätzt und der Arm mittels „Multi-surface Sliding Control“ beziehungsweise in Kombination mit

„Computed Torque“ geregelt. Für die Entwicklungen in dieser Arbeit wird die Bewegung der Tafel jedoch nicht zwingend als gleichmäßige Bewegung angenommen.

In [10] wird ein gemeinsames Modell für zwei Roboter beschrieben und dieses mittels exakter Linearisierung geregelt. Die Entwicklungen in dieser Arbeit sollen jedoch nicht auf die spezielle Bewegung der Tafel mittels einer Linearachse beschränkt sein, sondern erweitert werden können. Dadurch wird kein gemeinsames Modell des Roboterarms und der Linearachse verwendet.

Für medizinische Anwendungen, wie einem chirurgischen Eingriff im Ohr, wird in [11] ein „Sliding Mode Enhanced Adaptive Disturbance Observer“ beschrieben.

Eine weitere Möglichkeit mehrere Roboter zu synchronisieren wird in [12] gezeigt. Dafür wird der Fehler zur gewünschten Trajektorie mehrerer Roboter zu einem gemeinsamen Fehler kombiniert und zu null geregelt. In [4] wird ein Regelgesetz entwickelt, bei dem zusätzlich zum Trajektorienfehler die Synchronisationsfehler der Roboter zueinander verwendet werden.

Mehrere Arbeiten in Verbindung mit dem KUKA *LBR 4* wurden veröffentlicht. In [13] wird ein Pfadfolgeregler durch einen nichtlinearen modellprädiktiven Regler implementiert. Dabei werden jedoch im Versuch nur zwei Gelenke des Roboters als beweglich angenommen. Die Pfadvorgabe erfolgt im Konfigurationsraum. In [14] werden drei Gelenke verwendet und die Pfade werden im Arbeitsraum vorgegeben. In [15] wird ebenfalls ein modellprädiktiver Regler für die Pfadfolge jedoch mit zusätzlichem Kraftregler verwendet. Für praktische Versuche wird auch in diesem Beitrag der Roboter auf drei bewegliche Gelenke beschränkt. Die Parameter des Roboters werden in allen drei Arbeiten von [16] übernommen. Für die beschriebenen Konzepte folgt der Endeffektor dem vorgegebenen Pfad gut. Es werden jedoch nicht alle zur Verfügung stehenden Freiheitsgrade genutzt. Zudem wird der Pfad fest im Raum vorgegeben.

In [17] wird das Modell eines parallelen Roboters in Teilsysteme zerlegt, um eine Pfadfolgeregelung mit einer Nachgiebigkeitsregelung zu kombinieren.

1.3 Ziele der Arbeit

Die Ziele der Arbeit werden in diesem Abschnitt zusammengefasst. Es soll ein Roboter mit einem Stift als Endeffektor auf eine im Arbeitsraum bewegte Tafel zeichnen. Um das Verhalten des Roboters beschreiben zu können, soll ein mathematisches Modell entworfen werden. Zudem soll zur Ansteuerung des Roboters ein Regelungskonzept entwickelt werden. Das Regelungskonzept muss die Position und Orientierung des Endeffektors des Roboters regeln, wobei die Nulldynamik des Roboters berücksichtigt werden muss. Die Dynamik des Endeffektors soll im Bezug auf das Objekt vorgegeben werden können. Um Arbeitsabläufe auf dem bewegten Objekt zu simulieren, sollen verschiedene Pfade berechnet werden. Zusätzlich soll ein Pfad, der eine *pick and place* Aufgabe simuliert, entworfen werden. Um den Stift am Roboter zu befestigen, muss eine Halterung entwickelt werden. Zudem soll der Pfadfortschritt verändert werden können.

1.4 Gliederung der Arbeit

In Kapitel 2 wird das mathematische Modell für die weitere Verwendung im Regelungskonzept hergeleitet. Dabei wird in Abschnitt 2.1 auf die Kinematik des verwendeten Roboters eingegangen. Das dynamische Modell wird in Abschnitt 2.2 hergeleitet. In Abschnitt 2.3 wird die Reibung in den Gelenken bestimmt, bevor in Abschnitt 2.4 das verwendete Modell zusammengefasst wird.

Das Regelungskonzept wird in Kapitel 3 vorgestellt. Dafür wird in Abschnitt 3.1 ein Regelgesetz der inversen Dynamik des Roboters berechnet und die Stabilität der Fehlersysteme gezeigt. In Abschnitt 3.2 wird auf die Regelung der Nulldynamik eingegangen.

Um die Bewegungen des Roboters zu planen, werden in Kapitel 4 die erforderlichen Pfade bestimmt. Dazu werden in Abschnitt 4.1 Pfade zur Simulation der Arbeitsabläufe berechnet. In Abschnitt 4.2 wird die Messung und Verarbeitung der Pose der bewegten Tafel dargestellt. Im darauffolgenden Abschnitt 4.3 werden die vorgestellten Konzepte kombiniert. Die dynamische Veränderung des Pfadfortschritts wird in Abschnitt 4.4 hinzugefügt.

Die Ergebnisse dieser Arbeit werden in Kapitel 5 dargestellt. Dabei wird in Abschnitt 5.1 die Messung bei einer Bewegung der Tafel mit einer Linearachse gezeigt. Die Ergebnisse bei einer Bewegung der Tafel im Raum werden in Abschnitt 5.2 vorgestellt, wobei die mit dem Stift gezeichneten Pfade in Abschnitt 5.3 diskutiert werden.

Zuletzt wird in Kapitel 6 die Arbeit zusammengefasst und es werden mögliche Erweiterungen vorgeschlagen.

2 Mathematisches Modell

In diesem Kapitel wird die mathematische Beschreibung des verwendeten Roboters hergeleitet. Es handelt sich um den Roboter KUKA *LBR 4+*, der speziell für kollaborative Anwendungen entworfen wurde.

In Abschnitt 2.1 wird die Kinematik des Roboters beschrieben. Die Gleichungen zur Beschreibung der Dynamik werden in Abschnitt 2.2 gezeigt. In Abschnitt 2.3 wird auf die Identifikation der Reibung in den Gelenken eingegangen. Abschließend wird in Abschnitt 2.4 ein vereinfachtes Modell des Roboters dargestellt.

2.1 Kinematik

In Abschnitt 2.1.1 wird die direkte Kinematik des Roboters hergeleitet. Darauf aufbauend werden in Abschnitt 2.1.2 die Jacobi-Matrizen beschrieben.

2.1.1 Direkte Kinematik

In diesem ersten Abschnitt wird eine Funktion hergeleitet, die aus allen Gelenkwinkeln des Roboters die Pose des Endeffektors im Raum berechnet. Dieses Problem wird *direkte Kinematik* genannt, siehe [18]. Zu Beginn wird jedem Starrkörper des Roboters ein körperfestes Koordinatensystem zugeordnet. Ein zusätzliches Koordinatensystem ist an der Spitze des Endeffektors platziert. Der Roboter mit allen Koordinatensystemen ist in Abbildung 2.1 dargestellt.

Um die Anordnung der Koordinatensysteme Σ systematisch zu gestalten, sind diese wie in [19] nach der Denavit-Hartenberg (DH) Konvention gewählt, siehe [18]. Dabei wird

- die z_i -Achse in Richtung der Drehachse $i + 1$,
- die x_i -Achse in Richtung der gemeinsamen Normalen der Achsen z_{i-1} und z_i , wobei der Ursprung $\mathbf{0}_i$ in den Schnittpunkt mit z_i gelegt wird,
- die y_i -Achse für ein rechtswendig orientiertes Koordinatensystem

gesetzt. Das Basis-Koordinatensystem Σ_0 ist bis auf die z_0 -Achse frei wählbar und wird wie in Abbildung 2.1 angenommen. Bei parallelen z -Achsen kann der Ursprung beliebig gewählt werden. Schneiden sich zwei aufeinanderfolgende z -Achsen, wird die x_i -Achse normal zur aufgespannten Ebene von z_{i-1} und z_i angenommen. Die Transformation zwischen den Koordinatensystemen Σ_{i-1} und Σ_i lässt sich über vier DH-Parameter $(q_i, \alpha_i, d_i, a_i)$ beschreiben.

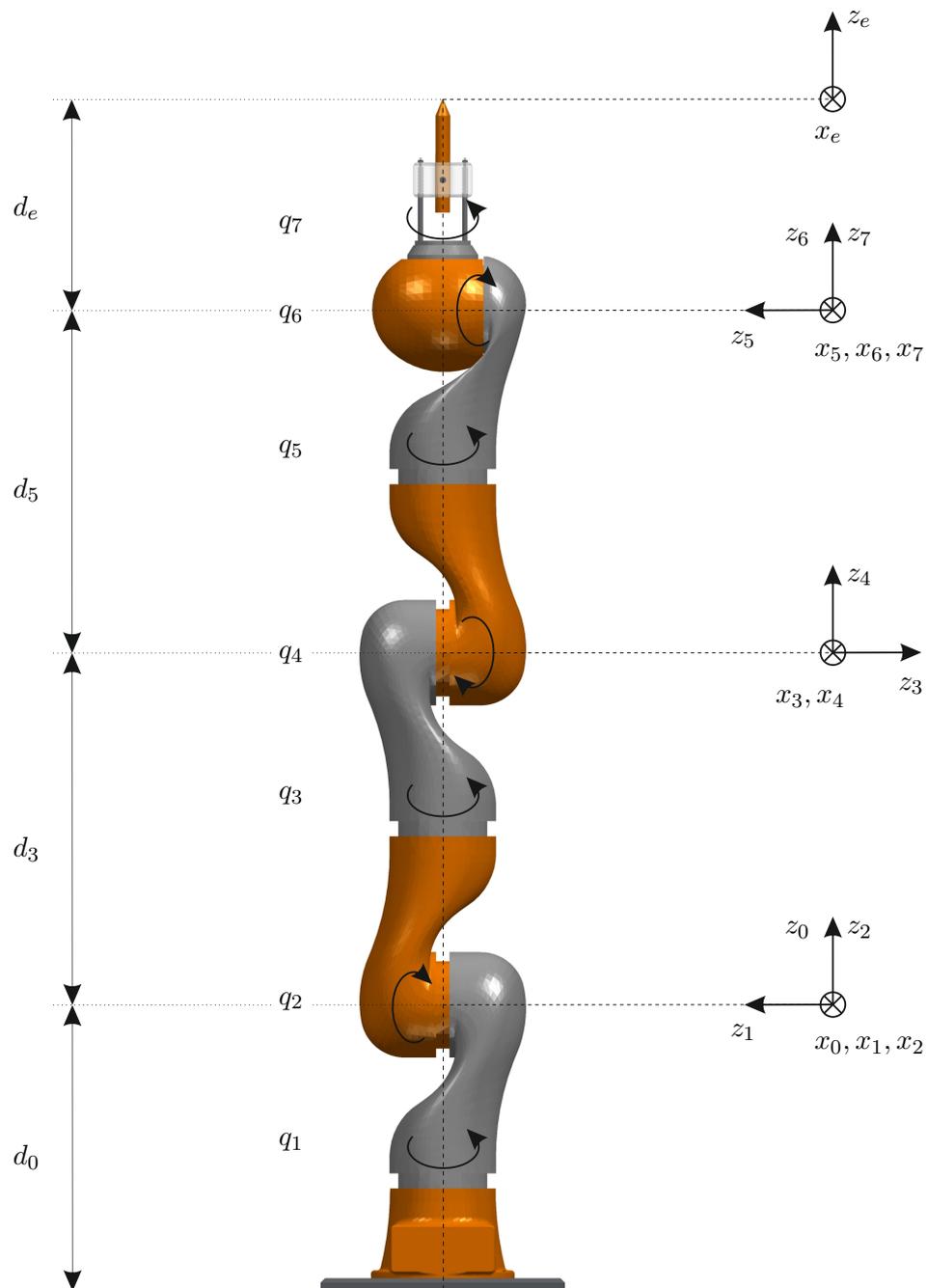


Abbildung 2.1: Koordinatensysteme des Roboters unter Verwendung der CAD-Modelle der Robotergelenke aus [1].

- q_i beschreibt den Winkel zwischen x_{i-1} und x_i um die z_{i-1} -Achse des Drehgelenks
- d_i beschreibt die Verschiebung entlang der z_{i-1} -Achse
- a_i beschreibt die Verschiebung entlang der x_i -Achse
- α_i beschreibt die Drehung zwischen z_{i-1} und z_i um die x_i -Achse

Dabei werden Winkel gegen den Uhrzeigersinn positiv gezählt. Die DH-Parameter für den Roboter sind in Tabelle 2.1 dargestellt, siehe auch [19]. Die Bauteilzeichnungen für die federgelagerte Halterung des Endeffektors sind in Abschnitt A.3 angeführt. Die Halterung ist entworfen, um sie auf den Flansch des Roboters anzubringen, siehe [20].

Σ_i	q_i	d_i	a_i in m	α_i in °
1	q_1	0	0	90
2	q_2	0	0	-90
3	q_3	d_3	0	-90
4	q_4	0	0	90
5	q_5	d_5	0	90
6	q_6	0	0	-90
7	q_7	0	0	0
e	0	d_e	0	0

Tabelle 2.1: DH-Parameter für den Roboter, Werte siehe Tabelle A.1

Stellt man diese Transformationen vom Koordinatensystem Σ_{i-1} zu Σ_i als homogene Transformationsmatrix

$$\mathbf{T}_{i-1}^i(q_i) = \begin{bmatrix} \mathbf{R}_{i-1}^i(q_i) & \mathbf{d}_{i-1}^i(q_i) \\ \mathbf{0} & 1 \end{bmatrix} \quad (2.1)$$

dar, ergibt sich

$$\mathbf{T}_{i-1}^i(q_i) = \begin{bmatrix} c_{q_i} & -s_{q_i} & 0 & 0 \\ s_{q_i} & c_{q_i} & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_{q_i} & -s_{q_i}c_{\alpha_i} & s_{q_i}s_{\alpha_i} & a_i c_{q_i} \\ s_{q_i} & c_{q_i}c_{\alpha_i} & -c_{q_i}s_{\alpha_i} & a_i s_{q_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.2)$$

siehe [18]. Dabei stellt $\mathbf{R}_{i-1}^i(q_i)$ die Rotationsmatrix und $\mathbf{d}_{i-1}^i(q_i)$ die Translation dar. Hier wurde c_{q_i} (s_{q_i}) als verkürzte Schreibweise für $\cos(q_i)$ ($\sin(q_i)$) verwendet.

Aus (2.2) mit den Werten von Tabelle 2.1 erhält man die Transformation vom Basis-Koordinatensystem Σ_0 zu dem Koordinatensystem Σ_i über

$$\mathbf{T}_0^i(\mathbf{q}) = \mathbf{T}_0^1(q_1)\mathbf{T}_1^2(q_2)\dots\mathbf{T}_{i-1}^i(q_i) \quad (2.3)$$

und die direkte Kinematik

$$\mathbf{T}_0^e(\mathbf{q}, d_e) = \mathbf{T}_0^1(q_1)\mathbf{T}_1^2(q_2)\mathbf{T}_2^3(q_3)\mathbf{T}_3^4(q_4)\mathbf{T}_4^5(q_5)\mathbf{T}_5^6(q_6)\mathbf{T}_6^7(q_7)\mathbf{T}_7^e(d_e). \quad (2.4)$$

2.1.2 Jacobi-Matrizen

In diesem Abschnitt werden die geometrischen Jacobi-Matrizen hergeleitet, siehe [18]. Die geometrischen Jacobi-Matrizen der Winkelgeschwindigkeiten $(\mathbf{J}_\omega)_0^i$ stellen den Zusammenhang zwischen den Gelenksgeschwindigkeiten $\dot{\mathbf{q}}$ zu den Drehwinkelgeschwindigkeiten $\boldsymbol{\omega}_0^i$ über

$$\boldsymbol{\omega}_0^i = (\mathbf{J}_\omega)_0^i(\mathbf{q})\dot{\mathbf{q}} \quad (2.5)$$

dar. Dabei bezeichnet $\dot{\mathbf{q}}$ die totale Zeitableitung der Gelenkwinkel $\frac{d\mathbf{q}}{dt}$. Die Indizes von $\boldsymbol{\omega}_0^i$ geben an, dass die Größe vom Koordinatensystem Σ_i im Basis-Koordinatensystem Σ_0 dargestellt wird. Führt man die Matrizen

$$\mathbf{S}(\boldsymbol{\omega}_0^i) = \dot{\mathbf{R}}_0^i(\mathbf{q}) \left(\mathbf{R}_0^i(\mathbf{q}) \right)^T = \sum_{j=1}^7 \left(\frac{\partial}{\partial q_j} \mathbf{R}_0^i(\mathbf{q}) \right) \left(\mathbf{R}_0^i(\mathbf{q}) \right)^T \dot{q}_j \quad (2.6)$$

mit Verwendung von (2.1) und (2.3) ein, so zeigt sich, dass diese die schiefsymmetrische Form

$$\mathbf{S}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (2.7)$$

aufweisen. Durch Vergleich von (2.5), (2.6) und (2.7) ergibt sich eine Vorschrift, um die Jacobi-Matrizen der Drehwinkelgeschwindigkeiten $(\mathbf{J}_\omega)_0^i$ zu berechnen.

Die geometrischen Jacobi-Matrizen der translatorischen Geschwindigkeiten $(\mathbf{J}_v)_0^i$ berechnen sich über

$$\mathbf{v}_0^i = \dot{\mathbf{p}}_0^i(\mathbf{q}) = \sum_{j=1}^7 \left(\frac{\partial}{\partial q_j} \mathbf{p}_0^i(\mathbf{q}) \right) \dot{q}_j = (\mathbf{J}_v)_0^i(\mathbf{q})\dot{\mathbf{q}}. \quad (2.8)$$

Dabei bezeichnet $\mathbf{p}_0^i = (\mathbf{p}_s)_0^i$ die Position des Massenschwerpunkts vom Starrkörper i im Basis-Koordinatensystems Σ_0 , wobei für die Geschwindigkeit $\mathbf{v}_0^i = \dot{\mathbf{p}}_0^i$ gilt. Die Position \mathbf{p}_0^i wird mit (2.3) über

$$\begin{bmatrix} \mathbf{p}_0^i(\mathbf{q}) \\ 1 \end{bmatrix} = \mathbf{T}_0^i(\mathbf{q}) \begin{bmatrix} \mathbf{p}_i^i \\ 1 \end{bmatrix} \quad (2.9)$$

berechnet, wobei für $\mathbf{p}_i^i = (\mathbf{p}_s)_i^i$ der Vektor zum Schwerpunkt des Starrkörpers i im Koordinatensystem Σ_i eingesetzt wird. Bei der Berechnung der geometrischen Jacobi-Matrix der translatorischen Geschwindigkeit am Endeffektor $(\mathbf{J}_v)_0^e$ wird die Position des Ursprungs des Koordinatensystems Σ_e anstatt des Massenschwerpunkts verwendet, daher gilt $\mathbf{p}_e^e = \mathbf{0}$ in (2.8) und (2.9).

2.2 Dynamik

In diesem Kapitel werden die Bewegungsgleichungen des Roboters hergeleitet. Dafür wird in Abschnitt 2.2.1 das dynamische Modell bei starren Gelenken beschrieben. In Abschnitt 2.2.2 wird das Robotermodell um flexible Gelenke erweitert.

2.2.1 Dynamik mit starren Gelenken

In diesem Abschnitt wird das dynamische Modell bei starren Gelenken über die Euler-Lagrange Gleichungen hergeleitet, siehe [21]. Diese lauten

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i} \right) - \frac{\partial \mathcal{L}}{\partial q_i} = \tau_i, \quad i = 1, \dots, n, \quad (2.10)$$

wobei für die Lagrange-Funktion $\mathcal{L} = \mathcal{T} - \mathcal{V}$ gilt. Dabei bezeichnet \mathcal{T} die kinetische Energie und \mathcal{V} die potenzielle Energie. Bei q_i handelt es sich im Allgemeinen um generalisierte Koordinaten, die in diesem Fall die Gelenkwinkel beschreiben. Die externen generalisierten Kräfte werden mit τ_i bezeichnet.

Die Berechnung der kinetischen Energie \mathcal{T} erfolgt über

$$\mathcal{T} = \frac{1}{2} \sum_{i=1}^7 \left(m_i (\mathbf{v}_0^i)^T \mathbf{v}_0^i + (\boldsymbol{\omega}_0^i)^T \mathbf{I}_0^i \boldsymbol{\omega}_0^i \right). \quad (2.11)$$

Dabei stellt m_i die Masse des Starrkörpers i und \mathbf{I}_0^i den entsprechenden Trägheitstensor am Schwerpunkt im Basis-Koordinatensystem Σ_0 dar. Dieser Trägheitstensor wird unter Verwendung von (2.1) und (2.3) in der Form

$$\mathbf{I}_0^i = (\mathbf{R}_i^0)^T \mathbf{I}_i^i \mathbf{R}_i^0 \quad (2.12)$$

berechnet, wobei \mathbf{I}_i^i der Trägheitstensor am Schwerpunkt ausgedrückt im körperfesten Koordinatensystem Σ_i ist. Wird (2.5) und (2.8) in (2.11) eingesetzt, erhält man

$$\mathcal{T} = \frac{1}{2} \dot{\mathbf{q}}^T \underbrace{\sum_{i=1}^7 \left(m_i \left((\mathbf{J}_v)_0^i \right)^T (\mathbf{J}_v)_0^i + \left((\mathbf{J}_\omega)_0^i \right)^T \mathbf{I}_0^i (\mathbf{J}_\omega)_0^i \right)}_{\mathbf{M}(\mathbf{q})} \dot{\mathbf{q}}, \quad (2.13)$$

wobei $\mathbf{M}(\mathbf{q})$ die symmetrische und positiv definite Massenmatrix darstellt. Die potenzielle Energie \mathcal{V} wird mit g als Gravitationsbeschleunigung über

$$\mathcal{V}(\mathbf{q}) = \sum_{i=1}^7 \begin{bmatrix} 0 & 0 & g \end{bmatrix} \mathbf{p}_0^i(\mathbf{q}) m_i \quad (2.14)$$

berechnet.

Für den Vektor der externen generalisierten Kräfte $\boldsymbol{\tau} = [\tau_1 \ \cdots \ \tau_7]^\top$ gilt mit Verwendung von (2.1) und (2.3)

$$\boldsymbol{\tau} = \sum_{i=1}^7 \left((\mathbf{J}_\omega)_0^i \right)^\top \mathbf{R}_0^{i-1} \left(\begin{bmatrix} 0 \\ 0 \\ \mathcal{M}_i \end{bmatrix} + \mathbf{R}_{i-1}^i \begin{bmatrix} 0 \\ 0 \\ -\mathcal{M}_{i+1} \end{bmatrix} \right) = \begin{bmatrix} \mathcal{M}_1 \\ \mathcal{M}_2 \\ \mathcal{M}_3 \\ \mathcal{M}_4 \\ \mathcal{M}_5 \\ \mathcal{M}_6 \\ \mathcal{M}_7 \end{bmatrix}, \quad (2.15)$$

wobei \mathcal{M}_i das Drehmoment am Gelenk i darstellt. Die Gleichung (2.15) ergibt sich, da auf einen Starrkörper des Roboters Drehmomente an beiden Enden wirken und diese im selben Koordinatensystem ausgedrückt werden müssen. Anschließend werden die Momente über eine Rotation in das Basis-Koordinatensystem Σ_0 transformiert.

Wird die Lagrange-Funktion $\mathcal{L} = \mathcal{T} - \mathcal{V}$ mit (2.13) und (2.14) berechnet und mit τ_i aus (2.15) in die Euler-Lagrange Gleichung (2.10) eingesetzt, erhält man die Bewegungsgleichungen in Komponentenschreibweise

$$\sum_{j=1}^7 M_{ij}(\mathbf{q}) \ddot{q}_j + \sum_{j=1}^7 \sum_{k=1}^7 C_{ijk}(\mathbf{q}) \dot{q}_k \dot{q}_j + g_i(\mathbf{q}) = \tau_i, \quad i = 1, \dots, 7. \quad (2.16)$$

Dabei stellen $M_{ij}(\mathbf{q})$ die Komponenten der Massenmatrix $\mathbf{M}(\mathbf{q})$ und $C_{ijk}(\mathbf{q})$ die Christoffel-Symbole erster Art

$$C_{ijk}(\mathbf{q}) = \frac{1}{2} \left(\frac{\partial M_{ij}(\mathbf{q})}{\partial q_k} + \frac{\partial M_{ik}(\mathbf{q})}{\partial q_j} - \frac{\partial M_{jk}(\mathbf{q})}{\partial q_i} \right), \quad (2.17)$$

dar. Die Terme $g_i(\mathbf{q})$ berechnen sich zu

$$g_i(\mathbf{q}) = \frac{\partial \mathcal{V}(\mathbf{q})}{\partial q_i}. \quad (2.18)$$

Fasst man die einzelnen Komponenten aus (2.16) zusammen, wobei

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})[i, j] = \sum_{k=1}^7 C_{ijk}(\mathbf{q}) \dot{q}_k \quad (2.19)$$

gilt, ergeben sich die Bewegungsgleichungen in Matrixschreibweise zu

$$\mathbf{M}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}. \quad (2.20)$$

Sämtliche Parameter des Roboters sind in [19, 22] zu finden, wo eine Identifikation der Roboterparameter über eine nichtlineare Optimierung beschrieben wird.

2.2.2 Dynamik mit flexiblen Gelenken

In diesem Abschnitt wird das dynamische Modell aus Abschnitt 2.2.1 um die Gelenkelastizitäten erweitert. Danach wird dieses erweiterte dynamische Modell mit Hilfe der singulären Störtheorie vereinfacht. Die Ausführungen in diesem Abschnitt sind aus [23] entnommen.

In dieser Arbeit wird ein reduziertes Modell des Roboters mit flexiblen Gelenken verwendet, siehe [24]. Die Herleitung des vollständigen Modells ist in [23] gezeigt. Um das reduzierte Modell zu verwenden, wird die Annahme getroffen, dass der rotatorische Anteil der kinetischen Energie des Motors nur durch die relative Bewegung zum vorherigen Gelenk bestimmt wird und daher die dynamische Kopplung der mechanischen Teilsysteme vernachlässigt werden kann. Dies wird durch die spezielle Bauform des Getriebes und sein hohes Übersetzungsverhältnis gerechtfertigt. Dadurch kommt es zu einer großen Motordrehzahl, wodurch die kinetische Energie des Motors kaum durch die Bewegung des Gelenks beeinflusst wird. Der Aufbau des Gelenks ist schematisch in Abbildung 2.2 dargestellt, wobei die Gelenkelastizität als lineare Feder angenommen wird.

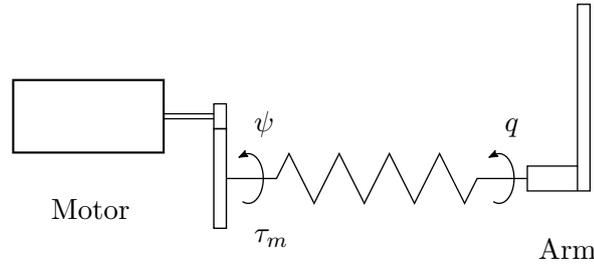


Abbildung 2.2: Schematischer Aufbau des Robotergelenks, vergleiche [23].

Die Modellgleichungen des erweiterten Modells ergeben sich mit (2.20) zu

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{K}(\boldsymbol{\psi} - \mathbf{q}) + \boldsymbol{\tau}_{ext} , \quad (2.21a)$$

$$\mathbf{B}\ddot{\boldsymbol{\psi}} + \mathbf{K}(\boldsymbol{\psi} - \mathbf{q}) = \boldsymbol{\tau}_m . \quad (2.21b)$$

Dabei beschreibt \mathbf{B} die Trägheitsmatrix, $\boldsymbol{\psi}$ die Drehwinkel und $\boldsymbol{\tau}_m$ die Drehmomente der Motoren an der Abtriebsseite der Getriebe. Die diagonale Steifigkeitsmatrix der Gelenke wird mit \mathbf{K} bezeichnet. Die Variable $\boldsymbol{\tau}_{ext}$ stellt mögliche externe Drehmomente dar.

Das erweiterte Modell (2.21) wird mittels der singulären Störtheorie reduziert. Dazu wird

$$\boldsymbol{\tau}_c = \mathbf{K}(\boldsymbol{\psi} - \mathbf{q}), \quad \ddot{\boldsymbol{\psi}} = \mathbf{K}^{-1}\ddot{\boldsymbol{\tau}}_c + \ddot{\mathbf{q}} \quad (2.22)$$

eingeführt und in (2.21) eingesetzt. Daraus folgt

$$\ddot{\mathbf{q}} = (\mathbf{M}(\mathbf{q}))^{-1} (\boldsymbol{\tau}_c + \boldsymbol{\tau}_{ext} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q})) , \quad (2.23a)$$

$$\mathbf{B}\mathbf{K}^{-1}\ddot{\boldsymbol{\tau}}_c = \boldsymbol{\tau}_m - \boldsymbol{\tau}_c - \mathbf{B}\ddot{\mathbf{q}} . \quad (2.23b)$$

Wenn die Einträge der Steifigkeitsmatrix \mathbf{K} sehr groß sind, lassen sich die Drehmomente durch

$$\bar{\boldsymbol{\tau}}_c = \bar{\boldsymbol{\tau}}_m - \mathbf{B} (\mathbf{M}(\bar{\mathbf{q}}))^{-1} (\bar{\boldsymbol{\tau}}_c + \bar{\boldsymbol{\tau}}_{ext} - \mathbf{C}(\bar{\mathbf{q}}, \dot{\bar{\mathbf{q}}})\dot{\bar{\mathbf{q}}} - \mathbf{g}(\bar{\mathbf{q}})) \quad (2.24)$$

anschreiben. Dabei stellen $\bar{\mathbf{q}}$ und $\bar{\boldsymbol{\tau}}$ die quasi-stationären Größen dar, die bei der Annahme $\mathbf{BK}^{-1}\bar{\boldsymbol{\tau}}_c = \mathbf{0}$ die Gleichung (2.23) und (2.24) erfüllen.

Wird ein Drehmomentregler der Form

$$\bar{\boldsymbol{\tau}}_m = \bar{\boldsymbol{\tau}}_d - \mathbf{K}_\tau(\bar{\boldsymbol{\tau}}_c - \bar{\boldsymbol{\tau}}_d) \quad (2.25)$$

verwendet, in (2.24) eingesetzt und auf $\bar{\boldsymbol{\tau}}_c$ umgeformt, so erhält man

$$\bar{\boldsymbol{\tau}}_c = \left(\mathbf{E} + \mathbf{K}_\tau + \mathbf{B}(\mathbf{M}(\bar{\mathbf{q}}))^{-1} \right)^{-1} \left((\mathbf{E} + \mathbf{K}_\tau)\bar{\boldsymbol{\tau}}_d + \mathbf{B}(\mathbf{M}(\bar{\mathbf{q}}))^{-1} (\mathbf{C}(\bar{\mathbf{q}}, \dot{\bar{\mathbf{q}}})\dot{\bar{\mathbf{q}}} + \mathbf{g}(\bar{\mathbf{q}}) - \bar{\boldsymbol{\tau}}_{ext}) \right). \quad (2.26)$$

Dabei stellt $\boldsymbol{\tau}_d$ die gewünschten Drehmomente, \mathbf{K}_τ die Proportionalverstärkung und \mathbf{E} die Einheitsmatrix dar. Durch Einsetzen von (2.26) in (2.23a) errechnet sich das quasi-stationäre Modell zu

$$\left(\mathbf{M}(\ddot{\bar{\mathbf{q}}}) + (\mathbf{E} + \mathbf{K}_\tau)^{-1}\mathbf{B} \right) \ddot{\bar{\mathbf{q}}} + \mathbf{C}(\bar{\mathbf{q}}, \dot{\bar{\mathbf{q}}})\dot{\bar{\mathbf{q}}} + \mathbf{g}(\bar{\mathbf{q}}) = \bar{\boldsymbol{\tau}}_d + \bar{\boldsymbol{\tau}}_{ext} \quad (2.27)$$

Diese Gleichung zeigt, dass durch Hinzufügen der flexiblen Gelenkdynamik und Anwenden der singulären Störtheorie lediglich die Massenmatrix verändert wird.

2.3 Modellierung der Gleitreibung

Der Roboter weist in den Gelenken starke Reibung auf, weswegen in der KUKA Steuerung eine Reibkompensation durchgeführt wird. Da noch Reibung vorhanden ist, wird in diesem Abschnitt die Identifikation der verbleibenden Gleitreibung erläutert und es werden Messergebnisse gezeigt.

Um die mittlere Gleitreibung $\tau_{fri,v} = f(\dot{q}_i)$ zu erhalten, muss sich das Gelenk mit einer konstanten Gelenksgeschwindigkeit bewegen. Dabei wird das dafür benötigte Drehmoment des Motors gemessen. Durch die Kombination von Messungen mit unterschiedlichen Geschwindigkeiten wird eine Reibkurve vermessen.

Es wird ein diskreter PI-Geschwindigkeitsregler

$$\tau_{d,k} = K_{Pv}e_{v,k} + K_{Iv}e_{Iv,k} \quad (2.28)$$

für jedes Gelenk entworfen, wobei die Fehler e über

$$e_{v,k} = \dot{q}_{d,k} - \dot{q}_k \quad (2.29a)$$

$$e_{Iv,k+1} = e_{Iv,k} + T_a e_{v,k}, \quad e_{Iv,0} = 0 \quad (2.29b)$$

gebildet werden. Der Index k gibt dabei den Abtastschritt ($e_{v,k} = e_v(kT_a)$), T_a die Abtastzeit und $\dot{q}_{d,k}$ die gewünschte Gelenksgeschwindigkeit zum Abtastschritt k an. Die Reglerparameter K_{Pv} und K_{Iv} sind positive Konstanten und sind in Tabelle A.1 angegeben. Für das zu vermessende Gelenk i wird ein gewünschter Gelenksgeschwindigkeitsverlauf $\dot{q}_{d,k}$ vorgegeben. Um den Einfluss der nicht gemessenen Gelenke $j \neq i, j \in \{1, \dots, 7\}$ zu unterdrücken, wird für diese $\dot{q}_{d,k} = 0$ gewählt.

In Abbildung 2.3 ist ein Ausschnitt der Messung für das erste Gelenk dargestellt. In der oberen Abbildung ist die gewünschte Gelenksgeschwindigkeit $\dot{q}_{1,d}$ und die tatsächliche

Gelenksgeschwindigkeit \dot{q}_1 gezeigt. Der dazugehörige Drehmomentverlauf $\tau_{fri,1}$ ist in der unteren Abbildung gezeigt. Beim Vorzeichenwechsel der Gelenksgeschwindigkeit ist ein starker Drehmomentsprung erkennbar, der den Roboter abbremst und auf die neue Geschwindigkeit beschleunigt. Dieser Einschwingvorgang wird nicht für die Ermittlung der Reibkurve verwendet. Für die stationären Werte von \dot{q}_1 und $\tau_{fri,1}$ wird der Mittelwert berechnet. In weiterer Folge wird die Gelenksgeschwindigkeit erhöht und das zugehörige Drehmoment gemessen. Die so aufgenommenen Messpunkte sind in Abbildung 2.4 dargestellt.

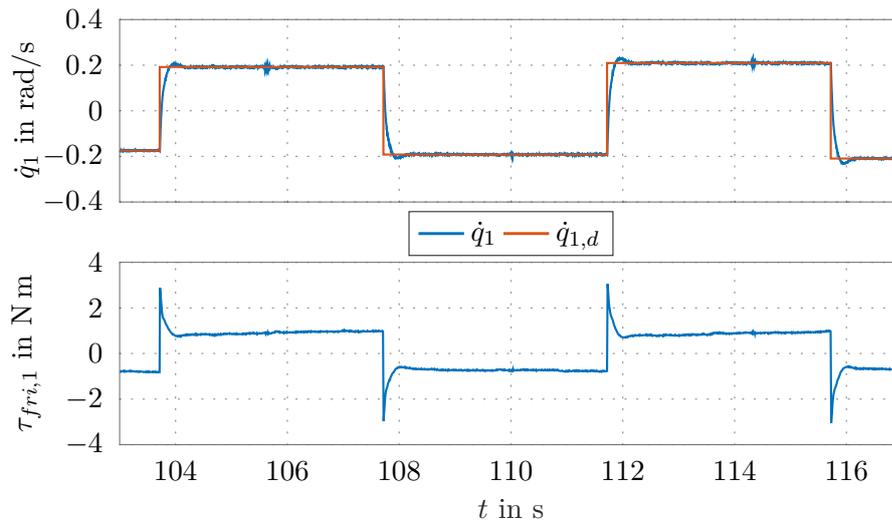


Abbildung 2.3: Ausschnitt der Messung der Gleitreibung für Gelenk 1.

In Abbildung 2.4 sind zwei Reibkurven desselben Gelenks nach unterschiedlicher Betriebsdauer dargestellt. Bei den Messdaten für eine kurze Betriebsdauer erkennt man einen gleichen Verlauf der Reibmomente bei positiver und negativer Gelenksgeschwindigkeit. Es fällt jedoch auf, dass die Kennlinie einen Offset aufweist. Diese Verschiebung um den Nullpunkt ändert sich nach einer Neukalibrierung der Drehmomentsensoren und nachdem der Roboter durch einen längeren Betrieb eine höhere Temperatur aufweist, wie in Abbildung 2.4 zu sehen ist. Da die Reibung mit der Temperatur und Betriebsdauer des Roboters schwankt, ist die Messung nicht reproduzierbar. Daher wird die Reibung im Modell nicht berücksichtigt.

Die Reibkurven der übrigen Gelenke sind in Abbildung 2.5 gezeigt. Dabei ist bei den Reibkurven der Gelenke 2, 4 und 6 eine Asymmetrie bei positiver und negativer Gelenksgeschwindigkeit zu sehen. Die Rotationsachsen dieser Gelenke liegen in Kerzenstellung horizontal, siehe Abbildung 2.1. Somit lässt sich die Asymmetrie auf den Einfluss der Gravitationskraft zurückführen, die nach der Gravitationskompensation der KUKA Steuerung vorhanden ist. Daher besteht eine Ungenauigkeit in der Kompensation der Gravitation. In den Reibkurven der Gelenke 3 und 5 ist der Stribeck-Effekt zu erkennen. Außerdem entsteht ein Anstieg des Reibmoments zu größeren Gelenksgeschwindigkeiten durch den viskosen Reibanteil, siehe [25].

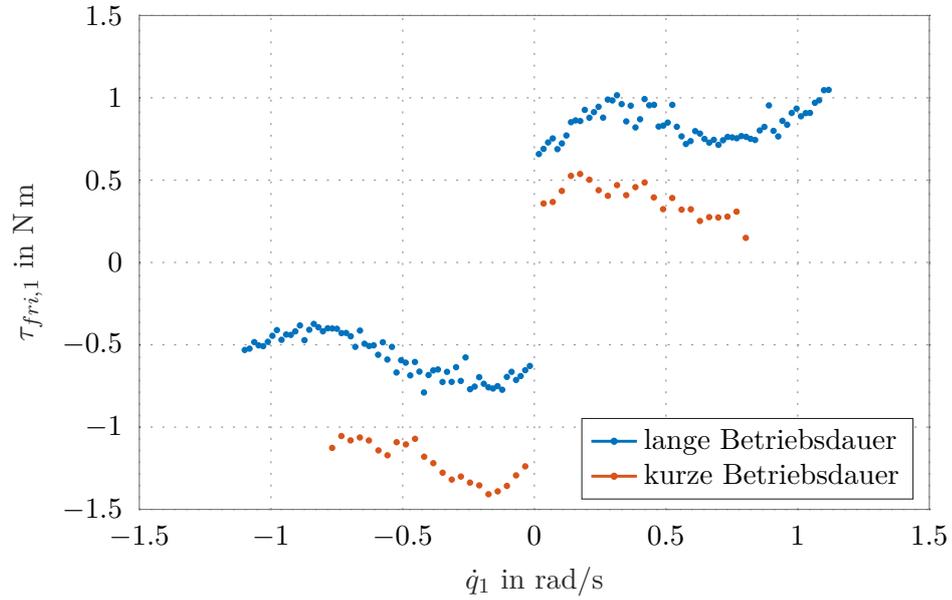


Abbildung 2.4: Reibkurve für Gelenk 1.

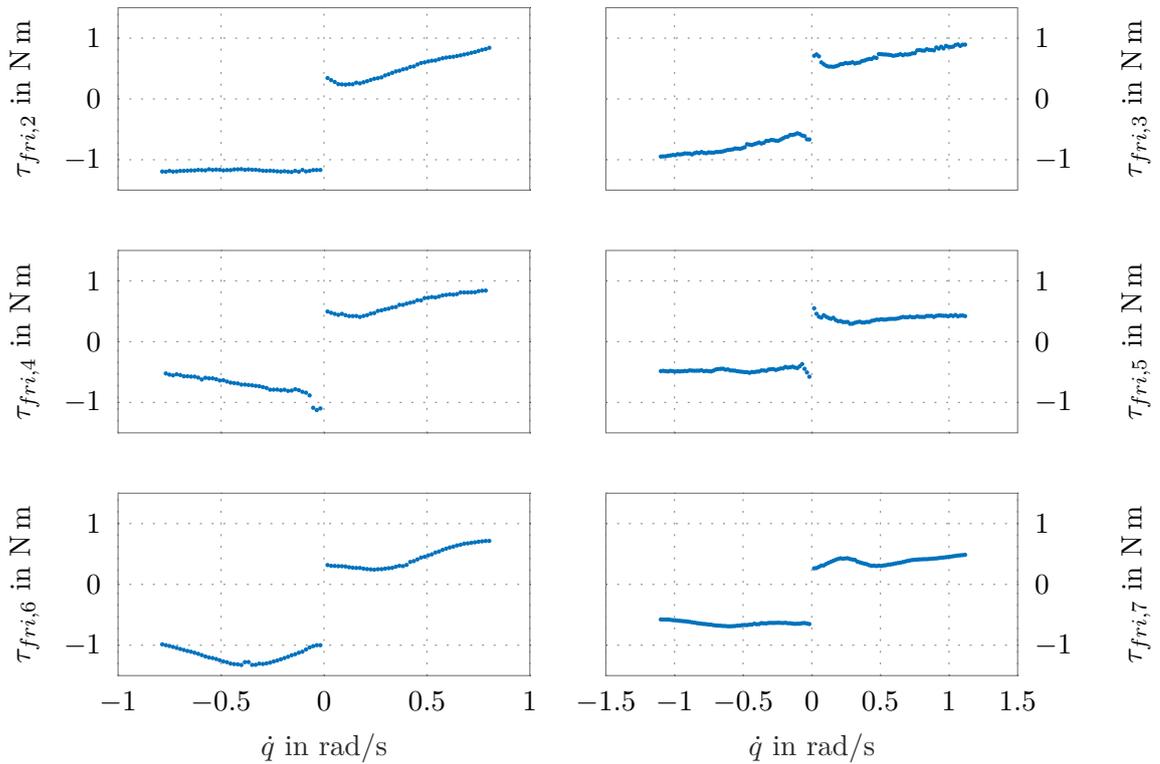


Abbildung 2.5: Reibkurven der Gelenke 2,...,7.

2.4 Mathematisches Modell für den Reglerentwurf

In diesem Abschnitt wird das im Weiteren verwendete Modell näher erläutert. Die direkte Kinematik ist in (2.4) und die Jacobi-Matrizen sind in (2.5) und (2.8) dargestellt. Die quasi-stationären Größen $\bar{\mathbf{q}}$ und $\bar{\boldsymbol{\tau}}$ in (2.27) werden der Einfachheit halber ohne Überstrich dargestellt.

- Der KUKA Steuerung werden über das Fast Research Interface (FRI) die gewünschten Drehmomente $\boldsymbol{\tau}_d$ mit einer Zykluszeit von 1 ms übergeben, siehe [26]. Eine Modellierung des Roboters mit flexiblen Gelenken ist in Abschnitt 2.2.2 gezeigt, wobei die Proportionalverstärkung der Drehmomente \mathbf{K}_τ sowie die Werte der Trägheitsmatrix \mathbf{B} nicht zur Verfügung stehen. Daher wird zur Vereinfachung die aus unbekanntem Matrizen bestehende Matrix $\tilde{\mathbf{B}} = (\mathbf{E} + \mathbf{K}_\tau)^{-1}\mathbf{B}$ eingeführt. Die Matrix wird zur Einheitsmatrix $\tilde{\mathbf{B}} = \mathbf{E}$ gewählt, um gute Ergebnisse zu erzielen.
- Da der Roboter an der Decke montiert ist, ist die Gravitationsbeschleunigung g in (2.14) etwa -9.81 m/s^2 . Es besteht jedoch die Möglichkeit, in der Robotersteuerung die Gravitation zu kompensieren. Daher wird der Term $\mathbf{g}(\mathbf{q})$ im Modell nicht berücksichtigt.
- Messungen zeigen außerdem, dass der Term $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ bei wenig dynamischen Anwendungen vernachlässigbar klein ist. Dadurch wird dieser Term im Modell ebenfalls vernachlässigt.
- Die Messungen der Gleitreibung sind in Abschnitt 2.3 gezeigt, wobei die Reibmomente $\boldsymbol{\tau}_{fri}$ als negative externe Drehmomente $\boldsymbol{\tau}_{ext}$ in die Gleichung (2.27) eingehen. Auf eine Berücksichtigung in der Regelung wird jedoch verzichtet, da die Werte starken Schwankungen unterworfen und nicht reproduzierbar sind.

Somit lässt sich das vereinfachte dynamische Modell für die Regelung durch

$$(\mathbf{M}(\mathbf{q}) + \mathbf{E}) \ddot{\mathbf{q}} = \tilde{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} = \boldsymbol{\tau}_d \quad (2.30)$$

beschreiben.

In diesem Kapitel wurde das mathematische Modell des Roboters hergeleitet. Um die Kinematik zu beschreiben, wurde jedem Starrkörper sowie dem Endeffektor ein Koordinatensystem zugeordnet. Außerdem wurden die Jacobi-Matrizen berechnet. Die Kinematik konnte genau beschrieben werden, da die dafür notwendigen Parameter zur Verfügung stehen. Die Bewegungsgleichungen beschreiben die Dynamik des Roboters, wobei die Gleichungen für starre Gelenke um die Eigenschaften von flexiblen Gelenken erweitert wurden. Durch die Verwendung des Modells mit flexiblen Gelenken sind zusätzliche Informationen über die Motoren notwendig. Die Identifikation der Gleitreibung sowie die dazugehörigen Messergebnisse wurden ebenfalls gezeigt. Da die Reibung stark variiert und von der Betriebsdauer (Erwärmung) und den Umgebungs- und Lastbedingungen abhängig ist, wurde diese im mathematischen Modell nicht berücksichtigt. Zuletzt wurden die Erkenntnisse dieses Kapitels zusammengefasst, wodurch ein stark vereinfachtes dynamisches Modell des Roboters für den Reglerentwurf zur Verfügung steht.

3 Regelungskonzept

In diesem Kapitel wird das verwendete Regelungskonzept vorgestellt. Zuerst wird in Abschnitt 3.1 das Regelgesetz der inversen Dynamik erläutert. Für dieses Regelgesetz wird in Abschnitt 3.1.1 und Abschnitt 3.1.2 die Stabilität untersucht. In Abschnitt 3.2 wird auf die Regelung der Nulldynamik des Roboters näher eingegangen.

3.1 Regelgesetz der inversen Dynamik

In diesem Abschnitt wird das Regelgesetz der inversen Dynamik hergeleitet, siehe [18]. Das dynamische Modell aus (2.30) lautet

$$\tilde{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{q}} = \boldsymbol{\tau}_d, \quad (3.1)$$

mit den Drehmomenten $\boldsymbol{\tau}_d$ als Stellgröße. Da die Regelung in Bezug auf das Basis-Koordinatensystem Σ_0 entworfen wird, benötigt man die Gleichung

$$\begin{bmatrix} \dot{\mathbf{p}}_e \\ \boldsymbol{\omega}_e \end{bmatrix} = \begin{bmatrix} \mathbf{J}_v(\mathbf{q}) \\ \mathbf{J}_\omega(\mathbf{q}) \end{bmatrix} \dot{\mathbf{q}}, \quad (3.2)$$

mit den Jacobi-Matrizen aus (2.5) und (2.8). Damit wird die zeitliche Ableitung der Gelenkskoordinaten $\dot{\mathbf{q}}$ in eine zeitliche Ableitung der Koordinaten des Endeffektors $[\dot{\mathbf{p}}_e^T \ \boldsymbol{\omega}_e^T]^T$ umgerechnet. Auf die Darstellung des Basis-Koordinatensystems Σ_0 im Index wird im Weiteren verzichtet. Außerdem werden in diesem Kapitel ausschließlich die Jacobi-Matrizen zum Endeffektor-Koordinatensystem Σ_e verwendet. Wird (3.2) nach der Zeit abgeleitet, kann $\ddot{\mathbf{q}}$ als

$$\ddot{\mathbf{q}} = \begin{bmatrix} (\mathbf{J}_v(\mathbf{q}))^\dagger (\ddot{\mathbf{p}}_e - \dot{\mathbf{J}}_v(\mathbf{q})\dot{\mathbf{q}}) \\ (\mathbf{J}_\omega(\mathbf{q}))^\dagger (\dot{\boldsymbol{\omega}}_e - \dot{\mathbf{J}}_\omega(\mathbf{q})\dot{\mathbf{q}}) \end{bmatrix} \quad (3.3)$$

angeschrieben werden. Dabei gibt \dagger die Pseudoinverse einer Matrix an, wobei

$$(\mathbf{J}(\mathbf{q}))^\dagger = \left((\mathbf{J}(\mathbf{q}))^T \mathbf{J}(\mathbf{q}) \right)^{-1} (\mathbf{J}(\mathbf{q}))^T \quad (3.4)$$

gilt. Setzt man (3.3) in (3.1) ein, erhält man

$$\tilde{\mathbf{M}}(\mathbf{q}) \begin{bmatrix} (\mathbf{J}_v(\mathbf{q}))^\dagger (\ddot{\mathbf{p}}_e - \dot{\mathbf{J}}_v(\mathbf{q})\dot{\mathbf{q}}) \\ (\mathbf{J}_\omega(\mathbf{q}))^\dagger (\dot{\boldsymbol{\omega}}_e - \dot{\mathbf{J}}_\omega(\mathbf{q})\dot{\mathbf{q}}) \end{bmatrix} = \boldsymbol{\tau}_d. \quad (3.5)$$

Wird die Stellgröße τ_d zu

$$\tau_d = \tilde{\mathbf{M}}(\mathbf{q}) \begin{bmatrix} (\mathbf{J}_v(\mathbf{q}))^\dagger \left(\boldsymbol{\nu}_p - \dot{\mathbf{J}}_v(\mathbf{q})\dot{\mathbf{q}} \right) \\ (\mathbf{J}_\omega(\mathbf{q}))^\dagger \left(\boldsymbol{\nu}_o - \dot{\mathbf{J}}_\omega(\mathbf{q})\dot{\mathbf{q}} \right) \end{bmatrix} \quad (3.6)$$

gewählt und in (3.5) eingesetzt, kann die neue Stellgröße $\boldsymbol{\nu}$ in der Form

$$\begin{bmatrix} \boldsymbol{\nu}_p \\ \boldsymbol{\nu}_o \end{bmatrix} = \begin{bmatrix} \ddot{\mathbf{p}}_e \\ \dot{\boldsymbol{\omega}}_e \end{bmatrix} = \begin{bmatrix} \ddot{\mathbf{p}}_d + \tilde{\mathbf{K}}_v(\dot{\mathbf{p}}_d - \dot{\mathbf{p}}_e) + \tilde{\mathbf{K}}_p(\mathbf{p}_d - \mathbf{p}_e) + \tilde{\mathbf{K}}_{Ip} \int (\mathbf{p}_d - \mathbf{p}_e) dt \\ \dot{\boldsymbol{\omega}}_d + \mathbf{K}_\omega(\boldsymbol{\omega}_d - \boldsymbol{\omega}_e) + \mathbf{K}_o \mathbf{e}_o \end{bmatrix} \quad (3.7)$$

festgelegt werden, vergleiche [27]. Dabei stellt $\mathbf{p}_d(t)$ den gewünschten Verlauf der kartesischen Koordinaten und $\boldsymbol{\omega}_d(t)$ den Verlauf der gewünschten Drehwinkelgeschwindigkeiten des Endeffektors sowie \mathbf{e}_o den Orientierungsfehler dar. Die Position des Endeffektors \mathbf{p}_e wird durch den Translationsvektor von (2.4) mit (2.1) bestimmt. Die Matrizen $\tilde{\mathbf{K}}_v$, $\tilde{\mathbf{K}}_p$ und $\tilde{\mathbf{K}}_{Ip}$ werden über

$$\tilde{\mathbf{K}}_v = \mathbf{R}_d \mathbf{K}_v \mathbf{R}_d^\top \quad (3.8a)$$

$$\tilde{\mathbf{K}}_p = \mathbf{R}_d \mathbf{K}_p \mathbf{R}_d^\top \quad (3.8b)$$

$$\tilde{\mathbf{K}}_{Ip} = \mathbf{R}_d \mathbf{K}_{Ip} \mathbf{R}_d^\top \quad (3.8c)$$

berechnet. Dabei stellt \mathbf{R}_d die gewünschte Rotation des Endeffektors gegenüber dem Basis-Koordinatensystem Σ_0 des Roboters dar. Mittels der Transformation (3.8) lässt sich die Dynamik des Positionsfehlers mithilfe des Koordinatensystems der gewünschten Orientierung einstellen. Die Matrizen \mathbf{K}_v , \mathbf{K}_p , \mathbf{K}_{Ip} , \mathbf{K}_ω und \mathbf{K}_o werden als Diagonalmatrizen mit positiven Einträgen gewählt und sind in Tabelle A.1 aufgelistet.

3.1.1 Stabilität der Dynamik des Positionsfehlers

Um die Stabilität der Positionsfehlerdynamik zu zeigen, wird $\boldsymbol{\nu}_p$ von (3.7) mit dem Zustand $\boldsymbol{\xi} = [\tilde{\mathbf{p}}^\top \quad \dot{\tilde{\mathbf{p}}}^\top \quad \ddot{\tilde{\mathbf{p}}}^\top]^\top$ und (3.8) als

$$\begin{aligned} \dot{\boldsymbol{\xi}} &= \tilde{\mathbf{A}} \boldsymbol{\xi} = \begin{bmatrix} \mathbf{0} & \mathbf{E} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{E} \\ -\tilde{\mathbf{K}}_{Ip} & -\tilde{\mathbf{K}}_p & -\tilde{\mathbf{K}}_v \end{bmatrix} \boldsymbol{\xi} = \\ &= \mathbf{V} \mathbf{A} \mathbf{V}^{-1} \boldsymbol{\xi} = \begin{bmatrix} \mathbf{R}_d & 0 & 0 \\ 0 & \mathbf{R}_d & 0 \\ 0 & 0 & \mathbf{R}_d \end{bmatrix} \begin{bmatrix} \mathbf{0} & \mathbf{E} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{E} \\ -\mathbf{K}_{Ip} & -\mathbf{K}_p & -\mathbf{K}_v \end{bmatrix} \begin{bmatrix} \mathbf{R}_d^\top & 0 & 0 \\ 0 & \mathbf{R}_d^\top & 0 \\ 0 & 0 & \mathbf{R}_d^\top \end{bmatrix} \boldsymbol{\xi} \end{aligned} \quad (3.9)$$

angeschrieben. Dabei bezeichnet $\tilde{\mathbf{p}} = \mathbf{p}_d - \mathbf{p}_e$ den Positionsfehler. Da die Matrizen $\tilde{\mathbf{A}}$ und \mathbf{A} ähnlich sind und $\mathbf{R}_d^\top = \mathbf{R}_d^{-1}$ gilt, lässt sich das charakteristische Polynom über

$$\det(\lambda \mathbf{E} - \mathbf{A}) = \prod_{i=1}^3 \left(\lambda^3 + \lambda^2 K_{v,i} + \lambda K_{p,i} + K_{Ip,i} \right) \quad (3.10)$$

anschreiben. Dabei gibt der Index i den Eintrag der Diagonalmatrix \mathbf{K} an. Die Koeffizienten \mathbf{K} sind so gewählt, dass das charakteristische Polynom (3.10) Hurwitz ist. Somit weisen alle Eigenwerte von \mathbf{A} negativen Realteil auf, womit die globale asymptotische Stabilität der Ruhelage $\boldsymbol{\xi} = \mathbf{0}$ gezeigt ist, siehe [28, 29]. Die berechneten Eigenwerte sind in Tabelle A.1 dargestellt.

3.1.2 Stabilität der Dynamik des Orientierungsfehlers

Die Ausführungen für die Wahl des Orientierungsfehlers \mathbf{e}_o sind aus [18, 27] entnommen. Dabei wird die relative Orientierung zweier Rotationsmatrizen

$$\delta\mathbf{R} = \mathbf{R}_d\mathbf{R}_e^T \quad (3.11)$$

verwendet. Die Rotationsmatrix \mathbf{R}_e wird aus (2.4) mit (2.1) bestimmt und gibt die Rotation des Endeffektor-Koordinatensystems Σ_e gegenüber dem Basis-Koordinatensystem Σ_0 an. Werden die Rotationsmatrizen \mathbf{R} als Einheitsquaternionen $\mathbf{Q} = \begin{bmatrix} \eta & \boldsymbol{\varepsilon}^T \end{bmatrix}^T$ dargestellt, für die

$$\eta^2 + \boldsymbol{\varepsilon}^T \boldsymbol{\varepsilon} = 1 \quad (3.12)$$

gilt, gibt η den Skalaranteil und $\boldsymbol{\varepsilon}$ den Vektoranteil an. Das Quaternion $\delta\mathbf{Q}$ der relativen Rotationsmatrix $\delta\mathbf{R}$ aus (3.11) wird mit dem Quaternionenprodukt und (2.7) zu

$$\delta\mathbf{Q} = \begin{bmatrix} \delta\eta \\ \delta\boldsymbol{\varepsilon} \end{bmatrix} = \begin{bmatrix} \eta_e\eta_d + \boldsymbol{\varepsilon}_e^T \boldsymbol{\varepsilon}_d \\ \eta_e\boldsymbol{\varepsilon}_d - \eta_d\boldsymbol{\varepsilon}_e - \mathbf{S}(\boldsymbol{\varepsilon}_d)\boldsymbol{\varepsilon}_e \end{bmatrix} \quad (3.13)$$

angeschrieben. Die Definition des Quaternionenprodukts \otimes und eine detaillierte Berechnung von (3.13) ist in Abschnitt A.1.1 gezeigt. Danach wird die positiv definite Lyapunov Funktion

$$\begin{aligned} V(\boldsymbol{\zeta}) &= K_o(\eta_d - \eta_e)^2 + (\boldsymbol{\varepsilon}_d - \boldsymbol{\varepsilon}_e)^T \mathbf{K}_o(\boldsymbol{\varepsilon}_d - \boldsymbol{\varepsilon}_e) + \frac{1}{2}(\boldsymbol{\omega}_d - \boldsymbol{\omega}_e)^T(\boldsymbol{\omega}_d - \boldsymbol{\omega}_e) \\ &= K_o\tilde{\eta}^2 + \tilde{\boldsymbol{\varepsilon}}^T \mathbf{K}_o\tilde{\boldsymbol{\varepsilon}} + \frac{1}{2}(\tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}))^T \tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}) \end{aligned} \quad (3.14)$$

mit den Zuständen $\boldsymbol{\zeta} = (\tilde{\eta}, \tilde{\boldsymbol{\varepsilon}}, \dot{\tilde{\eta}}, \dot{\tilde{\boldsymbol{\varepsilon}}})$ definiert, wobei für die Zeitableitung des Quaternions mit (2.7)

$$\dot{\mathbf{Q}} = \begin{bmatrix} \dot{\eta} \\ \dot{\boldsymbol{\varepsilon}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \otimes \begin{bmatrix} \eta \\ \boldsymbol{\varepsilon} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -\boldsymbol{\omega}^T \\ \boldsymbol{\omega} & \mathbf{S}(\boldsymbol{\omega}) \end{bmatrix} \begin{bmatrix} \eta \\ \boldsymbol{\varepsilon} \end{bmatrix} \quad (3.15)$$

gilt, siehe [18]. Dabei stellt $\tilde{\eta} = \eta_d - \eta_e$, $\tilde{\boldsymbol{\varepsilon}} = \boldsymbol{\varepsilon}_d - \boldsymbol{\varepsilon}_e$ und $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega}_d - \boldsymbol{\omega}_e$ dar. Außerdem wird \mathbf{K}_o zu

$$\mathbf{K}_o = K_o\mathbf{E} \quad (3.16)$$

festgelegt. Wird (3.14) nach der Zeit abgeleitet, erhält man nach kurzer Rechnung, siehe Abschnitt A.1.3,

$$\begin{aligned} \dot{V}(\boldsymbol{\zeta}) &= -(\boldsymbol{\omega}_d - \boldsymbol{\omega}_e)^T \mathbf{K}_\omega(\boldsymbol{\omega}_d - \boldsymbol{\omega}_e) + (\boldsymbol{\omega}_d - \boldsymbol{\omega}_e)^T \mathbf{K}_o(\delta\boldsymbol{\varepsilon} - \mathbf{e}_o) \\ &= -(\tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}))^T \mathbf{K}_\omega\tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}) + (\tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}))^T \mathbf{K}_o(\delta\boldsymbol{\varepsilon} - \mathbf{e}_o). \end{aligned} \quad (3.17)$$

Wird der Orientierungsfehler \mathbf{e}_o zu

$$\mathbf{e}_o = \delta \boldsymbol{\varepsilon} \quad (3.18)$$

gewählt, erhält man durch Einsetzen in (3.17) die negativ semidefinite zeitliche Ableitung der Lyapunov Funktion

$$\dot{V}(\boldsymbol{\zeta}) = -(\tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}))^T \mathbf{K}_\omega \tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}) . \quad (3.19)$$

Durch Untersuchung der Menge $\mathcal{R} = \{\boldsymbol{\zeta} : \dot{V}(\boldsymbol{\zeta}) = 0\}$ ergibt sich $\tilde{\boldsymbol{\omega}}(\boldsymbol{\zeta}) = \mathbf{0}$, wodurch aus (3.7)

$$\mathbf{0} = \mathbf{K}_o \mathbf{e}_o(\boldsymbol{\zeta}) \quad (3.20)$$

folgt. Da \mathbf{K}_o positiv definit ist, verbleibt man nur dann in der Menge \mathcal{R} , wenn $\mathbf{e}_o = \mathbf{0}$ gilt. Da die größte positiv invariante Teilmenge von $\mathcal{R} = \{\boldsymbol{\zeta} : \dot{V}(\boldsymbol{\zeta}) = 0\}$ somit die Nullmenge $\{\mathbf{0}\}$ ist, folgt aus dem Invarianzprinzip von LaSalle die asymptotische Stabilität der Ruhelage $\boldsymbol{\zeta} = \mathbf{0}$. Da die Niveaumenge

$$\{\boldsymbol{\zeta} : \sup V(\boldsymbol{\zeta}) \leq L\}, \quad \forall L > 0 \quad (3.21)$$

kompakt ist, folgt das $V(\boldsymbol{\zeta})$ aus (3.14) eine eigentliche Abbildung ist und somit die globale asymptotische Stabilität der Ruhelage, siehe [27, 28, 30–32].

3.2 Regelung der Nulldynamik

In diesem Abschnitt wird auf die Regelung der Nulldynamik des Roboters eingegangen. Diese ist notwendig, da die Gelenkskoordinaten eine um eins höhere Dimension als die Endeffektor-Koordinaten haben. Somit besteht ein zusätzlicher Freiheitsgrad, der zum Erreichen eines Punkts im Raum mit beliebiger Orientierung des Endeffektors notwendig ist. Um diesen Freiheitsgrad zu regeln, wird ein zusätzlicher Term zur Stellgröße $\boldsymbol{\tau}_d$ aus (3.6) addiert, sodass

$$\boldsymbol{\tau}_d = \tilde{\mathbf{M}}(\mathbf{q}) \left(\begin{array}{c} \left[\begin{array}{c} (\mathbf{J}_v(\mathbf{q}))^\dagger (\boldsymbol{\nu}_p - \dot{\mathbf{J}}_v(\mathbf{q})\dot{\mathbf{q}}) \\ (\mathbf{J}_\omega(\mathbf{q}))^\dagger (\boldsymbol{\nu}_o - \dot{\mathbf{J}}_\omega(\mathbf{q})\dot{\mathbf{q}}) \end{array} \right] + \left(\mathbf{E} - \begin{bmatrix} \mathbf{J}_v(\mathbf{q}) \\ \mathbf{J}_\omega(\mathbf{q}) \end{bmatrix}^\dagger \begin{bmatrix} \mathbf{J}_v(\mathbf{q}) \\ \mathbf{J}_\omega(\mathbf{q}) \end{bmatrix} \right) \boldsymbol{\nu}_n \end{array} \right) \quad (3.22)$$

gilt, siehe [18]. Durch diesen Term wird der Vektor $\boldsymbol{\nu}_n$ in Gelenkskoordinaten in den Nullraum projiziert. Dieser Vektor kann verwendet werden, um Beschränkungen einzuhalten. Um die Gelenklimits bestmöglich auszunutzen, wird

$$\boldsymbol{\nu}_n = \ddot{\mathbf{q}} = -\mathbf{K}_{vn}\dot{\mathbf{q}} - \mathbf{K}_{pn}\mathbf{q} \quad (3.23)$$

gewählt, wobei die Matrizen \mathbf{K}_{vn} und \mathbf{K}_{pn} die Dynamik vorgeben. Dadurch wird die Bewegung im Nullraum gedämpft und zusätzlich werden die Gelenke im Nullraum in die Mitte der beidseitigen Achsbeschränkungen geregelt. Schreibt man (3.23) mit dem Zustand $\boldsymbol{\chi} = \begin{bmatrix} \mathbf{q}^T & \dot{\mathbf{q}}^T \end{bmatrix}^T$ als

$$\dot{\boldsymbol{\chi}} = \mathbf{A}_n \boldsymbol{\chi} = \begin{bmatrix} \mathbf{0} & \mathbf{E} \\ -\mathbf{K}_{pn} & -\mathbf{K}_{vn} \end{bmatrix} \boldsymbol{\chi}, \quad (3.24)$$

berechnet sich das charakteristische Polynom zu

$$\det(\lambda_n \mathbf{E} - \mathbf{A}_n) = \prod_{i=1}^7 \left(\lambda_n^2 + \lambda_n K_{vn,i} + K_{pn,i} \right) . \quad (3.25)$$

Der Index i gibt hier den Eintrag der Diagonalmatrix \mathbf{K} an. Damit das charakteristische Polynom (3.25) ein Hurwitzpolynom ist, werden die Matrizen \mathbf{K}_{pn} und \mathbf{K}_{vn} wie in Tabelle A.1 gewählt. Die Eigenwerte von \mathbf{A}_n haben dadurch negativen Realteil, wodurch die Ruhelage $\boldsymbol{\chi} = \mathbf{0}$ global asymptotisch stabil ist, siehe [28, 29]. Die berechneten Eigenwerte sind in Tabelle A.1 dargestellt.

In diesem Kapitel wurde ein Regelgesetz der inversen Dynamik hergeleitet. Dafür wurde das im vorigen Kapitel berechnete Modell in Endeffektor-Koordinaten transformiert. Dadurch wurde eine Stellgröße gefunden, womit eine Fehlerdynamik vorgegeben werden kann. Als Orientierungsfehler wurde dabei ein Quaternionenfehler verwendet. Die Stabilität der Positionsregelung wurde mittels Eigenwerten und die Stabilität der Orientierungsregelung durch die Lyapunov-Theorie gezeigt. Für den verwendeten Roboter ist eine Regelung der Nulldynamik notwendig, die ebenfalls vorgestellt wurde. Zusammengefasst lässt sich das Regelgesetz mit den Gleichungen (3.22) mit (3.7) und (3.23) anschreiben.

4 Pfadplanung

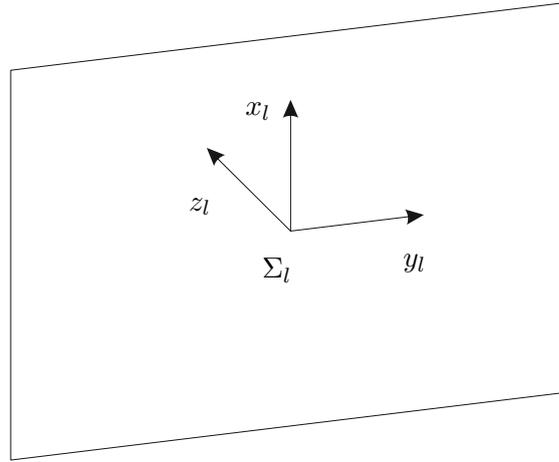
Der Endeffektor des Roboters folgt durch das Regelgesetz aus Kapitel 3 einem definierten Pfad, der im Basis-Koordinatensystem des Roboters Σ_0 ausgedrückt werden muss. Die Bestimmung dieses Pfades wird in diesem Kapitel erläutert. Zuerst wird im körperfesten Koordinatensystem der Tafel Σ_l ein Pfad konstruiert. Dieser Pfad könnte einem Arbeitsablauf des Roboters entsprechen und wird in Abschnitt 4.1 vorgestellt. Damit dieser Arbeitsablauf auf einer bewegten Tafel ausgeführt werden kann, ist eine Messung der Pose der Tafel notwendig. Das Messsystem und die Verarbeitung der Messung sind in Abschnitt 4.2 beschrieben. In Abschnitt 4.3 wird der Pfad des Arbeitsablaufs mit Hilfe der Messinformation in das Basis-Koordinatensystem Σ_0 des Roboters transformiert. Zuletzt wird in Abschnitt 4.4 der Fortschritt des Arbeitsablaufs auf der Tafel dynamisch verändert.

4.1 Pfade im körperfesten Koordinatensystem der Tafel

Um Arbeitsabläufe auf einem Werkstück zu simulieren, werden in diesem Abschnitt Pfade im körperfesten Tafel-Koordinatensystem Σ_l festgelegt. Diesen Pfaden soll der Endeffektor des Roboters relativ zur Tafel folgen. Dabei zeichnet der Stift am Roboter die Pfade auf die Tafel. Dafür werden in Abschnitt 4.1.1 Pfade konstruiert, die sich permanent auf der Oberfläche einer ebenen Tafel befinden. Um eine *pick and place* Aufgabe zu simulieren, wird in Abschnitt 4.1.2 ein Pfad geplant, der stückweise von der Oberfläche der Tafel abhebt. Somit entsteht ein unterbrochener gezeichneter Pfad auf der Tafel. Damit der Roboter auf eine gekrümmte Fläche zeichnen kann, muss der Pfad auf diese Fläche projiziert werden. Beispielhaft wird die Projektion der Pfade in Abschnitt 4.1.3 auf eine zylindrische Tafel gezeigt.

4.1.1 Ebene Pfade

In diesem Abschnitt werden beispielhaft zwei Pfade vorgestellt, die sich in der Ebene der Tafel befinden. Dabei handelt es sich um eine Lissajous-Kurve und eine Kreiskurve. Die Kurven werden in Parameterdarstellung durch die Koordinaten (x_l, y_l, z_l) in einem Koordinatensystem im Mittelpunkt der Tafel Σ_l angegeben, das in Abbildung 4.1 dargestellt ist.

Abbildung 4.1: Tafel-Koordinatensystem Σ_l im Mittelpunkt der ebenen Tafel.

Lissajous-Kurve

Die gewählte Lissajous-Kurve besitzt die Parameterdarstellung

$$\mathbf{d}_l^d(t) = \begin{bmatrix} x_l(t) \\ y_l(t) \\ z_l(t) \end{bmatrix} = \begin{bmatrix} r \sin(2\varphi(t)) \\ r \cos(\varphi(t)) \\ 0 \end{bmatrix}. \quad (4.1)$$

Dieser Vektor beschreibt die Position des Pfades ausgehend vom Tafel-Koordinatensystem Σ_l . Dabei gibt r den Radius und φ den Pfadfortschritt an. Im Regelgesetz (3.7) werden die ersten beiden zeitlichen Ableitungen des Pfades im Basis-Koordinatensystem Σ_0 verwendet, wobei die zweite zeitliche Ableitung vernachlässigt wird. Die erste zeitliche Ableitung des Pfades im Basis-Koordinatensystem Σ_0 wird in Abschnitt 4.3 berechnet, wozu die erste zeitliche Ableitung des Pfades (4.1) im Tafel-Koordinatensystem Σ_l

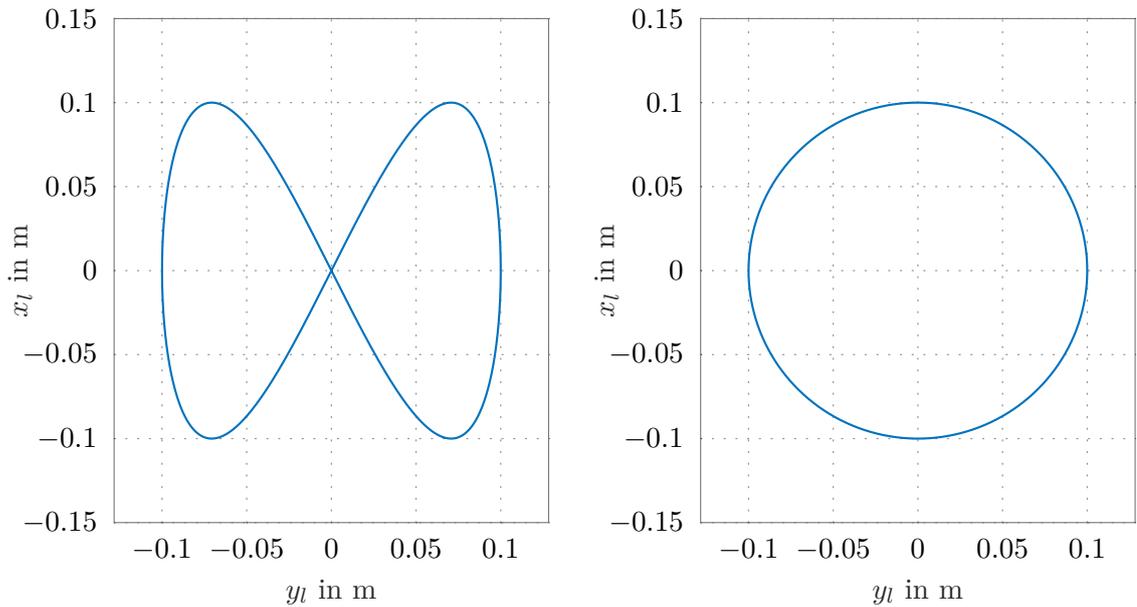
$$\dot{\mathbf{d}}_l^d(t) = \begin{bmatrix} \dot{x}_l(t) \\ \dot{y}_l(t) \\ \dot{z}_l(t) \end{bmatrix} = \begin{bmatrix} 2r \cos(2\varphi(t)) \dot{\varphi}(t) \\ -r \sin(\varphi(t)) \dot{\varphi}(t) \\ 0 \end{bmatrix} \quad (4.2)$$

benötigt wird. Die Lissajous-Kurve von (4.1) wird in Abbildung 4.2 links dargestellt.

Kreiskurve

Die Kreiskurve wird durch

$$\mathbf{d}_l^d(t) = \begin{bmatrix} x_l(t) \\ y_l(t) \\ z_l(t) \end{bmatrix} = \begin{bmatrix} r \sin(\varphi(t)) \\ r \cos(\varphi(t)) \\ 0 \end{bmatrix} \quad (4.3)$$

Abbildung 4.2: Lissajous-Kurve und Kreiskurve im Tafel-Koordinatensystem Σ_l .

beschrieben. Die zeitliche Ableitung des Pfades entspricht

$$\dot{\mathbf{d}}_l^d(t) = \begin{bmatrix} \dot{x}_l(t) \\ \dot{y}_l(t) \\ \dot{z}_l(t) \end{bmatrix} = \begin{bmatrix} r \cos(\varphi(t)) \dot{\varphi}(t) \\ -r \sin(\varphi(t)) \dot{\varphi}(t) \\ 0 \end{bmatrix}. \quad (4.4)$$

Die Kreiskurve von (4.3) wird in Abbildung 4.2 rechts dargestellt.

Um den Pfad für das System aus (3.7) zu beschreiben, ist zusätzlich zur Position die Orientierung notwendig. Dazu wird ein neues Koordinatensystem auf dem momentanen Punkt am Pfad Σ_d eingeführt. Die Matrix

$$\mathbf{T}_l^d = \begin{bmatrix} \mathbf{E} & \mathbf{d}_l^d \\ \mathbf{0} & 1 \end{bmatrix} \quad (4.5)$$

gibt die homogene Transformation (2.1) vom Koordinatensystem am Pfad Σ_d zum Tafel-Koordinatensystem Σ_l an. Dabei wird die Gleichung (4.1) oder (4.3) als Translation verwendet. Die Rotationsmatrix wird zur Einheitsmatrix \mathbf{E} gewählt, damit der Stift senkrecht auf die Oberfläche der ebenen Tafel steht, vergleiche Abbildung 2.1 mit Abbildung 4.1. Die zeitliche Ableitung von (4.5) wird mit (4.2) oder (4.4) über

$$\dot{\mathbf{T}}_l^d = \begin{bmatrix} \mathbf{0} & \dot{\mathbf{d}}_l^d \\ \mathbf{0} & 0 \end{bmatrix} \quad (4.6)$$

berechnet.

4.1.2 Räumliche Pfade

Bei vielen Arbeitsabläufen muss der Endeffektor vom Werkstück abheben. Ein Beispiel dafür sind *pick and place* Aufgaben. Um diese Aufgaben zu simulieren, soll der Stift von der Tafel absetzen. Dadurch wird auf der Tafel eine unterbrochene Kurve gezeichnet.

Dafür wird der Abstand in z_l -Richtung verändert, siehe Abbildung 4.1. Für den Übergang während des Abhebens von der Oberfläche wird ein Polynom fünften Grades verwendet

$$z_{pol}(\varphi(t)) = a_5\varphi(t)^5 + a_4\varphi(t)^4 + a_3\varphi(t)^3 + a_2\varphi(t)^2 + a_1\varphi(t) + a_0 . \quad (4.7)$$

Werden die Anfangsbedingungen

$$z_{pol}(\varphi_{sta}) = z_{sta}, \quad z_{pol}(\varphi_{end}) = z_{end} \quad (4.8a)$$

$$\dot{z}_{pol}(\varphi_{sta}) = 0, \quad \dot{z}_{pol}(\varphi_{end}) = 0 \quad (4.8b)$$

$$\ddot{z}_{pol}(\varphi_{sta}) = 0, \quad \ddot{z}_{pol}(\varphi_{end}) = 0 \quad (4.8c)$$

in (4.7) und die ersten beiden zeitlichen Ableitungen eingesetzt, lassen sich die Koeffizienten a_0, \dots, a_5 berechnen.

Die erste zeitliche Ableitung von (4.7) lässt sich über

$$\dot{z}_{pol}(\varphi(t)) = \left(5a_5\varphi(t)^4 + 4a_4\varphi(t)^3 + 3a_3\varphi(t)^2 + 2a_2\varphi(t) + a_1 \right) \dot{\varphi}(t) \quad (4.9)$$

anschreiben. Die gesamte Funktion $z_l(\varphi(t))$, inklusive Phasen mit konstantem Abstand, lässt sich als stückweise 2-fach stetig differenzierbare Funktion

$$z_l(\varphi(t)) = \begin{cases} z_{sta} & \text{für } 0 \leq \tilde{\varphi} < \varphi_1 \\ z_{pol}(\tilde{\varphi}(t) - \varphi_1) & \text{für } \varphi_1 \leq \tilde{\varphi} < \varphi_2 \\ z_{end} & \text{für } \varphi_2 \leq \tilde{\varphi} < \varphi_3 \\ z_{end} + z_{sta} - z_{pol}(\tilde{\varphi}(t) - \varphi_3) & \text{für } \varphi_3 \leq \tilde{\varphi} < \varphi_4 \\ z_{sta} & \text{für } \varphi_4 \leq \tilde{\varphi} < \varphi_5 \end{cases} \quad (4.10)$$

mit

$$\tilde{\varphi}(t) = \text{mod}(\varphi(t) - \varphi_v, \varphi_5) \quad (4.11)$$

darstellen. Die zeitliche Ableitung von (4.10) lautet

$$\dot{z}_l(\varphi(t)) = \begin{cases} 0 & \text{für } 0 \leq \tilde{\varphi} < \varphi_1 \\ \dot{z}_{pol}(\tilde{\varphi}(t) - \varphi_1) & \text{für } \varphi_1 \leq \tilde{\varphi} < \varphi_2 \\ 0 & \text{für } \varphi_2 \leq \tilde{\varphi} < \varphi_3 \\ -\dot{z}_{pol}(\tilde{\varphi}(t) - \varphi_3) & \text{für } \varphi_3 \leq \tilde{\varphi} < \varphi_4 \\ 0 & \text{für } \varphi_4 \leq \tilde{\varphi} < \varphi_5 . \end{cases} \quad (4.12)$$

Mit den Grenzen $\varphi_1, \dots, \varphi_5$ und φ_v lassen sich die Dauer, Anzahl und Verschiebung der Phasen bestimmen. Die verwendeten Werte, sodass der Pfad fünf mal während einer Umdrehung der Kurve von der Oberfläche abhebt, sind in Tabelle A.2 angeführt. Die Koordinaten (x_l, y_l) aus (4.1) oder (4.3) und z_l aus (4.10) sowie deren zeitliche Ableitungen (4.2), (4.4) und (4.12) lassen sich über (4.5) und (4.6) berechnen. Der Verlauf des Pfades in z_l -Richtung ist in Abbildung 4.3 links und in Kombination mit den berechneten (x_l, y_l) -Koordinaten von (4.3) in Abbildung 4.3 rechts dargestellt.

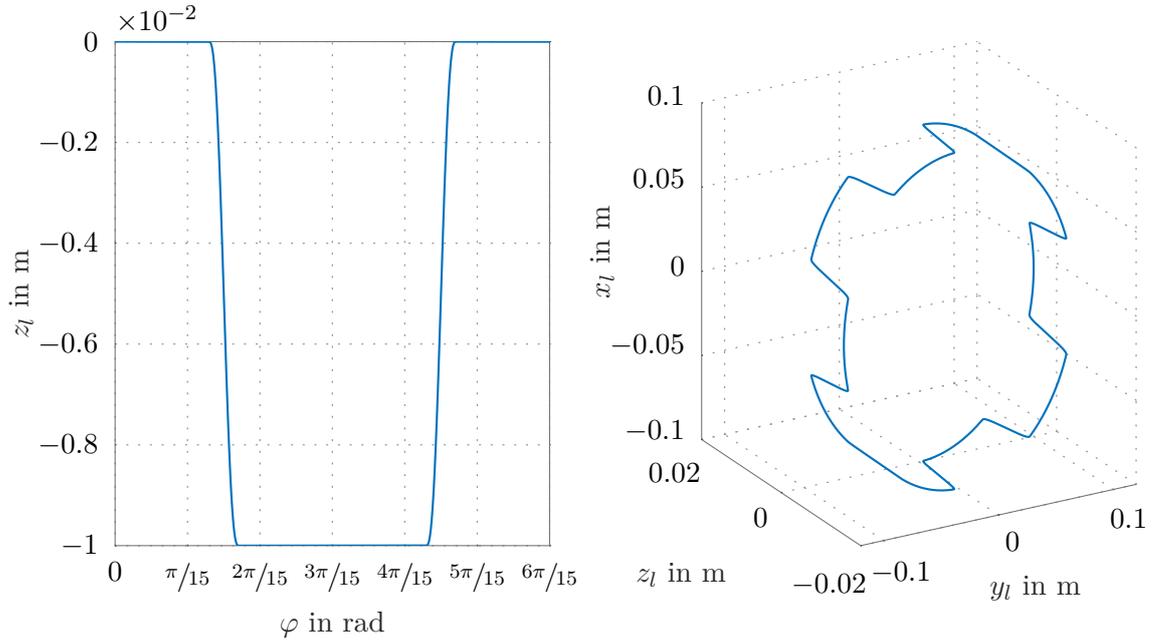


Abbildung 4.3: Pfad zum Abheben des Endeffektors von der Oberfläche der Tafel.

4.1.3 Pfade auf gekrümmten Flächen

Um die Pfade aus Abschnitt 4.1.1 und Abschnitt 4.1.2 auf gekrümmten Flächen zeichnen zu können, müssen die Pfade auf diese Flächen projiziert werden. Beispielhaft wird in diesem Abschnitt die Projektion auf eine zylindrische Tafel gezeigt. Dazu wird wie in Abbildung 4.1 ein Tafel-Koordinatensystem Σ_l im Mittelpunkt der Zylinderoberfläche eingeführt, das in Abbildung 4.4 links dargestellt ist. Die Koordinaten \mathbf{d}_l^d der Pfade auf der ebenen Tafel aus (4.1), (4.3) sowie (4.10) werden in diesem Abschnitt mit $(\hat{x}_l, \hat{y}_l, \hat{z}_l)$ bezeichnet. Wird die Länge \hat{y}_l als Kreisbogen betrachtet, kann der Winkel

$$\gamma(t) = \frac{\hat{y}_l(t)}{l} \quad (4.13)$$

eingeführt werden. Der Radius des Zylinders l , siehe Tabelle A.2, und der Winkel γ sind in Abbildung 4.4 rechts dargestellt. Die projizierten Koordinaten berechnen sich mit (4.13) zu

$$\mathbf{d}_l^d(t) = \begin{bmatrix} x_l(t) \\ y_l(t) \\ z_l(t) \end{bmatrix} = \begin{bmatrix} \hat{x}_l \\ (l - \hat{z}_l) \sin(\gamma) \\ l - (l - \hat{z}_l) \cos(\gamma) \end{bmatrix} \quad (4.14)$$

Die zeitliche Ableitung ergibt sich zu

$$\dot{\mathbf{d}}_l^d(t) = \begin{bmatrix} \dot{x}_l(t) \\ \dot{y}_l(t) \\ \dot{z}_l(t) \end{bmatrix} = \begin{bmatrix} \dot{\hat{x}}_l \\ (l - \hat{z}_l) \cos(\gamma) \dot{\gamma} - \dot{\hat{z}}_l \sin(\gamma) \\ (l - \hat{z}_l) \sin(\gamma) \dot{\gamma} + \dot{\hat{z}}_l \cos(\gamma) \end{bmatrix}, \quad (4.15)$$

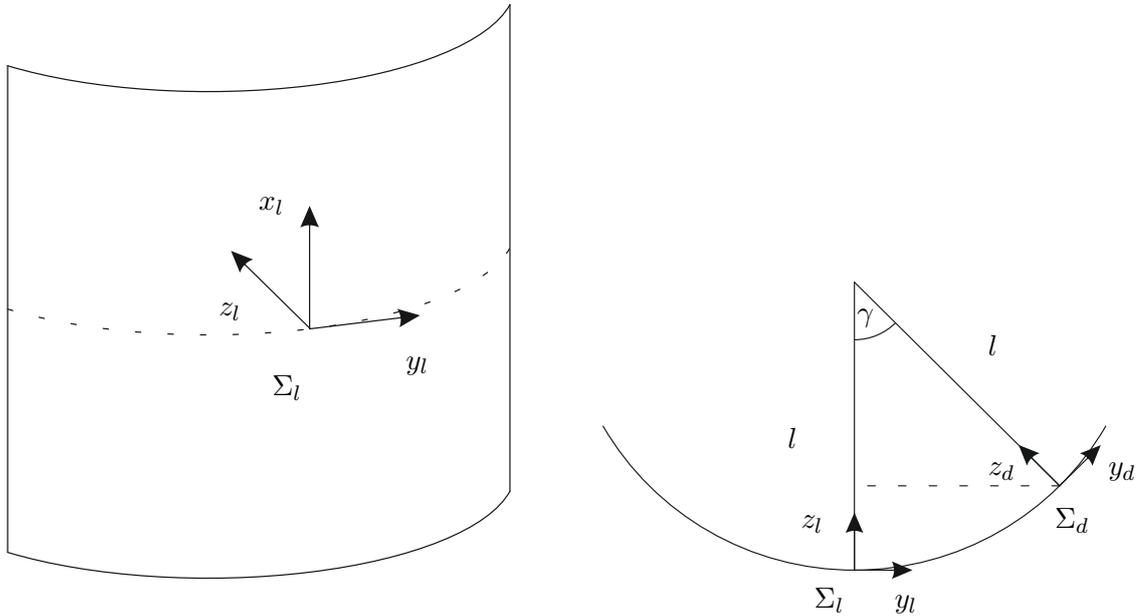


Abbildung 4.4: Tafel-Koordinatensystem Σ_l im Mittelpunkt der zylindrischen Tafel.

wobei

$$\dot{\gamma}(t) = \frac{\dot{\hat{y}}_l(t)}{l} \quad (4.16)$$

verwendet wird.

Damit der Stift des Roboters auf dem gesamten Pfad normal zur Oberfläche des Zylinders steht, muss die gewünschte Orientierung des Endeffektors in Abhängigkeit des Pfadfortschritts gegenüber dem Tafel-Koordinatensystem Σ_l angepasst werden. Für eine Drehung um die x_l -Achse von Σ_l um den Winkel γ , wird die Rotationsmatrix

$$\mathbf{R}_l^d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) \\ 0 & \sin(\gamma) & \cos(\gamma) \end{bmatrix} \quad (4.17)$$

und deren zeitliche Ableitung

$$\dot{\mathbf{R}}_l^d = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\sin(\gamma) & -\cos(\gamma) \\ 0 & \cos(\gamma) & -\sin(\gamma) \end{bmatrix} \dot{\gamma}(t) \quad (4.18)$$

angeschrieben, siehe [18]. Es wird die homogene Transformation aus (2.1) mit (4.14) und (4.17) als

$$\mathbf{T}_l^d = \begin{bmatrix} \mathbf{R}_l^d & \mathbf{d}_l^d \\ \mathbf{0} & 1 \end{bmatrix} \quad (4.19)$$

und mit den zeitlichen Ableitungen (4.15) und (4.18)

$$\dot{\mathbf{T}}_l^d = \begin{bmatrix} \dot{\mathbf{R}}_l^d & \dot{\mathbf{d}}_l^d \\ \mathbf{0} & 0 \end{bmatrix} \quad (4.20)$$

dargestellt. Der projizierte Pfad einer Lissajous-Kurve auf der zylindrischen Oberfläche ist in Abbildung 4.5 gezeigt.

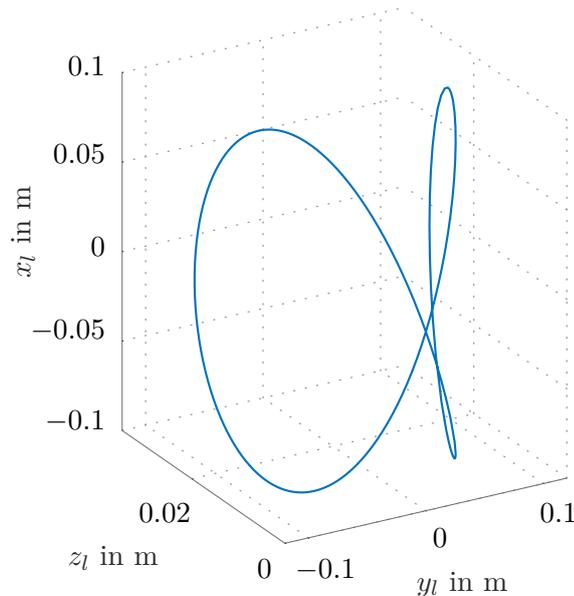


Abbildung 4.5: Projektion der Lissajous-Kurve auf der zylindrischen Tafel.

4.2 Messung der Pose der Tafel

In Abschnitt 4.1 ist die Berechnung des Pfades im körperfesten Tafel-Koordinatensystem Σ_l gezeigt. Damit diesem Pfad vom Roboter gefolgt werden kann, muss der Pfad in das Basis-Koordinatensystem des Roboters Σ_0 transformiert werden. In diesem Abschnitt wird die dafür erforderliche Messung der Pose der Tafel beschrieben. Dazu werden sechs *Prime 17W* Kameras der Firma OPTITRACK verwendet, die mit der Software *Motive* die Position und Orientierung eines definierten Messobjekts messen. Dieses Messobjekt ist auf der Tafel befestigt und besteht aus sechs Markern, die von den Kameras ausgestrahlte Infrarotstrahlung reflektieren. Die Messungen werden mit einer maximalen Wiederholrate von 360 Hz und einer angegebenen Latenzzeit von 2.8 ms aufgenommen, siehe [33]. Um eine Transformation vom Koordinatensystem der Tafel Σ_l in das Basis-Koordinatensystem des Roboters Σ_0 zu erhalten, muss man zusätzliche Transformationen zum Messergebnis berücksichtigen. Diese Transformationen sind in Abschnitt 4.2.1 erklärt. Die Verarbeitung der Messdaten durch ein Filter ist in Abschnitt 4.2.2 beschrieben.

4.2.1 Transformation der Messergebnisse

Das Messsystem gibt die Position und Orientierung des Messobjekts bezüglich eines eigenen Basis-Koordinatensystems Σ_b an. Da jedoch das Messobjekt nicht im Koordinatensystem der Tafel Σ_l und das Basis-Koordinatensystem Σ_b nicht im Basis-Koordinatensystem Σ_0 des Roboters liegt, sind zusätzliche Transformationen zur Messung notwendig. Dabei werden die elementaren Rotationsmatrizen

$$\mathbf{R}_x(\beta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\beta & -s_\beta \\ 0 & s_\beta & c_\beta \end{bmatrix}, \quad (4.21a)$$

$$\mathbf{R}_y(\beta) = \begin{bmatrix} c_\beta & 0 & s_\beta \\ 0 & 1 & 0 \\ -s_\beta & 0 & c_\beta \end{bmatrix}, \quad (4.21b)$$

$$\mathbf{R}_z(\beta) = \begin{bmatrix} c_\beta & -s_\beta & 0 \\ s_\beta & c_\beta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (4.21c)$$

verwendet, die eine Drehung um die x, y oder z -Achse um den Winkel β berechnen, siehe [18]. Es wird die verkürzte Schreibweise c_β (s_β) anstatt $\cos(\beta)$ ($\sin(\beta)$) verwendet, vergleiche (2.2).

Transformation für ebene Tafel

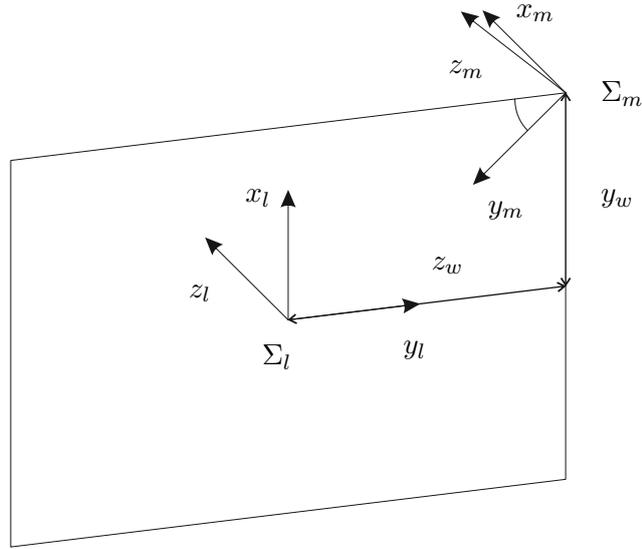
Die Anordnung des Messobjekts auf der ebenen Tafel gegenüber dem Koordinatensystem der Tafel Σ_l ist in Abbildung 4.6 gezeigt. Dem Messobjekt ist ein körperfestes Koordinatensystem Σ_m zugeordnet. Die Transformation von Σ_l zu Σ_m ergibt sich zu

$$\mathbf{T}_m^l = \begin{bmatrix} \mathbf{R}_x(-45^\circ) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_x(-90^\circ) & \mathbf{d}_w \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_y(90^\circ) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}. \quad (4.22)$$

Die Einträge des Translationsvektors

$$\mathbf{d}_w = [x_w \quad y_w \quad z_w]^T \quad (4.23)$$

sind in Abbildung 4.6 gezeigt, wobei die Werte in Tabelle A.2 dargestellt sind. Durch (4.22) wird das Koordinatensystem Σ_m zuerst um -45° um die x_m -Achse gedreht. Somit kann der Translationsvektor (4.23) einfach angegeben werden. Danach wird das Koordinatensystem mit einer Drehung von -90° um x und anschließend mit einer Drehung von 90° um y in das Koordinatensystem der Tafel Σ_l gedreht.

Abbildung 4.6: Transformation zum Tafel-Koordinatensystem Σ_l bei ebener Tafel.

Transformation für zylindrische Tafel

Das Messobjekt wird auf der zylindrischen Tafel an einer anderen Stelle befestigt. Daher wird eine neue Transformation vom Koordinatensystem Σ_m zum Tafel-Koordinatensystem Σ_l im Mittelpunkt benötigt. In Abbildung 4.7 werden die beiden Koordinatensysteme gezeigt, wodurch sich die Transformation

$$\mathbf{T}_m^l = \begin{bmatrix} \mathbf{R}_x(-45^\circ) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_y(150^\circ) & \mathbf{d}_c \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_z(-90^\circ) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \quad (4.24)$$

berechnet. Der Translationsvektor \mathbf{d}_c wird durch

$$\mathbf{d}_c = [x_c \quad y_c \quad z_c]^T \quad (4.25)$$

dargestellt, wobei die Einträge des Vektors in Tabelle A.2 aufgelistet sind.

Transformation in Basis-Koordinatensystem des Roboters

Da das Messsystem die Messung bezüglich eines eigenen Basis-Koordinatensystems Σ_b vornimmt, müssen die Messwerte in das Basis-Koordinatensystem des Roboters Σ_0 transformiert werden. Die Koordinatensysteme sind in Abbildung 4.8 dargestellt, vergleiche Abbildung 2.1.

Somit wird die Transformation mit

$$\mathbf{T}_0^b = \begin{bmatrix} \mathbf{R}_x(90^\circ) & \mathbf{d}_r \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_y(-90^\circ) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \quad (4.26)$$

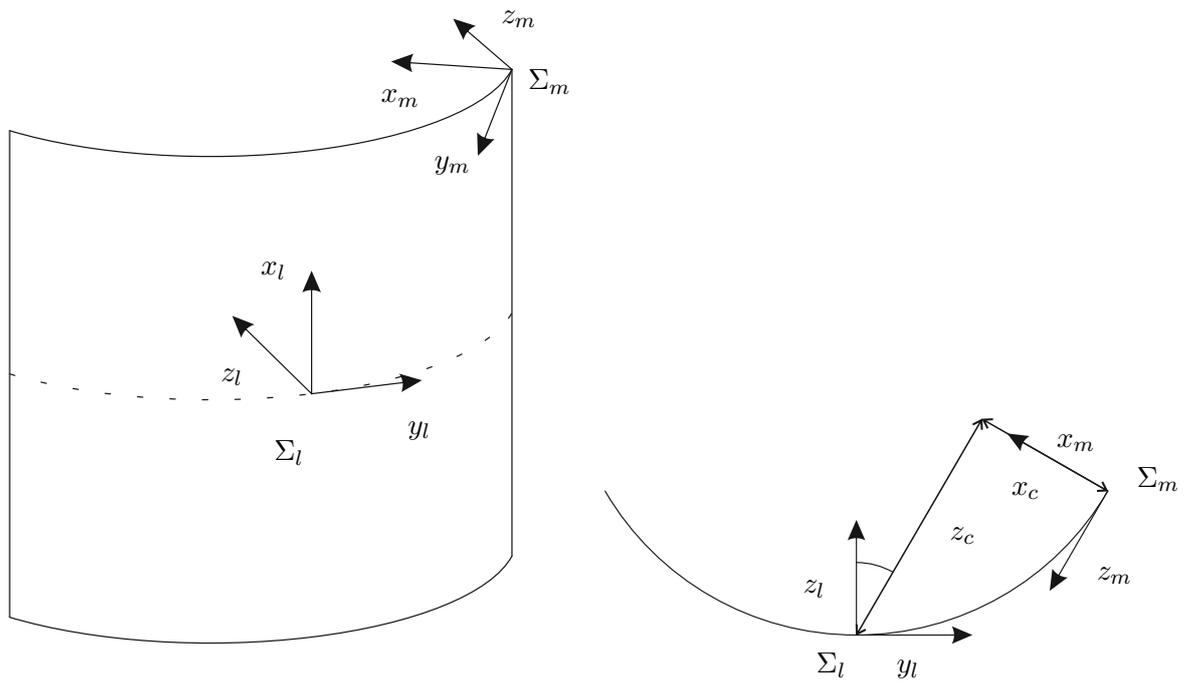
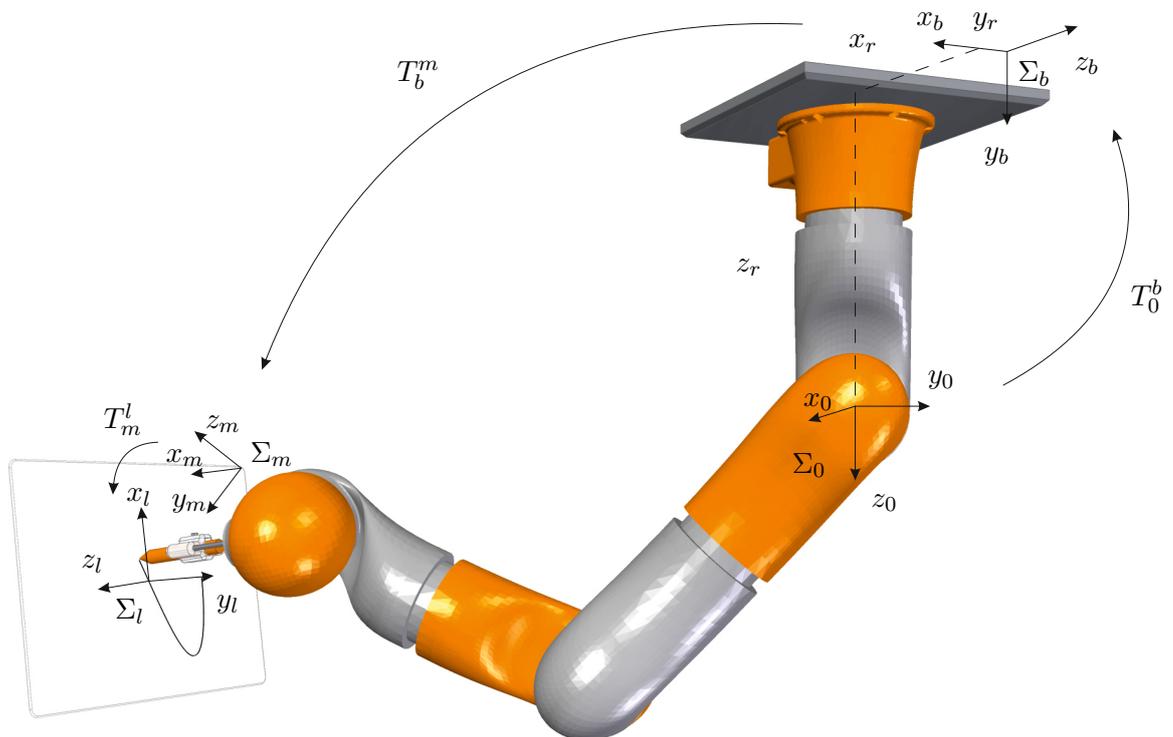
Abbildung 4.7: Transformation zum Tafel-Koordinatensystem Σ_l bei zylindrischer Tafel.

Abbildung 4.8: Transformationen der Messung unter Verwendung der CAD-Modelle der Robotergelenke aus [1].

berechnet. Dabei stellt \mathbf{d}_r den Translationsvektor

$$\mathbf{d}_r = \begin{bmatrix} x_r & y_r & z_r \end{bmatrix}^T \quad (4.27)$$

dar, wobei die Werte in Tabelle A.2 angegeben sind.

Kombination der Transformationen der Messung

Das Messsystem gibt die Pose des Messobjekts als Position und Quaternion aus. Diese Pose wird als homogene Transformation \mathbf{T}_b^m gemäß (2.1) dargestellt, wobei die Rotationsmatrix aus dem Quaternion berechnet wird, siehe [18]. Die Transformation vom Tafel-Koordinatensystem Σ_l zum Basis-Koordinatensystem des Roboters Σ_0 erhält man mit \mathbf{T}_b^m und den in diesem Abschnitt vorgestellten Transformationen zu

$$\mathbf{T}_0^l = \mathbf{T}_0^b \mathbf{T}_b^m \mathbf{T}_m^l . \quad (4.28)$$

Dabei wird \mathbf{T}_m^l je nach Form der Tafel aus (4.22) oder (4.24) gewählt. Die Transformationen von (4.28) werden am Beispiel für eine ebene Tafel in Abbildung 4.8 dargestellt.

4.2.2 Filterung

Aus den Messwerten der Position und Orientierung des Messobjekts wird die Transformation \mathbf{T}_b^m berechnet. Mit (4.28) wird die Transformation vom Tafel-Koordinatensystem Σ_l in das Basis-Koordinatensystem Σ_0 des Roboters bestimmt. Die Einträge der Transformation \mathbf{T}_b^m sind durch die Messung mit Rauschen behaftet, wodurch dies ebenfalls für die Einträge der Transformation \mathbf{T}_0^l gilt. Weiters sind im Regelgesetz von (3.7) die ersten beiden zeitlichen Ableitungen der gewünschten Pose des Endeffektors gefordert, wobei die zweite zeitliche Ableitung vernachlässigt wird. Daher werden in diesem Abschnitt zwei Ableitungsfiler als Zustandsfilter vorgestellt. Für die Filterung wird aus (4.28) die Translation als kartesische Koordinaten und die Rotation als Einheitsquaternion berechnet. Die verwendeten Filter haben die Form

$$\dot{\boldsymbol{\kappa}} = \mathbf{A}\boldsymbol{\kappa} + \mathbf{B}\mathbf{u}, \quad \boldsymbol{\kappa}(0) = \boldsymbol{\kappa}_0 \quad (4.29)$$

wobei $\boldsymbol{\kappa}$ die gefilterten Werte darstellen. Die Matrizen von (4.29) werden dabei zu

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -b_0 & -b_1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 \\ b_0 \end{bmatrix} \quad (4.30)$$

gewählt.

Filterung der Position

Wird als Eingang $\mathbf{u} = (\mathbf{d}_0^l)^T$ die Translation aus (4.28) mit (2.1) gewählt, entspricht $\boldsymbol{\kappa} = \begin{bmatrix} \check{\mathbf{d}}_0^l & \dot{\check{\mathbf{d}}}_0^l \end{bmatrix}^T$ der gefilterten Position des Ursprungs des Tafel-Koordinatensystems Σ_l sowie deren zeitliche Ableitung. Die Koeffizienten aus (4.30) werden über

$$\det(\lambda_p \mathbf{E} - \mathbf{A}) = \lambda_p^2 + \lambda_p b_1 + b_0 \stackrel{!}{=} (\lambda_p + 40)^2 \quad (4.31)$$

gewählt, sodass das charakteristische Polynom Hurwitz ist.

Filterung der Orientierung

Für das Filter der Orientierung wird der Eingang $\mathbf{u} = \begin{bmatrix} \eta & \boldsymbol{\varepsilon}^T \end{bmatrix}$ als Quaternion von (4.28) mit (2.1) gewählt, wobei die Rotationsmatrix dafür in ein Einheitsquaternion umgerechnet wird, siehe [18]. Daher stellt der Zustand $\boldsymbol{\kappa} = \left[\begin{bmatrix} \check{\eta} & \check{\boldsymbol{\varepsilon}}^T \end{bmatrix}^T \begin{bmatrix} \dot{\eta} & \dot{\boldsymbol{\varepsilon}}^T \end{bmatrix}^T \right]^T$ das gefilterte Quaternion und dessen zeitliche Ableitung dar. Die Koeffizienten aus (4.30) werden gewählt, sodass das charakteristische Polynom Hurwitz ist und die Nullstellen alle bei -5 liegen. Somit ist das Filter langsamer als mit den Koeffizienten aus (4.31).

Diskretisierung des Filters

Das Filter aus (4.29) wird mit einem *zero-order-hold* mit einer Abtastzeit T_a aus Tabelle A.1 diskretisiert, woraus

$$\boldsymbol{\kappa}_{k+1} = \boldsymbol{\Phi}\boldsymbol{\kappa}_k + \boldsymbol{\Gamma}\mathbf{u}_k, \quad \boldsymbol{\kappa}(0) = \boldsymbol{\kappa}_0 \quad (4.32)$$

mit

$$\boldsymbol{\Phi} = \exp(\mathbf{A}T_a), \quad \boldsymbol{\Gamma} = \int_0^{T_a} \exp(\mathbf{A}\tau) d\tau \mathbf{B} \quad (4.33)$$

folgt, siehe [29]. Der Index k stellt hier den Abtastschritt dar.

Darstellung als homogene Transformation

Die gefilterten Werte $\boldsymbol{\kappa}_k$ der Position und der Orientierung werden durch die homogene Transformation

$$\check{\mathbf{T}}_0^l = \begin{bmatrix} \check{\mathbf{R}}_0^l & \check{\mathbf{d}}_0^l \\ \mathbf{0} & 1 \end{bmatrix} \quad (4.34)$$

dargestellt, wofür das gefilterte Quaternion in eine Rotationsmatrix umgerechnet wird, siehe [18]. Um die zeitliche Ableitung von (4.34) zu berechnen, wird (3.15) als

$$\check{\boldsymbol{\omega}} = 2 \left(\check{\eta}\dot{\boldsymbol{\varepsilon}} - \dot{\eta}\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon} \times \dot{\boldsymbol{\varepsilon}} \right) \quad (4.35)$$

angeschrieben, siehe Abschnitt A.1.2. Damit wird mit (2.6) und (2.7) sowie der gefilterten Rotationsmatrix aus (4.34) die zeitlich abgeleitete gefilterte Rotationsmatrix

$$\dot{\check{\mathbf{R}}}_0^l = \mathbf{S}(\check{\boldsymbol{\omega}}_0^l) \check{\mathbf{R}}_0^l \quad (4.36)$$

berechnet. Dadurch wird die zeitliche Ableitung der homogenen Transformation (4.34) als

$$\dot{\check{\mathbf{T}}}_0^l = \begin{bmatrix} \dot{\check{\mathbf{R}}}_0^l & \dot{\check{\mathbf{d}}}_0^l \\ \mathbf{0} & 0 \end{bmatrix} \quad (4.37)$$

dargestellt.

4.3 Bestimmung der gewünschten Pose des Endeffektors

In Abschnitt 4.1 wurde ein Pfad ausgehend vom Koordinatensystem der Tafel Σ_l geplant. Dieser Pfad kann für eine ebene oder zylindrische Tafel berechnet und als Transformation dargestellt werden. Damit der Endeffektor des Roboters diesem Pfad relativ zu einer bewegten Tafel folgen kann, wird die Pose der Tafel gemessen und ebenfalls als Transformation angegeben, wie in Abschnitt 4.2 gezeigt wurde. In diesem Abschnitt werden die beiden Transformationen durch

$$\mathbf{T}_0^d = \check{\mathbf{T}}_0^l \mathbf{T}_l^d \quad (4.38)$$

kombiniert. Somit erhält man eine Transformation vom Koordinatensystem am momentanen Punkt am Pfad Σ_d in das Basis-Koordinatensystem Σ_0 des Roboters. Die Translation \mathbf{d}_0^d der homogenen Transformation (2.1) von (4.38) gibt die Position des momentanen Punkts am Pfad im Basis-Koordinatensystem Σ_0 an. Durch die Rotationsmatrix \mathbf{R}_0^d von (4.38) wird die Rotation vom Koordinatensystem am Pfad Σ_d in das Basis-Koordinatensystem Σ_0 des Roboters beschrieben. Somit ist für das Regelgesetz aus (3.7) die gewünschte Position \mathbf{p}_d und mit (3.11) der Orientierungsfehler \mathbf{e}_o bestimmt. Die zeitliche Ableitung der homogenen Transformation (4.38) wird über

$$\dot{\mathbf{T}}_0^d = \dot{\check{\mathbf{T}}}_0^l \mathbf{T}_l^d + \check{\mathbf{T}}_0^l \dot{\mathbf{T}}_l^d \quad (4.39)$$

berechnet. Aus dieser Transformation wird mit (2.1) der Vektor $\dot{\mathbf{p}}_d$ und über (2.6) und (2.7) $\boldsymbol{\omega}_d$ bestimmt. Für (4.38) und (4.39) werden die Transformationen der gefilterten Messwerte von (4.34) und deren zeitliche Ableitung (4.37) benötigt. Je nach Form der Tafel wird (4.22) oder (4.24) mit (4.26) in (4.28) verwendet. Der dazugehörige Pfad und deren zeitliche Ableitung werden durch (4.5) und (4.6) oder (4.19) und (4.20) beschrieben.

4.4 Dynamischer Pfadfortschritt

In (4.1), (4.3) und (4.10) sowie den daraus folgenden Gleichungen wird der Pfadparameter φ verwendet, um den Pfadfortschritt auf der Tafel anzugeben. Der Verlauf von φ wird selbst gewählt, wodurch er nicht ausschließlich von der Zeit abhängen muss. In diesem Abschnitt wird der Pfadfortschritt abhängig von der Geschwindigkeit der Tafel verlangsamt oder sogar gestoppt. Dafür wird eine Variable ϑ eingeführt, die eine selbst definierte Zeit darstellt. Zu Beginn wird das Polynom

$$\varphi(\vartheta(t)) = c_5 \vartheta(t)^5 + c_4 \vartheta(t)^4 + c_3 \vartheta(t)^3 + c_2 \vartheta(t)^2 + c_1 \vartheta(t) + c_0 \quad (4.40)$$

angesetzt, mit der ein hinreichend glatter Verlauf von φ entsteht, vergleiche (4.7). Es werden die Anfangsbedingungen

$$\varphi(\vartheta_{beg}) = \varphi_{beg}, \quad \varphi(\vartheta_{sto}) = \varphi_{sto} \quad (4.41a)$$

$$\dot{\varphi}(\vartheta_{beg}) = 0, \quad \dot{\varphi}(\vartheta_{sto}) = 0 \quad (4.41b)$$

$$\ddot{\varphi}(\vartheta_{beg}) = 0, \quad \ddot{\varphi}(\vartheta_{sto}) = 0 \quad (4.41c)$$

in (4.40) und die ersten beiden zeitlichen Ableitungen eingesetzt, um die Koeffizienten c_0, \dots, c_5 zu berechnen. Die Anfangsbedingungen sind in Tabelle A.2 aufgelistet, wobei

der Pfadparameter φ vom Wert 0 bis zu einer zweifachen Umdrehung 4π verläuft. Die zeitliche Ableitung des Polynoms (4.40) lautet

$$\dot{\varphi}(\vartheta(t)) = \left(5c_5\vartheta(t)^4 + 4c_4\vartheta(t)^3 + 3c_3\vartheta(t)^2 + 2c_2\vartheta(t) + c_1\right) \dot{\vartheta}(t). \quad (4.42)$$

Wird die Variable ϑ durch die Zeit t ersetzt, erhält man einen hinreichend glatten Verlauf des Pfadparameters φ , der jedoch ausschließlich von der Zeit abhängt. Um den Verlauf von φ abhängig von der Geschwindigkeit u der Tafel zu verändern, wird die Differenzialgleichung

$$\ddot{\vartheta} = -\left(\dot{\vartheta} - 1\right) - u, \quad \vartheta(0) = 0, \quad \dot{\vartheta}(0) = 0 \quad (4.43)$$

betrachtet. Ist der Eingang $u = 0$ stellt sich eine konstante Geschwindigkeit $\dot{\vartheta} = 1$ ein, daher folgt $\vartheta \approx t + \text{const.}$ Wird der Eingang $u \in [0, 1]$ vergrößert, bremst der Fortschritt von ϑ bis $\dot{\vartheta} = \ddot{\vartheta} = 0$ bei $u = 1$.

Um einen glatten Übergang beim Bremsen oder Beschleunigen des Fortschritts von ϑ zu erhalten, wird (4.43) zu

$$\vartheta^{(4)} + b_1 \ddot{\vartheta} + b_0 \dot{\vartheta} = b_0 n_b \left(-(\dot{\vartheta} - 1) - u\right) \quad (4.44)$$

und in Systemdarstellung zu

$$\dot{\mathbf{v}} = \mathbf{f}(\mathbf{v}) + \mathbf{g}u = \begin{bmatrix} \dot{\vartheta} \\ \ddot{\vartheta} \\ \ddot{\vartheta} \\ -b_0 n_b (\dot{\vartheta} - 1) - b_0 \dot{\vartheta} - b_1 \ddot{\vartheta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -b_0 n_b \end{bmatrix} u, \quad \mathbf{v}(0) = \mathbf{0} \quad (4.45)$$

gewählt, wobei für den Zustand $\mathbf{v} = [\vartheta \quad \dot{\vartheta} \quad \ddot{\vartheta} \quad \ddot{\vartheta}]^T$ gilt. Der Parameter n_b passt den Maximalwert der Beschleunigung von 1 auf n_b an, vergleiche (4.43) und rechte Seite von (4.44). Um bei großen Werten des Eingangs u den Pfadfortschritt schneller zu bremsen, wird n_b über

$$n_b = \begin{cases} 3 & \text{für } u > 0.5 \\ 1 & \text{für } u \leq 0.5 \end{cases} \quad (4.46)$$

bestimmt. Die Koeffizienten b_1 und b_0 werden aus

$$\lambda_{\vartheta}^2 (\lambda_{\vartheta}^2 + \lambda_{\vartheta} b_1 + b_0) \stackrel{!}{=} \lambda_{\vartheta}^2 (\lambda_{\vartheta} + 10)^2 \quad (4.47)$$

berechnet.

Die Gleichung (4.45) wird als Näherung mittels der Euler-Integration als diskretes System

$$\mathbf{v}_{k+1} = \mathbf{v}_k + (\mathbf{f}(\mathbf{v}_k) + \mathbf{g}u_k) T_a \quad (4.48)$$

implementiert, siehe [30]. Dabei stellt k den Abtastschritt und T_a die Abtastzeit dar, die in Tabelle A.1 angegeben ist.

Um den Pfadfortschritt abhängig von der Geschwindigkeit der Tafel zu stoppen, wird der Eingang u des Systems (4.45) über

$$u = \begin{cases} 1 & \text{für } u \geq 1 \\ n_u \left\| \dot{\mathbf{d}}_0^j \right\|_2 & \text{für } u < 1 \end{cases} \quad (4.49)$$

berechnet, wobei die 2-Norm der gefilterten Geschwindigkeit der Tafel aus (4.37) verwendet wird. Der Eingang u kann somit maximal den Wert 1 erreichen, was ein Verkleinern von ϑ und somit ein Rückwärtslaufen entlang des Pfades verhindert. Der Parameter n_u passt die Größenordnung von $\left\| \dot{\mathbf{d}}_0^j \right\|_2$ in (4.49) an und ist in Tabelle A.2 dargestellt. Eine Simulation des Verlaufs von φ aus (4.40), wobei ϑ durch das System (4.45) beschrieben wird, ist in Abbildung 4.9 gezeigt. In dieser Abbildung ist der Verlauf des Pfadparameters φ über der

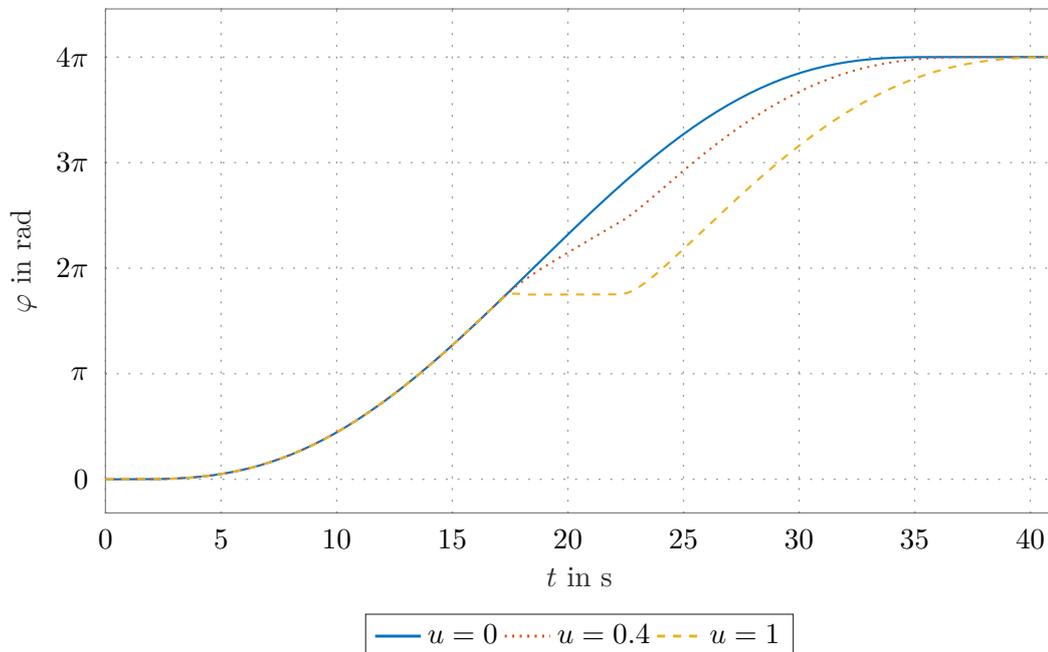


Abbildung 4.9: Veränderung des Fortschritts des Pfadparameters φ im Bereich 17 s bis 22 s.

Zeit t dargestellt. Durch das Polynom (4.40) wird ein hinreichend glatter Verlauf von φ erreicht. Zudem ist ein langsames Beschleunigen und Bremsen am Start- und Endpunkt von φ sichtbar. Somit wird dem Bediener, der die Tafel führt, ausreichend Zeit gegeben, die selbstständige Bewegung des Roboters zum Zeichnen des Pfades zu erkennen. In einem Zeitraum von 0 s bis 17 s ist der Eingang von (4.45) bei allen gezeigten Kurven $u = 0$ und φ folgt dem geplanten Polynom. Von 17 s bis 22 s wird ein Eingang $u \neq 0$ angelegt, wodurch der Pfadfortschritt verlangsamt beziehungsweise gestoppt wird, wie man an der gepunkteten und strichlierten Kurve sehen kann. Nachdem für den Eingang wieder $u = 0$ gilt, nähert sich der Pfadfortschritt erneut mit einem glatten Übergang dem

geplanten Polynom an. Außerdem ist ein schnelleres Abbremsen als Beschleunigen des Pfadparameters bei der strichlierten Kurve sichtbar.

In diesem Kapitel wurde der gewünschte Pfad des Endeffektors berechnet. Dazu wurden Pfade definiert, die der Endeffektor auf einer ebenen oder zylindrischen Tafel zeichnen soll. Damit den Pfaden auf einer bewegten Tafel gefolgt werden kann, muss die Pose der Tafel gemessen werden. Die Messung muss durch Transformationen und Filter zusätzlich verarbeitet werden. Kombiniert man den Pfad auf der Tafel mit der Messung der Pose der Tafel, ist eine Vorgabe für die Pose des Endeffektors gefunden. Mit dem mathematischen Modell und dem Regelgesetz der vorherigen Kapitel kann der Roboter der gewünschten Vorgabe folgen und auf eine bewegte Tafel zeichnen. Zuletzt wurde der Pfadfortschritt abhängig von der Bewegung der Tafel berechnet, um bei schnellen Bewegungen der Tafel den Pfadfortschritt anzuhalten.

5 Ergebnisse

Um die erlangten Erkenntnisse der vorigen Kapitel anzuwenden, wird ein Laborversuch aufgebaut. Dieser soll zeigen, wie der KUKA *LBR 4+* auf eine bewegte Tafel zeichnet. In diesem Kapitel werden Messungen zu verschiedenen Versuchen mit dem Laboraufbau gezeigt und diskutiert.

Um den experimentellen Versuch durchzuführen, wird das Regelgesetz aus Kapitel 3 in MATLAB/SIMULINK implementiert. Weiters ist eine Messung der Pose der Tafel durch das optische Kamerasystem notwendig. Der Roboter wird über eine KUKA Steuerung angesteuert, der das vom Regelgesetz berechnete Drehmoment übergeben wird. Die momentanen Gelenkwinkel und Gelenkwinkelgeschwindigkeiten werden durch die KUKA Steuerung zur Verfügung gestellt. Alle Berechnungen werden in Echtzeit auf einem *Core i7-2600* Prozessor mit *16GB* Arbeitsspeicher durchgeführt, wobei *TwinCat3* als Laufzeitumgebung in *Visual Studio* dient.

Im ersten Abschnitt 5.1 wird ein Versuch gezeigt, bei der die Tafel durch eine Linearachse bewegt wird. Der Endeffektor des Roboters folgt dieser Bewegung und zeichnet eine Lissajous-Kurve auf die Tafel. In Abschnitt 5.2 werden Testfälle diskutiert, bei der die Tafel im Raum bewegt wird. Es werden drei Versuche mit unterschiedlichen Pfaden auf der Tafel vorgestellt. Im letzten Abschnitt 5.3 werden die so gezeichneten Ergebnisse auf der Tafel gezeigt.

5.1 Linearbewegung der Tafel

In diesem Abschnitt wird ein Testfall beschrieben, bei dem eine ebene Tafel durch eine Linearachse in eine Raumrichtung bewegt wird. Der Endeffektor des Roboters zeichnet eine Lissajous-Kurve auf die bewegte Tafel. Es wird das Modell aus (2.30) und das Regelgesetz aus (3.22) mit (3.7) und (3.23) verwendet, wobei die beiden zeitlichen Ableitungen der gewünschten Position $\dot{\mathbf{p}}_d, \ddot{\mathbf{p}}_d$ sowie die gewünschte Winkelgeschwindigkeit und deren zeitliche Ableitung $\boldsymbol{\omega}_d, \dot{\boldsymbol{\omega}}_d$ vernachlässigt werden. Zusätzlich wird der integrale Anteil der Vektoranteile des Fehlerquaternions $\int \delta \boldsymbol{\varepsilon} dt$ in $\boldsymbol{\nu}_\omega$ von (3.7), siehe auch (3.13) und (3.18), mit der Verstärkung 0.2 hinzugefügt. Außerdem wird in diesem Versuch die Messgröße für die Position der Tafel vom Encoder der Linearachse ausgelesen und mit einem Savitzky-Golay Filter geglättet. Somit vereinfachen sich die Transformationen aus Kapitel 4. Der Pfadparameter φ wird direkt über die Zeit parametrisiert, wodurch für die Variable $\vartheta = t$ in (4.40) gilt.

In Abbildung 5.1a ist die Trajektorie des Endeffektors \mathbf{p}_e im Basis-Koordinatensystem Σ_0 gezeigt. Der Vektor \mathbf{n} gibt die Orientierung des Endeffektors an. In dieser Abbildung ist ersichtlich, dass die Trajektorie durch die Bewegung der Tafel gegenüber der Lissajous-Kurve im Tafel-Koordinatensystem Σ_l in Abbildung 5.1b verzerrt ist. In Abbildung 5.1b ist die gewünschte und die tatsächliche Position des Endeffektors \mathbf{p}_d und \mathbf{p}_e dargestellt, wobei

eine Abweichung in z_l -Richtung zu beobachten ist. In Abbildung 5.1c ist die Trajektorie ebenfalls im Tafel-Koordinatensystem Σ_l dargestellt, jedoch werden ausschließlich die (x_l, y_l) -Koordinaten gezeigt. In diesen beiden Raumrichtungen stimmt die gewünschte und die tatsächliche Position gut überein. Die stärkere Abweichung in z_l -Richtung in Abbildung 5.1b ist durch die kleinere Proportionalverstärkung \mathbf{K}_p in dieser Raumrichtung gegenüber der Verstärkung für die (x_l, y_l) -Richtungen erklärbar, siehe Tabelle A.1. Die Parameter wurden gewählt, um Regelfehler in z_l -Richtung geringer zu gewichten. In Abbildung 5.1d ist der Pfadfortschritt φ der Lissajous-Kurve über der Zeit t aufgetragen.

Die Positionsfehler und die Orientierungsfehler sind im Basis-Koordinatensystem Σ_0 in Abbildung 5.2 dargestellt. Beim Positionsfehler in z_0 -Richtung ist eine Schwingung von etwa ± 2 mm sichtbar. Da die z_0 -Richtung des Basis-Koordinatensystems Σ_0 bei diesem Versuch der negativen x_l -Richtung im Tafel-Koordinatensystem Σ_l entspricht, entsteht diese Schwingung durch die Auf- und Abbewegung des Endeffektors, um der Lissajous-Kurve zu folgen, siehe Abbildung 2.1 und Abbildung 4.1. Die x_0 -Richtung in Σ_0 entspricht der z_l -Richtung in Σ_l . Für diese Raumrichtung bleibt der Fehler abschnittsweise konstant.

In den Positionsfehlern e_y und e_z sind kleine überlagerte Schwingungen mit abklingendem Verhalten zu beobachten. Für die Orientierungsfehler fällt eine stärkere überlagerte Schwingung auf, obwohl die gewünschte Orientierung in diesem Versuch konstant gewählt ist. Das Rauschen entsteht durch die rauschbehaftete Messung von $\dot{\mathbf{q}}$.

In Abbildung 5.3 sind die Verläufe aller sieben Gelenkwinkel und in Abbildung 5.4 die dazugehörigen Drehmomentverläufe dargestellt. Darin erkennt man für q_1 im Zeitabschnitt 38 s bis 42 s eine langsame Änderung des Winkels. Das Drehmoment τ_1 ist daher gering. Bei langsamen Änderungen des Winkels q_1 wird die Bewegung für den Fortlauf der Kurve in y_l -Richtung hauptsächlich durch die Linearachse ausgeführt. Ist jedoch die Bewegung der Linearachse gleich zu der Bewegung der Lissajous-Kurve in y_l -Richtung, muss der Roboter den Winkel q_1 schnell ändern. Dies ist im Zeitraum um 46 s sichtbar.

Die Rotationsachsen der Gelenke 2, 4 und 6 liegen in Kerzenstellung horizontal, vergleiche Abbildung 2.1. In den Winkelverläufen dieser Gelenke sieht man eine Bewegung gleicher Frequenz. Diese entstehen damit die Orientierung des Endeffektors bei Bewegung des Endeffektors in x_l -Richtung gleich bleibt, siehe Abbildung 4.8. Die Drehmomente aller drei Gelenke folgen für diese Bewegung näherungsweise einer Rechteckfunktion mit gleicher Frequenz.

Der Verlauf von q_5 zeigt qualitativ den negativen Verlauf von q_7 . Die Rotationsachsen der Gelenke 5 und 7 liegen wie 1 und 3 in Kerzenstellung senkrecht, vergleiche Abbildung 2.1. Daher müssen die Gelenke 5 und 7 gegengleiche Bewegungen ausführen, damit die Orientierung des Endeffektors bei Bewegung der Tafel in y_l -Richtung gleich bleibt, siehe Abbildung 4.8. Weiters ist anzumerken, dass für diesen Versuch der Verlauf von q_3 klein gegenüber denen von q_5 und q_7 ist.

Das Rauschen im Drehmomentverlauf entsteht durch das verrauschte Messsignal des Encoders der Linearachse und das verrauschte Messsignal von $\dot{\mathbf{q}}$. Dadurch entsteht ein verrauschter Regelfehler, wie in Abbildung 5.2 sichtbar ist. Das Regelgesetz berechnet daraus ein verrauschtes Drehmoment, das an die Robotersteuerung übergeben wird, siehe Abbildung 5.4.

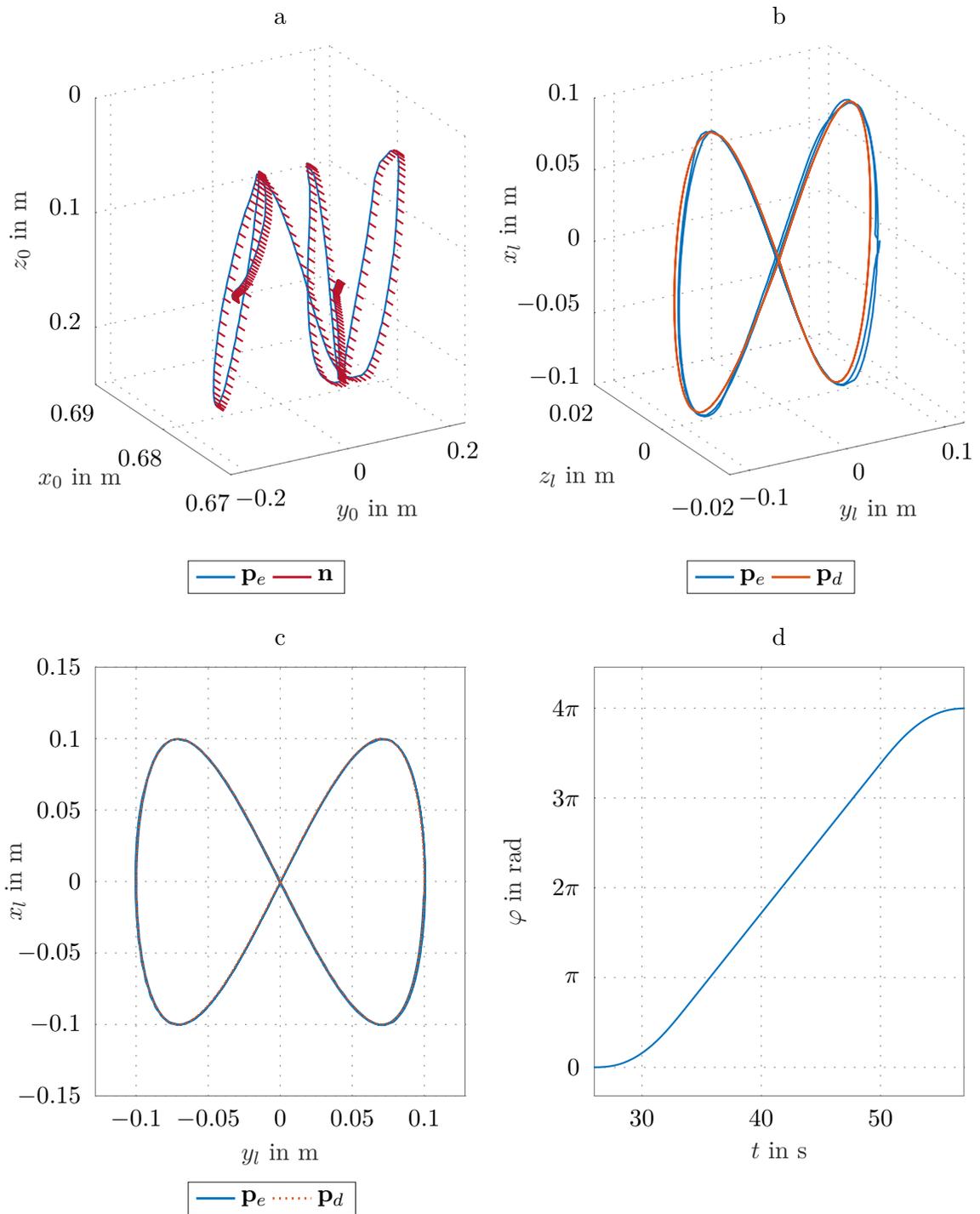


Abbildung 5.1: Trajektorien bei Linearbewegung der Tafel.

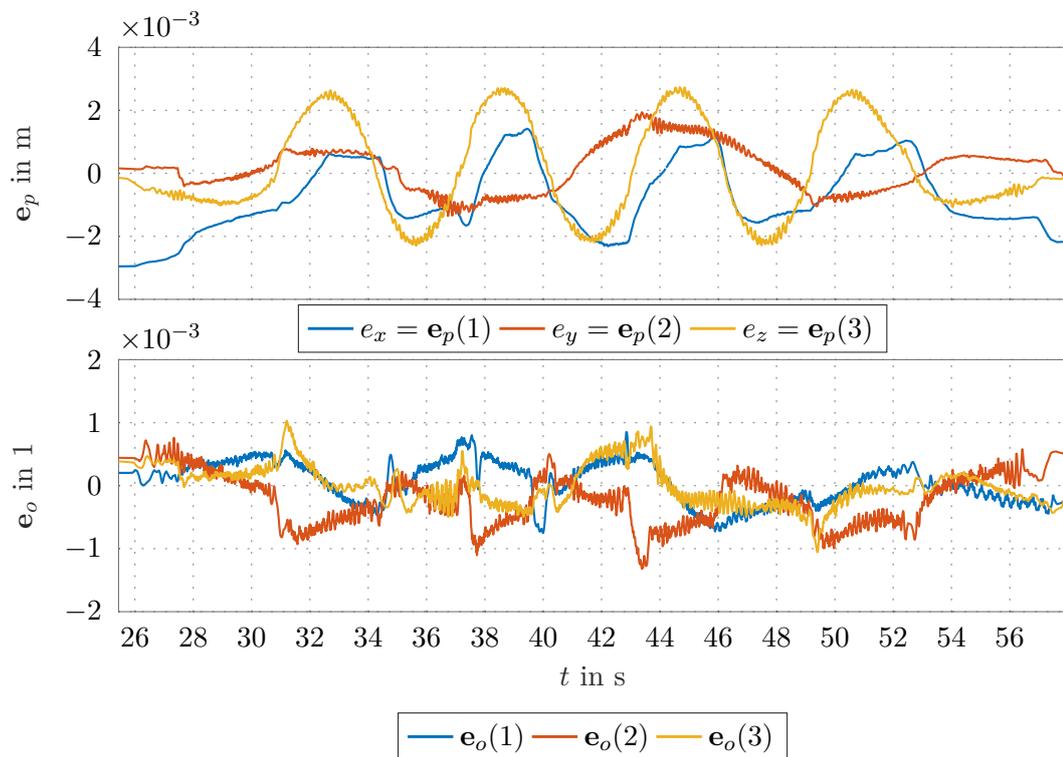


Abbildung 5.2: Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 bei Linearbewegung der Tafel.

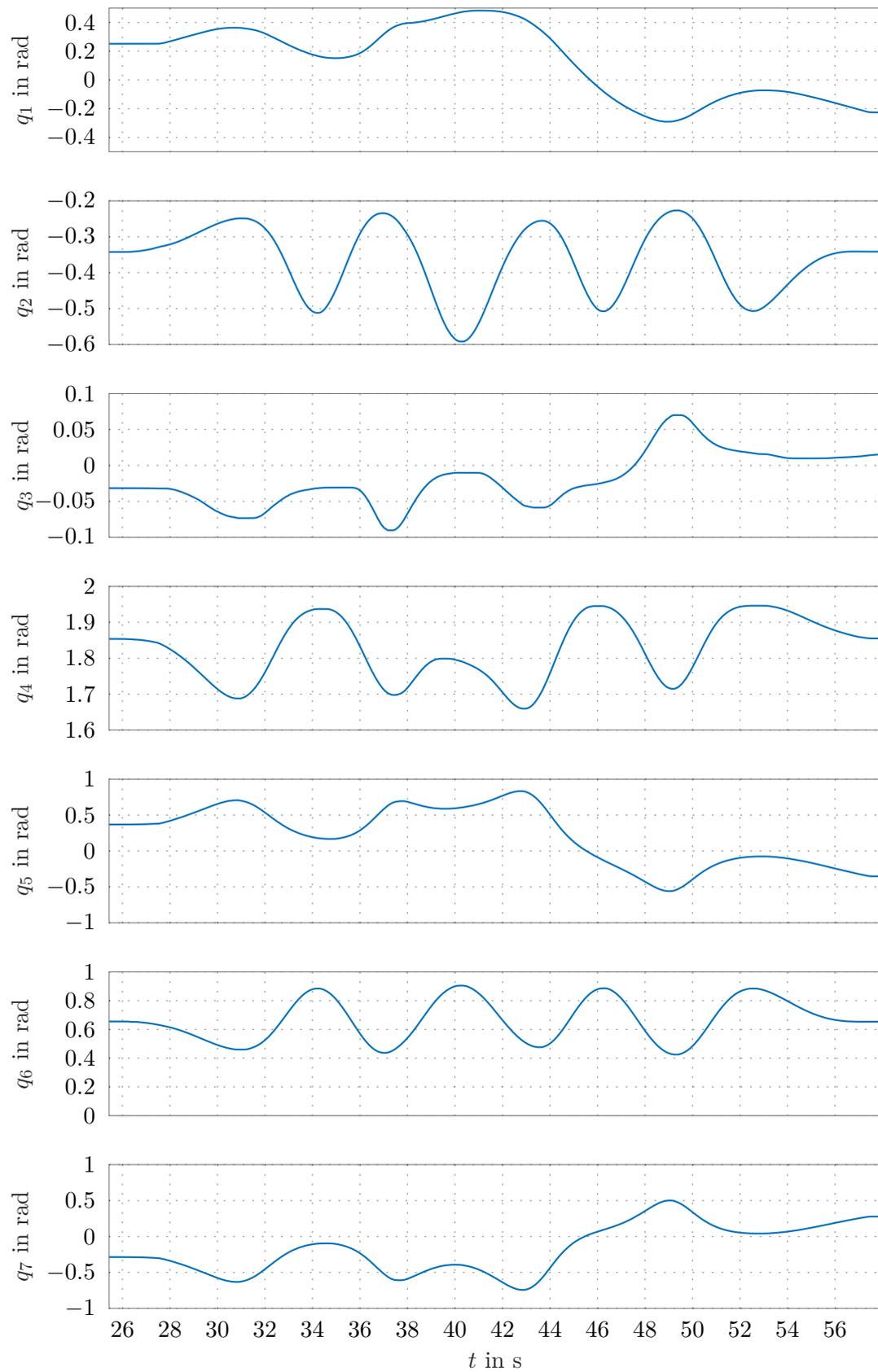


Abbildung 5.3: Gelenkwinkel bei Linearbewegung der Tafel.

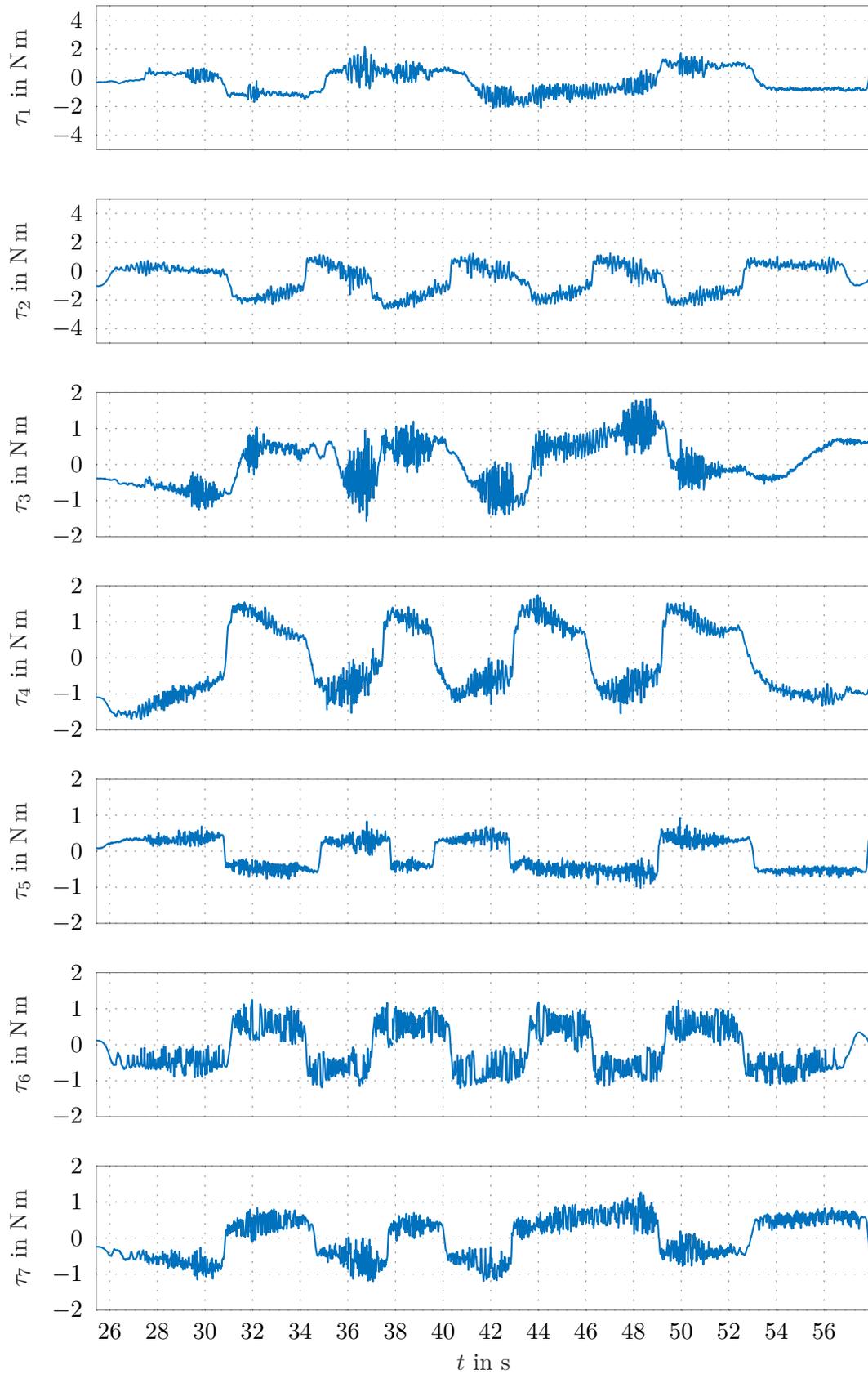


Abbildung 5.4: Drehmomente bei Linearbewegung der Tafel.

5.2 Bewegung der Tafel im Raum

Ein Ziel dieser Arbeit besteht darin, dass der Roboter einen Pfad auf einer bewegten Tafel zeichnet. Die Tafel soll dabei beliebig im Arbeitsraum des Roboters von einem Menschen bewegt werden. Dabei werden drei verschiedene Pfade auf der bewegten Tafel gezeichnet. Die Messungen zu diesen Versuchen werden in diesem Abschnitt zusammengefasst und diskutiert.

Das mathematische Modell (2.30) wird im Regelgesetz (3.22) mit (3.7) und (3.23) verwendet. Bei diesen Versuchen wird die zeitliche Ableitung der gewünschten Position $\dot{\mathbf{p}}_d$ und die gewünschte Winkelgeschwindigkeit $\boldsymbol{\omega}_d$ in (3.7) verwendet. Die Berechnung dieser Größen sowie die Pfadplanung ist in Kapitel 4 gezeigt. Die Terme der Vorsteuerung $\ddot{\mathbf{p}}_d$ und $\dot{\boldsymbol{\omega}}_d$ werden vernachlässigt, da Messrauschen durch die zweifache zeitliche Ableitung zu starkem Rauschen führt. Dieses Rauschen würde sich auf die berechneten Drehmomente $\boldsymbol{\tau}_d$ auswirken. Ein integraler Anteil des Vektoranteils des Fehlerquaternions $\int \delta \boldsymbol{\varepsilon} dt$, siehe auch (3.13) und (3.18), wird mit einer Verstärkung von 0.02 in $\boldsymbol{\nu}_\omega$ von (3.7) hinzugefügt. Zusätzlich wird in weiterer Folge immer der dynamische Pfadfortschritt aus Abschnitt 4.4 genutzt.

Um den Endeffektor des Roboters mit der Bewegung der Tafel zu synchronisieren, wird der Roboter zuerst mittels eines Reglers der Gelenkwinkel aus der Kerzenposition gefahren. Anschließend wird auf das Regelgesetz aus Kapitel 3 umgeschaltet, wobei die Reglerparameter \mathbf{K}_p und \mathbf{K}_o linear mit der Zeit von $\mathbf{0}$ auf den Zwischenwert $\mathbf{K}_{p,l}$ und $\mathbf{K}_{o,l}$ einlaufen. Sobald der Endeffektor eine vorgegebene Positionsabweichung zur Startposition des gewünschten Pfades unterschreitet, werden die Reglerparameter \mathbf{K}_p und \mathbf{K}_o linear mit der Zeit auf den Endwert vergrößert. Die Werte von $\mathbf{K}_{p,l}$ und $\mathbf{K}_{o,l}$ sowie die Zeitdauer für den Übergang sind in Tabelle A.2 aufgelistet.

Beim ersten Versuch zeichnet der Endeffektor eine Lissajous-Kurve auf eine ebene bewegte Tafel. Die Ergebnisse der Messung sind in Abschnitt 5.2.1 gezeigt. Um eine *pick and place* Aufgabe zu simulieren, soll eine unterbrochene Kurve auf die Tafel gezeichnet werden. Die Messergebnisse dieses Versuchs werden in Abschnitt 5.2.2 diskutiert. Im letzten gezeigten Versuch in Abschnitt 5.2.3 zeichnet der Stift des Roboters eine Lissajous-Kurve auf eine bewegte zylindrische Tafel.

5.2.1 Lissajous-Kurve auf ebener Tafel

Ähnlich zu Abschnitt 5.1 folgt in diesem Versuch der Endeffektor des Roboters einer Lissajous-Kurve auf einer ebenen Tafel. Die Tafel wird in diesem Testfall jedoch frei im Arbeitsraum des Roboters bewegt.

In Abbildung 5.5a ist der Pfad des Endeffektors inklusive der Orientierung im Basis-Koordinatensystem des Roboters Σ_0 gezeigt. Es ist eine Bewegung des Pfades in y_0 -Richtung von Σ_0 zu sehen, vergleiche Abbildung 5.1a. Die Tafel wird zu Beginn ruhig gehalten, wodurch die Form der Lissajous-Kurve sichtbar ist. Danach wird die Tafel mehrere Male bewegt. In Abbildung 5.5b wird die Lissajous-Kurve im Tafel-Koordinatensystem Σ_l dargestellt. Es ist trotz der freien Bewegung der Tafel eine gute Übereinstimmung des tatsächlichen mit dem gewünschten Pfad auf der Tafel sichtbar, was auch in Abbildung 5.5c zu sehen ist. Der Pfadparameter φ ist in Abbildung 5.5d über der Zeit t aufgetragen.

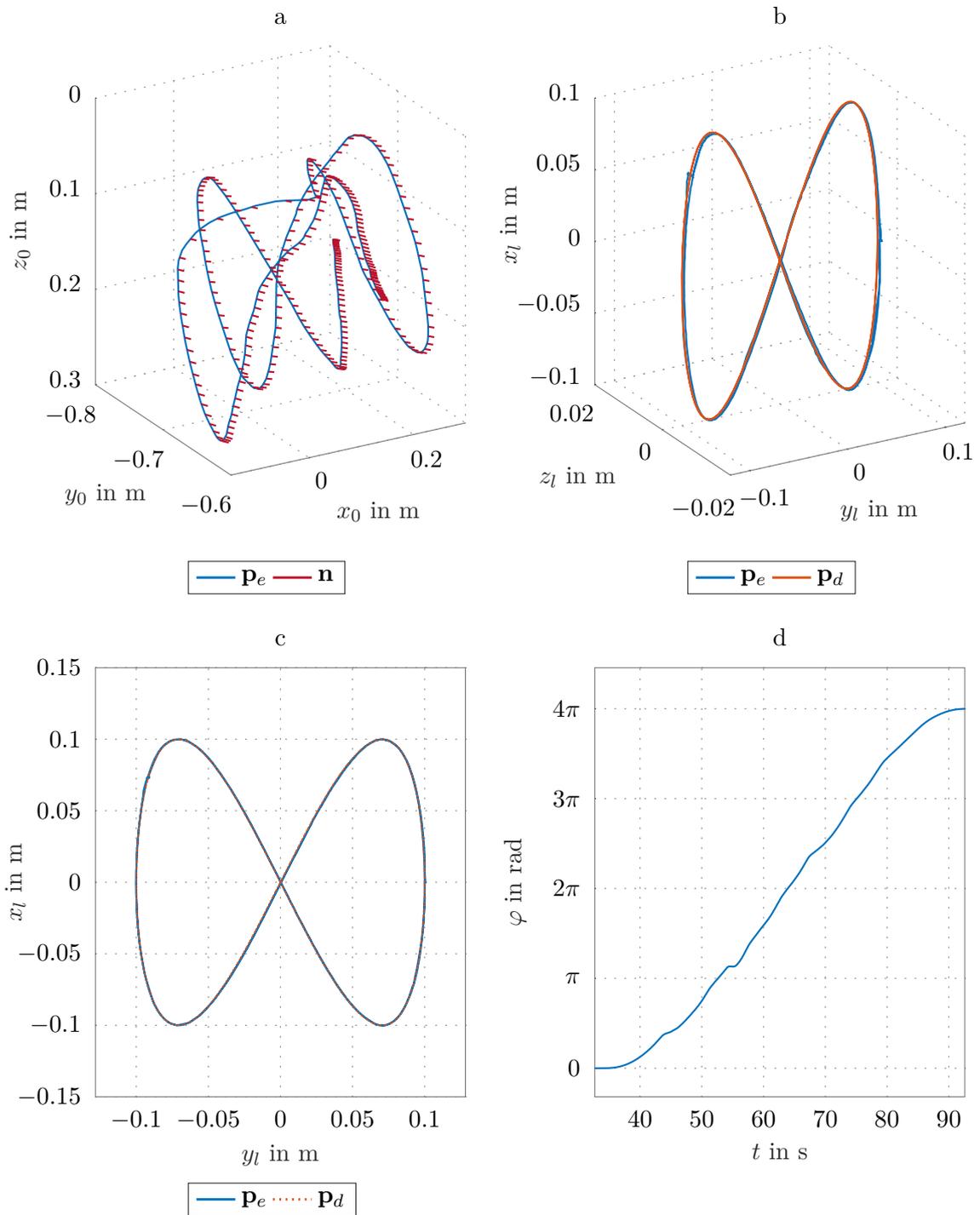


Abbildung 5.5: Pfade beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.

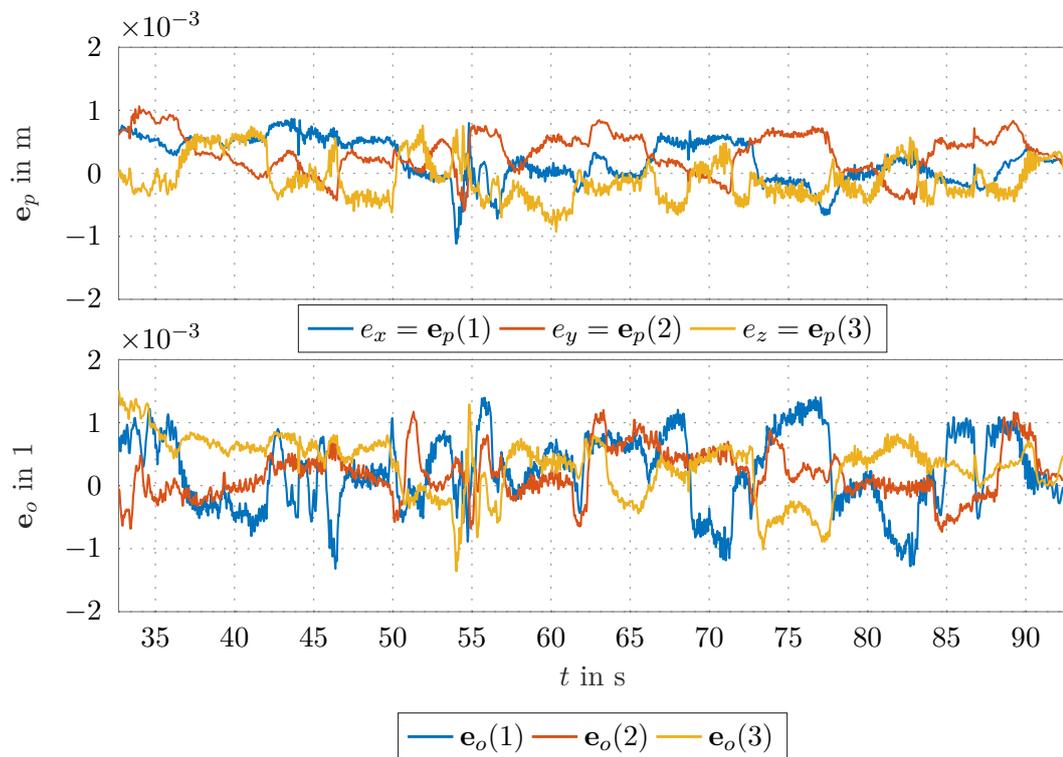


Abbildung 5.6: Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.

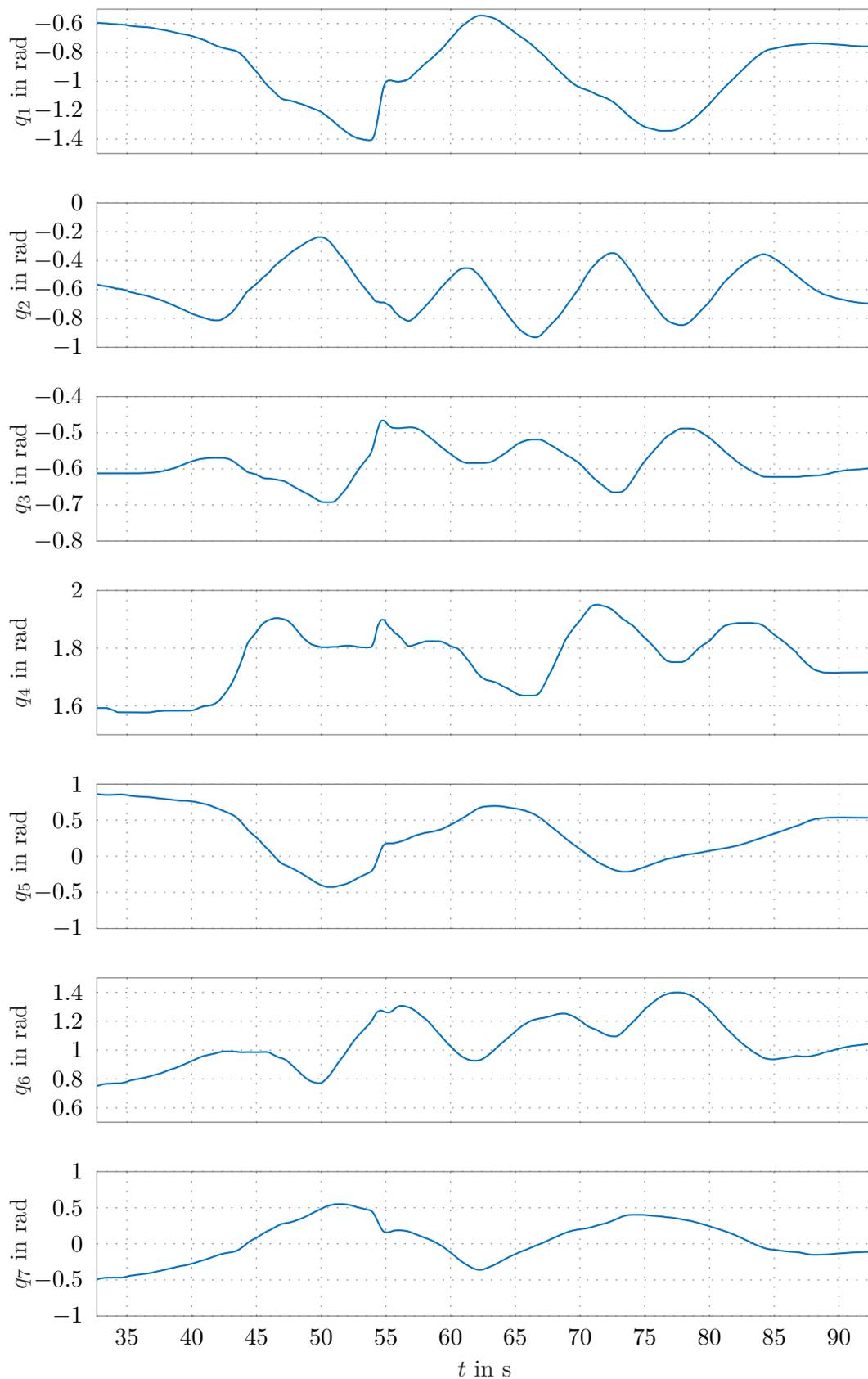


Abbildung 5.7: Gelenkwinkel beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.

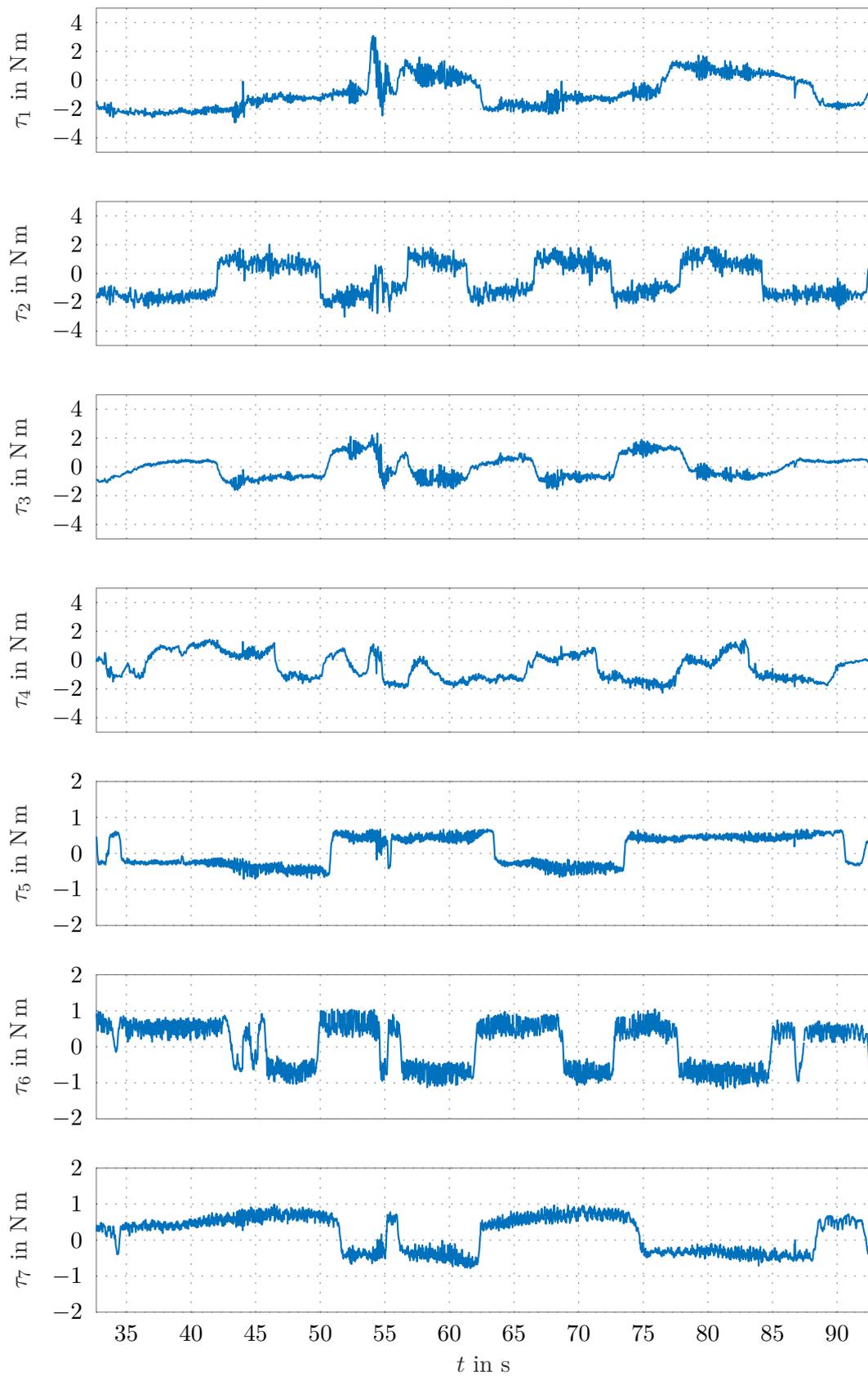


Abbildung 5.8: Drehmomente beim Zeichnen einer Lissajous-Kurve auf einer ebenen Tafel.

Der Verlauf des Pfadparameters Abbildung 5.5d unterscheidet sich deutlich von dem Verlauf in Abbildung 5.1d, da der dynamische Pfadfortschritt gemäß Abschnitt 4.4 verwendet wird. In Abbildung 5.5d ist sichtbar, dass bei geringer Bewegung der Tafel der Pfadfortschritt verlangsamt wird. Bei starker Bewegung der Tafel stoppt der Fortschritt am Pfad. Ein Stoppen des Pfadfortschritts ist nach etwa 54s zu sehen. Ebenfalls ist das schnellere Anhalten als Fortsetzen des Pfadfortschritts sichtbar.

Die Regelfehler der Position und Orientierung sind in Abbildung 5.6 dargestellt. Diese Fehler hängen stark von der Bewegung der Tafel ab. Es ist eine schnelle Veränderung der Verläufe der Regelfehler bei 54s zu sehen. Dieser Verlauf entsteht durch eine schnelle Bewegung der Tafel, wodurch der Pfadfortschritt angehalten wird, vergleiche Abbildung 5.5d. Aus systemtheoretischer Sicht kann die Bewegung der Tafel als auszuregelnde Störgröße betrachtet werden. Außerdem fällt im Vergleich zu Abbildung 5.2 eine kleinere Rauschamplitude beim Orientierungsfehler auf.

In Abbildung 5.7 und Abbildung 5.8 sind die Winkel- und Drehmomentverläufe aller Gelenke dargestellt. Wie im vorherigen Abschnitt fällt die gegengleiche Bewegung der Gelenke 2 und 6 auf. Deutlich sichtbar ist die schnelle Bewegung der Tafel bei 54s, der der Roboter folgen muss. Bei allen Gelenken wird eine schnelle Veränderung der Gelenkwinkel gemessen. Diese schnelle Änderung wird durch ein großes Drehmoment ermöglicht, das in Abbildung 5.8 im Speziellen bei Gelenk 1 sichtbar wird. Das Rauschen der Drehmomente hat keine sichtbare Auswirkung auf die Roboterbewegung, was auf die Trägheit des Aufbaus zurückzuführen ist. Werden die Reglerparameter \mathbf{K}_v aus Tabelle A.1 größer gewählt, ist ein Vibrieren in allen Gelenken mess- und hörbar. Durch die erhöhten Reglerparameter \mathbf{K}_v würde der Roboter schneller auf Sollwertsprünge reagieren.

Eine weitere Möglichkeit, um die Dynamik des Roboters bei schnellen Änderungen der Tafel zu erhöhen, ist die Anpassung des Filters aus Abschnitt 4.2. Werden die Eigenwerte der Filter größer gewählt, würde der Ausgang der Filter schneller auf die tatsächliche Messgröße einschwingen. Vor allem bei der schnellen Änderung der Orientierung würden somit viele Gelenke gleichzeitig schnell bewegt werden. Dieses Verhalten kann für den Anwender, der die Tafel bewegt, unvorhersehbar beziehungsweise unangenehm sein. Die Parameter aus Tabelle A.1 stellen ein Mittelmaß aus Schnelligkeit und Benutzerfreundlichkeit dar.

5.2.2 Unterbrochene Kreiskurve auf ebener Tafel

In diesem Abschnitt wird ein Versuch gezeigt, bei dem der Stift am Roboter eine unterbrochene Kreiskurve auf die ebene Tafel zeichnet. Zeitgleich wird die Tafel im Arbeitsraum des Roboters frei bewegt.

Der Pfad des Endeffektors ist in Abbildung 5.9a im Basis-Koordinatensystem des Roboters Σ_0 gezeigt. Darin ist sichtbar, dass der Endeffektor große Bewegungen in x_0 -Richtung ausgeführt hat. Die Bewegung des Roboters wird durch Bewegungen der Tafel ausgelöst. Wie man auch in Abbildung 5.9b sieht, gibt es zusätzlich kurze Bewegungen des Endeffektors in z_l -Richtung. Während dieser Bewegungen ändert sich die Orientierung des Endeffektors nicht. An diesen Stellen hebt der Endeffektor von der Oberfläche der Ebene ab. Dem gewünschten Pfad wird in allen Raumrichtungen gut gefolgt, was in Abbildung 5.9c nochmals speziell für die Richtungen (x_l, y_l) im Tafel-Koordinatensystem Σ_l dargestellt ist.

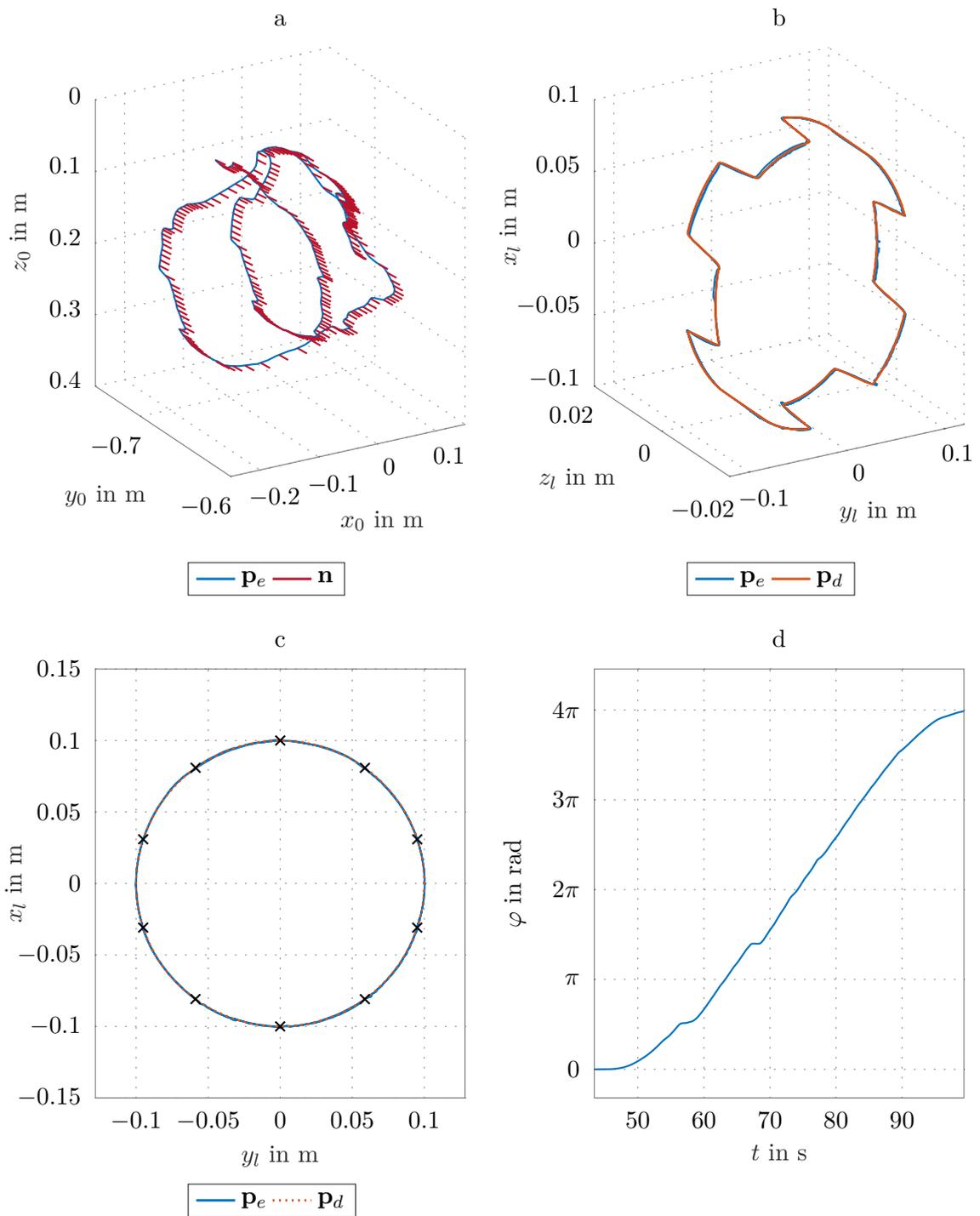


Abbildung 5.9: Pfade beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.

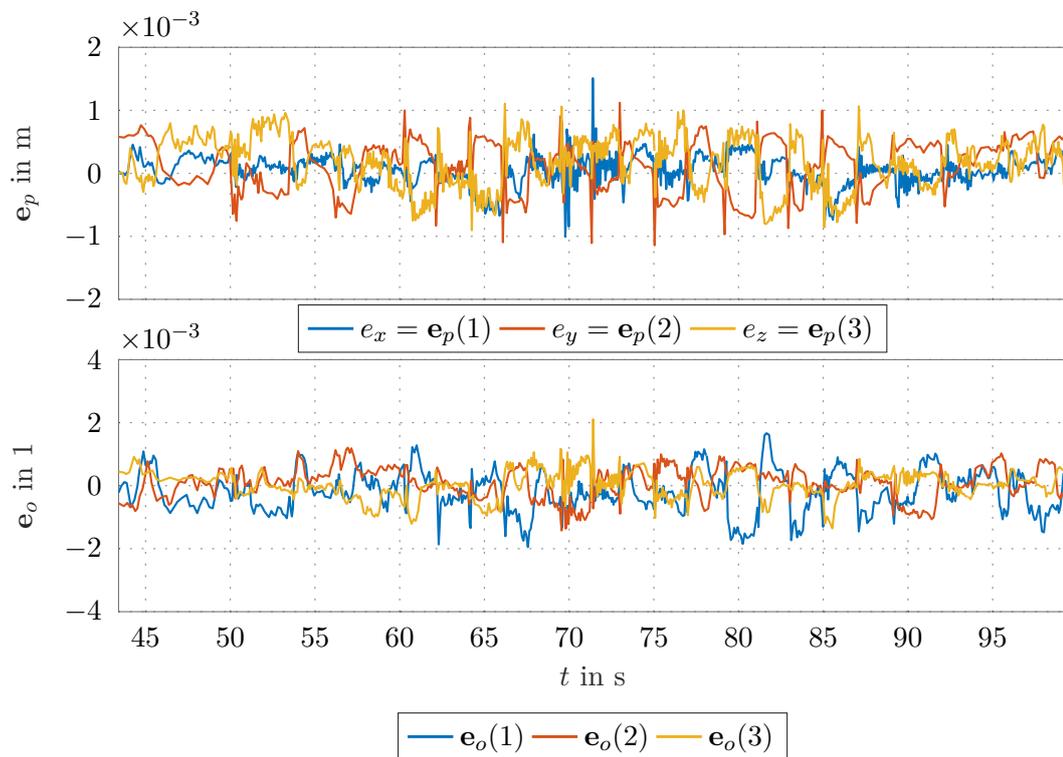


Abbildung 5.10: Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.

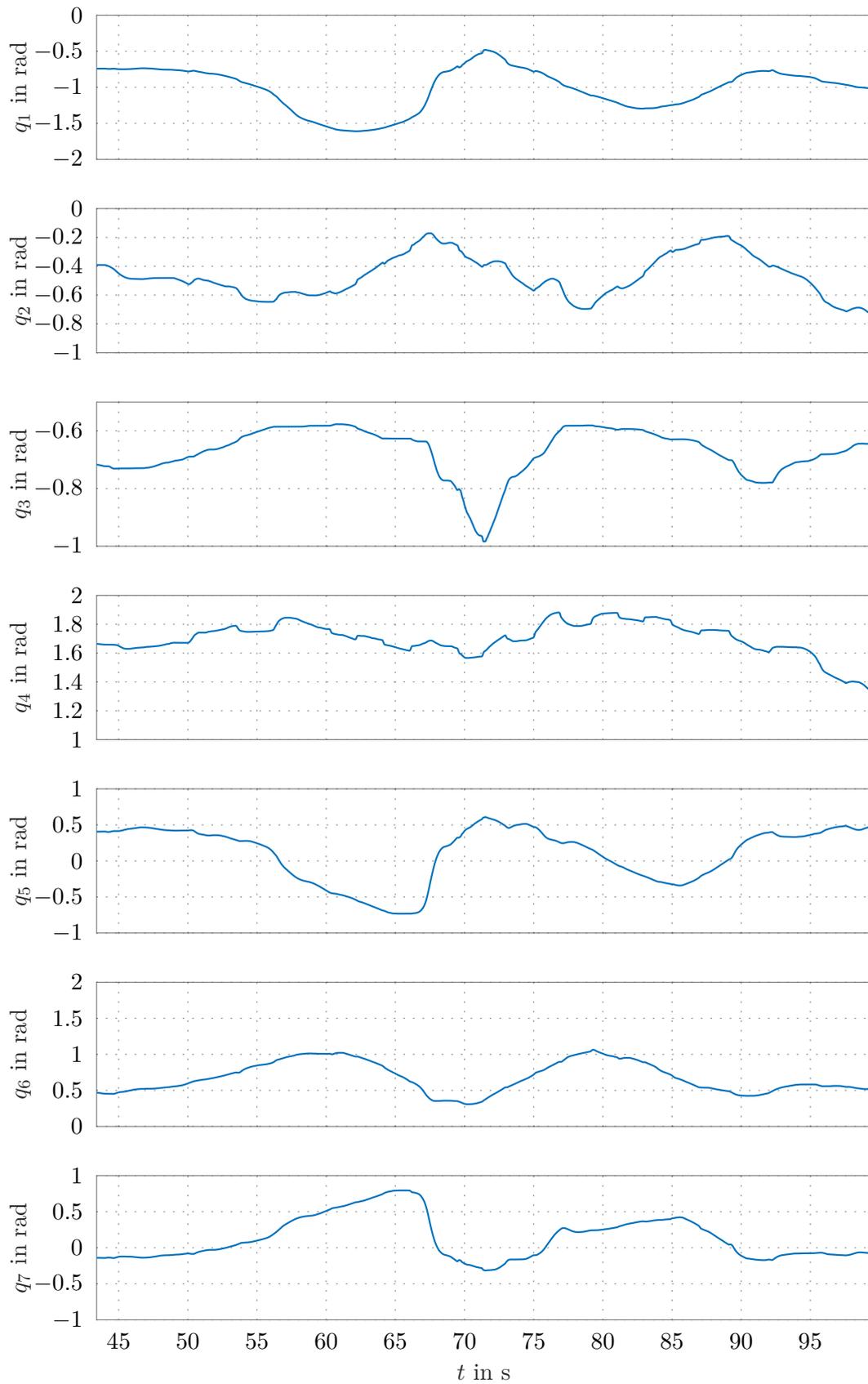


Abbildung 5.11: Gelenkwinkel beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.

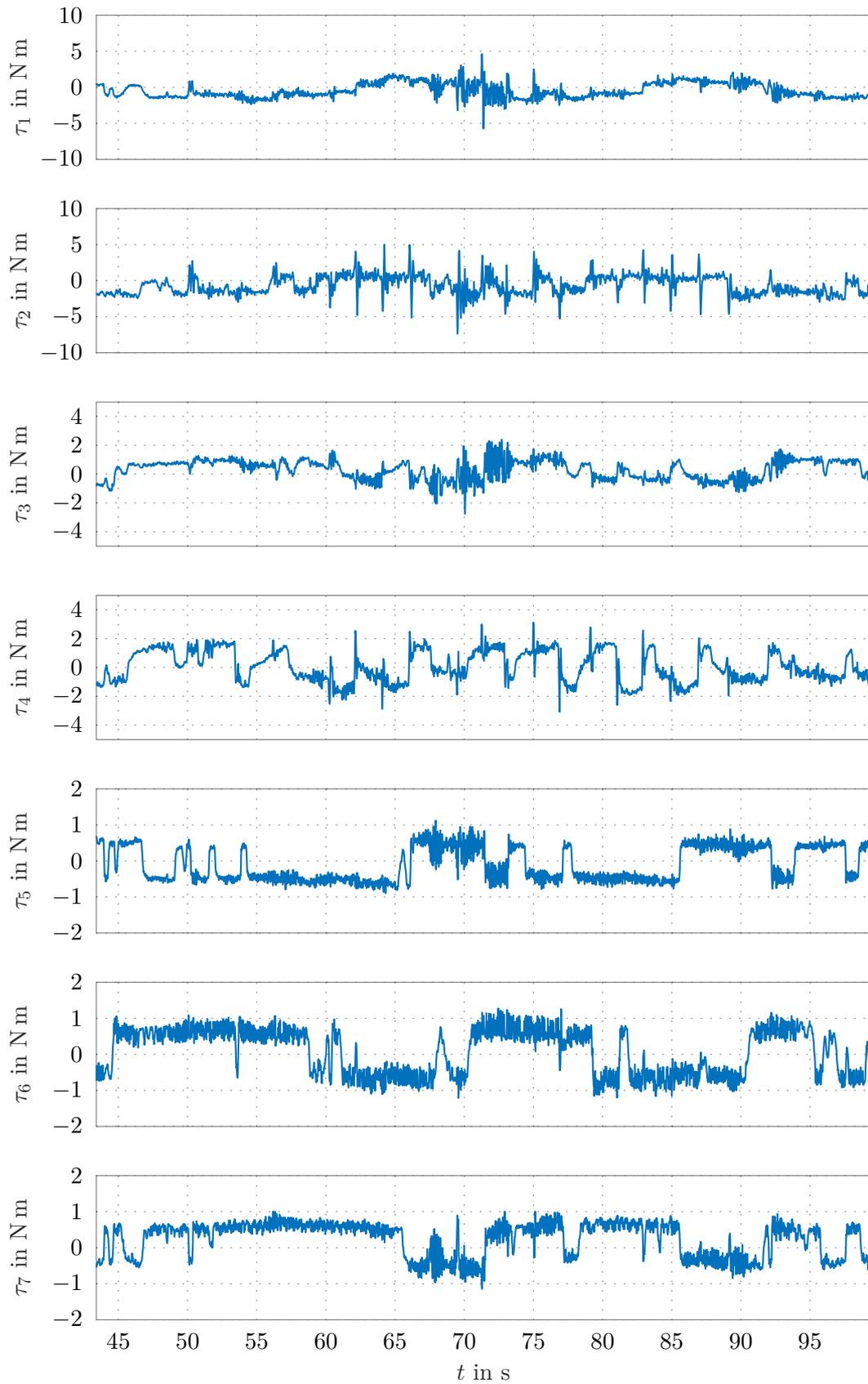


Abbildung 5.12: Drehmomente beim Zeichnen einer unterbrochenen Kreiskurve auf einer ebenen Tafel.

Die x-Markierungen in Abbildung 5.9c kennzeichnen die Punkte, an denen der Endeffektor von der Oberfläche der Tafel abhebt oder sich annähert. Im Verlauf des Pfadparameters φ in Abbildung 5.9d ist sichtbar, dass der Pfadfortschritt durch die schnellen Bewegungen der Tafel angehalten wird.

In Abbildung 5.10 sind die Positions- und Orientierungsfehler dargestellt, die stark von der Bewegung der Tafel abhängen. Es ist ein Positionsfehler in der Größenordnung von etwa ± 1 mm sichtbar. Auffällig sind die starken Spitzen des Positionsfehlers, die durch die schnelle Änderung des gewünschten Pfades zum Abheben des Endeffektors von der Oberfläche entstehen. Diese Fehler werden schnell ausgeregelt. Im Orientierungsfehler sind diese Spitzen kaum vorhanden, da die Orientierung des Endeffektors während des Abhebens gleich bleibt.

Die Verläufe aller Gelenkwinkel sind in Abbildung 5.11 dargestellt. Die Gelenke 2 und 6 zeigen einen gegengleichen Verlauf um die Orientierung bei Bewegung in der x_l -Achse im Tafel-Koordinatensystem Σ_l beizubehalten. Bei etwa 67 s ist eine schnelle Änderung der Gelenke 1, 3, 5 und 7 zu sehen, da die Tafel zu diesem Zeitpunkt schnell bewegt wird, vergleiche den Pfadfortschritt in Abbildung 5.9d. Das schnelle Abheben von der Oberfläche wird hauptsächlich von den Gelenken 2 und 4 ausgeführt. Die dazugehörigen Drehmomentverläufe aus Abbildung 5.12 zeigen in den Gelenken 2 und 4 starke Spitzen, die bis zu 5 N m annehmen können.

5.2.3 Lissajous-Kurve auf zylindrischer Tafel

Im letzten gezeigten Versuch zeichnet der Stift am Roboter eine Lissajous-Kurve auf eine zylindrische Tafel. Die Tafel wird dabei frei im Arbeitsraum des Roboters bewegt. Um die Lissajous-Kurve auf die gekrümmte Tafel zu zeichnen, wird die Kurve auf die zylindrische Fläche projiziert.

In Abbildung 5.13a ist der Pfad des Endeffektors im Basis-Koordinatensystem Σ_0 des Roboters gezeigt. Es ist die Lissajous-Kurve erkennbar, obwohl die Tafel während des Versuchs bewegt wird. Die Orientierung des Endeffektors ändert sich abhängig von der Position am Pfad auf der Tafel. In Abbildung 5.13b ist die projizierte Lissajous-Kurve im Tafel-Koordinatensystem Σ_l dargestellt. Der Endeffektor des Roboters folgt dem gewünschten Pfad gut. Dies ist ebenfalls in Abbildung 5.13c erkennbar, wo nur die (x_l, y_l) -Richtung des Pfades dargestellt wird. Durch die langsamen Bewegungen der Tafel bleibt der Pfadfortschritt nahezu unverändert, was in Abbildung 5.13d zu sehen ist.

Die Regelfehler der Position sowie der Orientierung sind in Abbildung 5.14 gezeigt. Die Werte sind größer als in den vorher gezeigten Versuchen. Bei der Projektion der Lissajous-Kurve auf die zylindrische Tafel wird die Orientierung abhängig von der Position am Pfad angepasst. Dadurch steht der Endeffektor immer senkrecht zur Tafel. Somit müssen die Gelenke des Roboters größere Wege zurücklegen, wodurch die Regelfehler nicht so schnell wie in den zuvor gezeigten Versuchen ausgeregelt werden.

In diesem Versuch wird außerdem ein größerer Messfehler der Pose der Tafel als in den vorherigen Versuchen beobachtet. Es wird eine zylindrische Tafel aus Metall verwendet, die eine starke Spiegelung aufweist. Diese Spiegelung beeinflusst die optische Messung. Außerdem werden einzelne Marker des Messobjekts kurzzeitig vom Messsystem nicht erkannt, was sich auf die geneigte Befestigung des Messobjekts zurückführen lässt.

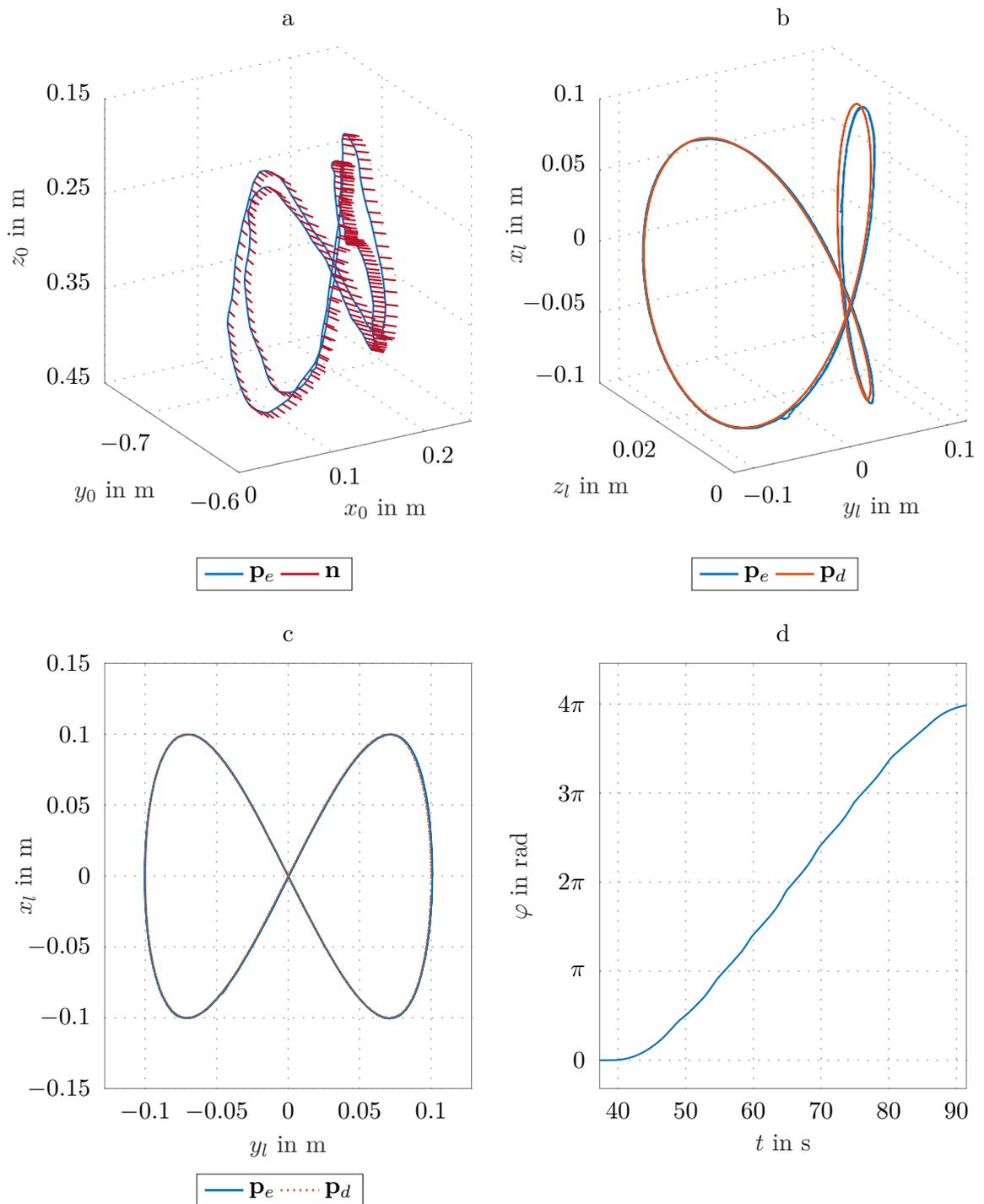


Abbildung 5.13: Pfade beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.

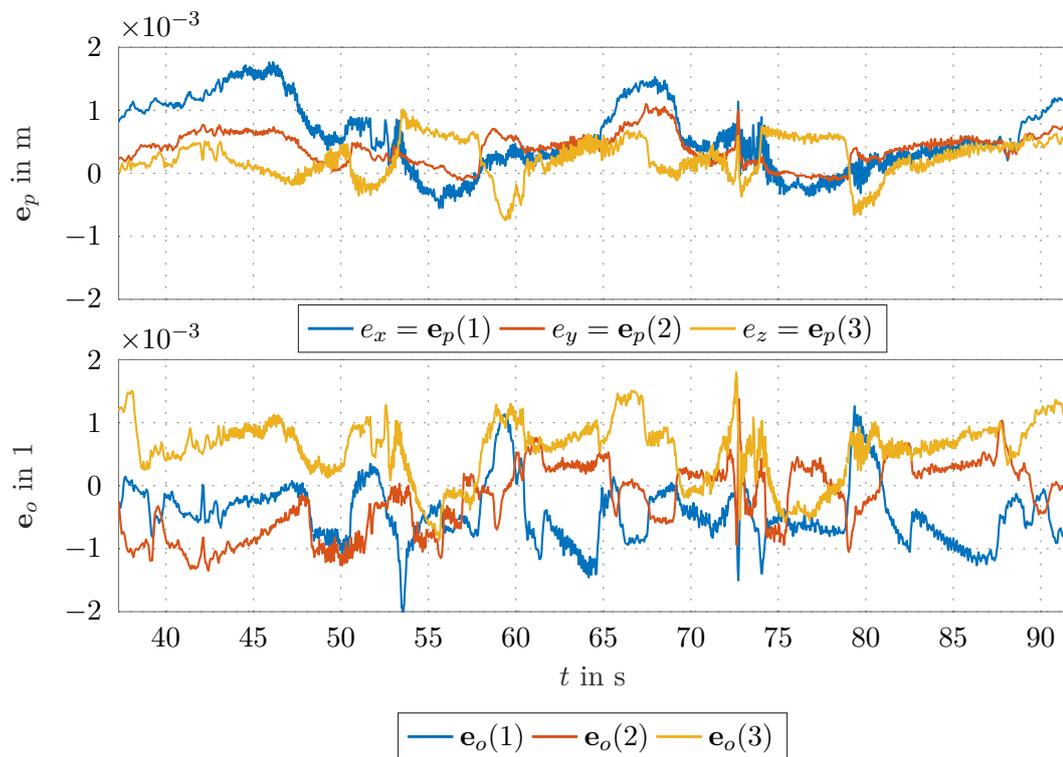


Abbildung 5.14: Positions- und Orientierungsfehler im Basis-Koordinatensystem Σ_0 beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.

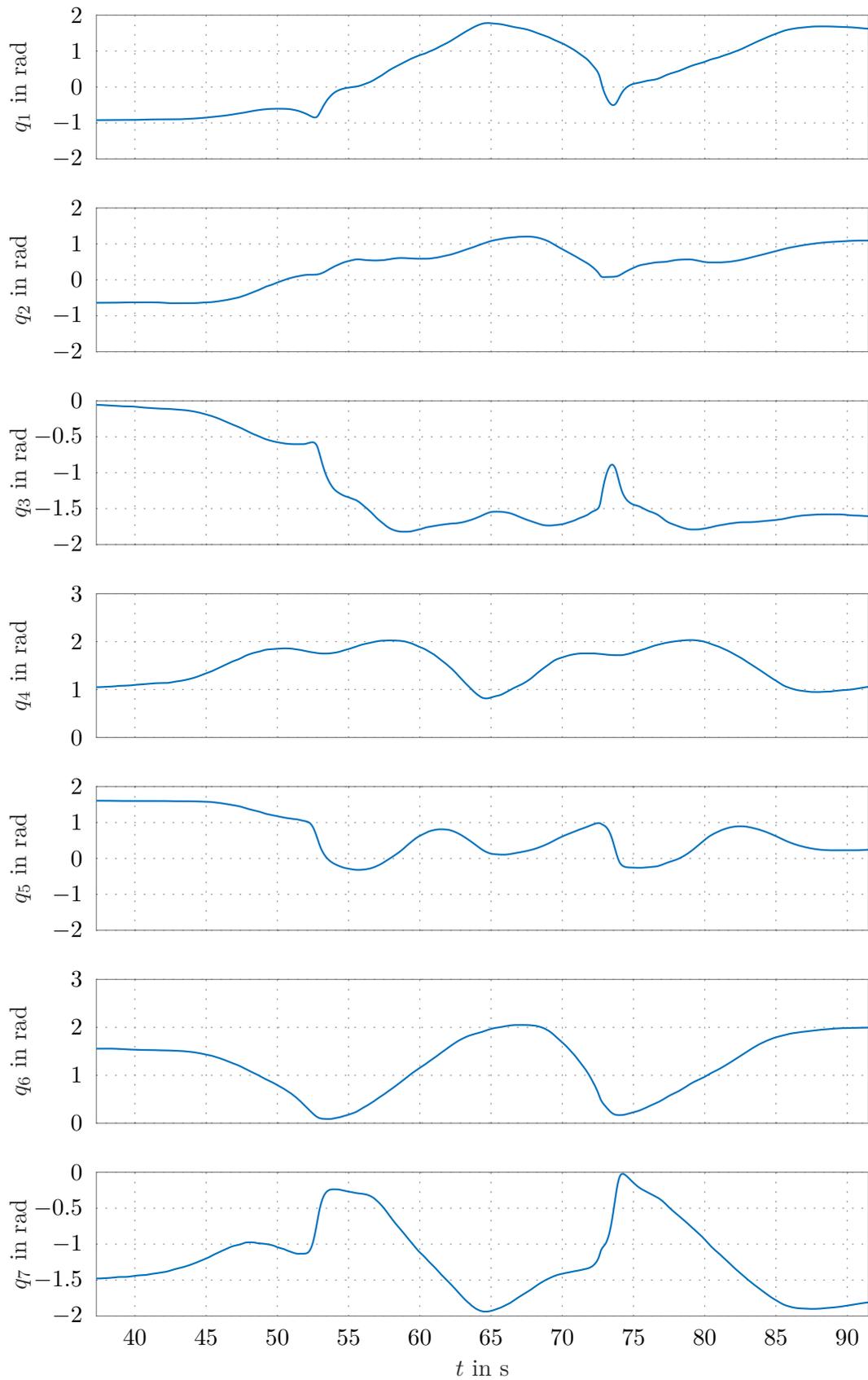


Abbildung 5.15: Gelenkwinkel beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.

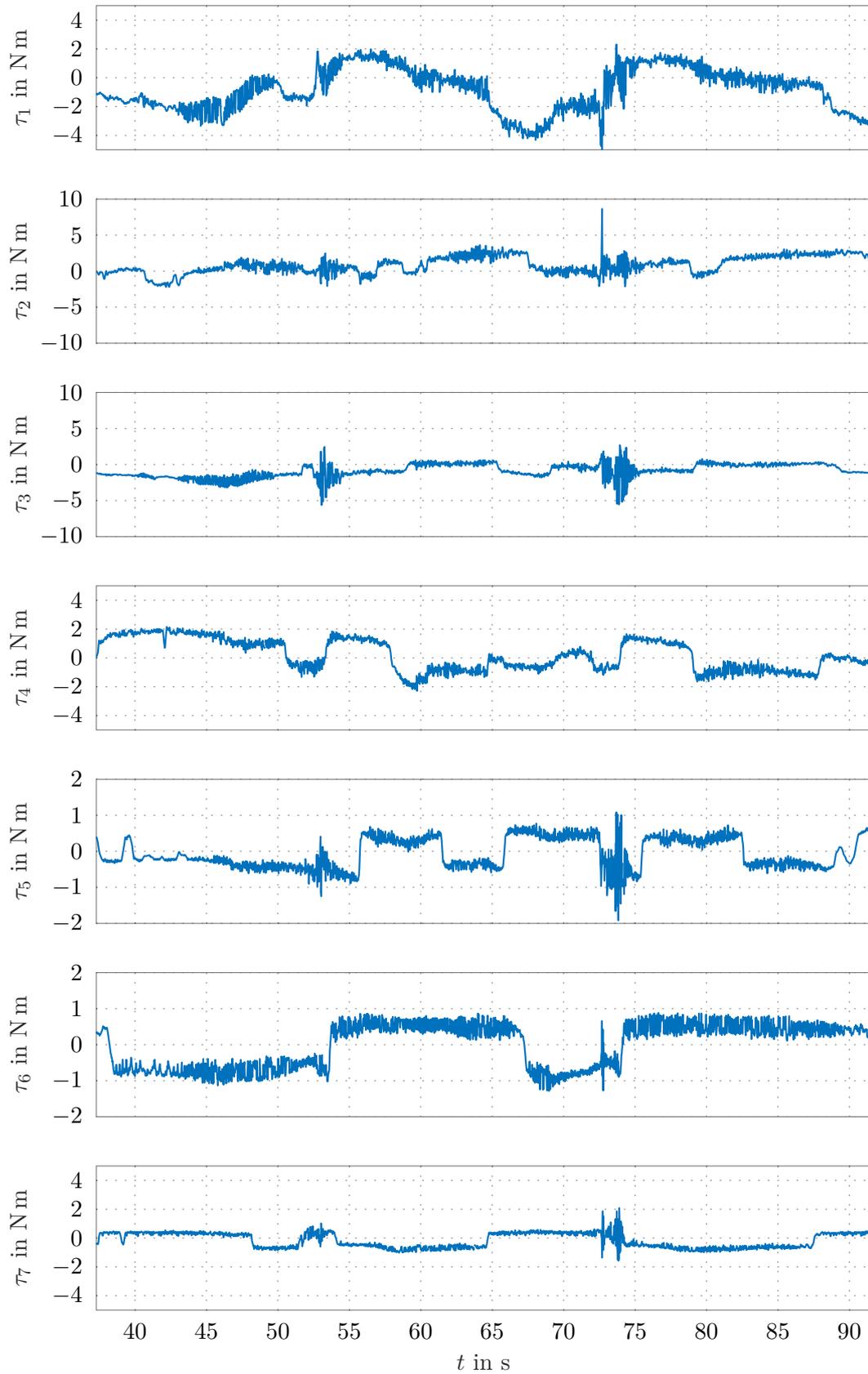


Abbildung 5.16: Drehmomente beim Zeichnen einer Lissajous-Kurve auf einer zylindrischen Tafel.

Die Verläufe der Gelenkwinkel und der Drehmomente sind in Abbildung 5.15 und Abbildung 5.16 gezeigt. Dabei wird deutlich, dass alle Gelenke größere Winkelbereiche überstreichen im Vergleich zu Abbildung 5.7.

5.3 Ergebnisse auf der Tafel

In Abschnitt 5.1 und Abschnitt 5.2 wurden vier Versuche vorgestellt, bei denen der Roboter mit einem Stift auf eine Tafel zeichnet. Die Ergebnisse dieser Zeichnungen sind in Abbildung 5.17 dargestellt und werden in diesem Abschnitt diskutiert.

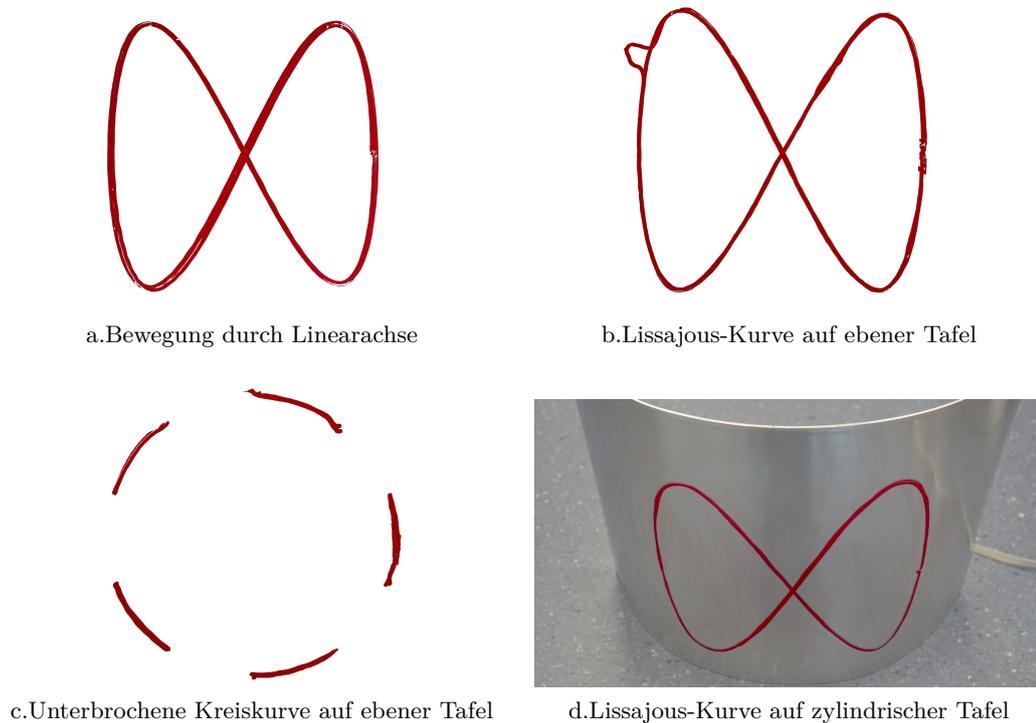


Abbildung 5.17: Gezeichnete Ergebnisse auf der Tafel.

In Abbildung 5.17a ist die gezeichnete Lissajous-Kurve auf einer ebenen Tafel gezeigt. Die Tafel wurde während des Zeichnens durch eine Linearachse bewegt. Im linken unteren Bereich der Kurve ist eine Abweichung vom Pfad sichtbar. Diese Abweichung lässt sich durch die begrenzte absolute Genauigkeit des Roboters erklären. Außerdem wird in diesem Versuch die Positionsmessung der Tafel durch eine relative Messung auf Basis des Encoders der Linearachse durchgeführt.

In Abbildung 5.17b ist das Ergebnis eines Versuchs gezeigt, bei dem die Tafel frei im Raum bewegt wird, während der Endeffektor eine Lissajous-Kurve auf die Tafel zeichnet. Obwohl der Endeffektor der gewünschten Position \mathbf{p}_d genau nachgeregelt wird, siehe Abbildung 5.5b, entsteht eine Abweichung in Abbildung 5.17b vom gewünschten Pfad. Die Abweichung im linken oberen Bereich der Kurve entsteht durch eine schnelle Bewegung

der Tafel. Zur Berechnung der Pose des Endeffektors wird die Pose der Tafel gemessen und gefiltert. Durch die Filterung entsteht eine Abweichung zwischen der gemessenen zur gefilterten Pose der Tafel. Damit folgt der Roboter sehr schnellen Bewegungen der Tafel verlangsamt. Diese Einschränkung wurde aufgrund des kollaborativen Betriebs des Roboters mit einem Menschen gewählt. Durch die hohe Dämpfung des Fehlersystems wird die Bewegung der Robotergelenke ebenfalls vermindert.

Ein systematischer Fehler bei dieser Arbeit ist die absolute Genauigkeit des Roboters, die etwa 1 cm beträgt. Wird am Endeffektor ein zusätzliches Messobjekt angebracht, kann die Position und Orientierung des Endeffektors absolut im Raum gemessen werden. Der Fehler der absoluten Genauigkeit des Roboters könnte somit kompensiert werden. Ein weiterer systematischer Fehler ist durch die absolute Genauigkeit und die Latenzzeit des optischen Kamerasystems selbst gegeben.

In Abbildung 5.17c und Abbildung 5.17d sind die gezeichneten Ergebnisse einer unterbrochenen Kreiskurve auf einer ebenen Tafel und einer Lissajous-Kurve auf einer zylindrischen Tafel gezeigt. Man erkennt bei beiden Abbildungen ein gutes Ergebnis. In dieser Arbeit wird der Freiheitsgrad des Gelenks 7 des Roboters nicht genutzt, da sich der rotationssymmetrische Stift in der Drehachse dieses Gelenks befindet. Wird jedoch ein anderer Endeffektor verwendet, wäre dieser Freiheitsgrad nutzbar.

In diesem Kapitel wurden Messergebnisse von verschiedenen Versuchen gezeigt und diskutiert. Im ersten Versuch zeichnete der Endeffektor des Roboters eine Lissajous-Kurve auf eine Tafel, während diese durch eine Linearachse bewegt wurde. Es wurde ein gutes Ergebnis erzielt, obwohl eine relative Messung der Position der Tafel verwendet wurde. In den drei darauffolgenden Versuchen wurde die Tafel frei im Arbeitsraum des Roboters durch einen Menschen bewegt. Gleichzeitig zeichnete der Roboter verschiedene Kurven auf die Tafel. Es wurde durch die Regelfehler, die Gelenkwinkel sowie die Drehmomente das Verhalten des Systems sichtbar gemacht. Weiters wurde gezeigt, dass der Pfadfortschritt während einer schnellen Bewegung der Tafel angehalten wurde. Zuletzt wurde noch auf die Ergebnisse der gezeichneten Kurven auf der Tafel eingegangen, wobei Gründe für etwaige Abweichungen diskutiert wurden. Durch die Filterung der Messung wird das Ergebnis stark beeinflusst. Werden die Messdaten nicht gefiltert und der Regelfehler so schnell wie möglich ausgeregelt, führt dies zu schnellen Bewegungen des Roboters. Diese schnellen Bewegungen sind bei kollaborativen Arbeiten mit dem Menschen zu vermeiden.

6 Zusammenfassung

Diese Arbeit zeigt, wie ein Roboter seinen Arbeitsablauf mit einem bewegten Objekt synchronisieren kann. Diese Aufgabenstellung kann beispielsweise in der Fertigung verwendet werden, wenn sich ein Bauteil relativ zu einem Roboter bewegt. In dieser Arbeit wurde die Aufgabenstellung mit einem Stift am Roboter simuliert, wodurch die Pfade auf einer Tafel sichtbar gemacht wurden.

Um diese Aufgabe zu lösen, wurde in Kapitel 2 ein mathematisches Modell des verwendeten Roboters berechnet. Dazu wurde zuerst die Kinematik des Roboters beschrieben. Weiters wurde das dynamische Modell des Roboters mit starren Gelenken betrachtet, wobei dieses in einem weiteren Schritt um flexible Gelenke erweitert wurde. Die Kinematik konnte in dieser Arbeit genau modelliert werden, da diese von genau bekannten Parametern abhängig ist. Für das dynamische Modell mit flexiblen Gelenken werden jedoch Parameter benötigt, die nicht exakt bekannt sind. Weiters konnten diese durch die spezielle Art der Ansteuerung nicht identifiziert werden. Zudem bestehen Ungenauigkeiten in Bezug auf die Reibung sowie die vom Hersteller implementierte Gravitationskompensation des Roboters.

In Kapitel 3 wurde das verwendete Regelungskonzept dieser Arbeit gezeigt. Es wurde ein Regelgesetz der inversen Dynamik des Roboters im Weltkoordinatensystem verwendet. Um die Dynamik des Endeffektors festzulegen, wurden die Reglerparameter der Positionsregelung in Bezug auf die Tafel gewählt. Die Orientierung des Endeffektors wurde mit Hilfe von Quaternionen geregelt und für die Positions- und Orientierungsregelung wurde ein Stabilitätsbeweis angegeben. Das Regelgesetz berücksichtigt weiters die Nulldynamik des Roboters. Es hat sich gezeigt, dass das gewählte Regelgesetz robust gegen die oben angeführten Parameterunsicherheiten ist. Für andere Regelungskonzepte, wie eine Kraftregelung des Endeffektors auf der Oberfläche, wären weitere Sensoren notwendig. Zusätzlich könnte die Position des Endeffektors gemessen werden. Mit diesen Konzepten wäre ein Ausgleich von Abweichungen durch die begrenzte absolute Genauigkeit des Roboters möglich.

Die gesamte Pfadplanung und Messung der bewegten Tafel wurde in Kapitel 4 dargestellt. Es wurde die Planung des Pfades auf der Tafel gezeigt, wobei durch ein Abheben von der Oberfläche der Tafel *pick and place* Aufgaben simuliert werden konnten. Außerdem wurden die Pfade auch auf ein zylindrisches Objekt projiziert. Die Pose der Tafel wurde über ein fixes Kamerasystem gemessen, wobei eine Filterung notwendig war. Durch die Messung des Objekts mit diesem Kamerasystem ist der Arbeitsraum des Roboters jedoch vorgegeben. Möchte man den Roboter an verschiedenen Standorten einsetzen, könnte ein Kamerasystem direkt am Endeffektor des Roboters verwendet werden. Somit würde eine absolute Messung des Objekts umgangen werden. Zuletzt wurde gezeigt, wie der gewünschte Pfad dynamisch verlangsamt oder gestoppt werden kann. Dadurch ist ein kurzzeitiges Stoppen des Arbeitsablaufs auf der Tafel möglich, wenn sich diese zu schnell bewegt.

Die Ergebnisse der Versuche wurden in Kapitel 5 dargestellt. Dabei wurde gezeigt, welche Ergebnisse das Regelungskonzept bei einer linearen Bewegung der Tafel erzielt. Danach wurde auf einer beliebig im Raum bewegten Tafel verschiedenen Pfaden gefolgt, wobei die Messergebnisse ausführlich diskutiert wurden. Wird die Bewegung der Tafel durch einen Menschen ausgeführt, sind zusätzliche Anpassungen an das Verhalten des Roboters erforderlich. Die Bewegung des Roboters sollte nicht zu schnell und für den Anwender vorhersehbar erfolgen.

In dieser Arbeit ist beschrieben, wie sich ein Roboter mit einem bewegten Objekt synchronisieren lässt und zusätzlich darauf einem Pfad folgt. Genaues Wissen über das Modell des Roboters ist dabei von Vorteil, da damit Arbeitsabläufe genau ausgeführt werden können. Weiters ist das Ergebnis von der Genauigkeit und präzisen Einbindung des Messsystems in das Gesamtsystem abhängig. Bei der Entwicklung der Regelung sowie der Pfadplanung muss im Speziellen bei kollaborativen Robotern verstärkt auf die Sicherheit von Mensch und Maschine geachtet werden.

A Anhang

A.1 Herleitungen

A.1.1 Quaternionenprodukt

In diesem Abschnitt wird das Quaternionenprodukt definiert und Gleichung (3.13) hergeleitet, siehe [18]. Ein Einheitsquaternion wird als $\mathbf{Q} = [\eta \ \boldsymbol{\varepsilon}^T]^T$ dargestellt, wobei η den Skalaranteil und $\boldsymbol{\varepsilon}$ den Vektoranteil angibt. Das Produkt zweier Quaternionen \otimes wird mit

$$\mathbf{Q}_1 \otimes \mathbf{Q}_2 = \begin{bmatrix} \eta_1 \\ \boldsymbol{\varepsilon}_1 \end{bmatrix} \otimes \begin{bmatrix} \eta_2 \\ \boldsymbol{\varepsilon}_2 \end{bmatrix} = \begin{bmatrix} \eta_1 \eta_2 - \boldsymbol{\varepsilon}_1^T \boldsymbol{\varepsilon}_2 \\ \eta_1 \boldsymbol{\varepsilon}_2 + \eta_2 \boldsymbol{\varepsilon}_1 + \boldsymbol{\varepsilon}_1 \times \boldsymbol{\varepsilon}_2 \end{bmatrix} \quad (\text{A.1})$$

definiert. Dabei gibt \times das Kreuzprodukt zweier Vektoren an. Wird die Gleichung (3.11) mit Quaternionen beschrieben, wobei $\mathbf{R}^T = \mathbf{R}^{-1}$ und für das inverse Quaternion $\mathbf{Q}^{-1} = [\eta \ -\boldsymbol{\varepsilon}^T]^T$ gilt, erhält man

$$\delta \mathbf{Q} = \mathbf{Q}_d \otimes \mathbf{Q}_e^{-1} = \begin{bmatrix} \delta \eta \\ \delta \boldsymbol{\varepsilon} \end{bmatrix} = \begin{bmatrix} \eta_d \\ \boldsymbol{\varepsilon}_d \end{bmatrix} \otimes \begin{bmatrix} \eta_e \\ -\boldsymbol{\varepsilon}_e \end{bmatrix} = \begin{bmatrix} \eta_d \eta_e + \boldsymbol{\varepsilon}_d^T \boldsymbol{\varepsilon}_e \\ \eta_e \boldsymbol{\varepsilon}_d - \eta_d \boldsymbol{\varepsilon}_e - \boldsymbol{\varepsilon}_d \times \boldsymbol{\varepsilon}_e \end{bmatrix}. \quad (\text{A.2})$$

A.1.2 Winkelgeschwindigkeit aus Quaternionen

In diesem Abschnitt wird die Gleichung (4.35) aus (3.15) hergeleitet, siehe [18]. Wird auf (3.15) das inverse Quaternion \mathbf{Q}^{-1} mit dem Quaternionenprodukt \otimes multipliziert, erhält man

$$\dot{\mathbf{Q}} \otimes \mathbf{Q}^{-1} = \frac{1}{2} \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \otimes \mathbf{Q} \otimes \mathbf{Q}^{-1}. \quad (\text{A.3})$$

Da wegen (3.12) und (A.1) die Beziehung

$$\mathbf{Q} \otimes \mathbf{Q}^{-1} = \begin{bmatrix} \eta \\ \boldsymbol{\varepsilon} \end{bmatrix} \otimes \begin{bmatrix} \eta \\ -\boldsymbol{\varepsilon} \end{bmatrix} = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} \quad (\text{A.4})$$

gilt, berechnet sich (A.3) zu

$$\begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} = 2 \dot{\mathbf{Q}} \otimes \mathbf{Q}^{-1} = 2 \begin{bmatrix} \dot{\eta} \\ \dot{\boldsymbol{\varepsilon}} \end{bmatrix} \otimes \begin{bmatrix} \eta \\ -\boldsymbol{\varepsilon} \end{bmatrix}. \quad (\text{A.5})$$

Wird das Quaternionenprodukt (A.1) von (A.5) berechnet, erhält man die Gleichung (4.35)

$$\boldsymbol{\omega} = 2(\eta \dot{\boldsymbol{\varepsilon}} - \dot{\eta} \boldsymbol{\varepsilon} - \dot{\boldsymbol{\varepsilon}} \times \boldsymbol{\varepsilon}). \quad (\text{A.6})$$

A.1.3 Ableitung der Lyapunov Funktion

In diesem Abschnitt wird die zeitliche Ableitung der Lyapunov Funktion (3.14) zu dem Ergebnis (3.17) detailliert hergeleitet, siehe [18, 27]. Wird

$$V(\zeta) = K_o(\eta_d - \eta_e)^2 + (\varepsilon_d - \varepsilon_e)^T \mathbf{K}_o(\varepsilon_d - \varepsilon_e) + \frac{1}{2}(\omega_d - \omega_e)^T(\omega_d - \omega_e) \quad (\text{A.7})$$

nach der Zeit abgeleitet, erhält man

$$\dot{V} = \underbrace{2(\eta_d - \eta_e)K_o(\dot{\eta}_d - \dot{\eta}_e) + 2(\varepsilon_d - \varepsilon_e)^T \mathbf{K}_o(\dot{\varepsilon}_d - \dot{\varepsilon}_e)}_a + \underbrace{(\omega_d - \omega_e)^T(\dot{\omega}_d - \dot{\omega}_e)}_b. \quad (\text{A.8})$$

Der Term a lässt sich mit (3.15), (3.16) und (2.7) als

$$\begin{aligned} a &= (\eta_d - \eta_e)K_o(-\omega_d^T \varepsilon_d + \omega_e^T \varepsilon_e) + (\varepsilon_d - \varepsilon_e)^T \mathbf{K}_o(\omega_d \eta_d + \mathbf{S}(\omega_d)\varepsilon_d - \omega_e \eta_e - \mathbf{S}(\omega_e)\varepsilon_e) = \\ &= \omega_d^T \mathbf{K}_o(\eta_e \varepsilon_d - \eta_d \varepsilon_e) - \omega_e^T \mathbf{K}_o(\eta_e \varepsilon_d - \eta_d \varepsilon_e) - \varepsilon_d^T \mathbf{K}_o \mathbf{S}(\omega_e)\varepsilon_e - \varepsilon_e^T \mathbf{K}_o \mathbf{S}(\omega_d)\varepsilon_d \end{aligned} \quad (\text{A.9})$$

anschreiben, wobei mit dem Kreuzprodukt zweier Vektoren \times

$$\varepsilon_e^T \mathbf{K}_o \mathbf{S}(\omega_d)\varepsilon_d = \varepsilon_e^T \mathbf{K}_o(\omega_d \times \varepsilon_d) = \omega_d^T \mathbf{K}_o(\varepsilon_d \times \varepsilon_e) \quad (\text{A.10})$$

und

$$\varepsilon_d^T \mathbf{K}_o \mathbf{S}(\omega_e)\varepsilon_e = -\omega_e^T \mathbf{K}_o(\varepsilon_d \times \varepsilon_e) \quad (\text{A.11})$$

gilt. Daraus folgt mit (3.13)

$$a = (\omega_d - \omega_e)^T \mathbf{K}_o \delta \varepsilon. \quad (\text{A.12})$$

Der Term b ergibt sich mit (3.7) zu

$$b = (\omega_d - \omega_e)^T (-\mathbf{K}_\omega(\omega_d - \omega_e) - \mathbf{K}_o \mathbf{e}_o) \quad (\text{A.13})$$

Aus (A.12) und (A.13) berechnet sich die Formel (3.17) zu

$$\dot{V} = -(\omega_d - \omega_e)^T \mathbf{K}_\omega(\omega_d - \omega_e) + (\omega_d - \omega_e)^T \mathbf{K}_o(\delta \varepsilon - \mathbf{e}_o) \quad (\text{A.14})$$

A.2 Parameter

In der Parametertabelle Tabelle A.1 gibt $\text{diag}(\mathbf{K})$ an, dass es sich bei der Matrix \mathbf{K} um eine Diagonalmatrix handelt, wobei lediglich die Diagonalelemente dargestellt werden. Zusätzlich werden in Tabelle A.2 weitere Parameter angeführt.

	Bezeichnung	Symbol	Wert	Einheit
Kinematik	Distanz von Adapterplatte zu Σ_0	d_0	312	mm
	Distanz von Σ_2 zu Σ_3	d_3	400	mm
	Distanz von Σ_4 zu Σ_5	d_5	390	mm
	Distanz von Σ_7 zu Σ_e	d_e	240	mm
	Grav.beschleunigung	g	-9.81	m/s ²
Reibident.	Abtastzeit	T_a	1	ms
	Proportionalverstärkung,	K_{Pv}	10	N m s/rad
	Integralverstärkung des Gelenkgeschw.fehlers	K_{Iv}	100	N m s/rad
Reglermatrizen	Differentialverstärkung,	$\text{diag}(\mathbf{K}_v)$	19, 19, 19	1/s
	Proportionalverstärkung,	$\text{diag}(\mathbf{K}_p)$	$10^3(1, 1, 0.5)$	1/s ²
	Integralverstärkung des Positionsfehlers	$\text{diag}(\mathbf{K}_{Ip})$	$2 \cdot 10^{-1}(1, 1, 1)$	1/s ³
	Proportionalverstärkung des Winkelgeschw.fehlers	$\text{diag}(\mathbf{K}_\omega)$	19, 19, 19	1/s
	Proportionalverstärkung des Quaternionenfehlers	$\text{diag}(\mathbf{K}_o)$	$10^3(1, 1, 1)$	1/s ²
	Differentialverstärkung, Proportionalverstärkung der Gelenkwinkel	$\text{diag}(\mathbf{K}_{vn})$ $\text{diag}(\mathbf{K}_{pn})$	$20(1, 1, 1, 1, 1, 1, 1)$ $30(1, 1, 1, 1, 1, 1, 1)$	1/s 1/s ²
Eigenwerte	mit alg. Vielfachheit 2	λ_1	$-2 \cdot 10^{-4}$	1
	mit alg. Vielfachheit 1	λ_2	$-4 \cdot 10^{-4}$	1
	mit alg. Vielfachheit 1	λ_3	$-9.4998 \pm 20.2422\text{I}$	1
	mit alg. Vielfachheit 2	λ_4	$-9.4999 \pm 30.1620\text{I}$	1
	mit alg. Vielfachheit 7	$\lambda_{n,1}$	-1.6334	1
	mit alg. Vielfachheit 7	$\lambda_{n,2}$	-18.3666	1

Tabelle A.1: Systemparameter, Kinematik siehe [20].

	Bezeichnung	Symbol	Wert	Einheit
Pfadplanung	Radius Kurve	r	100	mm
	Radius Zylinder	l	200	mm
	Startpunkt	φ_{sta}	0	°
	Endpunkt	$\varphi_{end} = \varphi_{\ddot{u}be}$	5	°
	Anfangswert	z_{sta}	0	mm
	Endwert	z_{end}	-10	mm
	Dauer	φ_{dau}	31	°
	Verschiebung	φ_v	0	°
	Ende Verzögerung	$\varphi_1 = \varphi_{dau}/2$	15.5	°
	Ende 1-ter Übergang	$\varphi_2 = \varphi_1 + \varphi_{\ddot{u}be}$	20.5	°
	Ende z_{end} -Plateau	$\varphi_3 = \varphi_2 + \varphi_{dau}$	51.5	°
	Ende 2-ter Übergang	$\varphi_4 = \varphi_3 + \varphi_{\ddot{u}be}$	56.5	°
	Ende z_{sta} -Plateau	$\varphi_5 = \varphi_4 + \varphi_1$	72	°
Translation	x-Koordinate Ebene	x_w	-15	mm
	y-Koordinate Ebene	y_w	156	mm
	z-Koordinate Ebene	z_w	195	mm
	x-Koordinate Zylinder	x_c	70	mm
	y-Koordinate Zylinder	y_c	160	mm
	z-Koordinate Zylinder	z_c	210	mm
	x-Koordinate Basis	x_r	-147.5	mm
y-Koordinate Basis	y_r	110	mm	
z-Koordinate Basis	z_r	-325.5	mm	
Filter	Eigenwerte			
	mit alg. Vielfachheit 2	λ_p	-40	1
	mit alg. Vielfachheit 2	λ_o	-5	1
Pfadfortschritt	Beginn	ϑ_{beg}	0	1
	Stopp	ϑ_{sto}	35	1
	Anfangswert	φ_{beg}	0	°
	Endwert	φ_{sto}	720	°
	Eingangsverstärkung	n_u	10	1
	Eigenwerte			
	mit alg. Vielfachheit 2	λ_ϑ	-10	1
Synchronisierung	Proportionalverstärkung des Positionsfehlers	$\text{diag}(\mathbf{K}_{p,l})$	10(1, 1, 1)	1/s ²
	Proportionalverstärkung des Quaternionenfehlers	$\text{diag}(\mathbf{K}_{o,l})$	10(1, 1, 1)	1/s ²
	Zeitrampe 1	t_{r1}	3	s
	Zeitrampe 2	t_{r2}	10	s

Tabelle A.2: Systemparameter

A.3 Stifthalterung

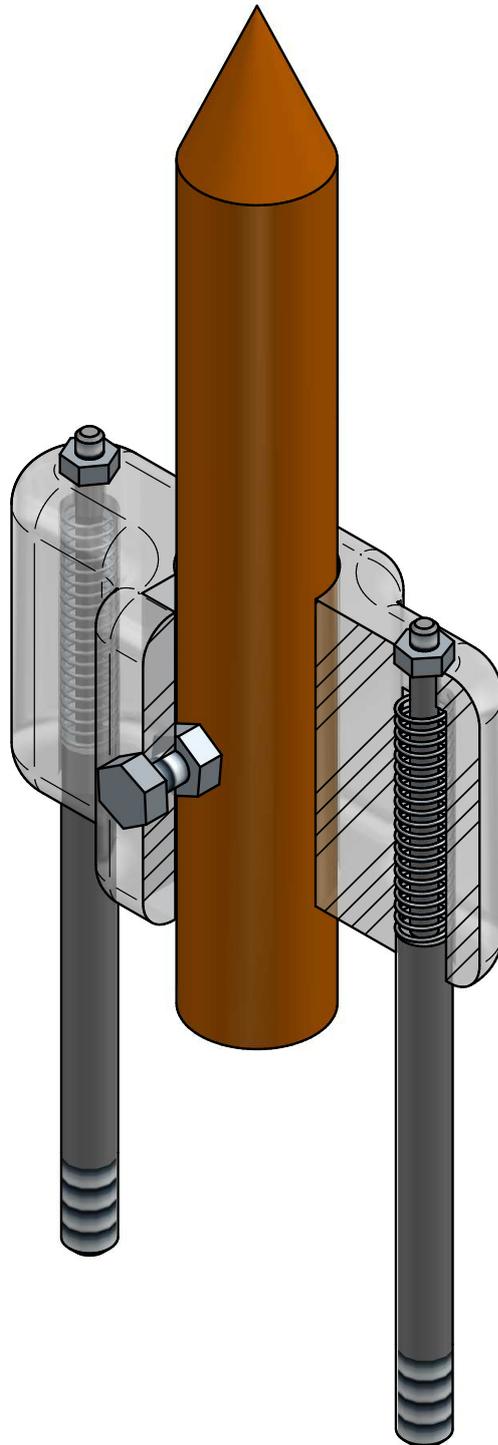


Abbildung A.1: Modell der federgelagerten Stifthalterung.

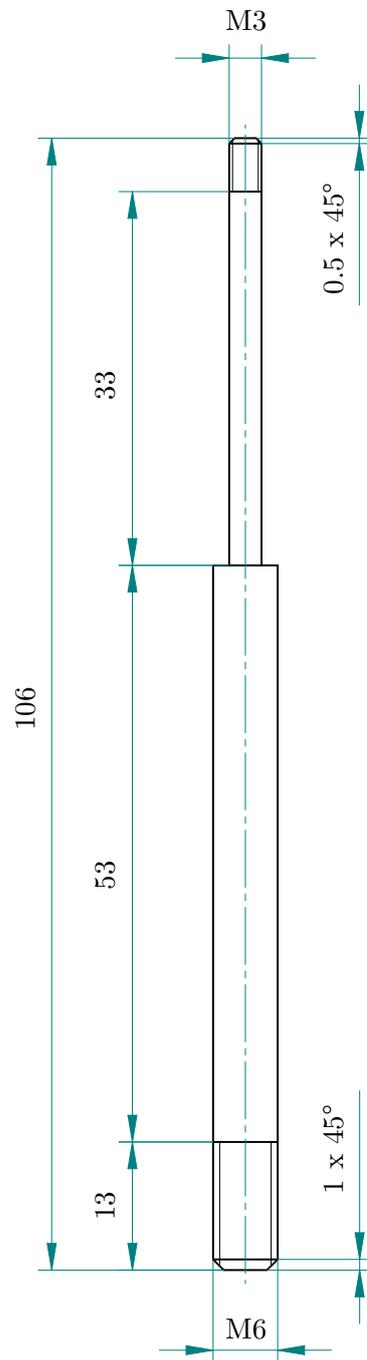


Abbildung A.2: Bauteilzeichnung der zwei Führungsstäbe für die Stifthalterung, Maße in mm.

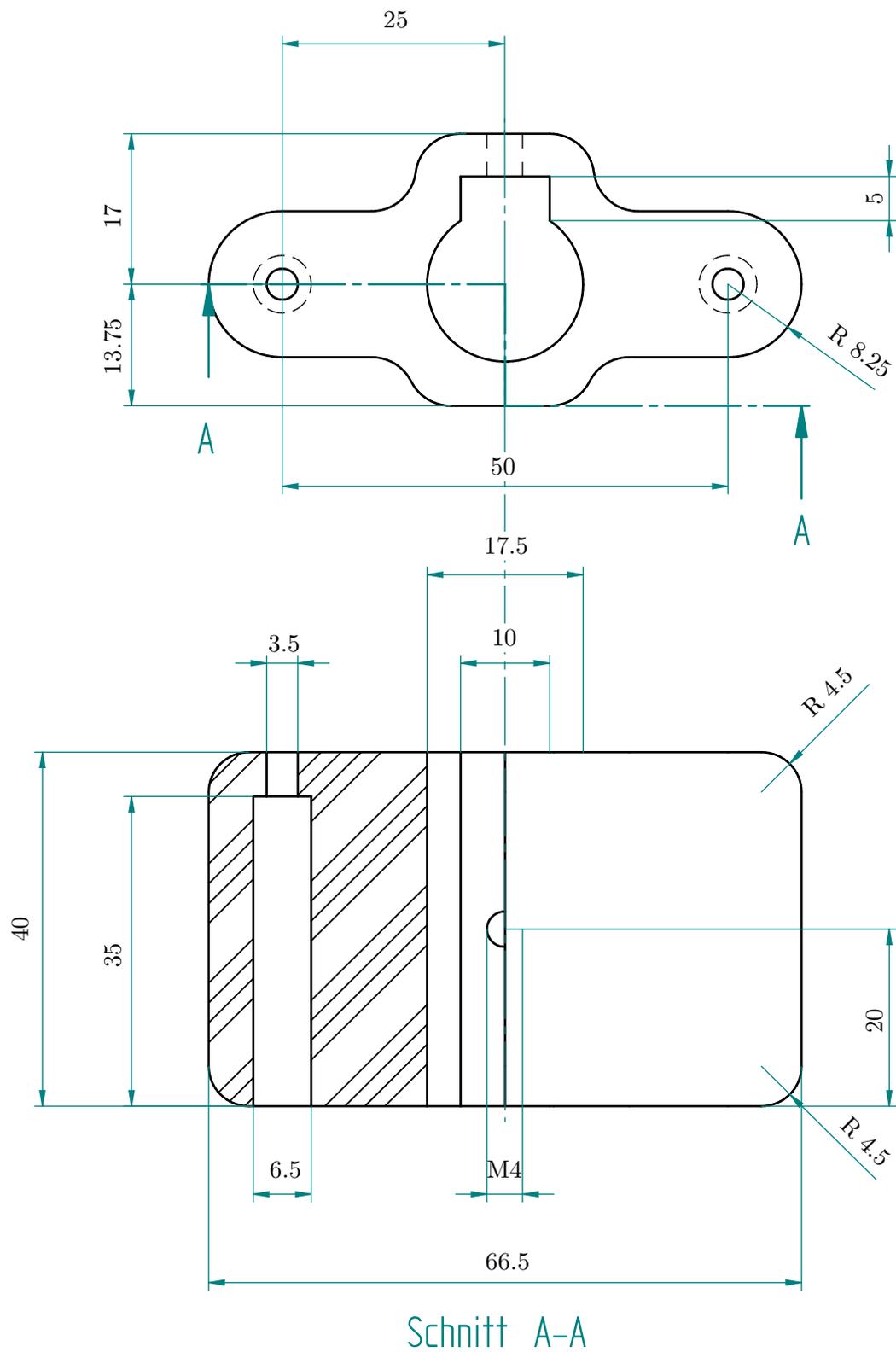


Abbildung A.3: Bauteilzeichnung der Stifthalterung, alle unbemaßten Radien R 5, Maße in mm.

Literatur

- [1] I. Kresse, *This models are licensed under CC-BY-3.0: Copyright (c) 2010 by Ingo Kresse <kresse@in.tum.de>.* *This work is made available under the terms of the Creative Commons Attribution 3.0 license*, aufgerufen am: 18. Okt. 2018, 2010. Adresse: <http://creativecommons.org/licenses/by/3.0/>.
- [2] I. I. Blekhman, P. S. Landa und M. G. Rosenblum, „Synchronization and Chaotization in Interacting Dynamical Systems“, *Applied Mechanics Reviews*, Bd. 48, Nr. 11, S. 733–752, 1995.
- [3] I. Blekhman, A. Fradkov, H. Nijmeijer und A. Pogromsky, „On self-synchronization and controlled synchronization“, *Systems & Control Letters*, Bd. 31, Nr. 5, S. 299–305, 1997.
- [4] A. Rodriguez-Angeles und H. Nijmeijer, „Mutual Synchronization of Robots via Estimated State Feedback: A Cooperative Approach“, *IEEE Transactions on Control Systems Technology*, Bd. 12, Nr. 4, S. 542–554, 2004.
- [5] Y. Bouteraa, J. Ghommam, N. Derbel und G. Poisson, „Mutual and external synchronization control of multi-robot systems“, in *2010 7th International Multi-Conference on Systems, Signals and Devices*, IEEE, Amman, Juni 2010, S. 1–6.
- [6] A. Rodriguez-Angeles und H. Nijmeijer, „Coordination of two robot manipulators based on position measurements only“, *International Journal of Control*, Bd. 74, Nr. 13, S. 1311–1323, 2001.
- [7] DIN EN ISO 10218-1, *Industrieroboter- Sicherheitsanforderungen Teil 1: Roboter (ISO/DIS 1028-1:2011); deutsche Fassung EN ISO 10218-1:2011*, Norm, 2011.
- [8] C. Wang, C.-Y. Lin und M. Tomizuka, „Visual Servoing Considering Sensing Dynamics and Robot Dynamics“, *IFAC Proceedings Volumes*, Bd. 46, Nr. 5, S. 45–52, 2013.
- [9] C. Wang, C.-Y. Lin und M. Tomizuka, „Design of kinematic controller for real-time vision guided robot manipulators“, in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Hong Kong, Mai 2014, S. 4141–4146.
- [10] X. Yun, T. Tarn und A. Bejczy, „Dynamic coordinated control of two robot manipulators“, in *1989 28th IEEE Conference on Decision and Control*, IEEE, Bd. 3, Tampa, Dez. 1989, S. 2476–2481.
- [11] J. Y. Lau, W. Liang und K. K. Tan, „Motion tracking control of surgical device based on sliding mode enhanced adaptive disturbance observer strategy“, in *2017 11th Asian Control Conference (ASCC)*, IEEE, Gold Coast, Dez. 2017, S. 2304–2309.

- [12] D. Sun und J. Mills, „Adaptive synchronized control for coordination of multirobot assembly tasks“, *IEEE Transactions on Robotics and Automation*, Bd. 18, Nr. 4, S. 498–510, 2002.
- [13] T. Faulwasser, J. Matschek, P. Zometa und R. Findeisen, „Predictive path-following control: Concept and implementation for an industrial robot“, in *2013 IEEE International Conference on Control Applications (CCA)*, IEEE, Hyderabad, Aug. 2013, S. 128–133.
- [14] T. Faulwasser, T. Weber, P. Zometa und R. Findeisen, „Implementation of Non-linear Model Predictive Path-Following Control for an Industrial Robot“, *IEEE Transactions on Control Systems Technology*, Bd. 25, Nr. 4, S. 1505–1511, 2017.
- [15] J. Matschek, J. Bethge, P. Zometa und R. Findeisen, „Force Feedback and Path Following using Predictive Control: Concept and Application to a Lightweight Robot“, *IFAC-PapersOnLine*, Bd. 50, Nr. 1, S. 9827–9832, 2017.
- [16] V. Bargsten, P. Zometa und R. Findeisen, „Modeling, parameter identification and model-based control of a lightweight robotic manipulator“, in *2013 IEEE International Conference on Control Applications (CCA)*, IEEE, Hyderabad, Aug. 2013, S. 134–139.
- [17] B. Bischof, T. Glück und A. Kugi, „Combined Path Following and Compliance Control for Fully Actuated Rigid Body Systems in 3-D Space“, *IEEE Transactions on Control Systems Technology*, Bd. 25, Nr. 5, S. 1750–1760, 2017.
- [18] B. Siciliano, L. Sciavicco, L. Villani und G. Oriolo, *Robotics: Modelling, Planning and Control*. London: Springer-Verlag, 2009.
- [19] C. Gaz, F. Flacco und A. D. Luca, „Identifying the dynamic model used by the KUKA LWR: A reverse engineering approach“, in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Hong Kong, Mai 2014, S. 1386–1392.
- [20] KUKA Roboter GmbH, „Leichtbauroboter 4+ Betriebsanleitung“, Version BA LBR 4+ V2 de, Juli 2010.
- [21] M. W. Spong, S. Hutchinson und M. Vidyasagar, *Robot Dynamics and Control*, Second Edition. 28. Jän. 2004, aufgerufen am: 18. Okt. 2018. Adresse: <http://home.deib.polimi.it/gini/robot/docs/spong.pdf>.
- [22] C. Gaz, F. Flacco und A. D. Luca, „Extracting feasible robot parameters from dynamic coefficients using nonlinear optimization methods“, in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Stockholm, Mai 2016, S. 2075–2081.
- [23] C. Ott, *Cartesian Impedance Control of Redundant and Flexible-Joint Robots*, Ser. Springer Tracts in Advanced Robotics 49. Berlin: Springer-Verlag, 2008.
- [24] M. W. Spong, „Modeling and Control of Elastic Joint Robots“, *Journal of Dynamic Systems, Measurement, and Control*, Bd. 109, Nr. 4, S. 310–319, 1987.
- [25] V. L. Popov, *Kontaktmechanik und Reibung: Ein Lehr- und Anwendungsbuch von der Nanotribologie bis zur numerischen Simulation*. Berlin: Springer-Verlag, 2009.

-
- [26] KUKA Roboter GmbH, „KUKA.FastResearchInterface 1.0“, KUKA.FRI 1.0 V2 de, Sep. 2011.
 - [27] J. Yuan, „Closed-loop manipulator control using quaternion feedback“, *IEEE Journal on Robotics and Automation*, Bd. 4, Nr. 4, S. 434–440, 1988.
 - [28] H. K. Khalil, *Nonlinear Systems*, Third Edition. New Jersey: Prentice Hall, 2002.
 - [29] C.-T. Chen, *Linear System Theory and Design*, Third Edition. New York: Oxford University Press, 1999.
 - [30] J.-J. Slotine und W. Li, *Applied Nonlinear Control*. New Jersey: Prentice Hall, 1991.
 - [31] F. M. Michael Malisoff, *Constructions of Strict Lyapunov Functions*. London: Springer-Verlag, 2009.
 - [32] M. W. Hirsch und S. Smale, *Differential Equations, Dynamical Systems, and Linear Algebra*. London: Academic Press, 1974.
 - [33] OptiTrack, *Specifications*, aufgerufen am: 12. Okt. 2018. Adresse: <https://optitrack.com/products/prime-17w/specs.html>.

Eidesstattliche Erklärung

Hiermit erkläre ich, dass die vorliegende Arbeit gemäß dem Code of Conduct – Regeln zur Sicherung guter wissenschaftlicher Praxis (in der aktuellen Fassung des jeweiligen Mitteilungsblattes der TU Wien), insbesondere ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel, angefertigt wurde. Die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder in ähnlicher Form in anderen Prüfungsverfahren vorgelegt.

Wien, im November 2018

Thomas Weingartshofer