

**Lehrstuhl für Elektrische Antriebssysteme
Technische Universität München**

**Modellbasierte Regelung mit
nichtlinearen inversen Systemen und Beobachtern
von Robotern mit elastischen Gelenken**

Michael Thümmel

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik
der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender:

Univ.-Prof. Dr.-Ing. Hans-Georg Herzog

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing. Dr.-Ing. h.c. Dierk Schröder
2. Prof. Dr. Alessandro De Luca,
Univ. degli Studi di Roma „La Sapienza“ / Italien
3. Hon.-Prof. Dr.-Ing. Martin Otter

Die Dissertation wurde am 19.01.2006 bei der Technischen Universität München eingereicht
und durch die Fakultät für Elektrotechnik und Informationstechnik am 26.06.2006
angenommen.

Vorwort

Die vorliegende Arbeit entstand im Laufe meiner Tätigkeit als wissenschaftlicher Mitarbeiter in der Abteilung Entwurfsorientierte Regelungstechnik am Institut für Robotik und Mechatronik des Deutschen Zentrums für Luft- und Raumfahrt (DLR) in Oberpfaffenhofen. Während dieser Zeit gab es viele, die zum Gelingen der Arbeit beigetragen haben und ohne deren Hilfe die Arbeit nicht in der vorliegenden Form möglich gewesen wäre. Bei Ihnen möchte ich mich an dieser Stelle recht herzlich bedanken.

Herrn Prof. Dr.-Ing. Dr.-Ing. h. c. Dierk Schröder danke ich für die Betreuung und zielgerichtete Unterstützung der Arbeit an der TU München und die Übernahme des Erstgutachtens.

Mein Dank gilt auch Prof. Dr. Alessandro De Luca, der sich spontan bereit erklärt hat, das Zweitgutachten zu übernehmen und dessen großes Interesse an der Arbeit mich besonders gefreut hat.

Danken möchte ich auch Prof. Dr.-Ing. Martin Otter für die langjährige Begleitung der Arbeit beim DLR, für Anregungen und Unterstützung, insbesondere in Modellierungsfragen und die Erstellung des dritten Gutachtens.

Des Weiteren danke ich Prof. Dr.-Ing. Hans-Georg Herzog für die Übernahme des Prüfungsvorsitzes.

Für die Entstehung der Arbeit war eine Umgebung notwendig, in der regelungstechnische Methoden und Werkzeuge mit konkreten Problemen aus der Robotik zusammentreffen. Bedanken möchte ich mich deshalb ebenfalls bei Dr.-Ing. Johann Bals, dessen Abteilung eine solche Umgebung bot und der durch zahlreiche Anregungen zu dieser Arbeit beitrug.

Auch der Firma KUKA Roboter GmbH, die mich bei Anfragen und Versuchen stets unterstützt hat, möchte ich in diesem Zusammenhang danken.

Meine tägliche Arbeit beim DLR hätte nicht so viel Spaß gemacht ohne die Kollegen, die mir mit Rat und Tat zur Seite standen und insbesondere die Mitglieder des Robotik Teams. Hervorheben möchte ich hierbei Christian Ballauf, Carina Dittmann, Robert Höpler, Astrid Jaschinski, Gerhard Schillhuber und Stefan Schneider, denen ich für unzählige, nicht nur technische Diskussionen und gemeinsam bewältigte Aufgaben herzlich danken möchte.

Besonders danken möchte ich zudem Robert Höpler für die lange, erfolgreiche Zusammenarbeit und für seine Hilfsbereitschaft in allen formalen Fragen beim Verfassen dieser Arbeit.

Schließlich ist es mir wichtig mich ganz besonders bei meiner Frau Martina zu bedanken, deren Unterstützung und Geduld bei der Entstehung dieser Arbeit ein wichtiger Rückhalt für mich waren.

München, 28. März 2007

Michael Thümmel

Kurzzusammenfassung

In der vorliegenden Arbeit wird eine modellbasierte Regelung für Roboter mit elastischen Gelenken entwickelt und experimentell erprobt. Das Ziel dieser Regelung ist die Optimierung der Roboterdynamik in Bezug auf das Schwingungsverhalten und die Bahnengenauigkeit des Roboters. Dazu werden bei der Modellierung sowohl die Dynamik des Mehrkörpersystems als auch dynamische Effekte der Antriebsstränge in Form detaillierter, physikalisch motivierter Modelle berücksichtigt. Das Modell eines sechsachsigen Industrieroboter wird aufgestellt und die Parameter der Antriebsstränge identifiziert.

Um die gewünschte Dynamik zu erreichen wird eine Regelungsstruktur mit zwei strukturellen Freiheitsgraden eingesetzt. Diese benutzt in der Steuerung eine aufwändige, nichtlineare Inverse des Robotermodells. Der Schwerpunkt der Arbeit liegt deshalb auf der Berechnung von inversen Systemen, wofür eine leistungsfähige und effiziente Möglichkeit auf Basis der Indexreduktion von differential-algebraischen Gleichungssystemen vorgestellt wird, und auf der Analyse der Inversen des Robotermodells.

Die Regelung erfolgt über Gelenkregler mit variablen Parametern und einen zentralen Beobachter. Es wird gezeigt, dass die Gelenke im Beobachter durch Verwendung von Beschleunigungssensoren weitgehend entkoppeln werden können, wodurch sich die Komplexität des Modells reduzieren lässt. Versuche an einem Industrieroboter belegen die Verbesserung des dynamischen Verhaltens und zeigen die praktische Anwendbarkeit des entworfenen Reglers.

Abstract

This paper deals with model based control of robots with elastic joints. A controller aiming at the optimization of the robot dynamics with respect to oscillations and path accuracy is developed and tested experimentally. The controller uses detailed, physically motivated models for the robots multibody dynamics as well as for the dynamic effects of the drives. Based on these, the model of a six axis articulated robot arm is derived and its drive parameters identified.

In order to achieve the desired dynamics, a controller framework with two degrees of freedom is used. The feedforward controller uses an inverse of the complete nonlinear robot dynamics. Therefore the main emphasis of this paper is on efficient calculation of inverse systems in general and the analysis of the robots inverse dynamics in particular. A powerful and efficient way for the computation of inverse systems, based on methods for index reduction in systems of differential-algebraic equations, is demonstrated.

The feedback controller consists of an observer for the entire robot and independent joint controllers with gain scheduling. It is shown that acceleration sensors on the robot arm achieve largely decoupled joints in the observer and thus a reduction in the model's complexity. The experimental results using the proposed controller demonstrate its applicability and a significantly improved dynamic behaviour of the six axis articulated robot arm.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Problemstellung	1
1.2	Stand der Technik	2
1.3	Eingesetzte Regelungsstruktur	4
1.4	Beitrag und Organisation der Arbeit	9
2	Modellierung der Roboterdynamik	11
2.1	Mehrkörperdynamik	11
2.1.1	Lagrange Formalismus	12
2.1.2	Modellierung des Mehrkörpersystems	12
2.1.3	Mehrkörperdynamik von Robotern mit elastischen Gelenken	14
2.2	Detailliertes Antriebsstrangmodell	19
2.2.1	Getriebe	19
	Funktionsprinzip und Aufbau	20
	Übersetzung mit Drehungleichförmigkeiten	20
	Nichtlineare Elastizität	24
	Getriebeverluste	26
2.2.2	Motor und Leistungselektronik	31
2.2.3	Gesamtes Antriebsstrangmodell	32
2.3	Gesamtmodell des Roboters	33
2.4	Modellierung mit Modelica	33
2.5	Modellierung des KUKA KR15/2 Roboters	35
2.5.1	Beschreibung des KR15/2	35
2.5.2	Modelica-Modell des KR15/2	36

3	Identifikation der Modellparameter	39
3.1	Methoden zur Parameteridentifikation	39
3.2	Parameteridentifikation mit nichtlinearen Simulationsmodellen	40
3.3	Identifikation der Parameter des KR15/2	42
3.3.1	Sensorausstattung	42
3.3.2	Direkte Messung einzelner Parameter	43
	Steifigkeit	43
	Reibung	44
	Übersetzung	47
3.3.3	Simulationsmodell zur Parameteridentifikation	49
4	Invertierung von Systemen	53
4.1	Grundlagen für die Invertierung	54
4.1.1	Relativer Grad	54
4.1.2	Transformation in Normalform	56
4.1.3	Feedback-Linearisierung	57
4.2	Invertierbarkeit von Systemen	58
4.3	Dynamik von inversen Systemen	59
4.3.1	Interne Dynamik	60
4.3.2	Nulldynamik	60
4.3.3	Stabilität der internen Dynamik	61
4.4	Invertierung von flachen Systemen	61
4.5	Berechnung von inversen Systemen	66
4.5.1	Berechnung über die Normalform	66
4.5.2	Hirschorn Algorithmus	67
4.5.3	Berechnung als DAE	67
4.6	Berechnung von inversen Systemen mit Indexreduktion	71
4.6.1	Pantelides Algorithmus	71
4.6.2	Dummy-Derivative Methode	75
	Beispiel: lineares Zwei-Massen-System	78
4.6.3	Invertierung mit Dymola	80

5	Invertierung des Robotermodells	83
5.1	Relativer Grad	83
5.2	Analyse des Robotermodells	85
5.2.1	Einachsiger Roboter	85
5.2.2	Mehrachsiges Roboter	89
5.3	Stabilitätsbeweis für die inverse Dynamik	94
5.3.1	Nulldynamik des Robotermodells	94
5.3.2	Input-to-State Stability der internen Dynamik	96
5.4	Inverse Dynamik des KR15/2	98
6	Regler- und Beobachterentwurf	102
6.1	Kompensation der motorseitigen Lagerreibung	102
6.2	Beobachter mit Beschleunigungsmessung	103
6.2.1	Kartesische Beschleunigungsmessung	103
6.2.2	Robotersystem mit Beschleunigungssensoren	105
6.2.3	Entkopplung durch Beschleunigungsmessungen	106
6.2.4	Beobachterstruktur und Konvergenz	108
6.2.5	Auslegung des Beobachters	111
6.2.6	Beobachterausslegung für den KR15/2	112
6.3	Reglerentwurf	115
6.3.1	PI-Zustandsregler	115
6.3.2	Positionsabhängiges Gain-Scheduling der Reglerparameter	117
6.3.3	Auslegung am KR15/2	119
7	Regleroptimierung und experimentelle Ergebnisse	121
7.1	Hardware-in-the-loop-Optimierung	122
7.1.1	Bewertungskriterien für das Fahrverhalten	122
7.1.2	Ablauf der Hardware-in-the-loop-Optimierung	123
7.2	Experimentelle Ergebnisse	125
7.2.1	Positioniersuche	126
7.2.2	Bahnfahren	130
7.2.3	Robustheit	133
8	Zusammenfassung und Ausblick	135

A	Mathematische Grundlagen	139
A.1	Block-Lower-Triangular-Transformation	139
A.2	Brunovski-Normalform	139
A.3	Input-to-State Stability	140
A.4	Lie-Ableitungen	142
B	Modelica Bibliothek	143
B.1	Mehrkörpersystem	143
B.2	Antriebsstrang	146
B.3	Invertierung von Modellen	147
B.4	Glieder des KR15/2	148
B.5	Antriebe des KR15/2	150
C	Bezeichnungen	152
C.1	Typografische Kennzeichnungen	152
C.2	Formelzeichen	152
C.3	Index	155

Abbildungsverzeichnis

1.1	Regelung mit zwei strukturellen Freiheitsgraden	5
1.2	Regelung mit zwei strukturellen Freiheitsgraden und Referenzmodell	7
1.3	Regelungsstruktur mit Referenzmodell und Beobachter	8
2.1	Aufbau des Roboters	13
2.2	Modell eines Antriebsstranges zur MKS-Modellierung	14
2.3	Definition der Größen an den Flanschen von Antriebsstrangelementen	19
2.4	Schematische Darstellung eines Zykloidengetriebes	21
2.5	Übertragungsfehler von Zykloidengetrieben am Beispiel eines Teijin RV-320E-171 Getriebes	22
2.6	Schwingungsamplitude am Abtrieb bei konstanter Antriebsdrehzahl am Bei- spiel eines Fine Cyclo F2C-T Getriebes	22
2.7	Schematische Darstellung des Einflusses eines exzentrischen Rades auf die Übersetzung	23
2.8	Qualitativer Verlauf der Steifigkeitskennlinie eines Zykloidengetriebes	25
2.9	Gebräuchliche statische Reibmodelle	28
2.10	Zahnflanken eines Stirnradgetriebes nach [NW02]	29
2.11	Darstellung des Gesamtmodells für den Antriebsstrang	32
2.12	Aufbau des KR15/2	36
2.13	Gesamtmodell des KR15/2 als Modelica Objektdiagramm	37
3.1	Parameteridentifikation mit nichtlinearem Simulationsmodell	41
3.2	Versuchsaufbau zur Messung der Steifigkeit der ersten Achse	44
3.3	Steifigkeitskennlinie der ersten Achse	45
3.4	Gesamte Reibung der ersten Achse bei konstanter Drehzahl	45
3.5	Einschwingvorgang der ersten Achse zur Abschätzung der Dämpfung	46
3.6	Lastabhängige Verluste der ersten Achse des KR15/2	47

3.7	Gemessene Drehungleichförmigkeiten der ersten Achse	48
3.8	Ausschnitt aus den Messdaten zur Drehungleichförmigkeit	48
3.9	Modelica-Modell des Antriebsstranges für die Identifikation	49
3.10	Vergleich der gemessenen Gelenkbewegungen mit dem Modell	51
3.11	Vergleich der gemessenen Drehungleichförmigkeiten mit dem Modell	52
3.12	Vergleich der Motormomente aus Messung und Modell bei verschiedenen Positionierbewegungen	52
4.1	Graphen des Pantelides Algorithmus 1	74
4.2	Graphen des Pantelides Algorithmus 2	75
4.3	Graphen des Pantelides Algorithmus 3	76
4.4	Objektdiagramm eines Systems mit zwei Massen	81
4.5	Inversion mit Integrator-kette am Eingang	82
5.1	Vergleich der Wirkungsgradmodelle	88
5.2	Vergleich des exakten und approximierten Reibmodells	89
5.3	Aufbau des DLR LBR 2	92
5.4	Darstellung der Nulldynamik als Reihenschaltung	95
5.5	Objektdiagramm des invertierbaren Getriebemodells mit approximierten Teilmodellen	100
5.6	Objektdiagramm des Modells zum Test der Inversen Dynamik des KR15/2	101
5.7	Mit Hilfe der inversen Dynamik des KR15/2 berechnete Sollgrößen	101
6.1	Regelungsstruktur mit Reibungskompensation	103
6.2	Darstellung von Welt- und Sensorkoordinatensystem	104
6.3	Position des Beschleunigungssensors am Arm und singuläre Konfigurationen	113
6.4	Vergleich von gemessenen und geschätzten Zuständen der ersten Achse des KR15/2	114
6.5	Aufbau des PI-Zustandsreglers	118
6.6	Aufbau der Regelung	118
6.7	Durch den Regler einstellbare Eigendynamik der ersten Achse	120
7.1	Prüfstand zur experimentellen Erprobung der Regelung	121
7.2	Bestimmung der Bahnabweichung eines Punktes p_k	122
7.3	Ablauf der Hardware-in-the-loop-Optimierung am Roboter	124
7.4	Optimierungsverlauf am Beispiel einer HIL-Optimierung der Reglerparameter für die ersten Achse	125

7.5	Vergleich des Positionierverhaltens der ersten Achse bei Bewegungen unterschiedlicher Länge an verschiedenen Arbeitspunkten	127
7.6	Positionen und Bewegungen für das Positionieren	128
7.7	Vergleich des Positionierverhaltens mit allen Achsen bei unterschiedlichen Bewegungen	129
7.8	Testbahnen zur Bewertung der Bahntreue	130
7.9	Vergleich der Regler anhand von Geraden	131
7.10	Vergleich der Regler anhand von Kreisen	132
7.11	Wahrscheinlichkeitsdichten des Kriteriums EP_2 beim Positionieren mit der ersten Achse an verschiedenen Arbeitspunkten	134
A.1	Kaskade mit externem Eingang für beide Systeme	141

Kapitel 1

Einleitung

1.1 Problemstellung

In den letzten Jahrzehnten wurde der Einsatz von Robotern in der industriellen Fertigung immer weiter verstärkt. Heute sind Roboter in vielen Bereichen der Fertigung unentbehrlich. Ein Grund für die hohe Verbreitung von Robotersystemen sind die im Laufe der Jahre stark gefallen Preise, welche den Einsatz von Robotern immer wirtschaftlicher machen. Eine ebenso große Rolle spielt auch die sich stetig steigernde Leistungsfähigkeit der Roboter in Bezug auf Genauigkeit und Geschwindigkeit. Während Roboter früher, z. B. beim Punktschweißen in der Automobilindustrie, fast ausschließlich für Positionieraufgaben eingesetzt wurden, ist in den letzten Jahren eine Fülle von Applikationen entstanden, bei denen der Roboter einer bestimmten Bahn folgen soll. Zu den typischen Aufgaben in diesem Bereich zählen das Laserschweißen, das Kleben und das Entgraten von Bauteilen. Die klassischen Positionieraufgaben können dabei als eine Untermenge der Bahnapplikationen gesehen werden, bei der nur der Endpunkt der Bahn durch die Aufgabe fest vorgegeben ist und die Bahn selbst vom Roboter nach Kriterien wie z. B. minimaler Verfahrszeit oder Energieverbrauch optimiert und festgelegt wird. Damit ist die Güte, mit welcher ein Roboter einer vorgegebenen Bahn folgen kann, von zentraler Bedeutung für dessen Einsatzmöglichkeiten.

Im Laufe der Entwicklung sind bezüglich Kinematik und Aktuierung der Roboter die verschiedensten Bauformen entstanden. Aufgrund ihrer hohen Flexibilität und des preisgünstigen mechanischen Aufbaus sind serielle Kinematiken, wie z. B. bei Knickarmrobotern, universell einsetzbar und haben in der industriellen Fertigung mit Abstand die größte Verbreitung gefunden. Die Glieder dieser kinematischen Kette können durch geeignete Konstruktion sehr steif gebaut werden und verhalten sich daher annähernd ideal. Für die Gelenke ist dagegen ein solches ideales Verhalten in der Regel nicht realisierbar. Die Ursache dafür sind die Getriebe welche verwendet werden, um die Drehbewegung der Motoren ins Langsame zu übersetzen. Aufgrund der notwendigen hohen Übersetzung bei geringer Baugröße und hoher Last werden hier Planetengetriebe, Zykloidengetriebe oder Harmonic-Drive-Getriebe eingesetzt. Diese Getriebetypen besitzen jedoch eine relativ niedrige Steifigkeit in Drehrichtung, welche sich zudem konstruktiv schlecht beseitigen lässt. Als Auswirkung der niedrigen Steifigkeit der Getriebe in Bewegungsrichtung verhalten sich die Gelenke des Roboters nicht ideal, sondern weisen ein elastisches Verhalten auf, welches einen wesentlichen Einfluss auf die Bahngenauigkeit hat.

Eine attraktive Möglichkeit für den Umgang mit der bei Bahnapplikationen unerwünschten Elastizität ist der Einsatz regelungstechnischer Methoden. Mit der stetigen Steigerung der verfügbaren Rechenleistung können diese Methoden inzwischen auch auf komplexe technische Systeme wie die untersuchten Roboter angewandt werden, welchen aufwändige Modelle zugrunde liegen. Das Ziel dieser Arbeit ist deshalb der Entwurf einer praktikablen Regelungsstruktur für Industrieroboter, welche gute Bahnfolgeeigenschaften sicherstellt.

Aufgrund der großen Bedeutung von Robotern mit elastischen Gelenken entstanden zahlreiche Arbeiten, die sich mit der Regelung dieser Systeme beschäftigen. Der folgende Abschnitt gibt einen Überblick über die entstandene Literatur.

1.2 Stand der Technik

Die Literatur zur Regelung von elastischen Robotern kann recht gut in theoretisch und praktisch motivierte Arbeiten eingeteilt werden. In den theoretischen Arbeiten kommen dabei eine Vielzahl moderner Regelungsmethoden zum Einsatz, wobei fast ausschließlich entweder das sogenannte vollständige Modell oder das reduzierte Modell verwendet werden. Das vollständige Modell besteht aus der Mehrkörperdynamik des Roboters wie sie in Abschnitt 2.1.2 dargestellt ist, wobei der Antriebsstrang als lineare Feder angenommen wird. Beim auch als Spong-Modell bekannten reduzierten Modell wird bei der Modellierung der Mehrkörperdynamik zusätzlich die Bewegung der Motoren im Raum vernachlässigt. Auch dieser Fall ist in Abschnitt 2.1.2 angegeben. Für einen ersten Vergleich vorhandener Modellvorstellungen bei der Roboterregelung und eine Klassifizierung der Regelungsverfahren kann der Übersichtsartikel [SDO99] verwendet werden. Einen Überblick über die frühen Arbeiten zu Robotern mit elastischen Gelenken und ein sehr umfassendes Literaturverzeichnis liefert [Spo90]. Eine aktuellere Übersicht gibt [DT96], worin ebenfalls eine Fülle von Referenzen enthalten sind.

Einen Schwerpunkt der theoretischen Arbeiten bildet dabei die Feedbacklinearisierung. In [Spo87] wurde gezeigt, dass die Zustände des Spong-Modells mit statischem Feedback linearisierbar sind. Erste Ergebnisse zur Linearisierung des vollständigen Modells für spezielle Kinematiken lieferten [DIN85] und [De 88]. In [DL95] konnte die Linearisierbarkeit des vollständigen Modells mit dynamischem Feedback allgemein nachgewiesen werden. Schließlich wurde in [DL98] ein Algorithmus zur Zustandslinearisierung des vollständigen Modells mit dynamischem Feedback anhand der Bewegungsgleichungen gezeigt.

Ein weiterer Schwerpunkt wird von Arbeiten gebildet, welche den elastischen Roboter mit Hilfe der Theorie singulärer Störungen (singular perturbation) betrachten, wobei die Bewegung der Gelenke dem langsamen Teilsystem, die Bewegung der Motoren dagegen dem schnellen Teilsystem zugeordnet werden. Stellvertretend seien hier [KK85, SKK87, GS00] genannt.

Desweiteren existieren Stabilitätsbeweise für das Positionieren mit einer Reihe von einfachen Reglern, basierend auf unterschiedlichen Annahmen über die verfügbaren Messgrößen. Am häufigsten zitiert wird vermutlich die Arbeit [Tom91] in der gezeigt wird, dass für das Positionieren von elastischen Robotern mit Hilfe eines PD-Reglers der Motorposition globale asymptotische Stabilität erreicht werden kann. Dazu wird eine Gravitationskompensation auf Basis der Sollposition verwendet. Die Erweiterung auf eine Online-Kompensation ist in [DSZ05] dargestellt. Auch wenn der D-Anteil des Reglers aus [Tom91] durch eine Approximation erster Ordnung ersetzt wird, kann die Stabilität nachgewiesen werden [KOAL94]. Die globale asymptotische Stabilität des Positioniervorgangs basierend auf der Motorposition kann

entsprechend [AL96] auch mit linearen, zeitinvarianten Regler-Beobachter-Konfigurationen erreicht werden. Dies ist auch dann noch möglich, wenn das Modell um eine einfache Motordynamik ergänzt wird [ALLG97], wobei lineare Regler mit Rückführung der Motorzustände möglich sind. Für den Fall, dass nur die Gelenkwinkel gemessen werden, kann mit Hilfe des dynamischen linearen zeitvarianten Reglers in [BL95] die globale asymptotische Stabilität des Systems sichergestellt werden. Wird mit der Motorposition, den Gelenkmomenten und den Zeitableitungen dieser Größen der komplette Zustandsvektor gemessen, dann zeigt der Beweis in [ASH00] dass der Roboter mit einem linearen Zustandsregler für eine vorgegebene Position global stabilisiert werden kann. Wird der Regler um einen integralen Anteil erweitert, was z. B. dann sinnvoll ist, wenn die Momente aufgrund der Schwerkraft nicht exakt bekannt sind, dann kann in der Regel keine globale Stabilität mehr nachgewiesen werden. In [ARCB01] wird gezeigt, dass ein PD-Regler für die Motorposition und ein I-Regler für die Gelenkposition den Positioniervorgang asymptotisch stabilisieren können. Eine simulative Untersuchung zur Verbesserung des Positionierverhaltens durch eine geeignete Steuerung für einen planaren Arm mit zwei Gelenken wurde in [WA87] gegeben.

Wie aus der bisher angegebenen Literatur ersichtlich ist, lässt sich für das Positionieren die globale asymptotische Stabilität mit einfachen linearen Reglern erreichen. Dies ist für die Trajektorienfolge nicht mehr der Fall. Einen Vergleich der gebräuchlichsten Regler für globale Trajektorienfolge zeigt [BOL95]. Neben Reglern auf Basis der Feedbacklinearisierung sind dies vor allem Regler welche auf der Entkopplung des Systems in kaskadierte Teilsysteme [NT93] und den Passivitätseigenschaften des Systems beruhen. Dabei benutzen fast alle Regler zur Trajektorienfolge ausgiebige Modellberechnungen. So ist z. B. in [DZQD98] ein Regler gezeigt, welcher basierend auf exaktem Modellwissen bei Messung der Gelenkposition, eine global asymptotische Trajektorienfolge garantiert. Werden zusätzliche Störungen mit betrachtet, dann kann mit dem Regler aus [Tom94] unter Zuhilfenahme von Messungen der Motorposition und -geschwindigkeit und der Gelenkposition und -geschwindigkeit der Folgefehler beliebig beschränkt werden. Globale asymptotische Stabilität kann jedoch nicht mehr erzielt werden.

Gegenstand der theoretischen Untersuchungen ist in fast allen Fällen der Nachweis der Stabilität für den jeweiligen Regler. Die Güte der Regelung bleibt in diesen Betrachtungen jedoch meist außen vor. Bestenfalls sind zur Illustration Ergebnisse aus Simulationen mit einfachen Modellen gezeigt, welche jedoch keinen Vergleich zwischen den Methoden erlauben. Die von der praktischen Anwendung motivierten Arbeiten zeigen dagegen einen anderen Schwerpunkt. Hier stehen detailliertere Modelle der Antriebsstränge, beispielsweise für die Reibung, im Vordergrund. Ferner sind für die Realisierung der Regler gewöhnlich Beschränkungen bezüglich der verfügbaren Sensorik und Rechenleistung vorhanden.

Eine der ersten praktischen Arbeiten zu elastischen Robotern ist in [SG85] zu finden. Hier wird bereits die Problematik der Regelung von Robotern mit elastischen Gelenken aufgezeigt und ein detailliertes Antriebsmodell vorgestellt. In [KJ85] wird ein KUKA 250 Roboter mit Elastizität, Dämpfung, Lose und Reibung modelliert. Das Bahnfolgeverhalten kann dabei bereits mit Hilfe eines einfachen, auf Messungen der Motorposition und -geschwindigkeit und der Armposition und -geschwindigkeit basierenden Reglers verbessert werden. Die Regelung mit Hilfe von Beschleunigungssensoren, welche über Filter auf die Stellgröße zurückgekoppelt werden, wurde in [FKH83] an einem Motoman L10 Roboter experimentell untersucht. Experimente zur Anwendung eines Reglers basierend auf der Theorie singulärer Störungen für eine Achse werden in [Hun91] vorgestellt, wobei wiederum die Positionen und Geschwindigkeiten gemessen werden. Vergleiche zwischen mehreren nichtlinearen Regelungsansätzen am

zweiachsigen Laborroboter Capri zeigen [BRPB98] und [BR98]. Diese Arbeiten können als experimentelle Erprobung der Regler in [BOL95] verstanden werden. Umfangreiche Arbeiten zur Impedanzregelung von Robotern mit elastischen Gelenken sind in [AS02] zusammengefasst, wobei in Experimenten mit den DLR Leichtbaurobotern achsweise Zustandsregler auf Basis der Messung von Motorposition und Gelenkmoment zum Einsatz kommen. In [Ös02] wird für einen ABB IRB 1400 Roboter ein linearer LQG-Regler entworfen und evaluiert.

Alternativ zu den oben genannten Arbeiten in denen eine Verbesserung des Roboters durch Rückführung der verfügbaren Messwerte erfolgt, sind eine Reihe von Arbeiten entstanden die sich mit der Steuerung von Robotern mit elastischen Gelenken befassen. Dabei wurden verschiedene Ansätze verfolgt.

Ein einfaches, teilweise heuristisches Input-Shaping für einen sechsachsigen Hyundai Industrieroboter wird in [PCH99] gezeigt. Dabei wird im Wesentlichen die arbeitspunktabhängige Schwingungsdauer des Roboters geschätzt und für die Bahnplanung verwendet. In [Alb01] erfolgt die Steuerung mit linearen Übertragungsfunktionen, welche abhängig vom Arbeitspunkt und der Last umgeschaltet und um einige Nichtlinearitäten ergänzt werden. Die Experimente fanden an einem Reis RV 15 Roboter statt, wobei durch die Steuerung eine Korrektur der Sollposition erfolgte. In der gleichen Weise wurden die Korrekturen in [GH02] aufgeschaltet. Die Steuerung basiert in dieser Arbeit auf einem Starrkörpermodell des Roboters mit Reibung. Experimente wurden an einem Siemens manutec-r15 und einem KUKA KR15 durchgeführt. Ferner wird in [Lan03] eine adaptive sensorbasierte Steuerung vorgestellt, welche aus einem linearen Anteil und neuronalen Netzen besteht. Diese Steuerung ist in der Lage Abweichungen von der Bahn sensorgestützt zu erkennen und ihre Dynamik automatisch zu adaptieren.

Neben den Arbeiten, welche sich direkt mit Robotern mit elastischen Gelenken befassen, sind auch noch Arbeiten zur Regelung einzelner elastischer Antriebsstränge interessant. Hier seien stellvertretend [BG98] worin PI-Zustandsregler angewendet werden, die Analyse von LQG Reglern in [Gs01], [Faß02] für verschiedene lineare Regler und [Rau03] für die prädiktive Regelung mit lernfähigen Modellen genannt.

1.3 Eingesetzte Regelungsstruktur

Die vorhandene Literatur zeigt, dass Bahnfolgeregler ausgiebige Modellberechnungen verwenden. Dies ist auch naheliegend, da im Gegensatz zu Reglern für das Positionieren nicht nur das statische Haltemoment, sondern die komplette Dynamik des Roboters für die Erreichung des Regelziels relevant ist.

Die Modelle können dabei sowohl in der Steuerung als auch im Regler benutzt werden. Im Falle der bekannten Computed-Torque-Regelung, deren Erweiterung auf Roboter mit elastischen Gelenken die Feedbacklinearisierung darstellt, wird die Inverse des Modells im Regler direkt zur Entkopplung und Linearisierung der Dynamik verwendet. Aus praktischer Sicht hat dieses Vorgehen jedoch einige Nachteile. Neben der Notwendigkeit den kompletten Zustand zu messen ist dies insbesondere der extrem hohe Rechenzeitbedarf durch die Online-Auswertung des Modells im schnellen Regelungstakt. Daher wird die Inverse des Modells in der Praxis gerne in der Steuerung eingesetzt, was einer Linearisierung um die Solltrajektorie entspricht. Aus diesem Grund wird das Verfahren auch als Feedforward-Linearisierung bezeichnet. In der Literatur sind Arbeiten zu finden, welche sich mit dem experimentellen Vergleich verschiedener

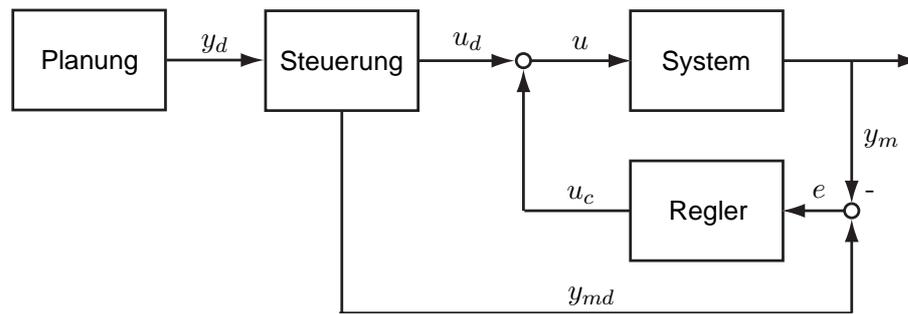


Abbildung 1.1: Regelung mit zwei strukturellen Freiheitsgraden

Varianten von Feedforward- und Feedbacklinearisierung beschäftigen. In [AAGH89] werden fünf Reglervarianten an einem Arm mit Direktantrieben verglichen. Für den CMU DD Arm II ist in [KK88] ein experimenteller Vergleich von Computed-Torque, Feedforward-Linearisierung und approximierter Feedforward-Linearisierung zu finden. Noch mehr Varianten mit unterschiedlich komplexen Modellen wurden in [LS89] für einen Puma Roboter untersucht. Für Roboter mit elastischen Gelenken zeigt [JE94] einen Vergleich von Feedforward- und Feedbacklinearisierung. Die Ergebnisse dieser Arbeiten können dabei wie folgt zusammengefasst werden: Die Bahnfehler werden umso geringer, je kompletter das zur Regelung verwendete Modell ist. Dabei macht es in Bezug auf Regelgüte und Robustheit keinen Unterschied, ob das Modell zur Feedforward- oder zur Feedbacklinearisierung verwendet wird. Damit ist der Einsatz des Modells in der Steuerung aus praktischer Sicht jedoch attraktiver, da hierfür nicht die Messung des kompletten Zustands notwendig ist und die Auswertung des Modells, welches für eine hohe Regelgüte möglichst vollständig verwendet werden sollte, nicht in Echtzeit erfolgen muss. Daher wird in dieser Arbeit eine Reglerstruktur eingesetzt, welche die Verwendung des Modells in der Steuerung erlaubt.

Problematisch ist bei den praktischen Arbeiten meist das Zusammenspiel von Steuerung und Regler. Ein Teil dieser Arbeiten konzentriert sich auf den Entwurf der Rückführung, verwendet aber dabei nur vergleichsweise rudimentäre Steuerungen. Andere Arbeiten beschäftigen sich ausschließlich mit dem Steuerungsentwurf. Dabei wird jedoch gewöhnlich eine Steuerung für den geregelten Roboter entworfen, was allein schon daran ersichtlich ist, dass die Wirkung der Steuerung auf einer Modifikation der Sollgelenkwinkel beruht. Damit ist es jedoch nicht mehr möglich Steuerung und Regler unabhängig voneinander zu entwerfen.

Um diesen Nachteil zu beseitigen, wird in der vorliegenden Arbeit eine Reglerstruktur mit zwei strukturellen Freiheitsgraden eingesetzt, deren Aufbau in Abbildung 1.1 dargestellt ist. Eine anschauliche Interpretation dieser Reglerstruktur anhand linearer SISO-Systeme liefert [Kre99]. Mit der Anwendung auf die Bahnfolgeregelung von nichtlinearen Systemen und den inhärenten Eigenschaften dieser Reglerstruktur setzt sich z. B. [van97] auseinander. Ganz allgemein bilden Steuerung und Regler die beiden Freiheitsgrade der Struktur. Dabei berechnet die Steuerung aus der Solltrajektorie des Ausgangs y_d den Soll-Eingang u_d und die Soll-Messgrößen y_{md} des Systems. Dies entspricht einer Feedforward-Linearisierung mit Hilfe der Inversen des zu regelnden Systems. Der Regler dagegen stabilisiert das System um die aktuelle Solltrajektorie. Durch die Feedforward-Linearisierung ist es insbesondere auch möglich vergleichsweise einfache, lineare Regler zu verwenden. Diese müssen dann lediglich in der Lage sein das System um die gewünschte Trajektorie zu stabilisieren.

Neben der direkten Linearisierung durch die Steuerung sind auch andere Maßnahmen möglich, um die Bahnengenauigkeit zu beeinflussen. Einen Vergleich mehrerer Methoden zur Trajektorienplanung für nichtlineare Systeme liefert z. B. [Koz03]. Für die Vermeidung von Schwingungen ist vor allem das sogenannte Input-Shaping bekannt. Dabei wird die Sollbahn mit einer Reihe von Impulsen, dem sogenannten Input-Shaper, gefaltet. Der Input-Shaper lässt sich somit auch als Filter mit endlicher Impulsantwort darstellen. Sogenannte Zero-Vibration-Input-Shaper entfernen dabei aus der Sollbahn alle Eigenfrequenzen der Strecke. Dies zeigt aber auch schon ein Problem der Methode auf. Bei stark nichtlinearen Systemen wie Robotern ändern sich die Frequenzen über den Arbeitsraum deutlich, weshalb der Input-Shaper angepasst werden muss. Ein Anwendungsbeispiel für Roboter mit elastischen Gelenken zeigt [KM97]. Ein weiterer Nachteil der Methode ist, dass sie sich nur für das Positionieren eignet, da die Sollbahn durch den Input-Shaper unter Umständen deutlich verändert wird. Als Vorteil kann dagegen angesehen werden, dass im Gegensatz zur Feedforward-Linearisierung kein besonders genaues Modell benötigt wird.

Für die Festlegung des Aufbaus der Steuerung sind ausserdem noch die Eigenschaften der Sollbahnen interessant, da bei der Auslegung einer Bahnfolgeregelung die Klasse der Bahnen, welchen der Regler folgen können soll, eine zentrale Rolle spielt. Wie in [GDLL94] dargestellt kann die Solltrajektorie verschiedene Eigenschaften besitzen:

Beliebige Trajektorie:

Die Trajektorie besitzt keine speziellen Eigenschaften

Die Trajektorie ist aus \mathcal{C}^N :

Die Trajektorie ist aus dem Raum der N -fach kontinuierlich differenzierbaren Funktionen $\mathcal{C}_m^N[0, \infty)$ mit $\sup_{t \geq 0} \{\|y_d(t)\|, \dots, \|y_d^{(N)}(t)\|\} < \epsilon$ für $N > 0$ und einem geeigneten $\epsilon > 0$.

Von einem nichtautonomen dynamischen System erzeugt:

Wenn die Trajektorie von einem nichtautonomen dynamischen System erzeugt wird, dann handelt es sich um das Problem der Modellfolgeregelung.

Von einem Exosystem erzeugt:

Die Trajektorie wird von einem autonomen System der Form

$$\dot{s} = f(s), \tag{1.1}$$

dem sogenannten Exosystem erzeugt. In diesem Fall spricht man von einem *Regulator* oder *Servomechanism problem* wie es z. B. in [IB90] beschrieben ist.

Für Anwendungen in der Robotik ist es gewöhnlich nicht möglich die Trajektorie durch ein Exosystem generieren zu lassen, da sich die Planungsalgorithmen für die Bahn nicht als Exosystem darstellen lassen. Theoretisch wäre es möglich eine Bahn stückweise aus Bahnen zusammenzusetzen, die von einem Exosystem generiert wurden. Dieses Vorgehen führt aber zu Transienten beim Umschalten des Exosystems. Dagegen werden immer gewisse Differenzierbarkeitseigenschaften der Bahn gefordert sein, allein schon um beschränkte Stellmomente sicherzustellen. So wird sich in Kapitel 5 zeigen, dass zur Berechnung der Feedforward-Linearisierung bestimmte Anforderungen bezüglich der Differenzierbarkeit der Solltrajektorien des Ausgangs erfüllt sein müssen. Je nach Planungsalgorithmus ist es aber unter

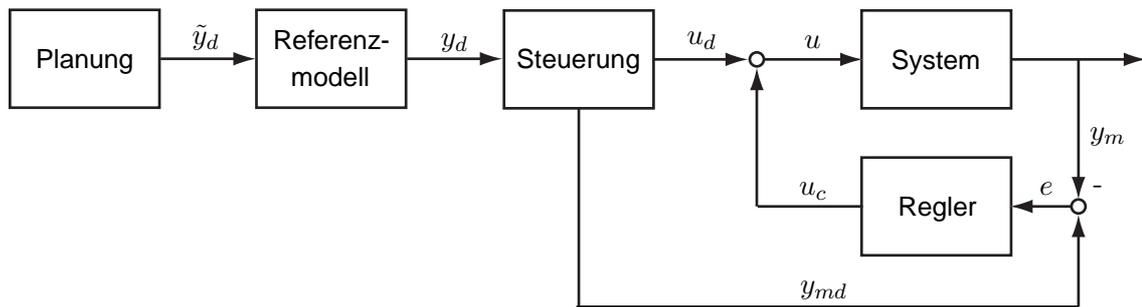


Abbildung 1.2: Regelung mit zwei strukturellen Freiheitsgraden und Referenzmodell

Umständen nicht möglich diese sicherzustellen, was insbesondere für zeitoptimale Bahnen gilt. Noch ungünstiger ist die Situation bei sensorgeführten Bahnen. Hier können beliebige Trajektorien entstehen, deren Ableitungen nur numerisch bestimmt werden können. Aus Sicht der Bahnplanung ist es damit wünschenswert, möglichst wenige Vorgaben für die Trajektorien erfüllen zu müssen. Im Folgenden wird deshalb davon ausgegangen, dass die Bahnplanung lediglich die Sollgelenkwinkel liefert, deren Verlauf vom Roboter aufgrund seiner mechanischen Beschränkungen allerdings auch realisierbar sein sollte. Die Steuerung muss dann die notwendigen Ableitungen zur Berechnung der Linearisierung selbst bestimmen.

Da das zur Linearisierung verwendete Modell sicherlich nicht ganz exakt sein wird, soll zusätzlich ein Input-Shaping der Sollbahnen durchgeführt werden, um die Anregung nicht modellierter Dynamik zu unterdrücken. Auf diese Weise können die Vorteile beider Methoden kombiniert werden: Bereiche, in denen bei der Linearisierung nicht modellierte Dynamiken wie z. B. Strukturelastizitäten auftreten, werden ohne detailliertes Modellwissen über diese Dynamiken durch das Input-Shaping unterdrückt. Die Güte des Modells in der Steuerung bestimmt somit die mögliche Bandbreite der Sollgrößen.

Wie bereits ausgeführt, kann die Modifikation der Sollgröße als Filter dargestellt werden. Durch geeignete Wahl des Filters ist es zudem möglich, auch die in der Steuerung benötigten Ableitungen der Sollgröße zu berechnen. Insofern stellt die Kombination aus Filter und Feedforward-Linearisierung in der Steuerung eine praktikable Lösung dar. Die Struktur des Reglers erweitert sich damit, wie in Abbildung 1.2 dargestellt, um die Filterung der Sollgröße. Das Filter kann dabei auch als Referenzmodell für die Dynamik des Roboters angesehen werden, was einer Modellfolgeregelung entspricht. Die Erzeugung der Solltrajektorie für den Ausgang ist nicht Teil dieser Arbeit. Für Simulationen und die experimentelle Erprobung werden an dieser Stelle übliche Planungsverfahren für starre Roboter eingesetzt. Damit lassen sich z. B. zeitoptimale Bahnen für das Positionieren und die Trajektorien der Gelenke für vorgegebene geometrische Bahnen erzeugen.

Die Stabilisierung des Systems um die erzeugte Trajektorie ist Aufgabe des Reglers. Hier werden einfache Regler, wie sie vom Positionieren bekannt sind, eingesetzt, weshalb ein solches Vorgehen auch als *Nonlinear Regulation* bezeichnet wird. Neben der Stabilisierung der beim Roboter instabilen Systemdynamik müssen die Regler zusätzlich für eine günstige Eigendynamik des geregelten Systems sorgen, um Modellfehler in der Feedforward-Linearisierung ausgleichen zu können. Da die Dynamik von Robotern stark vom aktuellen Arbeitspunkt abhängt ist es zudem sinnvoll, die Reglerparameter an den Arbeitspunkt anzupassen. Dieses *Gain-Scheduling* ermöglicht eine bessere Eigendynamik der geregelten Strecke, da der Regler

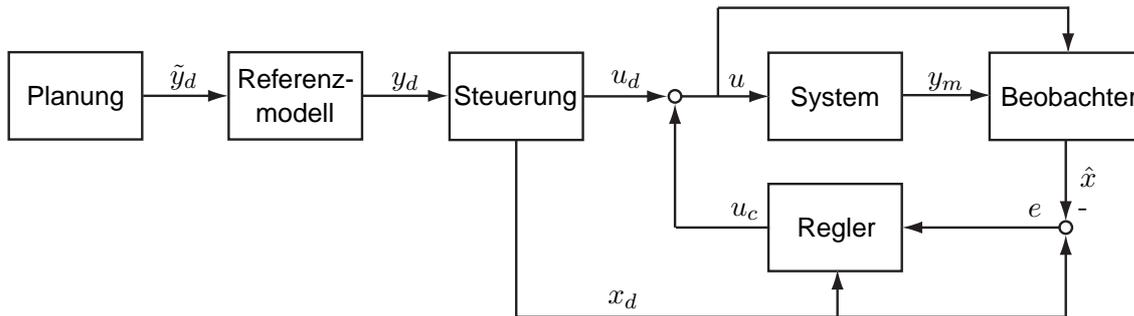


Abbildung 1.3: Regelungsstruktur mit Referenzmodell und Beobachter

nicht mehr robust bezüglich aller möglichen Arbeitspunkte ausgelegt werden muss.

Aus der Literatur zu Robotern mit elastischen Gelenken und Arbeiten zur Regelung von elastischen Antriebssträngen ist bekannt, dass für eine gute Eigendynamik des geregelten Systems die Kenntnis der Zustände von Motoren und Gelenken wichtig ist. An üblichen kommerziellen Robotern wie auch dem betrachteten KR15/2 stehen jedoch nur Messungen auf der Motorseite zur Verfügung, da diese mit geringerem Aufwand möglich sind. Die nachträgliche Erweiterung um Messungen der Gelenkgrößen ist dagegen in der Praxis mit einigen Problemen verbunden.

Da sich Sensoren, wie optische Drehgeber oder Momentensensoren, im Gelenk kaum mehr nachrüsten lassen, stellen inertielle Sensoren eine elegante Möglichkeit zur Bestimmung der Gelenkgrößen dar, da sie einfach nachträglich am Arm befestigt werden können. Der Nachteil dieser Sensoren ist, dass die gemessenen linearen Beschleunigungen und Drehgeschwindigkeiten zur Bestimmung der Gelenkgrößen erst noch transformiert und integriert werden müssen. Dies ist das Problem der Inertialnavigation. Eine Kombination aus 3 orthogonalen Beschleunigungssensoren und 3 Drehratensensoren an einem Punkt des Armes bildet ein sogenanntes *strap-down* System und ermöglicht durch Integration der Sensorwerte die Bestimmung von Position und Orientierung des Punktes im Raum. Um die Fehler bei der Integration durch Sensordrift und -rauschen klein zu halten, werden hochwertige Sensoren benötigt. Bei längeren Messungen ist auch dies nicht ausreichend. Dann muss die Integration durch eine zusätzliche Positionsmessung gestützt werden. Ein solches System wird z. B. in [Die95] und [Alb01] benutzt. Da Roboter jedoch bereits über Positionssensoren an den Motoren verfügen wird in dieser Arbeit ein anderer Ansatz verfolgt. Die Dynamik der Gelenkbewegung wird mit Hilfe von am Arm verteilten Beschleunigungssensoren gemessen. Basierend auf den Messungen der Beschleunigungs- und Motorsensoren wird ein Beobachter entworfen, welcher die Zustände des Roboters schätzt. Unter der Annahme, dass es möglich ist den Beobachter so schnell auszulegen, dass die geschätzten Zustände quasi-ideal sind, kann die Kombination aus System und Beobachter für den Entwurf von Steuerung und Regler als idealisiertes System mit vollständiger Zustandsmessung angesehen werden. Damit ergibt sich die um den Beobachter und das Gain-Scheduling erweiterte Regelungsstruktur, welche in Abbildung 1.3 dargestellt ist.

1.4 Beitrag und Organisation der Arbeit

Mit steigenden Anforderungen an die Bahngenauigkeit ergibt sich auch die Notwendigkeit zur Verwendung detaillierterer Modelle. Hierbei werden häufig Black-Box-Modelle eingesetzt, welche z. B. durch Identifikation entsprechender linearer Systeme oder den Einsatz allgemeiner neuronaler Netze entstehen. Für stark nichtlineare Systeme, bei denen zudem die physikalischen Zusammenhänge bekannt sind, eignen sich solche Modelle jedoch weniger. Zum Einen ist meist eine Adaption der Modelle an den aktuellen Arbeitspunkt notwendig, zum Anderen kann die Generalisierungsleistung für nicht optimierte Bereiche des Arbeitsraumes sehr schlecht sein. Ein wesentlicher Beitrag der Arbeit ist daher die Entwicklung einer neuen detaillierten, physikalisch motivierten Modellvorstellung für Roboter mit elastischen Gelenken, wobei sowohl die Dynamik des Mehrkörpersystems als auch die Dynamik der Antriebsstränge berücksichtigt werden. Die Modellvorstellungen für die Mehrkörperdynamik, die Antriebe und das Gesamtmodell werden in Kapitel 2 motiviert und entwickelt. Ferner werden die Teilm Modelle als Modelica-Bibliothek realisiert, um Simulationen durchführen zu können.

Nachdem das Ziel der Arbeit eine in der Praxis einsetzbare Methode zur Regelung von Robotern mit elastischen Gelenken ist, soll deren Anwendung auch an einem praktisch relevanten Roboter gezeigt werden. Dazu wird ein Kuka KR15/2 verwendet. Die Modellierung dieses Roboters und die Identifikation der zugehörigen Modellparameter erfolgt in Kapitel 3. Dabei wird zur Identifikation der Parameter eine Kombination aus speziellen Versuchen zur möglichst direkten Bestimmung einzelner Parameter und Optimierungsmethoden für nichtlineare Systeme verwendet.

Wie in Kapitel 4 gezeigt, wird zur Berechnung der Solltrajektorien und -zustände in der Steuerung die Inverse des zu regelnden Systems benötigt. Da bei dieser Berechnung das komplette Robotermodell benutzt wird, spielt die Inverse eine zentrale Rolle für die erreichbare Güte der Regelung und bildet das Kernstück der Regelungsstruktur. Dem entsprechend liegt der Schwerpunkt der Arbeit auf der Berechnung von inversen Systemen und der Invertierung der Roboterdynamik welche in Kapitel 4 und Kapitel 5 behandelt werden. Dabei werden Möglichkeiten zur Erzeugung und Auswertung der Inversen von komplexen nichtlinearen Systemen untersucht. Diese Problematik wird in der Literatur kaum behandelt, stellt aber den Schlüssel für die praktische Anwendung von inversen Systemen dar. Insbesondere wird eine neue Methode zur Berechnung von inversen Systemen gezeigt, welche auf bekannten Algorithmen zur Behandlung von DAE-Systemen mit höherem Index basiert.

Mit Hilfe dieser Methode wird in Kapitel 5 das in Kapitel 2 entwickelte Robotermodell invertiert. Dazu wird nicht, wie sonst üblich, eine Zustandsraumdarstellung verwendet. Statt dessen wird gezeigt, dass eine direkte Analyse der Bewegungsgleichungen eine intuitivere Invertierung erlaubt. Da das Modell, welches in Kapitel 2 erstellt wurde, nicht invertierbar ist, wird ein neues, approximiertes Modell zur Invertierung aufgestellt. Ferner zeigt die Analyse der Bewegungsgleichungen weitere interessante Eigenschaften des Modells. So kann insbesondere ein neuer Algorithmus zur Bestimmung der notwendigen Ableitungen der Bewegungsgleichungen angegeben werden. Anschließend wird eine Stabilitätsanalyse des inversen Robotermodells durchgeführt und Bedingungen für die Stabilität angegeben. Schließlich wird die inverse Dynamik des KR15/2 mit den beschriebenen Methoden berechnet.

In Kapitel 6 wird zunächst der Beobachter entworfen. Dabei wird ausgenutzt, dass die Komplexität des für die Schätzung notwendigen Modells durch zusätzliche Verwendung von Be-

schleunigungssensoren deutlich reduziert werden kann. Zunächst wird dazu eine allgemeine Methode zur Transformation der gemessenen Beschleunigungen auf Gelenkbeschleunigungen abgeleitet. Basierend darauf wird ein neuer „High-Gain“-Beobachter entworfen, welcher lediglich ein exaktes Modell des Antriebs benötigt, um unabhängig von externen Störmomenten die Konvergenz der Schätzwerte gegen den echten Zustand sicherzustellen. Dieser wird für die ersten drei Achsen des KR15/2 ausgelegt und evaluiert.

Basierend auf dem geschätzten Zustand und linearen Achsmodellen wird anschließend der Regler entworfen. Dabei wird gezeigt, dass durch geeignete, achsweise PI-Zustandsregler volle Entwurfsfreiheit bei gleichzeitiger Einhaltung der gewünschten stationären Genauigkeit erzielt werden kann. Um der Veränderung der Roboterdynamik über den Arbeitsraum Rechnung zu tragen, wird zusätzlich ein Gain-Scheduling der Reglerparameter entworfen. Dabei wird eine Möglichkeit gezeigt, die Pole der Gelenkmodelle indirekt, anhand von Steifigkeits- und Dämpfungsvorgaben in Abhängigkeit des Arbeitspunktes zu berechnen. Anschließend werden die Regler für die ersten drei Achsen des KR15/2 ausgelegt und die praktisch erreichbare Eigendynamik des geregelten Roboters untersucht.

In Kapitel 7 wird die entworfene Reglerstruktur experimentell an einem Kuka KR15/2 erprobt. Dafür werden zunächst Kriterien zur Bewertung der Bahnabweichungen und des Positionierverhaltens aufgestellt. Die Feineinstellung der Reglerparameter erfolgt mit Hardware-in-the-loop-Optimierungen, welche eine neue Methode zur Auslegung von Bahnfolgereglern für Roboter darstellen. Schließlich wird der optimierte neue Regler bezüglich seiner Positioniereigenschaften, Bahntreue und Robustheit experimentell untersucht und mit dem Originalregler des Roboters verglichen.

Kapitel 2

Modellierung der Roboterdynamik

In diesem Kapitel wird das Modell des Roboters entwickelt. Das Ziel dieser Entwicklung ist ein regelungstechnisches Modell, das heißt ein Modell, welches sich aufgrund seiner Komplexität für den regelungstechnischen Entwurf eignet. Dazu wird ein physikalisch motiviertes Modell benutzt, bei welchem physikalische Eigenschaften des Roboters eine möglichst direkte Entsprechung im Modell finden.

Dabei werden nur Roboter betrachtet, deren Strukturelemente als starr angenommen werden können. Diese Annahme trifft für viele Roboter und insbesondere für Industrieroboter wie den untersuchten KUKA KR15/2 [KUK96b] zu, dessen Glieder aus Aluminiumguss praktisch starr sind. Bedingt durch die bei Robotern eingesetzten Getriebe mit hoher Übersetzung bei gleichzeitig geringer Baugröße, wie z. B. Harmonic Drive und Zykloidengetriebe, sind die Gelenke jedoch in Drehrichtung elastisch. Die Steifigkeit senkrecht zur Drehrichtung ist dagegen hoch. Roboter, die diese Eigenschaften erfüllen, werden in der Literatur als *Rigid Link Flexible Joint* (RLFJ) Roboter bezeichnet.

Da während des Regelungsentwurfs die Modelleigenschaften ausgiebig benutzt werden, wird zunächst eine Herleitung des Mehrkörpermodells gegeben. Diese orientiert sich an [DT96, Spo87, Rea94], verwendet aber ein etwas allgemeineres Modell. Neben der hoch nichtlinearen Mehrkörperdynamik spielt die Dynamik des Antriebsstranges eine entscheidende Rolle, weshalb anschließend ein detailliertes Antriebsstrangmodell entwickelt wird.

2.1 Mehrkörperdynamik

Für die Herleitung der Bewegungsgleichungen eines Mehrkörpersystems (MKS) können sowohl der Lagrange Formalismus als auch der Newton-Euler Formalismus verwendet werden [Sil82]. Ersterer hat den Vorteil, dass er wichtige strukturelle Eigenschaften des Modells liefert, ohne dass es notwendig ist, die Gleichungen für den Roboter explizit aufzustellen. Der Newton-Euler Formalismus ist dagegen für Systeme mit einer größeren Anzahl von Achsen effizienter. Eine Analyse von Robotern mit elastischen Gelenken auf Basis des Newton-Euler Formalismus liefert z. B. [MWS90]. Da vorerst vor allem die strukturellen Eigenschaften des Modells interessieren, werden im Folgenden die Bewegungsgleichungen des MKS-Teils des Robotermodells über den Lagrange Formalismus hergeleitet.

2.1.1 Lagrange Formalismus

Beim Lagrange Formalismus werden, ausgehend von der Lagrange-Funktion

$$\mathcal{L} = \mathcal{T}(\Theta, \dot{\Theta}) - \mathcal{U}(\Theta), \quad (2.1)$$

welche die Differenz zwischen der kinetischen Energie $\mathcal{T}(\Theta, \dot{\Theta})$ und der potentiellen Energie $\mathcal{U}(\Theta)$ des MKS in den verallgemeinerten Koordinaten $\Theta \in \mathbb{R}^n$ beschreibt, mit Hilfe der Lagrange-Gleichungen 2. Art

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\Theta}_i} \right) - \frac{\partial \mathcal{L}}{\partial \Theta_i} = f_i, \quad i = 1 \dots n \quad (2.2)$$

die Bewegungsgleichungen aufgestellt. Dabei ist f der Vektor der externen Kräfte über den insbesondere auch nichtkonservative Kräfte wie z. B. Reibung berücksichtigt werden können. Unter Verwendung der Tatsache, dass sich die kinetische Energie mit der Massenmatrix $M(\Theta)$ immer als quadratische Funktion der verallgemeinerten Geschwindigkeiten $\dot{\Theta}$ schreiben lässt,

$$\mathcal{T}(\Theta, \dot{\Theta}) = \frac{1}{2} \dot{\Theta}^T M(\Theta) \dot{\Theta} \quad (2.3)$$

ergibt sich durch Einsetzen und Auswerten der Gleichungen (2.1) und (2.3) in (2.2) für die Bewegungsgleichungen der folgende Aufbau:

$$M(\Theta)\ddot{\Theta} + C(\Theta, \dot{\Theta})\dot{\Theta} + g(\Theta) = f \quad (2.4)$$

$$C(\Theta, \dot{\Theta})\dot{\Theta} = \dot{M}(\Theta)\dot{\Theta} - \frac{1}{2} \left(\frac{\partial}{\partial \Theta} (\dot{\Theta}^T M(\Theta) \dot{\Theta}) \right)^T \quad (2.5)$$

$$g(\Theta) = \left(\frac{\partial \mathcal{U}(\Theta)}{\partial \Theta} \right)^T \quad (2.6)$$

Dabei ist $C(\Theta, \dot{\Theta})$ die Matrix der Coriolis- und Zentrifugalkräfte und $g(\Theta)$ der Vektor der Potentialkräfte. Im Folgenden werden nun die Bewegungsgleichungen für den Spezialfall eines RLFJ Roboters hergeleitet.

2.1.2 Modellierung des Mehrkörpersystems

In dieser Arbeit werden Roboter mit starren Gliedern und elastischen Gelenken betrachtet, deren Kinematik eine offene Kette ist. Die Gelenke und Glieder des Roboters werden von der Basis zum Werkzeugflansch durchnummeriert, mit dem Index 0 für die Basis. Die Gelenke haben einen mechanischen Freiheitsgrad und die Anzahl der Gelenke und Freiheitsgrade wird mit N bezeichnet. Daraus resultiert ein Aufbau des Roboters wie in Abbildung 2.1 dargestellt. Jedem Glied i wird ein Frame G_i zugeordnet, dessen z-Achse der Drehachse des Gelenkes i entspricht. Ansonsten ist die Lage von G_i nicht weiter vorgegeben und kann z. B. entsprechend Denavit-Hartenberg [DH55] oder, wie in 2.1 dargestellt, im Gelenk gewählt werden.

Da die Motoren elastisch angekoppelt sind, bringen sie zusätzliche Freiheitsgrade ins System, weshalb zur Beschreibung des Systems $2N$ verallgemeinerte Koordinaten benötigt werden. Für die Herleitung werden die Koordinaten

$$\Theta = \begin{pmatrix} q \\ \theta \end{pmatrix} \in \mathbb{R}^{2N}$$

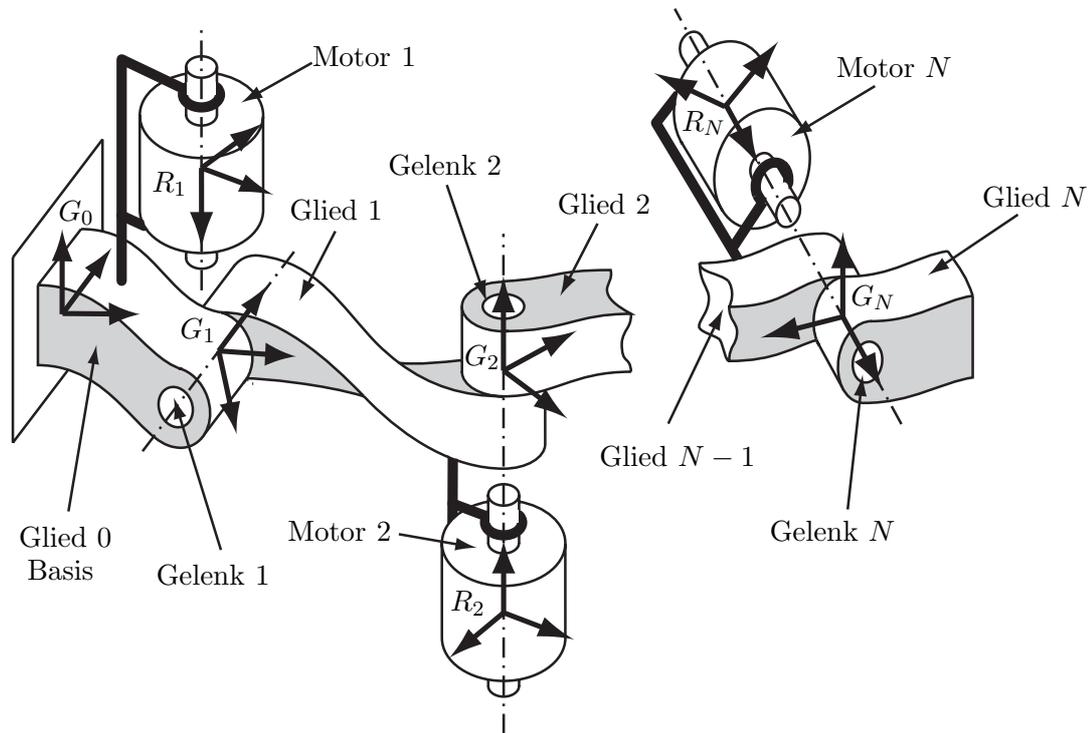


Abbildung 2.1: Aufbau des Roboters

verwendet, wobei q der $N \times 1$ Vektor der Gelenkpositionen und θ der $N \times 1$ Vektor der Rotorpositionen ist.

Die Antriebe seien rotatorisch und an den Gliedern angebracht. Der Vektor b beschreibt, an welchen Gliedern die Motoren befestigt sind, $b_i = j$ bedeutet, dass der Motor i an Glied j montiert ist. Um die bewegten Massen klein zu halten, sind die Motoren praktisch immer vor dem Gelenk befestigt, das heißt:

$$b_i < i, \quad \forall i \in \{1, \dots, N\} \quad (2.7)$$

Im Folgenden werden nur Roboter betrachtet, welche diese Bedingung erfüllen. In der Literatur wird durchgängig der Fall

$$b = (0 \ 1 \ \dots \ N-1) \quad (2.8)$$

betrachtet. Zur Vereinfachung der Modellierung wird noch eine in der Literatur übliche Annahme getroffen.

Annahme 2.1. Die Rotoren der Motoren seien rotationssymmetrisch bezüglich der Drehachsen der Motoren. Ihre Schwerpunkte liegen auf den jeweiligen Drehachsen.

Diese Annahme trifft in sehr guter Näherung auf die Motoren von Robotern zu und ist sehr realistisch. Zur Beschreibung des MKS wird nun noch dem Rotor jedes Motors i ein Frame R_i zugeordnet. Der Ursprung dieses Frames befindet sich im Schwerpunkt des Rotors, und seine Achsen entsprechen den Hauptträgheitsachsen des Rotors, wobei die z-Achse wiederum identisch mit der Drehachse des Motors ist.

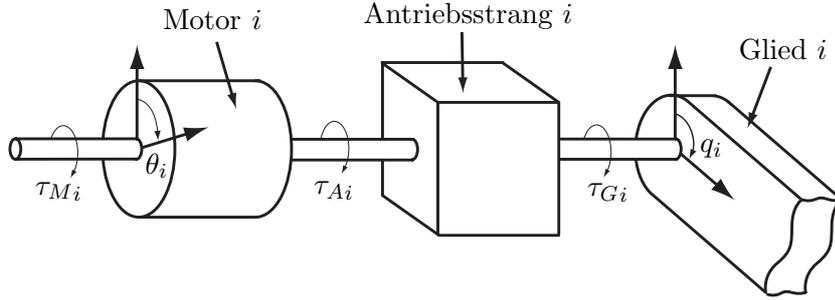


Abbildung 2.2: Modell eines Antriebsstranges zur MKS-Modellierung

In der Literatur wird das Antriebsstrangmodell gewöhnlich bereits beim Aufstellen der Lagrange-Funktion berücksichtigt. Dieses Vorgehen eignet sich gut für einfache Antriebsmodelle, welche z. B. nur aus einer linearen Elastizität bestehen. Für detailliertere Modelle ist eine Trennung der Modellierung von MKS und Antriebsstrang sinnvoll. Dazu werden für die Modellierung des MKS nur die Glieder und Rotoren des Roboters betrachtet. Die Kopplungen aufgrund der Antriebe werden dabei als externe Kräfte angesehen, deren Gleichungen später mit Hilfe der Antriebsstrangmodelle bestimmt werden. Dies ist auf einfache Weise möglich, da in den Antrieben nur eindimensionale, direkt zuzuordnende Größen auftreten. Damit ergibt sich für die Modellierung des MKS ein Achsmodell wie in Abbildung 2.2 dargestellt. Dabei bezeichnet $\tau_G \in \mathbb{R}^N$ den Vektor der auf die Glieder ausgeübten Gelenkmomente, $\tau_A \in \mathbb{R}^N$ den Vektor der von den Antriebssträngen auf die Rotoren ausgeübten Momente und $\tau_M \in \mathbb{R}^N$ den Vektor der aufgrund elektrischer Anregung auf die Rotoren wirkenden Momente.

2.1.3 Mehrkörperdynamik von Robotern mit elastischen Gelenken

Zunächst wird die Lagrange-Funktion für den Roboter

$$\mathcal{L} = \mathcal{T}(q, \dot{q}, \theta, \dot{\theta}) - \mathcal{U}(q, \theta) \quad (2.9)$$

aufgestellt, wobei $\mathcal{T}(q, \dot{q}, \theta, \dot{\theta})$ die kinetische Energie und $\mathcal{U}(q, \theta)$ die potentielle Energie des Roboters ist.

Die kinetische Energie setzt sich aus der Energie in den Gliedern und der Energie in den Motoren zusammen. Da die Statoren der Motoren fest mit den Gliedern verbunden sind, werden sie als Teile dieser angesehen und in $\mathcal{T}_{Glieder}$ berücksichtigt. Damit kann die gesamte kinetische Energie auch angegeben werden als:

$$\mathcal{T} = \mathcal{T}_{Glieder} + \sum_{i=1}^N \mathcal{T}_{Rotor_i} \quad (2.10)$$

Die kinetische Energie des Rotors i , ausgedrückt im Rotorframe R_i ist

$$\mathcal{T}_{Rotor_i} = \frac{1}{2} m_{R_i} {}^{R_i}v_{R_i}^T {}^{TR_i}v_{R_i} + \frac{1}{2} {}^{R_i}\omega_{R_i}^T {}^{TR_i}I_{R_i} {}^{R_i}\omega_{R_i}, \quad (2.11)$$

wobei ${}^{R_i}v_{R_i}$ die absolute Geschwindigkeit des Rotorschwerpunktes im Raum, ${}^{R_i}\omega_{R_i}$ die Winkelgeschwindigkeit, m_{R_i} die Rotormasse und ${}^{R_i}I_{R_i}$ der Trägheitstensor des Rotors bezüglich

seines Schwerpunktes ist. Mit Annahme 2.1 ist der im Motorkoordinatensystem angegebene Tensor ${}^{R_i}I_{R_i}$ bereits in Hauptachsenform und kann deshalb als

$${}^{R_i}I_{R_i} = \text{diag}({}^{R_i}I_{R_{i}xx}, {}^{R_i}I_{R_{i}yy}, {}^{R_i}I_{R_{i}zz})$$

dargestellt werden. Außerdem ist unter Annahme 2.1 die Massenverteilung des Roboters unabhängig von den Motorpositionen θ , womit der erste Summand von Gleichung (2.11) nur von den Koordinaten der Roboterglieder abhängt. Deshalb wird dieser Teil der kinetischen Energie ebenfalls den Gliedern zugeschlagen. Es verbleibt die Rotation des Rotors um seine Hauptachsen. Die für den Motor i zu berücksichtigende kinetische Energie ist damit

$$\mathcal{T}_{Rotor_i} = \frac{1}{2} {}^{R_i}\omega_{R_i}^T {}^{R_i}I_{R_i} {}^{R_i}\omega_{R_i}. \quad (2.12)$$

Zur Berechnung der Winkelgeschwindigkeiten ${}^{R_i}\omega_{R_i}$ kann zunächst eine Vorwärtsrekursion für die Winkelgeschwindigkeiten der Glieder ${}^{G_i}\omega_{G_i}$ verwendet werden

$${}^{G_i}\omega_{G_i} = {}^{G_i}R^{G_{i-1}}(q_i) {}^{G_{i-1}}\omega_{G_{i-1}} + (0 \ 0 \ \dot{q}_i)^T, \quad {}^{G_0}\omega_{G_0} = 0_{3 \times 1}, \quad (2.13)$$

wobei ${}^{G_i}R^{G_{i-1}}(q_i)$ die Rotationsmatrix zwischen den Frames von Glied i und $i-1$ ist. Basierend darauf kann nun ${}^{R_i}\omega_{R_i}$ mit

$${}^{R_i}\omega_{R_i} = {}^{R_i}R^{G_{bi}} {}^{G_{bi}}\omega_{G_{bi}} + (0 \ 0 \ \dot{\theta}_i)^T \quad (2.14)$$

berechnet werden, wobei ${}^{R_i}R^{G_{bi}}$ die Rotationsmatrix zwischen dem Frame des Gliedes an welchem der Motor montiert ist und dem Rotorframe ist. Dabei muss die Drehachse des Rotors nicht koaxial mit der Drehachse des angetriebenen Gelenkes sein.

Bei der Berechnung der kinetischen Energie der Glieder wird ausgenutzt, dass sich deren kinetische Energie als quadratische Funktion der Gelenkgeschwindigkeiten \dot{q} mit der Massenmatrix der Glieder $M_G(q)$ darstellen lässt.

$$\mathcal{T}_{Glieder} = \frac{1}{2} \dot{q}^T M_G(q) \dot{q} \quad (2.15)$$

Um die strukturellen Eigenschaften des Modells zu erkennen, ist es dabei nicht notwendig, $M_G(q)$ explizit aufzustellen.

Die potentielle Energie des Systems entspricht der Energie im Schwerfeld $\mathcal{U}_{Grav}(q)$.

$$\mathcal{U} = \mathcal{U}_{Grav}(q) \quad (2.16)$$

$\mathcal{U}_{Grav}(q)$ muss wiederum nicht explizit aufgestellt werden. Es wurde allerdings bereits berücksichtigt, dass aufgrund von Annahme 1 die Massenverteilung des Roboters und damit auch die Energie im Schwerfeld unabhängig von der Motorposition θ ist. Durch Einsetzen der Gleichungen (2.10), (2.12), (2.15) und (2.16) in (2.9) erhält man als Lagrange-Funktion des Roboters

$$\mathcal{L} = \frac{1}{2} \dot{q}^T M_G(q) \dot{q} + \sum_{i=1}^N \frac{1}{2} {}^{R_i}\omega_{R_i}^T {}^{R_i}I_{R_i} {}^{R_i}\omega_{R_i} - \mathcal{U}_{Grav}(q). \quad (2.17)$$

Die Lagrange-Gleichungen (2.2) können nun angeschrieben werden als

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} = \tau_G + \tau_{Ext} \quad (2.18a)$$

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}} \right) - \frac{\partial \mathcal{L}}{\partial \theta} = \tau_A + \tau_M \quad (2.18b)$$

wobei τ_{Ext} der Vektor extern aufgebrachtener Momente, z. B. durch Prozesskräfte, ist und die Momente τ_G , τ_A und τ_M entsprechend Abbildung 2.2 definiert sind. Einsetzen und Auswerten von Gleichung (2.17) in (2.18a) und (2.18b) liefert die Bewegungsgleichung des MKS

$$\begin{pmatrix} M_{Arm}(q) & S(q) \\ S(q)^T & J \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\theta} \end{pmatrix} + \begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0_{N \times 1} \end{pmatrix} - \begin{pmatrix} \tau_G \\ \tau_A \end{pmatrix} = \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix} \quad (2.19)$$

mit der Massenmatrix der Roboterglieder $M_{Arm}(q)$, der Kopplungsmatrix zwischen den Gliedern und Motoren $S(q)$, der Matrix der Motorträgheiten J , den Vektoren der Coriolis- und Zentrifugalkräfte auf den Arm $c_{Arm}(q, \dot{q}, \dot{\theta})$ und die Motoren $c_{Motor}(q, \dot{q})$ und dem Schwerkraftvektor $g_{Arm}(q)$. Die relevanten Eigenschaften und der Aufbau der Matrizen in den Bewegungsgleichungen (2.19) werden im Folgenden hergeleitet.

Terme, in denen die Beschleunigung auftritt, können gemäß den Gleichungen (2.18a) und (2.18b) nicht aus den partiellen Ableitungen der Lagrangefunktion nach einer verallgemeinerten Koordinate kommen. Die Matrix J kann deshalb durch Betrachtung des Terms

$$\frac{d}{dt} \left(\frac{\partial \sum_{i=1}^N \mathcal{T}_{Rotor_i}}{\partial \dot{\theta}} \right) \quad (2.20)$$

bestimmt werden, da die Motorgeschwindigkeiten nur in der kinetischen Energie der Rotoren auftauchen. Einsetzen der Gleichungen (2.12) und (2.14) in (2.20) liefert

$$\frac{d}{dt} \left(\frac{\partial \left(\sum_{i=1}^N (0 \ 0 \ \dot{\theta}_i)^{R_i} I_{R_{i,zz}}^{R_i} R^{G_{b_i} G_{b_i}} \omega_{G_{b_i}}(q, \dot{q}) + \frac{1}{2} I_{R_{i,zz}} \dot{\theta}_i^2 \right)}{\partial \dot{\theta}} \right)$$

wobei Terme, die unabhängig von $\dot{\theta}$ sind, bereits entfernt wurden. Man erkennt, dass nur im letzten Term, in dem $\dot{\theta}$ quadratisch vorkommt, $\dot{\theta}$ beim Differenzieren nicht verschwindet und somit einen Beitrag zur Motorträgheitsmatrix leisten kann. Damit ergibt sich für J die Form

$$J = \text{diag}(I_{1,zz}, \dots, I_{N,zz}). \quad (2.21)$$

Äquivalent kann die Matrix $S(q)$ durch Betrachtung von

$$\frac{d}{dt} \left(\frac{\partial \sum_{i=1}^N \mathcal{T}_{Rotor_i}}{\partial \dot{q}} \right) \quad (2.22)$$

bestimmt werden. Durch Einsetzen von Gleichung (2.12) und Gleichung (2.14) in (2.22) ergibt sich

$$\begin{aligned} & \frac{d}{dt} \left(\frac{\partial \sum_{i=1}^N \frac{1}{2} (R_i R^{G_{b_i} G_{b_i}} \omega_{G_{b_i}}(q, \dot{q}))^T R_i I_{R_i}^{R_i} R^{G_{b_i} G_{b_i}} \omega_{G_{b_i}}(q, \dot{q})}{\partial \dot{q}} \right) + \\ & \frac{d}{dt} \left(\sum_{i=1}^N (0 \ 0 \ \dot{\theta}_i)^{R_i} I_{R_i}^{R_i} R^{G_{b_i}} \frac{\partial^{G_{b_i}} \omega_{G_{b_i}}(q, \dot{q})}{\partial \dot{q}} \right) \end{aligned} \quad (2.23)$$

wobei Terme, die unabhängig von \dot{q} sind, bereits entfernt wurden. Da nur der zweite Summand im Ausdruck (2.23) von θ abhängig ist und damit zu S beiträgt, ergeben sich die Elemente der S Matrix zu

$$S_{j,i} = {}^{R_i}I_{R_{i,zz}} ({}^{R_i}R^{G_{b_i}})_3 \frac{\partial {}^{G_{b_i}}\omega_{G_{b_i}}(q, \dot{q})}{\partial \dot{q}_j}. \quad (2.24)$$

Die Winkelgeschwindigkeiten der Glieder aus der Rekursion in Gleichung (2.13) können auch ausgeschrieben angegeben werden als

$$\begin{aligned} {}^{G_i}\omega_{G_i} &= \sum_{j=1}^i {}^{G_i}R^{G_j} \begin{pmatrix} 0 & 0 & \dot{q}_j \end{pmatrix}^T \\ &= \sum_{j=1}^i ({}^{G_{b_i}}R^{G_j^T})_3 \dot{q}_j \end{aligned}$$

Aus dieser Gleichung wird ersichtlich, dass nur ein Term in der Summe vom jeweiligen \dot{q}_j aus Gleichung (2.24) abhängig sein kann und auch dieser nur auftritt wenn $b_i \geq j$. Damit können die Elemente von S auch dargestellt werden als:

$$S_{j,i} = \begin{cases} {}^{R_i}I_{R_{i,zz}} ({}^{R_i}R^{G_{b_i}})_3 ({}^{G_{b_i}}R^{G_j^T})_3 & \text{für } b_i \geq j \\ 0 & \text{sonst} \end{cases} \quad (2.25)$$

Ferner zeigen die Rotationsmatrizen und damit auch die Elemente von S die folgenden Abhängigkeiten

$${}^{G_{b_i}}R^{G_j} = \begin{cases} {}^{G_{b_i}}R^{G_j}(q_{j+1}, \dots, q_{b_i}) & b_i > j \\ I_{3 \times 3} & b_i = j \end{cases} \quad (2.26)$$

Damit folgt z. B. für den in der Literatur üblicherweise betrachteten Fall [DT96] entsprechend Gleichung (2.8) für die Matrix S die Struktur

$$S(q) = \begin{pmatrix} 0 & S_{1,2} & S_{1,3}(q_2) & \dots & S_{1,N}(q_2, \dots, q_{N-1}) \\ 0 & 0 & S_{2,3} & \dots & S_{2,N}(q_3, \dots, q_{N-1}) \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \ddots & S_{N-1,N} \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}$$

Die Massenmatrix der Roboterglieder M_{Arm} ergibt sich als die Summe aus M_G und den von der Gelenkbeschleunigung abhängigen Termen aus der Ableitung der Rotorenergien. Sie kann unter Verwendung von Formeln aus [Rea94] als

$$M_{Arm} = M_G + S J^{-1} S^T \quad (2.27)$$

dargestellt werden.

Entsprechend Gleichung (2.5) können die Coriolis- und Zentrifugalkräfte aus der Massenmatrix bestimmt werden. Wenn die Bewegungsgleichungen die Struktur aus Gleichung (2.19)

haben, ergibt sich mit Gleichung (2.5) für die Coriolis- und Zentrifugalkräfte:

$$c_{Arm}(q, \dot{q}, \dot{\theta}) = \begin{pmatrix} M_{Arm}(q) & \dot{S}(q) \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\theta} \end{pmatrix} - \frac{1}{2} \left(\frac{\partial}{\partial q} X \right)^T \quad (2.28)$$

$$c_{Motor}(q, \dot{q}) = \begin{pmatrix} \dot{S}(q)^T & J \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\theta} \end{pmatrix} - \frac{1}{2} \left(\frac{\partial}{\partial \theta} X \right)^T \quad (2.29)$$

$$X = \begin{pmatrix} \dot{q}^T \dot{\theta}^T \end{pmatrix} \begin{pmatrix} M_{Arm}(q) & S(q) \\ S(q)^T & J \end{pmatrix} \begin{pmatrix} \dot{q} \\ \dot{\theta} \end{pmatrix}$$

Da X unabhängig von θ und die Matrix J konstant ist, vereinfacht sich Gleichung (2.29) zu

$$c_{Motor}(q, \dot{q}) = \dot{S}(q)^T \dot{q}$$

oder in elementweiser Darstellung zu

$$c_{Motor\,i}(q, \dot{q}) = \dot{q}^T \frac{\partial (S(q)^T)_i}{\partial q} \dot{q}. \quad (2.30)$$

Unter Berücksichtigung der Tatsache, dass die Matrix J unabhängig von q ist kann Gleichung (2.28) vereinfacht werden zu

$$c_{Arm}(q, \dot{q}, \dot{\theta}) = M_{Arm}(q) \dot{q} - \frac{1}{2} \left(\frac{\partial \dot{q}^T M_{Arm}(q) \dot{q}}{\partial q} \right)^T + \dot{S}(q) \dot{\theta} - \left(\frac{\partial \dot{q}^T S(q) \dot{\theta}}{\partial q} \right)^T. \quad (2.31)$$

Aufgrund der Struktur der S -Matrix können dabei die folgenden Abhängigkeiten der Elemente von c_{Arm} festgestellt werden

$$c_{Arm\,k} = c_{Arm\,k}(q, \dot{q}, \{\dot{\theta}_i | (b_i \geq k \wedge \exists S_{j,i} \neq 0) \vee \dot{S}_{k,i} \neq 0\}) \quad k = 1 \dots N. \quad (2.32)$$

Für die weiteren Betrachtungen interessiert noch die höchste, in den jeweiligen Bewegungsgleichungen des Armes vorkommende Ableitung von θ . Mit den Gleichungen (2.31) und (2.32) sieht man, dass θ_i nur dann in $c_{Arm\,k}$ auftreten kann, wenn $S_{k,i} \neq 0$ ist. Damit ist diese Gleichung aber auch von $\dot{\theta}_i$ abhängig. Zur Veranschaulichung dieses Ergebnisses sind die folgenden Überlegungen hilfreich: Um durch Corioliskräfte mit dem Motor i ein Moment auf die Achse k des MKS erzeugen zu können, muss der Motor um eine Achse außerhalb der Ebene ki drehbar sein. Somit kann durch diese Drehung die Achse des Motors aber auch so gestellt werden, dass i nicht senkrecht auf k ist. In diesem Fall kann durch Beschleunigung des Motors i ein Moment auf die Achse k erzeugt werden, womit $S_{j,i} \neq 0$ werden muss.

Die Potentialkäfte in Gleichung (2.19) ergeben sich aufgrund der Schwerkraft, wobei bereits berücksichtigt wurde, dass die Motorpositionen entsprechend Annahme 2.1 keinen Einfluss auf die Massenverteilung des Systems haben und damit

$$\left(\frac{\partial \mathcal{U}_{Grav}(q)}{\partial \theta} \right)^T = 0_{N \times 1}$$

gilt. Der verbleibende Term für die Potentialkräfte aufgrund der Gravitation ergibt sich unter Berücksichtigung der Gleichungen (2.6) und (2.16) zu

$$g_{Arm}(q) = \left(\frac{\partial \mathcal{U}_{Grav}(q)}{\partial q} \right)^T.$$

Eine häufig verwendete Annahme bei der Modellierung von RLFJ Robotern ist, dass die Bewegung der Motoren im Raum vernachlässigt wird und nur die Rotation um ihre Drehachse betrachtet wird. Dies ist äquivalent mit ${}^{b_i}\omega_{b_i} = 0_{3 \times 1}$ in Gleichung (2.24). Als Folge davon gilt $S = 0_{N \times N}$ was unter Berücksichtigung von Gleichung (2.31) auf Bewegungsgleichungen der Form

$$\begin{pmatrix} M_{Arm}(q) & 0_{N \times N} \\ 0_{N \times N} & J \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\theta} \end{pmatrix} + \begin{pmatrix} c_{Arm}(q, \dot{q}) \\ 0_{N \times 1} \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0_{N \times 1} \end{pmatrix} + \begin{pmatrix} \tau_G \\ \tau_A \end{pmatrix} = \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix} \quad (2.33)$$

führt.

2.2 Detailliertes Antriebsstrangmodell

Gewöhnlich wird der Antriebsstrang bei der Modellierung von RLFJ Robotern auf eine einfache Elastizität reduziert. In der Praxis hat er jedoch einen dominanten Einfluss auf die erreichbare Regelgüte. Aus diesem Grund wird im Anschluss ein detailliertes Antriebsstrangmodell hergeleitet.

Zum Antriebsstrang werden das Getriebe, der Motor und die Leistungselektronik gezählt. Für diese Komponenten existieren bereits aufwändige Modelle, für Getriebe z. B. in [Wan03], für Motoren in [Sch00a] und für die Leistungselektronik in [Sch96]. Ziel der Modellierung in dieser Arbeit ist ein regelungstechnisch relevantes und verwendbares Modell, wodurch der maximal mögliche Detaillierungsgrad eingeschränkt ist. So wird z. B. der mechanische Teil des Antriebsstranges durch 1D-rotatorische Elemente beschrieben, welche lediglich, wie in Abbildung 2.2 dargestellt, durch Antriebs- und Gelenkmomente mit dem Mehrkörpersystem verbunden sind. Elemente, deren Einfluss auf die Mehrkörperdynamik berücksichtigt wird, werden bereits im Mehrkörpermodell modelliert. Dieses Vorgehen wurde z. B. bei den Rotoren der Motoren gewählt.

Es werden zunächst einige Bezeichnungen für Elemente in mechanischen Antriebssträngen eingeführt. Entsprechend Abbildung 2.3 besitzt ein Antriebsstrangelement einen Antriebsflansch der durch den Antriebswinkel φ_a und das Antriebsmoment τ_a beschrieben ist und einen Abtriebsflansch mit den Größen φ_b und τ_b . Basierend auf diesem Elementarmodell werden im Folgenden verschiedene mechanische Effekte modelliert.

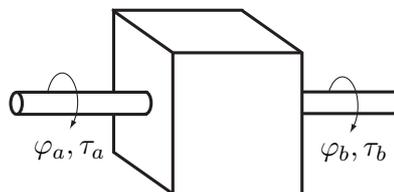


Abbildung 2.3: Definition der Größen an den Flanschen von Antriebsstrangelementen

2.2.1 Getriebe

In diesem Abschnitt wird das im Weiteren verwendete Getriebemodell mit den für den regelungstechnischen Entwurf wichtigen Effekten exemplarisch an einem Zykloidengetriebe her-

geleitet. Dieser Getriebetyp ist aufgrund seiner für Roboter günstigen Eigenschaften weit verbreitet und insbesondere am später betrachteten KUKA KR15/2 verbaut. Aber auch andere Getriebetypen wie Harmonic Drive und Planetengetriebe zeigen qualitativ ähnliche Effekte. Damit ist die Verwendung des Modells nicht auf Zykloidengetriebe beschränkt. In der Literatur finden sich nur wenige Veröffentlichungen über Zykloidengetriebe. Eine Beschreibung der Wirkungsweise, sowie die Berechnung und Messung von Kräften innerhalb des Getriebes ist z. B. in [Leh76] enthalten.

Funktionsprinzip und Aufbau

Das Funktionsprinzip des Zykloidengetriebes ist das eines Planetengetriebes. Die Besonderheit gegenüber einem herkömmlichen Planetengetriebe besteht darin, dass die Zykloiden­scheiben, welche den Planetenrädern entsprechen, so groß sind, dass sie über die Mitte des Hohlrades hinaus reichen. Dadurch ist die Verwendung eines Sonnenrades ausgeschlossen und an die Stelle des Sonnenrades tritt ein Mitnehmersystem, das die Drehung und das Moment der Zykloiden­scheiben auf den Abtrieb überträgt. Dem Hohlrad im Planetengetriebe entspricht der Bolzenring im Zykloidengetriebe mit den drehbaren Bolzen als „Zähnen“.

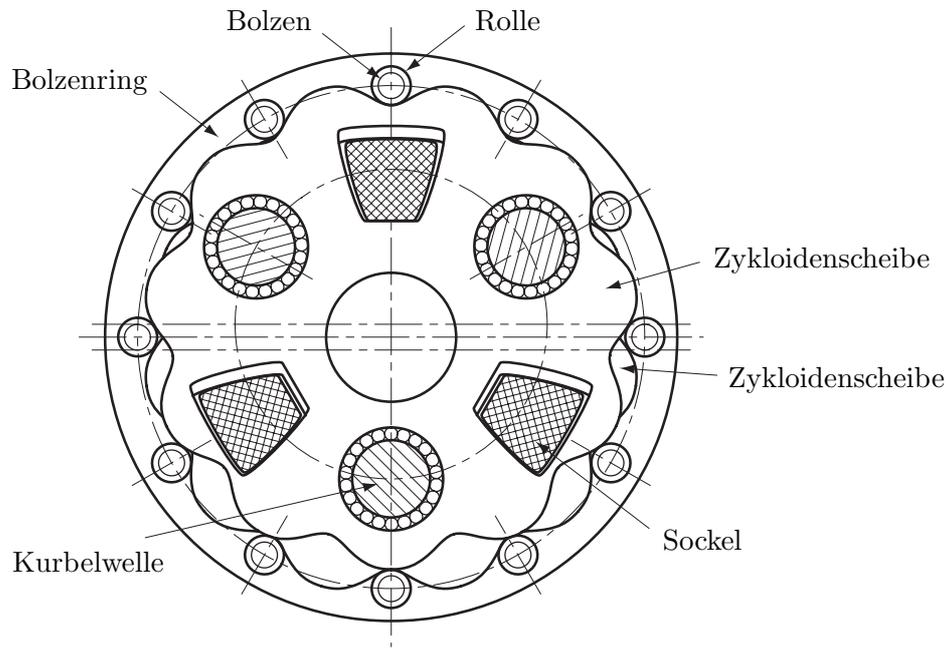
Der Aufbau eines Zykloidengetriebes wird im Folgenden an Hand einer schematischen Schnittzeichnung normal zur Getriebeachse und einer stark vereinfachten schematischen Darstellung in Abbildung 2.4 beschrieben. Es sind abweichend von diesen Abbildungen auch Bauformen mit nur einem Exzenter und unterschiedlichen Scheibenzahlen möglich, die Wirkungsweise bleibt aber ähnlich. Das dargestellte Getriebe besteht aus zwei Stufen. Die Vorstufe ist als Stirnradgetriebe realisiert und ermöglicht die Momenteneinleitung in das eigentliche Zykloidengetriebe über drei Kurbelwellen. Unterschiedliche Übersetzungsverhältnisse des Getriebes werden durch verschiedene Übersetzungen der Vorstufe realisiert, was aus Herstellersicht den Vorteil hat, dass die aufwändigen Hauptstufen identisch sind. Durch die Kurbelwellen werden zwei Scheiben, deren Zahnflanken verkürzte Zykloiden sind, in einem gehäusefesten Bolzenring abgewälzt.

Die beiden Zykloiden­scheiben sind um 180° versetzt, um eine gleichmäßige Lastaufteilung zu gewährleisten. Da die Scheiben w „Zähne“ weniger haben als der Bolzenring Bolzen, bewegen sie sich bei einer Umdrehung der Kurbelwellen um w/z Umdrehungen in entgegengesetzter Richtung, wobei z die Anzahl der Kurvenabschnitte einer Zykloiden­scheibe ist. Bei der Bewegung der Zykloiden­scheiben sind die Zähne außen in Berührung mit den zylinderförmigen Bolzen, wobei die Last immer auf mehrere Bolzen aufgeteilt ist. Um Reibungsverluste und Verschleiß auf ein Minimum zu reduzieren, wird durch Rollen auf den Bolzen für eine wälzende Kraftübertragung zwischen den Zykloiden­scheiben und dem Bolzenring gesorgt.

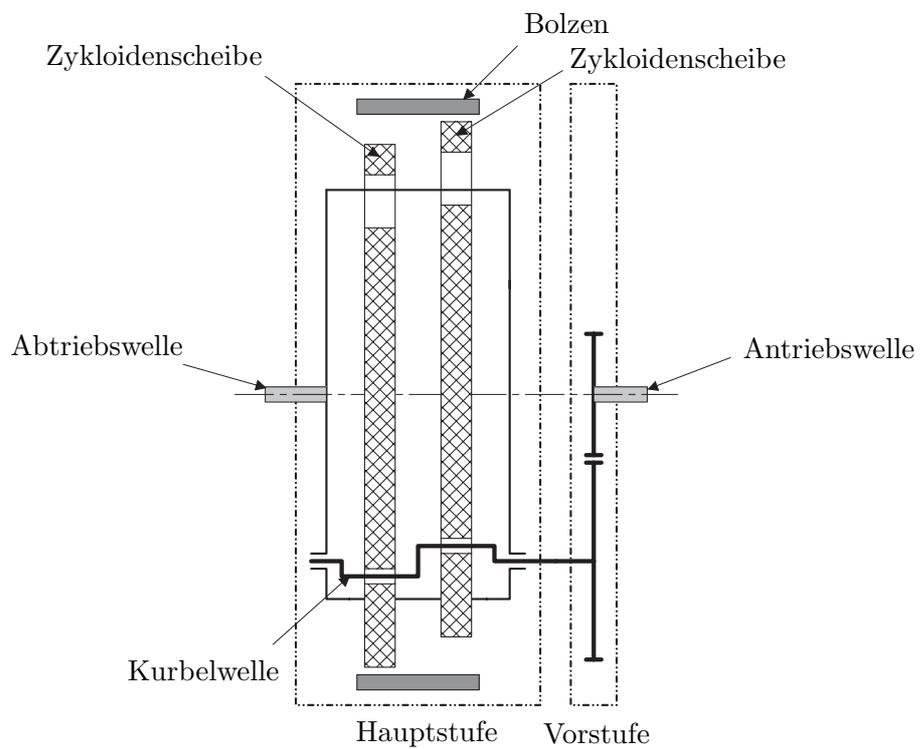
Übersetzung mit Drehungleichförmigkeiten

Der eigentliche Sinn des Getriebes ist die Übersetzung der Motorbewegung ins Langsame. Für ein ideales Getriebe mit der Getriebeübersetzung u würde gelten

$$\begin{aligned}\dot{\varphi}_a &= u\dot{\varphi}_b \\ \tau_a &= -\frac{\tau_b}{u}\end{aligned}$$



(a) Schnitt normal zur Getriebeachse



(b) Aufbau des Getriebes

Abbildung 2.4: Schematische Darstellung eines Zykloidengetriebes basierend auf [Abe03]

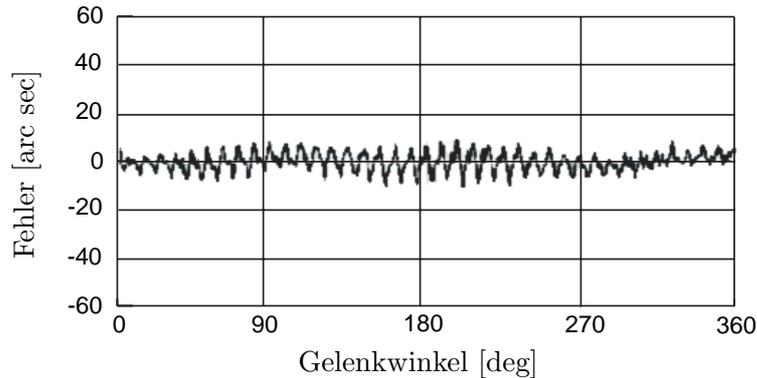


Abbildung 2.5: Übertragungsfehler von Zykloidengetrieben am Beispiel eines Teijin RV-320E-171 Getriebes [Tei]

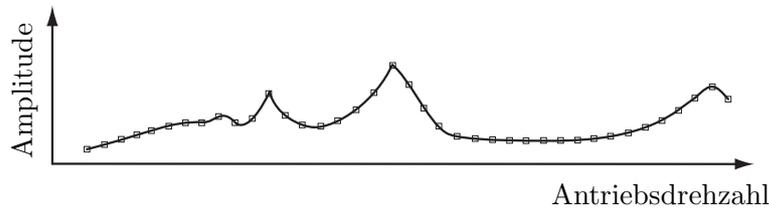


Abbildung 2.6: Schwingungsamplitude am Abtrieb bei konstanter Antriebsdrehzahl am Beispiel eines Fine Cyclo F2C-T Getriebes [Sum96]

Nun ist aber bekannt, dass Getriebe aufgrund von Fehlern in der Getriebekinematik Drehungleichförmigkeiten aufweisen. Aus diesem Grund werden, z. B. bei Stirnradgetrieben, Wälzprüfungen durchgeführt [MMW84]. Die Ergebnisse solcher Messungen sind qualitativ aus Getriebedatenblättern zu entnehmen und zeigen z. B. ein Verhalten wie in Abbildung 2.5. Zudem ist bekannt, dass Zykloidengetriebe bei konstanter Drehzahl am Antrieb eine Welligkeit am Abtrieb erzeugen, welche im Idealfall nicht vorhanden wäre. Deren Frequenz und Amplitude hängen von der Drehzahl am Antrieb ab. Ein Beispiel dafür ist in Abbildung 2.6 angegeben. Diese Effekte werden im Folgenden modelliert. Die zugrunde liegende Annahme dabei ist, dass die im Getriebe auftretende Drehwinkelabweichung zwischen theoretischem und tatsächlichem Abtriebsdrehwinkel durch eine positionsabhängige Getriebeübersetzung entsteht [Abe03], welche auch die Welligkeit des Abtriebs bei konstanter Antriebsdrehzahl verursacht. Physikalisch kann diese Annahme z. B. durch ein exzentrisch laufendes Zahnrad erklärt werden [Kum00], was für ein Exzentergetriebe, zu denen die betrachteten Zykloidengetriebe zählen, wahrscheinlich ist. Entsprechend kann die Charakteristik von $u(\cdot)$ über Abbildung 2.7 motiviert werden. Hier ist die Bewegung zweier Zahnräder idealisiert durch den Abrollvorgang ihrer Wälzkreise beschrieben, wobei ein Rad eine Exzentrizität e aufweist. Für dieses Beispiel ergibt sich unter Vernachlässigung der von φ_b abhängigen Bewegung des Berührungspunktes der Wälzkreise eine Übersetzung von

$$\mathbf{u}(\varphi_b) = -\frac{r_M + e \cos(\varphi_b)}{r_A}.$$

Bei einem Zykloidengetriebe sind die Verhältnisse etwas komplizierter. Hier rollt die Zykloidscheibe an den Bolzen im Bolzenring ab. Die Bolzen werden dabei als ideal angenommen.

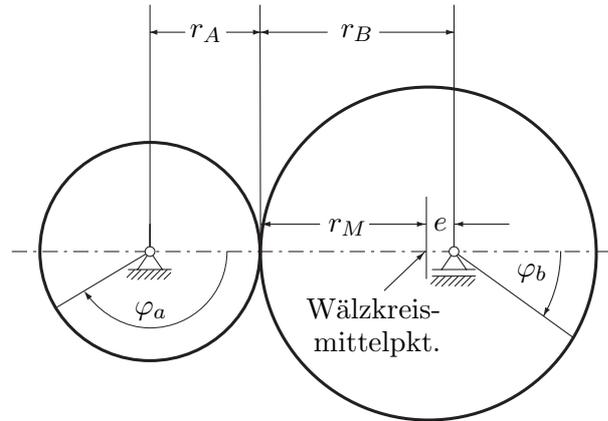


Abbildung 2.7: Schematische Darstellung des Einflusses eines exzentrischen Rades auf die Übersetzung

Fehler in der Zahnform oder der Bewegung der Zykloidscheibe wirken sich aber auch hier als Übersetzungsfehler aus. Entsprechend diesen Überlegungen ist die Übersetzung von der Lage der Zykloidscheibe abhängig und damit eine Funktion des Abtriebswinkels φ_b . Folglich gilt für das Getriebe:

$$\dot{\varphi}_a = u(\varphi_b)\dot{\varphi}_b \quad (2.34a)$$

$$\tau_a = -\frac{\tau_b}{u(\varphi_b)} \quad (2.34b)$$

Eine exakte Herleitung der Gleichung für $u(\varphi_b)$ ist naturgemäß nicht möglich, da die Drehungleichförmigkeit durch Fehler in der Getriebekinematik entsteht, die zunächst einmal nicht als einfache Gleichung angegeben werden können. Da sich die Fehler in der Getriebekinematik nach spätestens einer Umdrehung des Abtriebs wiederholen werden, können sie statt dessen als Fourier-Reihe dargestellt werden. Mit n_u berücksichtigten, ganzzahligen Frequenzen \tilde{u} ergibt sich die Gleichung für die Übersetzung zu

$$u(\varphi_b) = u_0 + \sum_{j=1}^{n_u} \hat{u}_j \cos(\tilde{u}_j \varphi_b + \bar{u}_j), \quad (2.35)$$

wobei mit u_0 die Basisübersetzung, mit \hat{u} die Amplituden und mit $0 \leq \bar{u} < 2\pi$ die Phasenlagen der Übersetzungsschwankungen bezeichnet werden. Die Differentialgleichung (2.34a) kann durch Trennung der Variablen gelöst werden:

$$\int d\varphi_a = \int u(\varphi_b) d\varphi_b.$$

Einsetzen von Gleichung (2.35) liefert

$$\begin{aligned} \varphi_a &= \int u_0 + \sum_{j=1}^{n_u} \hat{u}_j \cos(\tilde{u}_j \varphi_b + \bar{u}_j) d\varphi_b \\ &= u_0 \varphi_b + \sum_{j=1}^{n_u} \frac{\hat{u}_j}{\tilde{u}_j} \sin(\tilde{u}_j \varphi_b + \bar{u}_j) + u_C \end{aligned}$$

Mit den Anfangsbedingungen $\varphi_a(t=0) = \varphi_{a0}$ und $\varphi_b(t=0) = \varphi_{b0}$ kann nun die Integrationskonstante

$$\mathbf{u}_C = \varphi_{a0} - \mathbf{u}_0\varphi_{b0} - \sum_{j=1}^{n_u} \frac{\hat{\mathbf{u}}_j}{\bar{\mathbf{u}}_j} \sin(\tilde{\mathbf{u}}_j\varphi_{b0} + \bar{\mathbf{u}}_j)$$

bestimmt werden. Damit erhält man für die Übersetzung die folgenden Gleichungen:

$$\mathbf{u} = \mathbf{u}_0 + \sum_{j=1}^{n_u} \hat{\mathbf{u}}_j \cos(\tilde{\mathbf{u}}_j\varphi_b + \bar{\mathbf{u}}_j) \quad (2.36a)$$

$$\varphi_a = \mathbf{u}_0\varphi_b + \sum_{j=1}^{n_u} \frac{\hat{\mathbf{u}}_j}{\bar{\mathbf{u}}_j} \sin(\tilde{\mathbf{u}}_j\varphi_b + \bar{\mathbf{u}}_j) + \mathbf{u}_C \quad (2.36b)$$

$$\tau_a = -\frac{\tau_b}{\mathbf{u}} \quad (2.36c)$$

In der Praxis ist es nicht notwendig ein hohes n_u zu verwenden. Auswertungen an gemessenen Kurven haben gezeigt, dass die auftretenden Schwingungen bereits mit wenigen Frequenzen gut wiedergegeben werden können, und qualitativ eine hohe Übereinstimmung von Messung und Simulation erreicht wird [Abe03]. Eine unsaubere Bewegung der Zykloidscheibe wird einen Anteil bei $\tilde{\mathbf{u}}_j = 1$ und wenige Oberwellen erzeugen. Fehler in der Zahnform der Zykloidscheibe werden eine Frequenz entsprechend der Zähnezahl der Scheibe $\tilde{\mathbf{u}}_j = z$ und einige Oberwellen niedriger Ordnung erzeugen.

Da sich die Ungleichförmigkeiten durch Verschleiß im Laufe der Zeit ändern und es zudem schwierig ist, sie getrennt von anderen Getriebeeffekten messtechnisch zu erfassen, stellen die Gleichungen (2.36) für den regelungstechnischen Entwurf eine hinreichend gute Näherung dar.

Nichtlineare Elastizität

In den meisten Veröffentlichungen zur Modellierung von RLFJ Robotern wird mit der Begründung von kleinen Schwingungen eine lineare Elastizität angenommen. Diese Argumentation ist jedoch nicht ganz korrekt. Auch wenn die Auslenkungen aufgrund der Elastizität relativ zur Roboterbewegung klein sind, wird doch im Betrieb des Roboters das maximal zulässige Getriebemoment ausgenutzt werden, sonst wäre ja ein kleineres Getriebe verbaut worden. Nun ist aber bekannt, dass die eingesetzten Getriebe eine nichtlineare Steifigkeitskennlinie besitzen, welche damit bei hochdynamischen Roboterbewegungen auch komplett durchfahren wird. Aus diesem Grund wird in dieser Arbeit ein detaillierteres Steifigkeitsmodell verwendet.

Bei Zykloidengetrieben zeigt die Steifigkeitskennlinie qualitativ einen Verlauf wie in Abbildung 2.8 angegeben. Dabei ist im Bereich niedriger Last eine reduzierte Steifigkeit erkennbar. Für hohe Lasten nähert sich die Kennlinie einer Geraden entsprechend einer höheren Steifigkeit an. Zusätzlich weist die Kennlinie gewöhnlich ein Hystereseverhalten auf, welches in Abbildung 2.8 durch die eingeschlossene Fläche des Nachgiebigkeitsverlaufs deutlich wird. Dieses wird in Datenblättern z. B. durch Angabe von *lost motion* und *backlash* charakterisiert.

Die gezeigte Kurvenform entsteht durch den gleichzeitigen, verteilten Eingriff mehrerer Zähne im Getriebe. Zur Modellierung dieses Effekts existieren extrem detaillierte und aufwändige Modelle, welche die Kontaktzustände zwischen den Formflächen mit Finite-Element-Methoden betrachten. Siehe hierzu z. B. [Wan03]. Vereinfacht kann der Eingriff mehrerer

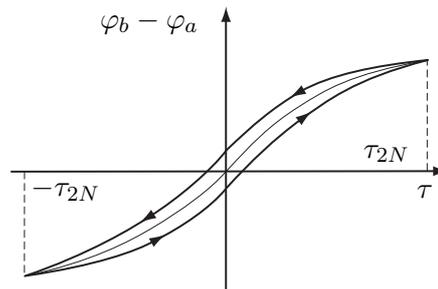


Abbildung 2.8: Qualitativer Verlauf der Steifigkeitskennlinie eines Zykloidengetriebes

Zähne als eine parallele Anordnung mehrerer Federn in Kombination mit Reibelementen modelliert werden. Bedingt durch die Getriebekinematik weisen die Federn dann unterschiedliche Vorspannungen auf. Daraus resultiert eine lastabhängige Steifigkeit, da bei niedrigen Lasten aufgrund der verteilten Elastizitäten weniger Zähne im Eingriff sind als bei hoher Belastung. Aufgrund der Getriebekinematik hängt die Steifigkeit zusätzlich in geringem Maße vom Abtriebswinkel ab, was in [Abe03] an einem Zykloidengetriebe auch experimentell bestätigt wurde. Die Reibelemente können sich, abhängig von der vorherigen Belastung, zusätzlich noch im Haften oder bereits im Gleiten befinden. Diese Eigenschaft wird als Strukturgedächtnis [Kum00] bezeichnet und verursacht das dargestellte Hystereseverhalten. Einen Überblick über die Modellierung von Hysterese gibt z. B. [May91]. Ein praktikables, statisches Hysterese-Modell für Antriebsstrangelemente auf Basis von Preisach-Modellen wird in [Kum00] aufgestellt, ein effizientes dynamisches Modell in [Sch03]. Eine verallgemeinerte Methode zur Modellierung und Kompensation von Hysterese ist in [Kuh03] gegeben.

Aus Datenblättern und Messungen [Abe03] ist jedoch ersichtlich, dass die eingesetzten Getriebe in der Regel nur in sehr geringem Maße Hystereseeffekte zeigen. Dies liegt darin begründet, dass sich Hystereseverluste direkt in der wichtigen Wiederholgenauigkeit des Roboters bemerkbar machen. Daher kommen für Roboter, bei denen eine präzise Positionierung wichtig ist, nur Getriebe zum Einsatz, bei denen diese Verluste gering sind.

Deshalb wird in dieser Arbeit auf die Modellierung von Hystereseeffekten verzichtet. Berücksichtigt wird jedoch der progressive Verlauf der Steifigkeit. Diese wird als nichtlineare Kennlinie modelliert. Die verteilten Reibungen werden zu einem parallel geschalteten Reibelement zusammengefasst. Damit ergibt sich das Modell des elastischen Verhaltens zu

$$\varphi_{\Delta} = \varphi_b - \varphi_a \quad (2.37a)$$

$$\tau_b = k(\varphi_{\Delta}) + d\dot{\varphi}_{\Delta} \quad (2.37b)$$

$$\tau_a = -\tau_b \quad (2.37c)$$

mit der Kennlinie für die Steifigkeit k und der Dämpfung d . Eine übliche Methode zur Parametrisierung der Steifigkeitskennlinie ist ein Polynomansatz

$$k(\varphi_{\Delta}) = c_1\varphi_{\Delta} + c_2\varphi_{\Delta}^3$$

oder z. B. die Funktion

$$k(\varphi_{\Delta}) = c_1\varphi_{\Delta} + c_2\operatorname{sgn}(\varphi_{\Delta})|\varphi_{\Delta}|^{c_3}.$$

Welche Funktionen für die Kennlinien tatsächlich im Modell eingesetzt werden, hängt letztlich von den Messungen am jeweils betrachteten Getriebe ab.

Getriebeverluste

In diesem Abschnitt werden, basierend auf [NW02], die Getriebeverluste betrachtet. Grundsätzlich gilt für das Getriebe, dass die Summe aus den Leistungen an den Flanschen P_a und P_b und der Verlustleistung P_v Null sein muss, also

$$P_a + P_b + P_v = 0,$$

wobei abgeführte Leistung und Verlustleistung negativ gezählt werden. Die Verlustleistung besteht zum einen aus den geschwindigkeitsabhängigen Verlusten P_{vG} aufgrund der Lagerbelastung der Wälzlager, Dichtungsreibung und Ölpanschverlusten. Zum anderen entstehen durch die Gleit-Wälz-Bewegung der Zahnflanken Verluste an der Kraftübertragungsstelle, zu denen auch Quetschverluste durch die Ölverdrängung aus den Zahnlücken gezählt werden. Diese Verluste sind hauptsächlich proportional zur übertragenen Leistung und werden mit P_{vP} bezeichnet. Damit ist

$$P_v = P_{vG} + P_{vP}.$$

Schon seit geraumer Zeit ist die Modellierung von Reibungsverlusten Gegenstand der Forschung. Aus diesem Grund ist auch eine Fülle an Modellvorstellungen und Literatur entstanden. Trotzdem sind die physikalischen Grundlagen der Reibung noch nicht völlig geklärt [Müs03]. Einen guten Überblick über gebräuchliche Modellvorstellungen zur Reibung geben z. B. [AHDd94, AHd96, Ols96, OÅd⁺98]. Man unterscheidet generell statische und dynamische Modelle. Zur Modellierung der *geschwindigkeitsabhängigen Verluste* werden häufig die folgenden statischen Reibmodelle kombiniert. Die Auflistung orientiert sich an [Ols96].

Coulomb Reibmodell:

Zugrunde liegt die Vorstellung, dass die Reibung der Bewegung entgegenwirkt und unabhängig von Geschwindigkeit und Kontaktfläche ist. Das Modell kann daher unter Verwendung der Coulomb-Reibung τ_C durch

$$\tau_R = \tau_C \operatorname{sgn}(\dot{\varphi}) \quad (2.38)$$

beschrieben werden, woraus sich ein Verlauf entsprechend Abbildung 2.9 (a) ergibt. Eine direkte Implementierung nach Gleichung (2.38) in der Simulation ist auf Grund der Definition der Signumfunktion mit

$$\operatorname{sgn}(x) = \begin{cases} +1 & \text{für } x > 1 \\ 0 & \text{für } x = 0 \\ -1 & \text{für } x < 1 \end{cases}$$

nicht möglich, da die Reibung beim Haften verschwindet. In translatorischen mechanischen Systemen wird häufig angenommen, dass die Reibung proportional zur Normalkraft f_N ist. Die Reibkraft berechnet sich dann mit der Gleitreibungszahl μ und der relativen Geschwindigkeit v_{rel} zu

$$f_R = \mu f_N \operatorname{sgn}(v_{rel}). \quad (2.39)$$

Viskose Reibung:

Aus der Theorie der Hydrodynamik entstand ein Reibmodell, bei dem die Reibung auf

Grund der Viskosität von Fluiden berücksichtigt wird. In rotatorischen mechanischen Systemen wird die viskose Reibung mit dem Koeffizienten der viskosen Reibung ν durch

$$\tau_R = \nu \dot{\varphi}$$

beschrieben. Viskose Reibung wird meistens, wie in Abbildung 2.9 (b) dargestellt, mit Coulomb-Reibung kombiniert. Für ein realistischeres Modell kann die viskose Reibung als nichtlinear in der Geschwindigkeit modelliert werden. Eine verallgemeinerte Form der viskosen Reibung ist daher

$$\tau_R = \nu |\dot{\varphi}|^{\delta_\nu} \operatorname{sgn}(\dot{\varphi}), \quad (2.40)$$

wobei δ_ν von den jeweiligen geometrischen Bedingungen abhängt.

Haftreibung:

Beim Haftreibmodell wird das Reibmoment im Stillstand nicht durch eine Funktion der Geschwindigkeit beschrieben, sondern durch eine Funktion des von außen einwirkenden Moments τ_e .

$$\tau_R = \begin{cases} \tau_e & \text{für } \dot{\varphi} = 0 \wedge |\tau_e| < \tau_H \\ \tau_H \operatorname{sgn}(\tau_e) & \text{für } \dot{\varphi} = 0 \wedge |\tau_e| \geq \tau_H \end{cases}$$

Beim Haften kann das Reibmoment damit jeden Wert annehmen, der zwischen den beiden Extrema $-\tau_H$ und τ_H liegt. In Kombination mit Coulomb- und Haftreibung ergibt sich ein Verlauf wie in Abbildung 2.9 (c).

Stribeck Effekt:

Stribeck stellte fest, dass die Reibung nicht wie in Abbildung 2.9 (c) nach dem Losbrechen unstetig abnimmt, sondern dass die Geschwindigkeitsabhängigkeit stetig ist, wie in Abbildung 2.9 (d) dargestellt. Eine allgemeinere Beschreibung der Reibung als mit dem klassischen Modell ist daher

$$\tau_R = \begin{cases} \tau(\dot{\varphi}) & \text{für } \dot{\varphi} \neq 0 \\ \tau_e & \text{für } \dot{\varphi} = 0 \wedge |\tau_e| < \tau_H \\ \tau_H \operatorname{sgn}(\tau_e) & \text{sonst} \end{cases}$$

mit einer beliebigen Funktion $\tau(\dot{\varphi})$, die wie in Abbildung 2.9 (d) aussehen kann. Für die Parametrierung dieser Funktion $\tau(\dot{\varphi})$ existieren viele Möglichkeiten welche von der Art der Anwendung abhängen, siehe dazu [AH91]. Eine vielfach verwendete Form der Nichtlinearität ist dabei

$$\tau(\dot{\varphi}) = \left(\tau_C + (\tau_H - \tau_C) e^{-|\dot{\varphi}/\dot{\varphi}_S|^{\delta_S}} \right) \operatorname{sgn}(\dot{\varphi}) + \nu \dot{\varphi} \quad (2.41)$$

mit der Stribeck-Geschwindigkeit $\dot{\varphi}_S$ und einem Parameter δ_S , der im Bereich von 0.5 bis 2 liegt.

Ferner sind noch eine Reihe von Erweiterungen dieser Modelle, wie z. B. das Karnopp-Modell [AHd96] und das Armstrong-Modell [AHDd94] bekannt. Auch bei diesen handelt es sich um statische Modelle.

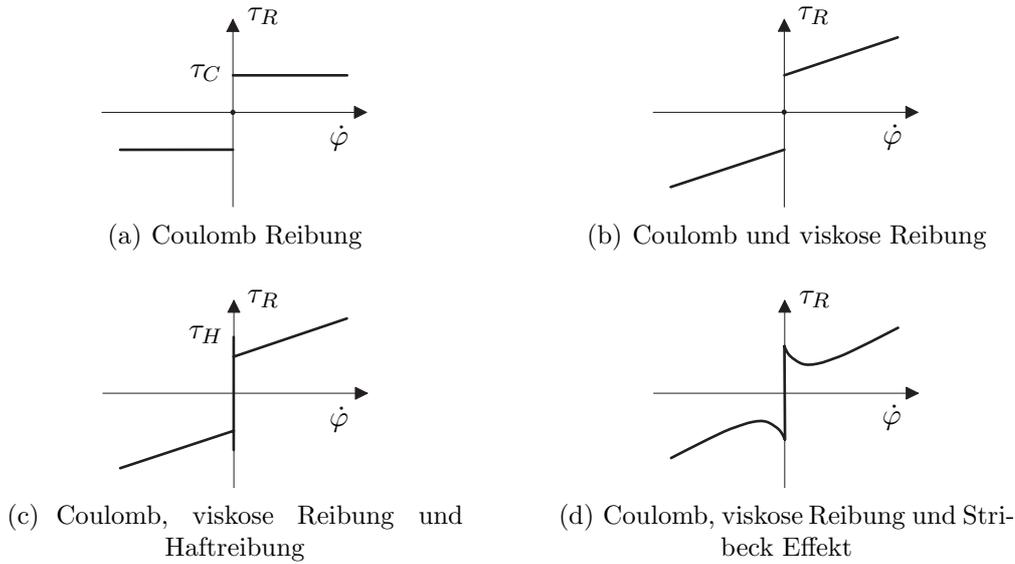


Abbildung 2.9: Gebräuchliche statische Reibmodelle

Im Gegensatz dazu basieren die dynamischen Modelle auf der Vorstellung, dass der Kontakt von zwei Oberflächen durch viele mikroskopische Federn bzw. Bürsten erfolgt, welche beim Übergang vom Haften zum Gleiten langsam durchtrennt werden. Besonders gebräuchlich sind hier das Dahl-Modell und das LuGre-Modell [dOÅL95]. Der wesentliche Vorteil dieser Modelle ist, dass durch die Zustände im Reibmodell der Bereich des *presliding* besser modelliert werden kann. Unter *presliding* versteht man, dass auch im Haften innerhalb eines bestimmten Bereichs mikroskopische Bewegungen stattfinden können.

Nachteilig an den dynamischen Modellen ist allerdings, dass sie auf steife Differentialgleichungssysteme führen, welche sich schlecht für den Regelungsentwurf und Simulationen eignen. Zusätzlich stellt das Parameteridentifikationsproblem sehr hohe Anforderungen an die Messgenauigkeit des Prüfstandes. Zieht man jetzt noch in Betracht, dass die Reibung wesentlich von weiteren Einflussgrößen wie Temperatur, Schmierung und Verschleißzustand des Getriebes abhängig ist, dann ist der Nutzen der dynamischen Modelle für regelungstechnische Zwecke fraglich und die Entscheidung naheliegend, ein statisches Reibmodell zu verwenden. Zur Modellierung der geschwindigkeitsabhängigen Reibung wird deshalb das folgende Modell verwendet:

$$\varphi_a = \varphi_b \quad (2.42a)$$

$$\tau_R = \tau_a + \tau_b \quad (2.42b)$$

$$\tau_R = \begin{cases} r(\dot{\varphi}_b) & \text{für } \dot{\varphi}_b \neq 0 \\ \tau_a + \tau_b & \text{für } \dot{\varphi}_b = 0 \wedge |\tau_a + \tau_b| < \tau_H \\ \tau_H \text{sgn}(\tau_e) & \text{sonst} \end{cases} \quad (2.42c)$$

Die Form und Parametrisierung der Kennlinie $r(\dot{\varphi}_b)$ wird auch hier entsprechend dem jeweils eingesetzten Getriebe gewählt.

Das Reibmodell (2.42) ist in der Anwendung aufgrund der Abhängigkeit der Reibung von den externen Momenten im Stillstand nicht ganz unproblematisch. So müssen zur effizienten

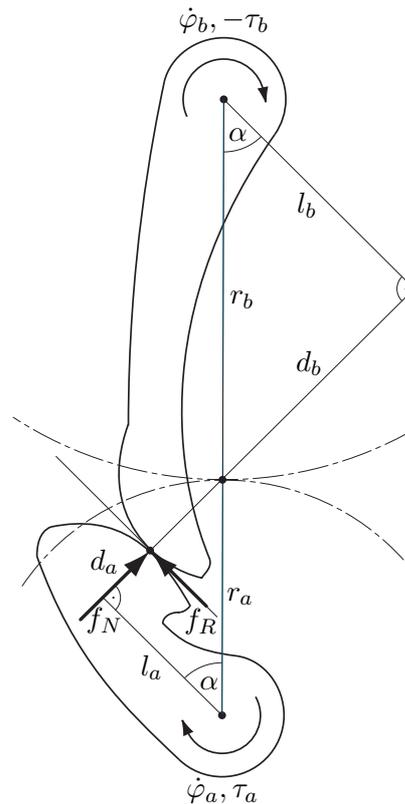


Abbildung 2.10: Zahnflanken eines Stirnradgetriebes nach [NW02]

Simulation und Invertierung geeignete Maßnahmen ergriffen werden, auf welche im Lauf der Arbeit noch eingegangen wird.

Verluste durch Gleitbewegungen von Formflächen bei der Kraftübertragung treten im Getriebe an zahlreichen Stellen auf, wozu beispielsweise die Zahnflanken des Stirnradgetriebes der Getriebevorstufe gehören. Diese Konfiguration ist in Abbildung 2.10 dargestellt. Die folgende Herleitung der *leistungsabhängigen Verluste* P_{vP} erweitert die Darstellungen in [TOB01].

Zunächst wird ein punktförmiger Kontakt angenommen, an welchem ein Zahnepaar mit der Kraft f_N zusammengedrückt wird. Damit sich die beiden Zahnflächen bei Bewegung der Zahnräder weder durchdringen noch voneinander abheben, müssen die Normalgeschwindigkeiten der Flächen im Berührungspunkt identisch sein. Damit sind jedoch die Tangentialgeschwindigkeiten im Berührungspunkt verschieden [KR92], d.h. die beiden Flächen gleiten aufeinander ab. Durch diesen Gleitvorgang wirkt am Kontaktpunkt der Flächen Reibung, welche als Coulombreibung entsprechend Gleichung (2.39) angenommen wird. Durch Aufstellen der Momentengleichgewichte für das Antriebsmoment τ_a und das Abtriebsmoment τ_b

$$\begin{aligned}\tau_a &= f_N l_a - f_R d_a \\ -\tau_b &= f_N l_b - f_R d_b\end{aligned}$$

und Einsetzen von Gleichung (2.39) ergibt sich

$$\tau_b = -\frac{l_b(1 - \operatorname{sgn}(v_{rel})\mu\frac{d_b}{l_b})}{l_a(1 - \operatorname{sgn}(v_{rel})\mu\frac{d_a}{l_a})}\tau_a \quad (2.43)$$

Aufgrund der Verzahnungsgeometrie sind die Vorzeichen der Relativgeschwindigkeit und der Drehzahl ω_a identisch [NW02]:

$$\operatorname{sgn}(v_{rel}) = \operatorname{sgn}(\dot{\varphi}_a) \quad (2.44)$$

Werden ferner noch die geometrischen Abmessungen berücksichtigt,

$$\cos \alpha = \frac{l_a}{r_a} = \frac{l_b}{r_b} \quad \Rightarrow \quad \frac{l_b}{l_a} = \frac{r_b}{r_a} = -\mathbf{u} \quad (2.45)$$

dann kann Gleichung (2.43) durch Einsetzen der Gleichungen (2.44) und (2.45) umgeformt werden auf

$$\tau_b = \mathbf{u}\eta\tau_a \quad (2.46)$$

$$\eta = \frac{1 - \operatorname{sgn}(\dot{\varphi}_a)\mu\frac{d_b}{l_b}}{1 - \operatorname{sgn}(\dot{\varphi}_a)\mu\frac{d_a}{l_a}} \quad (2.47)$$

mit dem Wirkungsgrad η . Der in Abbildung 2.10(a) dargestellte Fall setzt $\tau_a > 0$ voraus. Damit ist η entsprechend Gleichung (2.47) von der Richtung des Leistungsflusses $P_a = \tau_a\omega_a$ abhängig, welche bei positivem τ_a durch ω_a bestimmt ist. Für den Fall $\tau_a < 0$ sind anschaulich die Zahnflanken auf der anderen Seite der Zähne im Eingriff, die Herleitung kann in der gleichen Art und Weise erfolgen.

Die leistungsabhängigen Verluste treten im Getriebe jedoch nicht nur an den gerade betrachteten Zahnflanken, sondern an zahlreichen Stellen auf. Da sich die jeweiligen geometrischen Verhältnisse d_a , d_b , l_a und l_b zusätzlich noch mit dem Winkel φ_a verändern werden und die jeweiligen Gleitreibzahlen μ unsicher sind, erscheint eine analytische Berechnung unrealistisch. Die Herleitung hat aber bereits gezeigt, dass die leistungsabhängigen Verluste die Form

$$\tau_b = \eta\tau_a \quad (2.48)$$

haben, wobei der Wirkungsgrad η von der Leistungsflussrichtung abhängig ist. Der Fall $P_a = 0$ ist bislang noch nicht definiert und erfordert eine gesonderte Behandlung. $P_a = 0$ ist zum einen möglich durch $\omega_a = 0$. Da die Reibung an den Formflächen mit Gleichung (2.39) modelliert wurde, verschwindet sie bei $\omega_a = 0$. Damit gilt $\eta = 1$, was auch aus Gleichung (2.47) folgt. Zum anderen ist der Fall $\tau_a = 0$ möglich. Wie aus Gleichung (2.48) ersichtlich, ist der Wirkungsgrad in diesem Fall beliebig. Daher wird auch hier $\eta = 1$ gewählt.

Bei der Betrachtung von Getrieben haben Winkelgeschwindigkeit und Moment definitionsgemäß am Antrieb gleiche und am Abtrieb verschiedene Vorzeichen, was auch in Abbildung 2.10 berücksichtigt wurde. Entsprechend Abbildung 2.3 haben in dieser Arbeit bei Antriebsstrangelementen Winkelgeschwindigkeit und Moment aber an beiden Flanschen gleiche Vorzeichen. Um eine mit Abbildung 2.3 konsistente Beschreibung zu erhalten, muss daher noch τ_b durch $-\tau_b$ ersetzt werden. Damit ergibt sich zur Modellierung der leistungsabhängigen

Verluste das folgende Modell

$$\varphi_a = \varphi_b \quad (2.49a)$$

$$\tau_b = -\eta\tau_a \quad (2.49b)$$

$$\eta = \begin{cases} \eta_p & \text{für } \tau_a \dot{\varphi}_a > 0 \\ \frac{1}{\eta_n} & \text{für } \tau_a \dot{\varphi}_a < 0, \\ 1 & \text{für } \tau_a \dot{\varphi}_a = 0 \end{cases} \quad (2.49c)$$

wobei η_p den Wirkungsgrad bei positivem und η_n den Wirkungsgrad bei negativem Leistungsfluss $\tau_a \omega_a$ bezeichnet. Da die Reibung immer der Bewegung entgegenwirkt, gilt für die Wertebereiche von η_p und η_n :

$$0 \leq \eta_p \leq 1$$

$$0 \leq \eta_n \leq 1$$

2.2.2 Motor und Leistungselektronik

Für den Antrieb von Robotern stehen die verschiedensten Motortypen zur Verfügung. Für Industrieroboter wie den KUKA KR15/2 werden wegen ihrer hohen Leistungsdichte typischerweise permanentmagneterregte Servosynchronmotoren eingesetzt, welche über mit Insulated-Gate-Bipolar Transistoren (IGBT) bestückte Wechselrichter angesteuert werden. Für das System Motor / Stromregler existieren in der Literatur detaillierte Modelle [Sch00a]. Für Roboter-Manipulatoren gilt jedoch, dass die Zeitkonstanten der Stromregelung sehr viel geringer sind als die der Robotermechanik. Aus diesem Grund wird die Dynamik der Stromregelung im regelungstechnischen Entwurf gern als

$$\tau(s) = \frac{k_T}{1 + t_{SR}s} I_d$$

mit dem Übersetzungsfaktor von Strom auf Moment k_T , der Ersatzzeitkonstante der Stromregelung t_{SR} und dem Sollstrom I_d modelliert. Die Bewegungsgleichungen der Rotoren mit den Motormomenten τ_M und den Antriebsmomenten τ_A wurde bereits im Rahmen der MKS-Modellierung aufgestellt. Für einen einzelnen Rotor i lautet die Gleichung entsprechend (2.19)

$$\begin{aligned} \tau_{Mi} &= I_{zzi} \ddot{\theta}_i - \tau_{Ai} + \tau_{MKS} \\ \tau_{MKS} &= (S(q)^T)_i \ddot{q} + c_{Motori}(q, \dot{q}) \end{aligned}$$

Bei der Identifikation am Roboter hat sich jedoch gezeigt, dass die Zeitkonstanten der Stromregelung so klein sind, dass sie auch für ein genaues Robotermodell vernachlässigt werden können. Deshalb, und um die weiteren Betrachtungen übersichtlicher zu halten, wurde auf das Modell des Stromreglers verzichtet und das Motormoment τ_M als Stellgröße angesehen. Grundsätzlich kann der Regelungsentwurf in dieser Arbeit auch unter Verwendung des Modells nach Gleichung (2.2.2) durchgeführt werden. Auf die dafür notwendigen Erweiterungen wird an den entsprechenden Stellen hingewiesen. Die in der Lagerung des Motors auftretende Reibung wird innerhalb des Antriebsstrangmodells berücksichtigt, wofür wiederum das Modell gemäß den Gleichungen (2.42a), (2.42b) und (2.42c) benutzt wird.

2.2.3 Gesamtes Antriebsstrangmodell

Aus den Überlegungen in den vorhergehenden Abschnitten soll nun ein Modell für den in Abbildung 2.2 dargestellten Antriebsstrang aufgestellt werden. Dieses enthält die Modelle für Motorreibung, Elastizität, Übersetzungsfehler, Wirkungsgrad und Gelenkreibung. Da die Elastizität, der Übersetzungsfehler und der Wirkungsgrad Approximationen im Getriebe verteilte auftretender Effekte sind, ergibt sich für deren Anordnung im Allgemeinen keine zwingende Reihenfolge. Um ein möglichst universell einsetzbares Modell zu erhalten, wird eine Anordnung entsprechend Abbildung 2.11 gewählt. Dabei sind Reibelemente vor und nach der Elastizität angeordnet, um verschiedene Getriebeverhältnisse modellieren zu können. Wie in Ab-

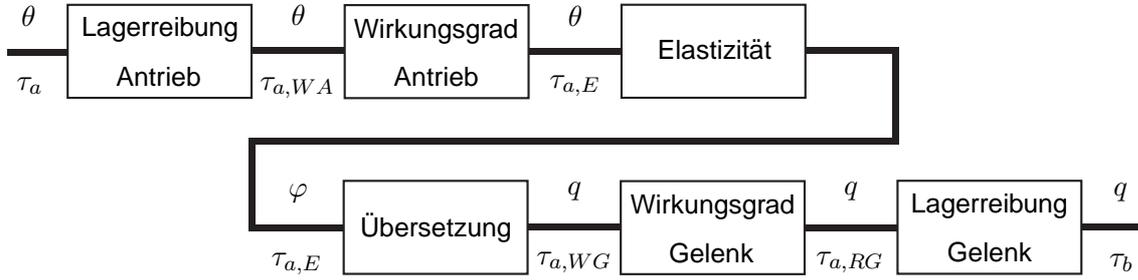


Abbildung 2.11: Darstellung des Gesamtmodells für den Antriebsstrang

bildung 2.11 zu sehen, treten im Antriebsstrang drei Koordinaten auf. Die Gelenkposition q_i , die Motorposition θ_i und der Winkel φ_i , welcher aufgrund der variablen Getriebeübersetzung entsteht. Die Variablen $\tau_{a,Element}$ bezeichnen jeweils das Moment am Flansch a des entsprechenden Elements. Damit ergibt sich für einen einzelnen Antriebsstrang das Gleichungssystem

$$\tau_a = r_A(\dot{\theta}, \tau_A, \tau_{a,WA}) + \tau_{a,WA} \quad (2.50a)$$

$$\tau_{a,WA} = \frac{1}{\eta_A(\tau_{a,WA}, \dot{\theta})} \tau_{a,E} \quad (2.50b)$$

$$\tau_{a,E} = -k(\varphi(q) - \theta) - d(\dot{\varphi}(q, \dot{q}) - \dot{\theta}) \quad (2.50c)$$

$$\tau_{a,WG} = \mathbf{u}(q) \tau_{a,E} \quad (2.50d)$$

$$\tau_{a,RG} = \eta_G(\tau_{a,WG}, \dot{q}) \tau_{a,WG} \quad (2.50e)$$

$$\tau_b = r_G(\dot{q}, \tau_G, \tau_{a,RG}) - \tau_{a,RG} \quad (2.50f)$$

$$\varphi = \mathbf{u}_0 q + \sum_{j=1}^{n_u} \frac{\hat{u}_j}{\bar{u}_j} \sin(\tilde{u}_j q + \bar{u}_j) + \mathbf{u}_C \quad (2.50g)$$

$$\mathbf{u} = \mathbf{u}_0 + \sum_{j=1}^{n_u} \hat{u}_j \cos(\tilde{u}_j q + \bar{u}_j) \quad (2.50h)$$

welches zugunsten einer einfacheren Zuordnung zu Abbildung 2.11 an dieser Stelle noch nicht weiter zusammengefasst ist.

2.3 Gesamtmodell des Roboters

Nun sollen die kompletten Gleichungen für die Dynamik des Roboters aufgestellt werden. Dazu wird das Antriebsstrangmodell (2.50) entsprechend Abbildung 2.2 in das MKS-Modell eingesetzt. Mit Definition von

$$H_G = \text{diag}(\eta_{G_1}(\tau_a, W_{G_1}, \dot{q}_1), \dots, \eta_{G_N}(\tau_a, W_{G_N}, \dot{q}_N)) \quad (2.51)$$

$$H_A = \text{diag}(\eta_{A_1}(\tau_a, W_{A_1}, \dot{\theta}_1), \dots, \eta_{A_N}(\tau_a, W_{A_N}, \dot{\theta}_N)) \quad (2.52)$$

$$U = \text{diag}(u_1(q_1), \dots, u_N(q_N)) \quad (2.53)$$

$$k = (k_1(\varphi_1 - \theta_1), \dots, k_N(\varphi_N - \theta_N)) \quad (2.54)$$

$$d = (d_1(\dot{\varphi}_1 - \dot{\theta}_1), \dots, d_N(\dot{\varphi}_1 - \dot{\theta}_N)) \quad (2.55)$$

$$\varphi = (\varphi_1(q_1), \dots, \varphi_N(q_N)) \quad (2.56)$$

$$\tau_{RA} = (r_{A_1}(\dot{\theta}_1, \tau_{A_1}, \tau_a, W_{A_1}), \dots, r_{A_N}(\dot{\theta}_N, \tau_{A_N}, \tau_a, W_{A_N})) \quad (2.57)$$

$$\tau_{RG} = (r_{G_1}(\dot{q}_1, \tau_{G_1}, \tau_a, W_{G_1}), \dots, r_{G_N}(\dot{q}_N, \tau_{G_N}, \tau_a, W_{G_N})) \quad (2.58)$$

ergibt sich das Gesamtmodell des Roboters zu

$$\begin{pmatrix} M_{Arm}(q) & S(q) \\ S(q)^T & J \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \dot{\theta} \end{pmatrix} + \begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} + \begin{pmatrix} H_G U (k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta})) + \tau_{RG} \\ -H_A^{-1} (k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta})) + \tau_{RA} \end{pmatrix} = \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix} \quad (2.59)$$

Diese Modellgleichungen dienen als Grundlage für die simulative Untersuchung des Robotermodells.

2.4 Modellierung mit Modelica

Im letzten Abschnitt wurde die Struktur des Modells aufgestellt. Für Untersuchungen mit dem Modell, insbesondere Simulationen, ist ein lauffähiger Code des realen Modells notwendig. Eine direkte Implementierung aus den Gleichungen ist jedoch nicht geeignet. Zur Modellierung und Simulation von MKS existiert eine Vielzahl von Werkzeugen wie z. B. ADAMS [MSC05] und SIMPACK [INT05]. Die Dynamik der Antriebsstränge kann in diesen noch vergleichsweise leicht hinzugefügt werden, die Möglichkeiten zur Anbindung von aufwändigen, modellbasierten Regelungen sind jedoch nicht zufriedenstellend. Werkzeuge wie Matlab/Simulink eignen sich zwar relativ gut zur Beschreibung der Regler, unterstützen aber die Simulation von mechanischen Systemen nur in eingeschränkter Weise. Aus diesen Gründen wird die Modellierung mit Hilfe der objektorientierten Modellierungssprache Modelica [Mod04] und dem Modelica-Simulationswerkzeug Dymola [Dyn04] durchgeführt, was sich insbesondere beim Entwurf von modellbasierten Reglern als vorteilhaft erweisen wird. Grundsätzlich werden dabei Modelle in Form von mathematischen Gleichungen unterstützt.

Im Folgenden werden die relevanten Eigenschaften von Modelica / Dymola kurz vorgestellt. Eine ausführliche Beschreibung kann zum Beispiel in [Ott00] gefunden werden.

Objektorientierung:

Zugrunde liegt die Idee aus einfachen Teilmodellen durch Verschaltung komplexe Gesamtmodelle aufbauen zu können. Zu diesem Zweck wird ein objektorientierter Entwurf unterstützt, welcher für den Aufbau der Teilmodelle die üblichen Vorteile einer objektorientierten Vorgehensweise wie Kapselung, Vererbung und Wiederverwendbarkeit von Modellen bietet und zudem den hierarchischen Aufbau der Modelle bis hin zum Gesamtmodell erlaubt. Dazu sind Standardkomponenten bereits in Modellbibliotheken verfügbar.

Konnektorbasierte Beschreibung der Modelle:

Die Verschaltung von Teilmodellen erfolgt über sogenannte Konnektoren, in denen die für die Interaktion zwischen den Modellen relevanten physikalischen Größen enthalten sind. Im Gegensatz zu Blockschaltbildern stellen die Verbindungen jedoch keine Signalflüsse, sondern physikalische Verbindungen dar. Aus mathematischer Sicht entspricht eine Verbindung zwischen Konnektoren einfach zusätzlichen Gleichungen. Zu diesem Zweck werden in den Konnektoren Potential- und Flussvariablen unterschieden, wobei die Potentialvariablen verbundener Konnektoren gleich sind und sich die Flussvariablen zu Null summieren.

Multidisziplinäre Modellierung:

Aufgrund der allgemeinen, gleichungsbasierten Vorgehensweise können multidisziplinäre Probleme, wie die Modellierung von Robotern, einheitlich behandelt werden.

Effizienter Code:

Modellierung in der beschriebenen Form führt im Allgemeinen auf sehr große, dünn besetzte Gleichungssysteme. Mit Dymola steht ein Werkzeug zur Verfügung, das die notwendigen leistungsfähigen Algorithmen zur symbolischen Gleichungsmanipulation beinhaltet und effizienten Simulationscode generiert. Wenn ein MKS bereits objektorientiert modelliert wird, bietet sich zum Aufstellen der Gleichungen der Newton-Euler Formalismus an, auf welchem die Mehrkörperdynamik in Modelica auch beruht. Aufgrund des bekannt effizienten Newton-Euler Formalismus in Kombination mit der symbolischen Manipulation des Gleichungssystems kann Dymola sehr schnellen Code erzeugen.

Direkte Behandlung schaltender bzw. diskreter Systeme:

Das Werkzeug erlaubt auf elegante Weise die gleichzeitige Behandlung von kontinuierlichen und diskreten Teilsystemen [OEM99], was z. B. für Simulationen mit diskreten Reglern, oder die Behandlung von Haftreibung notwendig ist.

Als Beispiel für ein Teilmodell wird hier der Modelica Code eines idealisierten Getriebes gezeigt.

```

model IdealGear "Idealisiertes Getriebe"
  extends Interfaces.TwoFlanges;
  parameter Real ratio "Übersetzung";

equation
  flange_a.phi = ratio*flange_b.phi;
  0 = ratio*flange_a.tau + flange_b.tau;
end IdealGear;

```

Das Modell *IdealGear* entsteht durch Ableitung von der Klasse *Interfaces.TwoFlanges*, der Basisklasse für rotatorische Antriebsstrangelemente mit zwei Konnektoren. Als Parameter der Übersetzung des Getriebes wird anschließend *ratio* deklariert. Das eigentliche Verhalten des Getriebes wird anhand der Gleichungen unter *equation* beschrieben, hier durch die Gleichungen für die Übersetzung der Winkel und Momente zwischen *flange_a* und *flange_b*. Wie in diesem Beispiel zu erkennen ist, enthalten die Teilmodelle vergleichsweise einfache Gleichungen. Durch die Verschaltung der Teilmodelle können aber sehr komplexe Gesamtmodelle erstellt werden.

In dieser Art und Weise wurde eine Modellbibliothek erstellt, welche alle notwendigen Teilmodelle zur Modellierung des Roboters enthält.

Die Modellierung des Mehrkörpersystems basiert auf der Modelica Mehrkörperbibliothek, deren Grundlagen in [Ott95] dargestellt sind und deren aktualisierte Version detailliert in [OEM03] beschrieben ist. Dabei handelt es sich aufgrund der objektorientierten Darstellung um einen rekursiven Ansatz, dessen Auswertung ähnlich wie das Newton-Euler Verfahren verläuft. Im Anhang B.1 sind die notwendigen Teilmodelle dargestellt. Entsprechend wurden Modelica-Modelle für den Antriebsstrang auf Basis der Überlegungen in Kapitel 2.2 erstellt. Diese sind im Anhang B.2 dargestellt.

2.5 Modellierung des KUKA KR15/2 Roboters

Im diesem Abschnitt wird das Modelica-Modell des in dieser Arbeit zur experimentellen Erprobung verwendeten KR15/2 entwickelt. Ausgehend von einer kurzen Beschreibung des Roboters werden die Komponenten des Robotermodells definiert und die Struktur des Robotermodells aufgebaut.

2.5.1 Beschreibung des KR15/2

Der KR15/2 ist ein 6-achsiger Knickarmroboter aus Aluminiumguss mit einer zulässigen Nutzlast von 15kg. Die Lage und Drehrichtung der Achsen sind in Abbildung 2.12 (a) zu sehen, die Abmessungen des resultierenden Arbeitsraumes in Abbildung 2.12 (b). Zusätzliche Informationen zur Kinematik können den Denavit-Hartenberg-Parametern [DH55] entnommen werden, welche in Tabelle 2.1 angegeben sind. Das Gesamtgewicht des Roboters liegt bei 205 kg. Detailliertere Angaben sind in [KUK96a] und [KUK96b] zu finden, woraus auch die Grafiken in Abbildung 2.12 entnommen sind. Die für den Antrieb eingesetzten Servomotoren

i	a_i [mm]	α_i [°]	d_i [mm]	θ_i [°]
1	300	90	675	θ_1
2	650	0	0	θ_2
3	-155	-90	0	θ_3
4	0	90	600	θ_4
5	0	-90	0	θ_5
6	0	0	140	θ_6

Tabelle 2.1: Denavit-Hartenberg-Parameter des KR15/2

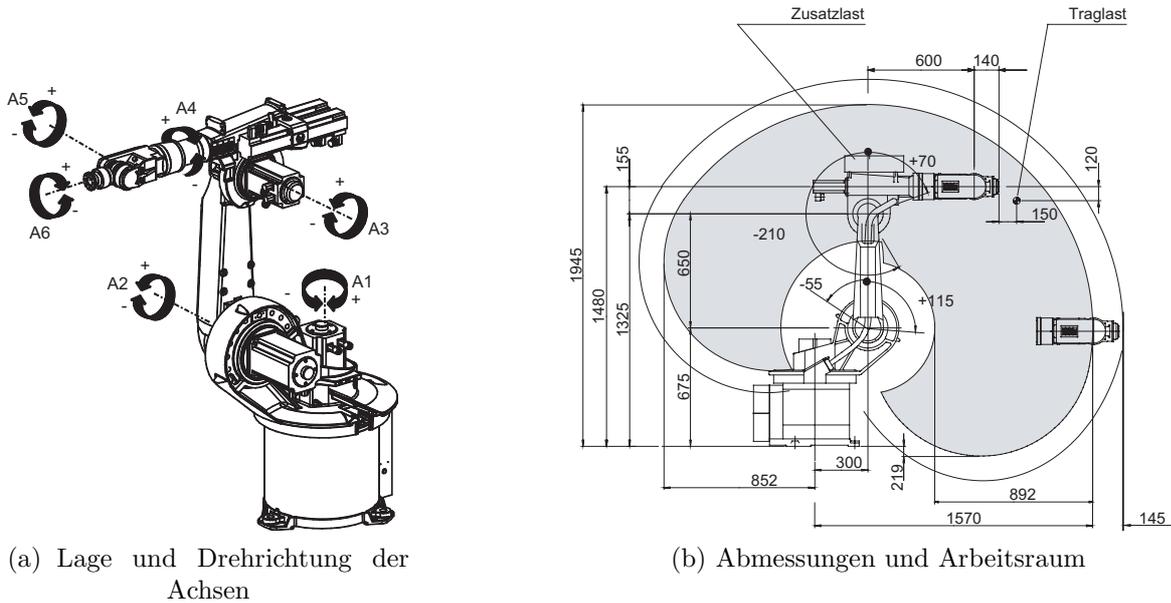


Abbildung 2.12: Aufbau des KR15/2

sind dauermagneterregte Synchronmotoren der Firma Siemens vom Typ 1FK6 [Sie04]. Deren Leistung wird für die ersten drei Achsen über Sumitomo Zykloidengetriebe vom Typ F2C-T übertragen [Sum96]. Bei den Achsen vier bis sechs kommen zusätzliche Übertragungselemente von den Motoren zur Achse hinzu. In der Leistungselektronik werden mit IGBT's bestückte Powermodule von Lenze verwendet.

2.5.2 Modelica-Modell des KR15/2

Im ersten Schritt wird die Struktur des KR15/2 in ein Modelica Modell abgebildet, wozu die bereits definierten, allgemeinen Modelica-Objekte verwendet werden können. Die Parametrierung des Robotermodells erfolgt später und wird in Kapitel 3 beschrieben.

Der Roboter besteht aus sieben Gliedern und einer zusätzlichen Traglast. Diese werden von der Klasse *Glied* abgeleitet, wobei die allgemeinen Parameter der Abmessungen ${}^A r_{AB}$, r_{CM} und der Masseneigenschaften m und ${}^A I^A$ in der Klasse *Glied* durch die Werte der Roboterglieder ersetzt werden. Dabei werden zur Vereinfachung nur die kinematisch und dynamisch relevanten Daten benutzt, wodurch z. B. Trägheiten um Achsen, um welche sich das Glied aufgrund der Kinematik des Roboters nicht bewegen kann, zu Null gesetzt werden. Die resultierenden Modelle für die Glieder des KR15/2 sind im Anhang B.4 dargestellt.

Auf die gleiche Weise werden die Modelle der Antriebe erstellt. Dazu werden ein neues Rotor-Objekt und ein neues Getriebe-Objekt erstellt, deren Aufbau im Anhang B.5 gezeigt ist. Von diesen werden jeweils wieder durch Parametrisierung die Modelle der einzelnen Antriebsstränge abgeleitet. Insgesamt entsteht für den Roboter damit das in Abbildung 2.13 als Objektdiagramm dargestellte Modelica-Modell. Das gesamte Robotermodell hat 24 Zustände und ca. 1000 nicht triviale Gleichungen. Der von Dymola anhand des Modells erzeugte Simulationscode der direkten Dynamik benötigt dabei mit dem Integrator Dassl für die Berechnung

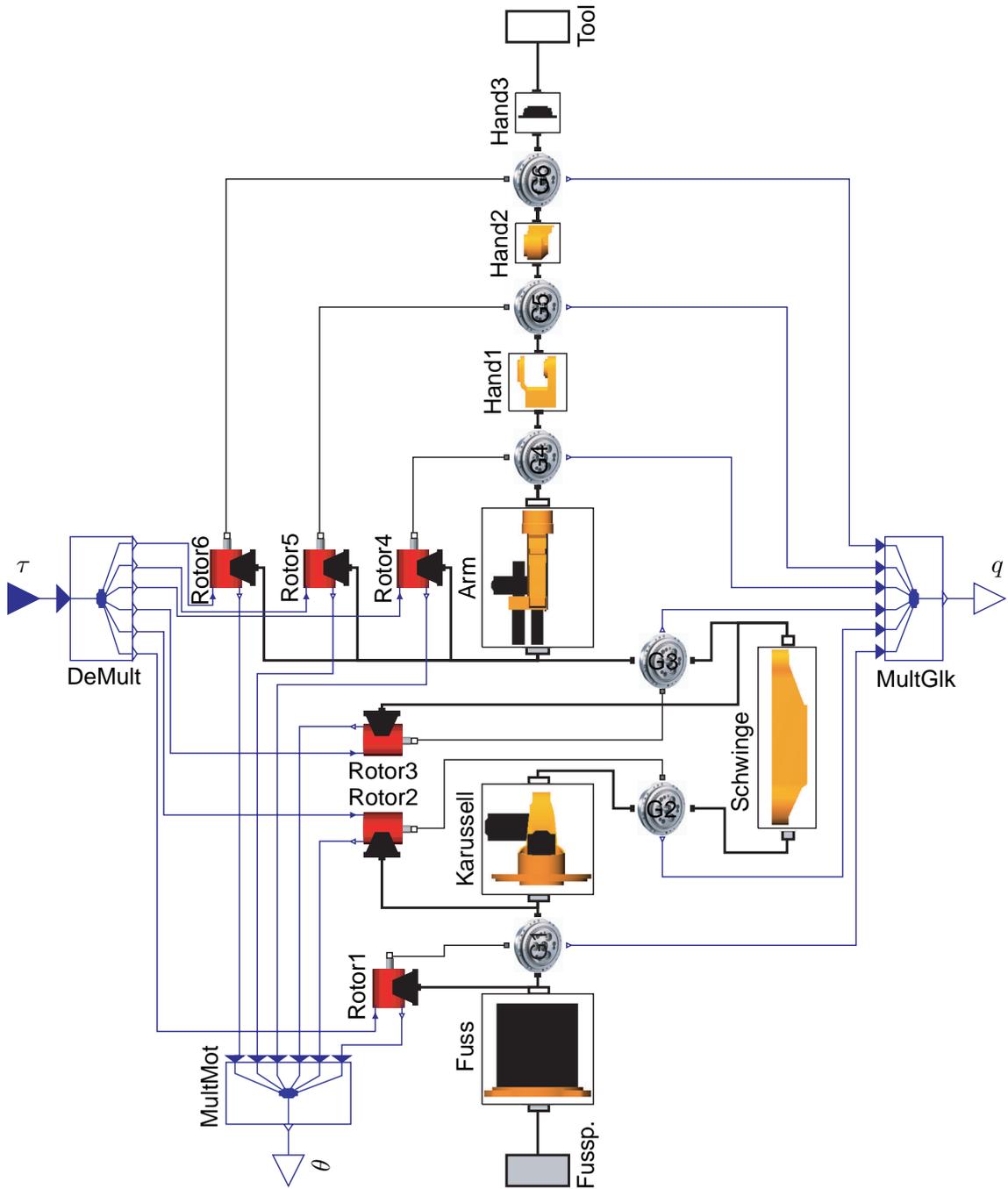


Abbildung 2.13: Gesamtmodell des KR15/2 als Modelica Objektdiagramm

einer Bewegung mit einer Dauer von einer Sekunde auf einem 1.7 GHz Pentium Prozessor etwa 2s. Ein Großteil der Rechenleistung wird dabei aufgrund von schaltenden Kennlinien verbraucht, welche lokal eine sehr kleine Integrationsschrittweite erfordern. Wird statt dessen ein Integrator mit fester Schrittweite, wie z. B. das Runge-Kutta Verfahren 4. Ordnung mit einer Millisekunde Schrittweite verwendet, dann sinkt die Berechnungsdauer auf etwa 0.4s. Dies geschieht jedoch auf Kosten der Genauigkeit der Lösung.

Kapitel 3

Identifikation der Modellparameter

Im vorigen Kapitel wurde die Modellierung von Robotern mit elastischen Gelenken anhand deren struktureller Eigenschaften durchgeführt. Mit Hilfe der dabei abgeleiteten Gleichungen sollte sich eine große Klasse von Robotern detailliert beschreiben lassen. Insbesondere wurde die Struktur des KR15/2 in ein Modelica-Modell abgebildet. Zur Komplettierung des Modells aus Kapitel 2 müssen nun noch die zugehörigen Modellparameter wie z. B. Massen und Steifigkeiten bestimmt werden.

3.1 Methoden zur Parameteridentifikation

Ein Robotermodell besitzt eine Vielzahl von Parametern für deren Identifikation in der Literatur bereits einige Arbeiten vorliegen. Stellvertretend seien hier [KS96] für einen theoretischen Überblick und [Höl94, Ös02, Gro03] für praktische Arbeiten genannt.

Mit der in dieser Arbeit verwendeten Art der Modellierung besteht das Problem der Identifikation in der Bestimmung physikalischer Parameter eines nichtlinearen Systems. Für die Identifikation von komplexen Systemen ist es häufig wünschenswert, einzelne Komponenten wie z. B. Motoren, Getriebe oder Glieder einzeln zu vermessen, um das Problem in kleinere Teilprobleme aufzuteilen. Ein solches Vorgehen wird in der vorliegenden Arbeit jedoch nicht untersucht, da dazu zum einen der Roboter demontiert werden muss und zum anderen entsprechende Prüfstände, z. B. für Motoren und Getriebe, benötigt werden. Aufgrund des Aufwands zur einzelnen Vermessung von Komponenten erscheint dieses Vorgehen als nur dann gerechtfertigt wenn, wie in [Abe03], strukturelle Modelleigenschaften der Komponenten untersucht werden sollen. Damit kommen in dieser Arbeit nur Methoden in Frage, bei denen der Roboter als Gesamtsystem betrieben und identifiziert wird.

Eine Hauptaufgabe für die Identifikation stellt in der Regel die Bestimmung der Parameter des Mehrkörpersystems dar. Die geometrischen Parameter können dabei z. B. über eine Regressormatrix iterativ mit Hilfe der sogenannten *kinematischen Kalibrierung* ermittelt werden. Dieses Verfahren wurde für die hier relevanten Industrieroboter beispielsweise in [Wie01] angewendet. Zur Bestimmung der für die Dynamik des Mehrkörpersystems relevanten Masseigenschaften werden die Gleichungen des Roboters gewöhnlich so formuliert, dass die gesuchten Parameter linear eingehen. Mit dieser Formulierung können die gesuchten Parameter dann unter Verwendung von Least-Squares-Methoden direkt bestimmt werden. Einen

Überblick über die Identifikation der Parameter des Mehrkörpersystems und eine Sammlung weiterführender Literatur gibt z. B. [KS96]. Das Vorgehen zur Umformung der Bewegungsgleichungen auf eine Form die linear in den Parametern ist, sowie die Reduktion auf den minimalen Parametersatz ist z. B. in [Gro03] beschrieben. Dort wurde auch der aus 21 Parametern bestehende minimale Parametersatz für den hier relevanten KR15/2 aufgestellt und eine Identifikation des Mehrkörpersystems durchgeführt.

In der vorliegenden Arbeit war die Anwendung dieser Methoden jedoch nicht notwendig, da die Parameterwerte freundlicherweise vom Hersteller KUKA zur Verfügung gestellt wurden. Die Denavit-Hartenberg-Parameter z. B. wurden bereits in Kapitel 2.5 angegeben. Zudem konnten eigene experimentelle Tests anhand von Halte- und Beschleunigungsversuchen die Parameterwerte bestätigen. Damit verbleiben für die Identifikation die Parameter der Antriebsstränge.

Eine ausführliche Übersicht über Identifikationsmethoden für Systeme mit verschiedenen Eigenschaften und deren Anwendung ist in [Ise92a] und [Ise92b] enthalten. Das entwickelte Robotermodell ist nichtlinear, nicht stetig differenzierbar und nichtlinear in den Parametern. Zur Identifikation solcher Systeme werden gewöhnlich iterative Methoden verwendet. Um eine möglichst große Klasse von Modellen identifizieren zu können, werden die Antriebsstrangparameter durch Parameteroptimierung auf der Basis von nichtlinearen Simulationsmodellen identifiziert. Dieses Vorgehen ist aus [Ise92b] auch als Parametereinstellung durch Modellabgleich bekannt und wird in Kapitel 3.2 näher beschrieben. In Kapitel 3.3 wird anhand des KR15/2 die Anwendung dieser Methode demonstriert. Um gute Konvergenzeigenschaften der enthaltenen Parameteroptimierung sicherzustellen und um eine geeignete Parametrierung beispielsweise der Reibkennlinien zu finden, werden dabei vor der Optimierung zusätzliche Versuche durchgeführt, welche eine möglichst direkte Schätzung für einzelne Parameter erlauben. Diese sind in Abschnitt 3.3.2 ausführlicher dargestellt.

3.2 Parameteridentifikation mit nichtlinearen Simulationsmodellen

Aufgrund der bekannten Parameter des Mehrkörpersystems bietet sich eine achsweise Identifikation an. Auch für die Achsen liegt jedoch mit den Gleichungen (2.50) ein nichtlineares, parametrisches Modell vor, welches sich in der Regel nicht als linear in seinen Parametern darstellen lassen wird. Um die Parameter einer möglichst großen Klasse von Modellen identifizieren zu können, werden diese anhand von Simulationen mit dem nichtlinearen Modell eingestellt, was in Abbildung 3.1 dargestellt ist. Als Anregung des Roboters werden deterministische Signale verwendet. Die Auswertung der Messungen erfolgt offline, wobei das Modell mit Hilfe eines Optimierers iterativ verbessert wird. Dazu wird zunächst das nichtlineare Modell mit den aktuellen Parametern und den gegebenen Eingangsgrößen simuliert. Die Simulationsergebnisse werden mit den vorverarbeiteten Messungen am Roboter verglichen und Kriterien für die Güte des Modells bestimmt. Diese Kriterien können z. B. Normen der Abweichungen zwischen Messgrößen und Simulationsergebnissen sein. Auf Basis solcher Kriterien kann nun ein mehrzieliges Optimierungsverfahren die Parameter des Modells optimieren, wofür üblicherweise noch eine Gewichtung der Kriterien zueinander erfolgen muss.

Vorteilhaft bei dieser Art der Parameteridentifikation ist sicherlich die universelle Anwendbarkeit der Methode auf verschiedenste Modelle und Parametrierungen. Der wesentliche Nachteil

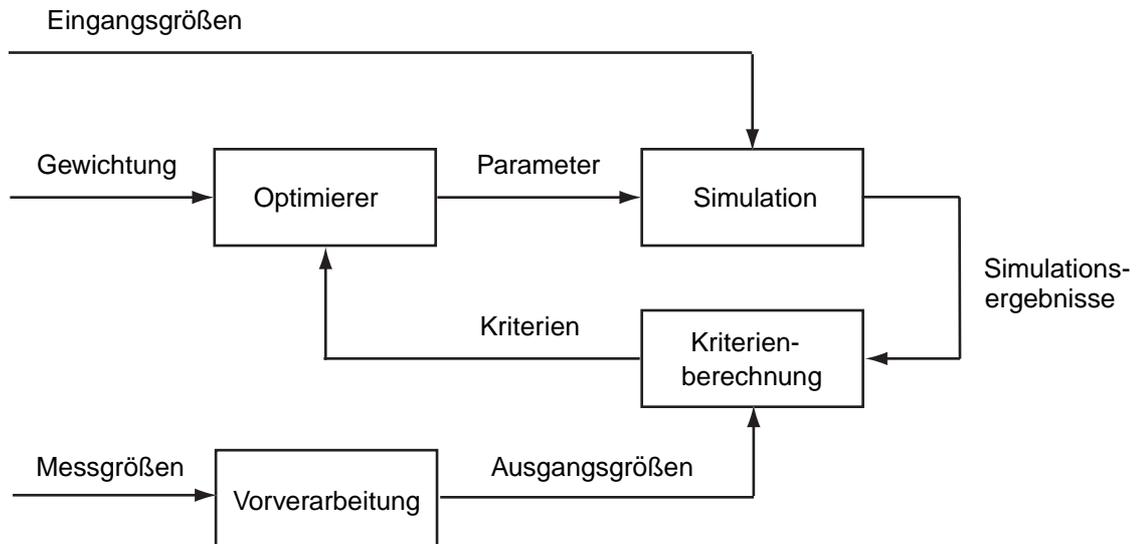


Abbildung 3.1: Parameteridentifikation mit nichtlinearem Simulationsmodell

dagegen ist, dass für die Identifikation ein nichtlineares Optimierungsproblem gelöst werden muss, welches in den meisten Fällen nicht konvex sein wird. Damit besteht die Gefahr, bei der Optimierung nur ein lokales Minimum zu finden. Um dieses Risiko zu reduzieren, können eine Reihe von Maßnahmen getroffen werden. Dazu zählen:

Wahl guter Startwerte und Parameterbereiche:

Um gute Startwerte zu erhalten, welche bereits in der Nähe der echten Werte liegen, wird versucht, einzelne Parameter in Vorversuchen möglichst direkt zu messen. Wird außerdem noch die Genauigkeit der Messung berücksichtigt, dann kann zusätzlich der Wertebereich der Parameter für die Optimierung eingeschränkt werden. Ein mögliches Vorgehen dazu wird in Abschnitt 3.3.2 angegeben.

Variation der Startwerte:

Optimierungen mit verschiedenen Startwerten der Parameter erhöhen die Chance, das globale Minimum zu finden. Dieser Weg ist allerdings sehr rechenintensiv, womit die Anzahl der möglichen Variationen begrenzt ist.

Verwendung geeigneter Kriterien:

Die Wahl geeigneter Kriterien spielt eine wichtige Rolle bei der Optimierung. Aufgrund verrauschter Messdaten eignet sich beispielsweise die ∞ -Norm des Modellfehlers schlecht. Besser geeignet sind glättende Funktionen wie z. B. die 2-Norm.

Häufig gibt es nicht nur eine Möglichkeit zur Wahl der Ein- und Ausgangsgrößen, wodurch sich auch das Simulationsmodell ändert. Somit besteht die Möglichkeit über ein geeignetes Simulationsmodell den Verlauf der Optimierung günstig zu beeinflussen. Beispielsweise kann das Modell für eine Kombination aus Ein- und Ausgängen instabil und damit extrem sensitiv auf einzelne Parameter und Störungen sein, mit anderen Ein- und Ausgängen dagegen stabil. Die Wahl geeigneter Ein- und Ausgänge und damit des Simulationsmodells und der

notwendigen Sensoren hängt naturgemäß stark vom zu identifizierenden System ab. Das Simulationsmodell zur Identifikation der in dieser Arbeit verwendeten Achsmodelle wird in Kapitel 3.3 angegeben.

3.3 Identifikation der Parameter des KR15/2

Problematisch bei der Identifikation des Robotermodells ist, dass das Robotersystem mit den Motormomenten als Eingang instabil ist. Damit ist eine Identifikation ohne Regler mit Schwierigkeiten verbunden, da z. B. durch geeignete Wahl der Eingänge sichergestellt werden muss, dass keine Kollisionen auftreten, wofür aber bereits ein Modell benötigt wird. Da die Identifikation durch Modellabgleich aber auch mit einem Regler möglich ist, werden die Versuche am geregelten Roboter durchgeführt. Nachteilig an diesem Vorgehen ist, dass der Regler unter Umständen Teile der Dynamik unterdrücken kann, welche dann nur noch schlecht identifiziert werden können. Um dieses Problem zu entschärfen, werden für die Identifikation achsweise PID Positionsregler für die Motoren gewählt. Bei hohen Verstärkungen folgen die Motoren mit diesen Reglern der Sollposition sehr gut, womit die Achse und damit der Roboter mit einer großen Bandbreite angeregt werden kann.

3.3.1 Sensorausstattung

Für die Identifikation werden zusätzlich zu den bereits im Roboter eingebauten Sensoren weitere externe Sensoren verwendet. Dabei kommen nur Sensoren in Frage, die sich ohne mechanische Veränderung des Roboters nachträglich anbringen lassen. Für die Identifikation wurde in dieser Arbeit die folgende Sensorik verwendet:

Resolver:

Diese bereits in den Motoren eingebauten Sensoren messen die Positionen der Motoren mit hoher Genauigkeit und Bandbreite. Dazu wird die in den Spulen des Resolvers induzierte, positionsabhängige Spannung ausgewertet.

Optisches Koordinatenmessgerät:

Zur Vermessung der Bewegung des Roboterarmes wird das optische Koordinatenmessgerät Roboscope 6D der Firma Krypton [Kry01] eingesetzt, mit welchem sich die Koordinaten von frei am Arm verteilbaren Infrarot-Leuchtdioden durch Triangulation messen lassen. In einem großen Messvolumen werden Genauigkeiten von ca. 0,1 mm erreicht.

Beschleunigungssensoren:

Zur Vermessung der Beschleunigung des Armes werden beim DLR aufgebaute Beschleunigungssensoren auf der Basis von Sensorchips mit mikromechanischen Sensoren verwendet [ST02]. Diese erlauben die dreidimensionale Messung der Beschleunigung des Sensors mit hoher Bandbreite.

Drehratensensor:

Zur Erfassung von Drehbewegungen wird der Drehratensensor BEI Gyrochip II [BT05] der Firma Systron Donner verwendet. Dieser Sensor bietet gute dynamische Eigenschaften und zeigt ausreichend wenig Rauschen und Drift.

Kraftsensor:

Für die Messung von Kräften, welche in den Arm eingeleitet werden, wird ein eindimensionaler Kraftsensor KT1500 [MEG05] der Firma Megatron benutzt, welcher mit Dehnmessstreifen als S-Beam aufgebaut ist.

Je nach Experiment wurden verschiedene Kombinationen und Anordnungen der Sensoren benutzt. Die zur Identifikation durchgeführten Experimente werden im Folgenden beschrieben.

3.3.2 Direkte Messung einzelner Parameter

Sollen einzelne Parameter direkt bestimmt werden, dann ist es sinnvoll möglichst viele Größen auf Null zu halten oder durch vorherige Experimente zu bestimmen. Aus diesem Grund wird zur Identifikation der Achsmodelle immer nur die zu identifizierende Achse bewegt. Die anderen Achsen werden so in eine feste Position gebracht, dass Kopplungen mit der zu identifizierenden Achse minimiert werden.

Um vorab möglichst wenig Modellwissen zu benötigen, werden die Versuche in der folgenden Reihenfolge durchgeführt, wobei immer mehr Terme der Bewegungsgleichungen ungleich Null sind:

1. Konstante Position
2. Konstante Geschwindigkeit
3. Konstante Beschleunigung
4. Freie Bewegung.

Im Folgenden wird das Vorgehen am Beispiel von Messungen aus der Identifikation der ersten Achse des KR15/2 gezeigt.

Steifigkeit

Zur Vermessung der Gelenksteifigkeit bieten sich Zugversuche an. Grundsätzlich ist eine Bestimmung auch über die Frequenz der Gelenkschwingung möglich. In diesem Fall muss aber die Roboterdynamik mit berücksichtigt werden. Außerdem ist die Auswertung bei nichtlinearen Elastizitäten schwieriger. Da die Bestimmung mit Hilfe von Zugversuchen zudem kein Modell der Dynamik benötigt, wird die Vermessung auf diese Weise durchgeführt. Dazu wird der Motor mit Hilfe der eingebauten Bremsen bei $\theta_1 = 0$ gehalten. Stationär vereinfacht sich damit die Bewegungsgleichung des Armes (2.59) zu

$$\mathbf{u}_1(q_1)k_1(\varphi_1(q_1)) + \tau_{RG1} = \tau_{Ext1}. \quad (3.1)$$

Für eine Schätzung der Steifigkeit ist es sicher ausreichend, anstatt der positionsabhängigen Übersetzung die Basisübersetzung zu verwenden. Damit ergibt sich aus Gleichung (3.1):

$$\mathbf{u}_{01}k_1(\mathbf{u}_{01}q_1) + \tau_{RG1} = \tau_{Ext1}. \quad (3.2)$$

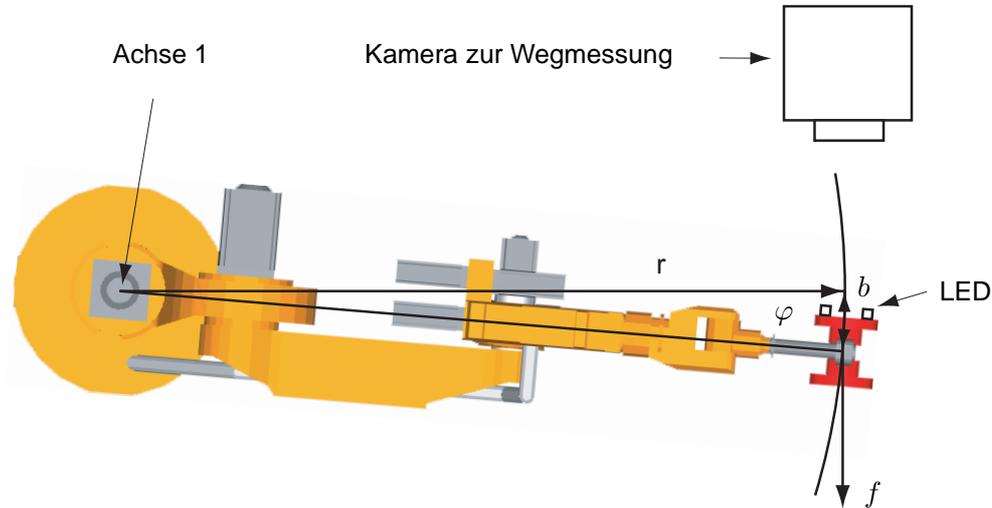


Abbildung 3.2: Versuchsaufbau zur Messung der Steifigkeit der ersten Achse

Bei vorgegebenem externen Moment kann damit die Steifigkeit k_1 durch Messung von q_1 geschätzt werden, wobei die gelenkseitige Haftreibung als Fehler eingeht.

Im Versuch wird daher über einen Kraftsensor ein externes Moment auf das Gelenk aufgebracht und dessen Torsion über ein Wegmesssystem aufgenommen. Der entsprechende Aufbau zur Messung der Steifigkeit von Achse 1 ist in Abbildung 3.2 dargestellt. Bei kleinen Torsionen, wie sie bei Industrierobotern auftreten, kann der Winkelfehler der aufgebrachten Kraft vernachlässigt werden. Damit ergeben sich Achsmoment und Torsion näherungsweise zu:

$$\begin{aligned}\tau_{Ext1} &= fr \\ q_1 &= \frac{b}{r}\end{aligned}$$

Mit diesen Zusammenhängen kann die Steifigkeitskennlinie des Getriebes von Achse 1 vermessen werden. Diese ist, bezogen auf das maximale Getriebemoment des Gelenkes τ_{Gmax1} , in Abbildung 3.3 dargestellt und zeigt einen im Wesentlichen linearen Verlauf. Eine sehr gute Approximation der Messwerte erhält man mit einer Kennlinie nach Gleichung (2.2.1), welche ebenfalls in Abbildung 3.3 dargestellt ist. Die Vermessung der anderen Achsen kann in ähnlicher Weise erfolgen. Falls das Moment aufgrund der Schwerkraft g_{Arm} für eine Achse ungleich Null ist, muss es zuvor durch Messung der Haltemomente in verschiedenen Positionen bestimmt werden. Ferner kann eine Messung der Verformung relativ zu einem anderen Punkt auf dem Arm notwendig sein, z. B. um die Steifigkeiten der zweiten und dritten Achse trennen zu können.

Reibung

Reibung kann, entsprechend dem Antriebsmodell (2.50) in Form von Antriebsverlusten, Gelenkverlusten und Gelenkdämpfung auftreten. Experimentell ist es schwierig die einzelnen Effekte zu trennen. Es ist jedoch trotzdem eine Abschätzung der Größe der Reibmomente möglich. Dazu wird die Achse zunächst mit verschiedenen, konstanten Geschwindigkeiten

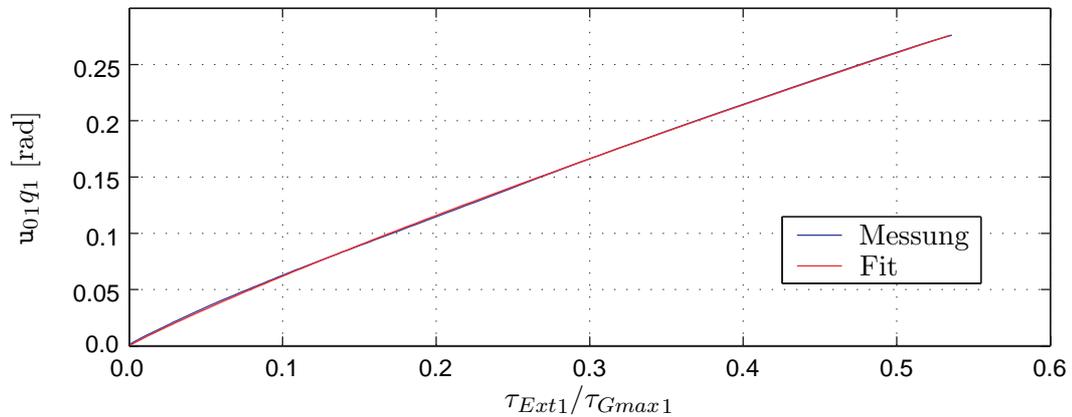


Abbildung 3.3: Steifigkeitskennlinie der ersten Achse

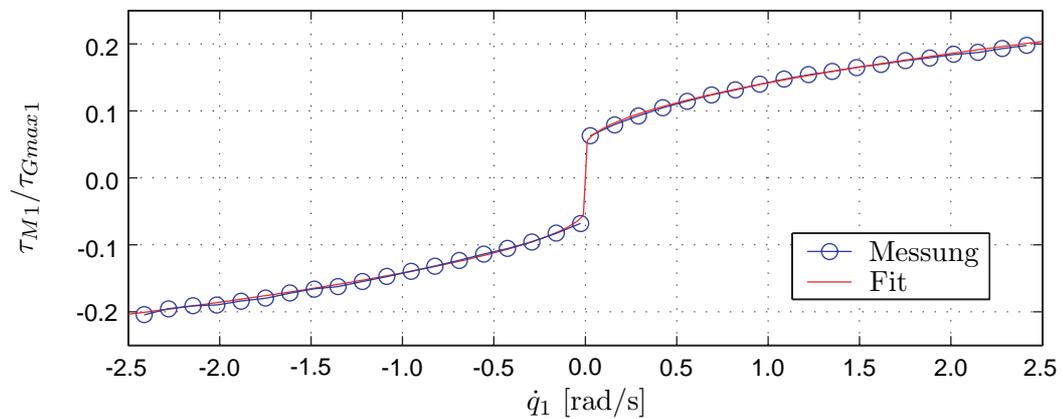


Abbildung 3.4: Gesamte Reibung der ersten Achse bei konstanter Drehzahl

bewegt. Nach dem Abklingen von Einschwingvorgängen kann unter Verwendung der Bewegungsgleichungen (2.59) der folgende Zusammenhang angegeben werden:

$$\frac{r_{G1}}{\eta_{A1}\eta_{G1}\mathbf{u}_{01}} + r_{A1} = \tau_{M1} = \tau_{Rvk1} \quad (3.3)$$

Dabei wurde die Übersetzung wieder durch die Basisübersetzung approximiert. Da nur eine Achse bewegt wird, sind zudem die Coriolis- und Zentrifugalkräfte auf diese Achse Null. Damit gibt das Motormoment die Summe aus Antriebs- und Gelenkverlusten der Achse an. Die Motormomente der ersten Achse, bezogen auf das maximale Getriebemoment τ_{Gmax1} des Gelenkes, welche für eine Bewegung mit konstanter Geschwindigkeit benötigt werden, sind in Abbildung 3.4 aufgetragen. Die Form der Kennlinie kann gut durch eine Kombination aus Coulombreibung nach Gleichung (2.38) und viskoser Reibung nach Gleichung (2.40) mit

$$\tau_{Rvk1} = \tau_{C1}\text{sgn}(\dot{q}_1) + \nu_1|\dot{q}_1|^{\delta\nu_1}\text{sgn}(\dot{q}_1) \quad (3.4)$$

angenähert werden, welche ebenfalls in Abbildung 3.4 dargestellt ist.

Um die Reibung besser aufteilen zu können, sind zusätzliche Versuche notwendig. Für eine Abschätzung der Gelenkverluste wird der Motor durch die Bremse gehalten und das Abklingen

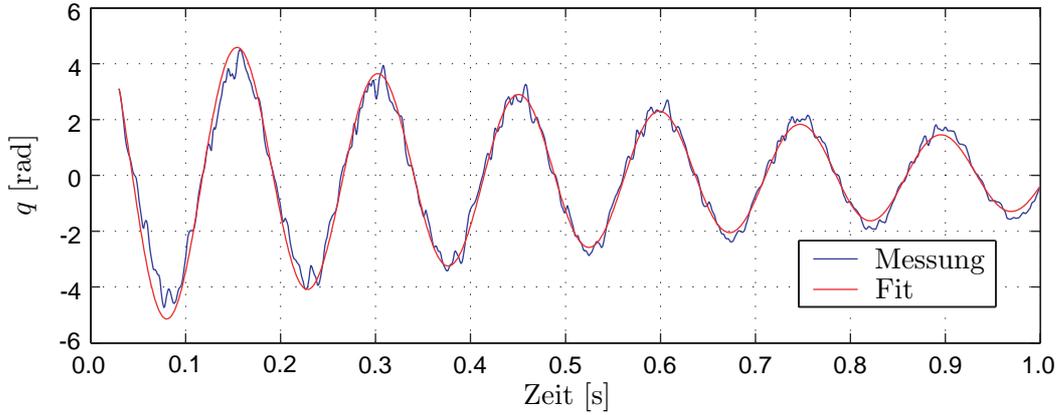


Abbildung 3.5: Einschwingvorgang der ersten Achse zur Abschätzung der Dämpfung

der Schwingung des Gelenkes nach einer externen Störung betrachtet, wie es in Abbildung 3.5 dargestellt ist. Zur Schätzung der Verluste wird für den Einschwingvorgang das Modell einer schwingungsfähigen Masse mit der bereits ermittelten Steifigkeit und linearer Reibung angesetzt, welche die Summe aus Gelenkdämpfung und Gelenkreibung approximiert. Der Wert der Reibung wird durch Abgleich des Einschwingverhaltens bestimmt, welches für das abgegliche Modell ebenfalls in Abbildung 3.5 dargestellt ist. Damit lässt sich zum Vergleich mit der Reibungskurve in Abbildung 3.4 eine auf das maximale Getriebemoment bezogene Reibung von

$$\frac{\tau_{RE1}}{\tau_{Gmax1}} = 0.37 \frac{s}{rad}$$

angeben. Man erkennt, dass ein Großteil dieser Verluste von der Dämpfung im Getriebe kommen muss, da die durch $(\tau_{RE1}/\tau_{Gmax1}) \dot{q}$ definierte Gerade eine deutlich höhere Steigung als die Kurve in Abbildung 3.4 aufweist.

Eine weitere Aufteilung der Reibung könnte durch Fixierung des Armes und Analyse des Einschwingverhaltens des Motors gewonnen werden. Dieser Versuch erfordert jedoch einen aufwändigeren mechanischen Aufbau zur Fixierung des Armes und wurde deshalb nicht durchgeführt. Wie sich bei der Identifikation des Gesamtmodells zeigen wird, sind die beschriebenen Messungen zu den drehzahlabhängigen Verlusten ausreichend.

Die Abschätzung der lastabhängigen Verluste erfordert weiteres Vorwissen, da für die erste Achse eine Last nur durch Beschleunigung des Armes erzeugt werden kann. Dementsprechend werden Versuche mit unterschiedlichen, konstanten Beschleunigungen gefahren. Die Beschleunigungsprofile sind dabei so zu wählen, dass möglichst wenig Schwingungen angeregt werden. Unter Vernachlässigung der Wechselwirkungen mit den nicht bewegten Achsen ergibt sich für das Motormoment:

$$\frac{r_{G1} + M_{Arm1,1} \ddot{q}_1}{\eta_{A1} \eta_{G1} u_{01}} + r_{A1} + J_{1,1} \ddot{\theta}_1 = \tau_{M1}. \quad (3.5)$$

Die Motorbeschleunigung $\ddot{\theta}_1$ kann durch Messung der Motorposition und numerisches Differenzieren bestimmt werden, die Gelenkbeschleunigung \ddot{q}_1 mit Hilfe eines Beschleunigungssensors. Unter Verwendung der bekannten geschwindigkeitsabhängigen Reibung aus Gleichung

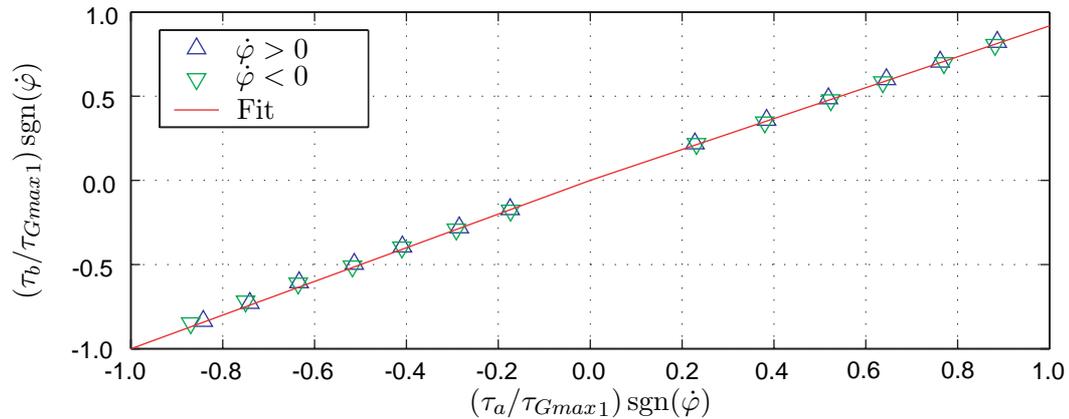


Abbildung 3.6: Lastabhängige Verluste der ersten Achse des KR15/2

(3.4) kann Gleichung (3.5) umgeformt werden zu:

$$\begin{aligned} (\tau_{M1} - J_{1,1}\ddot{\theta}_1 - \tau_{Rvk}) \mathbf{u}_{01} \eta_{A1} \eta_{G1} &= M_{Arm1,1} \ddot{q}_1 \\ \tau_a \eta_{A1} \eta_{G1} &= \tau_b \end{aligned}$$

Damit kann das Produkt der Wirkungsgrade bestimmt werden. Die Ergebnisse der Messungen mit verschiedenen Beschleunigungen und Drehrichtungen zeigt Abbildung 3.6. Die Bestimmung der Wirkungsgrade für positiven und negativen Leistungsfluss erfolgt durch zwei Ausgleichsgeraden, welche ebenfalls in Abbildung 3.6 dargestellt sind. Für eine positive Leistungsflussrichtung ergibt sich ein Wirkungsgrad von $\eta_{Gp}\eta_{Ap} = 0.9$, bei negativem Leistungsfluss von $\eta_{Gn}\eta_{An} = 1$.

Ein ähnliches Vorgehen ist auch bei den anderen Achsen möglich. Auch hier ist es für schwerkraftbelastete Achsen notwendig, zunächst die Momente aufgrund der Schwerkraft zu bestimmen. Bei der Untersuchung des Einschwingverhaltens kann bei der Bestimmung der Steifigkeit eine relative Messung notwendig sein, um die Einflüsse einzelner Achsen zu trennen. Vorteilhaft kann die Schwerkraft dagegen bei der Bestimmung des Wirkungsgrades sein, da sie eine langsam über dem Gelenkwinkel veränderliche Last erzeugt.

Übersetzung

Zur Bestimmung der positionsabhängigen Übersetzung aus Abschnitt 2.2.1 sind sehr präzise Messungen der Antriebs- und Gelenkposition bei einer langsamen Bewegung erforderlich. Mit dem vorhandenen Koordinatenmessgerät war die Messung des Gelenkwinkels jedoch nicht in der notwendigen Genauigkeit möglich. Um dennoch Aussagen über die Drehungleichförmigkeiten zu gewinnen, wurde die Drehrate des Gelenks mit einem Drehratensensor gemessen und aufintegriert. Vorteilhaft bei diesem Vorgehen ist die hohe lokale Genauigkeit, nachteilig dagegen der Random-Walk des Signals, welcher durch die Integration des verrauschten Sensorsignals entsteht. Damit kann die Bewegung aber nicht mehr sehr langsam ausgeführt werden, da sonst der Random-Walk zu groß wird. Wird die Bewegung aber schneller ausgeführt, dann ist, wie auch schon Abbildung 2.6 zeigt, die Amplitude der Schwingungen des Getriebes, welche durch die Drehungleichförmigkeiten verursacht werden, abhängig von der Drehzahl. Die

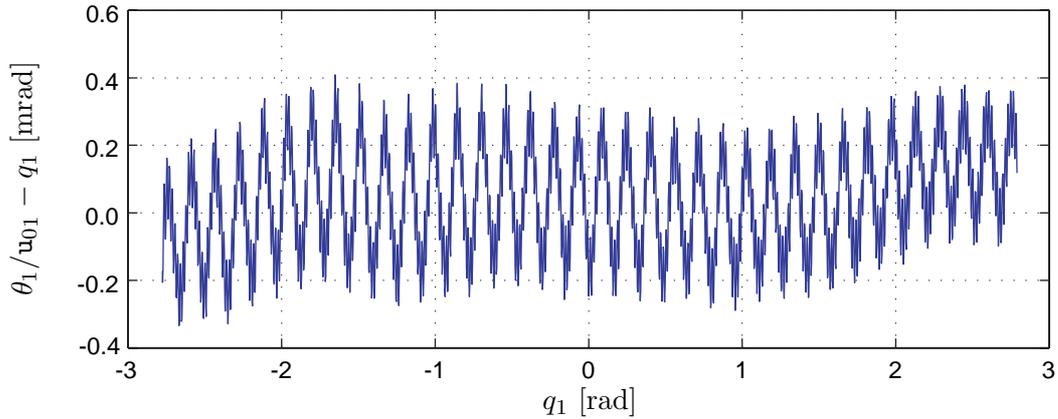


Abbildung 3.7: Gemessene Drehungleichförmigkeiten der ersten Achse

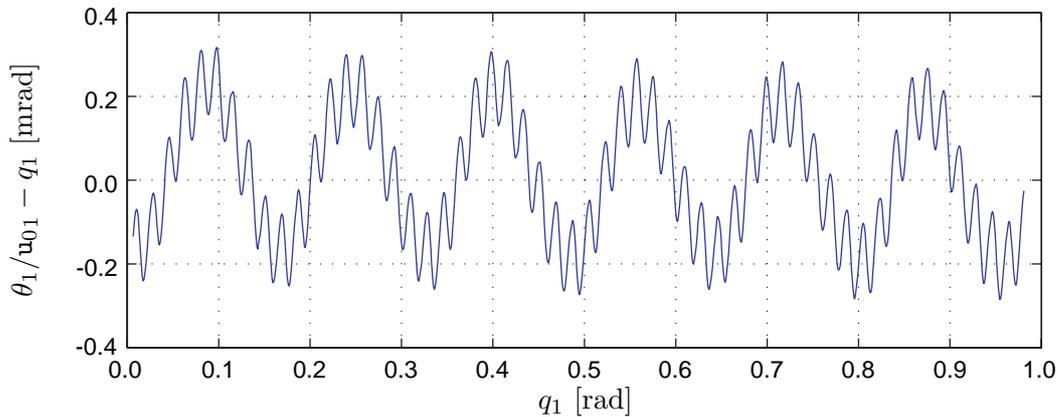


Abbildung 3.8: Ausschnitt aus den Messdaten zur Drehungleichförmigkeit

Ursache dafür ist, dass die durch die Drehungleichförmigkeit angeregten Schwingungen, deren Frequenz proportional zur Drehzahl ist, entsprechend der Eigenfrequenz des elastischen Getriebes verstärkt oder gedämpft werden. Durch Variation der Drehzahl können die Frequenzen der Drehungleichförmigkeit damit aber auch deutlicher sichtbar gemacht werden. Bei günstiger Wahl der Drehzahl erhält man den in Abbildung 3.7 dargestellten Verlauf des Winkelfehlers über dem Gelenkwinkel.

Während der niederfrequente Anteil des Signals aufgrund des bereits diskutierten Random-Walk nicht zuverlässig ist, können doch dominante Frequenzen identifiziert werden. Abbildung 3.8 zeigt zur Verdeutlichung einen Ausschnitt aus Abbildung 3.7.

Hier sind deutlich zwei überlagerte Schwingungen erkennbar, deren Frequenzen bei $401/(2\pi)$ und $3601/(2\pi)$ liegen. Die Amplituden und Phasenlagen der Schwingungen können nicht direkt aus den Abbildungen entnommen werden, da sie auch von den restlichen Parametern des Getriebes abhängen. Sie werden daher im Laufe der Parameteroptimierung eingestellt.

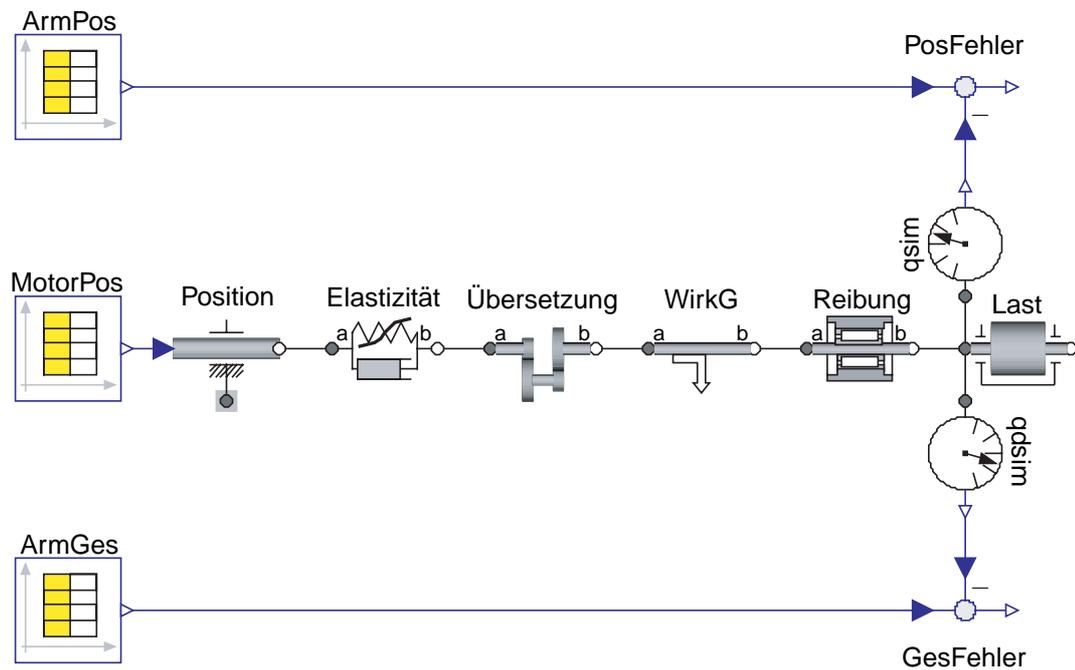


Abbildung 3.9: Modelica-Modell des Antriebsstranges für die Identifikation

3.3.3 Simulationsmodell zur Parameteridentifikation

Wie bereits bei den direkten Messungen wird auch bei der Optimierung der Antriebsstrangparameter der Achse nur eine Achse bewegt. Durch Fusion der Daten des Koordinatenmessgeräts, des Drehraten- und des Beschleunigungssensors werden Gelenkposition, -geschwindigkeit und -beschleunigung konsistent bestimmt. Dieses Vorgehen ist auch aus der Inertialnavigation [Sti00] bekannt, wo ein Koordinatenmessgerät die Messungen der inertialen Sensoren stützt. Zusätzlich wird die Motorposition über den Resolver gemessen. Durch numerisches Differenzieren steht damit auch die Motorgeschwindigkeit und somit der komplette Zustand der Achse zur Verfügung. Schließlich wird noch das anhand des Motorstroms berechnete Motormoment aufgezeichnet.

Eine Simulation des durch das Motormoment angetriebenen Antriebsstrangs hat sich als ungeeignet für die Identifikation herausgestellt, da das zugrunde liegende System instabil ist. Damit ist das System sehr empfindlich auf Fehler im Reibmodell und im Gravitationsmoment. Zur Lösung dieses Problems wird als Eingang nicht das Motormoment benutzt, sondern die gemessene Motorposition und -geschwindigkeit. Auf diese Weise zerfällt das Modell in die stabile Dynamik des durch die Motorposition getriebenen Arms und algebraische Gleichungen zur Berechnung des Motormoments.

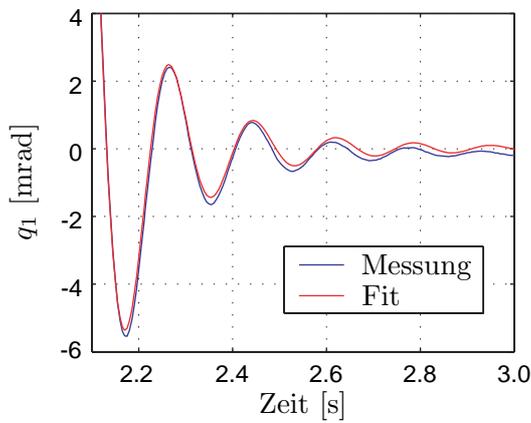
Entsprechend Kapitel 2.2 ist die Dynamik des Antriebsstranges von der Motorposition zum Arm durch die Gleichungen (2.50c) bis (2.50h) gegeben, wobei Gleichung (2.50c) durch Vorgabe von θ und $\dot{\theta}$ zu einer Differentialgleichung erster Ordnung in φ wird. Für die notwendigen Simulationen wurde das Modelica-Modell in Abbildung 3.9 verwendet. Zur Identifikation der gesuchten Modellparameter wurde eine Nonlinear-Least-Squares Optimierung mit Parameterbeschränkungen eingesetzt, welche in der Optimierungsumgebung MOPS [JBL⁺02] zur

Verfügung steht. Die Bewertung der Unterschiede zwischen Messung und Simulation erfolgt bei diesem Optimierungsverfahren anhand der Summen der Fehlerquadrate von Gelenkposition und -geschwindigkeit. Die Anfangsbedingungen für die Bewegung der Last werden dabei mit Hilfe der Messungen so gesetzt, dass die Fehler zu Beginn der Simulation verschwinden.

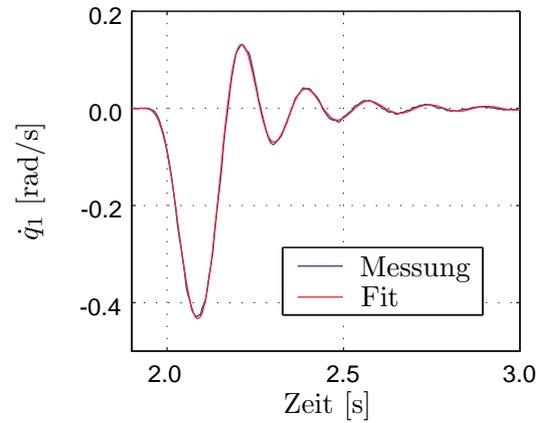
In Abbildung 3.10 sind Vergleiche des optimierten Modells mit Messungen der Gelenkposition und -geschwindigkeit für verschiedene Bewegungen dargestellt. Dabei ist beim Vergleich der Positionen nur der Einschwingvorgang am Ende der Bewegung abgebildet, da die Unterschiede im Vergleich zur Gesamtbewegung sonst nicht sichtbar sind. Beim Vergleich der Geschwindigkeiten ist dagegen die komplette Bewegung dargestellt. In allen Fällen ist eine sehr gute Übereinstimmung von Messung und Modell zu erkennen. Für die Optimierung der Amplituden und Phasenlagen des Übersetzungsfehlers wurde zusätzlich eine Bahn verwendet, bei welcher die Achse längere Zeit mit konstanter Geschwindigkeit bewegt wird. Für diese Bahn kann, wie schon in Abbildung 3.8 dargestellt, die Drehungleichförmigkeit besser beurteilt werden. Das Ergebnis der Optimierung zeigt Abbildung 3.11. Man erkennt, dass die Form der Kurve qualitativ richtig wiedergegeben wird. Trotzdem sind noch deutliche Unterschiede zwischen Messung und Simulation festzustellen. Aufgrund der bereits diskutierten Probleme mit der Vermessung der Drehungleichförmigkeit, ist ein theoretisch möglicher genaueres Abgleich durch ein Modell mit zusätzlichen Frequenzen jedoch nicht sinnvoll.

Nach der Bestimmung des Modells für die Bewegung des Armes bei vorgegebener Motorposition, können nun die Modelle für die antriebsseitige Reibung erstellt werden. Dazu kann entweder nochmals eine Optimierung durchgeführt werden oder es werden die Ergebnisse der direkten Messungen benutzt. Im letzteren Fall ist die gewichtete Summe der Reibungen von Antrieb und Gelenk τ_{Rvk1} und das Produkt der Wirkungsgrade von Antrieb und Gelenk aus den direkten Messungen bekannt. Die Gelenkreibung und der Gelenkwirkungsgrad können aus dem optimierten Modell entnommen werden. Damit lassen sich die Antriebsverluste direkt berechnen. Einen Vergleich der gemessenen Motormomente mit denen des so gewonnenen Modells zeigt Abbildung 3.12. Auch bei den Motormomenten ist eine sehr hohe Übereinstimmung erkennbar. Im Stillstand zeigt die Simulation ein stärkeres „Rauschen“ als die Messung. Dieses wird durch die Mehrdeutigkeit des Moments aufgrund der modellierten Haftreibung verursacht. In der Messung ist dieser Effekt nicht zu sehen, da hier das Moment nicht aus der Bewegung bestimmt wird, sondern durch den Motorstrom vorgegeben ist. Dieses Vorgehen zur Bestimmung der antriebsseitigen Reibung kann auch für die anderen Achsen angewendet werden.

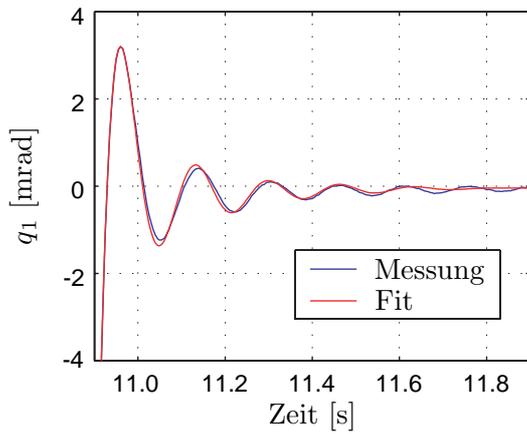
Bei der Identifikation der Antriebsstränge wurde für die ersten drei Achsen das vollständige Antriebsstrangmodell verwendet. Für die Achsen vier bis sechs, welche zur „Hand“ gehören, erwies sich ein reduziertes Modell des Antriebsstrangs als ausreichend. Dieses ist ebenfalls im Anhang B.5 dargestellt.



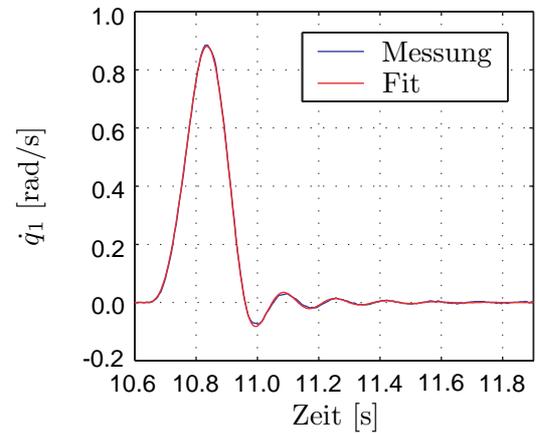
(a) Position bei kurzer Bewegung



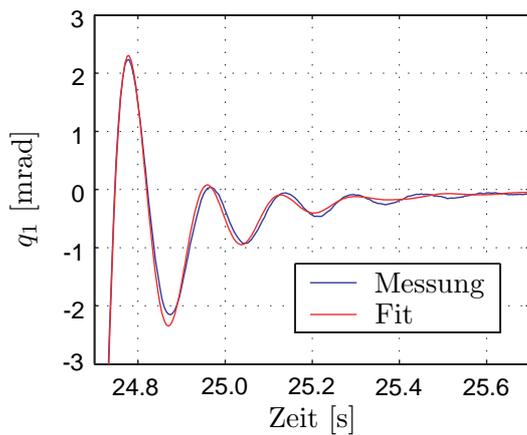
(b) Geschwindigkeit bei kurzer Bewegung



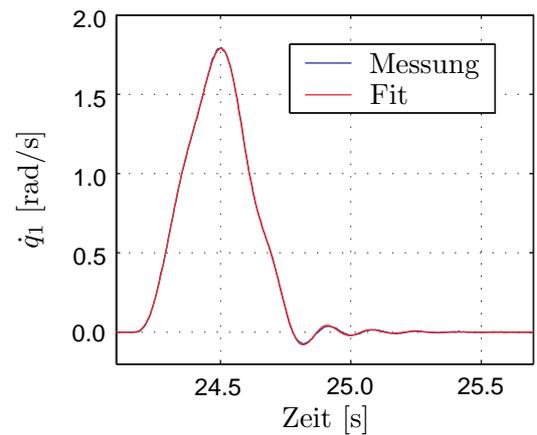
(c) Position bei mittlerer Bewegung



(d) Geschwindigkeit bei mittlerer Bewegung



(e) Position bei langer Bewegung



(f) Geschwindigkeit bei langer Bewegung

Abbildung 3.10: Vergleich der gemessenen Gelenkbewegungen mit dem Modell

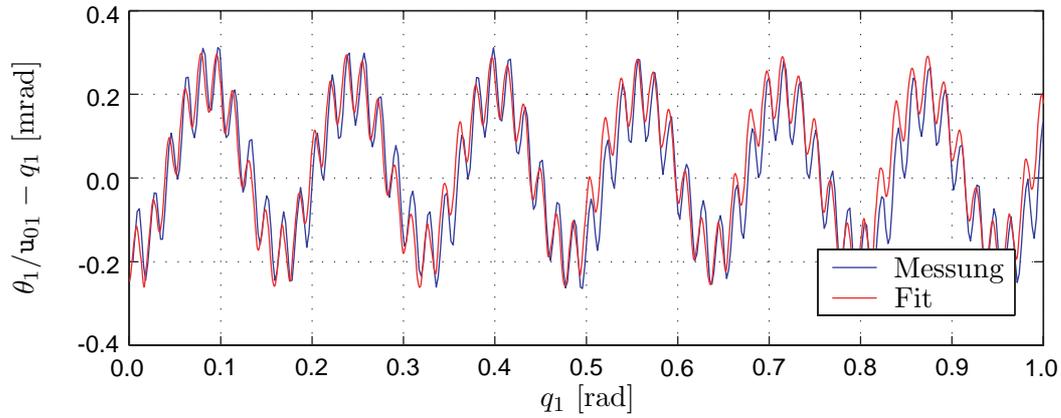


Abbildung 3.11: Vergleich der gemessenen Drehungleichförmigkeiten mit dem Modell

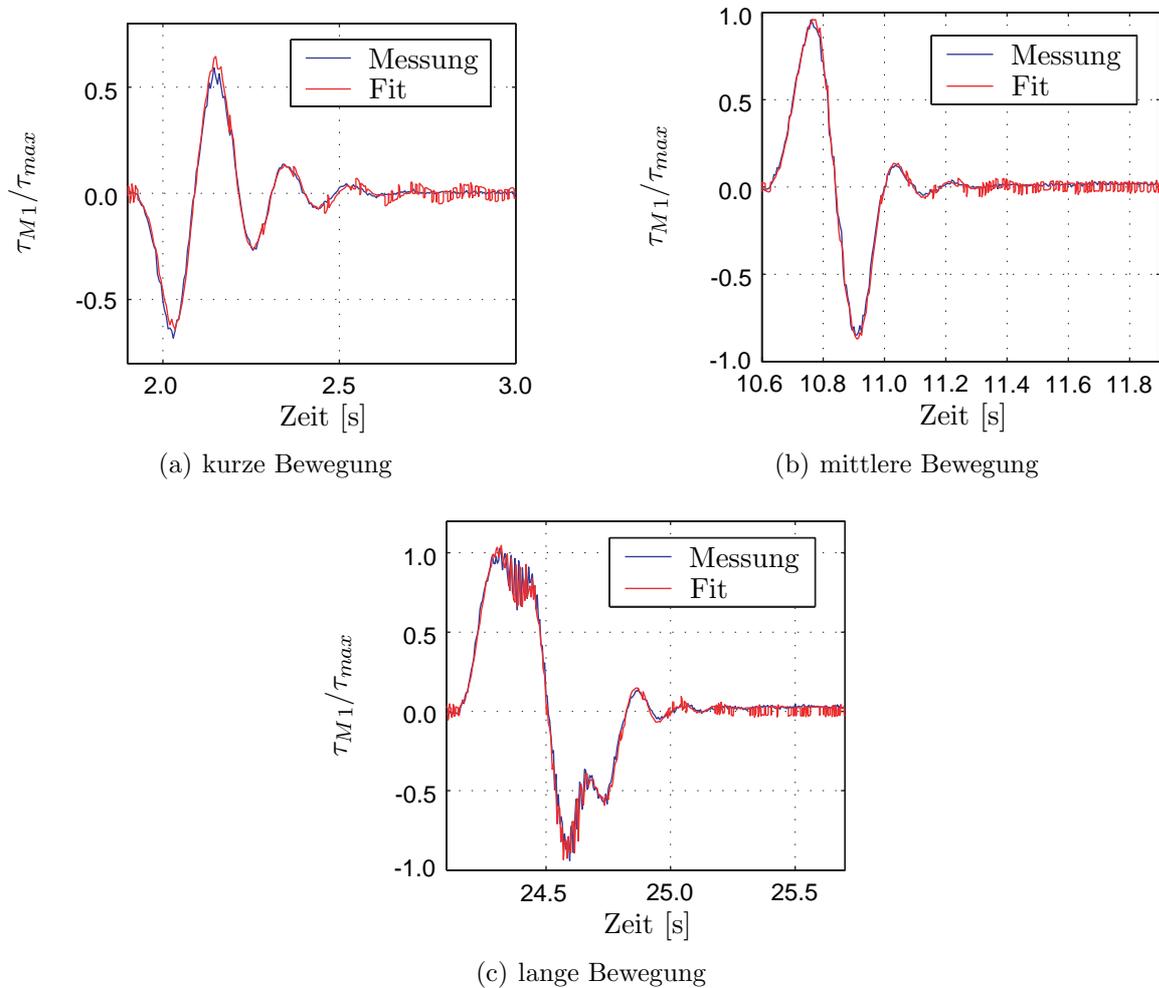


Abbildung 3.12: Vergleich der Motormomente aus Messung und Modell bei verschiedenen Positionierbewegungen

Kapitel 4

Invertierung von Systemen

Entsprechend den vorausgegangenen Überlegungen wird für die Steuerung ein System gesucht, welches bei gegebenen Solltrajektorien für die Gelenkwinkel den Verlauf der Zustandsgrößen und die Motormomente so bestimmt, dass der Roboter mit diesen Motormomenten die Solltrajektorien der Gelenkwinkel exakt reproduziert.

Im üblichen Sprachgebrauch wird ein System, welches bei Vorgabe der gewünschten Ausgangsgrößen und der Anfangszustände eines Systems die notwendigen Eingangsgrößen für das System bestimmt, als *Rechtsinverse* bezeichnet. Entsprechend wird ein System, welches aus den Ausgangsgrößen des Systems dessen Eingang bestimmt, als *Linksinverse* bezeichnet. In dieser Arbeit werden Systeme behandelt, bei denen die Anzahl der Eingänge gleich der Anzahl der Ausgänge ist. Für solche Systeme sind Linksinverse und Rechtsinverse identisch und werden im Folgenden als *Inverse des Systems* oder *inverses System* bezeichnet.

Die Bestimmung und Berechnung von inversen Systemen ist aufgrund der großen Bedeutung inverser Systeme schon lange Gegenstand der Forschung. Dabei werden die zu invertierenden Systeme in verschiedene Klassen eingeteilt. Entsprechend dem in Kapitel 2 hergeleiteten Modell ist der Roboter in dieser Arbeit in Form eines multivariablen, nichtlinearen Systems gegeben, dessen Eingänge linear eingehen. Die Invertierbarkeit solcher Systeme wurde erstmals in [Hir79] untersucht.

In diesem Kapitel werden die zur Invertierung von Systemen relevanten Systemeigenschaften vorgestellt und Möglichkeiten zur Berechnung inverser Systeme untersucht. Nach der Wiederholung grundlegender Definitionen werden Bedingungen für die Invertierbarkeit angegeben und die Dynamik inverser Systeme, sowie Besonderheiten bei der Invertierung flacher Systeme untersucht. Danach werden Möglichkeiten gezeigt, um inverse Systeme aufzustellen und zu berechnen. Insbesondere wird eine neue Methode vorgestellt, welche auf der Reduktion des differentiellen Index von DAE-Systemen basiert. Die zur Indexreduktion eingesetzten Algorithmen werden beschrieben und an einem Beispielsystem angewendet. Abschließend wird gezeigt wie, basierend auf dieser Methode, mit Modelica inverse Systeme modelliert werden können.

4.1 Grundlagen für die Invertierung

Zunächst werden einige Definitionen angegeben, welche bei der Invertierung von Systemen benutzt werden. Diese sind in verschiedenen Büchern zur nichtlinearen Regelung in jeweils etwas anderer Notation enthalten. Die hier verwendete Darstellung orientiert sich an [SJK97].

4.1.1 Relativer Grad

Von grundlegender Bedeutung für die Betrachtung inverser Systeme ist der relative Grad, welcher zunächst für Systeme der Form

$$\dot{x} = f(x) + g(x)u \quad (4.1a)$$

$$y = h(x) + j(x)u \quad x \in \mathbb{R}^n; u, y \in \mathbb{R} \quad (4.1b)$$

angegeben werden soll. Diese Systeme besitzen einen Eingang und einen Ausgang, weshalb sie auch als SISO (Single Input Single Output) Systeme bezeichnet werden.

Definition 4.1 (Relativer Grad eines SISO-Systems).

Der relative Grad eines nichtlinearen SISO-Systems (4.1) an der Stelle $x = x_0$ ist diejenige ganze Zahl r für die gilt

$$(i) \quad L_g L_f^k h(x) \equiv 0 \quad \text{für } k = 0, \dots, r-2 \text{ und } x \text{ in einer Umgebung von } x_0$$

$$(ii) \quad L_g L_f^{r-1} h(x_0) \neq 0$$

Anschaulich gesprochen geben die Bedingungen an, dass in der Gleichung der r -ten Ableitung des Ausgangs zum ersten Mal der Eingang auftritt, was mit der folgenden Betrachtung verdeutlicht werden soll. Für $j(x_0) \neq 0$ gilt $r = 0$, da in diesem Fall u direkt in der Gleichung des Ausgangs auftritt. Ist $j(x_0) = 0$, dann führt die Differentiation der Ausgangsgleichung (4.1b) auf

$$\dot{y} = \frac{\partial h(x)}{\partial x} \dot{x} = L_f h(x) + L_g h(x)u.$$

Wenn nun $L_g h(x_0) = L_g L_f^0 h(x_0) \neq 0$ ist, dann tritt in der ersten Ableitung von y der Eingang u auf und entsprechend ist der relative Grad 1. Andernfalls ergeben sich durch fortgesetztes Differenzieren von y die bereits in (ii) angegebenen Terme $L_g L_f^{r-1} h(x_0)$ als Multiplikatoren von u .

Im Falle eines linearen Systems entspricht der relative Grad bekanntermaßen der Differenz zwischen Nennergrad und Zählergrad bzw. zwischen der Anzahl der Pole und der Anzahl der Nullstellen der Übertragungsfunktion.

$$H(s) = k \frac{q_0 + q_1 s + \dots + s^{n-r}}{p_0 + p_1 s + \dots + s^n}.$$

Um diese Bedingung für lineare Systeme in Zustandsform anwenden zu können und um eine Verbindung zu Definition 4.1 herzustellen, wird das lineare SISO-System

$$\dot{x} = Ax + bu \quad (4.2a)$$

$$y = cx \quad x \in \mathbb{R}^n; u, y \in \mathbb{R} \quad (4.2b)$$

als Übertragungsfunktion

$$\begin{aligned} H(s) &= c(sI - A)^{-1}b \\ &= cb\frac{1}{s} + cAb\frac{1}{s^2} + \dots + cA^{r-1}b\frac{1}{s^r} + \dots \end{aligned} \quad (4.3)$$

dargestellt. Die Bedingungen zur Bestimmung von r aus Definition 4.1 vereinfachen sich für Systeme entsprechend den Gleichungen (4.2) zu

$$(i) \quad cA^k b = 0 \quad \text{für } k = 0, \dots, r-2 \quad (4.4a)$$

$$(ii) \quad cA^{r-1}b \neq 0 \quad (4.4b)$$

Dies erlaubt mit Gleichung (4.3) wiederum die anschauliche Interpretation des relativen Grades als diejenige Ableitung des Ausgangs, in welcher der Eingang das erste Mal auftritt. Wird das System nach Gleichung (4.2) mit $y = cx + du$ um einen direkten Durchgriff zwischen Ein- und Ausgang erweitert ist dementsprechend der relative Grad $r = 0$. Bei einem linearen System entsprechend den Gleichungen (4.2) ist der relative Grad darüber hinaus gemäß den Gleichungen (4.4) immer für alle $x \in \mathbb{R}^n$ definiert.

Im Gegensatz dazu ist der relative Grad bei einem nichtlinearen System zunächst nur lokal definiert. Wenn die Bedingungen aus Definition 4.1 global gültig sind, bezeichnet man r entsprechend als *globalen relativen Grad*.

Auf ähnliche Weise kann der relative Grad auch für Systeme mit mehreren Eingängen und Ausgängen definiert werden. Diese werden als MIMO (Multiple Input Multiple Output) Systeme bezeichnet. Dabei wird nur der Fall betrachtet, in dem die Anzahl der Ausgänge gleich der Anzahl der Eingänge ist, also Systeme der Form

$$\dot{x} = f(x) + g(x)u \quad (4.5a)$$

$$y = h(x) + j(x)u, \quad x \in \mathbb{R}^n; u, y \in \mathbb{R}^m. \quad (4.5b)$$

Erweiterungen für den Fall, in dem die Zahl der Eingänge größer als die Zahl der Ausgänge ist, sind möglich, werden in dieser Arbeit aber nicht benötigt.

Definition 4.2 (Relativer Grad eines MIMO-Systems).

Der vektorielle relative Grad eines nichtlinearen MIMO-Systems (4.5) an der Stelle $x = x_0$ ist diejenige Menge $\{r_1, \dots, r_m\}$ für die gilt

$$(i) \quad L_{g_j} L_f^k h_i(x) \equiv 0 \quad \text{für alle } 1 \leq i, j \leq m, k < r_i - 1 \text{ und } x \text{ in einer Umgebung von } x_0$$

$$(ii) \quad R(x) = \left[\frac{\partial y_i^{(r_i)}}{\partial u_j} \right]_{1 \leq i, j \leq m} = \begin{pmatrix} L_{g_1} L_f^{r_1-1} h_1(x) & \dots & L_{g_m} L_f^{r_1-1} h_1(x) \\ \vdots & & \vdots \\ L_{g_1} L_f^{r_m-1} h_m(x) & \dots & L_{g_m} L_f^{r_m-1} h_m(x) \end{pmatrix}$$

nicht singulär bei $x = x_0$.

Äquivalent zum SISO Fall gibt der vektorielle relative Grad eines MIMO-Systems an, wie oft die Ausgänge differenziert werden müssen, um aus den differenzierten Ausgängen die Eingänge berechnen zu können.

Des Weiteren wird bei MIMO-Systemen häufig noch eine Größe

$$r = r_1 + r_2 + \dots + r_m$$

angegeben, welche, ähnlich wie bei SISO-Systemen, als *gesamter relativer Grad* des MIMO-Systems angesehen werden kann.

4.1.2 Transformation in Normalform

Wenn der relative Grad eines Systems bei $x = x_0$ definiert ist, dann existiert zumindest lokal eine Transformation $T(x)$, welche das System in Normalform transformiert [Isi95]. Für das Beispiel eines SISO Systems gelte

$$(\xi, z) = T(x) \quad \xi \in \mathbb{R}^r; z \in \mathbb{R}^{n-r}$$

Die ersten r Komponenten von $T(x)$ werden dabei gewählt als

$$\begin{aligned} \xi_1 &= T_1(x) = y = h(x) \\ \xi_2 &= T_2(x) = \dot{y} = L_f h(x) \\ &\vdots \\ \xi_r &= T_r(x) = y^{(r-1)} = L_f^{r-1} h(x), \end{aligned}$$

womit die lineare Unabhängigkeit der Koordinaten ξ sichergestellt ist. Dies wird auch durch die Darstellung von ξ in Zustandsform deutlich:

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 \\ &\vdots \\ \dot{\xi}_{r-1} &= \xi_r \\ \dot{\xi}_r &= L_f^r h(x) + L_g L_f^{r-1} h(x) u \end{aligned}$$

Es ergibt sich eine Kette von Integratoren mit dem Eingang $L_f^r h(x) + L_g L_f^{r-1} h(x) u$. Im Fall eines MIMO Systems wird prinzipiell gleich verfahren, nur dass jeweils r_i Komponenten von $T(x)$ dem Ausgang y_i und seinen ersten $r_i - 1$ Ableitungen zugeordnet werden. Damit ergibt sich einfach:

$$\begin{aligned} \xi_1^i &= T_1^i(x) = y_i = h_i(x) \\ \xi_2^i &= T_2^i(x) = \dot{y}_i = L_f h_i(x) \\ &\vdots \\ \xi_r^i &= T_r^i(x) = y_i^{r_i-1} = L_f^{r_i-1} h_i(x) \end{aligned}$$

Im Allgemeinen ist die Transformation $T(x)$ mit der Angabe für die Elemente ξ jedoch noch nicht vollständig, sondern muss noch um $n - r$ Funktionen T_{r+1}, \dots, T_n erweitert werden. Damit die Transformation $(\xi, z) = T(x) \in \mathbb{R}^n$ ein Diffeomorphismus ist, muss die Matrix

$$\left(\frac{\partial T}{\partial x} \right) (x_0)$$

nicht singulär sein. Wie in [Isi95] bewiesen, ist es bei definiertem relativem Grad, durch geeignete Wahl der Koordinaten z immer möglich, diese Bedingung zu erfüllen. Unter Verwendung der Notation

$$\xi = (\xi^1, \dots, \xi^m), \quad \xi^i = (\xi_1^i, \dots, \xi_{r_i}^i)^T \quad (4.6)$$

$$z = (T_{r+1}(x), \dots, T_n(x))^T \quad (4.7)$$

$$a_i(\xi, z) = L_f^{r_i} h_i(T^{-1}(\xi, z)) \quad \text{für } 1 \leq i \leq m \quad (4.8)$$

kann das MIMO System entsprechend den Gleichungen (4.5) in der folgenden Normalform, welche auch als *Byrnes-Isidori-Normalform* bezeichnet wird, dargestellt werden:

$$\dot{z} = q(\xi, z) + \gamma(\xi, z)u \quad (4.9a)$$

$$\dot{\xi}_1^i = \xi_2^i \quad (4.9b)$$

$$\vdots \quad (4.9c)$$

$$\dot{\xi}_{r_i-1}^i = \xi_{r_i}^i \quad (4.9d)$$

$$\dot{\xi}_{r_i}^i = a_i(\xi, z) + (R(T^{-1}(\xi, z)))_i u_j \quad (4.9e)$$

$$y_i = \xi_1^i \quad 1 \leq i \leq m \quad (4.9f)$$

4.1.3 Feedback-Linearisierung

Der relative Grad und die Transformation auf Normalform werden häufig zur Feedback-Linearisierung von Systemen benutzt. Da die Linearisierung eines Systems durch Feedback eng mit dessen Invertierung zusammenhängt, wird im Folgenden noch eine kurze Übersicht zur Feedback-Linearisierung gegeben. Eine ausführliche Beschreibung kann z. B. [Isi95] entnommen werden.

Betrachtet werden für die Feedback-Linearisierung MIMO-Systeme der Form

$$\dot{x} = f(x) + g(x)u \quad (4.10a)$$

$$y = h(x), \quad x \in \mathbb{R}^n; u, y \in \mathbb{R}^m, \quad (4.10b)$$

welche durch geeignetes Feedback in lineare, entkoppelte Systeme transformiert werden sollen. Ist der vektorielle relative Grad des Systems (4.10) definiert, dann kann die gewünschte Linearisierung mit Hilfe von *statischem Feedback* der Form

$$u = \alpha(x) + \beta(x)v$$

mit dem neuen Eingang v erreicht werden. Dies folgt direkt aus der Darstellung des Systems in Normalform. Wenn der gesamte relative Grad r gleich der Systemordnung ist, dann ist die Transformation in die linearen Koordinaten durch die Elemente ξ bereits komplett beschrieben und das System wird durch das statische Feedback zustandslinearisiert. Ist $r < n$, dann kann das System lediglich bezüglich seines Ein-Ausgangsverhaltens linearisiert werden.

Wenn der vektorielle relative Grad nicht definiert ist, dann ist in vielen Fällen eine dynamische Erweiterung des Systems um Integratoren an einzelnen Eingängen so möglich, dass der vektorielle relative Grad des erweiterten Systems definiert ist. Nun erhält man für das erweiterte System wiederum eine Ein-Ausgangs- bzw. Zustandslinearisierung welche aber, durch die dynamische Erweiterung, einen dynamischen Kompensator beinhaltet.

$$u = \alpha(x, \zeta) + \beta(x, \zeta)v$$

$$\dot{\zeta} = \gamma(x, \zeta) + \delta(x, \zeta)v$$

In diesem Fall spricht man deshalb von einer Linearisierung durch *dynamisches Feedback*. Interessant ist in diesem Zusammenhang noch, dass dynamisches Feedback nur für MIMO-Systeme relevant ist. Kann dagegen ein SISO-System nicht mit statischem Feedback linearisiert werden, so wird dies auch durch Verwendung von dynamischem Feedback nicht gelingen.

Für Roboter mit elastischen Gelenken wurden diese Methoden bereits ausgiebig untersucht, wobei als Antriebsstrang jeweils lineare Federn angenommen wurden. Unter Vernachlässigung der Bewegung der Motoren im Raum, womit die Mehrkörperdynamik Gleichung (2.33) entspricht, wurde in [Spo87] gezeigt, dass sich der Roboter mit Hilfe von statischem Feedback linearisieren lässt. Für das vollständige Modell ist der vektorielle relative Grad jedoch nicht mehr definiert. In [DL95] konnte gezeigt werden, dass alle Roboter basierend auf dem vollständigen Mehrkörpermodell mit $S \neq 0_{N \times N}$ durch dynamisches Feedback linearisiert werden können, wobei der notwendige dynamische Kompensator vom jeweiligen Roboter abhängt. Dieses Resultat wird auch in [DL98] erreicht, allerdings erfolgt hier die Analyse nicht anhand der üblichen Zustandsdarstellung sondern direkt an den Bewegungsgleichungen, was zu einem eleganten und deutlich vereinfachten Entwurf führt.

4.2 Invertierbarkeit von Systemen

Die Berechnung der Eingänge aus den Ausgangsgrößen ist nicht für beliebige Systeme möglich. Ein Beispiel ist das lineare, steuer- und beobachtbare System aus [BM67]:

$$\dot{x} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ -1 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{pmatrix} u \quad (4.11a)$$

$$y = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} x \quad (4.11b)$$

Deshalb wird zunächst der Frage nachgegangen, wann ein System überhaupt invertierbar ist. Dabei wird die Invertierbarkeit eines Systems entsprechend Hirschorn [Hir79] definiert.

Definition 4.3 (Invertierbarkeit).

Ein nichtlineares System entsprechend den Gleichungen (4.5) mit $j(x) = 0$ wird als invertierbar an der Stelle $x = x_0 \in \mathbb{R}^n$ bezeichnet, wenn für alle Eingangstrajektorien u und \hat{u} , die unterscheidbar und zulässig sind, für den Ausgang $y(t, u, x_0)$ gilt

$$y(\cdot, u, x_0) \neq y(\cdot, \hat{u}, x_0).$$

Wenn unterscheidbare Eingänge verschiedene Ausgänge erzeugen, dann sollte es theoretisch möglich sein, aus den Ausgangsgrößen die zugehörigen Eingangsgrößen zu bestimmen. Beim Beispielsystem 4.11 ist die Invertierung schon deshalb nicht möglich, weil die Ausgangsgrößen des Systems die Gleichung $\dot{y}_2 = -y_1$ erfüllen müssen und somit auch durch beliebigen Eingangsgrößen nicht unabhängig voneinander vorgegeben werden können. Im Allgemeinen wird sich die Invertierbarkeit jedoch nicht so einfach zeigen lassen wie im gewählten Beispiel. Aus diesem Grund sind Algorithmen entstanden, welche anhand der Systemgleichungen die Invertierbarkeit eines Systems untersuchen [Hir79]. Für aufwändige Systeme kann die Invertierbarkeit aber auch elegant mit Hilfe des relativen Grades nachgewiesen werden.

Unter der Annahme, dass der relative Grad definiert ist, kann das System, wie gerade in Abschnitt 4.1.2 gezeigt, in die Byrnes-Isidori Normalform (4.9) transformiert werden. Unter Ausnutzung dieser Normalform ist es nun auch möglich, das inverse System anzugeben, bei

dem $y(t)$ dazu gezwungen werden soll, einem vorgegebenen $y_d(t)$ exakt zu folgen. Die Vorgabe $y(t) = y_d(t)$ beschränkt den Zustand ξ auf

$$\xi(t) \equiv \xi_d(t) = (\xi_d^1(t), \dots, \xi_d^m(t))^T$$

mit

$$\xi_d^i(t) = (y_{id}(t), \dots, y_{id}^{(r_i-1)}(t)).$$

Diese Beschränkungen können nur erfüllt werden wenn:

- (i) Für den Anfangszustand des Systems gilt

$$\xi(0) = \xi_d(0) \quad (4.12)$$

- (ii) Der Eingang $u(t)$ so gewählt ist, dass gilt

$$u(t) = u_d(t) = R^{-1} (T^{-1}(\xi_d(t), z(t))) \left(-a(\xi_d(t), z(t)) + \begin{pmatrix} y_{1d}^{(r_1)}(t) \\ \vdots \\ y_{md}^{(r_m)}(t) \end{pmatrix} \right) \quad (4.13)$$

wobei $z(t)$ die Lösung der Differentialgleichung

$$\dot{z} = q(\xi_d(t), z) + \gamma(\xi_d(t), z) R^{-1} (T^{-1}(\xi_d(t), z(t))) \left(-a(\xi_d(t), z(t)) + \begin{pmatrix} y_{1d}^{(r_1)}(t) \\ \vdots \\ y_{md}^{(r_m)}(t) \end{pmatrix} \right) \quad (4.14)$$

mit der Anfangsbedingung $z(0) = z_0$

ist. Bei definiertem relativem Grad existiert die Abbildung T und damit auch die Funktionen $a(\xi_d(t), z(t))$, $q(\xi_d(t), z)$ und $\gamma(\xi_d(t), z)$. Da T ein Diffeomorphismus ist, ist auch T^{-1} definiert. Entsprechend der Definition des relativen Grades ist darüber hinaus R nicht singulär und damit R^{-1} ebenfalls definiert. Damit stellt die Existenz des relativen Grades die Existenz des inversen Systems sicher. Entsprechend den Gleichungen (4.13) und (4.14) sind für alle Startwerte $\xi(0) = \xi_d(0)$ und $z(0) = z_0$ die Eingänge $u(t)$, welche die Ausgänge immer auf $y(t) = y_d(t)$ halten, eindeutig definiert. Bemerkenswert ist noch, dass z_0 dabei beliebig gewählt werden kann.

4.3 Dynamik von inversen Systemen

Von besonderem Interesse für den Einsatz eines inversen Systems ist dessen dynamisches Verhalten. Daher wird dieses im Folgenden für Systeme, deren relativer Grad definiert ist, näher betrachtet und die zugehörigen Begriffe der internen Dynamik und Nulldynamik eingeführt.

4.3.1 Interne Dynamik

Betrachtet man $y_R(t)$ als Eingang, $u(t)$ als Ausgang und $z(t)$ als Zustand eines Systems, dann erhält man ein neues, zum Originalsystem inverses System, welches auch als Inverse des Originalsystems bezeichnet wird.

$$\dot{z} = q(\xi_R(t), z) + \gamma(\xi_R(t), z) R^{-1} (T^{-1}(\xi_R(t), z(t))) \left(-a(\xi_R(t), z(t)) + \begin{pmatrix} y_{1R}^{(r_1)}(t) \\ \vdots \\ y_{mR}^{(r_m)}(t) \end{pmatrix} \right) \quad (4.15a)$$

$$u(t) = R^{-1} (T^{-1}(\xi_R(t), z(t))) \left(-a(\xi_R(t), z(t)) + \begin{pmatrix} y_{1R}^{(r_1)}(t) \\ \vdots \\ y_{mR}^{(r_m)}(t) \end{pmatrix} \right) \quad (4.15b)$$

Die Dynamik der Inversen, welche durch Gleichung (4.15a) gegeben ist, wird als *interne Dynamik* des Systems bezeichnet, da sie im Ein- Ausgangsverhalten des Systems nicht sichtbar wird. Entsprechend den Angaben zur Transformation T ist die Dimension der internen Dynamik $n - r$. Wie aus Gleichung (4.15a) ersichtlich, ist die interne Dynamik bei $\gamma \neq 0_{m \times 1}$ nicht autonom. Für SISO Systeme ist es immer möglich die Koordinaten der internen Dynamik z so zu wählen, dass $\gamma = 0_{m \times 1}$ [Isi95]. Bei MIMO Systemen ist dies dagegen nur in Sonderfällen möglich.

4.3.2 Nulldynamik

Häufig wird die sogenannte Nulldynamik eines Systems untersucht. Dabei handelt es sich um die interne Dynamik des Systems, wobei die Startwerte und Eingänge so gewählt werden, dass die Ausgänge konstant auf Null bleiben. In [IM86] wurde gezeigt, dass die Nulldynamik dann mit der internen Dynamik übereinstimmt, wenn das Inversionsproblem nicht singular bei $y = 0$ ist. Dies entspricht der Forderung, dass der relative Grad in der entsprechenden Umgebung definiert sein muss. Es handelt sich also um einen Spezialfall der internen Dynamik mit

$$\begin{aligned} y_{1R}(t) &\equiv 0 \\ &\vdots \\ y_{mR}(t) &\equiv 0 \end{aligned}$$

Der Eingang, welcher diese Bedingungen sicherstellt, ist gegeben durch

$$u(t) = -R^{-1} (T^{-1}(0, z(t))) a(0, z(t)),$$

was auf eine interne Dynamik von

$$\dot{z} = q(0, z) - \gamma(0, z) R^{-1} (T^{-1}(0, z(t))) a(0, z(t)) =: f_{ND}(z)$$

führt, welche nun als Nulldynamik bezeichnet wird. Ein wesentlicher Vorteil beim Arbeiten mit der Nulldynamik ist, dass es sich um ein autonomes System handelt, da die Anregung aufgrund $y_R(t) \equiv 0$ verschwunden ist. Damit vereinfachen sich z. B. Stabilitätsuntersuchungen wesentlich.

4.3.3 Stabilität der internen Dynamik

Wenn die Inverse eines Systems als Steuerung verwendbar sein soll, dann ist die Stabilität der Inversen und damit die Stabilität der internen Dynamik des Systems eine unverzichtbare Voraussetzung.

Die Bedingungen für die Stabilität der Inversen von linearen Systemen können auf anschauliche Weise mit dem linearen SISO System

$$H(s) = k \frac{q_0 + q_1 s + \dots + s^{n-r}}{p_0 + p_1 s + \dots + s^n}$$

gezeigt werden. Dessen Inverse kann sofort als

$$H^{-1}(s) = \frac{1}{k} \frac{p_0 + p_1 s + \dots + s^n}{q_0 + q_1 s + \dots + s^{n-r}}$$

angegeben werden und ist offenbar dann stabil, wenn die Nullstellen von $H(s)$ in der linken Halbebene der komplexen Ebene liegen. Lineare Systeme, welche diese Bedingung erfüllen, werden als *minimalphasig* bezeichnet. Bei nichtlinearen Systemen ist ein solches Vorgehen nicht mehr möglich. Statt dessen muss die Stabilität der nichtlinearen internen Dynamik gezeigt werden, wofür die üblichen Methoden für nichtlineare Systeme verwendet werden können. Ein direkter Stabilitätsnachweis scheidet jedoch häufig daran, dass die interne Dynamik nicht autonom ist. Daher wird gewöhnlich die Stabilität der Nulldynamik untersucht, wobei, in Anlehnung an den linearen Fall, die folgende Definition der Minimalphasigkeit verwendet wird.

Definition 4.4 (Minimalphasigkeit).

Ein System entsprechend den Gleichungen (4.5) wird als minimalphasig bezeichnet, wenn seine Nulldynamik stabil ist.

In der gleichen Art und Weise sind *asymptotische Minimalphasigkeit* und *exponentielle Minimalphasigkeit* definiert. Ferner wird das System als *global minimalphasig* bezeichnet, wenn seine Nulldynamik für alle z_0 stabil ist.

Die Stabilität der Nulldynamik garantiert jedoch, aufgrund ihrer Definition, nur die lokale Stabilität der internen Dynamik nahe $y = 0$.

Bei Verwendung des inversen Systems als Steuerung ist in der Praxis jedoch auch eine stabile, aber schlecht gedämpfte interne Dynamik problematisch. So werden am Ende einer Bewegung, wenn sich $y_R(t)$ bereits wieder in Ruhe befindet, trotzdem noch Oszillationen auf $u_R(t)$ vorhanden sein. Ist die Steuerung nun aufgrund von Modellierungsfehlern nicht exakt die Inverse des Systems, dann können auf diese Weise zusätzliche Schwingungen am Ausgang des Systems erzeugt werden.

4.4 Invertierung von flachen Systemen

Zur Berechnung der Steuergrößen von nichtlinearen Systemen werden häufig Ansätze verfolgt, welche auf der sogenannten *Flachheit* des Systems beruhen. Die die Flachheit eines

Systems eng mit dessen Invertierbarkeit und der Linearisierbarkeit durch Feedback verknüpft ist werden im Folgenden, soweit notwendig, einige Zusammenhänge mit der bisher gezeigten Inversion von Systemen gezeigt. Eingeführt wurde der Begriff der Flachheit in [FLMR95]. Einen anschaulichen Überblick gibt z. B. [RRZ97]. Bei der Betrachtung von flachen Systemen wird im allgemeinsten Fall von der Form

$$\dot{x} = f(x, u) \quad x \in \mathbb{R}^n, u \in \mathbb{R}^m \quad (4.16)$$

ausgegangen. In der Literatur wird Flachheit auf unterschiedliche Arten definiert. Die hier gegebene Definition stammt aus [RRZ97].

Definition 4.5 (Flachheit).

Ein nichtlineares System entsprechend Gleichung (4.16) wird als flach bezeichnet, wenn ein flacher Ausgang

$$y_F = \phi(x, u, \dot{u}, \dots, u^{(\alpha)})$$

mit $\dim y_F = \dim u$ existiert, über den sich die Zustände und Eingänge des Systems eindeutig als Funktionen des flachen Ausgangs darstellen lassen.

$$x = \psi_1(y_F, \dot{y}_F, \dots, y_F^{(\beta)}) \quad (4.17)$$

$$u = \psi_2(y_F, \dot{y}_F, \dots, y_F^{(\beta+1)}) \quad (4.18)$$

Im Hinblick auf die Berechnung inverser Systeme lässt sich feststellen, dass sich ein flaches System mit Hilfe von ψ_1 und ψ_2 von seinen flachen Ausgängen aus invertieren lässt. Die Besonderheit dabei ist, dass das inverse System nur aus algebraischen Gleichungen besteht.

Für Systeme entsprechend den Gleichungen (4.5), wie sie bei der Inversion betrachtet wurden, stellt Flachheit damit den Sonderfall dar, in dem der relative Grad r gleich der Systemordnung ist. Damit verschwinden die Koordinaten der internen Dynamik z und es verbleibt, mit der Notation aus Kapitel 4.1, die algebraische Transformation

$$x = T^{-1}(\xi), \quad x, \xi \in \mathbb{R}^n.$$

Wird für die Inversion ein flacher Ausgang gewählt, ergeben sich folglich die Funktionen ψ_1 und ψ_2 entsprechend den Überlegungen zur Systeminversion direkt als

$$\psi_1 = T^{-1}(\xi_R(t)) \quad (4.19a)$$

$$\psi_2 = R^{-1}(T^{-1}(\xi_R(t))) \left(-a(\xi_R(t)) + \begin{pmatrix} y_{F1}^{(r_1)}(t) \\ \vdots \\ y_{Fm}^{(r_m)}(t) \end{pmatrix} \right) \quad (4.19b)$$

wobei, entsprechend Gleichung (4.6) $\xi_R(t)$ durch

$$\xi_R(t) = (\xi^1, \dots, \xi^m), \quad \xi^i = (y_{Fi}, \dots, y_{Fi}^{(r_i-1)})^T$$

gegeben ist. Damit ist die Systeminvertierung von einem flachen Ausgang aus besonders attraktiv, da keine Probleme mit der Stabilität der internen Dynamik auftreten können. Schwierig ist im Allgemeinen jedoch der Nachweis der Flachheit eines Systems, da diese, entsprechend Definition 4.5, über die Existenz eines flachen Ausgangs gezeigt wird, welcher

gewöhnlich nicht ohne weiteres zu finden ist. Es existieren jedoch einige Beweise mit denen gezeigt wurde, dass bestimmte Klassen von Systemen grundsätzlich flach sind. Zu diesen gehören insbesondere Systeme, welche sich auf *Brunovski-Normalform* transformieren lassen.

Um das Vorgehen zur Bestimmung eines flachen Ausgangs und der zugehörigen Transformationen ψ_1 und ψ_2 zu verdeutlichen, wird ein einachsiger Roboter als lineares Zwei-Massen-System mit $x = (q, \dot{q}, \theta, \dot{\theta})$, $u = \tau_M$ und $y = q$ in Zustandsform betrachtet. Dessen Systemmatrizen sind gegeben als

$$\begin{aligned} A &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{J_a} & -\frac{d}{J_a} & \frac{k}{J_a} & \frac{d}{J_a} \\ 0 & 0 & 0 & 1 \\ \frac{k}{J_m} & \frac{d}{J_m} & -\frac{k}{J_m} & -\frac{d}{J_m} \end{pmatrix} \\ B &= \left(0 \ 0 \ 0 \ \frac{1}{J_m} \right)^T \\ C &= \left(1 \ 0 \ 0 \ 0 \right). \end{aligned}$$

Bekanntlich kann ein lineares System, wenn es steuerbar ist, in Brunovski-Normalform dargestellt werden. Damit bedeutet die Steuerbarkeit auch gleichzeitig die Flachheit des Systems. Zum Test der Steuerbarkeit wird deshalb die Steuerbarkeitsmatrix

$$Q_S = \left(B \ AB \ A^2B \ A^3B \right)$$

ermittelt, deren Determinante

$$\det(Q_S) = \frac{k^2}{J_a^2 J_m^4}$$

für nicht triviale Werte der Parameter k , J_a und J_m ungleich Null ist. Das System ist damit steuerbar und somit auch flach. Gesucht wird nun noch der flache Ausgang. Die Transformationsmatrix von den Originalkoordinaten x in die Koordinaten der Regelungsnormalform ξ

$$\xi = Tx$$

berechnet sich mit $(Q_s^{-1})_n$, der letzten Zeile der inversen Steuerbarkeitsmatrix, als

$$\begin{aligned} T &= \begin{pmatrix} (Q_s^{-1})_n A^0 \\ \vdots \\ (Q_s^{-1})_n A^3 \end{pmatrix} \\ &= \begin{pmatrix} -\frac{J_m(d^2 - kJ_a)}{k^2} & -\frac{J_m d J_a}{k^2} & \frac{J_m d^2}{k^2} & 0 \\ \frac{J_m d}{k} & \frac{J_a J_m}{k} & -\frac{J_m d}{k} & 0 \\ -J_m & 0 & J_m & 0 \\ 0 & -J_m & 0 & J_m \end{pmatrix} \end{aligned}$$

Die Systemmatrizen des in Brunovski-Normalform transformierten Systems ergeben sich da-

mit zu:

$$\begin{aligned}\tilde{A} &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{k(J_m+J_a)}{J_a J_m} & -\frac{d(J_m+J_a)}{J_a J_m} \end{pmatrix} \\ \tilde{B} &= (0 \ 0 \ 0 \ 1)^T \\ \tilde{C} &= \left(\frac{k}{J_a J_m} \quad \frac{d}{J_a J_m} \quad 0 \quad 0 \right)\end{aligned}$$

Vielfache von ξ_1 sind flache Ausgänge des Systems. Als einfachste Möglichkeit wird $y_F = \xi_1$ gewählt. Für das System in Regelungsnormalform gilt daher:

$$\begin{pmatrix} y_F \\ \dot{y}_F \\ \ddot{y}_F \\ y_F^{(3)} \end{pmatrix} = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \end{pmatrix}$$

Damit ergibt sich die algebraische Berechnung der Zustände des Originalsystems aus dem flachen Ausgang und seinen Ableitungen zu

$$x = \psi_1(y_F, \dots, y_F^{(3)}) = T^{-1} \begin{pmatrix} y_F \\ \dot{y}_F \\ \ddot{y}_F \\ y_F^{(3)} \end{pmatrix}.$$

Der Eingang u ergibt sich aus der letzten Gleichung der Systembeschreibung in Regelungsnormalform

$$u = \psi_2(y_F, \dots, y_F^{(4)}) = y_F^{(4)} - (\tilde{A})_n \begin{pmatrix} y_F \\ \dot{y}_F \\ \ddot{y}_F \\ y_F^{(3)} \end{pmatrix}$$

wobei $(\tilde{A})_n$ die letzten Zeile der Matrix \tilde{A} ist.

An dieser Stelle erscheint die Aufgabe zunächst gelöst: ein flacher Ausgang ist gefunden, und die Funktionen zur Berechnung der Zustände und des Eingangs sind bestimmt. Um den flachen Ausgang in den ursprünglichen physikalischen Koordinaten vorgeben zu können, wird y_F als Funktion von x dargestellt.

$$y_F = \xi_1 = \frac{J_m}{k^2} (-d^2 - kJ_a)x_1 - dJ_a x_2 + d^2 x_3$$

Mit $x = (q, \dot{q}, \theta, \dot{\theta})^T$ sieht man, dass es sich um eine Kombination aus Gelenkposition, Gelenkgeschwindigkeit und Motorgeschwindigkeit handelt, welche sich nicht ohne weiteres physikalisch sinnvoll interpretieren lässt.

Da die Aufgabe in einer Inversion bezüglich dem Ausgang $y = x_1$ besteht, wird mit Hilfe der Brunovski-Normalform der Zusammenhang zwischen den Ausgängen y und y_F ermittelt.

$$y = \tilde{C}\tilde{x} = \frac{k}{J_a J_m} y_F + \frac{d}{J_a J_m} \dot{y}_F$$

Es ergibt sich eine Differentialgleichung erster Ordnung in y_F mit dem Eingang y . Damit muss zur Invertierung aber trotz der Flachheit des Systems eine Differentialgleichung gelöst werden.

Die Ursache hierfür ist, dass das System bezüglich seines Ausgangs y eine interne Dynamik aufweist. Die interne Dynamik ist aber eine strukturelle Eigenschaft des Systems bei gegebenen Ein- und Ausgängen und invariant gegenüber Zustandsstransformationen [SJK97]. Daher ist die Invertierung eines flachen Systems von einem nicht flachen Ausgang aus nicht allein mit algebraischen Gleichungen möglich. Die zu lösende Differentialgleichung, welche der internen Dynamik entspricht, entsteht bei der Berechnung des flachen Ausgangs y_F aus dem Ausgang y . Dies wird auch deutlich, wenn das System (4.16) um Ausgangsgleichungen der Form

$$y = h(x), \quad y \in \mathbb{R}^m$$

ergänzt wird. Wird y in Abhängigkeit des flachen Ausgangs dargestellt ergibt sich mit Gleichung (4.17)

$$y = h\left(\psi_1(y_F, \dot{y}_F, \dots, y_F^{(\beta)})\right)$$

Dabei handelt es sich, abhängig von $h(\cdot)$ und $\psi_1(\cdot)$, um eine Differentialgleichung höherer Ordnung in y_F , welche es zunächst zu lösen gilt. Da die Dimension der internen Dynamik durch den relativen Grad vorgegeben ist, kann die Ausgangsgleichung auch als

$$y = h\left(\psi_1(y_F, \dot{y}_F, \dots, y_F^{(n-r)})\right) \quad (4.20)$$

angegeben werden [HZ04].

Grundsätzlich entstehen jedoch bei einem flachheitsbasierten Vorgehen die gleichen Probleme wie bei der in Kapitel 4.3 dargestellten Invertierung. Zusätzlich müssen, wie bereits diskutiert, ein flacher Ausgang und die zugehörigen Transformationen gefunden werden. Aus diesen Gründen erscheint es bei Systemen, deren natürlicher Ausgang nicht flach ist, sinnvoller die Invertierung direkt ohne den Umweg über einen flachen Ausgang durchzuführen.

In Sonderfällen kann das flachheitsbasierte Vorgehen jedoch auch für $y \neq y_F$ nützlich sein. Mit den Gleichungen (4.19) und (4.20) erkennt man, dass durch die Einführung eines flachen Ausgangs die interne Dynamik und die Berechnung der Zustände und Eingänge separiert werden. Dies kann z. B. ausgenutzt werden, wenn anstatt der Trajektorienfolge nur Arbeitspunktwechsel betrachtet werden [HZ04]. Gefordert wird in diesem Fall

$$\begin{aligned} x_0 &= x(0) \\ x_T &= x(T) \end{aligned}$$

mit der Bewegungsdauer T , wobei x_0 und x_T Ruhelagen des Systems sind. Wegen Gleichung (4.17) gilt insbesondere

$$\begin{aligned} x_0 &= \psi_1(y_F(0), \dot{y}_F(0), \dots, y_F^{(\beta)}(0)) \\ x_T &= \psi_1(y_F(T), \dot{y}_F(T), \dots, y_F^{(\beta)}(T)) \end{aligned}$$

und damit gilt aufgrund der Invertierbarkeit von ψ_1

$$\begin{aligned} \left(y_F(0), \dot{y}_F(0), \dots, y_F^{(\beta)}(0)\right)^T &= \psi_1^{-1}(x_0) \\ \left(y_F(T), \dot{y}_F(T), \dots, y_F^{(\beta)}(T)\right)^T &= \psi_1^{-1}(x_T). \end{aligned}$$

Damit sind die flachen Koordinaten am Anfang und am Ende der Bewegung vorgegeben. Wird nun die Bahn zwischen den Ruhelagen in flachen Koordinaten geplant, so ist es nicht mehr notwendig die interne Dynamik zu berechnen. Dies ist offensichtlich insbesondere dann ein Vorteil, wenn die interne Dynamik nicht stabil ist.

Alternativ kann der Arbeitspunktwechsel auch anhand der Ausgänge vorgegeben sein.

$$\begin{aligned}y_0 &= y(0) \\ y_T &= y(T)\end{aligned}$$

Äquivalent zum vorherigen Fall lassen sich dann die flachen Koordinaten an den Ruhelagen durch Lösung von (4.20) bestimmen:

$$y = h(\psi_1(y_F, 0, \dots, 0))$$

4.5 Berechnung von inversen Systemen

Nach den vor allem aus theoretischer Sicht interessanten Betrachtungen zur Existenz und zu den Eigenschaften inverser Systeme werden jetzt Möglichkeiten für deren Berechnung untersucht. Dabei stellt sich vor allem die Frage nach einem praktikablen Weg zur Erzeugung der Inversen von komplexen Systemen und deren effizienter Auswertung. Diese Problematik wird in der Literatur kaum behandelt, stellt aber den Schlüssel für die praktische Anwendung von inversen Systemen dar.

4.5.1 Berechnung über die Normalform

Die erste Methode orientiert sich an der vorausgegangenen Darstellung zur Invertierbarkeit eines Systems. Wenn das System in Normalform transformiert wurde, dann sind, entsprechend Kapitel 4.2, die Gleichungen des inversen Systems bereits gefunden. Zur Berechnung des inversen Systems kann die Differentialgleichung für die interne Dynamik numerisch gelöst und die Eingänge über die zugehörigen algebraischen Gleichungen bestimmt werden. Mit der Transformation T ist damit auch der Vektor der Zustände x in den ursprünglichen Koordinaten bekannt.

Nachteilig an dieser Methode ist, dass sie nur auf Systeme der Form 4.5 angewendet werden kann. Daher muss sich das System zunächst auf die benötigte Zustandsform transformieren lassen. Anschließend wird der relative Grad des Systems untersucht. Dazu müssen die Bedingungen aus Definition 4.2 durch fortgesetztes Differenzieren der Ausgänge überprüft werden. Bei definiertem relativem Grad existiert eine Transformation T in Normalform. Deren erste r Komponenten lassen sich, wie in Abschnitt 4.1.2 dargestellt, relativ einfach aus den Lie-Ableitungen der Ausgänge bestimmen. Die Wahl der restlichen $n - r$ Komponenten von T ist aufwändiger, da durch geeignete Wahl die Regularität der Jacobimatrix $\left(\frac{\partial T}{\partial x}\right)(x_0)$ sichergestellt werden muss.

Insgesamt erscheint ein solches manuelles Vorgehen nur für einfache Systeme praktikabel. Zudem ist es unbefriedigend, das System zuerst auf Zustandsform und dann auf Normalform transformieren zu müssen, nur um im nächsten Schritt den Zustandsvektor durch Vorgabe der Ausgänge zu beschränken, um zur Zustandsdarstellung des inversen Systems zu gelangen.

4.5.2 Hirschorn Algorithmus

In der Literatur sind Algorithmen bekannt, welche die notwendige Differentiationsordnung der Ausgänge bestimmen und damit das inverse System automatisiert aufstellen können. Stellvertretend wird hier der Algorithmus von Hirschorn [Hir79] kurz dargestellt. Erweiterungen dieses Algorithmus sind zum Beispiel in [Sin81] und [Nv90] beschrieben. Allen diesen Algorithmen ist gemein, dass sie, ausgehend von einer Zustandsdarstellung des Systems, induktiv eine Reihe von Systemen aufstellen, aus welcher sich letztlich das inverse System ergibt. Dabei werden bei [Hir79] in jedem Schritt die zu differenzierenden Ausgangsgleichungen ermittelt. Dazu wird das k -te System als

$$\dot{x} = f(x) + g(x)u \quad (4.21a)$$

$$z_k = h_k(x) + j_k(x)u \quad (4.21b)$$

dargestellt. Die Variable z_k enthält die differenzierten Ausgänge, welche so permutiert wurden, dass sich $j_k(x)$ als

$$j_k(x) = \begin{pmatrix} j_{k1}(x) \\ 0_{r_k \times m} \end{pmatrix}$$

mit $r_k = \text{Rang}(j_k(x))$ schreiben lässt. Solange $r_k < m$ ist, müssen die letzten $m - r_k$ Gleichungen von 4.21b weiter differenziert werden. Wenn nach einer ausreichenden Anzahl von Schritten $j_k(x)$ nicht mehr singulär ist, kann schließlich das inverse System als

$$\dot{\hat{x}} = \hat{f}(\hat{x}) + \hat{g}(\hat{x})\hat{u} \quad (4.22a)$$

$$\hat{y} = \hat{h}(\hat{x}) + \hat{j}(\hat{x})\hat{u} \quad (4.22b)$$

angegeben werden, wobei die Funktionen $\hat{f}(\hat{x}), \hat{g}(\hat{x}), \hat{h}(\hat{x}), \hat{j}(\hat{x})$ entsprechend [Hir79] vorgegeben werden.

Der wesentliche Nachteil dieser Methode ist, dass die Dynamik (4.22a) des inversen Systems (4.22) gleich der Ordnung des invertierten Systems ist. Im Gegensatz dazu entstand bei der Berechnung in Abschnitt 4.5.1 ein inverses System minimaler Ordnung. Eine Dynamik höherer Ordnung, welche im Ein- Ausgangsverhalten nicht sichtbar ist, erscheint schon aus numerischer Sicht ungünstig. Zusätzlich erschwert die höhere Ordnung die Analyse der Stabilität des inversen Systems. Ein weiterer Nachteil ist auch hier die Einschränkung auf Systeme der Form (4.5) mit $j(x) = 0_m$ sowie die Notwendigkeit, das System vorab auf diese Form zu transformieren. Damit ist die Anwendung des Algorithmus zur praktischen Berechnung von inversen Systemen mit einigen Schwierigkeiten behaftet.

4.5.3 Berechnung als DAE

Nachteilig an den vorherigen Methoden ist die Notwendigkeit, das Modell in Zustandsform verfügbar zu haben. Zum einen fällt das Modell bei der physikalischen Modellierung in der Regel nicht in dieser Form an, sondern muss erst auf Zustandsform transformiert werden. Zum anderen verstellt die Zustandsdarstellung den Blick auf die zugrunde liegenden Gleichungen, was eine spätere Analyse, zum Beispiel hinsichtlich der notwendigen Differenzierbarkeit von Teilen des Modells, erschwert. Aus diesen Gründen ist es naheliegend zu versuchen, die physikalische Darstellung der Inversen direkt zu bearbeiten, um zu einer Realisierung der Inversen

mit minimaler Ordnung zu gelangen. Dazu wird als Beispiel wieder die Inverse eines Zwei-Massen-Systems betrachtet. Deren Gleichungen können direkt als

$$J_a \ddot{q} + k(q - \theta) + d(\dot{q} - \dot{\theta}) = 0 \quad (4.23a)$$

$$J_m \ddot{\theta} - k(q - \theta) - d(\dot{q} - \dot{\theta}) = \tau_M \quad (4.23b)$$

$$q = u(t) \quad (4.23c)$$

angegeben werden. Bei bekanntem q und damit auch bekanntem \dot{q} und \ddot{q} , ist Gleichung (4.23a) eine Differentialgleichung erster Ordnung in θ . Zur Berechnung von τ_M wird zusätzlich noch $\ddot{\theta}$ benötigt. Diese Größe kann über die Ableitung von (4.23a) bestimmt werden.

$$J_a \dot{q}^{(3)} + k(\dot{q} - \dot{\theta}) + d(\ddot{q} - \ddot{\theta}) = 0$$

Damit ergibt sich die Inverse in Zustandsdarstellung als

$$\begin{aligned} \dot{\theta} &= -\frac{k}{d}\theta + \dot{u}(t) + \frac{1}{d}(J_a \ddot{u}(t) + k u(t)) \\ \tau_M &= \frac{J_m k^2}{d^2}\theta + \frac{J_a J_m}{d}u^{(3)}(t) + \left(J_a + J_m + \frac{J_a J_m k}{d^2}\right)\ddot{u}(t) - \frac{J_m k^2}{d^2}u(t) \end{aligned}$$

mit der Ordnung eins. Dies ist die minimale Ordnung, wie sich leicht anhand des relativen Grades des Systems zeigen lässt.

Eine Ableitung wie im Beispiel ist für größere Systeme verständlicherweise nicht mehr praktikabel. Daher werden Algorithmen gesucht, welche für eine möglichst große Klasse von Systemen die zugehörigen Inversen in Zustandsform bestimmen können.

Im allgemeinsten Fall ist das mathematische Modell als System von Differentialgleichungen und algebraischen Gleichungen gegeben, welches als DAE (Differential Algebraic Equation) System bezeichnet wird. Dieses hat die Form

$$0 = f(x, \dot{x}, t, u(t)), \quad (4.24)$$

wobei x alle unbekanntenen Größen beinhaltet und $u(t)$ bekannte Funktionen der Zeit. Wird nun zusätzlich eine Definition von Ausgängen y vorgenommen, dann entsteht ein System der Form

$$0 = f(x, \dot{x}, t, u(t)) \quad (4.25a)$$

$$0 = f_y(x, \dot{x}, t, u(t), y), \quad (4.25b)$$

welches über die zusätzlichen Gleichungen $0 = f_y$ den Zusammenhang zwischen den Eingängen und Ausgängen des Systems beschreibt. Werden als Eingänge die Stellgrößen und als Ausgänge die Regelgrößen des Systems gewählt, dann erhält man eine Beschreibung des Systems, welche hier in Anlehnung an die in der Robotik üblichen Bezeichnungen als (Vorwärts-)Dynamik bezeichnet wird. Werden dagegen die Regelgrößen als bekannte Eingänge angenommen und die Stellgrößen als Ausgänge berechnet dann erhält man das zur vorigen Beschreibung inverse System, welches als inverse Dynamik bezeichnet wird. Mit Gleichung (4.25) erkennt man, dass der Vorwärts- und der inversen Dynamik die selben Gleichungen $f(x, \dot{x}, t)$ zugrunde liegen, es wird lediglich eine andere Interpretation vorgenommen. Die Berechnung eines inversen Systems kann damit auch als Lösung eines DAE-Systems unter Verwendung der entsprechenden Methoden zur Berechnung dieser Systeme durchgeführt werden. Eine in diesem Zusammenhang wesentliche Größe ist der (differentielle) Index einer DAE, der zunächst nach [BCP89] definiert werden soll.

Definition 4.6 (Index einer DAE).

Die kleinste Anzahl an Differentiationen ν von Gleichung (4.24) oder Teilen davon bezüglich t , welche benötigt wird um \dot{x} als kontinuierliche Funktion von x und t darzustellen, wird als Index der DAE 4.24 bezeichnet.

Systeme mit einem Index größer als eins werden als Systeme mit höherem Index bezeichnet. Der Zusammenhang zwischen Systemen mit höherem Index und inversen Systemen soll am Beispiel der horizontalen, reibungsfreien Bewegung einer Masse m verdeutlicht werden. Diese wird, dargestellt als DAE, beschrieben durch die Gleichungen

$$\begin{aligned} 0 &= v - \dot{r} \\ 0 &= m\dot{v} - f \end{aligned}$$

Wird als bekannter Eingang u die Kraft f gewählt, was der Vorwärtsdynamik entspricht, dann lässt sich das System sofort in Zustandsform schreiben.

$$\begin{aligned} \dot{r} &= v \\ \dot{v} &= \frac{u(t)}{m} \end{aligned}$$

Es mussten keine Gleichungen differenziert werden, das heißt das System mit Eingang $u = f$ hat den Index $\nu = 0$. Im Gegensatz dazu entsteht für die inverse Dynamik mit $u(t) = r$ das Gleichungssystem

$$\begin{aligned} 0 &= v - \dot{r} \\ 0 &= m\dot{v} - f \\ 0 &= r - u(t) \end{aligned}$$

Die unbekanntenen Größen sind damit $x = (r, v, f)$. Um \dot{x} als kontinuierliche Funktion von x und t darstellen zu können, muss die letzte Gleichung dreimal differenziert werden. Entsprechendes Differenzieren und Einsetzen der Gleichungen liefert das System in Zustandsdarstellung.

$$\begin{aligned} \dot{r} &= \dot{u}(t) \\ \dot{v} &= \ddot{u}(t) \\ \dot{f} &= m\dot{u}^{(3)}(t) \end{aligned}$$

Damit hat das inverse System, das heißt das System mit dem Eingang $u(t) = r$, den Index $\nu = 3$.

Zur Berechnung des inversen Systems ist es allerdings meist nicht notwendig das System entsprechend Definition 4.6 in der Form $\dot{x} = f(x, t)$ darzustellen. Dies führt im Beispiel der bewegten Masse sogar zu einer „überflüssigen“ Differentiation. Im Allgemeinen ist zur Berechnung eine Darstellung als System mit Index 1 ausreichend, was für das Beispiel der bewegten Masse auf die Gleichungen

$$\begin{aligned} \dot{r} &= u(t) \\ \dot{v} &= \dot{u}(t) \\ f &= m\ddot{u}(t) \end{aligned}$$

führt.

Die Entstehung einer DAE mit höherem Index ist typisch für die Inversion von physikalischen Systemen. Da diese Systeme gewöhnlich einen kausalen Zusammenhang zwischen Stellgröße und Regelgröße zeigen, hat die Vorwärtsdynamik integrierenden Charakter. Wird nun zur Berechnung des inversen Systems die Regelgröße vorgegeben, dann wird eine zusätzliche Zwangsbedingung erzeugt in welcher erst durch Differenzieren wieder die Stellgröße auftritt. Dies führt in der Regel auf eine DAE mit höherem Index.

Die in Kapitel 4.1 bei der Invertierung betrachteten Systeme der Form (4.5) stellen dabei einen interessanten Sonderfall dar. Das inverse System lässt sich in diesem Fall einfach darstellen als

$$\dot{x} = f(x) + g(x)y \quad (4.26a)$$

$$u = h(x) + j(x)y. \quad (4.26b)$$

Hier ist das System als semi-explizite DAE gegeben. Um Gleichung (4.26) in die Zustandsform

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = f(x, y, t)$$

zu überführen, muss die Zwangsbedingung $u = h(x) + j(x)y$ so oft differenziert und eingesetzt werden, bis sie nach der Ableitung des Ausgangs aufgelöst werden kann. Gemäß Definition 4.2 ist die Ordnung der Ableitung, in welcher die Gleichung nach y gelöst werden kann, durch den relativen Grad r bestimmt. Da die Matrix R regulär ist, kann die Gleichung nach y aufgelöst werden.

$$y = R(x)^{-1} \begin{pmatrix} u_1^{(r_1)} \\ \vdots \\ u_m^{(r_m)} \end{pmatrix}$$

Ein weiterer Differenzierungsschritt liefert daraufhin mit

$$\dot{y} = \frac{d}{dt} \left(R(x)^{-1} \begin{pmatrix} u_1^{(r_1)} \\ \vdots \\ u_m^{(r_m)} \end{pmatrix} \right)$$

die gewünschte Form, womit das System auf Zustandsform transformiert ist. Dazu wurde eine zusätzliche Differentiation der Zwangsbedingung benötigt. Damit lässt sich der folgende Zusammenhang zwischen Index und relativem Grad angeben:

$$\nu = \max_{i=1:m} r_i + 1$$

Die Invertierung von Systemen der Form (4.5) führt also immer dann auf eine DAE mit höherem Index, wenn $j(x)$ singulär ist. Dies ist für die meisten Systeme und insbesondere für die betrachteten Roboter entsprechend dem Modell aus Kapitel 2 der Fall.

Insofern kann die Berechnung inverser Systeme auch als Lösung von DAE-Systemen mit höherem Index interpretiert werden. Grundsätzlich ist die Berechnung von inversen Systemen auf Basis einer DAE-Beschreibung wünschenswert, da hiermit die größte Klasse von Modellen direkt behandelt werden kann. Allerdings ist die Lösung von DAE-Systemen mit höherem Index im Allgemeinen schwierig. Im Folgenden wird deshalb eine Methode gezeigt, mit der sich solche Systeme automatisiert in eine für die numerische Lösung einfachere Form transformieren lassen.

4.6 Berechnung von inversen Systemen mit Indexreduktion

Bekanntermaßen ist die direkte Lösung von DAE Systemen mit höherem Index numerisch schwierig. Dies liegt vor allem an der sogenannten Drift der numerischen Lösung welche dadurch entsteht, dass die algebraischen Zwangsbedingungen in differenzierter Form benutzt werden müssen. Zur Lösung des Problems wurden Techniken entwickelt, welche in der Lage sind, die Zwangsbedingungen zu stabilisieren. Diese sind jedoch nur für bestimmte Klassen von Systemen anwendbar [Bau72]. Eine andere Lösungsmöglichkeit besteht darin, die DAE so zu transformieren, dass ein System mit niedrigerem Index entsteht. Auch hierfür existieren mehrere Möglichkeiten, auf welche z. B. in [MS93] verwiesen wird. Problematisch bei der Indexreduktion ist, dass durch die zusätzlichen, differenzierten Gleichungen ein zwar konsistentes, aber überbestimmtes Gleichungssystem entsteht. Dieses Problem kann mit der Dummy Derivative Methode von Mattsson und Söderlind [MS93] behoben werden, welche im Rest der Arbeit verwendet wird und im Folgenden näher beschrieben ist.

Die Grundidee ist, die DAE automatisiert auf eine Zustandsdarstellung mit Index 1 zu transformieren, da die Lösung solcher Systeme numerisch günstiger ist. Dazu müssen, wie bereits gezeigt, Teile der DAE differenziert werden. Das erste Problem ist damit festzustellen, welche Gleichungen wie oft differenziert werden müssen. Dazu wird bei der Dummy Derivative Methode der Algorithmus von Pantelides verwendet [Pan88], welcher in Abschnitt 4.6.1 skizziert wird. Die Transformation auf Zustandsform mit Hilfe von Dummy Derivatives wird anschließend in Abschnitt 4.6.2 beschrieben.

4.6.1 Pantelides Algorithmus

Ursprünglich ist der Algorithmus von Pantelides zur konsistenten Initialisierung der Anfangsbedingungen von DAE Systemen gedacht gewesen. Diese müssen nicht nur die DAE erfüllen, sondern auch deren zeitliche Ableitungen. Dabei können beim Differenzieren der Gleichungen zwei Fälle auftreten.

1. Beim Differenzieren von Gleichungen entstehen genügend neue Variablen (höhere Ableitungen vorhandener Variablen), so dass die Gleichungen durch geeignete Wahl der neuen Variablen immer erfüllt werden können.
2. Die differenzierten Gleichungen sind neue Zwangsbedingungen für das DAE System, da nicht genügend neue Variablen zur Verfügung stehen.

Zur konsistenten Initialisierung ist der 2. Fall interessant. Der Algorithmus von Pantelides findet dabei die minimale Untermenge an Gleichungen der DAE, bei welcher durch Differenzieren neue Zwangsbedingungen entstehen. Entsprechend [Pan88] und [MS93] sind dies aber auch die Gleichungen, welche differenziert werden müssen, um den Index des DAE Systems zu reduzieren. Das Ergebnis des Algorithmus ist ein um die differenzierten Gleichungen erweitertes DAE-System, welches konsistent, aber aufgrund der zusätzlichen Gleichungen überbestimmt ist.

Um die minimale Untermenge der zu differenzierenden Gleichungen zu finden, müssen entsprechend den obigen Überlegungen, alle Gleichungsuntermengen gefunden werden, welche beim Differenzieren weniger neue Variablen als Gleichungen erzeugen. Diese Untermengen

werden auch als minimal strukturell singulär bezeichnet. Das Problem wird in [Pan88] gelöst indem die DAE als bipartiter Graph dargestellt wird. Dieser ist aufgeteilt in eine Menge von Gleichungsknoten und einer Menge von Variablenknoten. Eine Kante $(i - j)$ bedeutet, dass Variable j in Gleichung i auftritt.

Um einer Variablen eine Gleichung zuzuordnen, benutzt der Algorithmus von Pantelides zusätzlich den sogenannten Zuordnungsalgorithmus, welcher in Algorithmus 1 angegeben ist. Der Pantelides-Algorithmus geht nun die Gleichungen der Reihe nach durch und versucht

Algorithmus 1 Zuordnung einer Gleichung zu einer Variablen

```

procedure AUGMENTPATH(i,PATHFOUND)
  Color i                                     ▷ Markiere Gleichung i
  if  $\exists$  V-node mit  $\exists$ edge(i-j) und ASSIGN(j)=0 then
    PATHFOUND  $\leftarrow$  TRUE                 ▷ Gleichung enthält freie Variable
    ASSIGN(j)  $\leftarrow$  i                     ▷ Zuordnung der Gleichung zur Variablen
    RETURN
  end if
  for alle j für die gilt  $\exists$ edge(i-j) und j nicht markiert do
    Color j                                   ▷ Markiere Variablen der Gleichung
    k  $\leftarrow$  ASSIGN(j)                     ▷ finde zugeordnete Gleichungen
    AUGMENTPATH(k,PATHFOUND)
    if PATHFOUND then
      ASSIGN(j)  $\leftarrow$  i
      RETURN
    end if
  end for
end procedure

```

jede Gleichung einer in der höchsten Ableitung auftretenden Variable zuzuordnen. Gelingt dies nicht, was bedeutet Algorithmus 1 findet keinen Pfad, dann ist eine minimal strukturell singuläre Untermenge gefunden. Die zur Untermenge gehörenden Gleichungen werden daher differenziert. Als nächstes wird die Ableitung der betrachteten Gleichung untersucht. Dieses Verfahren wird solange wiederholt, bis alle Gleichungen der DAE untersucht wurden. Der zugehörige Algorithmus ist in Algorithmus 2 angegeben.

Natürlich besitzen nicht alle Gleichungssysteme, welche sich als DAE darstellen lassen, eine eindeutige Lösung. In diesen Fällen wird die DAE auch als strukturell inkonsistent bezeichnet. Entsprechend [Pan88] konvergiert der Pantelides Algorithmus nur dann, wenn das DAE-System nicht strukturell inkonsistent ist. Nach [Pan88] gilt für strukturell inkonsistente DAE-Systeme, dass für die Funktion

$$0 = f(x, \dot{x}, t),$$

welche aus der DAE durch Ersetzung von \dot{x} durch die zugehörigen, nicht differenzierten Variablen entsteht, die Jacobimatrix

$$\begin{pmatrix} \frac{\partial f}{\partial x} \end{pmatrix}$$

strukturell singulär wird. Diese Bedingung kann als Test verwendet werden, um vorab sicherzustellen, dass der Algorithmus konvergiert.

Algorithmus 2 Ermittlung der zu differenzierenden Gleichungen

Require: N ▷ Anzahl Gleichungen
Require: M ▷ Anzahl Variablen
Require: $\{f\}—\{X\}$ ▷ Systemgraph
Require: A ▷ Variablen-Assoziierungsliste
 $ASSIGN \leftarrow 0_M$ ▷ Zuordnungsliste Variablen \leftrightarrow Gleichungen
 $B \leftarrow 0_N$ ▷ Gleichungs-Assoziierungsliste
 $N' \leftarrow N$
for $k=1$ to N' **do** ▷ Für alle Ursprungsgleichungen
 $i \leftarrow k$ ▷ Untersuche Gleichung k
 repeat
 \langle Entferne Variablenknoten mit $A(\cdot) \neq 0 \rangle$ ▷ nur höchste Abl. relevant
 $vMark \leftarrow 0_M$ ▷ Entferne Markierung der Variablen-Knoten
 $eMark \leftarrow 0_N$ ▷ Entferne Markierung der Gleichungs-Knoten
 $PATHFOUND \leftarrow FALSE$
 $AUGMENTPATH(i,PATHFOUND)$ ▷ Zuordnung der Gleichung zu einer Variable
 if $PATHFOUND=FALSE$ **then** ▷ nicht erfolgreich (singuläres Teilsystem)
 for markierte Variablen-Knoten j **do** ▷ differenzierte Variablen einführen
 $M = M + 1$ ▷ Anzahl Variablen erhöhen
 \langle Erzeuge neuen Variablen-Knoten $M \rangle$
 $A(j) \leftarrow M$ ▷ Var. M ist Abl. von Var. j
 end for
 for markierte Gleichungs-Knoten l **do** ▷ differenzierte Gleichungen einführen
 $N = N + 1$ ▷ Anzahl Gleichungen erhöhen
 \langle Erzeuge neuen Gleichungs-Knoten $N \rangle$
 \langle Erzeuge Kanten entsprechend Var. von $N \rangle$
 $B(l) \leftarrow N$ ▷ Gl. N ist Abl. von Gl. l
 end for
 for markierte Variablen-Knoten j **do** ▷ Zuordnung kopieren
 $ASSIGN(A(j)) \leftarrow B(ASSIGN(j))$
 end for
 $i = B(i)$ ▷ Untersuche differenzierte Gleichungen
 end if
 until $PATHFOUND$
end for

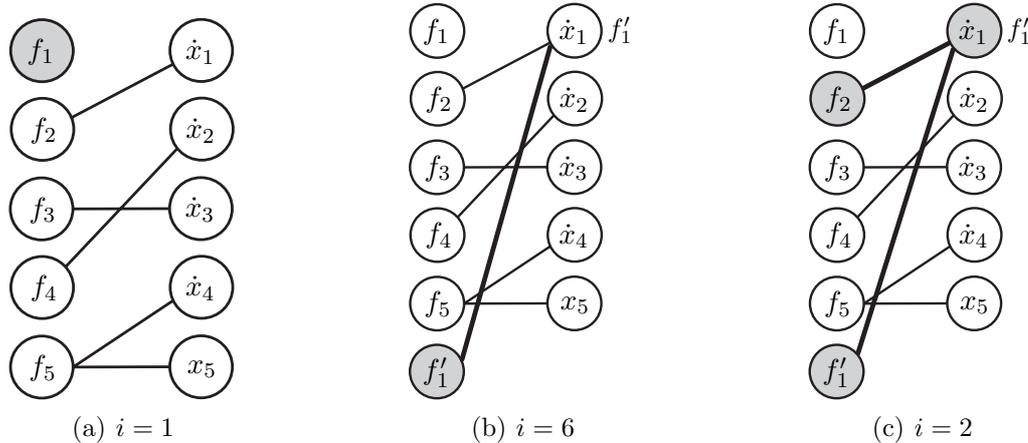


Abbildung 4.1: Graphen des Pantelides Algorithmus 1

Der Ablauf des Pantelides-Algorithmus wird im Folgenden am Beispiel der inversen Dynamik eines Zwei-Massen-Systems illustriert. Diese lässt sich in der benötigten DAE-Darstellung als

$$0 = f_1(\cdot) = x_1 - u(t) \quad (4.27a)$$

$$0 = f_2(\cdot) = \dot{x}_1 - x_2 \quad (4.27b)$$

$$0 = f_3(\cdot) = \dot{x}_3 - x_4 \quad (4.27c)$$

$$0 = f_4(\cdot) = J_A \dot{x}_2 + k(x_1 - x_3) + d(x_2 - x_4) \quad (4.27d)$$

$$0 = f_5(\cdot) = J_M \dot{x}_4 - k(x_1 - x_3) - d(x_2 - x_4) - x_5 \quad (4.27e)$$

darstellen. Zum Test auf strukturelle Singularität wird nach der Ersetzung von \dot{x} durch x in 4.27 die Jacobimatrix

$$\left(\frac{\partial f}{\partial x} \right) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ k & d + J_A & -k & -d & 0 \\ -k & -d & k & d + J_M & -1 \end{pmatrix}$$

bestimmt. Diese ist offensichtlich für Parameterwerte ungleich 0 nicht singular, womit sichergestellt ist, dass der Pantelides-Algorithmus konvergieren wird.

Der Algorithmus untersucht nun nacheinander alle Gleichungen und differenziert diese so oft wie nötig. Gezeigt werden im Anschluss immer die Graphen nach dem Aufruf von Algorithmus 1. Entsprechend Abbildung 4.1(a) kann f_1 nicht benutzt werden, um die höchste Ableitung einer der Variablen zu bestimmen. Daher wird die Gleichung differenziert. Die nachfolgende Untersuchung von f'_1 findet den in Abbildung 4.1(b) dargestellten Pfad, der Variablen \dot{x}_1 wird die Funktion f'_1 zugeordnet. Die Untersuchung der 2. Gleichung findet ebenfalls keinen Pfad, hinterlässt aber entsprechend Abbildung 4.1(c) die Knoten f_2, f'_1 und \dot{x}_1 markiert. Die entsprechenden Funktionen und Variablen werden daher differenziert und zum Graphen hinzugefügt. Die anschließende Analyse von Gleichung f'_2 findet, wie in Abbildung 4.2(a) dargestellt, wieder einen Pfad. Damit wird der Variablen \dot{x}_2 die Gleichung f'_2 zugewiesen. Die

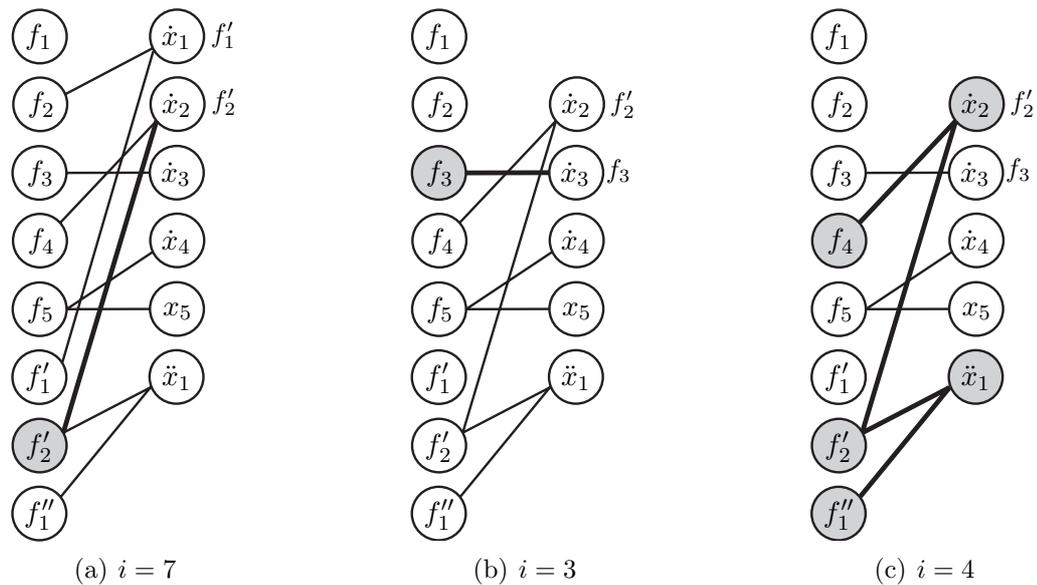


Abbildung 4.2: Graphen des Pantelides Algorithmus 2

ritte Gleichung muss nicht differenziert werden, sie wird nach Abbildung 4.2(b) direkt der Variablen \dot{x}_3 zugeordnet. Für die 4. Gleichung ist keine direkte Zuordnung möglich, statt dessen verbleiben in Abbildung 4.2(c) $f_4, f'_2, f''_1, \dot{x}_2$ und \ddot{x}_1 markiert. Diese werden wiederum differenziert. Die nachfolgende Untersuchung von f'_4 liefert, wie in Abbildung 4.3(a) dargestellt, die Zuordnung zu \dot{x}_4 . Im letzten Schritt kann die 5. Gleichung noch direkt der Variablen x_5 zugeordnet werden, was in Abbildung 4.3(a) zu sehen ist. Damit sind alle benötigten Ableitungen gefunden und der Algorithmus terminiert. Bei der Untersuchung der Gleichungen wurden 6 neue Gleichungen gefunden welche Zwangsbedingungen für das DAE System darstellen. Das um diese Gleichungen erweiterte System ist in 4.28 angegeben.

4.6.2 Dummy-Derivative Methode

Wird die DAE um die differenzierten Gleichungen ergänzt, dann liegt ein konsistentes, aber überbestimmtes Gleichungssystem vor. Um aus diesem wieder ein bestimmtes Gleichungssystem zu machen, werden ausgewählte Ableitungen von Variablen durch neue Variablen, die sogenannten Dummy-Derivatives, ersetzt. Da das Gleichungssystem in der Regel dünn besetzt sein wird, ist es sinnvoll, zunächst eine Block-Lower-Triangular (BLT) Transformation durchzuführen, wodurch das Problem in kleinere Teilprobleme unterteilt wird. Insgesamt ergibt sich damit für die Indexreduktion ein Ablauf wie in Algorithmus 3 dargestellt. Der Panteli-

Algorithmus 3 Vorgehen zur Indexreduktion

Require: DAE ist nicht strukturell singulär

Differentiation mit Pantelides Algorithmus

Permutation zur Transformation des differenzierten Problems auf BLT-Form

Blockweise (entsprechend BLT) Index Reduktion

des Algorithmus wurde bereits in Abschnitt 4.6.1 beschrieben. Die Transformation auf BLT

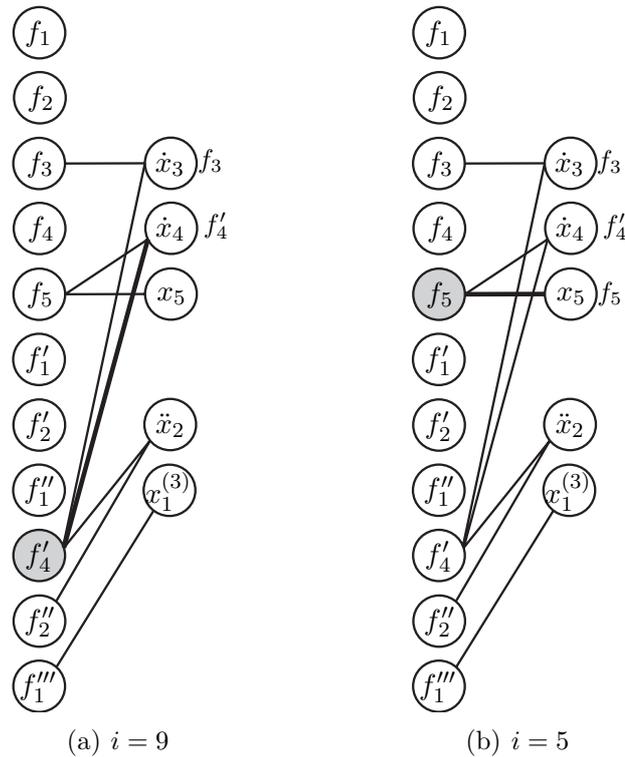


Abbildung 4.3: Graphen des Pantelides Algorithmus 3

Form ist ein Standardverfahren und muss für das Verständnis des Verfahrens nicht weiter erläutert werden. Im Folgenden wird deshalb nur der letzte Schritt, die Indexreduktion für einen Block i , ausführlicher beschrieben. Das Vorgehen zur Auswahl von Dummy-Derivatives ist in Algorithmus 4 dargestellt, Details und Beweise sind in [MS93] zu finden.

Vorausgesetzt wird, dass das Problem in BLT-Form vorliegt. Ferner seien die Gleichungen des i -ten Blocks g_i in absteigender Reihenfolge ihrer Differentiationsordnung angegeben. Schließlich wird angenommen, dass G_i , die quadratische Jacobimatrix von g_i bezüglich z_i , dem Vektor der höchsten Ableitungen der Variablen, vollen Rang besitzt. Der Algorithmus ersetzt nun schrittweise so lange Variablen aus z_i durch Dummy Derivatives, bis deren Anzahl gleich der Anzahl differenzierter Gleichungen ist. Die Nummer des Schrittes wird dabei durch den Index [Schritt] gekennzeichnet.

Zunächst wird eine Matrix $H_i^{[j]}$ aus den l Zeilen von $G_i^{[j]}$ erzeugt, deren zugehörige Gleichungen durch Differentiation entstanden sind. Nun werden l Spalten aus $H_i^{[j]}$ so ausgewählt, dass sie eine nicht singuläre Matrix $M_i^{[j]}$ bilden. Die zu diesen Spalten gehörenden Variablen aus $z_i^{[j]}$ werden später durch Dummy Derivatives ersetzt. Damit wurden für l abgeleitete Gleichungen l neue Variablen gefunden. Im nächsten Schritt werden die Gleichungen $g_i^{[j+1]}$ eine Differentiationsordnung niedriger untersucht. Dieses Vorgehen wird so lange wiederholt bis $g_i^{[j]}$ keine differenzierten Gleichungen mehr enthält. Nun kann das gesamte Gleichungssystem aufgestellt und die Ersetzung von Variablen durch Dummy-Derivatives durchgeführt werden.

Algorithmus 4 Auswahl von Dummy Derivatives

Require: < Problem ist in BLT-Form >**Require:** < g_i in absteigender Differentiationsordnung sortiert >**Require:** $\frac{\partial g_i}{\partial z_i}$ hat vollen Rang

$$z_i^{[1]} \leftarrow z_i$$

$$g_i^{[1]}(z_i^{[1]}) \leftarrow g_i(z_i)$$

$$G_i^{[1]} = \frac{\partial g_i^{[1]}}{\partial z_i^{[1]}} \leftarrow \frac{\partial g_i}{\partial z_i}$$

$$j \leftarrow 1$$

while $g_i^{[j]}$ enthält $m > 0$ differenzierte Gleichungen **do**

$$h_i^{[j]} = 0 \leftarrow g_i^{[j]} 1 \dots m$$

$$H_i^{[j]} = \frac{\partial h_i^{[j]}}{\partial z_i^{[j]}}$$

$$M_i^{[j]} \leftarrow H_i^{[j]} l_1, \dots, l_m$$

 $\triangleright l_1, \dots, l_m$ so, dass $M_i^{[j]}$ nicht singular

$$\hat{z}_i^{[j]} \leftarrow (z_{i,l_1}^{[j]}, \dots, z_{i,l_m}^{[j]})$$

$$g_i^{[j+1]} \leftarrow D^{-1} h_i^{[j]}$$

$$z_i^{[j+1]} \leftarrow D^{-1} \hat{z}_i^{[j]}$$

$$G_i^{[j+1]} \leftarrow M_i^{[j]}$$

$$j \leftarrow j + 1$$

end while

$$Nx = \begin{pmatrix} g_i^{[j]} \\ \dots \\ g_i^{[1]} \end{pmatrix} = 0$$

Alle $z_i^{[1]} \dots z_i^{[j]}$ durch Dummy Derivatives ersetzen

Beispiel: lineares Zwei-Massen-System

Das Vorgehen wird wieder am Beispiel des Zwei-Massen-Systems illustriert. Das erweiterte Gleichungssystem für die weitere Bearbeitung ergibt mit dem Pantelides Algorithmus:

$$0 = x_1 - u(t) \quad (4.28a)$$

$$0 = \dot{x}_1 - x_2 \quad (4.28b)$$

$$0 = \dot{x}_3 - x_4 \quad (4.28c)$$

$$0 = J_A \dot{x}_2 + k(x_1 - x_3) + d(x_2 - x_4) \quad (4.28d)$$

$$0 = J_M \dot{x}_4 - k(x_1 - x_3) - d(x_2 - x_4) - x_5 \quad (4.28e)$$

$$0 = \dot{x}_1 - \dot{u}(t) \quad (4.28a')$$

$$0 = \ddot{x}_1 - \dot{x}_2 \quad (4.28b')$$

$$0 = \ddot{x}_1 - \ddot{u}(t) \quad (4.28a'')$$

$$0 = J_A \ddot{x}_2 + k(\dot{x}_1 - \dot{x}_3) + d(\dot{x}_2 - \dot{x}_4) \quad (4.28d')$$

$$0 = x_1^{(3)} - \ddot{x}_2 \quad (4.28b''')$$

$$0 = x_1^{(3)} - u^{(3)}(t) \quad (4.28a''')$$

Das differenzierte Problem wird damit gebildet aus den Gleichungen (4.28a'''), (4.28b'''), (4.28d'), (4.28c) und (4.28e). Dargestellt in BLT Form bezüglich der höchsten Ableitungen ergibt sich somit

$$0 = x_1^{(3)} - u^{(3)}(t)$$

$$0 = x_1^{(3)} - \ddot{x}_2$$

$$0 = J_A \ddot{x}_2 + k(\dot{x}_1 - \dot{x}_3) + d(\dot{x}_2 - \dot{x}_4)$$

$$0 = \dot{x}_3 - x_4$$

$$0 = J_M \dot{x}_4 - k(x_1 - x_3) - d(x_2 - x_4) - x_5$$

Der Vektor der höchsten Ableitungen ist $z_1 = (x_1^{(3)} \quad \ddot{x}_2 \quad \dot{x}_3 \quad \dot{x}_4 \quad x_5)^T$. Die Jacobimatrix des differenzierten Problems bezüglich der höchsten Ableitungen z_1 ergibt sich dann zu

$$\frac{\partial g_1}{\partial z_1} = G_1^{[1]} = \begin{array}{l} (4.28a''') \\ (4.28b''') \\ (4.28d') \\ (4.28c) \\ (4.28e) \end{array} \begin{pmatrix} x_1^{(3)} & \ddot{x}_2 & \dot{x}_3 & \dot{x}_4 & x_5 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & J_A & -k & -d & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & J_M & -1 \end{pmatrix}$$

Im differenzierten Problem sind 3 differenzierte Gleichungen enthalten, daher wird

$$H_1^{[1]} = \begin{array}{l} (4.28a''') \\ (4.28b''') \\ (4.28d') \end{array} \begin{pmatrix} x_1^{(3)} & \ddot{x}_2 & \dot{x}_3 & \dot{x}_4 & x_5 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & J_A & -k & -d & 0 \end{pmatrix}$$

Um eine nicht singuläre Submatrix $M_1^{[1]}$ zu bilden, können entweder die vierte und fünfte Spalte, oder die dritte und fünfte Spalte gestrichen werden. Unter Verwendung der ersten

Möglichkeit ergibt sich

$$M_1^{[1]} = \begin{array}{l} (4.28a''') \\ (4.28b'') \\ (4.28d') \end{array} \begin{pmatrix} x_1^{(3)} & \ddot{x}_2 & \dot{x}_3 \\ 1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & J_A & -k \end{pmatrix}.$$

Damit werden die Ableitungen $x_1^{(3)}$, \ddot{x}_2 und \dot{x}_3 durch Dummy Derivatives ersetzt. Im nächsten Schritt wird das eine Differentiationsstufe geringere Vorgängerproblem betrachtet.

Der Vektor der höchsten Ableitungen ist jetzt $z_2 = (\dot{x}_1 \quad \dot{x}_2 \quad x_3)^T$. Die Jacobimatrix des Problems bezüglich z_2 ist nun

$$G_1^{[2]} = M_1^{[1]} = \begin{array}{l} (4.28a'') \\ (4.28b') \\ (4.28d) \end{array} \begin{pmatrix} \ddot{x}_1 & \dot{x}_2 & x_3 \\ 1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & J_A & -k \end{pmatrix}.$$

Es sind zwei differenzierte Gleichungen enthalten, daher wird $H_1^{[2]}$ zu

$$H_1^{[2]} = G_1^{[2]} = M_1^{[1]} = \begin{array}{l} (4.28a'') \\ (4.28b') \end{array} \begin{pmatrix} \ddot{x}_1 & \dot{x}_2 & x_3 \\ 1 & 0 & 0 \\ 1 & -1 & 0 \end{pmatrix}.$$

Eine nicht singuläre Submatrix wird durch die ersten beiden Spalten gebildet:

$$M_1^{[2]} = \begin{array}{l} (4.28a'') \\ (4.28b') \end{array} \begin{pmatrix} \ddot{x}_1 & \dot{x}_2 \\ 1 & 0 \\ 1 & -1 \end{pmatrix}$$

Damit werden zusätzlich \ddot{x}_1 und \dot{x}_2 durch Dummy Derivatives ersetzt. Eine weitere Differentiationsordnung niedriger ist $z_3 = (\dot{x}_1 \quad x_2)^T$ und

$$G_1^{[3]} = M_1^{[2]} = \begin{array}{l} (4.28a') \\ (4.28b) \end{array} \begin{pmatrix} \dot{x}_1 & x_2 \\ 1 & 0 \\ 1 & -1 \end{pmatrix}$$

Auch hier ist noch eine differenzierte Gleichung enthalten.

$$H_1^{[3]} = \begin{array}{l} (4.28a') \end{array} \begin{pmatrix} \dot{x}_1 & x_2 \\ 1 & 0 \end{pmatrix}$$

Die erste Spalte bildet eine nicht singuläre Submatrix,

$$M_1^{[3]} = \begin{array}{l} (4.28a') \end{array} \begin{pmatrix} \dot{x}_1 \\ 1 \end{pmatrix}$$

womit auch \dot{x}_1 zu einer Dummy Derivative wird. Im nächsten Schritt ist keine differenzierte Gleichung mehr vorhanden und die Schleife bricht ab.

$$G_1^{[4]} = M_1^{[3]} = \begin{array}{l} (4.28a) \end{array} \begin{pmatrix} x_1 \\ 1 \end{pmatrix}$$

Abschließend stellt der Algorithmus das zugehörige Index 1 Gleichungssystem auf und ersetzt die entsprechenden Ableitungen durch Dummy Derivatives.

$$0 = x_1 - u(t) \quad (4.29a)$$

$$0 = x_1' - \dot{u}(t) \quad (4.29b)$$

$$0 = x_1' - x_2 \quad (4.29c)$$

$$0 = x_1'' - \ddot{u}(t) \quad (4.29d)$$

$$0 = x_1'' - x_2' \quad (4.29e)$$

$$0 = J_A x_2' + k(x_1 - x_3) + d(x_2 - x_4) \quad (4.29f)$$

$$0 = x_1''' - u^{(3)}(t) \quad (4.29g)$$

$$0 = x_1''' - x_2'' \quad (4.29h)$$

$$0 = J_A x_2'' + k(x_1' - x_3') + d(x_2' - \dot{x}_4) \quad (4.29i)$$

$$0 = x_3' - x_4 \quad (4.29j)$$

$$0 = J_M \dot{x}_4 - k(x_1 - x_3) - d(x_2 - x_4) - x_5 \quad (4.29k)$$

Nun liegt ein Gleichungssystem mit 11 Gleichungen und Variablen vor, welches nur einen Zustand hat. Wie schon bei der Bestimmung von $M_1^{[1]}$ festgestellt, ist die Auswahl der Dummy Derivatives nicht eindeutig. Im gezeigten Fall verbleibt \dot{x}_4 in den Gleichungen, womit als Zustand $x_4 = \dot{\theta}$ gewählt ist. Werden bei der Festlegung von $M_1^{[1]}$ dagegen die dritte und fünfte Spalte gestrichen, dann wird \dot{x}_4 als Dummy Derivative gewählt und \dot{x}_3 verbleibt im Gleichungssystem. Damit ist $x_3 = \theta$ der Zustand des Systems.

Wie an diesem Beispiel illustriert, können mit der Dummy Derivative Methode inverse Systeme aufgestellt werden. Dazu wird das inverse System als DAE mit höherem Index formuliert und anschließend durch Differentiation von Gleichungen und Einführung neuer Variablen auf ein System mit Index 1 transformiert. Damit ist es, ausgehend von einer physikalischen Beschreibung des Systems möglich, das inverse System direkt in Zustandsform anzugeben. Dies macht die Dummy Derivative Methode zu einem wichtigen Werkzeug für die Berechnung von inversen Systemen. Allerdings bemerkt man bereits am einfachen Beispielsystem, dass der Aufwand zur Bestimmung der Inversen erheblich ist und eine manuelle Durchführung für komplexere Systeme unrealistisch ist. Daher werden entsprechende Werkzeuge benötigt. Ferner ist bei der Anwendung zu beachten, dass eine erfolgreiche Aufstellung der Inversen weder sicherstellt, dass die Inverse im gesamten Arbeitsraum definiert ist, noch ob die Inverse stabil ist. Die Methode ist somit sehr gut zur Synthese der Inversen geeignet, aber nicht unbedingt zu deren Analyse.

4.6.3 Invertierung mit Dymola

Es hat sich gezeigt, dass bei der objektorientierten Modellierung sehr häufig Probleme mit höherem Index auftreten. Als Beispiel wird das System in Abbildung 4.4 gezeigt, welches aus zwei verbundenen, angetriebenen Massen besteht, auf welche ein Antriebsmoment einwirkt. Die Gleichungen des Systems ergeben sich zu

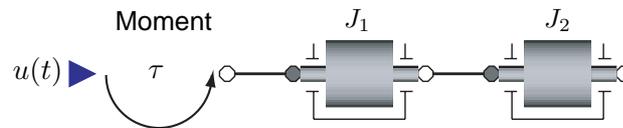


Abbildung 4.4: Objektdiagramm eines Systems mit zwei Massen

$$\tau_{a1} = u(t) \quad (4.30a)$$

$$J_1 \ddot{\varphi}_1 = \tau_{a1} - \tau_{b1} \quad (4.30b)$$

$$J_2 \ddot{\varphi}_2 = \tau_{a2} \quad (4.30c)$$

$$\varphi_1 = \varphi_2 \quad (4.30d)$$

$$\tau_{b1} = \tau_{a2} \quad (4.30e)$$

Offenbar muss Gleichung (4.30d) zur Transformation auf Zustandsform zweimal differenziert werden womit selbst für dieses einfache Modell ein Problem mit höherem Index vorliegt. Natürlich wäre es im gezeigten Beispiel einfach das Problem zu umgehen indem die beiden Massen durch eine einzige Masse mit der Trägheit $J_1 + J_2$ ersetzt werden. Bei aufwendigeren Modellen sind solche Zusammenhänge aber nicht mehr ohne weiteres erkennbar. Außerdem widerspricht ein solches Vorgehen der objektorientierten Modellierung, bei der einzelne Komponenten modelliert werden. Damit sind Algorithmen zur Behandlung solcher Probleme ein elementarer Baustein eines Simulationssystems für objektorientierte Modelle. Im verwendeten Programm Dymola [Dyn04] ist für die Behandlung von DAE Systemen mit höherem Index die gerade beschriebene Dummy Derivative Methode implementiert. Damit kann Dymola grundsätzlich auch zur Berechnung von inversen Modellen mit einem höheren Index, wie sie bei elastischen Robotern auftreten, verwendet werden.

Das System in Abbildung 4.4 kann von Dymola direkt behandelt werden. Für inverse Systeme sind noch ein paar formale Erweiterungen notwendig, um die Berechnung in Dymola zu ermöglichen. Dies soll am Beispiel der Inversion einer linearen Übertragungsfunktion gezeigt werden. Allgemein wird das Streckenmodell vordefinierte Ein- und Ausgänge besitzen, dargestellt durch die entsprechenden Konnektoren. Entsprechend den Regeln zum Aufbau von Blockschaltbildern lassen sich keine Eingänge oder Ausgänge untereinander verbinden. Genau dies ist für die Berechnung von inversen Modellen jedoch notwendig. Es werden daher noch zwei spezielle Blöcke benötigt, welche eine solche Verbindung ermöglichen. Diese sind im Anhang B.3 beschrieben.

Wie am Beispiel aus Abschnitt 4.6.2 gezeigt, ist es möglich, dass durch Differenzieren von Gleichungen Ableitungen des Eingangs des inversen Modells benötigt werden. Grundsätzlich wäre es möglich, die benötigten Ableitungen durch den Algorithmus zu sammeln und entsprechende Eingänge für das inverse Modell bereitzustellen, welche vom Benutzer dann geeignet belegt werden müssen. Bei der Implementierung in Dymola wird jedoch verlangt, dass ein simulationsfähiges Gesamtmodell vorgegeben wird. Somit müssen sich die notwendigen Ableitungen der Eingangsgrößen aus dem Modell selbst berechnen lassen. Dafür gibt es verschiedene Möglichkeiten.

Planungsalgorithmus:

Der Algorithmus zur Bahnplanung liefert Bahnen mit den nötigen Differenzierbarkeits-eigenschaften und lässt sich als Modelica Modell darstellen. Dies ist erfahrungsgemäß

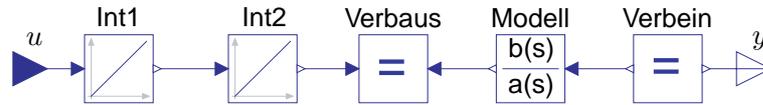


Abbildung 4.5: Inversion mit Integrator-kette am Eingang

nur in wenigen Fällen möglich. Insbesondere Algorithmen für optimale Bahnen lassen sich aufgrund der notwendigen Optimierungsverfahren gewöhnlich nicht ohne weiteres als Modelica Code implementieren.

Integrator-ketten:

Die Eingänge des inversen Systems werden als die Enden von hinreichend langen Integrator-ketten realisiert. Damit fallen die Ableitungen der Eingänge direkt an. Nachteilig ist jedoch, dass nun zur Simulation eine Planung auf Ebene der maximalen Ableitungen der Eingänge notwendig ist, welche gewöhnlich ebenfalls nicht ohne weiteres realisierbar ist.

Filter:

Die Eingänge des inversen Systems sind die Ausgänge von Filtern geeigneter Ordnung und Charakteristik. Geeignet sind z. B. Bessel-Tiefpässe. Mit Hilfe der Filter können die Ableitungen der Eingänge ebenfalls konsistent bestimmt werden. Die Trajektorienplanung kann weiterhin auf den bisherigen Eingangsgrößen durchgeführt werden, allerdings erfolgt im Anschluss noch eine Filterung der Trajektorie. Die Dynamik des Filters kann jedoch im Rahmen von numerischen Grenzen frei eingestellt werden, womit eine geringe Verzerrung des Eingangssignals sichergestellt werden kann.

Polynome:

Der Verlauf der Eingangsgrößen wird durch Polynome geeigneter Ordnung dargestellt, welche die notwendigen Differenzierbarkeitseigenschaften sicherstellen. Diese Lösung ermöglicht ebenfalls eine Trajektorienplanung auf den bisherigen Eingangsgrößen, allerdings mit Polynomen entsprechender Ordnung, was bei vorgegebenen Bahnen ebenfalls Schwierigkeiten mit sich bringt.

Je nach Anforderung bieten sich somit verschiedene Möglichkeiten an. Grundsätzlich ist es auch möglich, für die Codeerzeugung eine Integrator-kette vorzugeben, welche anschließend manuell aus dem Simulationscode entfernt und durch entsprechende Eingänge einer externen Planung ersetzt wird. Damit ist es z. B. möglich, einen beliebigen Planungsalgorithmus zu verwenden, welcher lediglich die gewünschten Differenzierbarkeitseigenschaften sicherstellen muss. Der Vorteil dabei ist, dass es nicht notwendig ist, den gesamten Algorithmus als Modelica-Modell darzustellen, was meistens nicht praktikabel ist.

Insgesamt bietet Dymola, aufgrund der implementierten Dummy Derivative Methode, eine sehr elegante Möglichkeit zur Berechnung von inversen Systemen. Dazu muss das inverse System zunächst als Modelica Modell beschrieben werden. Wenn für das System bereits ein Modelica Modell vorliegt, dann ist die Erweiterung auf das inverse Modell in einfacher Weise möglich. Wie beim Beispiel der Übertragungsfunktion in Abbildung 4.5, in dem die Übertragungsfunktion bereits als Modelica Modell vorliegt, müssen zur Definition der Inversen lediglich die Ein- und Ausgänge vertauscht werden und die nötigen Ableitungen der Ausgänge des Systems mit einer der genannten Methoden zur Verfügung gestellt werden.

Kapitel 5

Invertierung des Robotermodells

In diesem Kapitel wird die Inverse der Roboterdynamik aufgestellt, wozu die Bewegungsgleichungen analysiert werden. Die Analyse liefert einen neuen Algorithmus, mit welchem die zur Invertierung der Roboterdynamik notwendige Ordnung der Ableitungen der Bewegungsgleichungen bestimmt werden können. Da das in Kapitel 2 aufgestellte Modell die notwendigen Bedingungen zur Invertierung nicht erfüllt, wird ein approximiertes Modell aufgestellt. Anschließend wird die Stabilität des inversen Robotermodells analysiert und Bedingungen für die globale asymptotische Stabilität angegeben. Schließlich wird ein Modelica Modell der inversen Dynamik des KR15/2 aufgestellt und mit der in Kapitel 4 beschriebenen Methoden der Indexreduktion berechnet.

Bei der Invertierung des Robotermodells ist ein System gesucht, welches bei gegebenen Solltrajektorien für die Gelenkwinkel den zugehörigen Verlauf der Zustandsgrößen und Motormomente bestimmt. In der Robotik wird das System dessen Eingang die Stellgrößen und dessen Ausgang die Gelenkwinkel sind als *(Vorwärts-)Dynamik* bezeichnet. Entsprechend wird die gesuchte Inverse dieses Systems als *inverse Dynamik* bezeichnet. Die Eigenschaften der inversen Dynamik für RLFJ Roboter werden in diesem Kapitel untersucht und ein Modell zur Berechnung hergeleitet.

Dafür wird zunächst das Robotermodell nach Gleichung (2.59) nochmals angegeben:

$$\begin{pmatrix} M_{Arm}(q) & S(q) \\ S(q)^T & J \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\theta} \end{pmatrix} + \begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} + \begin{pmatrix} H_{GU} (k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta})) + \tau_{RG} \\ -H_A^{-1} (k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta})) + \tau_{RA} \end{pmatrix} = \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix}$$

5.1 Relativer Grad

Eine Möglichkeit zur Untersuchung der Invertierbarkeit ist die Bestimmung des relativen Grades, wozu die Dynamik zunächst in Zustandsform transformiert werden muss. Dies geschieht durch Invertierung der Massenmatrix,

$$M = \begin{pmatrix} M_{Arm}(q) & S(q) \\ S(q)^T & J \end{pmatrix}$$

was aufgrund der Eigenschaften von Massenmatrizen immer möglich ist. Unter Ausnutzung von Gleichung (2.27) kann die Inverse von M durch geeignete Faktorisierung [Rea94] als

$$M^{-1} = \begin{pmatrix} M_G^{-1} & -M_G^{-1}S J^{-1} \\ -J^{-1}S^T M_G^{-1} & J^{-1} + J^{-1}S^T M_G^{-1}S J^{-1} \end{pmatrix}$$

dargestellt werden.

Die von außen auf das System einwirkenden Stellgrößen sind $u = (\tau_{Ext} \ \tau_M)$, die Regelgrößen $y = (q \ \tau_{Ext})$. Offensichtlich sind die Vorgaben für τ_{Ext} immer erfüllbar, da τ_{Ext} ja als Eingang zur Verfügung steht. Aus diesem Grund spielt τ_{Ext} zur Bestimmung des relativen Grades keine Rolle, weshalb nur das System mit Eingang $u = \tau_M$ und Ausgang $y = q$ untersucht wird. Damit ergibt sich die Zustandsdarstellung:

$$\begin{aligned} \ddot{q} = & - (M_G^{-1} \quad -M_G^{-1}S J^{-1}) \left(\begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} \right) \\ & + \begin{pmatrix} H_G \mathbb{U} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) + \tau_{RG} \\ -H_A^{-1} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) + \tau_{RA} \end{pmatrix} - \begin{pmatrix} 0 \\ u \end{pmatrix} \end{aligned} \quad (5.1a)$$

$$\begin{aligned} \ddot{\theta} = & - (-J^{-1}S^T M_G^{-1} \quad J^{-1} + J^{-1}S^T M_G^{-1}S J^{-1}) \left(\begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} \right) \\ & + \begin{pmatrix} H_G \mathbb{U} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) + \tau_{RG} \\ -H_A^{-1} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) + \tau_{RA} \end{pmatrix} - \begin{pmatrix} 0 \\ u \end{pmatrix} \end{aligned} \quad (5.1b)$$

$$y = q \quad (5.1c)$$

Man erkennt, dass zweimaliges Differenzieren der Ausgangsgleichung (5.1c) auf $y = \ddot{q}$ und damit auf Gleichung (5.1a) führt. In dieser ist der Eingang u bereits enthalten. Um zu testen, ob sich die Eingänge mit dieser Gleichung bereits berechnen lassen, wird die Jacobimatrix $J_{\ddot{y}}(u)$ untersucht.

$$J_{\ddot{y}}(u) = \left[\frac{\partial \ddot{y}_i}{\partial u_j} \right] = -M_G^{-1}S J^{-1} \quad (5.2)$$

Die Matrix $J_{\ddot{y}}(u)$ ist nur dann invertierbar, wenn

$$M_G J_{\ddot{y}}(u) J = -S \quad (5.3)$$

invertierbar ist. Dies ist jedoch nicht möglich, da S entsprechend Gleichung (2.25) immer strukturell singulär ist. Damit sind weitere Ableitungen der Ausgänge notwendig.

Nachteilig an dieser Vorgehensweise ist jedoch, dass die Analyse durch die vorausgegangene Transformation in Zustandsform erschwert wird. Wie schon bei der zweiten Ableitung des Ausgangs, hängt die Invertierbarkeit von $J_{y^{(i)}}(u)$ von Kombinationen der Matrizen des Modells und deren strukturellen Eigenschaften ab. Damit wird die Untersuchung unnötig kompliziert. Wie im nächsten Kapitel gezeigt wird, können die Bedingungen zur Inversion der Roboterdynamik eleganter durch eine Analyse der Bewegungsgleichungen bestimmt werden.

5.2 Analyse des Robotermodells

In diesem Kapitel werden die Bedingungen zur Inversion der Bewegungsgleichungen untersucht. Dies geschieht zunächst für einen einachsigen Roboter. Anschließend wird das Vorgehen auf mehrachsige Roboter erweitert.

5.2.1 Einachsiger Roboter

Um die Herleitung übersichtlicher zu halten wird zunächst die inverse Dynamik für einen Roboter mit nur einer Achse, welche parallel zum Schwerfeld orientiert ist, betrachtet. Auf diese Weise können die Auswirkungen des Antriebsstranges unabhängig vom Mehrkörpersystem untersucht werden. Entsprechend der Herleitung in Kapitel 2 wird dieser Roboter durch die Gleichungen

$$\tau_G = -M_{Arm}\ddot{q} - \tau_{Ext} \quad (5.4)$$

$$\tau_{a,RG} = r_G(\dot{q}, \tau_G, \tau_{a,RG}) + \tau_G \quad (5.5)$$

$$\tau_{a,WG} = -\frac{1}{\eta_G(\tau_{a,WG}\dot{q})}\tau_{a,RG} \quad (5.6)$$

$$\mathbf{u}(q) = \mathbf{u}_0 + \sum_{j=1}^{n_u} \hat{\mathbf{u}}_j \cos(\tilde{\mathbf{u}}_j q + \bar{\mathbf{u}}_j) \quad (5.7)$$

$$\tau_{a,E} = \frac{1}{\mathbf{u}(q)}\tau_{a,WG} \quad (5.8)$$

$$\varphi = \mathbf{u}_0 q + \sum_{j=1}^{n_u} \frac{\hat{\mathbf{u}}_j}{\tilde{\mathbf{u}}_j} \sin(\tilde{\mathbf{u}}_j q + \bar{\mathbf{u}}_j) + \mathbf{u}_C \quad (5.9)$$

$$0 = d \cdot (\dot{\varphi} - \dot{\theta}) + k(\varphi - \theta) - \tau_{a,E} \quad (5.10)$$

$$\tau_{a,WA} = \frac{1}{\eta_A(\tau_{a,WA}\dot{\theta})}\tau_{a,E} \quad (5.11)$$

$$\tau_A = -r_A(\dot{\theta}, \tau_A, \tau_{a,WA}) - \tau_{a,WA} \quad (5.12)$$

$$\tau_M = J\ddot{\theta} - \tau_A \quad (5.13)$$

beschrieben, welche hier bereits entsprechend der Lösungsreihenfolge für die inverse Dynamik umgeformt sind.

Vorgegeben sind die Referenztrajektorien des Ausgangs $q(t)$ und des externen Moments τ_{Ext} sowie deren Ableitungen bis zu einer bestimmten Ordnung. Gesucht sind die Referenztrajektorien für die Zustände $q, \dot{q}, \theta, \dot{\theta}$ und die Stellgröße τ_M . Aus den Vorgaben lässt sich über Gleichung (5.4) τ_G direkt berechnen. Mit Gleichung (5.5) soll nun $\tau_{a,RG}$ bestimmt werden. Dies ist jedoch, entsprechend Gleichung (2.42c), nur für $\dot{q} \neq 0$ möglich, da aufgrund der Haftcharakteristik eine Berechnung von $\tau_{a,RG}$ im Stillstand nicht möglich ist. Deshalb muss bei der Berechnung der Inversen die Reibung r_G durch eine noch zu definierende Approximation $\hat{r}_G(\dot{q})$ ersetzt werden, welche die Berechnung von $\tau_{a,RG}$ erlaubt. Bei der folgenden Berechnung von $\tau_{a,WG}$ wird ausgenutzt, dass die Leistungsflussrichtung an beiden Flanschen des Wirkungsgradmodells aufgrund der Gleichungen (2.49a) und (2.49b) identisch ist. Damit

kann Gleichung (5.6) auch geschrieben werden als

$$\tau_{a,WG} = \frac{1}{\eta_G(\tau_{a,RG}\dot{q})} \tau_{a,RG} \quad (5.14)$$

Nach der Berechnung der aktuellen Übersetzung mit Gleichung (5.7) kann $\tau_{a,E}$ über (5.8) ermittelt werden. Zusätzlich wird der aktuelle Winkel φ auf der Antriebsseite der Übersetzung mit Gleichung (5.9) bestimmt. Zur Berechnung der in Gleichung (5.10) benötigten Winkelgeschwindigkeit $\dot{\varphi}$ muss Gleichung (5.9) differenziert werden, was auf

$$\dot{\varphi} = \left(\mathbf{u}_0 + \sum_{j=1}^{n_u} \hat{\mathbf{u}}_j \cos(\tilde{\mathbf{u}}_j q + \bar{\mathbf{u}}_j) \right) \dot{q} = \mathbf{u} \dot{q} \quad (5.15)$$

führt. Um die Trajektorie für den Motor zu bestimmen muss die nichtlineare Differentialgleichung (5.10) mit den Eingängen q , \dot{q} und $\tau_{a,E}$ gelöst werden. Die Ergebnisse sind θ und $\dot{\theta}$, mit welchen Gleichung (5.11) nach $\tau_{a,WA}$ gelöst werden kann. Auch dabei wird wieder die Gleichheit der Leistungsflussrichtung an den Flanschen ausgenutzt, so dass sich Gleichung (5.11) schreiben lässt als

$$\tau_{a,WA} = \frac{1}{\eta_A(\tau_{a,E}, \dot{\theta})} \tau_{a,E}$$

Für die Berechnung von τ_A über das Reibmodell in Gleichung (5.12) muss r_A wieder durch eine Approximation $\hat{r}_A(\dot{\theta})$ ersetzt werden. Zur Berechnung des Motormoments τ_M aus Gleichung (5.13) wird jetzt noch die Motorbeschleunigung $\ddot{\theta}$ benötigt. Diese kann bestimmt werden, indem Gleichung (5.10) einmal nach der Zeit differenziert wird. Da θ und $\dot{\theta}$ bereits bekannt sind, entsteht damit eine neue algebraische Gleichung

$$\ddot{\theta} = \ddot{\varphi} + \frac{1}{d} \left(\frac{\partial k(\varphi - \theta)}{\partial(\varphi - \theta)} (\dot{\varphi} - \dot{\theta}) - \dot{\tau}_{a,E} \right).$$

Hierin tauchen aber die neuen Größen $\ddot{\varphi}$ und $\dot{\tau}_{a,E}$ auf. Damit müssen zusätzlich die Gleichungen (5.4), (5.5), (5.14), (5.7), (5.8) und (5.15) differenziert werden, was auf weitere algebraische Gleichungen führt.

$$\dot{\tau}_G = M_{Arm} \dot{q}^{(3)} - \dot{\tau}_{Ext} \quad (5.16a)$$

$$\dot{\tau}_{a,RG} = \frac{\partial \hat{r}_G(\dot{q})}{\partial(\dot{q})} \ddot{q} + \dot{\tau}_G \quad (5.16b)$$

$$\dot{\tau}_{a,WG} = \frac{\partial}{\partial(\tau_{a,RG}, \dot{q})} \frac{1}{\eta_G(\tau_{a,RG}, \dot{q})} (\dot{\tau}_{a,RG} \dot{q} + \tau_{a,RG} \ddot{q}) \tau_{a,RG} + \frac{1}{\eta_G(\tau_{a,RG}, \dot{q})} \dot{\tau}_{a,RG} \quad (5.16c)$$

$$\dot{\mathbf{u}}(q, \dot{q}) = -\dot{q} \sum_{j=1}^{n_u} \hat{\mathbf{u}}_j \tilde{\mathbf{u}}_j \sin(\tilde{\mathbf{u}}_j q + \bar{\mathbf{u}}_j) \quad (5.16d)$$

$$\dot{\tau}_{a,E} = \frac{1}{\dot{\mathbf{u}}(q, \dot{q})} \tau_{a,WG} + \frac{1}{\mathbf{u}(q)} \dot{\tau}_{a,WG} \quad (5.16e)$$

$$\ddot{\varphi} = \dot{\mathbf{u}}(q, \dot{q}) \dot{q} + \mathbf{u}(q) \ddot{q} \quad (5.16f)$$

Unter Verwendung dieser Gleichungen kann nun das Motormoment bestimmt werden. Anhand der Herleitung werden die notwendigen Eigenschaften des Modells zur Invertierung deutlich. Um die inverse Dynamik berechnen zu können, muss insbesondere noch

- eine differenzierbare Funktion der Gelenkreibung $\hat{r}_G(\dot{q})$,
- eine Funktion der Motorreibung $\hat{r}_A(\dot{\theta})$,
- eine differenzierbare Funktion der Steifigkeit $k(\varphi_\Delta)$ und
- eine differenzierbare Funktion des inversen Wirkungsgrads im Gelenk $\widehat{\eta}_G^{-1}(\tau_{a,RG}, \dot{q})$

gefunden werden. Diese Anforderungen werden vom Modell noch nicht erfüllt. Zur Lösung des Problems werden die aufgelisteten kritischen Funktionen durch Approximationen ersetzt, was die Berechnung der inversen Dynamik bezüglich des approximierten Inversionsmodells erlaubt. Wie in Abschnitt 5.2.2 gezeigt wird, hängt die notwendige Anzahl an Differentiationen der Funktionen von der jeweiligen Roboterkinematik ab. Um universell einsetzbare Inversionsmodelle zu erhalten, sollen diese so konstruiert werden, dass die zugehörigen Funktionen aus \mathcal{C}^∞ sind.

Am einfachsten kann die Differenzierbarkeit der Steifigkeitsfunktion sichergestellt werden. Dazu muss zur Beschreibung der Steifigkeit im Modell (2.37) die Funktion $k(\varphi_\Delta)$ aus \mathcal{C}^∞ gewählt werden. Dies kann mit verschiedenen Funktionen erzielt werden. Als günstig hat sich hierbei z. B. die Funktion

$$k(\varphi_\Delta) = c_1\varphi_\Delta + c_2(\varphi_\Delta - \varphi_0)\frac{1}{1 + e^{-\delta(\varphi_\Delta - \varphi_0)}} + c_2(\varphi_\Delta + \varphi_0)\left(1 - \frac{1}{1 + e^{-\delta(\varphi_\Delta + \varphi_0)}}\right) \quad (5.17)$$

herausgestellt. Mit positiven Parameterwerten zeigt diese Funktion bei grosser Torsion die Steifigkeit c_2 und für kleine Torsion eine niedrigere Steifigkeit mit differenzierbarem Übergang.

Für eine Approximation

$$\tau_a = -\widehat{\eta}_G^{-1}(\dot{\varphi}_b, \tau_b)\tau_b$$

des inversen Wirkungsgrads im Gelenk

$$\tau_a = -\eta_G^{-1}(\dot{\varphi}_b, \tau_b)\tau_b \quad (5.18)$$

muss $\widehat{\eta}_G^{-1}(\dot{\varphi}_b, \tau_b) \in \mathcal{C}^\infty$ gelten. Zusätzlich sollte die Approximation statisch dem exakten Modell entsprechen

$$\widehat{\eta}_G^{-1}(0, \tau_b) = 1$$

und ansonsten Gleichung (5.18) möglichst gut wiedergeben. Diese Anforderungen werden z. B. von der Funktion

$$\widehat{\eta}_G^{-1}(\dot{\varphi}_b, \tau_b) = \frac{l}{\tau_b} + \frac{u - l}{\tau_b(1 + \exp^{-s\dot{\varphi}_b})} + \frac{2 - \eta_m - 1/\eta_p}{2} \exp^{-a^2\dot{\varphi}_b^2} \quad (5.19a)$$

$$u = \frac{1}{\rho} \ln \left(\exp^{\rho^1/\eta_p\tau_b} + \exp^{\rho\eta_m\tau_b} \right) \quad (5.19b)$$

$$l = -\frac{1}{\rho} \ln \left(\exp^{-\rho^1/\eta_p\tau_b} + \exp^{-\rho\eta_m\tau_b} \right) \quad (5.19c)$$

erfüllt. Die Güte der Approximation lässt sich über die zusätzlichen Parameter s , a und ρ einstellen. Hohe Werte führen zu einer besseren Approximation, aber auch zu ungünstigeren numerischen Eigenschaften, weshalb hier jeweils ein Kompromiss gefunden werden muss.

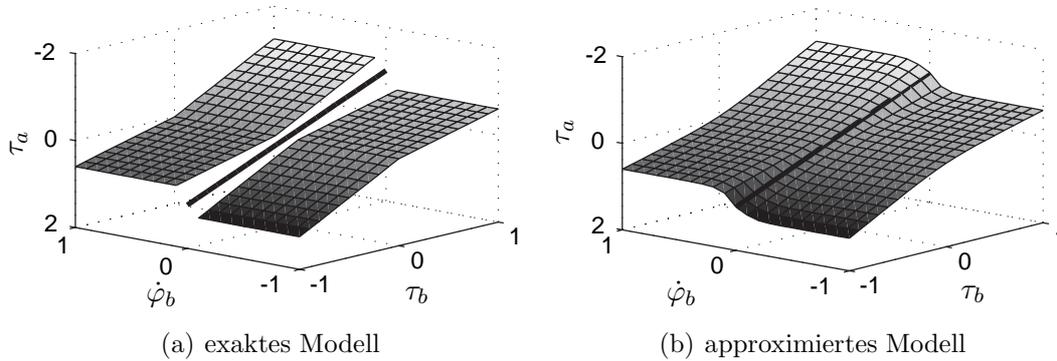


Abbildung 5.1: Vergleich der Wirkungsgradmodelle

Einen Vergleich der exakten Funktion für den Wirkungsgrad mit einer approximierten Funktion zeigt Abbildung 5.1.

Zur Berechnung der inversen Dynamik muss ferner die Funktion der Motorreibung invertierbar sein. Damit ist es jedoch nicht mehr möglich, die Haftreibung zu berücksichtigen, da dann das Reibmoment im Stillstand nicht eindeutig bestimmt werden kann. Für die Approximation kann daher nur ein Modell mit

$$\tau_a = \hat{r}_A(\dot{\varphi}_b) - \tau_b$$

verwendet werden. Für eine Approximation der Reibung bietet sich deshalb eine Kombination aus Coulomb Reibung und viskoser Reibung an wie sie in Abbildung 2.9 (b) dargestellt ist. Die Funktion der Motorreibung ist damit

$$\hat{r}_A(\dot{\varphi}_b) = \tau_C \operatorname{sgn}(\dot{\varphi}_b) + r_\nu(\dot{\varphi}_b) \quad (5.20)$$

mit einer Funktion für die viskose Reibung r_ν .

Die Funktion der Gelenkreibung muss zusätzlich noch differenzierbar sein. Damit kann auch das Coulomb-Reibmodell nicht mehr verwendet werden. Statt dessen muss im Modell der Gelenkreibung

$$\tau_a = \hat{r}_G(\dot{\varphi}_b) - \tau_b$$

die Funktion $\hat{r}_G \in \mathcal{C}^\infty$ sein. Dies kann beispielsweise durch

$$\hat{r}_G(\dot{\varphi}_b) = \frac{2\tau_C}{1 + \exp^{-s\dot{\varphi}_b}} - \tau_C + r_\nu(\dot{\varphi}_b) + \frac{\dot{\varphi}_b}{a\dot{\varphi}_b^2 + c} \quad (5.21)$$

mit den zusätzlichen Parametern s , a und $c > 0$ erreicht werden. Dabei nähern die ersten beiden Summanden die Coulomb-Reibung an, $r_\nu(\dot{\varphi}_b)$ die viskose Reibung und der letzte Summand einen Stribeck-Effekt. Ein Vergleich zwischen dem exakten Reibmodell, bei dem die Reibung mit Gleichung (2.41) modelliert wurde, und der Approximation ist in Abbildung 5.2 zu sehen. Alternativ kann die Approximation der Reibung auch durch radiale Basisfunktionen erfolgen, was in [Sch00b] gezeigt wurde. Durch die Wahl geeigneter Basisfunktionen können dabei insbesondere die notwendigen Differenzierbarkeitseigenschaften sichergestellt werden. Mit dieser Methode ist eine sehr exakte Approximation der Reibung möglich, allerdings auf Kosten eines höheren Rechenaufwands.

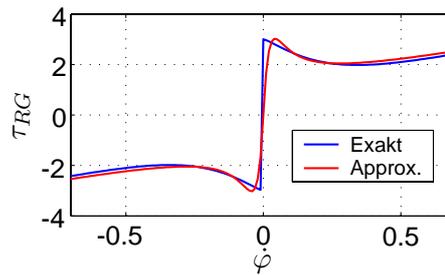


Abbildung 5.2: Vergleich des exakten und approximierten Reibmodells

Mit den approximierten Modellen ist es nun möglich, den Antriebsstrang zu invertieren und das Antriebsmoment τ_M zu berechnen. Dazu wird das Gelenkmoment bis zur ersten Ableitung benötigt, q bis zur dritten Ableitung und τ_{Ext} bis zur ersten Ableitung.

Aus numerischer Sicht ist die gezeigte Berechnung dann kritisch, wenn die Dämpfung d im Getriebe sehr klein ist, da zur Berechnung der Inversen durch d geteilt wird. Wie in [DFL05] gezeigt ist dies vor allem dann problematisch, wenn der Roboter mit Hilfe von Feedback linearisiert werden soll. Dabei entstehen durch das schlecht konditionierte Regelgesetz sehr hohe und unruhige Stellgrößen, welche entsprechend [DFL05] durch eine dynamische Erweiterung vermieden werden können. Wird die Inverse jedoch, wie in der vorliegenden Arbeit, zur Feedforward-Linearisierung verwendet, dann führen kleinere Werte von d im Wesentlichen zu einer erhöhten Sensitivität der Inversen gegenüber Fehlern bei der Modellierung der Dämpfung. Für Achsen mit extrem geringer Dämpfung empfiehlt sich eine Modellierung des Antriebs ohne Dämpfung im Gelenk, also mit $d = 0$. Im Gegensatz zu einem Antrieb mit $d \neq 0$ kann in diesem Fall aus Gleichung (5.10) nur die Motorposition θ bestimmt werden. Damit ist zur Berechnung der Inversen eine weitere Differentiation der Gleichungen (5.16) notwendig. Insgesamt wird somit bei einem Antrieb ohne Dämpfung das Gelenkmoment bis zur zweiten Ableitung benötigt, q bis zur vierten Ableitung und τ_{Ext} bis zur zweiten Ableitung. Für Roboter, bei denen das Modell des Stromreglers (2.2.2) mit berücksichtigt werden soll, kann das beschriebene Vorgehen einfach erweitert werden. Zur Berechnung des Sollstroms I_d aus dem gewünschten Motormoment τ_M wird dann zusätzlich die Ableitung von τ_M benötigt, was die Anzahl der notwendigen Differentiationen der Gleichungen um eins erhöht.

5.2.2 Mehrachsiger Roboter

Um die Berechnung der inversen Dynamik übersichtlicher zu halten, ist es sinnvoll, den Anteil der Mehrkörperdynamik und die Dynamik der Antriebsstränge getrennt zu betrachten. Am Beispiel des einachsigen Roboters wurde bereits die Berechnung des Antriebsmoments anhand des Gelenkmoments gezeigt. Diese Berechnung ist natürlich auch für einen mehrachsigen Roboter gültig. Daher wird für die Ableitung der inversen Dynamik nicht das Gesamtmodell (2.59), sondern das MKS-Modell (2.19) verwendet, welches hier umgeformt zur Berechnung der inversen Dynamik angegeben ist:

$$\tau_G = M_{Arm}(q)\ddot{q} + S(q)\ddot{\theta} + c_{Arm}(q, \dot{q}, \dot{\theta}) + g_{Arm}(q) - \tau_{Ext} \quad (5.22a)$$

$$\tau_M = S(q)^T \ddot{q} + J\ddot{\theta} + c_{Motor}(q, \dot{q}) - \tau_A \quad (5.22b)$$

Aus Gleichung (5.22a) wird deutlich, dass bei einem mehrachsigen Roboter die Getriebemomente über $S(q)\ddot{\theta}$ und $c_{Arm}(q, \dot{q}, \dot{\theta})$ von der Bewegung der Motoren abhängen. Aufgrund der Struktur der Gleichungen ist die Berechnung der inversen Dynamik dennoch in ähnlicher Form möglich wie für eine Achse. Das nötige Vorgehen wird im Folgenden gezeigt.

Die Spalten der Matrix $S(q)$ haben im allgemeinsten Fall die Form

$$(S(q)^T)_i = (S_{1,i} \quad \dots \quad S_{b_i,i} \quad 0_{1 \times N-b_i}) \quad (5.23)$$

Entsprechend Gleichung (2.7) gilt $b_i < i$, was bedeutet, dass alle von Null verschiedenen Elemente von $S(q)$ oberhalb der Matrixdiagonale liegen. Ferner können, entsprechend den Gleichungen (2.32) und (2.7), die Momente aufgrund von Coriolis- und Zentrifugalkräften höchstens die folgenden Abhängigkeiten zeigen:

$$\tau_i = c_{Arm_i}(q, \dot{q}, \dot{\theta}_{i+1}, \dots, \dot{\theta}_N) \quad (5.24)$$

Mit den Gleichungen (5.23) und (5.24) wird deutlich, dass die i -te Gleichung aus (5.22a) höchstens von der Bewegung der Motoren $i + 1$ bis N abhängig sein kann:

$$\tau_{G_i} = f \left(q, \dot{q}, \dot{\theta}_{i+1}, \dots, \dot{\theta}_N, \ddot{\theta}_{i+1}, \dots, \ddot{\theta}_N \right) \quad (5.25)$$

Dieses Ergebnis ist auch physikalisch nachvollziehbar, da sich die Motoren über den Arm an der Basis abstützen, wobei natürlich nur Gelenke betroffen sind, die zwischen Motor und Basis liegen. Damit ist die Lösungsreihenfolge zur Berechnung der inversen Dynamik von Gelenk N rückwärts nach Gelenk 1 vorgegeben.

Zur Berechnung der Motorbeschleunigung wird entsprechend Abschnitt 5.2.1 immer die Ableitung des Getriebemoments benötigt. Wenn die Getriebemomente selbst aber über die S -Matrix von der Beschleunigung der Motoren höherer Achsen abhängen, wird bei der Berechnung von $\dot{\tau}_G$ auch $\theta^{(3)}$ auftreten, was wiederum dazu führt, dass höhere Ableitungen der Bewegungsgleichung bestimmt werden müssen. Dies wird im Folgenden für den Fall einer S -Matrix, bei der alle Elemente oberhalb der Diagonalen ungleich Null sind, illustriert.

Zunächst wird τ_{G_N} mit der letzten Gleichung von (5.22a) bestimmt:

$$\tau_{G_N} = (M_{Arm}(q))_N \ddot{q} + c_{Arm_N}(q, \dot{q}) + g_{Arm_N}(q) - \tau_{Ext_N} \quad (5.26)$$

Zur Berechnung von τ_{M_N} wird, wie in Abschnitt 5.2.1 gezeigt, $\ddot{\theta}_N$ und damit auch $\dot{\tau}_{G_N}$ benötigt, wofür Gleichung (5.26) differenziert werden muss:

$$\dot{\tau}_{G_N} = (\dot{M}_{Arm}(q))_N \ddot{q} + (M_{Arm}(q))_N \dot{q}^{(3)} + \dot{c}_{Arm_N}(q, \dot{q}) + \dot{g}_{Arm_N}(q) - \dot{\tau}_{Ext_N}$$

Mit bekanntem $\ddot{\theta}_N$ kann auch das Gelenkmoment $\tau_{G_{N-1}}$ bestimmt werden, welches über die S Matrix von $\ddot{\theta}_N$ abhängt:

$$\tau_{G_{N-1}} = (M_{Arm}(q))_{N-1} \ddot{q} + S_{N-1,N} \ddot{\theta} + c_{Arm_{N-1}}(q, \dot{q}, \dot{\theta}_N) + g_{Arm_{N-1}}(q) - \tau_{Ext_{N-1}}$$

Zur Berechnung von $\tau_{M_{N-1}}$ ist es aber wiederum erforderlich, $\ddot{\theta}_{N-1}$ zu bestimmen, wofür $\dot{\tau}_{G_{N-1}}$ benötigt wird:

$$\begin{aligned} \dot{\tau}_{G_{N-1}} = & (\dot{M}_{Arm}(q))_{N-1} \ddot{q} + (M_{Arm}(q))_{N-1} \dot{q}^{(3)} + S_{N-1,N} \dot{\theta}^{(3)} \\ & + \dot{c}_{Arm_{N-1}}(q, \dot{q}, \dot{\theta}_N) + \dot{g}_{Arm_{N-1}}(q) - \dot{\tau}_{Ext_{N-1}} \end{aligned}$$

Zur Lösung dieser Gleichung wird aber $\theta_N^{(3)}$ benötigt, für dessen Berechnung Gleichung (5.26) ein weiteres Mal differenziert werden muss:

$$\begin{aligned} \ddot{\tau}_{GN} = & (\ddot{M}_{Arm}(q))_N \ddot{q} + (\dot{M}_{Arm}(q))_N \dot{q}^{(3)} + (M_{Arm}(q))_N q^{(4)} \\ & + \ddot{c}_{ArmN}(q, \dot{q}) + \ddot{g}_{ArmN}(q) - \ddot{\tau}_{ExtN} \end{aligned}$$

Das Vorgehen für die weiteren Achsen bis zur Basis ist entsprechend. Aufgrund der Struktur der S Matrix erfordert jede weitere Achse einen zusätzlichen Differentiationsschritt für die darüberliegenden Achsen. Damit wird die i -te Gleichung von (5.22a) auch i mal differenziert. Gleichzeitig werden für die Berechnung die Trajektorien von q bis zur $N + 2$ -ten Ableitung und von τ_{Ext} bis zur N -ten Ableitung benötigt.

Im Allgemeinen wird die S -Matrix jedoch nicht wie im Beispiel maximal besetzt sein, sondern weitere Nullelemente aufweisen. Dabei muss unterschieden werden zwischen strukturellen Nullelementen und Elementen, die nur lokal für bestimmte q gleich Null sind. Zur Berechnung des inversen Modells sind nur die strukturellen Eigenschaften interessant, da sie die notwendige Differentiationsordnung beeinflussen. Im Folgenden wird ein Algorithmus hergeleitet, welcher die notwendige Anzahl der Differentiationen der Bewegungsgleichungen ermittelt. Dabei ist es ausreichend die Matrix S zu untersuchen, da sie, entsprechend den Überlegungen in Abschnitt 2.1.3, die höchsten auftretenden Ableitungen von θ festlegt.

Zur Berechnung der inversen Dynamik muss entsprechend Abschnitt 5.2.1 in jedem Fall $\dot{\tau}_G$ bestimmt werden

$$\dot{\tau}_G = \dot{M}_{Arm}(q)\dot{q} + M_{Arm}(q)\dot{q}^{(3)} + \dot{S}(q)\ddot{\theta} + S(q)\theta^{(3)} + c_{Arm}(q, \dot{q}, \dot{\theta}) + g_{Arm}(q) - \dot{\tau}_{Ext}, \quad (5.27)$$

wobei der Term $S(q)\theta^{(3)}$ auftritt, für dessen Berechnung zusätzliche Ableitungen von τ_G notwendig sind. Dabei lässt sich die n -te Ableitung von τ_G immer schreiben als

$$\tau_G^{(n)} = f(q, \dots, q^{(n+2)}, \theta, \dots, \theta^{n+1}, \tau_{Ext}^{(n)}) + S(q)\theta^{(n+2)}, \quad (5.28)$$

was bedeutet, dass die höchste auftretende Ableitung von θ immer als Produkt mit $S(q)$ auftritt. Die zweite Ableitung von τ_G muss für die Achsen berechnet werden, deren zugehöriges $\theta^{(3)}$ in (5.27) auftaucht. Diese werden in der Menge M_2 zusammengefasst, welche definiert ist durch

$$M_2 = \{i | \exists S_{j,i} \neq 0, j \in \{1 \dots N\}\},$$

wobei $M_2 \subset \{1, \dots, N\}$ ist. Bei der Berechnung von $\dot{\tau}_G$ für die Elemente in M_2 kann wiederum entsprechend (5.28) $\theta^{(4)}$ auftreten. Wenn dies der Fall ist, muss eine Teilmenge der Gleichungen aus M_2 weiter differenziert werden. Diese kann, ähnlich wie zuvor, bestimmt werden durch

$$M_3 = \{i | \exists S_{j,i} \neq 0, j \in M_2\}.$$

Dieses Vorgehen wiederholt sich so lange, bis $M_n = \emptyset$. Damit lässt sich zur Bestimmung der notwendigen Ableitungen der Bewegungsgleichungen Algorithmus 5 angeben. Da der Algorithmus auf strukturellen Eigenschaften des Mehrkörpersystems beruht, kann er auch auf Roboter ohne Dämpfung in den Gelenken angewandt werden. In diesem Fall werden, entsprechend Abschnitt 5.2.1, zur Inversion der Antriebe jeweils die zweiten Ableitungen der Gelenkmomente benötigt. Damit muss in Algorithmus 5 D , beginnend bei 2, in jedem Durchlauf der Schleife um 2 statt um 1 erhöht werden. Die notwendige Differentiationsordnung

Algorithmus 5 Bestimmung der Differentiationsordnung

```

D ← 1
MD ← {1, ..., N}
while MD ≠ ∅ do
    MD+1 ← {i | ∃ Sj,i ≠ 0, j ∈ MD}
    D ← D + 1
end while

```

Roboter mit N Gelenken		
	Gleichungen	Trajektorie
ohne Dämpfung	$2N$	$2N + 2$
mit Dämpfung	N	$N + 2$

Tabelle 5.1: Obere Grenzen für die zur Berechnung der inversen Dynamik notwendigen Differentiationsordnungen

der Gleichungen verdoppelt sich damit, wenn keine Dämpfung in den Gelenken vorhanden ist. In Tabelle 5.1 sind die oberen Grenzen für die notwendigen Ableitungen der Bewegungsgleichungen und Gelenktrajektorien von Robotern mit N Gelenken angegeben. Diese Werte treten dann auf, wenn alle Elemente der S -Matrix oberhalb der Diagonalen ungleich Null sind. Der Ablauf des Algorithmus wird zur Veranschaulichung im Folgenden am Beispiel des DLR Leichtbauroboters LBR 2 gezeigt. Dabei handelt es sich um den in Abbildung 5.3 (a) dargestellten Roboter mit 7 Gelenken, dessen Kinematik in Abbildung 5.3 (b) gezeigt ist. Eine detailliertere Beschreibung kann [HASH⁺01] und [AS02] entnommen werden. Bei diesem Roboter sind die Motoren entsprechend Abbildung 5.3 (c) in den jeweiligen Gelenken montiert. Damit vereinfacht sich die Berechnung der Elemente der S -Matrix aus Gleichung (2.25) wegen $({}^{R_i}R^{G_{bi}})_3 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$ zu

$$S_{j,i} = \begin{cases} {}^{R_i}I_{R_{i,zz}} {}^{G_{bi}}R_{3,3}^{G_j} & i - 1 \geq j \\ 0 & \text{sonst} \end{cases}$$

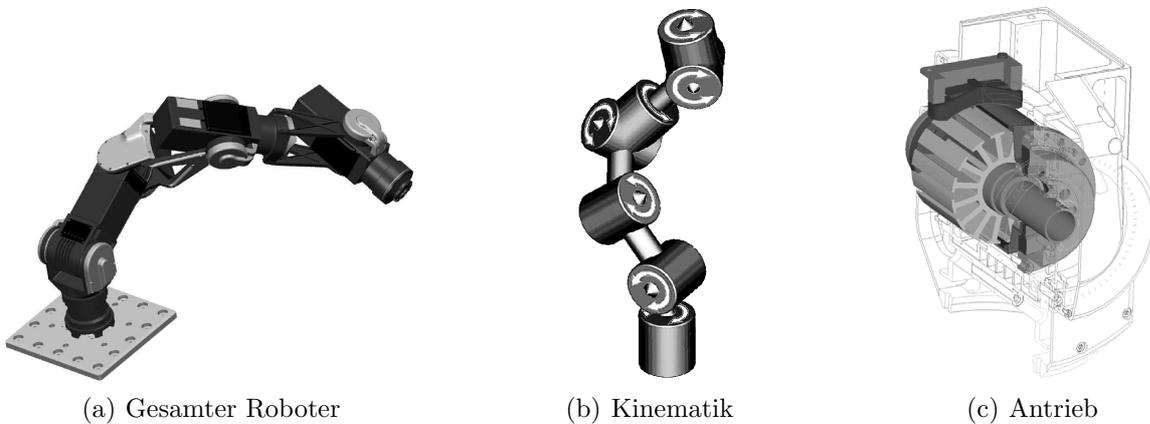


Abbildung 5.3: Aufbau des DLR LBR 2

Nachdem aber aufeinander folgende Achsen des LBR 2 immer senkrecht aufeinander stehen und damit ${}^{G_i}R_{3,3}^{G_i-1} = 0$ gilt, kann für den LBR2 eine vereinfachte S Matrix angegeben werden, bei der die Elemente auf der Nebendiagonalen ebenfalls Null sind.

$$S(q) = \begin{pmatrix} 0 & 0 & S_{1,3}(q_2) & \dots & S_{1,7}(q_2, \dots, q_6) \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & S_{5,7}(q_6) \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}$$

Dieses Ergebnis wurde auch in [HT04] durch Berechnung von S mit Hilfe der *Spatial Operator Algebra* (SOA) erreicht. Der Algorithmus startet mit

$$M_1 = \{1 \dots 7\}. \quad (5.29)$$

Entsprechend (5.29) sind die Spalten 3 bis 7 der S Matrix nicht leer, womit $\ddot{\theta}_3 \dots \ddot{\theta}_7$ benötigt werden und die zugehörigen Gleichungen weiter differenziert werden müssen.

$$M_2 = \{3 \dots 7\}$$

In den differenzierten Gleichungen tauchen, entsprechend der Zeilen 3 bis 7 der S Matrix, die Größen $\theta^{(3)}_4 \dots \theta^{(3)}_7$ auf. Der Algorithmus detektiert dies anhand der belegten Spalten der Submatrix $S_{3\dots7,1\dots7}$. Damit entsteht die Menge M_3 mit

$$M_3 = \{4 \dots 7\}$$

In der gleichen Weise führen die folgenden Schritte auf

$$M_4 = \{5 \dots 7\}$$

$$M_5 = \{6, 7\}$$

$$M_6 = \{7\}$$

Aus den Mengen M ist nun ersichtlich, wie oft die jeweiligen Gleichungen differenziert werden müssen: die 7. Gleichung 6mal, die 6. Gleichung 5mal und so weiter. Die in [HT04] angegebenen notwendigen Ableitungen wurden unter Vernachlässigung der Dämpfung bestimmt und können ebenfalls mit Algorithmus 5 bestimmt werden. Einen Vergleich zeigt Tabelle 5.2. Der Algorithmus bietet damit die Möglichkeit für beliebige RLFJ Roboter bereits durch einfache kinematische Betrachtungen anhand der S Matrix, die notwendige Differentiationsordnung der Bewegungsgleichungen zu bestimmen.

		DLR LBR 2 mit 7 Gelenken	
		Gleichungen	Trajektorie
ohne Dämpfung		12	14
mit Dämpfung		6	8

Tabelle 5.2: Zur Berechnung der inversen Dynamik notwendige Differentiationsordnungen für den DLR LBR 2

5.3 Stabilitätsbeweis für die inverse Dynamik

Die inverse Dynamik ist nur dann als Steuerung verwendbar, wenn sie stabil ist. Daher muss im Folgenden noch die Stabilität des Systems gezeigt werden.

5.3.1 Nulldynamik des Robotermodells

Bei einem linearen System ist die Stabilität der internen Dynamik durch die Lage der Nullstellen des Zählerpolynomes vorgegeben. Dieses Vorgehen ist so natürlich nicht auf nichtlineare Systeme übertragbar. Im nichtlinearen Fall wird statt dessen die Nulldynamik untersucht, also die Dynamik, die verbleibt, wenn die Ausgänge über die Eingänge auf Null gehalten werden. Dazu wird das approximierte Robotermodell zunächst für $q(t) = 0$ aufgestellt.

$$S(0)\ddot{\theta} + g_{Arm}(0) + \mathbf{U}(0) \left(k(\varphi(0) - \theta) + d(-\dot{\theta}) \right) = 0 \quad (5.30a)$$

$$J\ddot{\theta} - H_A^{-1} \left(k(\varphi(0) - \theta) + d(-\dot{\theta}) \right) + \tau_{RA} = \tau_M \quad (5.30b)$$

Die gelenkseitigen Reibungen wurden bereits aus Gleichung (5.30a) entfernt, da sie bei $\dot{q} = 0$ verschwinden. Ferner werden die Kreiselkräfte zu Null, da allein durch Bewegung der Rotoren keine Änderung der Drehachse derselben möglich ist. Gleichung (5.30b) ist eine Gleichung in θ , $\dot{\theta}$ und $\ddot{\theta}$, mit deren Hilfe die Eingänge τ_M bestimmt werden. Damit ist die Nulldynamik des Roboters durch Gleichung (5.30a) gegeben. Die Ruhelage der Nulldynamik ergibt sich mit $\dot{\theta} = 0$ und $\ddot{\theta} = 0$ aus

$$g_{Arm}(0) + \mathbf{U}(0)k(\varphi(0) - \theta) = 0 \quad (5.31)$$

zu

$$\theta = -k^{-1}(-\mathbf{U}(0)^{-1}g_{Arm}(0)) + \varphi(0) \quad (5.32)$$

Für den Stabilitätsnachweis ist die folgende Zustandstransformation sinnvoll, welche die Ruhelage in den Ursprung verschiebt.

$$\tilde{\theta} = \theta + k^{-1}(-\mathbf{U}(0)^{-1}g_{Arm}(0)) - \varphi(0) \quad (5.33a)$$

$$\dot{\tilde{\theta}} = \dot{\theta} \quad (5.33b)$$

$$\ddot{\tilde{\theta}} = \ddot{\theta} \quad (5.33c)$$

Damit ergibt sich die Nulldynamik in den neuen Koordinaten zu

$$S(0)\ddot{\tilde{\theta}} + g_{Arm}(0) + \mathbf{U}(0) \left(k(k^{-1}(-\mathbf{U}(0)^{-1}g_{Arm}(0)) - \tilde{\theta}) + d(-\dot{\tilde{\theta}}) \right) = 0 \quad (5.34)$$

Bei der Untersuchung der Stabilität dieser Gleichung kann, wie bereits beim Aufstellen der inversen Dynamik, die Struktur der Matrix $S(q)$ ausgenutzt werden. Da diese Matrix nur Einträge oberhalb der Diagonale besitzt, ist die Gleichung für die i -te Achse unabhängig von $\ddot{\tilde{\theta}}_i$. Ferner kann $\ddot{\tilde{\theta}}_i$ unter Verwendung der Gleichungen der Nulldynamik algebraisch aus $\dot{\tilde{\theta}}_i$ bestimmt werden.

$$\dot{\tilde{\theta}}_i = \frac{1}{d_i} \left(\frac{(S(0))_i \ddot{\tilde{\theta}}}{\mathbf{u}_i(0)} + \frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} + k_i \left(k_i^{-1} \left(-\frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} \right) - \tilde{\theta}_i \right) \right) \quad (5.35a)$$

$$\ddot{\tilde{\theta}}_i = \frac{1}{d_i} \frac{\partial k_i(-\tilde{\theta}_i)}{\partial \tilde{\theta}_i} \dot{\tilde{\theta}}_i \quad (5.35b)$$

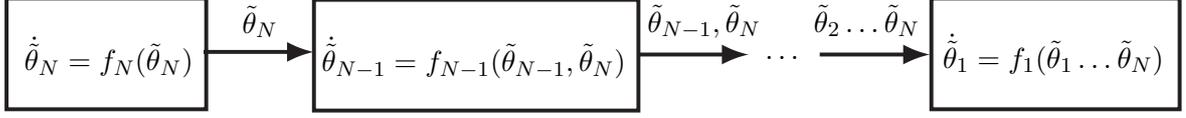


Abbildung 5.4: Darstellung der Nulldynamik als Reihenschaltung

Damit lässt sich die Nulldynamik jedoch wie in Abbildung 5.4 als eine Reihenschaltung der einzelnen Achsen darstellen, beginnend mit der N -ten Achse. Dabei wird der Term $(S(0))_i \ddot{\tilde{\theta}}$ mit Hilfe der Gleichungen (5.35) als Funktion der Motorpositionen $\tilde{\theta}_{i+1}, \dots, \tilde{\theta}_N$ ausgedrückt. Entsprechend [Isi99] garantiert die lokale asymptotische Stabilität der Funktionen $\dot{\tilde{\theta}}_i = f_i(\tilde{\theta}_i, 0)$ die lokale asymptotische Stabilität der Ruhelage $\tilde{\theta} = 0_{N \times 1}$. Damit ist es zum Nachweis der Stabilität der gesamten Nulldynamik ausreichend, die Stabilität der Nulldynamik der einzelnen Achsen zu untersuchen.

Als Kandidat für eine Lyapunov-Funktion der Nulldynamik einer Achse wird

$$V = \frac{1}{2} \tilde{\theta}_i^2 \quad (5.36)$$

gewählt. Diese Funktion ist offensichtlich positiv definit und $V \rightarrow \infty$ für $\theta \rightarrow \infty$. Die zeitliche Ableitung von V ergibt sich mit der i -ten Gleichung von (5.34) zu

$$\dot{V} = \frac{\partial V}{\partial \tilde{\theta}_i} \dot{\tilde{\theta}}_i = \tilde{\theta}_i \frac{1}{d_i} \left(\frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} + k_i \left(k_i^{-1} \left(-\frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} \right) - \tilde{\theta}_i \right) \right)$$

Entsprechend dieser Gleichung gilt $\dot{V}(0) = 0$. Damit \dot{V} negativ definit ist, muss die Funktion zudem für alle anderen Werte von $\tilde{\theta}$ negativ sein. Dies lässt sich unter Ausnutzung der strengen Monotonie der Funktion k_i zeigen:

$$\begin{aligned} \tilde{\theta}_i < 0 &\Rightarrow k_i \left(k_i^{-1} \left(-\frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} \right) - \tilde{\theta}_i \right) > -\frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} \Rightarrow \dot{V} < 0 \\ \tilde{\theta}_i > 0 &\Rightarrow k_i \left(k_i^{-1} \left(-\frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} \right) - \tilde{\theta}_i \right) < -\frac{g_{Arm_i}(0)}{\mathbf{u}_i(0)} \Rightarrow \dot{V} < 0 \end{aligned}$$

Damit ist die globale, asymptotische Stabilität der Nulldynamik für die i -te Achse gezeigt und somit auch die lokale, asymptotische Stabilität der gesamten Nulldynamik. Um globale asymptotische Stabilität der Nulldynamik sicherzustellen, muss zusätzlich gezeigt werden, dass die Lösung des Gesamtsystems für alle $t \geq 0$ definiert und beschränkt ist. Diese Bedingung ist für die einzelnen Achsen aufgrund deren globaler, asymptotischer Stabilität bereits erfüllt. Der einzige zusätzliche Term, welcher die Achsen verkoppelt, ist $S(0)\ddot{\tilde{\theta}}$. Damit ist es auch schon ausreichend zu zeigen, dass dieser Term für alle $t \geq 0$ definiert und beschränkt ist, da eine beschränkte Störung aufgrund der Stabilität der einzelnen Achsen auch nur einen beschränkten Zustand erzeugen wird. Mit Gleichung (5.35b) erkennt man, dass $\ddot{\tilde{\theta}}$ für differenzierbare Steifigkeitskennlinien mit beschränkter Steigung definiert und beschränkt ist. Ferner sind die Elemente von S entsprechend Gleichung (2.25) dann beschränkt, wenn die Rotorträgheiten ${}^{R_i}I_{R_i z z}$ beschränkt sind. Diese Bedingungen sind in der Praxis immer erfüllt. Damit ist $S(0)\ddot{\tilde{\theta}}$ für alle $t \geq 0$ definiert und beschränkt und die Nulldynamik global asymptotisch stabil.

5.3.2 Input-to-State Stability der internen Dynamik

Die gezeigte Stabilität der Nulldynamik garantiert jedoch nur die lokale Stabilität der internen Dynamik, da das System nur für $q(t) = 0$ betrachtet wurde. Damit wird insbesondere der Anteil des Mehrkörpersystems Arm, welches einen Großteil der Nichtlinearitäten verursacht, nicht berücksichtigt. Es ist deshalb zusätzlich notwendig die Stabilität der nicht-autonomen internen Dynamik, deren Eingänge die Solltrajektorien sind, zu untersuchen. Natürlich kann in diesem Fall nicht mehr erwartet werden, dass der Zustand gegen eine bestimmte Ruhelage konvergiert. Vielmehr geht es darum, sicherzustellen, dass der Zustand bei beschränktem Eingang ebenfalls beschränkt bleibt. Diese Aufgabenstellung kann mit Hilfe der Theorie der Input-to-State Stability (ISS) von E. D. Sontag behandelt werden, welche in Anhang A.3 kurz beschrieben ist.

Zum Nachweis der ISS ist es sinnvoll, zunächst die folgende Zustandstransformation durchzuführen.

$$\begin{pmatrix} q \\ \delta \end{pmatrix} = \begin{pmatrix} q \\ \varphi(q) - \theta \end{pmatrix}$$

Damit lässt sich das Modell des Roboters als

$$\begin{pmatrix} M_{Arm}(q) & S(q) \\ S(q)^T & J \end{pmatrix} \begin{pmatrix} \ddot{q} \\ \ddot{\varphi}(q) - \ddot{\delta} \end{pmatrix} + \begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\varphi}(q) - \dot{\delta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} + \begin{pmatrix} H_G \mathbf{U} \left(k(\delta) + d(\dot{\delta}) \right) + \tau_{RG} \\ H_A^{-1} \left(k(\delta) + d(\dot{\delta}) \right) + \tau_{RA} \end{pmatrix} = \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix} \quad (5.37)$$

darstellen. Wie aus der Analyse der Bewegungsgleichungen in Abschnitt 5.2 ersichtlich ist, entsprechen die in der internen Dynamik auftretenden Differentialgleichungen den ersten N Bewegungsgleichungen. Diese verbleiben somit für die Untersuchung der Stabilität der internen Dynamik:

$$\begin{aligned} \dot{\delta} = d^{-1} \left(-k(\delta) + \mathbf{U}^{-1} H_G^{-1} \left(\tau_{Ext} - M_{Arm}(q) \ddot{q} - S(q) (\ddot{\varphi}(q) - \ddot{\delta}) \right. \right. \\ \left. \left. - c_{Arm}(q, \dot{q}, \dot{\varphi}(q) - \dot{\delta}) - g_{Arm}(q) - \tau_{RG} \right) \right) \end{aligned} \quad (5.38)$$

Für den Beweis wird zusätzlich die folgende Annahme getroffen.

Annahme 5.1. Für den Roboter gelte $g_{Arm}(0) = 0$. Diese Eigenschaft kann, ohne Beschränkung der Allgemeinheit, durch eine geeignete Wahl der Nullpunkte der Gelenkwinkel sichergestellt werden.

Modelle, welche die obige Annahme nicht erfüllen, können über eine einfache Koordinatentransformation auf die benötigte Form gebracht werden.

In Anhang A.3 wird gezeigt, dass eine Reihenschaltung aus ISS Systemen mit Eingängen ebenfalls stabil ist. Entsprechend (5.38) kann die interne Dynamik in dieser Form dargestellt werden, wobei jede Achse eine ISS interne Dynamik bezüglich ihrer Eingänge besitzen muss. Damit ist es ausreichend, die ISS der internen Dynamik der einzelnen Achsen nachzuweisen.

Die interne Dynamik einer Achse kann dargestellt werden als

$$\dot{\delta}_i = -\frac{1}{d_i} k_i(\delta_i) + g_i(u)$$

mit

$$\begin{aligned} u &= (q, \dot{q}, \ddot{q}, \dot{\delta}_{i+1} \dots \dot{\delta}_N, \ddot{\delta}_{i+1} \dots \ddot{\delta}_N, \tau_{Exti}) \\ g_i(u) &= \frac{1}{d_i \mathbf{u}_i \eta_i} \left(\tau_{Exti} - (M_{Arm})_i(q) \ddot{q} - (S)_i(q) (\ddot{\varphi}(q) - \ddot{\delta}) - \right. \\ &\quad \left. c_{Arm_i}(q, \dot{q}, \dot{\varphi}(q) - \dot{\delta}) - g_{Arm_i}(q) - \tau_{RG_i} \right) \end{aligned}$$

Dabei werden in $g_i(u)$ die aufgrund der vorgegebenen Bewegung des Armes auf das Gelenk i entstehenden Rückwirkungen gesammelt. Zugunsten einer übersichtlicheren Darstellung werden die Indizes für die Achsen im Folgenden weggelassen. Als Kandidat für eine ISS Lyapunov-Funktion wird die Stammfunktion der Federkennlinie verwendet.

$$V(\delta) = K(\delta) \quad (5.39)$$

Diese ist aufgrund der Definition von k positiv definit und $V(0) = 0$. Gilt ferner, dass k eine Funktion aus \mathcal{K}_∞ ist, dann ist auch K und damit V eine Funktion aus \mathcal{K}_∞ . Damit ist die erste Bedingung für eine ISS-Lyapunov-Funktion, die Existenz von \mathcal{K}_∞ Funktionen $\underline{\alpha}(\cdot)$ und $\bar{\alpha}(\cdot)$, so dass

$$\underline{\alpha}(\|\delta\|) \leq V(\delta) \leq \bar{\alpha}(\|\delta\|) \quad \text{für alle } \delta \in \mathbb{R}$$

gilt, z. B. mit

$$\begin{aligned} \underline{\alpha}(\|\delta\|) &= \frac{1}{2} K(|\delta|) \\ \bar{\alpha}(\|\delta\|) &= 2K(|\delta|) \end{aligned}$$

trivial erfüllt. Die Ableitung von V ergibt sich zu

$$\dot{V} = k(\delta) \left(-\frac{1}{d} k(\delta) + g(u) \right).$$

Gesucht sind nun eine \mathcal{K}_∞ Funktion $\alpha(\cdot)$ und eine \mathcal{K} Funktion $\chi(\cdot)$, so dass

$$\|\delta\| \geq \chi(\|u\|) \Rightarrow \dot{V} \leq -\alpha(\|\delta\|) \quad \text{für alle } \delta \in \mathbb{R}$$

Dazu kann \dot{V} wie folgt abgeschätzt werden:

$$\dot{V} \leq -|k^2(\delta)| + |k(\delta)| |g(u)|$$

Besitzt des Weiteren k eine Obergrenze der Steigung $s_o > 0$ und eine Untergrenze $s_u > 0$, dann kann mit

$$\begin{aligned} |k(\delta)| &\geq s_u |\delta| \\ |k(\delta)| &\leq s_o |\delta| \end{aligned}$$

\dot{V} auch angegeben werden als

$$\dot{V} \leq -s_u^2 |\delta|^2 + s_o |\delta| |g(u)| = |\delta| (-s_u^2 |\delta| + s_o |g(u)|)$$

Wird nun für $\alpha(\cdot)$ die Funktion

$$\alpha(|\delta|) = \epsilon |\delta|^2$$

angesetzt, dann muss gelten

$$\dot{V} \leq |\delta| (-s_u^2 |\delta| + s_o |g(u)|) \leq -\epsilon |\delta|^2,$$

was für

$$(s_u^2 - \epsilon) |\delta| \geq s_o |g(u)|$$

der Fall ist. Wird nun, was immer möglich ist, $\epsilon < s_u^2$ gewählt, dann ergibt sich

$$|\delta| \geq \frac{s_o |g(u)|}{s_u^2 - \epsilon}.$$

Um zur benötigten Form $|\delta| \geq \chi(\|u\|)$ zu gelangen, wird noch eine Abschätzung von $|g(u)|$ mittels einer \mathcal{K} Funktion benötigt. Da entsprechend den Annahmen zu Beginn $|g(0)| = 0$ gilt, ist eine Abschätzung der Form

$$|g(u)| \leq \psi \|u\| \tag{5.40}$$

mit einem geeigneten Faktor ψ dann möglich, wenn die partiellen Ableitungen von $g(u)$ für alle u beschränkt sind. Dies ist für den Term

$$\tau_{MKS} = -(M_{Arm})_i(q)\ddot{q} - (S)_i(q)(\ddot{\varphi}(q) - \ddot{\delta}) - c_{Arm_i}(q, \dot{q}, \dot{\varphi}(q) - \dot{\delta}) - g_{Arm_i}(q),$$

welcher das Moment aufgrund der Bewegung des Mehrkörpersystems beschreibt, automatisch erfüllt, wenn Masse und Größe des MKS beschränkt sind. Damit verbleibt die approximierte Gelenkreibung, deren Ableitungen per Konstruktion der Funktion ebenfalls beschränkt sind. Es ist daher stets möglich, ψ so zu wählen, dass Gleichung (5.40) erfüllt ist. Damit ist die \mathcal{K} Funktion

$$\chi(\|u\|) = \frac{s_o \psi \|u\|}{s_u^2 - \epsilon}$$

gefunden. Abgesehen von praktisch nicht relevanten Sonderfällen, wie z. B. unendlicher Masse und $d = 0$, ist die Funktion (5.39) eine ISS-Lyapunov-Funktion für die interne Dynamik der Achse und diese somit ISS, wenn die Funktion der Federkraft k zur Klasse \mathcal{K}_∞ gehört. Damit ist die interne Dynamik des gesamten Roboters ISS, wenn die Funktionen der Federkräfte für alle Achsen zu \mathcal{K}_∞ gehören. Anschaulich bedeutet diese Bedingung, dass die Funktionen $k_i(\delta)$ streng monoton steigend sein müssen, durch den Ursprung verlaufen und für $\delta \rightarrow \infty$ gegen ∞ gehen müssen. Diese Vorgaben werden von üblichen Federkennlinien automatisch erfüllt. Lediglich eine Berücksichtigung von Lose ist nicht möglich, da in diesem Fall k_i nicht mehr streng monoton steigt.

5.4 Inverse Dynamik des KR15/2

Wie Algorithmus 5 in Abschnitt 5.2.2 zeigt, hängt die Anzahl der notwendigen Differentiationen der Bewegungsgleichungen und damit auch die benötigte Rechenleistung von der Kinematik des Roboters und insbesondere von dessen S -Matrix ab. Für den Extremfall eines Roboters mit N parallelen Gelenken deren Motoren koaxial an den Gelenken angebracht sind, müssen Teile der Bewegungsgleichungen N mal differenziert werden. Zur Bestimmung der Differentiationsordnung für den KR15/2 wird daher zunächst dessen S -Matrix bestimmt und Algorithmus 5 angewandt. Aus Abbildung 2.12 und Abbildung 2.13 erkennt man, dass

die Motoren der Gelenke 4 bis 6 an Glied 3 befestigt sind. Damit ist eine Kopplung dieser Motoren über die S -Matrix nur noch für die Gelenke 1 bis 3 möglich. Zudem sind die Drehachsen der Motoren 4 bis 6 immer orthogonal zu den Gelenken 2 und 3. Damit ist die Kopplung zwischen den Motoren 4 bis 6 und den Gelenken nur für das erste Gelenk ungleich Null. Die Achsen der Motoren 2 und 3 sind immer orthogonal zum ersten Gelenk, wodurch sie nicht mit diesem gekoppelt sein können. Dagegen wirkt der Motor des 3. Gelenkes auch Gelenk zwei zurück, da in diesem Fall die Drehachsen von Motor und Gelenk stets parallel sind. Durch Anwendung von Gleichung (2.25) kann die S -Matrix des KR15/2 deshalb zu

$$S = \begin{pmatrix} 0 & 0 & 0 & {}^{R_4}I_{R4zz} \sin(q_2 + q_3) & -{}^{R_5}I_{R5zz} \sin(q_2 + q_3) & -{}^{R_6}I_{R6zz} \sin(q_2 + q_3) \\ 0 & 0 & {}^{R_3}I_{R3zz} & 0 & 0 & 0 \\ & & & 0_{4 \times 6} & & \end{pmatrix} \quad (5.41)$$

bestimmt werden. Die Anwendung von Algorithmus 5 auf diese Matrix führt auf

$$\begin{aligned} M_1 &= \{1, \dots, 6\} \\ M_2 &= \{3, \dots, 6\}. \end{aligned}$$

Danach bricht der Algorithmus ab, was bedeutet, dass für die ersten beiden Gelenke lediglich die erste Ableitung der Bewegungsgleichungen und für die Gelenke 3 bis 6 die zweite Ableitung der Bewegungsgleichungen benötigt wird. Damit sind die Anforderungen an die Rechenleistung noch so gering, dass auf eine Vernachlässigung der Kopplungen wie im Spong-Modell verzichtet werden kann.

Zur Berechnung der inversen Dynamik des KR15/2 wird die in Kapitel 4.6 dargestellte Methode der Indexreduktion für DAE Systeme eingesetzt. Der C-Code zur Berechnung kann, wie in Abschnitt 4.6.3 gezeigt, nach Darstellung des Systems als Modelica-Modell automatisch durch das Programm Dymola erzeugt werden.

Wie in Abschnitt 5.2.1 festgestellt, ist das in Kapitel 2 entwickelte Modell nicht direkt invertierbar. Daher wurden in Abschnitt 5.2.1 für einige Teile des Modells Approximationen entwickelt, welche zunächst in entsprechende Modelica Modelle umgesetzt werden müssen. Da die Modelle der Drehungleichförmigkeiten die gemessenen Werte nur qualitativ richtig wiedergeben konnten, werden sie bei der Invertierung nicht berücksichtigt. Statt dessen wird nur die Basisübersetzung benutzt. Mit den approximierten Modellen wurde ein neues Getriebemodell aufgebaut, welches in Abbildung 5.5 dargestellt ist. Unter Verwendung des approximierten Getriebemodells kann, äquivalent zu Abbildung 2.13 ein neues, approximiertes Modell des Roboters aufgebaut werden. Dieses wiederum kann durch Vertauschung der Ein- und Ausgänge zur Berechnung der inversen Dynamik benutzt werden. Das komplette Modelica-Modell der inversen Dynamik des KR15/2 ist in Tabelle 5.3 dargestellt. Zum Test des Modells der inversen Dynamik wird für die Gelenkwinkel aller Achsen ein Sprung vom Gelenkwinkel 0 rad auf 0.2 rad vorgegeben, welcher noch mit einem Filter 4. Ordnung gefiltert wird. Damit können die notwendigen Ableitungen der Eingangsgröße berechnet werden. Das entsprechende Modelica-Modell zeigt Abbildung 5.6. Das Modell zur Berechnung der inversen Dynamik besteht aus etwa 800 nicht triviale Gleichungen. Erwartungsgemäß hat es 10 Zustände: die 6 Motorpositionen und 4 Zustände im Filter. Bei Integration mit dem Runge-Kutta Verfahren 4. Ordnung und einer Schrittweite von einer Millisekunde wird für eine Trajektorie von einer Sekunde auf einem Pentium 1.7 GHz etwa 0.1 s Rechenzeit benötigt. Ähnliche Werte werden auch bei variabler Schrittweite mit dem Integrator Dassl erreicht. Damit ist die Berechnung auf aktuellen PC-Prozessoren problemlos lauffähig.

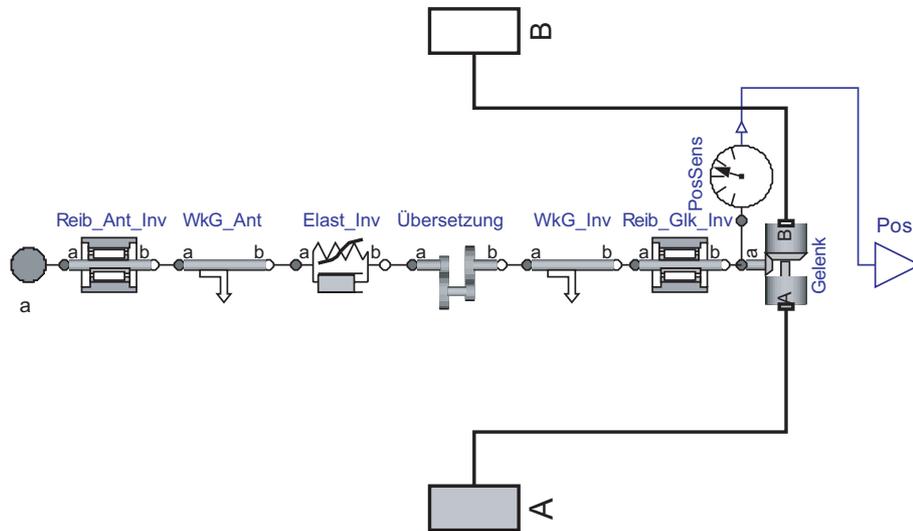


Abbildung 5.5: Objektdiagramm des invertierbaren Getriebemodells mit approximierten Teilmodellen

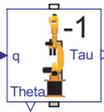
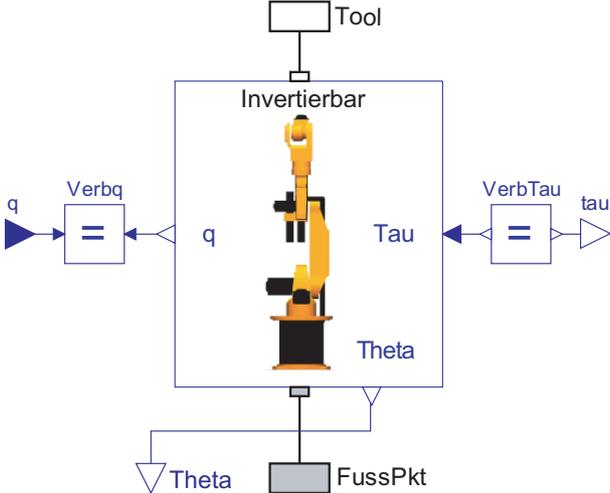
Symbol	Beschreibung des Teilmodells
	<p>Inverse Dynamik Modell der inversen Dynamik des KR15/2 auf Basis des approximierten Robotermodells.</p> 

Tabelle 5.3: Modelica Modell zur Berechnung der inversen Dynamik des KR15/2

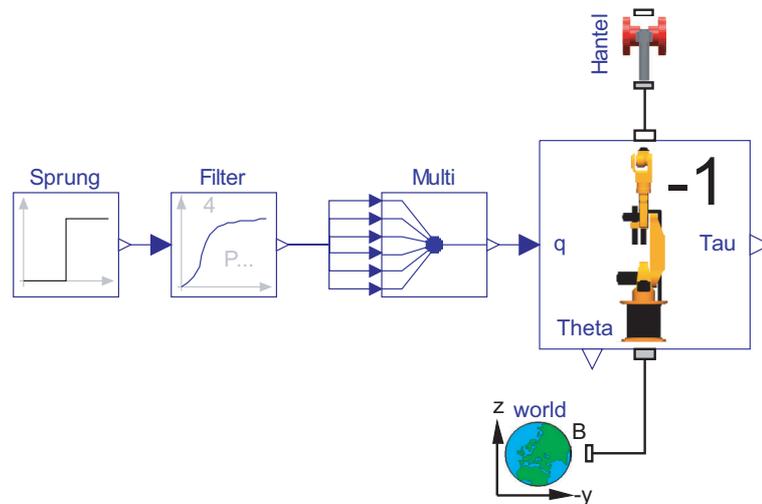
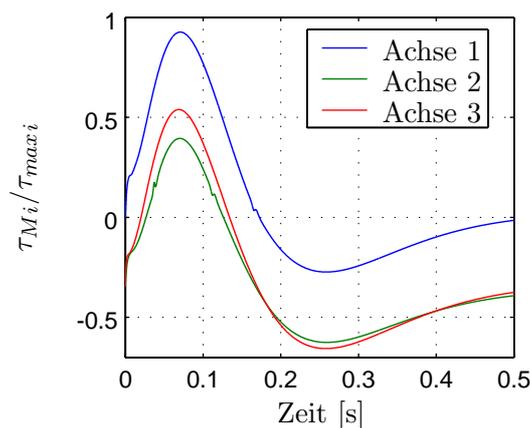
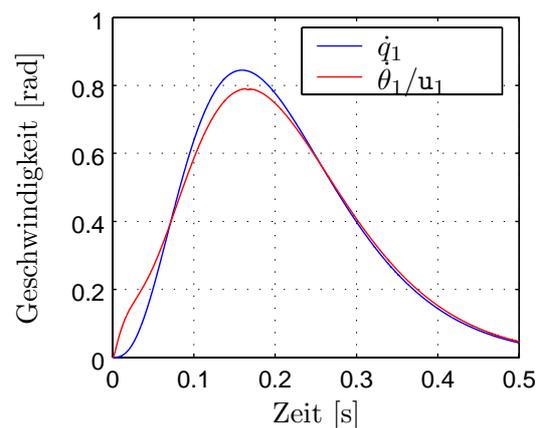


Abbildung 5.6: Objektdiagramm des Modells zum Test der Inversen Dynamik des KR15/2



(a) Motormomente der Achsen 1 bis 3



(b) Gelenk- und Motorgeschwindigkeit der ersten Achse

Abbildung 5.7: Mit Hilfe der inversen Dynamik des KR15/2 berechnete Sollgrößen

Zur Illustration zeigt Abbildung 5.7 Ergebnisse aus der Berechnung der inversen Dynamik für den gefilterten 0.2rad Sprung der Gelenkwinkel. Dabei sind in Abbildung 5.7 (a) die Motormomente der ersten drei Achsen bezogen auf die maximalen Motormomente und in Abbildung 5.7 (b) ein Vergleich zwischen der vorgegebenen Geschwindigkeit des ersten Gelenkes und der resultierenden, auf die Gelenkseite umgerechneten Motorgeschwindigkeit dargestellt.

Kapitel 6

Regler- und Beobachterentwurf

In diesem Kapitel wird der Entwurf des Reglers mit Beobachter beschrieben. Zunächst erfolgt eine direkte Kompensation der motorseitigen Lagerreibung. Danach wird unter Verwendung von Beschleunigungssensoren am Arm ein Beobachter entwickelt, bezüglich seiner Eigenschaften untersucht und für die ersten drei Achsen des KR15/2 ausgelegt. Anschließend wird der Regler mit der Forderung nach guter Eigendynamik und stationärer Genauigkeit anhand von linearen Achsmodellen entworfen, das benutzte Gain-Scheduling motiviert und experimentell am Roboter getestet.

6.1 Kompensation der motorseitigen Lagerreibung

Häufig ist es vorteilhaft nichtlineare Effekte in der Strecke direkt durch Aufschaltung einer geeigneten Stellgröße zu eliminieren. Dieses Vorgehen ist insbesondere dann geeignet, wenn die Nichtlinearität und die Stellgröße den gleichen Angriffspunkt haben. Betrachtet man das Robotermodell nach Gleichung (2.59) oder das Antriebsstrangmodell nach Gleichung (2.50) so fällt auf, dass die motorseitige Lagerreibung genauso in das Modell eingehen wie das Stellmoment. Damit bietet sich eine direkte Kompensation dieser Nichtlinearität auf der Basis von Messwerten an. Zur Kompensation kann dabei die bereits in Abschnitt 5.2.1 zur Inversion des Modells aufgestellte Funktion der Lagerreibung (5.20) verwendet werden. Damit ergibt sich für die Reibungskompensation der Zusammenhang

$$\tilde{u} = u + \tau_C \operatorname{sgn}(\dot{\theta}) + r_\nu(\dot{\theta}) \quad (6.1)$$

wobei $\dot{\theta}$ durch numerisches Differenzieren von θ berechnet wird. Alternativ können auch andere Methoden zur Kompensation der Reibung wie z. B. lernende Reibmodelle [Sch00b] benutzt werden. Problematisch an einer Reibungskompensation nach Gleichung (6.1) ist die Unstetigkeit bei $\dot{\theta} = 0$. Bei einer großen Coulomb-Reibung τ_C und einer durch das numerische Differenzieren verbrauchten Messungen von $\dot{\theta}$ kann das Stellmoment \tilde{u} im Stillstand ebenfalls ein störendes Rauschen aufweisen. Dieses kann z. B. reduziert werden, indem wie in Abschnitt 5.2.1 zur Kompensation ein approximiertes Reibmodell benutzt wird:

$$\tilde{u} = u + \frac{2\tau_C}{1 + \exp^{-s\dot{\theta}}} - \tau_C + r_\nu(\dot{\theta}) \quad (6.2)$$

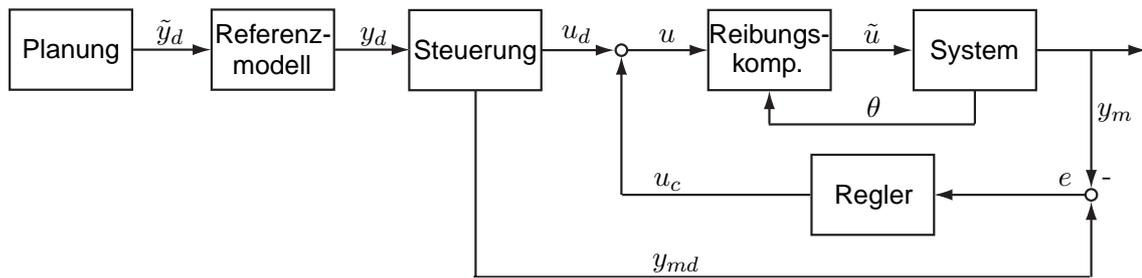


Abbildung 6.1: Regelungsstruktur mit Reibungskompensation

Für den experimentell untersuchten KR15/2 hat sich eine Kompensation nach Gleichung (6.2) als geeignet herausgestellt. Eine um die Kompensation der Lagerreibung erweiterte Regelungsstruktur zeigt Abbildung 6.1. Durch die Kompensation kann der Motor im Folgenden als reibungsfrei betrachtet werden. Damit ist die Berücksichtigung der Motorreibung in der Steuerung auf Basis der Sollgrößen nicht mehr notwendig, da die Reibungsverluste jetzt stattdessen mit Hilfe von Messwerten berechnet werden.

6.2 Beobachter mit Beschleunigungsmessung

Zur korrekten Schätzung der Zustände ist die Berücksichtigung der nichtlinearen Roboterdynamik und der Kopplungen zwischen den Achsen notwendig. Damit muss insbesondere die Mehrkörperdynamik des Roboters im schnellen Takt der Regelung ausgewertet werden. Dies stellt jedoch beachtliche Anforderungen an die verfügbare Rechenleistung und ist aus praktischer Sicht nicht wünschenswert. In diesem Kapitel wird gezeigt, dass die Komplexität des notwendigen Modells durch den Einsatz von Beschleunigungssensoren am Arm deutlich reduziert werden kann. Die Verwendung der Beschleunigungssensoren führt dabei auf weitestgehend entkoppelte Modelle der einzelnen Achsen im Beobachter, in denen die Mehrkörperdynamik nicht mehr auftritt. Dies erlaubt eine einfachere Struktur des Beobachters und vereinfacht die Untersuchung der Stabilitätseigenschaften.

6.2.1 Kartesische Beschleunigungsmessung

Beschleunigungssensoren messen die kartesische Beschleunigung eines Punktes im Raum. Werden die Sensoren am Arm des Roboters angebracht, hängt deren Messung über die Kinematik des Roboters von der Bewegung der Gelenke ab. Dieser Zusammenhang wird zunächst hergeleitet. Die zugehörigen Koordinatensysteme und Vektoren sind in Abbildung 6.2 dargestellt. Die Bewegung eines Punktes r_S auf dem Arm in Weltkoordinaten wird durch die Kinematik des Roboters beschrieben. Entsprechend berechnet sich die Beschleunigung dieses Punktes in Weltkoordinaten zu

$${}^W r_S = f_r(q) \quad (6.3a)$$

$${}^W \dot{r}_S = \frac{\partial f_r(q)}{\partial q} \dot{q} = J_{f_r}(q) \dot{q} \quad (6.3b)$$

$${}^W \ddot{r}_S = \dot{J}_{f_r}(q) \dot{q} + J_{f_r}(q) \ddot{q} \quad (6.3c)$$

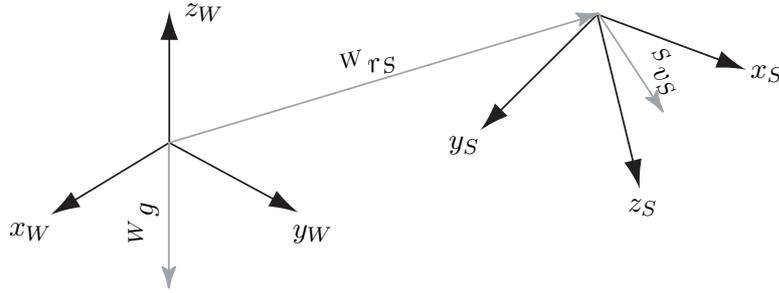


Abbildung 6.2: Darstellung von Welt- und Sensorkoordinatensystem

wobei J_{f_r} die Jacobimatrix der Funktion f_r ist. Ein am Arm angebrachter Beschleunigungssensor wird mitbewegt, das heißt, er misst nicht in Weltkoordinaten, sondern in seinem eigenen Sensorkoordinatensystem. Eine Besonderheit der Beschleunigungssensoren ist dabei, dass auch die Erdbeschleunigung mitgemessen wird. Damit ergibt sich mit Gleichung (6.3c) die Beschleunigung des Punktes r_S in Sensorkoordinaten zu

$$\begin{aligned} {}^S\ddot{r}_S &= {}^S R^W ({}^W\ddot{r}_S + {}^W g) \\ &= {}^S R^W J_{f_r}(q)\ddot{q} + {}^S R^W (\dot{J}_{f_r}(q)\dot{q} + {}^W g) \end{aligned} \quad (6.4)$$

Dabei muss an diesem Punkt nicht zwingend die Beschleunigung in allen Raumrichtungen gemessen werden. Mit dem Einheitsvektor der Messrichtung des Sensors ${}^S v_S$ im Sensorsystem ist der Messwert eines Sensors aus Gleichung (6.4):

$${}^S\ddot{r}_S = {}^S v_S {}^S R^W J_{f_r}(q)\ddot{q} + {}^S v_S {}^S R^W (\dot{J}_{f_r}(q)\dot{q} + {}^W g)$$

Werden nun m Sensoren am Arm verteilt, dann ergeben sich deren Messwerte zu

$$\begin{aligned} \ddot{r}_G &= \begin{pmatrix} {}^{S_1} v_{S_1} {}^{S_1} R^W J_{f_{r_1}}(q) \\ {}^{S_2} v_{S_2} {}^{S_2} R^W J_{f_{r_2}}(q) \\ \vdots \\ {}^{S_m} v_{S_m} {}^{S_m} R^W J_{f_{r_m}}(q) \end{pmatrix} \ddot{q} + \begin{pmatrix} {}^{S_1} v_{S_1} {}^{S_1} R^W (\dot{J}_{f_{r_1}}(q)\dot{q} + {}^W g) \\ {}^{S_2} v_{S_2} {}^{S_2} R^W (\dot{J}_{f_{r_2}}(q)\dot{q} + {}^W g) \\ \vdots \\ {}^{S_m} v_{S_m} {}^{S_m} R^W (\dot{J}_{f_{r_m}}(q)\dot{q} + {}^W g) \end{pmatrix} \\ &= {}^S T^Q(q)\ddot{q} + {}^S W(q, \dot{q}) \end{aligned} \quad (6.5)$$

Dabei bezeichnet ${}^S T^Q(q)$ die Matrix mit der die Beschleunigungen der Gelenke auf die Sensoren transformiert werden und ${}^S W(q, \dot{q})$ den Vektor der gemessenen Beschleunigungen aufgrund der von q und \dot{q} abhängigen Bewegung im Raum.

6.2.2 Robotersystem mit Beschleunigungssensoren

Mit den Messungen für die Beschleunigungen \ddot{r}_G und für die Motorpositionen θ ergibt sich das zu beobachtende System in Zustandsform aus den Gleichungen (5.1) und (6.5) als

$$\begin{aligned} \ddot{q} = & - (M_G^{-1} \quad -M_G^{-1} S J^{-1}) \left(\begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} \right) \\ & + \begin{pmatrix} H_G U \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) + \tau_{RG} \\ -H_A^{-1} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) \end{pmatrix} - \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix} \end{pmatrix} \quad (6.6a)$$

$$\begin{aligned} \ddot{\theta} = & - (-J^{-1} S^T M_G^{-1} \quad J^{-1} + J^{-1} S^T M_G^{-1} S J^{-1}) \left(\begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} \right) \\ & + \begin{pmatrix} H_G U \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) + \tau_{RG} \\ -H_A^{-1} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) \end{pmatrix} - \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix} \end{pmatrix} \quad (6.6b)$$

$$y = \begin{pmatrix} {}^{STQ}(q)\ddot{q} + {}^S W(q, \dot{q}) \\ \theta \end{pmatrix} \quad (6.6c)$$

Um die Anschaulichkeit zu erhöhen werden die physikalischen Bezeichnungen der Koordinaten q und θ und der Eingänge τ_{Ext} und τ_M vorerst noch beibehalten.

Es stellt sich nun die Frage, auf welche Art die Ausgänge y für den Beobachter genutzt werden können. Dabei ist insbesondere die Beobachtbarkeit des Roboters von den einzelnen Ausgängen aus interessant. Durch alleinige Verwendung der Beschleunigungsmessung sind die Zustände des Roboters im Allgemeinen nicht beobachtbar. Als einfachstes Beispiel kann zur Veranschaulichung ein Zwei-Massen-System verwendet werden. Die Determinante von dessen Beobachtbarkeitsmatrix Q_B ergibt sich mit

$$\begin{aligned} A &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{J_a} & -\frac{d}{J_a} & \frac{k}{J_a} & \frac{d}{J_a} \\ 0 & 0 & 0 & 1 \\ \frac{k}{J_m} & \frac{d}{J_m} & -\frac{k}{J_m} & -\frac{d}{J_m} \end{pmatrix} \\ c &= \begin{pmatrix} -\frac{k}{J_a} & -\frac{d}{J_a} & \frac{k}{J_a} & \frac{d}{J_a} \end{pmatrix} \\ Q_B &= \begin{pmatrix} c \\ cA \\ cA^2 \\ cA^3 \end{pmatrix} \end{aligned}$$

unabhängig von den Parametern des Zwei-Massen-Systems zu

$$\det Q_B = 0.$$

Dieses Ergebnis kann auch anschaulich nachvollzogen werden: Die gemessene Beschleunigung ist nur von der Differenz der Zustände von Motor und Arm abhängig, nicht aber von ihren absoluten Werten. Damit kann aber z. B. eine fehlerhafte Initialisierung der Positionen nicht erkannt und ausgeglichen werden.

Die Beobachtbarkeit des Roboters muss daher durch Messung der Motorposition sichergestellt werden. Die Messungen der Beschleunigungssensoren können aber statt dessen zur Beseitigung von Achskopplungen im Modell eingesetzt werden. Ein entsprechendes Vorgehen wird im Folgenden gezeigt.

6.2.3 Entkopplung durch Beschleunigungsmessungen

An Gleichungssystem (6.6) erkennt man, dass die Terme der Mehrkörperdynamik vor allem in Gleichung (6.6a) zur Berechnung von \ddot{q} benutzt werden. Entsprechend Gleichung (6.6c) ist \ddot{q} jedoch auch in den Messungen der Beschleunigungssensoren enthalten. Die Idee ist nun, die Berechnung der Gelenkbeschleunigungen auf Basis der Mehrkörperdynamik in Gleichung (6.6a) durch eine Berechnung auf Basis der gemessenen kartesischen Beschleunigungen zu ersetzen. Dazu wird zunächst die Transformation von ${}^S r_G$ auf \ddot{q} untersucht.

Für die Berechnung der Gelenkbeschleunigungen \ddot{q} stellt Gleichung (6.5) ein konsistentes System von m Gleichungen in N Unbekannten dar. Um dieses System lösen zu können werden somit mindestens $m = N$ Sensoren benötigt. Zudem muss durch geeignete Wahl der Messpunkte sichergestellt werden, dass das System immer eine eindeutige Lösung nach \ddot{q} besitzt. Dies sollte bei Verwendung von genügend vielen Sensoren immer möglich sein, wobei in der Regel jedoch $m > N$ Sensoren benötigt werden. In diesem Fall müssen die Sensoren so angebracht sein dass ${}^S T^Q$ injektiv ist, womit gilt:

$$\text{Rang}({}^S T^Q) = N$$

Ist dies der Fall, dann kann \ddot{q} mit Hilfe der Moore-Penrose-Pseudoinversen von ${}^S T^Q$ durch

$$\begin{aligned} \ddot{q} &= ({}^S T^Q T^S T^Q)^{-1} {}^S T^Q T^S \ddot{r}_G - ({}^S T^Q T^S T^Q)^{-1} {}^S T^Q T^S W \\ &= {}^Q T^S(q) \ddot{r}_G + {}^Q W(q, \dot{q}) \end{aligned} \quad (6.7)$$

bestimmt werden, wobei ${}^Q T^S$ die Pseudoinverse von ${}^S T^Q$ ist und ${}^Q W$ der Vektor der Gelenkbeschleunigungen aufgrund der Gelenkzustände q und \dot{q} . Mit Gleichung (6.7) kann das Gleichungssystem (6.6) auch als

$$\begin{aligned} \ddot{q} &= {}^Q T^S(q) \ddot{r}_G + {}^Q W(q, \dot{q}) \\ \ddot{\theta} &= - \left(-J^{-1} S^T M_G^{-1} \quad J^{-1} + J^{-1} S^T M_G^{-1} S J^{-1} \right) \left(\begin{pmatrix} c_{Arm}(q, \dot{q}, \dot{\theta}) \\ c_{Motor}(q, \dot{q}) \end{pmatrix} + \begin{pmatrix} g_{Arm}(q) \\ 0 \end{pmatrix} \right) \\ &\quad + \left(\begin{pmatrix} H_G U \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) + \tau_{RG} \\ -H_A^{-1} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) \end{pmatrix} - \begin{pmatrix} \tau_{Ext} \\ \tau_M \end{pmatrix} \right) \\ y &= \begin{pmatrix} {}^S T^Q(q) \ddot{q} + {}^S W(q, \dot{q}) \\ \theta \end{pmatrix} \end{aligned}$$

dargestellt werden. Da sich außerdem Gleichung (6.6b) durch Umformung auch in Abhängigkeit der Gelenkbeschleunigungen \ddot{q} darstellen lässt

$$\ddot{\theta} = -J^{-1} S^T \ddot{q} - J^{-1} \left(c_{Motor}(q, \dot{q}) - H_A^{-1} \left(k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta}) \right) - \tau_M \right)$$

können die Systemgleichung (6.6) weiter umgeformt werden zu

$$\ddot{q} = {}^Q T^S(q) \ddot{r}_G + {}^Q W(q, \dot{q}) \quad (6.9a)$$

$$\ddot{\theta} = -J^{-1} (S^T ({}^Q T^S(q) \ddot{r}_G + {}^Q W(q, \dot{q})) + c_{Motor}(q, \dot{q}) -$$

$$H_A^{-1} (k(\varphi - \theta) + d(\dot{\varphi} - \dot{\theta})) - \tau_M)$$

$$y = \begin{pmatrix} {}^S T^Q(q) \ddot{q} + {}^S W(q, \dot{q}) \\ \theta \end{pmatrix} \quad (6.9c)$$

Die Achsen des Roboters sind jetzt im Modell nur noch über S^T und $c_{Motor}(q, \dot{q})$ gekoppelt. Die restlichen Kopplungen werden über die Messwerte der Beschleunigungssensoren berücksichtigt.

Die Darstellung der Bewegungsgleichungen in der Form des Gleichungssystems (6.9) hat einige Vorteile. Aufgrund der bekannten kartesischen Beschleunigungen, welche als Eingang angesehen werden können, ist die Mehrkörperdynamik der Glieder nicht mehr enthalten. Damit ist insbesondere keine Kenntnis der Masseneigenschaften des Armes mehr notwendig. Dies ist vor allem dann wichtig, wenn sich die Masseneigenschaften, z. B. durch das Greifen einer Last, im Betrieb verändern können. Zusätzlich treten in Gleichungssystem (6.9) im Gegensatz zu (6.6) die durch externe Kräfte auf den Arm entstehenden Momente τ_{Ext} nicht mehr auf. Damit ist das Modell unabhängig von gewöhnlich nicht messbaren Störungen wie z. B. Prozesskräften geworden. Der Preis für diese Vereinfachungen des Systems ist die Messung von $m \geq N$ Beschleunigungen am Arm und die Berechnung der Transformation aus Gleichung (6.7). Diese hängt jedoch nur von der Kinematik des Roboters ab, womit deren Parameter zeitinvariant und deutlich einfacher zu bestimmen sind als die dynamischen Parameter. Zudem ist der erforderliche Rechenaufwand um einiges geringer.

Das System (6.9) wird im Folgenden als nichtlineares, affines System mit $x = (q \quad \dot{q} \quad \theta \quad \dot{\theta})^T$, $u = (\ddot{r}_G \quad \tau_M)^T$ und $y = \theta$ dargestellt, wobei die Messwerte \ddot{r}_G nun als Eingang angesehen werden. Ist die Berücksichtigung der Dynamik des Stromreglers erwünscht, dann kann das Motormoment einfach aus dem Motorstrom bestimmt werden: $\tau_M = k_T I$.

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned}$$

Die zugehörigen nichtlinearen Funktionen ergeben sich anhand Gleichung (6.9) zu

$$\begin{aligned} f(x) &= \begin{pmatrix} x_2 \\ {}^S W(x_1, x_2) \\ x_4 \\ -J^{-1} (S^T {}^S W(x_1, x_2) + c_{Motor}(x_1, x_2) - H_A^{-1} (k(\varphi(x_1) - x_3) + d(\dot{\varphi}(x_1, x_2) - x_4))) \end{pmatrix} \\ g(x) &= \begin{pmatrix} 0 & 0 \\ {}^S T^Q(x_1) & 0 \\ 0 & 0 \\ -J^{-1} S^T S T^Q(x_1) & -J^{-1} \end{pmatrix} \\ h(x) &= x_3 \end{aligned}$$

6.2.4 Beobachterstruktur und Konvergenz

Als Beobachter wird, wie im linearen Fall bei einem Luenberger Beobachter, die Gleichung der Strecke angesetzt und um eine lineare Rückführung des Schätzfehlers erweitert. Damit sind die Gleichungen des Beobachters als

$$\dot{\hat{x}} = f(\hat{x}) + g(\hat{x})u + L(y - \hat{y}) \quad (6.10a)$$

$$\hat{y} = \hat{x}_3 \quad (6.10b)$$

gegeben, wobei L der Vektor der Beobachterverstärkungen ist. Aufgrund der Nichtlinearität des Systems muss nun der Zusammenhang zwischen den Zuständen der Strecke und des Beobachters untersucht werden. Um die Bedingungen für die Konvergenz der Schätzwerte gegen die realen Werte herauszufinden, wird der Schätzfehler e definiert:

$$e = x - \hat{x}$$

Mit diesem ergibt sich die Fehlerdifferentialgleichung

$$\begin{aligned} \dot{e} &= \dot{x} - \dot{\hat{x}} \\ &= f(x) + g(x)u - f(\hat{x}) - g(\hat{x})u - L(y - \hat{y}). \end{aligned} \quad (6.11)$$

Zur Analyse dieser Gleichung wird das System zunächst in einen linearen und einen nichtlinearen Teil aufgespalten:

$$\begin{aligned} f(x) &= Ax + f_{NL}(x) \\ g(x) &= B + g_{NL}(x) \\ h(x) &= Cx \end{aligned}$$

Für den linearen Anteil werden N entkoppelte Zwei-Massen-Systeme angesetzt. Damit sind die Systemmatrizen des linearen Anteils

$$\begin{aligned} A &= \begin{pmatrix} 0_{N \times N} & I_{N \times N} & 0_{N \times N} & 0_{N \times N} \\ 0_{N \times N} & 0_{N \times N} & 0_{N \times N} & 0_{N \times N} \\ 0_{N \times N} & 0_{N \times N} & 0_{N \times N} & I_{N \times N} \\ J^{-1}k_L U & J^{-1}d_L U & -J^{-1}k_L & -J^{-1}d_L \end{pmatrix} \\ B &= \begin{pmatrix} 0_{N \times 1} & 0_{N \times 1} \\ 0_{N \times 1} & 0_{N \times 1} \\ 0_{N \times 1} & 0_{N \times 1} \\ 0_{N \times 1} & J^{-1} \end{pmatrix} \\ C &= (0_{N \times N} \quad 0_{N \times N} \quad I_{N \times N} \quad 0_{N \times N}) \end{aligned}$$

womit sich die Fehlerdifferentialgleichung (6.11) auch als

$$\dot{e} = (A - LC)e + (f_{NL}(x) - f_{NL}(\hat{x})) + (g_{NL}(x) - g_{NL}(\hat{x}))u_1 \quad (6.12a)$$

$$= (A - LC)e + f_{Rest}(x, \hat{x}, u_1) \quad (6.12b)$$

schreiben lässt. Um die Konvergenz von e gegen 0 sicherzustellen, wird im Folgenden versucht den linearen Teil so einzustellen, dass er den nichtlinearen Anteil überwiegt. Ein solches Vorgehen entspricht einem „high-gain“-Beobachterentwurf.

Zur Analyse wird deshalb zunächst nur der lineare Teil des System betrachtet, welcher durch

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx\end{aligned}$$

gegeben ist. Die Beobachtbarkeitsmatrix dieses linearen Teilsystems ergibt sich zu

$$\begin{aligned}Q_B &= \begin{pmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{pmatrix} \\ &= \begin{pmatrix} 0_{N \times N} & 0_{N \times N} & I_{N \times N} & 0_{N \times N} \\ 0_{N \times N} & 0_{N \times N} & 0_{N \times N} & I_{N \times N} \\ J^{-1}k_L \mathbf{U} & J^{-1}d_L \mathbf{U} & -J^{-1}k_L & -J^{-1}d_L \\ -J^{-2}k_L d_L \mathbf{U} & J^{-1}k_L \mathbf{U} - J^{-2}d_L^2 \mathbf{U} & J^{-2}k_L d_L & J^{-1}k_L - J^{-2}d_L^2 \end{pmatrix}\end{aligned}$$

Aufgrund von

$$\det(Q_B) = \frac{\prod_{i=1}^N k_{L,i}^2 u_i^2}{\prod_{i=1}^N J_i^2}$$

ist das System dann mit Hilfe der Motorpositionen beobachtbar, wenn alle linearen Steifigkeiten $k_{L,i}$ von Null verschieden und die Motorträgheiten J_i beschränkt sind. Diese Bedingungen sind für sinnvolle Modelle sicher immer erfüllt. Aufgrund der Beobachtbarkeit ist es auch möglich, durch geeignete Wahl von L die Dynamik des Schätzfehlers des linearen Systems

$$\dot{e} = (A - LC)e = A_B e$$

beliebig vorzugeben. Zur Untersuchung der Stabilitätseigenschaften der Fehlerdifferentialgleichung (6.12) soll Lyapunovs direkte Methode eingesetzt werden. Vorab wird dazu eine Bedingung von Lyapunov für die Stabilität von linearen Systemen gegeben, welche in verschiedenen Lehrbüchern zur nichtlinearen Regelung zu finden ist. Die Darstellung im folgenden Satz wurde aus [SL91] übernommen.

Satz 6.1 (Stabilität eines LTI Systems).

Eine notwendige und hinreichende Bedingung für die strikte Stabilität des linearen, zeitinvarianten Systems $\dot{x} = Ax$ ist, dass die eindeutige Lösung P der Lyapunov-Gleichung

$$A^T P + PA = -Q \tag{6.13}$$

für irgendeine symmetrische, positiv definite Matrix Q ebenfalls symmetrisch positiv definit ist.

Im Beweis des Satzes wird als Kandidat für die Lyapunov-Funktion des LTI Systems die quadratische Form

$$V(x) = x^T P x$$

verwendet, wobei P eine symmetrische, positiv definite Matrix ist. Damit ist $V(x)$ zwangsläufig ebenfalls positiv definit. Die Ableitung von $V(x)$ führt auf die Gleichung

$$\dot{V}(x) = \dot{x}^T P x + x^T P \dot{x}$$

welche sich durch Einsetzen von $\dot{x} = Ax$ und Verwendung der Lyapunov-Gleichung (6.13) als

$$\dot{V}(x) = -x^T Q x$$

darstellen lässt. Man erkennt, dass $V(x)$ dann eine Lyapunov-Funktion für das LTI System ist, wenn Q positiv definit ist, da dann $\dot{V}(x)$ stets negativ definit ist.

Mit einem ähnlichen Vorgehen kann die Stabilität der Fehlerdifferentialgleichung (6.19) untersucht werden. Als Kandidat für eine Lyapunov-Funktion wird, wie für lineare Systeme, die Funktion

$$V(e) = e^T P e$$

mit einer symmetrischen, positiv definiten Matrix P verwendet. Damit ist V automatisch positiv definit und radial unbeschränkt. Die Ableitung von V ist

$$\begin{aligned} \dot{V}(e) &= \dot{e}^T P e + e^T P \dot{e} \\ &= (A_B e + f_{Rest}(x, \hat{x}, u_1))^T P e + e^T P (A_B e + f_{Rest}(x, \hat{x}, u_1)) \end{aligned} \quad (6.14)$$

Wie für ein lineares System wird auch hier der lineare Anteil aus Gleichung (6.14) in Form einer Lyapunov-Gleichung zusammengefasst

$$A_B^T P + P A_B = -Q. \quad (6.15)$$

Damit kann Gleichung (6.14) auch als

$$\dot{V}(e) = -e^T Q e + 2f_{Rest}(x, \hat{x}, u_1)^T P e \quad (6.16)$$

geschrieben werden. Zur Untersuchung der negativen Definitheit von \dot{V} müssen die Matrizen Q und P bestimmt werden. Dazu wird Q als

$$Q = I_{4N \times 4N}$$

gewählt was, wie in [SL91] gezeigt, der maximalen Konvergenzrate entspricht. Damit $\dot{V} \leq 0$ ist, muss die Ungleichung

$$2f_{Rest}(x, \hat{x}, u_1)^T P e \leq e^T e$$

erfüllt sein. Dies ist mindestens dann der Fall, wenn gilt

$$\|2f_{Rest}(x, \hat{x}, u_1)^T P\| \leq \|e\|. \quad (6.17)$$

Damit \dot{V} für beliebige e negativ definit sein kann muss der Term der Nichtlinearitäten für $\|e\| \rightarrow 0$ ebenfalls gegen 0 gehen. Diese Bedingung ist aufgrund der Definition der Funktion f_{Rest} in Gleichung (6.12) automatisch erfüllt. Kann die Nichtlinearität nun zusätzlich noch durch eine Ungleichung der Form

$$\|f_{Rest}(x, \hat{x}, u_1)\| \leq \|\Delta\| \|e\|$$

mit einer Matrix Δ abgeschätzt werden, dann ist \dot{V} mindestens für

$$\|\Delta\| \|P\| < \frac{1}{2} \quad (6.18)$$

negativ definit. Wenn die Verstärkungen des Beobachters so ausgelegt werden können, dass mit der aus Gleichung (6.15) berechneten Matrix P die Ungleichung (6.18) erfüllt ist, dann wird der Schätzfehler asymptotisch zu Null gehen.

Die Bedingung in Ungleichung (6.18) kann zur Überprüfung der Konvergenz bei vorgegebener Dynamik des linearen Teils des Beobachters benutzt werden. Ein anschaulicherer Spezialfall entsteht, wenn negative, reelle, verschiedene Eigenwerte gewählt werden, womit die Matrix A_B diagonalisierbar wird. Damit kann, unter Zuhilfenahme der Transformation

$$e = T_{diag} \tilde{e}$$

die Fehlerdifferentialgleichung (6.12) in den neuen Zuständen \tilde{e} als

$$\begin{aligned} \dot{\tilde{e}} &= \tilde{A}_B \tilde{e} + T_{diag}^{-1} ((f_{NL}(x) - f_{NL}(\hat{x})) + (g_{NL}(x) - g_{NL}(\hat{x})) u_1) \\ &= \tilde{A}_B \tilde{e} + \tilde{f}_{Rest}(x, \hat{x}, u_1) \end{aligned} \quad (6.19)$$

dargestellt werden, wobei

$$\tilde{A}_B = \text{diag}(\lambda_1, \dots, \lambda_{4N})$$

die Eigenwerte des linearen Systems λ_i enthält. Aufgrund der Diagonalform der Matrix \tilde{A}_B kann die Lösung der Lyapunov-Gleichung (6.15) direkt als

$$\begin{aligned} P &= -\frac{1}{2} \tilde{A}^{-1} \\ &= \frac{1}{2} \text{diag}(-1/\lambda_1, \dots, -1/\lambda_{4N}) \end{aligned}$$

angegeben werden und ist somit sicherlich positiv definit. Die Norm dieser Matrix P ergibt sich mit

$$\|P\| = \frac{1}{2} \max_{i=1 \dots 4N} \left(-\frac{1}{\lambda_i} \right) = -\frac{1}{2\lambda_{min}} \quad (6.20)$$

aus dem langsamsten Eigenwert λ_{min} des linearen Beobachters, womit sich die Ungleichung (6.18) zu

$$-\lambda_{min} > \|\tilde{\Delta}\| \quad (6.21)$$

mit der Abschätzung der transformierten Nichtlinearitäten $\tilde{\Delta}$ vereinfacht.

6.2.5 Auslegung des Beobachters

Bei der Abschätzung des nichtlinearen Anteils der Fehlerdifferentialgleichung entsteht der größte Anteil von Δ durch den Fehler zwischen der Transformation der Beschleunigungswerte mit x und \hat{x} . Wird der Beobachter für beliebig große Schätzfehler e ausgelegt folgt daraus auch eine große Matrix Δ . In der Praxis ist dies jedoch nicht notwendig. Unter Verwendung der vorhandenen Positionsmessung an den Motoren und unter Ausnutzung der Tatsache, dass sich der Roboter beim Einschalten des Reglers in Ruhe befindet, können die Startwerte für $\hat{\theta}$, $\hat{\dot{\theta}}$ und \hat{q} praktisch exakt vorgegeben werden. Mit Hilfe einer Berechnung der Torsion aufgrund der Schwerkraft kann zudem bereits ein sehr guter Startwert für \hat{q} bestimmt werden. Dies erlaubt die Annahme beschränkter Schätzfehler e bei der Abschätzung der Matrix Δ und führt praktisch zu einer wesentlich geringeren Norm von Δ .

Bei der Untersuchung der Konvergenz des Beobachters wird davon ausgegangen, dass die Gelenkbeschleunigungen exakt bestimmt werden können. Da sie durch Messung und Transformation von Beschleunigungen am Arm gewonnen werden, sind die Gelenkbeschleunigungen jedoch nicht zwangsläufig exakt. So zeigen Beschleunigungssensoren im Betrieb häufig eine kleine Drift des Nullpunktes. Ferner kann auch die Transformation mit kleinen Fehlern behaftet sein. In beiden Fällen werden die berechneten Gelenkbeschleunigungen \ddot{q}_B von den echten Gelenkbeschleunigungen \ddot{q} abweichen. Mit Gleichung (6.11) und $L = (L_1 \ L_2 \ L_3 \ L_4)^T$ kann für den stationären Fehler des Beobachters das Gleichungssystem

$$0 = e_2 - L_1 e_3 \quad (6.22a)$$

$$0 = -\ddot{q}_B - L_2 e_3 \quad (6.22b)$$

$$0 = e_4 - L_3 e_3 \quad (6.22c)$$

$$0 = f_4(x) + g_4(x) \begin{pmatrix} 0 \\ u_2 \end{pmatrix} - f_4(\hat{x}) - g_4(\hat{x}) \begin{pmatrix} \ddot{q}_B \\ u_2 \end{pmatrix} - L_4 e_3 \quad (6.22d)$$

angegeben werden. Mit diesen Gleichungen erkennt man, dass die Fehler in der Zustandschätzung von den Verstärkungen L und insbesondere von L_2 abhängen, da e_3 mit Gleichung (6.22b) bestimmt wird. Der Fehler in \ddot{q}_B sollte durch die Wahl von Sensoren mit geringer Drift und eine möglichst exakte Bestimmung der Transformation in den Gelenkraum klein gehalten werden. Um zusätzlich die Auswirkungen des Beschleunigungsfehlers auf den Schätzfehler des Beobachters klein zu halten, können bei der Auslegung des Beobachters die Eigenwerte so gewählt werden, dass sich im Sinne von Gleichung (6.22) günstige Verstärkungen ergeben.

Bei einer beliebig schnellen Eigendynamik des linearen Beobachteranteils werden die geschätzten Motorpositionen des Beobachters praktisch exakt mit der Messung übereinstimmen. In diesem Fall geht der Beobachter in ein Parallelmodell über welches, basierend auf der Bewegung der Motoren, die Zustände der Gelenke berechnet. Damit besitzt er aber keine Filterwirkung bezüglich der Messung der Motorposition mehr, was ebenfalls ungünstig ist.

Deshalb muss, wie bei der Beobachterausslegung üblich, für die Eigendynamik ein Kompromiss gefunden werden. Dies geschieht in der vorliegenden Arbeit mit Hilfe von Optimierungen der Pole des Beobachters, welche ähnlich wie die Parameteroptimierungen zur Identifikation in Abbildung 3.1 verlaufen. Nach Vorgabe von oberen und unteren Schranken für die Eigenwerte wird der Beobachter jeweils mit zuvor aufgezeichneten Eingangsdaten berechnet und der geschätzte Zustand mit Messungen verglichen. Die 2-Normen der Schätzfehler dienen als Kriterien für die Optimierung.

6.2.6 Beobachterausslegung für den KR15/2

Aufgrund von Hardwarebeschränkungen wie z.B. der Anzahl verfügbarer Sensoreingänge beim KR15/2 war die Implementierung des Beobachters nicht für alle Achsen möglich. Die Struktur des Beobachters ermöglicht aber auch eine Anwendung auf die ersten Achsen, da entsprechend Gleichung (6.9) die Achsen des Roboters nur über S^T , c_{Motor} und die gemessenen Beschleunigungen gekoppelt sind. Sollen die Zustände der ersten drei Achsen geschätzt werden, dann darf die Transformation der gemessenen Beschleunigungen auf die Gelenkbeschleunigungen nur von den Winkeln der ersten drei Gelenke abhängen, was durch eine Montage der Sensoren vor der vierten Achse sichergestellt werden kann. Für die Berechnung der

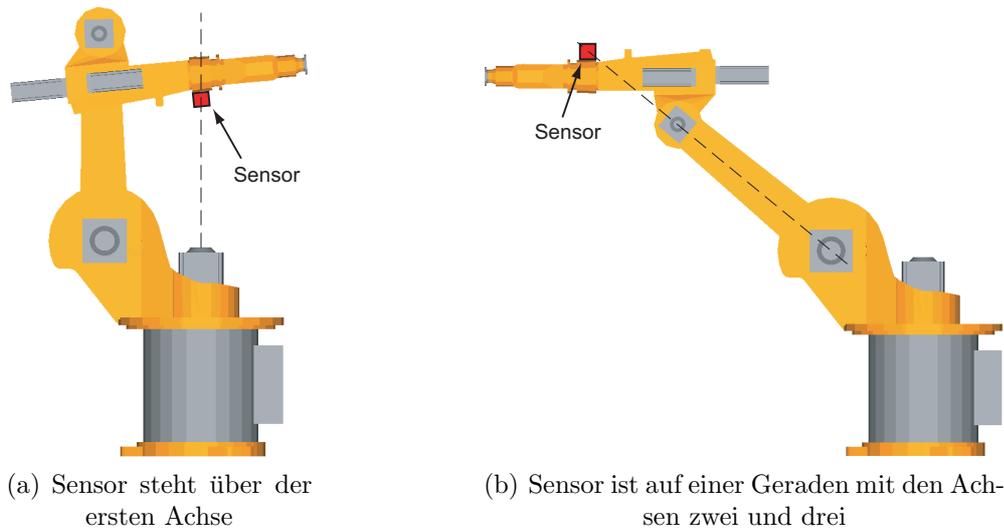


Abbildung 6.3: Position des Beschleunigungssensors am Arm und singuläre Konfigurationen

Kopplungen der Achsen über S^T und c_{Motor} werden, aufgrund der in Kapitel 2.1 gezeigten strukturellen Eigenschaften, ebenfalls nur die Gelenkwinkel und Beschleunigungen der ersten drei Gelenke benötigt. Der KR15/2 zeigt darüber hinaus bezüglich dieser Kopplungen eine besonders einfache Struktur. Mit den Gleichungen (2.30) und (5.41) ergibt sich für die ersten drei Achsen des KR15/2:

$$S_{A1...A3}^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & R_3 I_{R3zz} & 0 \end{pmatrix}$$

$$c_{Motor A1...A3} = 0_{3 \times 1}$$

Zur Berechnung der Gelenkbeschleunigungen wurde ein dreiachsiger Beschleunigungssensor, wie in Abbildung 6.3 dargestellt, vor der vierten Achse des Roboters angebracht. Bei dieser Positionierung am Arm wird die Transformation auf Gelenkbeschleunigungen in zwei Fällen singulär. Zum Einen, wenn der Sensor über der ersten Achse steht, wodurch die Beschleunigung dieser Achse nicht mehr bestimmt werden kann. Zum Anderen, wenn der Sensor auf einer Geraden mit den Achsen zwei und drei liegt, was die Bestimmung der Beschleunigungen dieser Achsen unmöglich macht. Diese Fälle sind ebenfalls in Abbildung 6.3 dargestellt. Zur Behebung der Singularitäten könnte noch ein Sensor in der Nähe der zweiten Achse angebracht werden. Dies war aufgrund der verfügbaren Sensoreingänge jedoch nicht möglich. Daher wurden bei der experimentellen Erprobung Positionen in der Nähe der Singularitäten vermieden, was lediglich den möglichen Arbeitsraum für die Durchführung von Experimenten einschränkt.

Die Auslegung des Beobachters erfolgte achsweise mit Hilfe des in Abschnitt 6.2.5 beschriebenen Optimierungsverfahrens. Einen Vergleich der gemessenen Zustände und der geschätzten Zustände des so eingestellten Beobachters für die erste Achse des KR15/2 zeigt Abbildung 6.4. Anstatt des Gelenkwinkels q wurde dabei die Torsion des Gelenkes $u\theta - q$ dargestellt, da die Abweichungen sonst kaum sichtbar sind. Erkennbar ist, dass der Beobachter die tatsächlichen

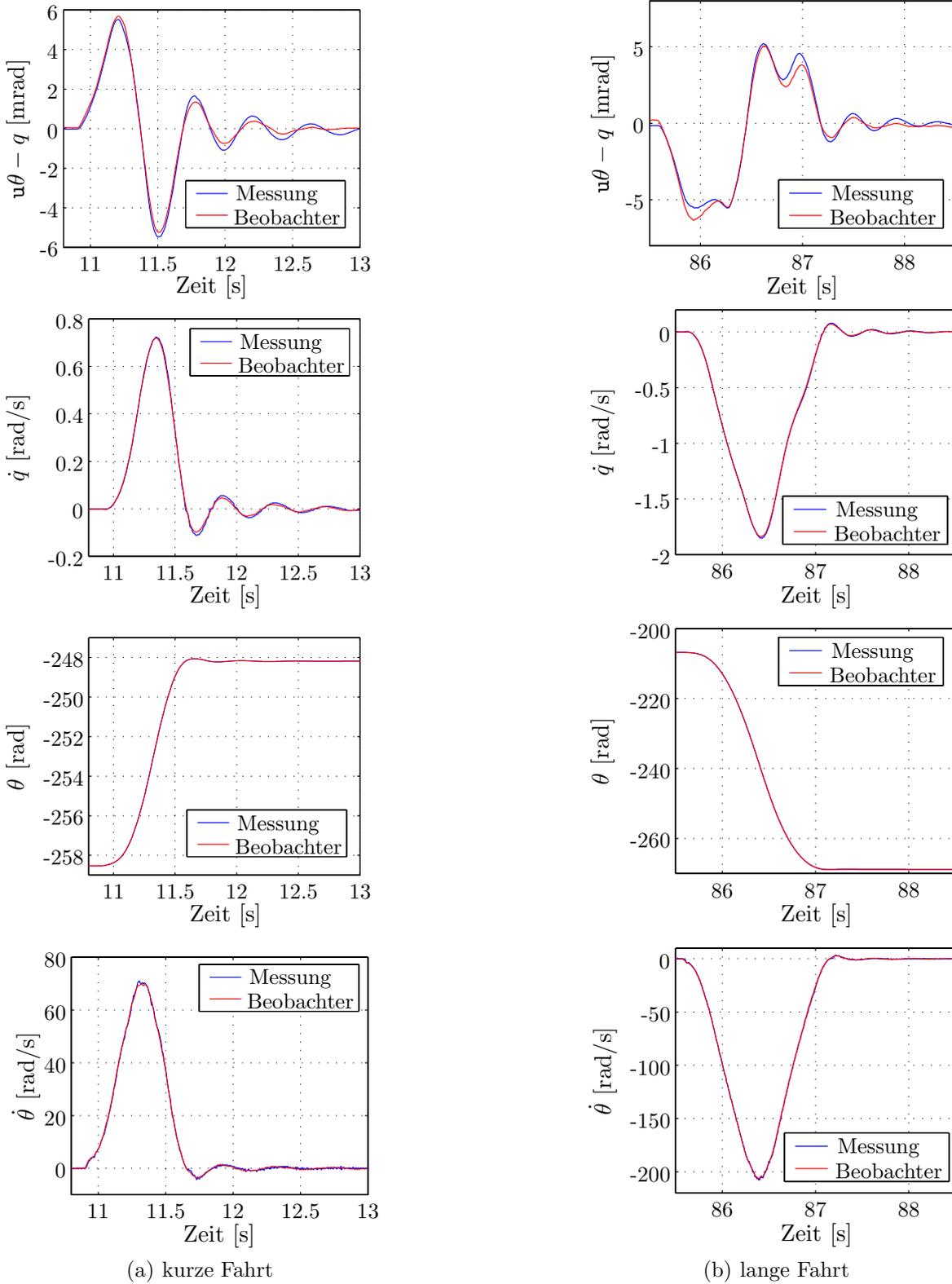


Abbildung 6.4: Vergleich von gemessenen und geschätzten Zuständen der ersten Achse des KR15/2

Zustände sehr gut wiedergibt, und die geschätzten Zustände kaum Rauschen aufweisen. Lediglich die geschätzte Torsion weicht geringfügig von der Messgröße ab, gibt das Verhalten aber insgesamt trotzdem gut wieder.

6.3 Reglerentwurf

Entsprechend den Überlegungen zur Regelungsstruktur in Kapitel 1.3 soll zur Stabilisierung des Systems ein einfacher linearer Regler mit Gain-Scheduling benutzt werden. Dabei wird davon ausgegangen, dass die Abweichungen zwischen den Sollgrößen und den Istgrößen klein bleiben. Ist dies der Fall, dann kompensiert das inverse Modell in der Steuerung bereits den Großteil der Roboterdynamik. Um die verbleibenden Abweichungen auszuregulieren werden einfache Positionsregler für die Achsen eingesetzt. Dieser Ansatz ist unter der Bezeichnung *Nonlinear Regulation* bekannt.

Das wichtigste Ziel bei der Auslegung des Reglers ist die Verbesserung der Eigendynamik, was für die betrachteten Roboter insbesondere eine Erhöhung der Dämpfung bedeutet. Zusätzlich soll der Regler die stationäre Genauigkeit der Position bei externen Momenten und Drift der Beschleunigungssensoren sicherstellen.

6.3.1 PI-Zustandsregler

Aus dem Bereich der Regelung von einzelnen elastischen Antriebssträngen ist bekannt, dass Zustandsregler in der Lage sind die gewünschte Verbesserung der Eigendynamik zu erzeugen [BG98, Gs01, Faß02]. In [AS02] wurde die Eignung von achsweisen Zustandsreglern für RLFJ-Roboter theoretisch und experimentell nachgewiesen. Da es sich bei Zustandsreglern zudem um einfache Regler mit geringem Rechenbedarf handelt, ist deren Anwendung sehr attraktiv. Um die zweite Anforderung, die stationäre Genauigkeit bei externen Momenten und Drift der Beschleunigungssensoren, sicherzustellen müssen die Regler allerdings noch erweitert werden.

Aus den Gleichungen des Gesamtmodells (2.59) wird dazu ein stationäres Modell abgeleitet. Zur Vereinfachung wird dabei angenommen, dass die Lagerreibung in Ruhe Null ist. Unter Ausnutzung von $\tau_M = u_d + u_c$ ergibt sich für das stationäre Modell:

$$g_{Arm}(q) + \mathbb{U}(q)k(\varphi(q) - \theta) = \tau_{Ext} \quad (6.23a)$$

$$-k(\varphi(q) - \theta) = u_d + u_c \quad (6.23b)$$

Gleichung (6.23a) beschreibt die Torsion des Getriebes unter der vorgegebenen Last $\tau_{Ext} - g_{Arm}(q)$. Da die Torsion vom nicht gemessenen externen Moment τ_{Ext} abhängt, ist klar, dass nur entweder die Gelenkpositionen q oder die Motorpositionen θ auf ihre Sollwerte geregelt werden können.

Bei den meisten Robotern werden nur die Positionen der Motoren gemessen. Für eine Regelung auf die Gelenkpositionen müssten daher mittels eines Modells geschätzte Werte verwendet werden. Diese werden bei Modellungenauigkeiten aber nicht exakt sein, womit auch eine Regelung auf die stationäre Genauigkeit dieser Werte keinen Sinn macht. Daher wird im Weiteren die stationäre Genauigkeit der Motorpositionen gefordert welche, für den Idealfall eines exakten Modells, ebenfalls die richtigen Positionen der Gelenke sicherstellt.

Um die Auswirkungen des Reglers auf die stationären Werte zu untersuchen wird Gleichung (6.23b) noch weiter untersucht. Einsetzen von Gleichung (6.23a) und dem Stationärwert der Steuerung $u_d = \mathbf{U}(q_d)^{-1}g_{Arm}(q_d) - \mathbf{U}(q_d)^{-1}\tau_{Ext d}$ liefert:

$$\mathbf{U}(q)^{-1}g_{Arm}(q) - \mathbf{U}(q_d)^{-1}g_{Arm}(q_d) - \mathbf{U}(q)^{-1}\tau_{Ext} + \mathbf{U}(q_d)^{-1}\tau_{Ext d} = u_c \quad (6.24)$$

$$\Delta\tau_{ff}(q, q_d, \tau_{Ext}, \tau_{Ext d}) = u_c \quad (6.25)$$

Dabei bezeichnet $\Delta\tau_{ff}$ den nach der Vorsteuerung verbleibenden Momentenfehler, welcher vom Regler ausgeglichen werden muss. Wird zur Regelung ein auf den Schätzwerten basierender Zustandsregler eingesetzt, dann ergibt sich das Reglermoment u_c mit der Matrix der Reglerverstärkungen K

$$K = (K_q \quad K_{\dot{q}} \quad K_{\theta} \quad K_{\dot{\theta}}) \quad (6.26)$$

$$K_q = \text{diag}(k_{q_1}, \dots, k_{q_N}) \quad (6.27)$$

$$K_{\dot{q}} = \text{diag}(k_{\dot{q}_1}, \dots, k_{\dot{q}_N}) \quad (6.28)$$

$$K_{\theta} = \text{diag}(k_{\theta_1}, \dots, k_{\theta_N}) \quad (6.29)$$

$$K_{\dot{\theta}} = \text{diag}(k_{\dot{\theta}_1}, \dots, k_{\dot{\theta}_N}) \quad (6.30)$$

zu

$$u_c = Ke \quad (6.31)$$

Wie bereits gezeigt, wirkt sich die Drift der Beschleunigungssensoren statisch als Offset in den geschätzten Zuständen aus. Damit kann das Reglermoment auch als

$$u_c = K(x_d - \hat{x}) = K(x_d - x + \hat{e}(\ddot{q}_B)) = Ke + \Delta\tau_{oe}(\ddot{q}_B) \quad (6.32)$$

dargestellt werden, wobei $\tau_{err}(\ddot{q}_B)$ das durch die Sensordrift erzeugte Moment ist. Einsetzen von Gleichung (6.32) in Gleichung (6.24) liefert

$$\Delta\tau_{ff}(q, q_d, \tau_{Ext}, \tau_{Ext d}) - \Delta\tau_{oe}(\ddot{q}_B) = K_q(q_d - q) + K_{\theta}(\theta_d - \theta) \quad (6.33)$$

wobei θ_d von der Steuerung als Lösung der Gleichung

$$\theta_d = \varphi(q_d) - k^{-1}(\mathbf{U}^{-1}(q_d)(\tau_{Ext d} - g_{Arm}(q_d))) \quad (6.34)$$

bestimmt wird. Man erkennt, dass sowohl für stationäre Ungenauigkeiten in der Steuerung, als auch für Drift der Sensoren, die Sollwerte der Positionen nicht erreicht werden.

Zur Lösung dieses Problems werden zusätzliche PI-Regler verwendet, was pro Achse auf einen PI-Zustandsregler führt. Dieser ist allgemein z. B. in [Föl94] beschrieben und wurde auch in [BG98] zur Regelung eines elastischen Antriebsstrangs eingesetzt.

Damit erweitert sich der Regler mit den Verstärkungsmatrizen des PI-Reglers

$$K_P = \text{diag}(k_{P_1}, \dots, k_{P_N}) \quad (6.35)$$

$$K_I = \text{diag}(k_{I_1}, \dots, k_{I_N}) \quad (6.36)$$

zu

$$u_c = K_P(\theta_d - \theta) + K_I \int_0^t (\theta_d - \theta) d\tau + Ke \quad (6.37)$$

Der durch den Integralanteil des Reglers gewonnene neue Zustand

$$\dot{x}_I = \theta_d - \theta \quad (6.38)$$

stellt in der Ruhelage die korrekte Position der Motoren

$$\theta_d = \theta \quad (6.39)$$

sicher. Die zugehörigen Positionen der Gelenke ergeben sich, abhängig von $g_{Arm}(q)$ und den externen Momenten τ_{Ext} , entsprechend den Steifigkeiten und Übersetzungen der Achsen aus Gleichung (6.23a). Beachtenswert ist in diesem Zusammenhang noch, dass eine Rückführung der geschätzten Zustände über den PI-Regler aufgrund der möglichen stationären Schätzfehler nicht die gewünschte Ruhelage einstellt. Dies ist leicht zu erkennen, indem in Gleichung (6.38) θ durch $\hat{\theta} + e_3$ ersetzt wird.

Die Auslegung der achsweisen PI-Zustandsregler erfolgt einzeln mit Hilfe von linearisierten Achsmodellen. Dazu werden die Achsen als Zwei-Massen-Systeme dargestellt. Wie in Kapitel 4.4 gezeigt, ist dieses System steuerbar. Entsprechend [Föl94] ist die um den PI-Zustandsregler erweiterte Strecke genau dann steuerbar, wenn die ursprüngliche Strecke (A,B) steuerbar ist und

$$\text{Rang} \begin{pmatrix} A & B \\ -C_R & 0 \end{pmatrix} = n + q \quad (6.40)$$

gilt. Dabei ist $y_{q \times 1} = C_R x_{n \times 1}$ der Regelvektor, welcher über den PI-Regler zurückgeführt wird. Im Fall des Zwei-Massen-Systems ist diese Bedingung für sinnvolle Parameterwerte immer erfüllt,

$$\text{Rang} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ -\frac{k}{J_a} & -\frac{d}{J_a} & \frac{k}{J_a} & \frac{d}{J_a} & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \frac{k}{J_m} & \frac{d}{J_m} & -\frac{k}{J_m} & -\frac{d}{J_m} & \frac{1}{J_m} \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} = 5$$

womit die Pole für die Achse durch den PI-Zustandsregler beliebig vorgegeben werden können. Der Regler bietet damit für die idealisierten Achsmodelle volle Entwurfsvollfreiheit bei Einhaltung der gewünschten stationären Genauigkeit. Zur Auslegung der Reglerverstärkungen sind daher die üblichen Ansätze wie z. B. Polvorgabe und Riccati-Entwurf möglich.

6.3.2 Positionsabhängiges Gain-Scheduling der Reglerparameter

Das lineare Modell hängt, speziell für die Achsen nahe am Roboterfuss, über das Armträgheitsmoment $J_a = J_a(q)$ stark vom aktuellen Arbeitspunkt ab. Deshalb wird eine gegenüber der Variation von J_a robuste Auslegung der Achsregler zu einer konservativen Einstellung der Regler führen. Da sich das Armträgheitsmoment aber nur vergleichsweise langsam mit der Position des Roboters ändert, kann zusätzlich ein positionsabhängiges Gain-Scheduling der Reglerparameter verwendet werden, um den Regler an den aktuellen Arbeitspunkt anzupassen. Der Regler mit Gain-Scheduling ist als Blockschaltbild in Abbildung 6.5 dargestellt, die gesamte Reglerstruktur in Abbildung 6.6. Wie schon erwähnt existiert zur Auslegung der Reglerparameter eine Fülle von Möglichkeiten. Dazu gehören insbesondere Verfahren wie Polfestlegung, Riccati-Entwurf und spezielle Auslegungsregeln wie z. B. das Symmetrische

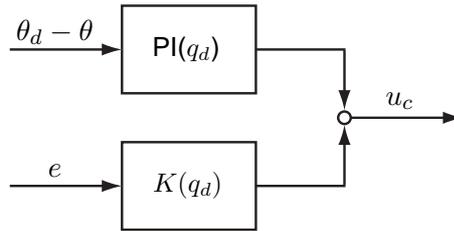


Abbildung 6.5: Aufbau des PI-Zustandsreglers

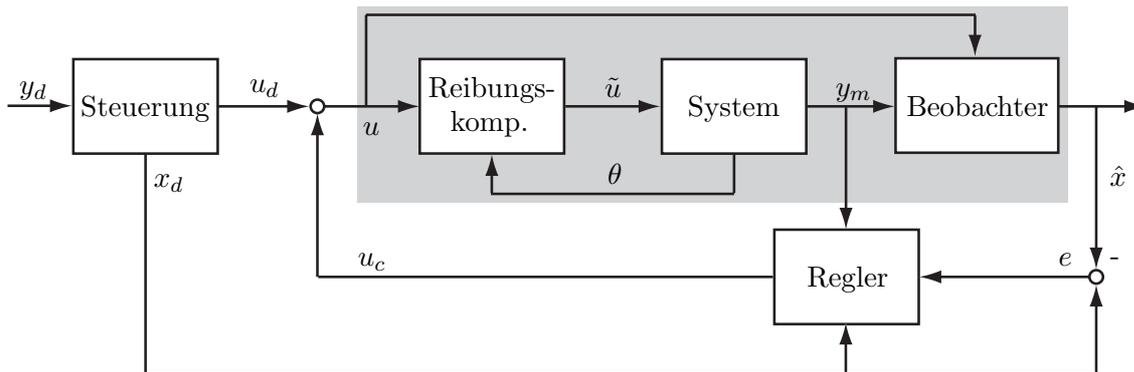


Abbildung 6.6: Aufbau der Regelung

Optimum oder das Dämpfungsoptimum. Je nach Anforderungen und Roboter eignen sich diese Verfahren unterschiedlich gut. Eine pauschale Aussage ist hier nicht angebracht.

In Anlehnung an [AS02], worin eine ausführliche Analyse verschiedener Varianten von Zustandsreglern mit variablen Parametern für RLFJ-Roboter zu finden ist, wird in dieser Arbeit jedoch eine etwas andere Methode gewählt. Für die Parametrisierung des Reglers in der Anwendung ist es anschaulich die gewünschten Steifigkeiten k_d für die geregelten Gelenke vorzugeben. Damit werden unter Verwendung der gewünschten Dämpfung δ_1 und der Armträgheit J_a mit

$$s^2 + \frac{d_d}{J_a}s + \frac{k_d}{J_a} = s^2 + 2\delta_1\omega_1s + \omega_1^2 \quad (6.41)$$

zwei Pole festgelegt, wobei ω_1 die resultierende Eigenfrequenz des Polpaars ist und d_d die entsprechende Dämpfung im Getriebe. Die verbleibenden drei Pole pro Achse werden als Kombination aus einem weiteren Polpaar und einem reellen Pol parametrisiert, womit sich für das charakteristische Polynom des geschlossenen Kreises die Parametrierung

$$(s^2 + 2\delta_1\sqrt{\frac{k_d}{J_a}}s + \frac{k_d}{J_a})(s^2 + 2\delta_2\omega_2s + \omega_2^2)(s + \omega_3) \quad (6.42)$$

mit den Parametern δ_1 , k_d , J_a , δ_2 , ω_2 und ω_3 ergibt. Um die über k_d und δ_1 vorgegebene Dynamik nicht zu stören, werden ω_2 und ω_3 deutlich höher gewählt als die aus den Vorgaben bestimmte Eigenfrequenz ω_1 . Anschaulich gesprochen wird die Dynamik der Achse damit in einen schnellen Anteil zur Regelung des Motors und einen langsameren Anteil zur Regelung des Gelenks aufgeteilt. Prinzipiell wäre es möglich die durch ω_2 , δ_2 und ω_3 gegebene Dynamik des schnellen Anteils z. B. anhand des berechneten ω_1 ebenfalls an den Arbeitspunkt anzupassen. Solche Erweiterungen ergaben in praktischen Versuchen jedoch keine nennenswerten

Verbesserungen und wurden deshalb nicht verwendet. Damit erfolgt die Vorgabe der Reglerverstärkungen durch geeignete Polvorgabe, wobei die Lage der Pole über Gleichung (6.42) an den Arbeitspunkt angepasst wird.

Für die Auslegung ist zu beachten, dass es sich, unter Verwendung des PI-Zustandsreglers, bei den Steifigkeiten k_d um „dynamische“ Steifigkeiten handelt. Normalerweise würde man erwarten, dass ein System mit der Steifigkeit k_d bei einer stationären Störung τ_{Ext} entsprechend

$$\varphi_{\Delta} = \frac{\tau_{Ext}}{k_d} \quad (6.43)$$

ausgelenkt wird. Durch den Integralanteil im Regler wird die Motorposition jedoch stationär immer auf den Sollwert geregelt, womit sich die tatsächliche Auslenkung unabhängig von k_d anhand der physikalischen Gelenksteifigkeit k ergibt:

$$\varphi_{\Delta} = \frac{\tau_{Ext}}{k} \quad (6.44)$$

Wenn das Gelenk auch stationär die gewünschte Steifigkeit zeigen soll muss daher ein Regler ohne PI-Anteil benutzt werden.

6.3.3 Auslegung am KR15/2

Die entworfenen PI-Zustandsregler auf Basis der geschätzten Zustände wurden an den ersten drei Achsen des KR15/2 implementiert und getestet. Da der Entwurf der Regler anhand von vereinfachten linearen Modellen durchgeführt wurde, stand dabei vor allem die praktische Untersuchung der Stabilitätseigenschaften im Vordergrund. Dazu wurden verschiedene Reglerparametersätze benutzt und die resultierende Eigendynamik des Roboters mit den Vorgaben verglichen. Diese Vergleiche sind am anschaulichsten, wenn der Roboter durch externe Momente angeregt wird, da dann die Eigendynamik des geregelten Systems unabhängig von der Steuerung ist und leichter bewertet werden kann. Aus diesem Grund wurden Versuche durchgeführt, in denen der Roboter manuell durch Stöße am Arm angeregt wird und die Dynamik anhand des Einschwingvorgangs bewertet. Da die Versuche aufgrund der manuellen Anregung des Roboters nicht exakt reproduzierbar sind, erlauben sie nur qualitative Aussagen über das Verhalten und sind untereinander nicht direkt vergleichbar. Dies ist jedoch zur Bewertung der Eigendynamik auch nicht notwendig.

Wegen der in Abschnitt 6.2.6 beschriebenen Singularitäten in der Transformation der Beschleunigungsmessungen auf Gelenkbeschleunigungen konnte der Regler in der Nähe der singulären Konfigurationen nicht getestet werden. Mit Ausnahme dieser Konfigurationen konnten die Regler bei geeigneter Parametrierung den Roboter jedoch im gesamten Arbeitsraum stabilisieren.

Der Bereich möglicher Parameter war dabei im Wesentlichen durch Obergrenzen für die Eigenfrequenzen beschränkt. Bei der Vorgabe zu hoher Eigenfrequenzen, z. B. durch die Wahl einer zu hohen Steifigkeit, begannen die Achsen zu brummen und konnten bei einer weiteren Erhöhung der Eigenfrequenzen sogar instabil werden. Insbesondere konnte nur Stabilität erreicht werden, wenn die Steifigkeiten kleiner als die physikalischen Steifigkeiten der Gelenke gewählt wurden. Eine solche Beobachtung ist auch in [AS02] für die Regelung des DLR LBR 2 zu finden. Allgemein entspricht dieses Verhalten der Tatsache, dass für die Regelung in der Praxis aufgrund von Unsicherheiten keine beliebig große Bandbreite erzielt werden kann.

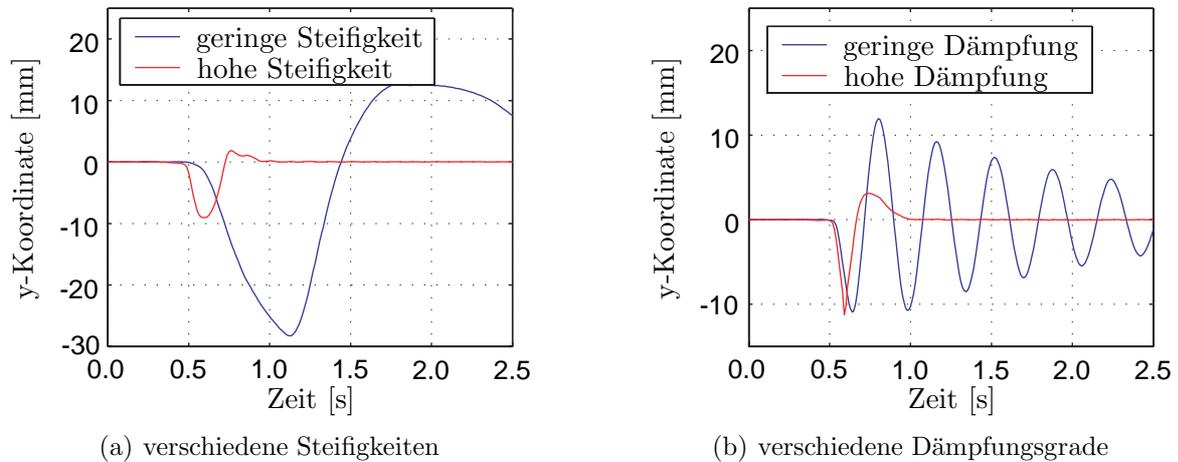


Abbildung 6.7: Durch den Regler einstellbare Eigendynamik der ersten Achse

Weitere Beschränkungen konnten bei sehr kleinen Parameterwerten festgestellt werden. So wurden die Achsen bei der Wahl extrem kleiner Dämpfungen teilweise ebenfalls instabil, was an der Empfindlichkeit einer solchen Polvorgabe gegenüber Modellfehlern liegen dürfte. Für Bahnfolgeanwendungen sind diese Beschränkungen jedoch kaum relevant, da hier stets eine schnelle, gute gedämpfte Dynamik gefordert wird.

Zur Illustration des mit den PI-Zustandsreglern möglichen Bereichs zur Vorgabe der Eigendynamik sind in Abbildung 6.7 Ergebnisse aus den Versuchen mit der ersten Achse dargestellt. Man erkennt, dass es mit Hilfe der Regler möglich ist die Steifigkeit und Dämpfung der geregelten Gelenke in einem großen Bereich vorzugeben. Insbesondere ist es innerhalb der Beschränkungen möglich eine schnelle, gut gedämpfte Eigendynamik der Gelenke zu erreichen.

Kapitel 7

Regleroptimierung und experimentelle Ergebnisse

In diesem Kapitel wird zunächst die Einstellung der Reglerparameter und insbesondere die dazu benutzte Hardware-in-the-loop-Optimierung (HIL-Optimierung) vorgestellt. Anschließend werden experimentelle Ergebnisse gezeigt, in denen das Fahrverhalten des neuen Reglers bezüglich Positionieren, Bahnfahren und Robustheit mit dem Originalregler verglichen wird. Der zur Erprobung des Reglers eingesetzte Prüfstand für den KUKA KR15/2 Roboter mit angebrachtem Beschleunigungssensor, dem Koordinatenmessgerät Krypton Roboscope 6D und den zugehörigen Rechnern ist in Abbildung 7.1 zu sehen.

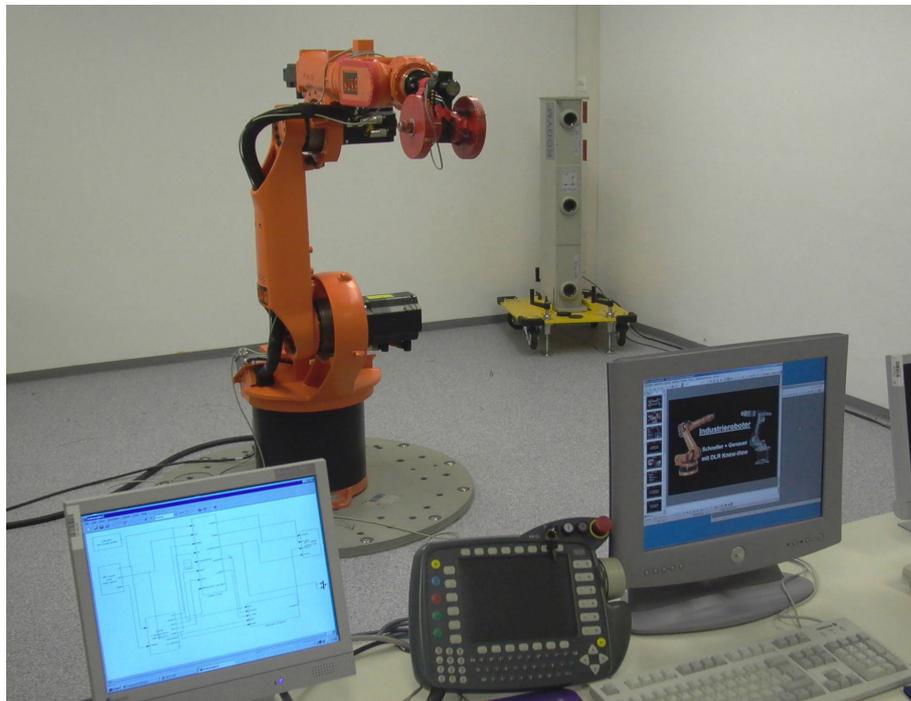


Abbildung 7.1: Prüfstand zur experimentellen Erprobung der Regelung

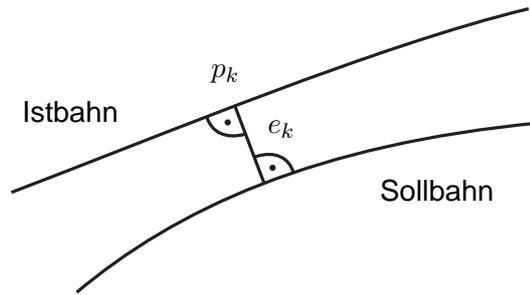


Abbildung 7.2: Bestimmung der Bahnabweichung eines Punktes p_k

7.1 Hardware-in-the-loop-Optimierung

Für die Bewertung der Güte eines Reglers zur Trajektorienfolge sind letztlich die Abweichungen des Roboters von seiner Sollbahn relevant. Daher werden zunächst Bewertungskriterien als Maß für die Bahngenauigkeit entwickelt. Anschließend werden die Reglerparameter mit Hilfe der aufgestellten Kriterien und einer Hardware-in-the-loop-Optimierung am Roboter optimiert. Das in dieser Arbeit verwandte Vorgehen zur HIL-Optimierung setzt auf dem in [GBJ97] und [ABJ97] beschriebenen Verfahren zur HIL-Optimierung des Positionierverhaltens auf und erweitert es um die Messung und Bewertung von Bahnfehlern.

7.1.1 Bewertungskriterien für das Fahrverhalten

Zur Messung der Güte des Fahrverhaltens von Robotern existieren bereits einige Arbeiten. Ein Überblick über relevante Normen und Maße für die Genauigkeit von Robotern ist z. B. in [Alb01] enthalten. Am bekanntesten ist vermutlich die Norm ISO 9283 [Int98], welche Kriterien für das Fahrverhalten definiert. Die in dieser Arbeit verwendeten Kriterien orientieren sich ebenfalls an der ISO 9283 Norm, wurden aber für eine bessere Eignung als Kriterien einer Optimierung etwas erweitert.

Die *Bahnabweichung* e_k eines Punktes p_k wird in dieser Arbeit wie auch in der ISO 9283, als Abstand des Punktes von der Sollbahn verstanden. Dabei spielt die zeitliche Zuordnung zwischen Sollbahn und Istbahn keine Rolle, da e_k der minimale Abstand des Punktes zur gesamten Sollbahn ist. Dies ist in Abbildung 7.2 veranschaulicht. Basierend auf den Bahnabweichungen der einzelnen Punkte werden zur Bewertung der Güte des betrachteten Bahnsegments die 2-Norm und die ∞ -Norm der Bahnabweichungen e benutzt:

$$E_{B,2} = \|e\|_2 = \sqrt{\sum_k |e_k|^2} \quad (7.1a)$$

$$E_{B,\infty} = \max_k |e_k| \quad (7.1b)$$

In der ISO 9283 Norm wird nur die maximale Abweichung von der Bahn verwendet, was dem Wert von $E_{B,\infty}$ entspricht. Dieser Wert eignet sich jedoch schlecht zur iterativen Einstellung der Reglerparameter, da er aufgrund der Maximumfunktion sehr empfindlich gegen Messrauschen ist und ein Großteil der Bahn unberücksichtigt bleibt. Der Wert von $E_{B,2}$ dagegen

berücksichtigt die gesamte Bahn und ist aufgrund der Aufsummierung unempfindlicher gegen Rauschen.

Zur *Bewertung des Positioniervorgangs* am Ende der Sollbahn werden zusätzliche Positionierkriterien verwendet. Die ISO 9283 Norm verwendet dazu eine „position stabilization time“ und einen „position overshoot“. Dabei wird ein Bereich um die Endposition, das sogenannte „limit band“ vorgegeben. Als Stabilisierungszeit wird die Zeit vom ersten Eintreten bis zum dauerhaften Verbleiben in diesen Bereich berechnet. In der gleichen Weise wird das Überschwingen als maximale Abweichung von der Endposition nach dem ersten Eintreten in den Bereich bestimmt. Nachteilig im Sinne einer Optimierung ist bei diesen Kriterien die Verwendung des limit band. Schon das Messrauschen kann dazu führen, dass der Eintritt der Bahn in den Bereich erst in der nächsten Schwingungsperiode erfolgt, was aber zu einer massiven Änderung der Kriterienwerte führt. Daher werden in dieser Arbeit zur Bewertung des Positioniervorgangs ebenfalls die 2-Norm und die ∞ -Norm benutzt. Zur Bestimmung des Beginns des Positioniervorgangs wird nicht das limit band verwendet, sondern das Ende der Sollbahn. Damit ergibt sich

$$E_{P,2} = \|e\|_2 = \sqrt{\sum_k |e_k|^2} \quad \forall k \text{ mit } \dot{y}_{d,k} = 0 \quad (7.2a)$$

$$E_{P,\infty} = \max_k |e_k|, \quad (7.2b)$$

wobei e_k der Abstandsvektor des Punktes p_k von der Endposition ist. Bei diesen Kriterien ist $E_{P,\infty}$ ähnlich dem position overshoot, jedoch berechnet ab dem Ende der Sollbahn. Die Zeit zum Erreichen der Position wird über das Kriterium $E_{P,2}$ mit berücksichtigt.

7.1.2 Ablauf der Hardware-in-the-loop-Optimierung

Aufgrund der komplexen nichtlinearen Zusammenhänge zwischen den Reglerparametern und den daraus resultierenden Bahnabweichungen und Kriterien ist die Bestimmung optimaler Reglerparameter nicht direkt möglich. Qualitative Aussagen über das Verhalten können mit Hilfe von Untersuchungen des Modells in Simulationen gewonnen werden. Eine komplette Auslegung des Reglers mit Hilfe von Simulationen ist jedoch problematisch, da dabei Modellunsicherheiten nur ungenügend berücksichtigt werden können. Aus diesen Gründen werden die Reglerparameter iterativ durch Versuche am Roboter eingestellt. Dieses Vorgehen wird als Hardware-in-the-loop (HIL) Optimierung bezeichnet. Der Ablauf der HIL-Optimierung ist in Abbildung 7.3 dargestellt. Der Roboter fährt mit den aktuellen Reglerparametern die vorgegebene Bahn ab. Die Bewegung des Roboters wird dabei von Sensoren aufgezeichnet. Bei der Optimierung des KR15/2 wurde hierfür das Koordinatenmessgerät Roboscope 6D [Kry01] verwendet, mit dem die Bahn des Roboters auf ca. 0,1 mm genau vermessen werden kann. Durch Auswertung der Soll- und Istbahn wird anschließend mit Hilfe der in Abschnitt 7.1.1 aufgestellten Kriterien das Fahrverhalten bewertet. Ein Werkzeug zur mehrzielligen Optimierung, wie z. B. MOPS [JBL⁺02], iteriert auf Basis der Kriterien $E_{B,2}$ und $E_{P,2}$ die Reglerparameter. Die Kriterien $E_{B,\infty}$ und $E_{P,\infty}$ werden während der Optimierung lediglich zur Kontrolle des Optimierungsverlaufs benutzt. Da die Kriterien aufgrund des Messrauschens der Sensoren ebenfalls ein gewisses Rauschen zeigen, sind gradientenbasierte Optimierungsverfahren für die HIL-Optimierung nicht gut geeignet. Daher wird das Verfahren *Pattern-Search* [HJ61]

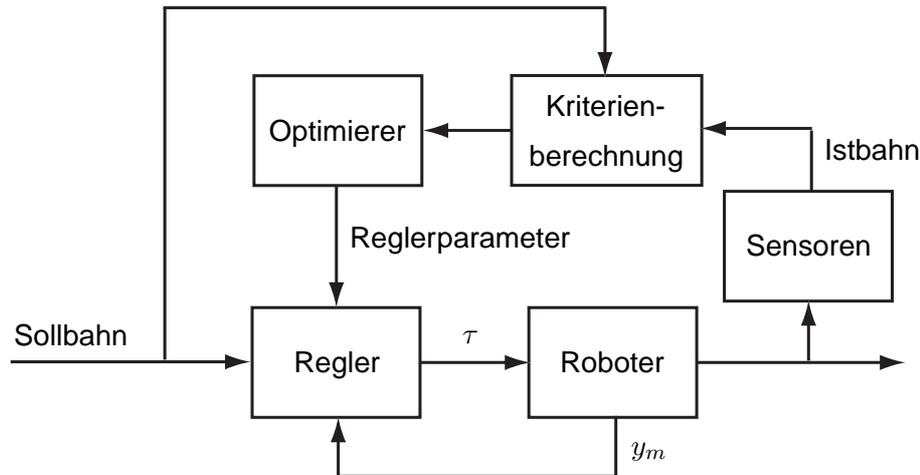


Abbildung 7.3: Ablauf der Hardware-in-the-loop-Optimierung am Roboter

benutzt, welches keine Gradienten benötigt aber in der Regel trotzdem mit einer geringen Anzahl von Optimierungsschritten auskommt.

Ein wesentlicher Vorteil bei der Einstellung der Reglerparameter über die beschriebene HIL-Optimierung ist die direkte Bewertung der Güte des Reglers anhand von aussagekräftigen Kriterien welche auch für spätere Applikationen relevant sind. So ist z. B. die maximale Bahnabweichung ein Maß, welches direkt mit den Anforderungen aus der Applikation des Roboters verglichen werden kann. Ein weiterer Vorteil ist, dass die Bewertung anhand des echten Roboters erfolgt und somit alle in der Praxis auftretenden Effekte und Unsicherheiten beinhaltet. Nachteilig dagegen ist, dass die Einstellung der Reglerparameter ein nichtlineares Optimierungsproblem darstellt und damit das Auffinden eines globalen Minimums nicht garantiert werden kann. Zur Entschärfung dieses Problems können die gleichen Maßnahmen wie bei der Identifikation der Modellparameter in Kapitel 3.2 getroffen werden:

Wahl guter Startwerte und Parameterbereiche:

Wenn das Robotermodell eine akzeptable Übereinstimmung mit der Realität besitzt, dann ist eine erste Auslegung des Reglers bereits auf Basis dieses Modells möglich. Zusätzlich kann das Stabilitätsgebiet abgeschätzt und damit der Bereich sinnvoller Reglerparameter für die Optimierung eingeschränkt werden. Über Experimente am Roboter kann der ermittelte Bereich verifiziert und gegebenenfalls angepasst werden.

Variation der Startwerte:

Optimierungen mit verschiedenen Startwerten der Parameter erhöhen die Chance das globale Minimum zu finden. Da eine Auswertung der Gütefunktion bei der HIL-Optimierung jedoch einem kompletten Roboterexperiment entspricht, ist dieser Weg sehr zeitaufwändig, womit die Anzahl der möglichen Variationen begrenzt ist.

Verwendung geeigneter Kriterien:

Wie bereits in Abschnitt 7.1.1 erläutert spielt auch bei der HIL-Optimierung die Wahl geeigneter Kriterien eine wichtige Rolle. Eine mindestens ebenso große Bedeutung kommt aber der Sollbahn zu, welche für die Optimierung verwendet wird. Diese sollte zum einen den Roboter ausreichend anregen um die nötige Sensitivität der Kriterien auf

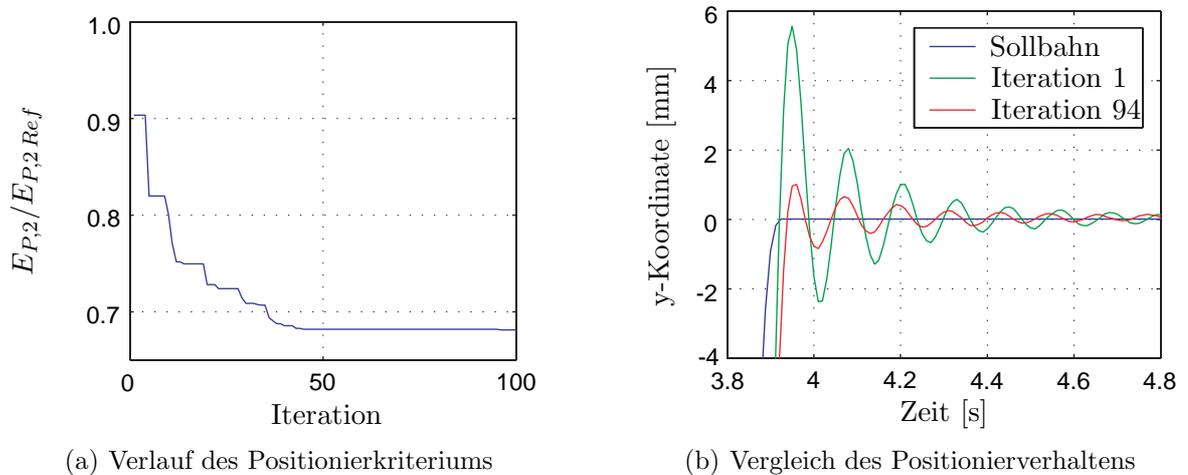


Abbildung 7.4: Optimierungsverlauf am Beispiel einer HIL-Optimierung der Reglerparameter für die ersten Achse

die Reglerparameter sicherzustellen. Zum anderen sollte sie verschiedene Bewegungen in einem möglichst großen Bereich des Arbeitsraumes beinhalten, da ansonsten die Gefahr besteht, dass die Optimierung nur ein lokales Minimum der Kriterien für die gewählte Sollbahn findet, andere Bahnen aber nicht ähnlich gut gefahren werden.

Damit erfolgt die Auslegung der Reglerparameter zweckmäßigerweise zunächst in der Simulation. Die anschließende HIL-Optimierung am Roboter dient der Feinabstimmung des Reglers. Um die Anzahl der gleichzeitig zu optimierenden Reglerparameter klein zu halten werden die Achsregler einzeln nacheinander HIL-optimiert. Dabei haben sich hochdynamische Positionierbewegungen an mehreren Punkten im Arbeitsraum als geeignet herausgestellt, da sie eine gleichzeitige Beurteilung der Bahnabweichungen während der Fahrt und damit der Kombination aus Steuerung und Regler ebenso erlauben wie die Beurteilung der Eigendynamik anhand des Einschwingvorganges am Ende der Bewegung. Mit diesem Vorgehen konnten bei der HIL-Optimierung des KR15/2 gute Konvergenzeigenschaften erzielt werden. Je nach Anzahl der zur Optimierung freigegebenen Reglerparameter waren gewöhnlich nach etwa 100 Iterationen keine nennenswerten Veränderungen der Kriterien mehr erkennbar. Einen typischen Verlauf des Optimierungskriteriums zeigt Abbildung 7.4(a). Einen Eindruck von den durch die Optimierung erzielten Verbesserungen gibt Abbildung 7.4(b). Bei dieser Optimierung wurden große Variationen der Reglerparameter von durchschnittlich etwa $\pm 50\%$ zugelassen um den Verlauf der Optimierung gut darstellen zu können.

7.2 Experimentelle Ergebnisse

Zur Beurteilung der Regelgüte wird der neu entwickelte Regler mit dem Originalregler des Roboters vom Hersteller verglichen. Dazu wird das Verhalten beim Positionieren und auf der Bahn anhand der aufgestellten Kriterien verglichen und mit Messschrieben veranschaulicht. Schließlich wird die Robustheit des Reglers mit Hilfe von Monte-Carlo-Analysen untersucht.

Als Referenzmodell wird ein Bessel-Filter 4. Ordnung verwendet, welches in der Lage ist die notwendigen Ableitungen der Sollgrößen zu berechnen. Aufgrund der von der Bahnplanung vorgegebenen Abtastfrequenz der Sollgrößen kann die Eckfrequenz maximal bei etwa 20 Hz gewählt werden. Um einen großen Dynamikbereich für die Anregung des Roboters zu ermöglichen wird dieser Wert in den folgenden Versuchen verwendet.

7.2.1 Positionierversuche

Beim KUKA KR15/2 werden zum Positionieren sogenannte Punkt-zu-Punkt (PTP) Bewegungen verwendet bei denen vom Benutzer nur der Start- und Zielpunkt vorgegeben werden. Die Bewegung selbst wird von der Bahnplanung des Roboters zeitoptimal so geplant, dass die möglichen Momente und Drehzahlen der Achsen eingehalten werden. Daher kann für die Sollbahnen keine einfache Gleichung angegeben werden. Aufgrund der zeitoptimalen Planung entstehen aber hochdynamischen Bewegungen, welche den Roboter stark anregen und sich daher auch gut zur Bewertung der Regelgüte eignen. In den Versuchen werden zunächst PTP-Bewegungen mit einzelnen Achsen an verschiedenen Arbeitspunkten durchgeführt. Stellvertretend werden an dieser Stelle wieder die Versuche mit der erste Achse beschrieben.

Die Untersuchung des Positionierverhaltens der ersten Achse erfolgt an drei Arbeitspunkten, welche in Tabelle 7.1 dargestellt sind. Diese unterscheiden sich vor allem hinsichtlich des Trägheitsmomentes um Achse 1. Im Arbeitspunkt „Außen“ hat der Arm bezüglich der ersten Achse fast das maximale, im Arbeitspunkt „Innen“ dagegen ein sehr kleines Trägheitsmoment.

Untersucht wurden pro Arbeitspunkt 10 Bewegungen mit unterschiedlichen Längen zwischen 2° und 30° . In Abbildung 7.5 sind Vergleiche des Positionierverhaltens an den Arbeitspunkten jeweils für eine kurze und eine lange Bewegung dargestellt. Zugunsten der Übersichtlichkeit wurde dabei nur die Abweichung in Bewegungsrichtung der Achse dargestellt, welche bei diesen Bewegungen auch den größten Anteil am Fehler darstellt. Für die Berechnung der Bewertungskriterien wurde jedoch die komplette räumliche Bewegung benutzt. Einen Vergleich der gemittelten Positionierkriterien für alle Bewegungen zeigt Tabelle 7.2. Man erkennt, dass die Kriterien des neuen Reglers gegenüber dem Originalregler auf etwa 20 Prozent ihres Wertes sinken. Zudem erkennt man, dass die Kriterien umso kleiner werden je weiter innen der Arbeitspunkt liegt obwohl, wie aus Abbildung 7.5 ersichtlich ist, die Fehler im Gelenkwinkel der ersten Achse an allen Arbeitspunkten in etwa gleich groß sind. Die Ursache dafür ist,

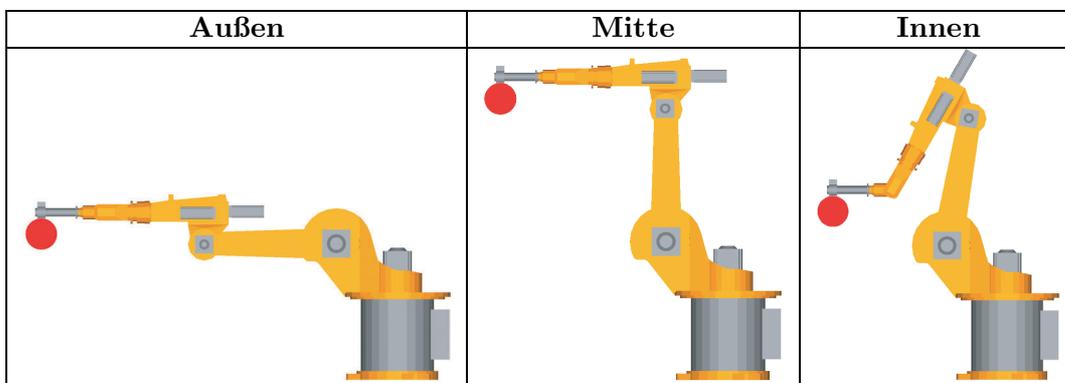


Tabelle 7.1: Arbeitspunkte für die Untersuchung des Positionierverhaltens der ersten Achse

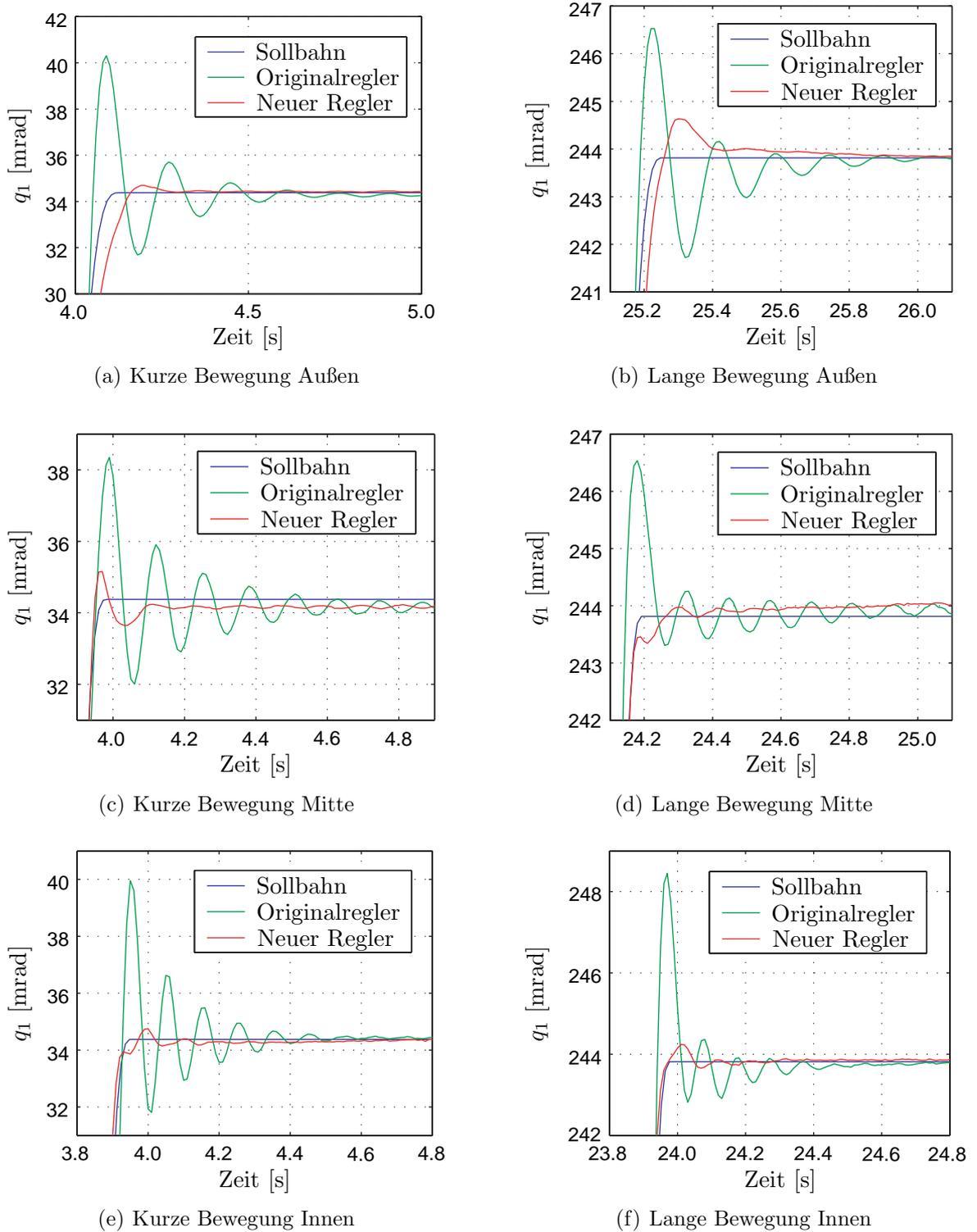


Abbildung 7.5: Vergleich des Positionierverhaltens der ersten Achse bei Bewegungen unterschiedlicher Länge an verschiedenen Arbeitspunkten

	$E_{P,2}$			$E_{P,\infty}$		
	Außen	Mitte	Innen	Außen	Mitte	Innen
Originalregler	14,06	8,21	7,10	5,55	3,61	3,58
Neuer Regler	2,67	2,17	0,99	0,99	0,85	0,35

Tabelle 7.2: Vergleich der Regler durch Positionierungsversuche mit der ersten Achse an verschiedenen Arbeitspunkten. Blau: Sollbahn, Grün: Originalregler, Rot: neuer Regler

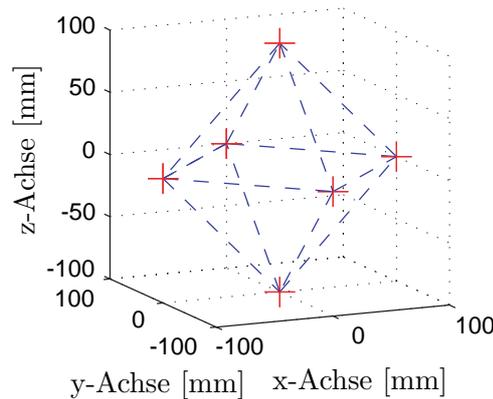


Abbildung 7.6: Positionen und Bewegungen für das Positionieren

dass in den Kriterien gemäß Gleichung (7.2) die kartesische Abweichung von der Sollposition benutzt wird. Diese ist bei gleichen Gelenkwinkelfehlern der Achsen umso größer, je weiter außen der Arbeitspunkt liegt.

Zur Bewertung des Verhaltens bei beliebigen Bewegungen wurden zusätzlich Versuche mit allen Achsen durchgeführt. Dazu wurden als Sollpositionen die Eckpunkte eines Oktaeder verwendet, womit sich die in Abbildung 7.6 dargestellten Sollpunkte und -bewegungen ergeben. Für diese mehrachsigen Bewegungen ist die Darstellung des Fehlers in nur einer Bewegungsrichtung nicht mehr sinnvoll. Statt dessen wird in Abbildung 7.7 zum Vergleich der Regler für einen Teil der Positionen die absolute Abweichung vom Zielpunkt aufgetragen. Werden alle Positionierungsvorgänge mit den Kriterien aus Gleichung (7.2) bewertet dann ergeben sich, aufgeteilt nach den Ebenen in denen die Bewegungen stattfinden, die Kriterien gemäß Tabelle 7.3. Man erkennt, dass die Kriterien mit dem Originalregler insbesondere dann hoch sind, wenn eine Bewegung in y-Richtung, d. h. unter Verwendung der ersten Achse vorliegt. Der neue Regler zeigt dagegen gleichmäßigere Kriterien welche im Mittel bei etwa 30 % der Originalwerte liegen.

	xy-Ebene		xz-Ebene		yz-Ebene	
	$E_{P,2}$	$E_{P,\infty}$	$E_{P,2}$	$E_{P,\infty}$	$E_{P,2}$	$E_{P,\infty}$
Originalregler	13,76	5,51	7,83	3,61	12,83	5,92
Neuer Regler	3,29	1,70	3,08	1,14	2,48	1,31

Tabelle 7.3: Vergleich der Regler durch Positionierungsversuche mit allen Achsen

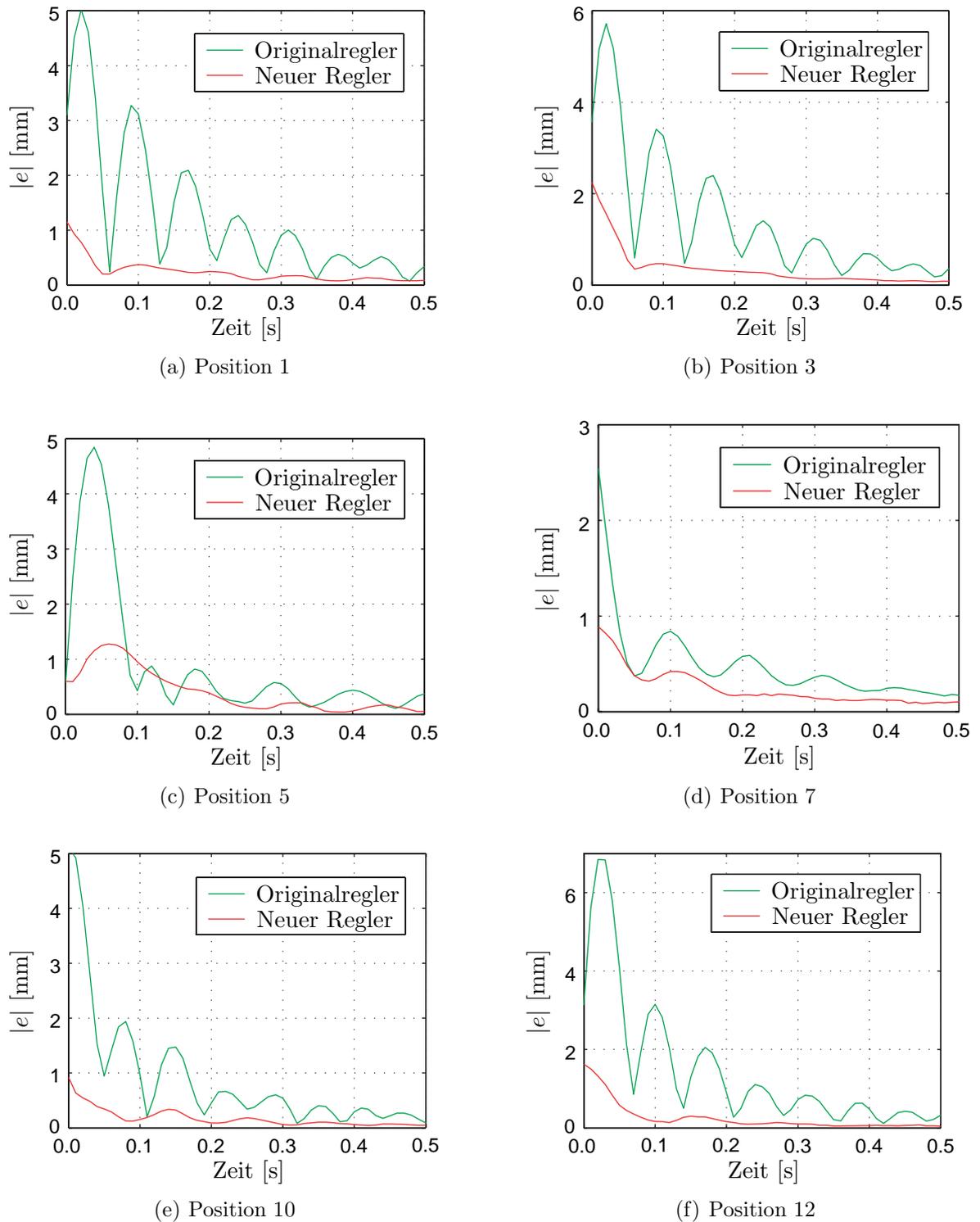


Abbildung 7.7: Vergleich des Positionierverhaltens mit allen Achsen bei unterschiedlichen Bewegungen

Neben der durch die Kriterien quantifizierten Verbesserung des Positionierverhaltens ist zusätzlich anhand von Abbildung 7.5 und Abbildung 7.7 auch eine qualitative Verbesserung der Dämpfung und reduziertes Überschwingen erkennbar. Insgesamt zeigt der neue Regler damit ein deutlich verbessertes Positionierverhalten.

7.2.2 Bahnfahren

Zur Bewertung der Bahntreue werden Geraden und Kreise in verschiedenen Raumrichtungen vorgegeben, welche mit möglichst hoher Bahngeschwindigkeit und -beschleunigung abgefahren werden. Auch hier werden die Bewegungen von der normalen Bahnplanung des Roboters erzeugt. Die Sollbahnen für diese Versuche sind in Abbildung 7.8 dargestellt. Dabei wur-

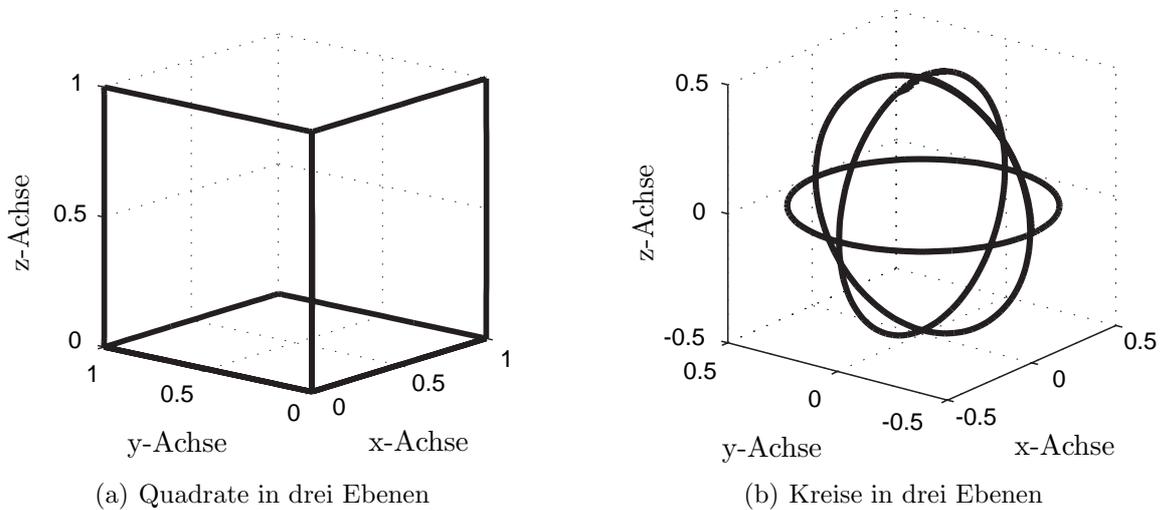


Abbildung 7.8: Testbahnen zur Bewertung der Bahntreue

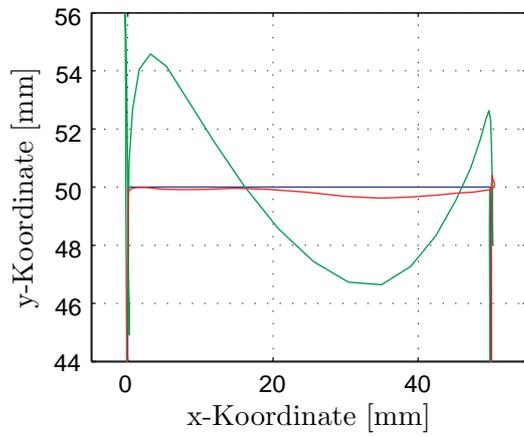
den durch Skalierung der Sollbahnen verschiedene Kantenlängen l bzw. Kreisdurchmesser d erzeugt. Die erreichten Bahngeschwindigkeiten liegen je nach Größe der Bahn zwischen 0,5 und 1,5 $\frac{m}{s}$. Einige typische Ausschnitte aus den Vergleichsmessungen der Regler sind in den Abbildungen 7.9 und 7.10 zu sehen.

Die Bewertung der Bahngenauigkeit erfolgt gemäß Gleichung (7.1) anhand der 2-Norm und der ∞ -Norm des Bahnfehlers. In Tabelle 7.4 werden die Regler bezüglich ihrer Bahngenauigkeit beim Fahren von Geraden verglichen, wobei die angegebenen Werte jeweils die Mittelwerte der Normen aller einzelnen Bewegungen sind. Dabei lässt sich feststellen, dass die maximalen

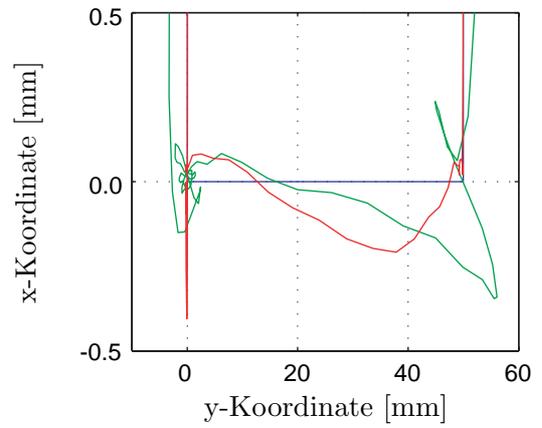
	$l = 50mm$		$l = 200mm$		$l = 400mm$	
	$E_{B,2}$	$E_{B,\infty}$	$E_{B,2}$	$E_{B,\infty}$	$E_{B,2}$	$E_{B,\infty}$
Originalregler	9,80	4,63	8,99	3,12	4,73	1,20
Neuer Regler	1,63	0,57	1,97	0,53	2,04	0,51

Tabelle 7.4: Vergleich der Regler beim Fahren von Geraden unterschiedlicher Längen

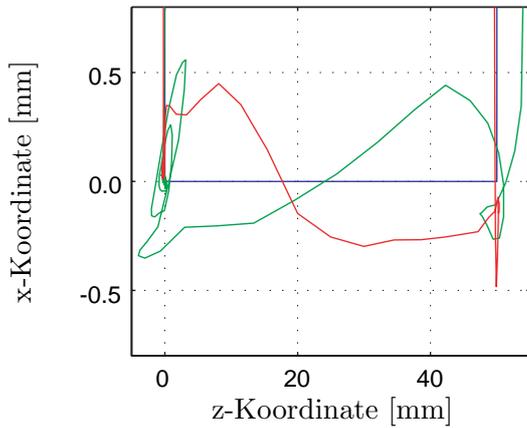
Abweichungen gegenüber dem Originalregler mehr als halbiert werden. Noch deutlicher ist



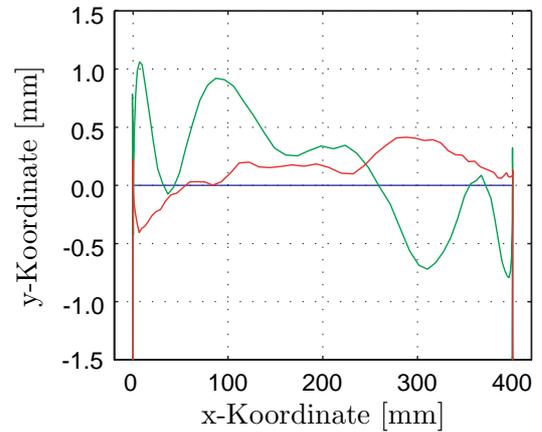
(a) Kurze Gerade in x-Richtung



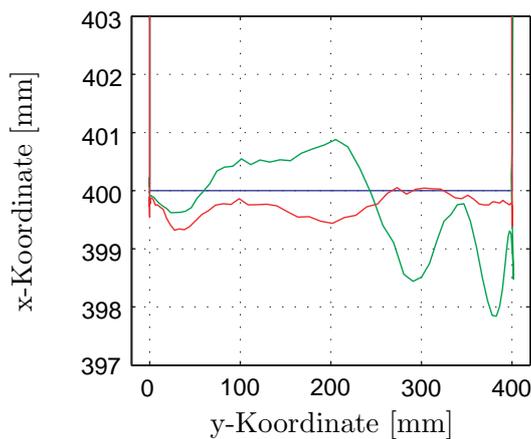
(b) Kurze Gerade in y-Richtung



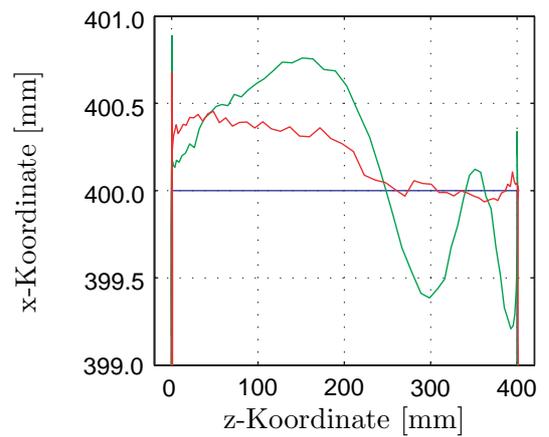
(c) Kurze Gerade in z-Richtung



(d) Lange Gerade in x-Richtung

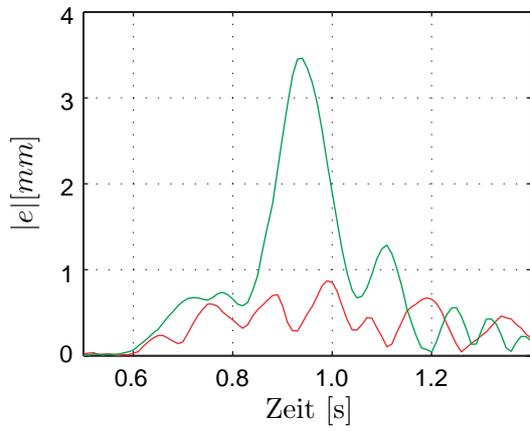


(e) Lange Gerade in y-Richtung

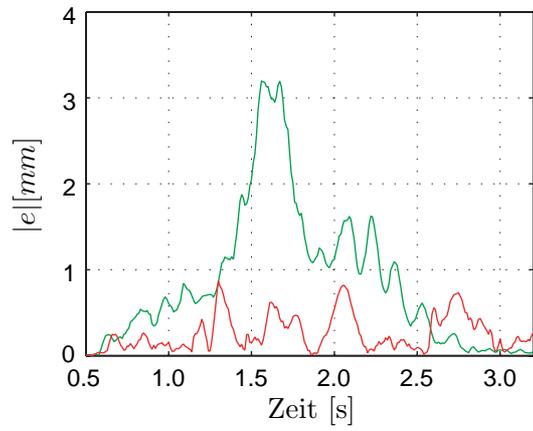


(f) Lange Gerade in z-Richtung

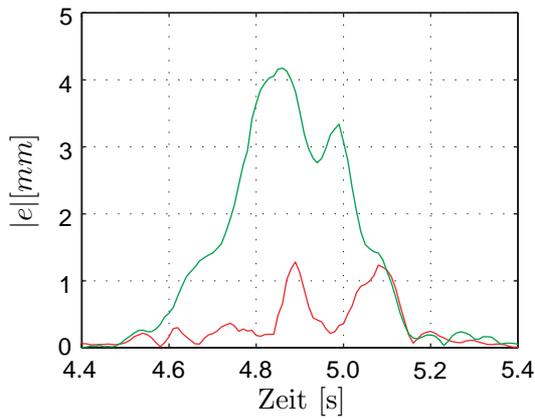
Abbildung 7.9: Vergleich der Regler anhand von Geraden. Blau: Sollbahn, Grün: Originalregler, Rot: neuer Regler



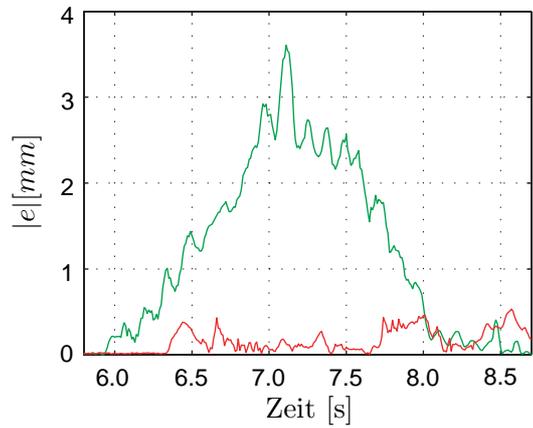
(a) Kleiner Kreis in xy-Ebene



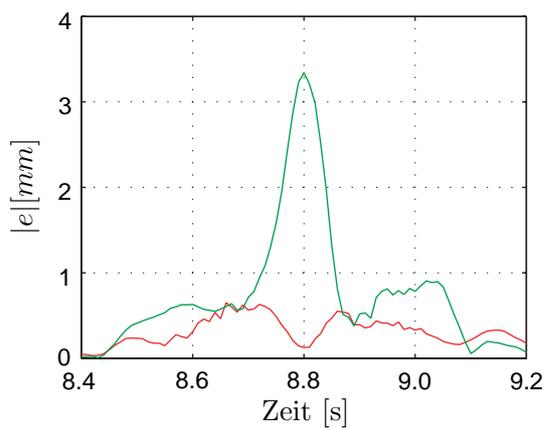
(b) Großer Kreis in xy-Ebene



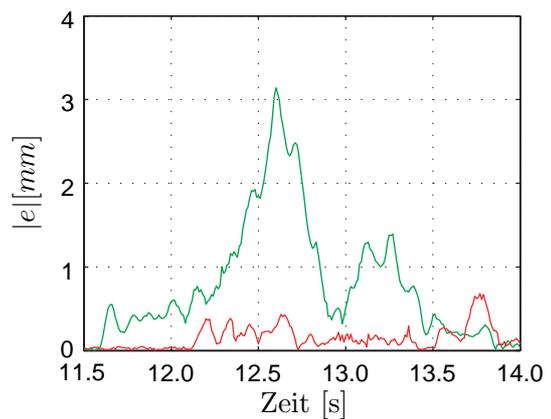
(c) Kleiner Kreis in yz-Ebene



(d) Großer Kreis in yz-Ebene



(e) Kleiner Kreis in xz-Ebene



(f) Großer Kreis in xz-Ebene

Abbildung 7.10: Vergleich der Regler anhand von Kreisen. Grün: Originalregler, Rot: neuer Regler

der Unterschied in der 2-Norm sichtbar. Diese reduziert sich auf weniger als ein Viertel des Wertes mit dem Originalregler.

In Tabelle 7.5 werden die Regler bezüglich der Bahngenauigkeit auf Kreisen gegenübergestellt. Dabei zeigt sich, dass sowohl die maximalen Abweichungen als auch die 2-Normen durch den

	$d = 50mm$		$d = 200mm$		$d = 400mm$	
	$E_{B,2}$	$E_{B,\infty}$	$E_{B,2}$	$E_{B,\infty}$	$E_{B,2}$	$E_{B,\infty}$
Originalregler	13,5	3,66	18,5	3,82	21,0	3,31
Neuer Regler	3,87	0,93	5,09	0,91	6,54	1,00

Tabelle 7.5: Vergleich der Regler beim Fahren von Kreisen unterschiedlicher Durchmesser

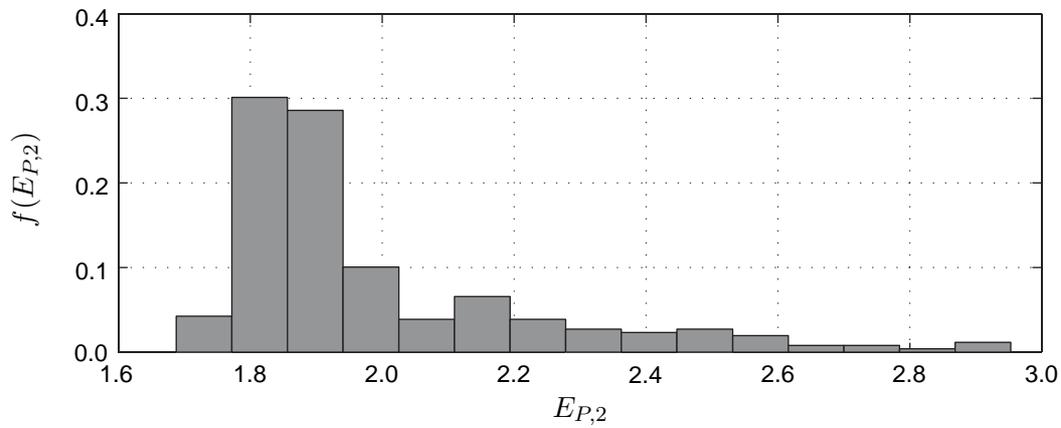
neuen Regler auf unter ein Drittel der Werte mit dem Originalregler sinken. Der neue Regler ist damit auch beim Bahnfahren deutlich besser als der Originalregler.

7.2.3 Robustheit

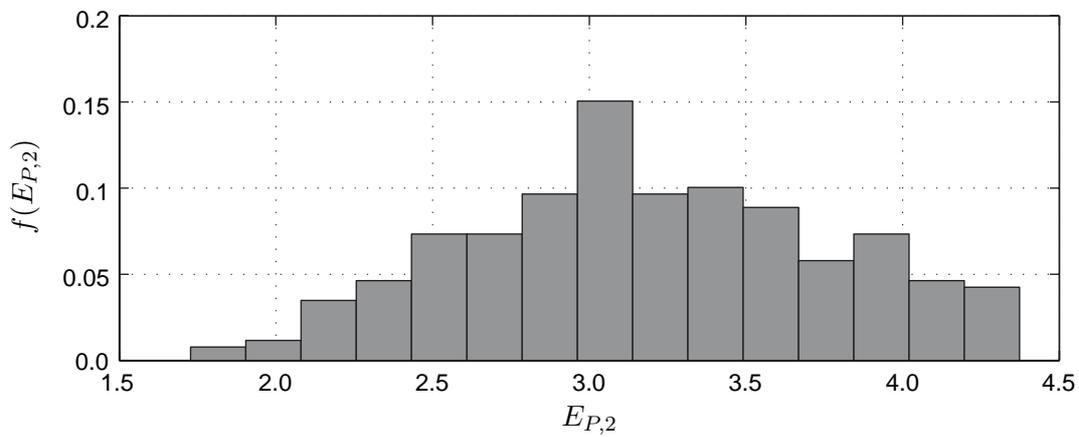
Da der entwickelte Regler ausgiebig Gebrauch vom Modell des Roboters macht stellt sich die Frage nach der Robustheit der Güte des Reglers gegenüber Modellfehlern. Um sicherzustellen, dass bei der Analyse alle praktisch relevanten Effekte berücksichtigt sind, wird die Robustheit in dieser Arbeit experimentell untersucht.

Eine bekannte und häufig verwendete Methode zur Untersuchung solcher Probleme für komplexe Systeme sind Monte-Carlo-Analysen. Bei diesen statistischen Analysen werden zunächst für die Parameterwerte des Systems die Funktionen der Wahrscheinlichkeitsdichten festgelegt. Anschließend wird eine genügend große Anzahl von Versuchen durchgeführt, wobei die Parameterwerte in den Versuchen entsprechend den zuvor festgelegten Wahrscheinlichkeiten gewählt werden. Eine abschließende Analyse liefert dann die Wahrscheinlichkeitsdichte der Versuchsergebnisse für die gewählten Verteilungen der Parameter. Die Grundlagen des Verfahrens sind z. B. in [Fis73] und [HS67] zu finden, eine weitere Beschreibung der Implementierung in [Joo05].

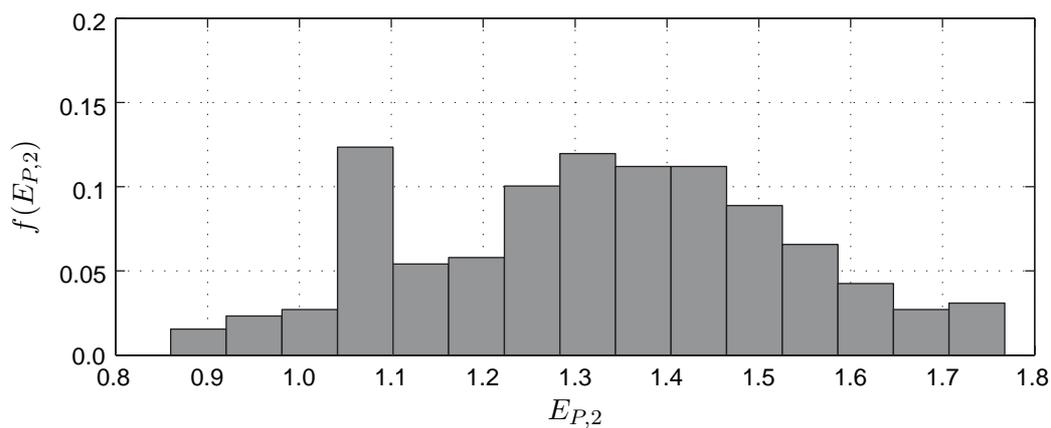
Übertragen auf die Untersuchung der Robustheit des Reglers gegenüber Modellfehlern bedeutet dies, dass Versuche durchgeführt werden in denen die im Regler benutzten Modellparameter nach einer vorgegebenen statistischen Verteilung gegenüber ihren Nominalwerten gestört werden und die Auswirkung auf die resultierende Verteilung der Kriterien zur Bewertung des Fahrverhaltens betrachtet wird. Da das Verfahren auf der Auswertung der Wahrscheinlichkeitsdichte der Kriterien beruht ist eine hinreichend große Zahl an Experimenten notwendig, welche mit der Anzahl der variierten Modellparameter steigt. Um die Anzahl der Versuche zu begrenzen wurden daher nur Einzelachsversuche gefahren, bei denen die entsprechenden Modellparameter der Achse variiert wurden. Dabei wurden Gleichverteilungen verwendet, wobei die Parameterwerte um $\pm 10\%$ von den Nominalwerten abweichen durften. Für die Analyse wurden die bereits in Abschnitt 7.2.1 beschriebenen Positioniersversuche mit einzelnen Achsen 250 mal durchgeführt. Die resultierenden Wahrscheinlichkeitsdichten für Bewegungen mit der ersten Achse sind in Abbildung 7.11 angegeben. Interessant ist dabei vor allem ein Vergleich mit Tabelle 7.2. Man erkennt, dass der neue Regler auch mit Parametern, deren Werte um $\pm 10\%$ von den Nominalwerten abweichen, deutlich geringere Kriterien als der Originalregler aufweist. Er ist somit hinreichend robust gegenüber Modellfehlern.



(a) Aussen



(b) Mitte



(c) Innen

Abbildung 7.11: Wahrscheinlichkeitsdichten des Kriteriums $E_{P,2}$ beim Positionieren mit der ersten Achse an verschiedenen Arbeitspunkten

Kapitel 8

Zusammenfassung und Ausblick

Abschließend wird in diesem Kapitel der Inhalt der vorliegenden Arbeit zusammengefasst sowie die wichtigsten Ergebnisse nochmals hervorgehoben. Zusätzlich wird ein Ausblick auf weiterführende Arbeiten gegeben.

Motiviert wurde die Arbeit durch die Tatsache, dass sich mit dem weiteren Vordringen von Robotern in neue Anwendungen auch die Anforderungen an die erzielbare Genauigkeit und Geschwindigkeit erhöhen. Die Güte, mit der ein Roboter einer vorgegebenen Bahn folgen kann ist daher von zentraler Bedeutung für dessen Einsatzmöglichkeiten, wobei das Positionieren auf einen Zielpunkt als spezielle Bahnaufgabe angesehen wird. Begrenzt wird die erreichbare Güte bei heutigen Robotern vor allem durch die verwendeten Getriebe deren Elastizität einen negativen Einfluss auf die Bahngenauigkeit hat.

Das Ziel dieser Arbeit war daher der Entwurf einer praktikablen Regelungsstruktur für Roboter mit elastischen Gelenken, welche in der Lage ist gute Bahnfolgeeigenschaften sicherzustellen. Bezüglich der Implementierung dieser Regelungsstruktur an realen Robotern mussten Beschränkungen wie eine geringe Anzahl verfügbarer Sensoren und begrenzte Rechenleistung berücksichtigt werden.

Zur Erreichung des Zieles wurde eine Regelungsstruktur mit zwei Freiheitsgraden entworfen. Diese ermöglicht es, einen Großteil der notwendigen Modellauswertungen in der Steuerung auszuführen, womit die Berechnungen nicht mehr in Echtzeit im schnellen Reglertakt erfolgen müssen. Gleichzeitig erlaubt diese Struktur den unabhängigen Entwurf von Steuerung und Regler, was für die Auslegung vorteilhaft ist. Als Steuerung wurde eine Kombination aus der Inversen Dynamik des Roboters und einem Referenzmodell verwendet, wodurch sich mehrere Vorteile ergeben: Zunächst kann durch die Inverse im Idealfall bereits die komplette Roboterdynamik kompensiert werden. Durch die Wahl des Referenzmodells als Filter geeigneter Ordnung können die zur Auswertung der Inversen notwendigen Ableitungen der Sollgröße bestimmt werden. Zusätzlich kann das Referenzmodell im Sinne eines Input-Shapers noch so ausgelegt werden, dass es die Anregung von in der Inversen nicht modellierter Dynamik unterdrückt. Die Güte der Inversen bestimmt auf diese Weise die mögliche Bandbreite der Sollgrößen. Zur Verbesserung der Eigendynamik des Roboters und um Störungen ausgleichen zu können, wurden als Regler achsweise lineare PI-Zustandsregler eingesetzt. Diese eignen sich aufgrund ihrer Leistungsfähigkeit und Einfachheit gut zur Regelung von Antriebssträngen, benötigen aber den kompletten Zustand des Systems. Um den Zustand besser schätzen zu

können, wurden deshalb Beschleunigungssensoren verwendet. Diese haben insbesondere den Vorteil einer einfachen nachträglichen Anbringung am Arm. Da die Roboterdynamik stark vom aktuellen Arbeitspunkt abhängt, wurde zusätzlich ein Gain-Scheduling der Reglerparameter benutzt.

Um die gewünschte Regelungsstruktur verwirklichen zu können, wurde in dieser Arbeit der komplette Reglerentwurf für einen Roboter mit elastischen Gelenken, von der Modellierung und Identifikation des Robotermodells, über Methoden und Werkzeuge zur Modellauswertung bis zur Reglerauslegung, der Optimierung der Reglerparameter und der experimentellen Erprobung behandelt und exemplarisch an einem KUKA KR15/2 Industrieroboter durchgeführt.

Dabei wurde anhand von physikalisch motivierten Teilmodellen eine neue detaillierte Modellvorstellung für Roboter mit elastischen Gelenken entwickelt. In dieser wird sowohl die Dynamik des Mehrkörpersystems als auch nichtlineare Elastizitäten, drehzahl- und lastabhängige Verluste und Drehungleichförmigkeiten der Antriebsstränge berücksichtigt. Basierend darauf wurde ein Modell des KR15/2 aufgestellt und die Modellparameter mit einer Kombination aus speziellen Versuchen zur Messung einzelner Parameter und Optimierungsmethoden identifiziert. Die Ergebnisse zeigen eine sehr gute Übereinstimmung des Modells mit der Realität.

In der gewählten Regelungsstruktur stellt die modellbasierte Steuerung auf Basis der inversen Dynamik das Kernstück dar. Die praktische Berechnung der Inversen von multivariablen, nichtlinearen Systemen wird jedoch in der Literatur kaum behandelt. Aus diesem Grund lag der Schwerpunkt der Arbeit auf Methoden zur Berechnung von inversen Systemen im Allgemeinen und der Invertierung der Roboterdynamik im Besonderen. Dabei wurden zunächst Begriffe aus der nichtlinearen Regelungstechnik wie relativer Grad, interne Dynamik und Flachheit in Bezug auf Ihre Bedeutung für die Systeminvertierung untersucht und bekannte Methoden zur Invertierung skizziert. Aus praktischer Sicht ist der wesentliche Nachteil der bekannten Methoden, dass diese das System in Zustandsform analysieren. In der Arbeit wurde dargestellt, dass die Invertierung auch auf Basis von Systemen, welche als DAE gegeben sind durchgeführt werden kann. Dabei wurde gezeigt, dass bei der Invertierung eines Systems in DAE-Darstellung im Wesentlichen eine andere Interpretation der Gleichungen vorgenommen wird und die Berechnung des inversen Systems auf die Lösung einer DAE mit höherem Index zurückgeführt werden kann. Anschließend wurden bekannte Algorithmen zur Behandlung von DAE-Systemen mit höherem Index vorgestellt, welche eine praktikable und effiziente Berechnung inverser Systeme ermöglichen. Der Ablauf dieser Algorithmen wurde anhand von Beispielen erläutert. Das beschriebene Vorgehen stellt eine neue, leistungsfähige Methode zur Berechnung der Inversen nichtlinearer Systeme dar.

Anschließend wurden die Eigenschaften der Inversen des entwickelten Robotermodells untersucht. Dabei wurde anstatt der üblichen Untersuchung der Zustandsraumdarstellung eine direkte Analyse der Bewegungsgleichungen für eine einzelne Achse und für den gesamten Roboter durchgeführt. Mit Hilfe dieser Analyse wurden nicht invertierbare Teile des Modells erkannt und ein neues, approximiertes Modell zur Invertierung aufgestellt. Zudem konnten weitere Modelleigenschaften ermittelt werden. Insbesondere wurde ein neuer Algorithmus zur Bestimmung der Ordnungen der notwendigen Ableitungen der Bewegungsgleichungen für RLFJ-Roboter bei beliebiger Anbringung der Motoren am Arm gefunden. Für die Inverse des approximierten Modells wurde die Input-to-State-Stabilität untersucht und Bedingungen für die Stabilität des inversen Robotermodells angegeben. Schließlich wurde das Modell der inversen Dynamik des KR15/2 mit den beschriebenen Methoden unter Verwendung des Modelica

Simulationswerkzeuges Dymola aufgestellt und die Berechnung für eine Beispieltrajektorie dargestellt.

Um die angestrebte Reglerstruktur realisieren zu können, wurde im Anschluss ein Beobachter für die Zustände des Systems entworfen. Dieser benutzt zusätzlich zur Messung der Motorpositionen am Arm verteilte Beschleunigungssensoren. Es wurde gezeigt, dass es bei geeigneter Anbringung der Sensoren unter Verwendung von deren Messwerten möglich ist, die Komplexität des Modells für die Zustandsschätzung deutlich zu vereinfachen. Dabei führt die gezeigte Verwendung der Beschleunigungssensoren auf weitestgehend entkoppelte Modelle für die einzelnen Achsen. Insbesondere entfällt die Berechnung der Dynamik des Mehrkörpersystems zugunsten einfacherer, kinematischer Transformationen der Messwerte mit einem deutlich geringeren Bedarf an Rechenzeit. Zusätzlich werden Störungen, z. B. durch Prozesskräfte, aufgrund der verwendeten Beschleunigungssensoren mit gemessen und in der Zustandsschätzung berücksichtigt. Der entworfene Beobachter wurde anschließend für die ersten drei Achsen des KR15/2 ausgelegt und evaluiert. Dabei zeigte sich im Vergleich mit Messungen, dass der Beobachter in der Lage ist, die Zustände gut wiederzugeben.

Mit Hilfe von linearen Gelenkmodellen wurde anschließend der Regler entworfen. Zur Erreichung einer guten Eigendynamik des geregelten Systems, bei gleichzeitiger stationärer Genauigkeit der Motorposition, wurde ein PI-Zustandsregler benutzt. Dabei wurde gezeigt, dass dieser Regler in der Lage ist, die stationäre Genauigkeit der Motorposition sowohl bei externen Störungen als auch bei einer Drift der Messwerte der Beschleunigungssensoren sicherzustellen. Gleichzeitig bietet er für die zum Entwurf benutzten linearen Achsmodelle bezüglich der Lage der Pole volle Entwurfsfreiheit. Um zusätzlich die Veränderung der Roboterdynamik über den Arbeitsraum zu berücksichtigen, wurde ein Gain-Scheduling für die Reglerparameter durchgeführt. Dabei werden anhand vorgegebener Steifigkeiten und Dämpfungen der geregelten Gelenke die Pole der Gelenkmodelle indirekt in Abhängigkeit des Arbeitspunktes berechnet. Versuche zur Eigendynamik der geregelten Achsen unter Verwendung von externen Störungen zeigen, dass es mit Hilfe der Regler möglich ist, die Steifigkeit und Dämpfung der geregelten Gelenke in einem großen Bereich zu variieren. Insbesondere ist es möglich, eine schnelle, gut gedämpfte Eigendynamik der Gelenke zu erreichen.

Im Anschluss an die beschriebene getrennte Auslegung einzelner Teile der Regelungsstruktur erfolgte die experimentelle Erprobung am KR15/2 für Positionier- und Bahnaufgaben. Um die Güte des geregelten Roboters bewerten zu können, wurden Kriterien für die Bahnabweichungen und das Positionierverhalten aufgestellt. Basierend auf diesen Kriterien erfolgten Hardware-in-the-loop-Optimierungen zur Abstimmung der Regelungsstruktur am realen Roboter. Zur Bewertung der Güte des Reglers wurde der neue Regler in einer Vielzahl von Versuchen mit dem Originalregler des Roboters verglichen. Dabei wurden insbesondere ein- und mehrachsige Positioniervorgänge an verschiedenen Arbeitspunkten sowie Bahnbewegungen unterschiedlicher Form und Größe durchgeführt.

Zusammenfassend reduziert der neue Regler in einem weiten Arbeitsbereich sowohl beim Positionieren als auch beim Bahnfahren, die Bahnabweichungen des Werkzeugflansches auf etwa ein Viertel der Werte des Originalreglers.

Abschließend wurden Monte-Carlo-Analysen zur Untersuchung der Robustheit des neuen Reglers gegenüber Parameterfehlern durchgeführt. Diese deuten auf gute Robustheitseigenschaften hin, da die Streuung der Kriterienwerte des neuen Reglers bei Fehlern in den Modellpa-

rametern von $\pm 10\%$ relativ gering ist und die Kriterien für alle getesteten Parameterkombinationen deutlich kleiner als beim Originalregler sind.

Die in dieser Arbeit entwickelte modellbasierte Regelung von Robotern mit elastischen Gelenken zeichnet sich anhand der bisherigen Ergebnisse durch geringe Fehler beim Bahnfahren, eine gute Schwingungsdämpfung und Robustheit gegenüber Modellfehlern aus. Vorteilhaft aus Sicht der Implementierung ist der geringe Bedarf an Rechenzeit sowie die einfache Integrationsmöglichkeit von Beschleunigungssensoren am Roboter.

Ein Nachteil der gewählten Regelung aus theoretischer Sicht ist, dass sie auf lokalen Stabilitätsaussagen beruht. Ein globaler Stabilitätsbeweis für das gesamte System unter Verwendung des kompletten Modells stellt daher noch ein offenes Problem dar. Ob Stabilitätsbedingungen aus einem solchen Beweis eine weitere Verbesserung des Reglers ermöglichen, ist jedoch nicht sicher. Arbeiten mit dem Ziel einer weiteren Verbesserung des dynamischen Verhaltens könnten dagegen beim zur Invertierung verwendeten Robotermodell angreifen. Interessant ist in diesem Zusammenhang zum Einen die Erweiterung auf online lernende Modelle um z. B. die von Temperatur und Verschleißzustand abhängigen Reibmodelle automatisch nachzuführen. Zum Anderen erscheint die Berücksichtigung von Elastizitäten in den Gliedern, welche insbesondere bei Robotern mit großem Arbeitsraum zunehmen, erfolgversprechend. Allerdings kann hier bei der Invertierung eine instabile interne Dynamik auftreten, für deren Behandlung geeignete Wege gefunden werden müssen. Zudem bieten sich, wie bereits beschrieben, für die Auslegung des Reglers, abhängig von den gewünschten Eigenschaften die verschiedensten Verfahren an. An dieser Stelle erscheinen für weitere Arbeiten vor allem Verfahren als attraktiv, welche aufgrund des vereinfachten Entwurfmodells entstandene Modellunsicherheiten bei der Auslegung berücksichtigen.

Anhang A

Mathematische Grundlagen

A.1 Block-Lower-Triangular-Transformation

Bei der objektorientierten Modellierung entstehen durch die Gleichungen in den Objekten und die Verbindungsgleichungen große, aber normalerweise dünn besetzte Gleichungssysteme. Zur effizienten Lösung eines solchen Systems ist es sinnvoll das Gleichungssystem möglichst explizit nach den Unbekannten aufzulösen. Das nötige Vorgehen ist z. B. in [Ott00] beschrieben und wird hier kurz wiedergegeben.

Die Gleichungsstruktur sei durch die Inzidenzmatrix S_I wiedergegeben. Dabei zeigt $S_{I,i,k}$ an, ob die k -te Variable in der i -ten Gleichung auftritt. Das Ziel ist nun die Inzidenzmatrix S_I mit den Permutationsmatrizen P und Q so auf untere Dreiecksform zu transformieren, dass die Diagonalblöcke $B_{i,i}$ minimale Dimension besitzen.

$$PS_IQ = \begin{pmatrix} B_{11} & 0 & \dots & 0 \\ B_{21} & B_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ B_{m1} & B_{m2} & \dots & B_{mm} \end{pmatrix}$$

Diese Transformation wird als *Block-Lower-Triangular* (BLT) Transformation bezeichnet. Der zugehörige Algorithmus zur Transformation kann z. B. aus [DER86] entnommen werden.

A.2 Brunovski-Normalform

Ein MIMO System in Brunovski-Normalform besteht aus mehreren entkoppelten Integratorketten

$$\begin{aligned} \dot{\xi}_j^i &= \xi_{j+1}^i & j &= 1, \dots, n_i \\ \dot{\xi}_{n_i}^i &= u_i \end{aligned}$$

und entsteht z. B. bei der Zustandslinearisierung eines Systems durch Feedback.

A.3 Input-to-State Stability

Die Bezeichnung und Untersuchung der Input-to-State Stability (ISS) geht zurück auf E. D. Sontag [Son89] und ist in ihren Grundzügen auch in [Isi99] dargestellt. Die folgende, kurze Übersicht beruht vor allem auf [Isi99].

Untersucht werden nichtlineare Systeme der Form

$$\dot{x} = f(x, u) \quad (\text{A.1})$$

mit dem Zustand $x \in \mathbb{R}^n$ und dem Eingang $u \in \mathbb{R}^m$, wobei $f(0, 0) = 0$ gilt und $f(x, u)$ lokal Lipschitzstetig auf $\mathbb{R}^n \times \mathbb{R}^m$ ist. Der Eingang kann eine beliebige stückweise-kontinuierliche, beschränkte Funktion sein. Die Menge all dieser Funktionen bewertet mit der Supremum-Norm

$$\|u(\cdot)\|_\infty = \sup_{t \geq 0} \|u(t)\| \quad (\text{A.2})$$

wird mit L_∞^m bezeichnet.

Bei der Untersuchung der ISS spielen Vergleichsfunktionen eine zentrale Rolle. Deshalb werden zunächst noch die dafür benötigten Definitionen angegeben.

Definition A.1 (Funktionenklasse \mathcal{K}).

Eine kontinuierliche Funktion $\alpha : [0, a) \rightarrow [0, \infty)$ wird als zur Klasse \mathcal{K} gehörend bezeichnet, wenn sie streng monoton steigend ist und $\alpha(0) = 0$ ist. Wenn $a = \infty$ und $\lim_{r \rightarrow \infty} \alpha(r) = \infty$ gilt, wird die Funktion als zur Klasse \mathcal{K}_∞ gehörend bezeichnet.

Definition A.2 (Funktionenklasse \mathcal{KL}).

Eine kontinuierliche Funktion $\beta : [0, a) \times [0, \infty) \rightarrow [0, \infty)$ wird als zur Klasse \mathcal{KL} gehörend bezeichnet, wenn für jedes konstante s die Funktion

$$\begin{aligned} \alpha : [0, a) &\rightarrow [0, \infty) \\ r &\mapsto \beta(r, s) \end{aligned}$$

zur Klasse \mathcal{K} gehört und für jedes konstante r die Funktion

$$\begin{aligned} \phi : [0, \infty) &\rightarrow [0, \infty) \\ s &\mapsto \beta(r, s) \end{aligned}$$

monoton fallend ist und $\lim_{s \rightarrow \infty} \phi(s) = 0$.

Basierend darauf kann nun die Definition für ISS angegeben werden.

Definition A.3 (Input-to-state Stabilität).

Das System $\dot{x} = f(x, u)$ wird als input-to-state-stabil bezeichnet, wenn eine \mathcal{KL} Funktion $\beta(\cdot, \cdot)$ und eine \mathcal{K} Verstärkungsfunktion $\gamma(\cdot)$ existieren, so dass für jeden Eingang $u(\cdot) \in L_\infty^m$ und für jedes $x^0 \in \mathbb{R}^n$ die Lösung $x(t)$ mit dem Anfangszustand $x(0) = x^0$ die Bedingung

$$\|x(t)\| \leq \beta(\|x^0\|, t) + \gamma(\|u(\cdot)\|_\infty)$$

für alle $t \geq 0$ erfüllt ist.

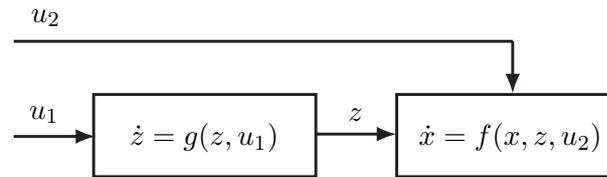


Abbildung A.1: Kaskade mit externem Eingang für beide Systeme

In [Isi99] wird gezeigt, dass ein solches System nur dann ISS ist, wenn eine sogenannte ISS-Lyapunov-Funktion für das System existiert.

Definition A.4 (ISS-Lyapunov-Funktion).

Eine C^1 Funktion $V : \mathbb{R}^n \rightarrow \mathbb{R}$ wird als ISS-Lyapunov-Funktion für das System $\dot{x} = f(x, u)$ bezeichnet, wenn \mathcal{K}_∞ Funktionen $\underline{\alpha}(\cdot)$, $\bar{\alpha}(\cdot)$, $\alpha(\cdot)$ und eine \mathcal{K} Funktion $\chi(\cdot)$ existieren so dass

$$\underline{\alpha}(\|x\|) \leq V(x) \leq \bar{\alpha}(\|x\|) \quad \forall x \in \mathbb{R}^n$$

und

$$\|x\| \geq \chi(\|u\|) \Rightarrow \frac{\partial V}{\partial x} f(x, u) \leq -\alpha(\|x\|) \quad \forall x \in \mathbb{R}^n$$

erfüllt sind.

Daneben existieren noch weitere Möglichkeiten zu prüfen, ob eine Funktion eine ISS-Lyapunov-Funktion für das System ist oder ob das System input-to-state-stabil ist. Diese werden hier jedoch nicht benutzt und daher auch nicht angegeben.

Benötigt werden dagegen die Eigenschaften von kaskadierten ISS-Systemen. Wie in [Isi99] gezeigt ist das System

$$\dot{x} = f(x, z) \tag{A.3a}$$

$$\dot{z} = g(z, u) \tag{A.3b}$$

dann ISS, wenn die Systeme (A.3a) und (A.3b) ISS sind. Diese Eigenschaft kann auch anschaulich nachvollzogen werden. Wenn z aufgrund der ISS von (A.3b) beschränkt bleibt, dann ist der Eingang von (A.3a) ebenfalls beschränkt. Mit der geforderten ISS von (A.3a) bleibt auch x beschränkt. Damit ist der komplette Zustand (x, z) beschränkt.

In dieser Arbeit wird zusätzlich die ISS von Kaskaden benötigt, bei denen beide Systeme einen externen Eingang aufweisen. Diese Konstellation ist in Abbildung A.1 dargestellt. Die Gleichungen des Systems sind gegenüber (A.3) um u_2 erweitert.

$$\dot{x} = f(x, z, u_2) \tag{A.4a}$$

$$\dot{z} = g(z, u_1) \tag{A.4b}$$

Auch für dieses System kann eine ähnliche Argumentation wie im vorigen Fall verwendet werden: Ist das System (A.4a) ISS, dann bleibt z beschränkt. Wenn nun zusätzlich (A.4b) ISS bezüglich des Eingangs (z, u_2) ist, dann bleibt auch x beschränkt. Somit stellt auch in diesem Fall die ISS der einzelnen Systeme bezüglich ihrer Eingänge die ISS des Gesamtsystems sicher.

A.4 Lie-Ableitungen

In dieser Arbeit werden Lie-Ableitungen vor allem benutzt, weil sie eine kompakte Darstellung der Ableitungen von Systemausgängen ermöglichen. Die folgende Darstellung orientiert sich an [SL91].

Definition A.5 (Lie-Ableitung).

Sei $h : \mathbb{R}^n \rightarrow \mathbb{R}$ eine stetige skalare Funktion und $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ ein stetiges Vektorfeld auf \mathbb{R}^n . Dann ist die Lie-Ableitung von h bezüglich f eine skalare Funktion welche durch

$$L_f h = \nabla h f$$

definiert ist.

Die Lie-Ableitung ist somit die Richtungsableitung von h in Richtung des Vektors f .

Lie-Ableitungen höherer Ordnung sind rekursiv durch

$$\begin{aligned} L_f^0 h &= h \\ L_f^i h &= L_f(L_f^{i-1} h) = \nabla(L_f^{i-1} h) f \quad i = 1, 2, \dots \end{aligned}$$

definiert. In der gleichen Weise kann mit einem weiteren Vektorfeld g die skalare Funktion $L_g L_f h$ definiert werden:

$$L_g L_f h = \nabla(L_f h) g$$

Für Systeme der Form

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned}$$

wie sie in Kapitel 4 behandelt werden, können damit die Ableitungen des Ausgangs kompakt als

$$\begin{aligned} \dot{y} &= \frac{\partial h(x)}{\partial x} \dot{x} = L_f h(x) + L_g h(x)u \\ \ddot{y} &= \frac{\partial(L_f h(x) + L_g h(x)u)}{\partial x} \dot{x} = L_f^2 h(x) + L_g L_f h(x)u \end{aligned}$$

dargestellt werden.

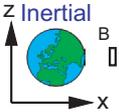
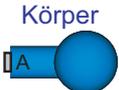
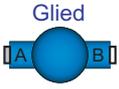
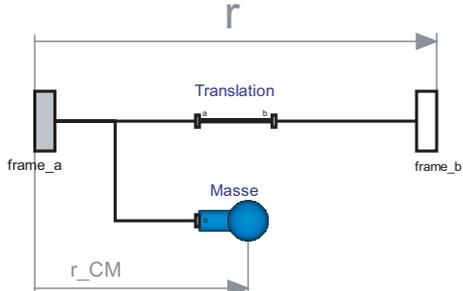
Anhang B

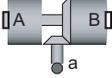
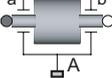
Modelica Bibliothek

In diesem Anhang werden die zur Modellierung des Roboters erstellten Modelica-Objekte mit ihren Symbolen, Gleichungen und Konnektoren dargestellt. Dabei werden nur die grundlegenden Gleichungen der Modelle angegeben. In den tatsächlichen Modelica-Modellen enthaltener zusätzlicher Code, z. B. zur Initialisierung, wird der Übersichtlichkeit halber nicht angegeben.

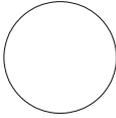
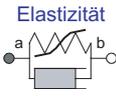
B.1 Mehrkörpersystem

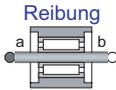
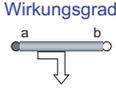
Symbol	Beschreibung des Teilmodells
	Mehrkörpersystem Frame A Konnektor für Mehrkörpersysteme. Enthält die Potentialvariablen <ul style="list-style-type: none">• 0r_A Vektor vom Inertialsystem zum Ursprung des Frames A, ausgedrückt in Inertialkoordinaten• ${}^0R^A$ Rotationsmatrix vom Frame A zum Inertialsystem und die Flussvariablen <ul style="list-style-type: none">• ${}^A f_A$ Schnittkräfte am Frame A im Koordinatensystem von Frame A• ${}^A \tau_A$ Schnittmomente am Frame A im Koordinatensystem von Frame A
	Mehrkörpersystem Frame B Wie Mehrkörpersystem Frame A. Der Konnektor dient der einfacheren Unterscheidung der Frames bei Objekten mit zwei Konnektoren und enthält <ul style="list-style-type: none">• 0r_B Vektor vom Inertialsystem zum Ursprung des Frames B, ausgedrückt in Inertialkoordinaten• ${}^0R^B$ Rotationsmatrix vom Frame B zum Inertialsystem• ${}^B f_B$ Schnittkräfte am Frame B im Koordinatensystem von Frame B• ${}^B \tau_B$ Schnittmomente am Frame B im Koordinatensystem von Frame B

Symbol	Beschreibung des Teilmodells
	<p>Inertialsystem</p> <p>Modell zur Festlegung des Bezugssystems. Zusätzlich wird die Gravitation entsprechend dem Äquivalenzprinzip über eine Beschleunigung des Bezugssystems berücksichtigt. Dafür wird im Inertialsystem aus einem normalisierten Richtungsvektor 0z und der Erdbeschleunigungskonstante g der Vektor der Erdbeschleunigung 0g berechnet und den anderen Objekten zur Verfügung gestellt.</p> ${}^0r_B = 0_{3 \times 1}$ ${}^0R^B = I_{3 \times 3}$ ${}^0g = {}^0zg$
	<p>Stab</p> <p>Diese Klasse modelliert einen masselosen Stab. Der Abstand zwischen den Frames A und B ist durch den Parameter ${}^A r_{AB}$ gegeben. Die Relationen zwischen den Frames ergeben sich damit zu</p> ${}^0r_B = {}^0r_A + {}^0R^{AA} r_{AB}$ ${}^0R^B = {}^0R^A$ $0_{3 \times 1} = {}^A f_A + {}^A R^{BB} f_B$ $0_{3 \times 1} = {}^A \tau_A + {}^A R^{BB} \tau_B + {}^A r_{AB} \times {}^A R^{BB} f_B.$
	<p>Körper</p> <p>Dieser Klasse modelliert einen starren Körper mit der Masse m, dem Vektor vom Frame A zum Massenschwerpunkt ${}^A r_{CM}$ und dem Trägheitstensor ${}^A I$. Der Zusammenhang zwischen den Kräften und Momenten am Frame A mit der Bewegung des Frames A ergibt sich zu</p> ${}^A f_A = m ({}^A R^0 ({}^0 \dot{v}_A - {}^0 g) + {}^A \dot{\omega}_A \times {}^A r_{CM} + {}^A \omega_A \times ({}^A \omega_A \times {}^A r_{CM}))$ ${}^A \tau_A = {}^A I^A \dot{\omega}_A + {}^A \omega_A \times {}^A I^A \omega_A + {}^A r_{CM} \times {}^A f_A.$ <p>${}^A \omega_A$ kann dabei aus der Rotationsmatrix bestimmt werden.</p>
	<p>Basisklasse für Roboterglieder</p> <p>Dieser Klasse enthält ein allgemeines Modell eines Robotergliedes mit zwei Frames, wobei die Abmessungen und Masseneigenschaften berücksichtigt werden. Sie wird, wie im Bild dargestellt, aus einem Stab und einem Körper gebildet.</p> 

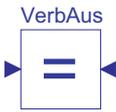
Symbol	Beschreibung des Teilmodells
<p data-bbox="268 360 389 389">Drehgelenk</p> 	<p data-bbox="416 331 587 365">Drehgelenk</p> <p data-bbox="416 371 1410 472">Modell eines Drehgelenkes für MKS mit einem Freiheitsgrad und vorgegebener Drehachse welche durch den normalisierten Vektor ${}^A z$ gegeben ist. Das Gelenk wird dargestellt durch die Gleichungen</p> $ \begin{aligned} {}^B R^A &= {}^B R^A({}^A z, \varphi_a) \\ {}^0 R^A &= {}^0 R^{BB} R^A \\ {}^0 r_A &= {}^0 r_B \\ {}^B f_B &= -{}^B R^{AA} f_A \\ {}^B \tau_B &= -{}^B R^{AA} \tau_A \\ \tau_a &= -{}^B \tau_B {}^A z. \end{aligned} $
<p data-bbox="280 857 376 887">Gyrostat</p> 	<p data-bbox="416 824 552 857">Gyrostat</p> <p data-bbox="416 864 1410 1149">Diese Klasse modelliert die Wechselwirkung zwischen dem sich drehenden Motor und dem MKS. Dabei hat der Motor die Drehachse ${}^A z$ und das Trägheitsmoment um diese Achse J. Wie in den Annahmen bei der Herleitung des Mehrkörpermodells mit dem Lagrange-Ansatz sind auch hier die Masse und die Trägheiten um andere Achsen als die Drehachse nicht enthalten, sondern werden bei den Gliedern berücksichtigt. Diese effiziente Methode zur Modellierung von Antriebselementen auf Mehrkörpersystemen ist ausführlicher in [Ott95] beschrieben.</p> $ \begin{aligned} \varphi_a &= \varphi_b \\ J \ddot{\varphi}_a &= \tau_a + \tau_b - {}^A z {}^A \dot{\omega}_A \\ {}^A \tau_A &= {}^A z \ddot{\varphi}_a + ({}^A \omega_A \times {}^A z) \dot{\varphi}_a \\ {}^A f_A &= 0_{3 \times 1} \end{aligned} $

B.2 Antriebsstrang

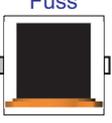
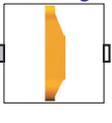
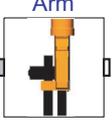
Symbol	Beschreibung des Teilmodells
	<p>Rotatorischer Flansch a Konnektor für 1D rotatorische Antriebselemente. Enthält die Potentialvariable</p> <ul style="list-style-type: none"> • φ_a Absoluter Drehwinkel am Flansch a <p>und die Flussvariable</p> <ul style="list-style-type: none"> • τ_a Schnittmoment im Flansch a
	<p>Rotatorischer Flansch b Wie der rotatorische Flansch a. Der zusätzliche Konnektor dient der Unterscheidung der Flansche bei Objekten mit zwei Konnektoren und enthält die Variablen</p> <ul style="list-style-type: none"> • φ_b Absoluter Drehwinkel am Flansch b • τ_b Schnittmoment im Flansch b
	<p>Basisklasse für Elastizität Modelliert die Nachgiebigkeit im Getriebe entsprechend Gleichung (2.37).</p> $\varphi_\Delta = \varphi_b - \varphi_a$ $\tau_b = k(\varphi_\Delta) + d\dot{\varphi}_\Delta$ $\tau_a = -\tau_b$
	<p>Übersetzung In dieser Klasse ist die positionsabhängige Übersetzung mit den Parametern u_0, n_u, \hat{u}, \tilde{u} und \bar{u} entsprechend Gleichung (2.36) modelliert.</p> $\mathbf{u} = \mathbf{u}_0 + \sum_{j=1}^{n_u} \hat{u}_j \cos(\tilde{u}_j \varphi_b + \bar{u}_j)$ $\varphi_a = u_0 \varphi_b + \sum_{j=1}^{n_u} \frac{\hat{u}_j}{\tilde{u}_j} \sin(\tilde{u}_j \varphi_b + \bar{u}_j) + u_C$ $\tau_a = -\frac{\tau_b}{u}$

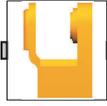
Symbol	Beschreibung des Teilmodells
	<p>Basisklasse für die Lagerreibung Modelliert die geschwindigkeitsabhängige Reibung entsprechend Gleichung (2.42):</p> $\begin{aligned} \varphi_a &= \varphi_b \\ \tau_R &= \tau_a + \tau_b \\ \tau_R &= \begin{cases} \tau(\dot{\varphi}_b) & \text{für } \dot{\varphi}_b \neq 0 \\ \tau_a + \tau_b & \text{für } \dot{\varphi}_b = 0 \wedge \tau_a + \tau_b < \tau_H \\ \tau_H \text{sgn}(\tau_e) & \text{sonst} \end{cases} \end{aligned}$ <p>Das Modell führt aufgrund der berücksichtigten Haftreibung zu Schaltvorgängen. Bei Geschwindigkeiten größer Null ergibt sich die Reibkraft als Funktion der Geschwindigkeit, im Stillstand als Funktion des externen Moments. Dies erfordert eine gesonderte Behandlung um effiziente Simulationen sicherzustellen, wofür ein Zustandsautomat eingesetzt wird. Das nötige Vorgehen ist in [OEM99] beschrieben und bildet die Grundlage des Modells.</p>
	<p>Zahnflankenreibung im Getriebe Modell der leistungsabhängigen Verluste im Getriebe, entsprechend Gleichung (2.49).</p> $\begin{aligned} \varphi_a &= \varphi_b \\ \tau_b &= -\eta\tau_a \\ \eta &= \begin{cases} \eta_p & \text{für } \tau_a\omega_a > 0 \\ \frac{1}{\eta_n} & \text{für } \tau_a\omega_a < 0 \\ 1 & \text{für } \tau_a\omega_a = 0 \end{cases} \end{aligned}$ <p>Bei der Verwendung des Modells ist der Punkt $\tau_a\omega_a = 0$ kritisch, da das Modell an diesem Punkt unstetig ist. Um eine effiziente Simulation sicherzustellen ist eine gesonderte Behandlung, wie sie in [PSO02] beschrieben ist, notwendig.</p>

B.3 Invertierung von Modellen

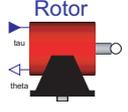
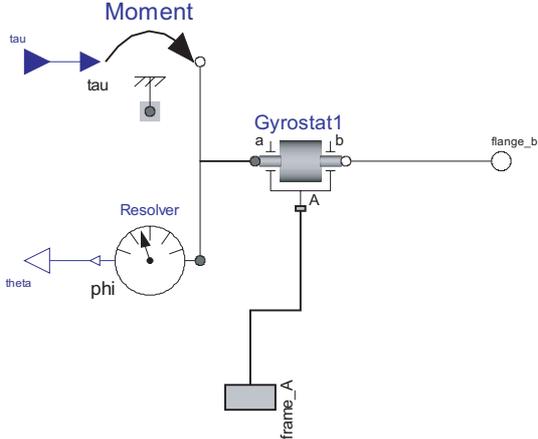
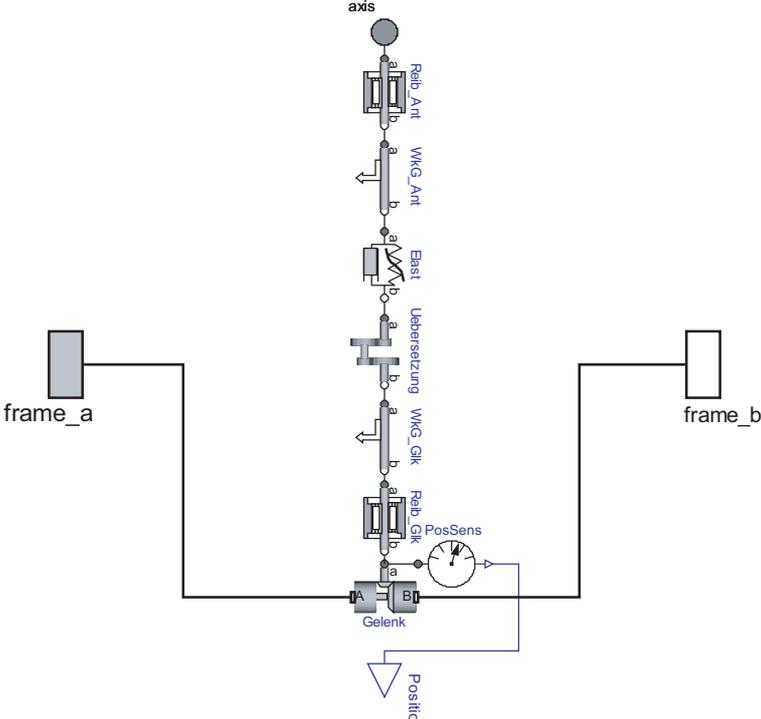
Symbol	Beschreibung des Teilmodells
	<p>Verbinder für Ausgänge Ermöglicht die Verbindung zweier Ausgänge in Blockschaltbild-Semantik.</p> $Input1 = Input2$
	<p>Verbinder für Eingänge Ermöglicht die Verbindung zweier Eingänge in Blockschaltbild-Semantik.</p> $Output1 = Output2$

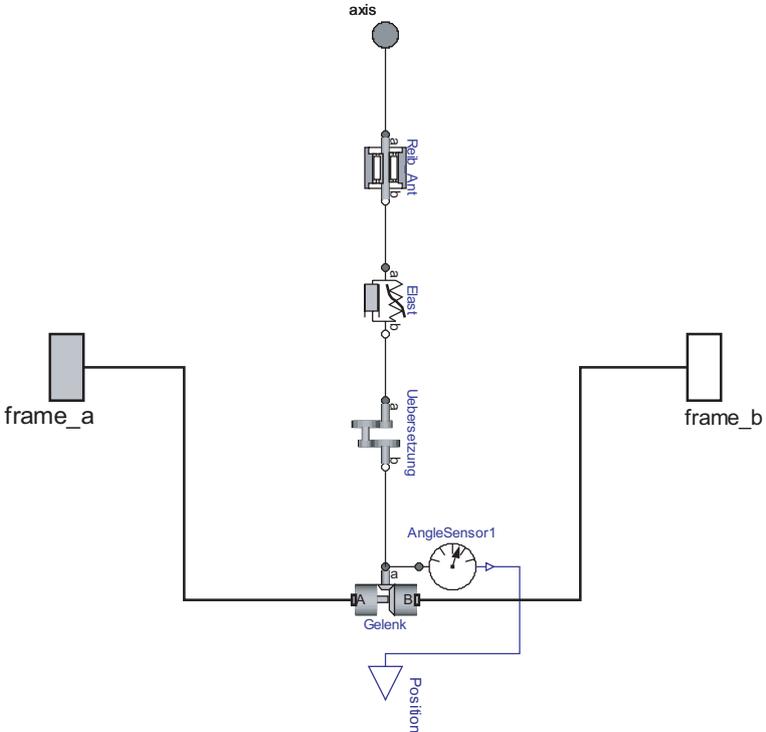
B.4 Glieder des KR15/2

Symbol	Beschreibung des Teilmodells
	<p>Fuss Abgeleitet von der Klasse Glied. Enthält die Parameter des Roboterfusses.</p> ${}^A r_{AB} = (0 \ 0 \ {}^A r_{AB,F_z})^T$ $r_{CM} = 0_{3 \times 1}$ $m = 0$ ${}^A I^A = 0_{3 \times 3}$
	<p>Karussell Abgeleitet von der Klasse Glied. Enthält die Parameter des Karussells.</p> ${}^A r_{AB} = {}^A r_{AB,K}$ $r_{CM} = r_{CM,K}$ $m = 0$ ${}^A I^A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & {}^A I_{zz}^A \end{pmatrix}$
	<p>Schwinge Abgeleitet von der Klasse Glied. Enthält die Parameter der Schwinge.</p> ${}^A r_{AB} = ({}^A r_{AB,S_x} \ 0 \ 0)^T$ $r_{CM} = (r_{CM,S_x} \ 0 \ 0)$ $m = m_S$ ${}^A I^A = {}^A I_S^A$
	<p>Arm Abgeleitet von der Klasse Glied. Enthält die Parameter des Armes.</p> ${}^A r_{AB} = {}^A r_{AB,A}$ $r_{CM} = r_{CM,A}$ $m = m_A$ ${}^A I^A = {}^A I_A^A$

Symbol	Beschreibung des Teilmodells
<p data-bbox="288 360 363 387">Hand 1</p> 	<p data-bbox="416 338 523 365">Hand 1</p> <p data-bbox="416 371 1409 439">Abgeleitet von der Klasse Glied. Enthält die Parameter des ersten Teils der Hand.</p> ${}^A r_{AB} = ({}^A r_{AB,H1x} \ 0 \ 0)^T$ $r_{CM} = r_{CM,H1}$ $m = m_{H1}$ ${}^A I^A = {}^A I_{H1}^A$
<p data-bbox="288 730 363 757">Hand 2</p> 	<p data-bbox="416 707 523 734">Hand 2</p> <p data-bbox="416 741 1409 808">Abgeleitet von der Klasse Glied. Enthält die Parameter des zweiten Teils der Hand.</p> ${}^A r_{AB} = ({}^A r_{AB,H2x} \ 0 \ 0)^T$ $r_{CM} = r_{CM,H2}$ $m = m_{H2}$ ${}^A I^A = {}^A I_{H2}^A$
<p data-bbox="288 1099 363 1126">Hand 3</p> 	<p data-bbox="416 1077 523 1104">Hand 3</p> <p data-bbox="416 1111 1409 1178">Abgeleitet von der Klasse Glied. Enthält die Parameter des dritten Teils der Hand.</p> ${}^A r_{AB} = ({}^A r_{AB,H3x} \ 0 \ 0)^T$ $r_{CM} = (r_{CM,H3x} \ 0 \ 0)$ $m = m_{H3}$ ${}^A I^A = {}^A I_{H3}^A$

B.5 Antriebe des KR15/2

Symbol	Beschreibung des Teilmodells
	<p>Rotormodell Modelliert den Rotor des Motors mit Positionssensor.</p> 
	<p>Getriebe Modelliert die Dynamik des Getriebes wie in Kapitel 2 beschrieben.</p> 

Symbol	Beschreibung des Teilmodells
 <p>Getriebe</p>	<p>Getriebe Hand Modelliert die Dynamik der Getriebe der Handachsen.</p> 

Anhang C

Bezeichnungen

C.1 Typografische Kennzeichnungen

Element einer Matrix	versehen mit $M_{i,j}$	$M_{i,j}$
Zeile einer Matrix	versehen mit $(\cdot)_i$	$(M)_i$
Spalte einer Matrix	versehen mit $(\cdot^T)_i$	$(M^T)_i$
Bezugssystem eines Vektors	versehen mit A	$A f$
Element eines Vektors	versehen mit v_i	v_i
Zeitableitung	versehen mit $\dot{\cdot}$	\dot{x}
Zeitableitung der Ordnung m	versehen mit $(\cdot)^{(m)}$	$y^{(m)}$
Dummy Derivative	versehen mit $'$	x'

C.2 Formelzeichen

δ	Dämpfungsgrad
δ_ν	Gleitreibungsexponent
δ_S	Stribeck-Exponent
η	Wirkungsgrad
θ	Motorposition
Θ	verallgemeinerte Koordinate
λ	Eigenwert
μ	Gleitreibungszahl
ν	differentieller Index einer DAE, Koeffizient der viskosen Reibung
ξ	Zustand bei Darstellung in Normalform
τ	Drehmoment
τ_A	Antriebsstrangmoment
τ_C	Coulomb-Reibmoment
τ_{Ext}	externes Moment
τ_G	Gelenkmoment

τ_H	Haftreibungs-Moment
τ_M	Motormoment
τ_R	Reibmoment
τ_{RA}	Lagerreibmoment des Antriebs
τ_{RG}	Lagerreibmoment des Gelenks
φ	Drehwinkel
φ_Δ	Torsion des Antriebsstranges
$\dot{\varphi}_S$	Stribeck-Geschwindigkeit
ω	Eigenfrequenz
${}^B\omega_A$	Winkelgeschwindigkeit von A im System B
A	Systemmatrix
A_B	Systemmatrix des Beobachters
b	Vektor der Motoranbringungen
B	Eingangsmatrix
c_1	Federkonstante 1
c_2	Federkonstante 2
c_3	Exponent der Federkennlinie
c_{Arm}	Coriolis- und Zentrifugalkräfte auf den Arm
c_{Motor}	Coriolis- und Zentrifugalkräfte auf den Motor
C	Coriolis- und Zentrifugalterme, Ausgangsmatrix
C_R	Ausgangsmatrix für den PI-Regler
d	mechanische Dämpfungskonstante
D	Durchgriffsmatrix
e	Zustandsfehler, Beobachterfehler
E_B	Bahnkriterium
E_P	Positionierkriterium
f	Kraft
f_N	Normalkraft
f_R	Reibkraft
f_{Rest}	nichtlinearer Teil der Fehlerdifferentialgleichung des Beobachters
g	Erdbeschleunigung
$g(q)$	Gravitationsvektor
$H(s)$	Übertragungsfunktion
H_A	Wirkungsgradmatrix der Antriebe
H_G	Wirkungsgradmatrix der Gelenke
I	Motorstrom
${}^B I_A$	Trägheitstensor von A im System B
J	Massenmatrix der Motoren
J_a	Armträgheitsmoment
J_m	Motorträgheitsmoment
k, K	Reglerverstärkung, Verstärkungsmatrix
$k(\varphi_\Delta)$	Federkennlinie
k_T	k_T -Faktor des Motors
l, L	Beobachterverstärkung, Beobachterverstärkungsmatrix
\mathcal{L}	Lagrange-Funktion
m	Masse
m_R	Rotormasse

M	Massenmatrix
M_{Arm}	Massenmatrix des Armes
M_G	Massenmatrix der Glieder
n	Anzahl der Zustände
N	Anzahl der Gelenke
P	Leistung
P_v	Verlustleistung
P_{vG}	geschwindigkeitsabhängige Verlustleistung
P_{vP}	leistungsabhängige Verlustleistung
q	Gelenkposition
\ddot{q}_B	aufgrund von Sensordrift beobachtete Armbeschleunigung
Q_B	Beobachtbarkeitsmatrix
Q_S	Steuerbarkeitsmatrix
r	Position, relativer Grad
r_A	Antriebsreibung
r_G	Gelenkreibung
R	Entkopplungsmatrix
${}^B R^A$	Rotationsmatrix vom System A ins System B
S	Matrix der inertialen Kopplungen zwischen den Gliedern und den Motoren
t_{SR}	Zeitkonstante des Stromreglers
T	Beschleunigungstransformation, Transformation auf Normalform
\mathcal{T}	kinetische Energie
$\mathcal{U}_{Glieder}$	kinetische Energie der Glieder
\mathcal{U}_{Rotor}	kinetische Energie des Rotors
u	Systemeingang
u_c	Regleranteil des Systemeingangs
u	Getriebeübersetzung
u_0	Basisübersetzung
\hat{u}	Amplitude der Übersetzungsschwankungen
\tilde{u}	Frequenz der Übersetzungsschwankungen
\bar{u}	Phasenlage der Übersetzungsschwankungen
n_u	Anzahl der für Übersetzungsschwankungen berücksichtigten Frequenzen
u_C	Integrationskonstante für die Übersetzung
U	Matrix der Getriebeübersetzungen
v	translatorische Geschwindigkeit
V	Lyapunov-Funktion
\mathcal{U}	potentielle Energie
\mathcal{U}_{Grav}	potentielle Energie aufgrund der Gravitation
W	Summand zur Transformation der Beschleunigungen
x	Systemzustand
\hat{x}	geschätzter Systemzustand
x_0	Anfangszustand
y	Systemausgang
y_F	flacher Ausgang
y_m	Messgrößen
z	Zustand der internen Dynamik

C.3 Index

<i>a</i>	Flansch a eines Antriebsstrangelements
<i>A</i>	Flansch A eines Mehrkörperelements
<i>b</i>	Flansch b eines Antriebsstrangelements
<i>B</i>	Flansch B eines Mehrkörperelements
<i>d</i>	Sollgröße

Literaturverzeichnis

- [AAGH89] AN, C. H. ; ATKESON, C. G. ; GRIFFITHS, J. D. ; HOLLERBACH, J. M.: Experimental Evaluation of Feedforward and Computed Torque Control. In: *IEEE Transactions on Robotics and Automation* 5 (1989), Nr. 3, S. 368–373
- [Abe03] ABERGER, M.: *Modellierung und Identifikation eines CYCLO-Getriebes für Industrieroboter*, Johannes Kepler Universität Linz, Diplomarbeit, 2003
- [ABJ97] AUER, E. ; BALS, J. ; JOOS, H.-D.: Automatische Reglereinstellung bei Robotern in einem ISO-Inbetriebnahmeprozess mit mehrzielliger Optimierung. In: *2. VDI/VDE-GMA-Aussprachetag Rechnergestützter Entwurf von Regelungssystemen*, 1997
- [AH91] ARMSTRONG-HÉLOUVRY, B.: *Control of Machines with Friction*. 1. Auflage. Kluwer Academic Publishers, 1991
- [AHd96] ARMSTRONG-HÉLOUVRY, B. ; DE WIT, C. C.: Friction Modeling and Compensation. In: LEVINE, W. S. (Hrsg.): *The Control Handbook*. Boca Raton : CRC Press, 1996, S. 1369–1382
- [AHDd94] ARMSTRONG-HÉLOUVRY, B. ; DUPONT, P. ; DE WIT, C. C.: A Survey of Models, Analysis Tools and Compensation Methods for the Control of Machines with Friction. In: *Automatica* 30 (1994), Nr. 7, S. 1083–1138
- [AL96] AILON, A. ; LOZANO, R.: Controller-Observers for Set-point Tracking of Flexible-joint Robots Including Coriolis and Centripetal Effects in Motor Dynamics. In: *Automatica* 32 (1996), Nr. 9, S. 1329–1331
- [Alb01] ALBAN, T.: *Dynamische Roboterkalibrierung zur Erhöhung der Bahngenauigkeit*, Universität des Saarlandes, Diss., 2001
- [ALLG97] AILON, A. ; LOZANO-LEAL, R. ; GIL, M.: Point-To-Point Regulation of a Robot with Flexible Joints Including Electrical Effects of Actuator Dynamics. In: *IEEE Transactions on Automatic Control* 42 (1997), Nr. 4, S. 559–564
- [ARCB01] ALAVAREZ-RAMIREZ, J. ; CERVANTES, I. ; BAUTISTA, R.: Robust PID Control for Robots Manipulators with Elastic Joints. In: *Proceedings of the 2001 IEEE International Conference on Control Applications*, 2001, S. 542–547
- [AS02] ALBU-SCHÄFFER, A.: *Regelung von Robotern mit elastischen Gelenken am Beispiel der DLR-Leichtbauarme*, TU München, Diss., 2002

- [ASH00] ALBU-SCHÄFFER, A. ; HIRZINGER, G.: State feedback controller for flexible joint robots: A globally stable approach implemented on DLR's light-weight robots. In: *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2000
- [Bau72] BAUMGARTE, J.: Stabilization of constraints and integrals of motion in dynamic systems. In: *Computer Methods in Applied Mechanics and Engineering* 1 (1972), Nr. 1, S. 1–16
- [BCP89] BRENNAN, K. E. ; CAMPBELL, S. L. ; PETZOLD, L. R.: *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Elsevier Science Publishing Co., Inc., 1989
- [BG98] BECK, H.-P. ; GOSLAR, M.: Aktive Schwingungsdämpfung in elektrischen Hochleistungsantrieben. In: *VDI Berichte* 1416 (1998)
- [BL95] BATTIOTTI, S. ; LANARI, L.: Global set point control via link position measurement for flexible joint robots. In: *Systems & Control Letters* 25 (1995), Nr. 1, S. 21–29
- [BM67] BIRTA, L. G. ; MUFTI, I. H.: Some Results on an Inverse Problem in Multivariable Systems. In: *IEEE Transactions on Automatic Control* 12 (1967), February, S. 99–101
- [BOL95] BROGLIATO, B. ; ORTEGA, R. ; LOZANO, R.: Global Tracking Controllers for Flexible-joint Manipulators: a Comparative Study. In: *Automatica* 31 (1995), Nr. 7, S. 941–956
- [BR98] BROGLIATO, B. ; REY, D.: Further experimental results on nonlinear control of flexible joint manipulators. In: *Proceedings of the American Control Conference*, 1998
- [BRPB98] BROGLIATO, B. ; REY, D. ; PASTORE, A. ; BARNIER, J.: Experimental Comparison of Nonlinear Controllers for Flexible Joint Manipulators. In: *The International Journal of Robotics Research* 17 (1998), Nr. 3, S. 260–281
- [BT05] BEI TECHNOLOGIES, Systron Donner Inertial D. *Datasheet BEI Gyrochip II*. <http://www.systron.com/PDFS/datasheets/GyroIIDS.pdf>. October 2005
- [De 88] DE LUCA, A.: Dynamic Control of Robots with Joint Elasticity. In: *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, 1988, S. 152–158
- [DER86] DUFF, I. S. ; ERISMAN, A. M. ; REID, J. K.: *Direct Methods for Sparse Matrices*. Clarendon Press, Oxford, 1986
- [DFL05] DE LUCA, A. ; FARINA, R. ; LUCIBELLO, P.: On the Control of Robots with Visco-Elastic Joints. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, S. 4308–4313
- [DH55] DENAVIT, J. ; HARTENBERG, R. S.: A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices. In: *Journal of Applied Mechanics* (1955), S. 215–221

- [Die95] DIEWALD, B. W.: *Über-alles-Kalibrierung von Industrierobotern zur lokalen Minimierung der Posefehler*, Universität des Saarlandes, Diss., 1995
- [DIN85] DE LUCA, A. ; ISIDORI, A. ; NICOLÒ, F.: Control of Robot Arm with Elastic Joints via Nonlinear Dynamic Feedback. In: *Proceedings of the 24th Conference on Decision and Control*, 1985, S. 1671–1679
- [DL95] DE LUCA, A. ; LANARI, L.: Robots with Elastic Joints are Linearizable via Dynamic Feedback. In: *Proceedings of the 34th IEEE Conference on Decision & Control*, 1995, S. 3895–3897
- [DL98] DE LUCA, A. ; LUCIBELLO, P.: A General Algorithm for Dynamic Feedback Linearization of Robots with Elastic Joints. In: *Proceedings of the 1998 IEEE International Conference on Robotics & Automation*, 1998, S. 504–510
- [dOÅL95] DE WIT, C. C. ; OLSSON, H. ; ÅSTRÖM, K. J. ; LISCHINSKY, P.: A New Model for Control of Systems with Friction. In: *IEEE Transactions on Automatic Control* 40 (1995), Nr. 3, S. 419–425
- [DSZ05] DE LUCA, A. ; SICILIANO, B. ; ZOLLO, L.: PD control with on-line gravity compensation for robots with elastic joints: Theory and experiments. In: *Automatica* 41 (2005), Nr. 10, S. 1809–1819
- [DT96] DE LUCA, A. ; TOMEI, P.: Elastic Joints. In: DE WIT, Carlos C. (Hrsg.) ; SICILIANO, Bruno (Hrsg.) ; BASTIN, Georges (Hrsg.): *Theory of Robot Control*. Springer-Verlag, 1996, S. 179–217
- [Dyn04] Dynasim AB: *Dymola, Dynamic Modeling Laboratory, User's Manual*. 2004
- [DZQD98] DIXON, W. E. ; ZERGEROGLU, E. ; DE QUEIROZ, M. S. ; DAWSON, D. M.: Global Output Feedback Tracking Control for Rigid-Link Flexible-Joint Robots. In: *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, 1998, S. 489–503
- [Faß02] FASSNACHT, J.: *Schwingungsbedämpfung in Servosystemen mit der direkten Drehmomentmittelwertregelung*, TU Darmstadt, Diss., 2002
- [Fis73] FISZ, M.: *Wahrscheinlichkeitsrechnung und mathematische Statistik*. VEB Deutscher Verlag der Wissenschaften, Berlin, 1973
- [FKH83] FUTAMI, S. ; KYURA, N. ; HARA, S.: Vibration Absorption Control of Industrial Robots by Acceleration Feedback. In: *IEEE Transactions on Industrial Electronics* IE-30 (1983), Nr. 3, S. 299–305
- [Föll94] FÖLLINGER, O.: *Regelungstechnik: Einführung in die Methoden und ihre Anwendung*. 8. Hüthig Buch Verlag Heidelberg, 1994
- [FLMR95] FLIESS, M. ; LÉVINE, J. ; MARTIN, P. ; ROUCHON, P.: Flatness and defect of non-linear systems: introductory theory and examples. In: *International Journal of Control* 61 (1995), Nr. 6, S. 1327–1361

- [GBJ97] GRÜBEL, G. ; BALS, G. ; JOOS, H.-D.: A CACSD Environment for Control Tuning. In: *Journal a* 38 (1997), Nr. 3, S. 30–33
- [GDLL94] GRIZZLE, J. W. ; DI BENEDETTO, M. D. ; LAMNABHI-LAGARRIGUE, F.: Necessary Conditions for Asymptotic Tracking in Nonlinear Systems. In: *IEEE Transactions on Automatic Control* 39 (1994), Nr. 9, S. 1782–1794
- [GH02] GROTJAHN, M. ; HEIMANN, B.: Model-based Feedforward Control in Industrial Robotics. In: *International Journal of Robotics Research* 21 (2002), Nr. 1, S. 45–60
- [Gro03] GROTJAHN, M.: *Kompensation nichtlinearer dynamischer Effekte bei seriellen und parallelen Robotern zur Erhöhung der Bahngenauigkeit*, Universität Hannover, Diss., 2003
- [GS00] GHORBEL, F. ; SPONG, M. W.: Integral manifolds of singularly perturbed systems with application to rigid-link flexible-joint multibody systems. In: *International Journal of Non-Linear Mechanics* 35 (2000), S. 133–155
- [Gs01] GUNNARSSON, S. ; ÖSTRING, M.: On Regulator Stability in Control of Flexible Mechanical Systems / Division of Automatic Control, Department of Electrical Engineering, Linköpings universitet. 2001. – Forschungsbericht. Auch: International Symposium on Robotics, April 2001.
- [HASH⁺01] HIRZINGER, G. ; ALBU-SCHÄFFER, A. ; HÄHNLE, M. ; SCHAEFER, I. ; SPORER, N.: On a New Generation of Torque Controlled Light-Weight Robots. In: *Proceedings of the 2001 IEEE International Conference on Robotics & Automation*, 2001, S. 3356–3363
- [Hir79] HIRSCHORN, R. M.: Invertibility of Multivariable Nonlinear Control Systems. In: *IEEE Transactions on Automatic Control* 24 (1979), Nr. 6, S. 855–865
- [HJ61] HOOKE, R. ; JEEVES, T. A.: „Direct Search“ Solution of Numerical and Statistical Problems. In: *J. ACM* 8 (1961), Nr. 2, S. 212–229
- [Höl94] HÖLZL, J.: *Modellierung, Identifikation und Simulation der Dynamik von Industrierobotern*, Technische Universität München, Diss., 1994
- [HS67] HAHN, G. J. ; SHAPIRO, S. S.: *Statistical Models in Engineering*. John Wiley & Sons, Inc. New York, 1967
- [HT04] HÖPLER, R. ; THÜMMEL, M.: Symbolic Computation of the Inverse Dynamics of Elastic Joint Robots. In: *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, 2004
- [Hun91] HUNG, J. Y.: Control of Industrial Robots that Have Transmission Elasticity. In: *IEEE Transactions on Industrial Electronics* 38 (1991), Nr. 6, S. 421–427
- [HZ04] HAGENMEYER, V. ; ZEITZ, M.: Flachheitsbasierter Entwurf von linearen und nichtlinearen Vorsteuerungen. In: *at - Automatisierungstechnik* 52 (2004), Nr. 1, S. 3–12

- [IB90] ISIDORI, A. ; BYRNES, C. I.: Output Regulation of Nonlinear Systems. In: *IEEE Transactions on Automatic Control* 35 (1990), Nr. 2, S. 131–140
- [IM86] ISIDORI, A. ; MOOG, C. H.: On the Nonlinear Equivalent of the Notion of Transmission Zeros. In: *I.I.A.S.A. Workshop on Modelling and Adaptive Control*, 1986, S. 146–158
- [Int98] International Organization for Standardization: *ISO 9283:1998 Manipulating industrial robots – Performance criteria and related test methods*. 2. April 1998
- [INT05] INTEC. *Simpack Webseite*. <http://www.simpack.de/>. August 2005
- [Ise92a] ISERMANN, R.: *Identifikation dynamischer Systeme 1, Grundlegende Methoden*. 2. Springer Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hongkong, Barcelona, Budapest, 1992
- [Ise92b] ISERMANN, R.: *Identifikation dynamischer Systeme 2, Besondere Methoden, Anwendungen*. 2. Springer Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hongkong, Barcelona, Budapest, 1992
- [Isi95] ISIDORI, A. ; SONTAG, E. D. (Hrsg.) ; THOMA, M. (Hrsg.) ; ISIDORI, A. (Hrsg.) ; VAN SCHUPPEN, J. H. (Hrsg.): *Nonlinear Control Systems*. 3. Springer Verlag, Berlin, Heidelberg, New York, 1995
- [Isi99] ISIDORI, A. ; SONTAG, E.D. (Hrsg.) ; THOMA, M. (Hrsg.): *Nonlinear Control Systems II*. 1. Springer Verlag, London, Berlin, Heidelberg, 1999
- [JBL⁺02] JOOS, H.-D. ; BALS, J. ; LOOYE, G. ; SCHNEPPER, K. ; VARGA, A.: A multi-objective optimisation based software environment for control systems design. In: *Proceedings of 2002 IEEE International Conference on Control Applications and International Symposium on Computer Aided Control Systems Design*, 2002
- [JE94] JANKOWSKI, K. P. ; ELMARAGHY, H. A.: Inverse Dynamics and Feedforward Controllers for High Precision Position/Force Tracking of Flexible Joint Robots. In: *Robotica* 12 (1994), S. 227–241
- [Joo05] JOOS, H.-D.: *MOPS - Multi-Objective Parameter Synthesis, User's Guide V5.0*. Oberpfaffenhofen, 82234 Wessling: DLR Deutsches Zentrum für Luft- und Raumfahrt, Institut für Robotik und Mechatronik, Oktober 2005
- [KJ85] KUNTZE, H.-B. ; JACUBASCH, A. H. K.: Control Algorithms for Stiffening an Elastic Industrial Robot. In: *IEEE Journal of Robotics and Automation* RA-1 (1985), Nr. 1, S. 71–78
- [KK85] KHORASANI, K. ; KOKOTOVIC, P. V.: Feedback linearization of a flexible manipulator near its rigid body manifold. In: *Systems & Control Letters* 6 (1985), Nr. 3, S. 187–192
- [KK88] KHOSLA, P. K. ; KANADE, T.: Experimental Evaluation of Nonlinear Feedback and Feedforward Control Schemes for Manipulators. In: *International Journal of Robotics Research* 7 (1988), Nr. 1, S. 18–28

- [KM97] KINCELER, R. ; MECKL, P. H.: Corrective Input Shaping for a Flexible-Joint Manipulator. In: *Proceedings of the 1997 American Control Conference*, 1997
- [KOAL94] KELLY, R. ; ORTEGA, R. ; AILON, A. ; LORIA, A.: Global Regulation of Flexible Joint Robots Using Approximate Differentiation. In: *IEEE Transactions on Automatic Control* 39 (1994), Nr. 6, S. 1222–1224
- [Koz03] KOZAK, K. C.: *Robust Command Generation For Nonlinear Systems*, Georgia Institute of Technology, Diss., 2003
- [KR92] KÖHLER, G. ; RÖGNITZ, H.: *Maschinenteile*. Bd. 2. 8. Teubner Verlag, 1992
- [Kre99] KREISSELMEIER, G.: Struktur mit zwei Freiheitsgraden. In: *at - Automatisierungstechnik* 47 (1999), Nr. 6, S. 266–
- [Kry01] Krypton Electronic Engineering n.v.: *Krypton Help Pages on Robot Performance Testing, Roboscope/ISO*. February 2001
- [KS96] KHALIL, W. ; SICILIANO, B.: Modelling and identification. In: DE WIT, Carlos C. (Hrsg.) ; SICILIANO, Bruno (Hrsg.) ; BASTIN, Georges (Hrsg.): *Theory of Robot Control*. Springer Verlag, Berlin, Heidelberg, New York, 1996, S. 179–217
- [Kuh03] KUHLEN, K.: Modeling, Identification and Compensation of Complex Hysteretic Nonlinearities. A modified Prandtl-Ishlinskii Approach. In: *European Journal of Control* 9 (2003), Nr. 4, S. 407–418
- [KUK96a] KUKA Roboter GmbH: *Spezifikation KR15/2*. 1996
- [KUK96b] KUKA Roboter GmbH: *Technische Daten KR15/2*. 1996
- [Kum00] KUMMETZ, J.: *Nichtlinear-dynamische Effekte in elektromechanischen Antrieben*, Universität Hannover, Diss., 2000
- [Lan03] LANGE, F.: *Adaptiv vorausplanende Steuerung für schnelle sensorbasierte Roboterbewegungen*, TH Karlsruhe, Diss., 2003
- [Leh76] LEHMANN, M.: *Berechnung und Messung der Kräfte in einem Zykloiden-Kurvenscheiben-Getriebe*, Technische Universität München, Diss., 1976
- [LS89] LEAHY JR., M. B. ; SARIDIS, G. N.: Compensation of Industrial Manipulator Dynamics. In: *International Journal of Robotics Research* 8 (1989), Nr. 4, S. 73–84
- [May91] MAYERGOYZ, I. D.: *Mathematical Models of Hysteresis*. Springer Verlag, 1991
- [MEG05] MEGATRON INDUSTRIESSENSORIK. *Datenblatt Megatron Kraftsensor K1500*. http://www.megatron.de/Kraftsensoren/Standardsensorik/Shortview_K1500-/k1500_v02_0.pdf. 10 2005
- [MMW84] MATEK, W. ; MUHS, D. ; WITTEL, H.: *Roloff/Matek - Maschinenelemente*. 9. Auflage. Braunschweig : Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, 1984

- [Mod04] Modelica Association: *Modelica - A Unified Object-Oriented Language for Physical Systems Modeling, Language Specification, Version 2.2*. January 2004
- [MS93] MATTSSON, S. E. ; SÖDERLIND, G.: Index Reduction in Differential-Algebraic Equations using Dummy Derivatives. In: *SIAM Journal of Scientific Computing* 14 (1993), Nr. 3, S. 667–692
- [Müs03] MÜSER, M. H.: Der mikroskopische Ursprung der Reibung. In: *Physik Journal* 2 (2003), Nr. 9, S. 43–48
- [MSC05] MSC. *Adams Webseite*. <http://www.mscsoftware.com>. August 2005
- [MWS90] MURPHY, S. H. ; WEN, J. T. ; SARIDIS, G. N.: Simulation and Analysis of Flexibly Jointed Manipulators. In: *Proceedings of the 29th Conference on Decision and Control*, 1990, S. 545–550
- [NT93] NICOSIA, S. ; TOMEI, P.: Design of Global Tracking Controllers for Flexible-Joint Robots. In: *Journal of Robotic Systems* 10 (1993), Nr. 6, S. 835–846
- [Nv90] NIJMEIJER, H. ; VAN DER SCHAFT, A.: *Nonlinear Dynamical Control Systems*. Springer Verlag, New York, Berlin, Heidelberg, London, Paris, Tokyo, Hongkong, 1990
- [NW02] NIEMANN, G. ; WINTER, H.: *Maschinenelemente*. Bd. 2: Getriebe allgemein, Zahnradgetriebe - Grundlagen, Stirnradgetriebe. 2. Springer Verlag, 2002
- [OÅd⁺98] OLSSON, H. ; ÅSTRÖM, K. J. ; DE WIT, C. C. ; GÄFVERT, M. ; LISCHINSKY, P.: Friction Models and Friction Compensation. In: *European Journal of Control* 4 (1998), Nr. 3
- [OEM99] OTTER, M. ; ELMQVIST, H. ; MATTSSON, S. E.: Hybrid Modeling in Modelica based on the Synchronous Data Flow Principle. In: *Proceedings of the 1999 IEEE International Symposium on Computer Aided Control System Design*, 1999, S. 151–157
- [OEM03] OTTER, M. ; ELMQVIST, H. ; MATTSSON, S. E.: The New Modelica MultiBody Library. In: *Proceedings of the 3rd International Modelica Conference*, 2003, S. 311–330
- [Ols96] OLSSON, H.: *Control Systems with Friction*, Department of Automatic Control, Lund Institute of Technology, Diss., 1996
- [Ös02] ÖSTRING, M.: *Identification, Diagnosis, and Control of a Flexible Robot Arm*, Linköpings Universitet, Diss., 2002
- [Ott95] OTTER, M.: *Objektorientierte Modellierung mechatronischer Systeme am Beispiel geregelter Roboter*, Ruhr-Universität Bochum, Diss., 1995
- [Ott00] OTTER, M.: Objektorientierte Modellierung Physikalischer Systeme (Teil 1-9). In: *at - Automatisierungstechnik* 48 (2000), Nr. 1-12

- [Pan88] PANTELIDES, C. C.: The Consistent Initialization of Differential-Algebraic Systems. In: *SIAM Journal of Scientific and Statistical Computing* 9 (1988), Nr. 2, S. 213–231
- [PCH99] PARK, H.-S. ; CHANG, P. H. ; HUR, J.-S.: Time-Varying Input Shaping Technique Applied to Vibration Reduction of An Industrial Robot. In: *Proceedings of the 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1999, S. 285–290
- [PSO02] PELCHEN, C. ; SCHWEIGER, C. ; OTTER, M.: Modeling and Simulating the Efficiency of Gearboxes and of Planetary Gearboxes. In: *Proceedings of the 2nd International Modelica Conference, Oberpfaffenhofen, Germany, 2002*, S. 257–266
- [Rau03] RAU, M.: *Nichtlineare modellbasierte prädiktive Regelung auf Basis lernfähiger Zustandsraummodelle*, Technische Universität München, Diss., 2003
- [Rea94] READMAN, M. C. ; SASTRY, S. S. (Hrsg.): *Flexible Joint Robots*. CRC Press, Boca Raton, Ann Arbor, London, Tokyo, 1994
- [RRZ97] ROTHFUSS, R. ; RUDOLPH, J. ; ZEITZ, M.: Flachheit: Ein neuer Zugang zur Steuerung und Regelung nichtlinearer Systeme. In: *at - Automatisierungstechnik* 45 (1997), Nr. 11, S. 517–525
- [Sch96] SCHRÖDER, D.: *Elektrische Antriebe 3, Leistungselektronische Bauelemente*. 1. Springer Verlag, 1996
- [Sch00a] SCHRÖDER, D.: *Elektrische Antriebe 1, Grundlagen*. 2. Springer Verlag, 2000
- [Sch00b] SCHRÖDER, D.: *Intelligent Observer and Control Design for Nonlinear Systems*. 1. Springer Verlag, 2000
- [Sch03] SCHRÖDER, D.: *Intelligente Verfahren für Mechatronische Systeme*. Oktober 2003
- [SDO99] SAGE, H. G. ; DE MATHELIN, M. F. ; OSTERTAG, E.: Robust control of robot manipulators: a survey. In: *International Journal of Control* 72 (1999), Nr. 16, S. 1498–1522
- [SG85] SWEET, L. M. ; GOOD, M. C.: Redefinition of the Robot Motion-Control Problem. In: *IEEE Control Systems Magazine* (1985), S. 18–25
- [Sie04] Siemens AG: *motion control Servomotoren*. 2004
- [Sil82] SILVER, W. M.: On the Equivalence of Lagrangian and Newton-Euler Dynamics for Manipulators. In: *International Journal of Robotics Research* 1 (1982), Nr. 2, S. 60–70
- [Sin81] SINGH, S. N.: A Modified Algorithm for Invertibility in Nonlinear Systems. In: *IEEE Transactions on Automatic Control* AC-26 (1981), Nr. 2, S. 595–598
- [SJK97] SEPULCHRE, R. ; JANKOVIC, M. ; KOKOTOVIC, P. V.: *Constructive Nonlinear Control*. Springer-Verlag, 1997

- [SKK87] SPONG, M. W. ; KHORASANI, K. ; KOKOTOVIC, P. V.: An Integral Manifold Approach to the Feedback Control of Flexible Joint Robots. In: *IEEE Journal of Robotics and Automation* 3 (1987), Nr. 4, S. 291–300
- [SL91] SLOTINE, J.-J. E. ; LI, W.: *Applied Nonlinear Control*. Prentice-Hall, Englewood Cliffs, New Jersey, 1991
- [Son89] SONTAG, E. D.: Smooth Stabilization Implies Coprime Factorization. In: *IEEE Transactions on Automatic Control* 34 (1989), April, Nr. 4, S. 435–443
- [Spo87] SPONG, M. W.: Modeling and Control of Elastic Joint Robots. In: *Transactions of the ASME* 109 (1987), S. 310–319
- [Spo90] SPONG, M. W.: Control of Flexible-Joint Robots: A Survey. In: CHEN, Goong (Hrsg.) ; LEE, E. B. (Hrsg.) ; LITTMAN, Walter (Hrsg.) ; MARKUS, Lawrence (Hrsg.): *Distributed Parameter Control Systems*. Marcel Dekker Inc., New York, Basel, Hongkong, 1990, S. 355–383
- [ST02] SCHOTT, J. ; THÜMMEL, M.: 1 dimensional Accelerometer / Deutsches Zentrum für Luft- und Raumfahrt, Institut für Robotik und Mechatronik. 2002 (515-02-16). – Institutsbericht
- [Sti00] STIELER, B. *Inertiale Meßtechnik und integrierte Navigation*. Juni 2000
- [Sum96] Sumitomo (SHI) Cyclo Drive Europe, Ltd.: *Serie Fine Cyclo*. 1996
- [Tei] Teijin-Seiki: *Catalogue RV-E Series*
- [TOB01] THÜMMEL, M. ; OTTER, M. ; BALS, J.: Simulation von Industrierobotern. In: BAUSCH-GALL, Ingrid (Hrsg.): *Simulation technischer Systeme*. SCS - The Society for Computer Simulation International, 2001 (FS 7), S. 115–149
- [Tom91] TOMEI, P.: A Simple PD Controller for Robots with Elastic Joints. In: *IEEE Transactions on Automatic Control* 36 (1991), Nr. 10, S. 1208–1213
- [Tom94] TOMEI, P.: Tracking Control of Flexible Joint Robots with Uncertain Parameters and Disturbances. In: *IEEE Transactions on Automatic Control* 39 (1994), Nr. 5, S. 1067–1072
- [van97] VAN NIEUWSTADT, M. J.: *Trajectory Generation for Nonlinear Control Systems*, California Institute of Technology, Diss., 1997
- [WA87] WIDMANN, G. R. ; AHMAD, S.: Control of Industrial Robots with Flexible Joints. In: *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*, 1987, S. 1561–1566
- [Wan03] WANG, J.: *Numerical and Experimental Analysis of Spur Gears in Mesh*, Curtin University of Technology, Diss., 2003
- [Wie01] WIEST, U.: *Kinematische Kalibrierung von Industrierobotern*, TH Karlsruhe, Diss., 2001