
Simulation des Fräsens mit Industrierobotern: Trajektorienplanung und experimentelle Parameteridentifikation

Simulation of milling with industrial robots: trajectory planning and experimental parameter identification

Bachelor-Thesis von Peter Englert

Mai 2011



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Informatik
Fachgebiet Simulation,
Systemoptimierung und Robotik

Simulation des Fräsens mit Industrierobotern: Trajektorienplanung und experimentelle
Parameteridentifikation
Simulation of milling with industrial robots: trajectory planning and experimental parameter
identification

Vorgelegte Bachelor-Thesis von Peter Englert

Gutachter: Prof. Dr. Oskar von Stryk
Betreuer: Dr.-Ing. Christian Reinl, Dipl.-Ing. Jörg Bauer

Tag der Einreichung:

Erklärung zur Bachelor-Thesis

Hiermit versichere ich, die vorliegende Bachelor-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 17. Mai 2011

(Peter Englert)

Zusammenfassung

Industrieroboter werden vermehrt zum Zerspanen von weichen Werkstoffen eingesetzt. Aufgrund ihrer im Vergleich zur klassischen Fräsmaschine hohen Nachgiebigkeit sind Industrieroboter jedoch nicht optimal für Fräsaufgaben mit hartem Material geeignet. Die entstehenden Kräfte bei der Bearbeitung rufen eine Abdrängung des Roboters hervor, die sich negativ auf die Qualität des Werkstücks auswirkt. Dem gegenüber stehen die Vorteile der hohen Flexibilität eines Industrieroboters, sowie den niedrigeren Kosten pro Arbeitsraum im Vergleich zur klassischen Fräsmaschine. Zur Erhöhung der Bearbeitungsgenauigkeit wird derzeit in aktueller Forschung und Entwicklung die Roboter-Trajektorie mit einer modellbasierten Offline-Kompensation vor der Fräsbearbeitung angepasst. Hierzu wird eine Simulation des dynamischen Roboterstrukturmodells mit einem gekoppelten Fräskraftmodell verwendet, die die Abdrängungen und Schwingungen des Roboters modelliert. Daraus wird anschließend eine kompensierte Fräsbahn für den Roboter berechnet. Im Rahmen der vorliegenden Arbeit werden zwei Aspekte der modellbasierten Offlinekompensation näher untersucht. Zum einen werden Versuche zur Parameteridentifikation durchgeführt, mit denen das vorhandene Robotermodell mit Messdaten an einen realen Roboter angepasst wird. Des Weiteren wird in das vorhandene Programm-Framework eine Trajektorienplanung implementiert mit der die Trajektorien generiert werden, auf denen die Simulation und Kompensation der Fräskräfte erfolgt. Damit wird die für die Qualität der Kompensationsstrategie maßgebliche Genauigkeit signifikant verbessert und die Anwendbarkeit der modellbasierten Offlinekompensation auf komplexe Bauteile erweitert.

Abstract

Recently, industrial robots are increasingly used for cutting soft material. Due to their high compliance compared to standard milling machines, industrial robots are not optimally suited for milling tasks with hard material. Because of the milling process and its process force, a static displacement and a low frequent oscillation of the Tool Centre Point (TCP) is induced. This has negative effects on the quality of a workpiece. In contrast, industrial robots are cost-saving with respect to workspace and more flexible for application to different tasks. For achieving higher accuracy of robots in milling applications, currently a model-based offline compensation is used in research and development. For this strategy, a coupled simulation of the robot model and the milling process model is used which calculates the deviation of the TCP under process forces. Based on these results a compensating cutting path is calculated with a path correction strategy. In this work, two aspects of the model-based offline compensation are examined. First, several tests are performed for parameter identification, that adapts the existing robot model with measurement data to a real robot. Furthermore, a trajectory planning module is implemented into an existing program framework for generating the trajectory on which the simulation and compensation of the milling forces is performed. This significantly improves the accuracy of the compensation strategy and extends the applicability of the model-based offline compensation to more complex workpieces.

Inhaltsverzeichnis

1	Einleitung	4
2	Grundlagen und vorhandene Modellstruktur	5
2.1	Strategie der modellbasierten Offline-Kompensation	5
2.2	Komponenten der Simulation	6
2.2.1	Roboterstrukturmodell	6
2.2.2	Fräskraftmodell	7
3	Experimentelle Parameteridentifikation	8
3.1	Steifigkeit	8
3.1.1	Grundlagen	8
3.1.2	Versuchsaufbau und Durchführung	8
3.2	Dämpfung	10
3.3	Validierung anhand der Gesamtsteifigkeit im Arbeitsraum	11
4	Trajektorienplanung	13
4.1	Stand der Technik	13
4.1.1	Geometrie der Bahn	13
4.1.2	Zeitliche Regulierung der Bewegung	15
4.2	Erstellung einer Trajektorienplanung	17
4.2.1	Auswahl geeigneter Variante	17
4.2.2	Implementierung	18
4.2.3	Auswertung	19
5	Zusammenfassung	21
	Anhang	22
A	Roll-Pitch-Yaw-Winkel	23
B	Analytische Berechnung der inversen Kinematik	24

1 Einleitung

Der Einsatzbereich von Industrierobotern weitet sich im Zuge der Fertigungsautomatisierung immer weiter aus. Ihre klassischen Aufgaben sind zum Beispiel das Montieren, Kleben und Schweißen in der Fließbandfertigung. All diese Aufgaben haben gemein, dass keine großen Kräfte auf den Endeffektor wirken und die Wiederhol- und Absolutgenauigkeit des Roboters ausreichend ist. Erweiterungen des Aufgabenfelds auf Tätigkeiten mit höheren Bearbeitungskräften, wie etwa der Fräsbearbeitung oder dem Blechformen, führen zu einer Abdrängung des Endeffektors von seiner Sollbahn und somit zu höheren Ungenauigkeiten bei der Bearbeitung.

Im Rahmen der vorliegenden Arbeit wird sich mit der Durchführung von spanenden Bearbeitungen mit dem Industrieroboter beschäftigt. Diese werden meist von klassischen Werkzeugmaschinen durchgeführt, weil diese eine höhere Bearbeitungsgenauigkeit von weniger als $0,1\text{mm}$ erreichen. Der Einsatz von Industrierobotern ist hierbei eine sinnvolle Alternative, da dieser im Vergleich zur Werkzeugmaschine geringere Investitionskosten aufweist und einen größeren Arbeitsraum bearbeiten kann. Aspekte für die niedrigere Bearbeitungsgenauigkeit von Industrierobotern in der spanenden Bearbeitung im Vergleich zu klassischen Werkzeugmaschinen sind [Abele, 2008]:

- eine geringere Grundgenauigkeit des Roboters,
- die geringere Steifigkeit aufgrund der Getriebe sowie der schlanken Roboterstruktur,
- variierende Nachgiebigkeit je nach Position und Orientierung des Endeffektors im Arbeitsraum und
- die indirekte Lageregelung der Achsen.

Wie in [Weigold, 2008] erläutert wird, sind die Ungenauigkeiten die durch die Prozesskräfte bei der spanenden Bearbeitung entstehen mit $0,5\text{mm}$ relativ hoch und bilden damit eine entscheidende Einflussgröße auf die resultierende Bearbeitungsqualität. Eine Möglichkeit diese Ungenauigkeiten zu reduzieren ist der Einsatz einer modellbasierten Offline-Kompensation. Hier wird eine Veränderung der Fräsbahnvorgabe vorgenommen, mit der die auftretenden Abdrängungen kompensiert werden. Die Veränderungen der Bahn werden dabei im Vorfeld der Bearbeitung mit einer Simulation ermittelt [Stelzer, 2008].

In der vorliegenden Bachelorarbeit werden für die modellbasierte Offline-Kompensation vorhandene Komponenten angepasst und neue hinzugefügt. Diese dienen im Wesentlichen dazu, die verfolgte Strategie an einem realen Roboter anzuwenden und dabei ihre Tauglichkeit zu untersuchen. Dafür wird eine vorhandene Modellstruktur an einen realen Roboter angepasst. Das Ziel dabei ist es ein möglichst präzises Simulationsmodell des Roboters zu erstellen. Um dieses zu erreichen werden Experimente zur Parameteridentifikation durchgeführt. Dabei werden die auftretenden Elastizitäten des Roboters mit Parametern in der Simulation beschrieben. Des Weiteren wird eine Trajektorienplanung entwickelt und in die Simulation integriert. Dabei dient der parametrisierte Roboterbefehlssatz als Basis für die Auswahl geeigneter Trajektorien. Ebenfalls werden dabei die Anforderungen an Trajektorien für die spanende Bearbeitung untersucht und berücksichtigt.

Im folgenden Kapitel 2 werden die bisherigen Tätigkeiten an dem Projekt vorgestellt und die notwendigen Komponenten für die vorliegende Arbeit näher betrachtet.

In Kapitel 3 wird gezeigt, wie die Parameter für das Roboterstrukturmodell experimentell ermittelt und anschließend angepasst wurden. Ebenfalls wird eine Evaluierung der Parameter durchgeführt, in der das Simulationsverhalten mit dem des realen Roboters verglichen wird.

Im 4. Kapitel wird zuerst der aktuelle Stand der Technik bei der Trajektorienplanung vorgestellt. Anschließend wird eine für die spanende Bearbeitung geeignete Variante gewählt, mit der die vorhandene Simulation erweitert wird.

2 Grundlagen und vorhandene Modellstruktur

Es wird die bisherige Arbeit im Rahmen der modellbasierten Offline-Kompensation vorgestellt auf der die vorliegende Arbeit aufbaut. Dabei wird zuerst die gewählte Strategie vorgestellt mit der die Bearbeitungsgenauigkeit erhöht wird. Im Anschluss werden die Komponenten der Simulation näher betrachtet.

2.1 Strategie der modellbasierten Offline-Kompensation

Die Verbesserung der Bearbeitungsgenauigkeit wird mit einem modellbasierten Ansatz untersucht. Dabei wird in einer Simulation das reale Verhalten des Roboters beim Fräsen modelliert. In dieser Simulation wird die Abweichung des Roboters durch die auftretenden Prozesskräfte von der Sollbahn-Vorgabe ermittelt. Mit der Strategie der Offline-Kompensation wird nun anhand der simulierten Abweichungen eine Bahnkorrektur vorgenommen. Die daraus resultierende Bahn kompensiert die entstehenden Abweichungen beim Fräsen wodurch eine höhere Bearbeitungsgenauigkeit erreicht wird. In der folgenden Grafik sind die einzelnen Schritte der modellbasierten Offline-Kompensation dargestellt.

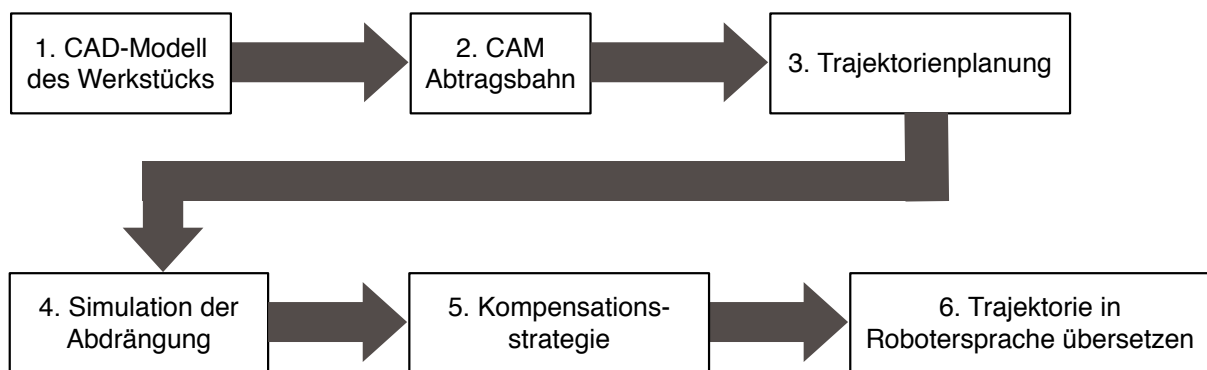


Abbildung 2.1: Vorgehensweise der modellbasierten Offline-Kompensation

Die einzelnen Schritte werden im folgenden näher beschrieben:

1. Zu Beginn wird eine technische Zeichnung des Werkstücks mit Hilfe einer CAD (Computer Aided Design) Software erstellt. Das erstellte CAD-Modell beschreibt die geometrische Form des zu fertigenden Werkstücks.
2. Mit Hilfe eines CAM (Computer Aided Manufacturing) Programms wird eine Abtragsbahn für das Werkstück ermittelt. Dazu wird zuerst das Ausgangs-Bauteil definiert, aus dem das konstruierte Werkstück heraus gefräst werden soll. Des Weiteren wird das verwendete Material und Eigenschaften des Fräswerkzeuges festgelegt. Hieraus wird eine geometrische Bahn berechnet mit der aus dem Ausgangs-Bauteil das Werkstück gefräst werden kann.
3. Aus der geometrischen Bahn wird durch festlegen von zeitlichen Beschränkungen eine Trajektorie generiert. Diese Trajektorie beschreibt zu jedem Zeitpunkt die Position, Orientierung, Geschwindigkeit und Beschleunigung vom TCP des Roboters.
4. Auf Basis dieser Trajektorie wird anschließend die Simulation durchgeführt. Mit einem Fräskraftmodell werden dabei die entstehenden Fräskräfte modelliert. Aus der Wechselwirkung dieser Kräfte mit dem Robotermodell lässt sich die Abweichung des TCP von der Sollbahn simulieren.
5. Mit einer geeigneten Kompensationsstrategie wird aus den ermittelten Abweichungen eine korrigierte Trajektorie erstellt.
6. Zum Schluss wird aus der kompensierten Trajektorie ein ausführbarer Programm in der Programmiersprache des Roboters erstellt. Dazu steht ein parametrisierter Roboterbefehlssatz zur Verfügung.

Die Schritte 1, 2 und 6 entsprechen dabei der Standard Vorgehensweise beim Fräsen mit Industrierobotern. In den Schritten 3, 4 und 5 kommen die neu hinzugefügten Komponenten der modellbasierten Offline-Kompensation zum Einsatz. Ein Vorteil dieser Strategie ist, dass keine Veränderungen an der Roboterstruktur oder Steuerung vorgenommen werden. Für die in Schritt 4 angesprochene Simulation wird ein Modell des Roboters und eines für den Fräsprozess benötigt, die im folgenden beschrieben werden.

2.2 Komponenten der Simulation

Der Industrieroboter wurde in einem Roboterstrukturmodell und der Fräsprozess in einem Fräskraftmodell abgebildet. Diese beiden Komponenten sind miteinander in einer Simulation gekoppelt (vgl. Abb. 2.2). Hiermit lässt sich die Wechselwirkung zwischen Fräsprozess und Roboterstruktur analysieren, welches ein charakterisierendes Merkmal der untersuchten Problemstellung ist. Im folgenden werden das verwendete Roboterstrukturmodell und das Fräskraftmodell näher vorgestellt.

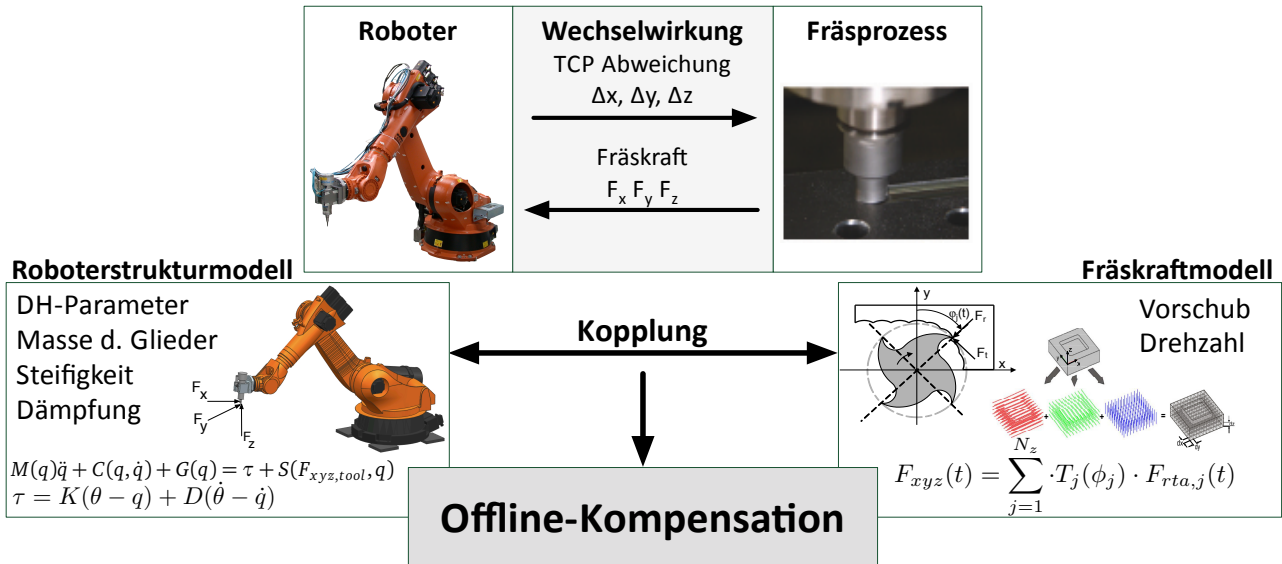


Abbildung 2.2: Überblick über die Kopplung des Roboterstrukturmodells mit dem Fräskraftmodell, um die Wechselwirkung der Roboterstruktur mit dem Fräsprozess zu simulieren

2.2.1 Roboterstrukturmodell

Bei dem untersuchten Roboter handelt es sich um den 6-achsigen Vertikal-Knickarmroboter KUKA KR 210 (Abb. 2.3). Wichtige Kennwerte des Roboters sind die Positionierwiederholgenauigkeit mit $< \pm 0,06$ mm und einer maximalen Traglast von 210 kg¹. Die Roboterdynamik wird durch die Gleichung [Craig, 1989]

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau + S(F_{xyz, tool}, q) \quad (2.1)$$

beschrieben. Hierbei entsprechen:

- $q \in \mathbb{R}^n, \dot{q} \in \mathbb{R}^n, \ddot{q} \in \mathbb{R}^n$ Winkel, Geschwindigkeit und Beschleunigung der Gelenke
- $C(q, \dot{q}) \in \mathbb{R}^n$ Zentrifugal- und Corioliskräfte
- $G(q) \in \mathbb{R}^n$ Gravitationskraft
- $M(q) \in \mathbb{R}^{n \times n}$ Massenträgheitsmatrix der Armeile
- $\tau \in \mathbb{R}^n$ Wirkende Drehmomente in den Gelenken
- $S(F, q) \in \mathbb{R}^n$ Kräfte und Momente die auf den TCP durch Prozesskräfte wirken.

Die Dimension der einzelnen Größen hängt von der Anzahl der modellierten Gelenke n ab. Die Gelenkdrehmomente werden durch ein Feder-Dämpfer Modell beschrieben. Sie können mit der Steifigkeitsmatrix $K \in \mathbb{R}^{n \times n}$ und der Dämpfungsmatrix $D \in \mathbb{R}^{n \times n}$ mit folgender Gleichung berechnet werden:

$$\tau = K(\theta - q) + D(\dot{\theta} - \dot{q}) \quad (2.2)$$

Die beiden Matrizen K und D sind dabei Diagonalmatrizen. Die Werte $\theta \in \mathbb{R}^n$ und $\dot{\theta} \in \mathbb{R}^n$ entsprechen dabei den Sollwerten der Roboterjelenke. Die Elastizitäten des Roboters lassen sich, wie man aus Gleichung 2.2 erkennen kann,

¹ Datenblatt KUKA KR 210-2 http://www.kuka-robotics.com/res/sps/f776ebab-f613-4818-9feb-527612db8dc4_PF0022_KR_210-2_de.pdf

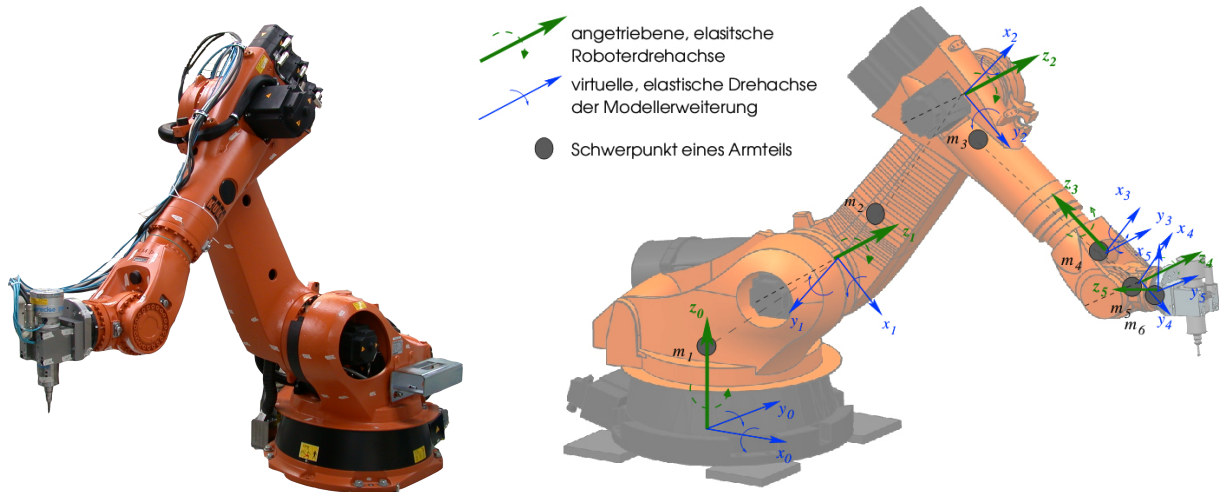


Abbildung 2.3: Kuka Roboter und die Achsen zur Modellierung von Elastizitäten bei der Gelenklagerung

um einzelne Achsen herum im Modell abbilden. Diese wirken nicht nur an den Antriebsachsen des Roboters. Auch an weiteren Stellen wurden Elastizitäten gemessen, die einen Einfluss auf die Abweichung des TCP haben. Um diese auch im Modell zu berücksichtigen wird ein Ansatz verwendet mit dem sich virtuelle Achsen an beliebigen Stellen in das Modell integrieren lassen. Neben den Elastizitäten um die Antriebsachsen wirken hauptsächlich weitere Steifigkeiten und Dämpfungen bei der Lagerung der Gelenke. Ihre Achsen verlaufen orthogonal zu den Antriebsachsen und liegen im Gelenkursprung. Abbildung 2.3 rechts zeigt eine Aufteilung in Antriebsachsen und virtuelle Achsen. Die dargestellten virtuellen Achsen bilden dabei die Elastizitäten ab, die bei der Lagerung der Gelenke entstehen. Das Modell erlaubt es weitere Achsen an beliebiger Position und Orientierung hinzuzufügen.

2.2.2 Fräskraftmodell

Für den Fräseinsatz wurde der oben beschriebene Roboter mit einer Spindel der Firma Precise ausgestattet (Abb. 2.3). Diese lässt sich mit unterschiedlichen Fräsern je nach Einsatzbedarf ausrüsten. Die Motorspindel erreicht eine Drehzahl bis zu 40000 Umdrehungen pro Minute. Die Berechnung der Fräskräfte basiert auf einer Materialabtrags-Simulation [Altintas and Ber, 2001]. Dabei wird eine Dixel-Darstellung des Werkstücks in 3 Raumrichtungen verwendet, wie in Abbildung 2.2 rechts unten dargestellt. Aus der Vorschubgeschwindigkeit, der Schnitttiefe und des verwendeten Fräasers können die entstehenden Fräskräfte und die Abtragsspandicke zum aktuellen Zeitpunkt berechnet werden.

Dieses Modell wurde mit dem Roboterstrukturmodell in eine Simulation gekoppelt. Die Kopplung wurde dabei über die Fräskraft F_{xyz} erreicht, die vom Fräskraftmodell ermittelt wird und in die Roboterdynamikgleichung 2.1 eingeht. Hieraus lassen sich nun zu jedem Zeitschritt die entstehenden Fräskräfte und die daraus resultierende Wechselwirkung mit dem Roboter berechnen. Die daraus ermittelten TCP Abweichungen von der Sollbahn-Vorgabe bilden die Grundlage für die weitere Kompensationsstrategie. Hierzu ist es erforderlich möglichst genaue Parameter der verwendeten Komponenten zu besitzen. Diese werden im folgenden Kapitel mit einem experimentellen Ansatz ermittelt.

3 Experimentelle Parameteridentifikation

Im folgenden wird ein experimentelles Verfahren zur Ermittlung der Parameter Steifigkeit und Dämpfung beschrieben. Diese beiden Parameter werden zur Modellierung der Roboterstruktur für die Simulation benötigt, um die wirkenden Drehmomente in den Achsen zu ermitteln (vgl. Gleichung 2.2). Ziel ist es dabei das Verhalten des realen Roboters so gut wie möglich durch die Parameteridentifikation in das Simulationsmodell abzubilden. Dabei wird zuerst eine statische Untersuchung des Roboters durchgeführt mit der Steifigkeitswerte der Achsen ermittelt werden. Anschließend werden dynamische Messungen durchgeführt mit der das Dämpfungsverhalten des Roboters in die Simulation übernommen wird. Zum Schluss wird das Verhalten des echten Roboters mit dem der Simulation verglichen und bewertet.

3.1 Steifigkeit

Zuerst wird eine statische Untersuchung des Roboters durchgeführt. Dabei wird mit Belastungs- und Entlastungszyklen eine Federkennlinie ermittelt. Neben den Elastizitäten in den Antriebsachsen des Roboters werden auch Kippsteifigkeiten untersucht. Diese lassen sich mit dem in Kapitel 2.2.1 beschriebenen Prinzip der virtuellen Achsen in dem Robotermodell abbilden.

3.1.1 Grundlagen

Als Rotationssteifigkeit k versteht man im Allgemeinen eine Größe, die den Widerstand eines Körpers gegen eine Verformung $\Delta\phi$ durch ein Moment M beschreibt:

$$k = \frac{M}{\Delta\phi}. \quad (3.1)$$

Als Kehrwert der Steifigkeit bezeichnet man die Nachgiebigkeit h . Generell wird versucht bei der Konstruktion von Industrierobotern die Nachgiebigkeit in den Gelenken so weit wie möglich zu vermeiden. Da sich diese aber nicht völlig eliminieren lassen, müssen sie für ein präzises Robotermodell berücksichtigt und dafür bestimmt werden. Die Steifigkeiten der einzelnen Komponenten des Roboters lassen sich durch Zug- und Druckversuche experimentell ermitteln [Weigold, 2008].

Das Konzept dieser Versuche um einen Steifigkeitswert zu bestimmen besteht darin, eine Kraft auf den Roboter aufzubringen und die damit verbundene Auslenkungen zu messen. Die Richtung der Kraft wird so gewählt, dass an der gerade betrachteten Achse, an der die Steifigkeit bestimmt werden soll, ein Moment wirkt. Durch Wegmessensensoren wird die Abdrängung des Roboters an mehreren Stellen gemessen. Dabei ist wichtig, dass die Messsensoren auf dem Roboter befestigt sind, damit nur die tatsächliche Veränderung der jeweiligen Achse gemessen wird. Die aufgebrachte Kraft und die daraus entstehenden Auslenkungen werden mit geeigneter Messtechnik synchronisiert aufgezeichnet. Die gemessene Kraft wird dann durch Multiplikation mit dem effektiven Hebelarm auf das wirkende Moment M des betrachteten Gelenks umgerechnet. Aus der gemessenen Auslenkungen Δx lässt sich die Veränderung des Gelenkwinkel $\Delta\phi$ errechnen. Diese beiden Werte lassen sich in einer Kraft-Verformungs-Kennlinie darstellen. Unter der Annahme einer linearen Federkennlinie lässt sich damit die Steifigkeit k_i der Gelenkachse i errechnen. Die Federkennlinie wird in den seltensten Fällen als exakt linear gemessen. Häufig treten Effekte wie Spiel oder Hysterese auf. Unter Spiel versteht man dabei den Bewegungsfreiraum in dem sich das Robotergelenk ohne Kraftübertragung frei bewegen lässt. Dieses entsteht meist aufgrund von Reibungen im Getriebe und Lager. Als Hysterese bezeichnet man, dass sich bei gleicher aufgebrachter Kraft mehrere Verformungszustände ergeben. Das hängt zum Beispiel davon ab, ob man sich gerade in einer Entlastungs- oder Belastungsphase befindet. Die beiden Effekte sind in Abbildung 3.1 dargestellt.

3.1.2 Versuchsaufbau und Durchführung

Die notwendigen Versuche zur Bestimmung der Parameter wurden am Institut PTW (Produktionsmanagement, Technologie und Werkzeugmaschinen) der TU Darmstadt durchgeführt. Zur Bestimmung der Steifigkeiten wurden jeweils für alle Roboterachsen Zug- und Druckversuche durchgeführt. Um das Prinzip dieser Versuche zu verdeutlichen wird im folgenden das Vorgehen zur Bestimmung der Rotationssteifigkeit um die 6te Antriebsachse beschrieben (vgl. Abb. 2.3 rechts, Achse z_5). Der zugehörige Versuchsaufbau ist in Abbildung 3.2 veranschaulicht. Als Werkzeug wurde dabei eine Kraftmessdose verwendet mit der eine Belastung auf den Roboter gegeben wurde. Diese Belastung wurde abwechselnd als Zug- und Druck-Kraft in mehreren Zyklen aufgebracht und gleichzeitig gemessen. Die maximale aufgebrachte Kraft

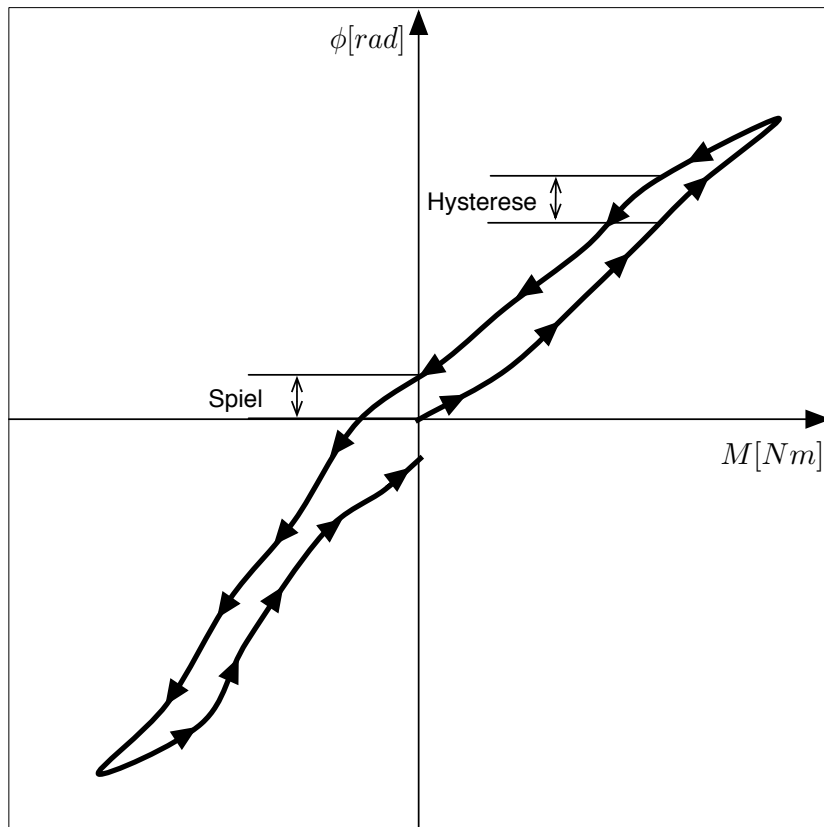


Abbildung 3.1: Kraft-Verformungs-Kennlinie eines Belastungs- und Entlastungszyklus mit Spiel und Hysterese

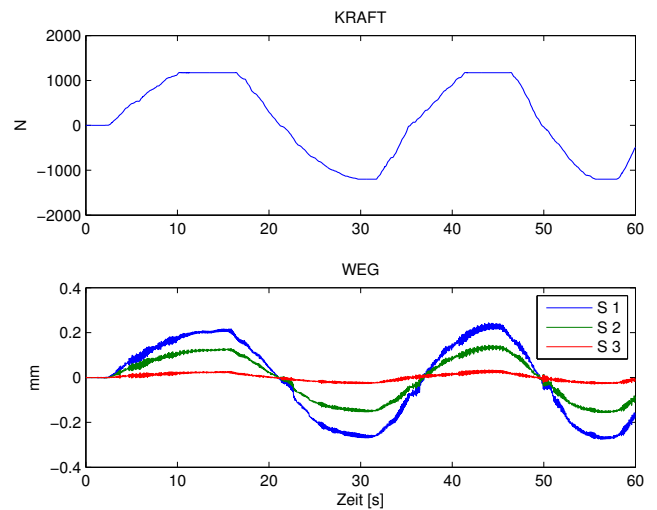
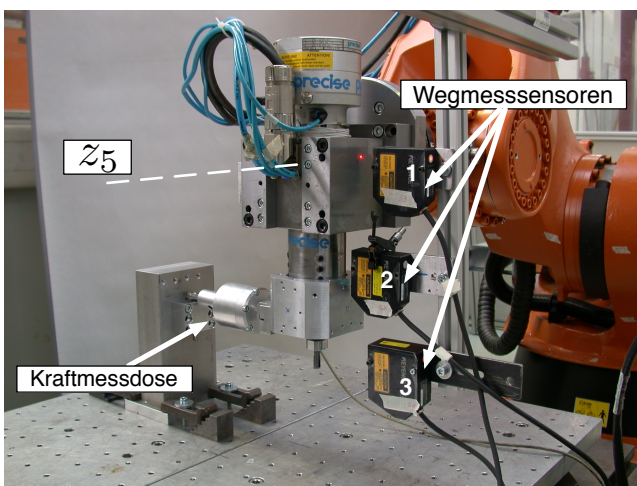


Abbildung 3.2: Versuchsaufbau zur Messung der Steifigkeit der 6ten Achse (links) und die zugehörige Auswertung (rechts)

lag dabei bei 1500 N. Zur Messung der Verformungen wurden hochauflösende Laser-Wegmesssensoren verwendet. Die Sensoren befinden sich hierbei auf einem Balken der unmittelbar vor der Drehachse auf dem Roboter befestigt ist. Damit wird sichergestellt, dass nur die Nachgiebigkeit der 6ten Achse gemessen wird. Durch Synchronisation der aufgetragenen Kraft und der gemessenen Abdrängung über der Zeit erhält man die in Abbildung 3.2 (rechts) dargestellten Messwerte. Im nächsten Schritt werden die aufgenommenen Messdaten auf das wirkende Moment und die entstandene Winkelveränderung umgerechnet. Die Winkelveränderung wird aus der aufgenommenen Abdrängungen und der Distanz der

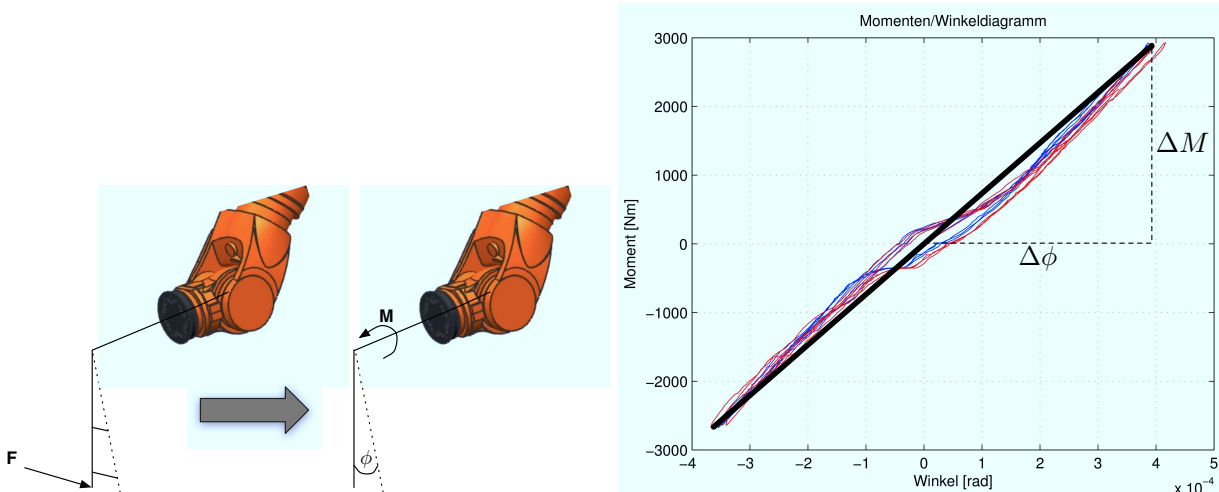


Abbildung 3.3: Umrechnung der Messdaten (links) und das entstehende Momenten/Winkeldiagramm (rechts)

Messsensoren von der Drehachse berechnet. Die aufgetragene Kraft wird durch den Hebelarm auf das wirkende Moment um Achse 6 umgerechnet (siehe Abb. 3.3 links). Mit den ermittelten Größen lässt sich ein Momenten/Winkeldiagramm (Abb. 3.3 rechts) zeichnen. Durch die Annahme eines linearen Kraft/Abweichungs Verhältnisses erhält man mit der Gleichung 3.1 den Steifigkeitswert k_6 . Die gemessenen Spiel- und Hysterese-Effekte wurden zum Anlass für eine Modellverfeinerung genommen, in der das Spiel der Getriebe ebenfalls modelliert wird. Nach Durchführung der Versuche für sowohl alle realen Roboterachsen, als auch für die virtuellen Achsen können die ermittelten Steifigkeitswerte (siehe Tabelle 3.1) direkt in das Modell integriert werden.

Gelenk	Rotationssteifigkeit	Kippsteifigkeit
1	5.6e+006	1.33e+007
2	8.0397e+006	2.61e+007
3	3.1205e+006	1.54e+006
4	7.1959e+005	1.93e+006
5	6.5862e+005	6.584e+005
6	3.1403e+005	1.899e+006

Tabelle 3.1: Rotations- und Kippsteifigkeiten des Roboters

3.2 Dämpfung

Nach der Bestimmung der Steifigkeitswerte für die jeweiligen Achsen wird im nächsten Schritt das dynamische Verhalten des Roboters untersucht. Jede sich in Bewegung befindliche Masse erzeugt Schwingungen. Diese Schwingungen werden in der Roboterdynamikgleichung 2.2 mit der Multiplikation einer Dämpfungsmatrix D mit der Geschwindigkeitsdifferenz $(\dot{\theta} - \dot{q})$ der Gelenke modelliert. Da die einzelnen Gelenke unterschiedlich große Schwingungen besitzen wird im folgenden für jedes Gelenk i die Dämpfungskonstante d_i der Diagonalmatrix D bestimmt. In [Hözl, 1994] wird ein Verfahren zur experimentellen Bestimmung der Dämpfungsparameter der einzelnen Gelenke eines PUMA Roboters beschrieben. Dies wird realisiert, indem die jeweils nicht berücksichtigten Roboterglieder fest eingespannt werden um ausschließlich die Schwingungen der untersuchten Achsen zu messen. Da dieses Verfahren bei einem Roboter mit 6 Achsen und den berücksichtigten virtuellen Achsen nicht realisierbar ist wird ein modifiziertes Verfahren angewandt. Dabei werden die Schwingungen des gesamten Roboters untersucht und anschließend wird im Modell durch Anpassung der Parameter

das Verhalten nachgebildet. Hierzu wird eine Modalanalyse mit dem Roboter durchgeführt um das dynamische Verhalten zu charakterisieren. Dabei wird die Roboterstruktur mit einem Impulshammer angeregt. Die daraus resultierenden Schwingungen des Roboters werden mit einem 3-achsigen Beschleunigungssensor am vorderen Werkzeugaufsatz gemessen. Diese Experimente werden wie schon bei den statischen Untersuchungen in die 3 Raumrichtungen des jeweils aufgebrachtten Impulses aufgeteilt. Anschließend wird aus den aufgenommenen Messdaten der Übertragungsfrequenzgang zwischen Anregung und Antwort berechnet. Dieses Experiment wird unter den gleichen Bedingungen in der Simulation durchgeführt wo ebenfalls der Übertragungsfrequenzgang gebildet wird. Dazu werden die aus Kapitel 3.1 ermittelten Steifigkeitswerte verwendet und für die Dämpfung wird als Initialwert in allen Achsen der gleiche Wert gesetzt. Danach werden die Dämpfungsparameter des Modells durch heuristische Variation im Frequenzbereich von 0 bis 50 Hz abgeglichen und an die Spitzenwerte des Experiments angepasst. In Abbildung 3.4 sind die Übertragungsfrequenzgänge H_{xx} , H_{yy} und H_{zz} des Experiments und der Simulation mit abgeglichenen Werten übereinander gezeichnet. Im Allgemeinen lässt sich eine gute Übereinstimmung des dynamischen Verhaltens beobachten. Die Höhen der einzelnen Spitzen im Modell konnten gut mit der Anpassung der Dämpfungskonstanten d_i passend skaliert werden.

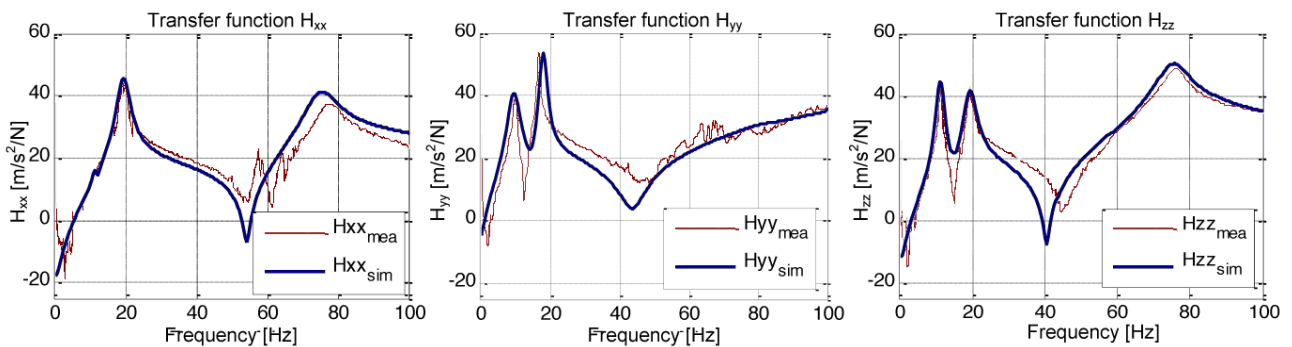


Abbildung 3.4: Experimenteller und simulierter Übertragungsfrequenzgang

3.3 Validierung anhand der Gesamtsteifigkeit im Arbeitsraum

Nach der Ermittlung der Steifigkeits- und Dämpfungsparameter mit den oben beschriebenen Verfahren sind die benötigten Parameter für die Dynamikmodellierung vorhanden. Um das Modell mit den ermittelten Werten zu evaluieren, wurde eine weitere Versuchsreihe durchgeführt in der das Verhalten des realen Roboters mit dem in der Simulation verglichen wird. Dabei werden Steifigkeitswerte des Roboters an mehreren Positionen im Arbeitsraum ermittelt. Diesmal wird aber nicht die Steifigkeit einzelner Komponenten betrachtet, sondern die Gesamtsteifigkeit des Roboters. Dafür werden im relevanten Arbeitsraum 9 Punkte definiert an denen jeweils in x-,y- und z-Richtung ein Steifigkeitswert ermittelt wird (vgl. Abb. 3.5 (links)). Der Kraftangriffspunkt und die Messpunkte befanden sich dabei immer am TCP und waren in die gleiche Richtung orientiert. Das Experiment wird anschließend unter gleichen Bedingungen in der Simulation nachgebildet. Im Anschluss werden die simulierten Werte mit denen des realen Roboters gegenübergestellt (Abb. 3.5 (rechts), 3.6), um eine Aussage über die Qualität der ermittelten Parameter zu treffen. Um den Einfluss der virtuellen Achsen zu untersuchen wurden die Experimente in der Simulation einmal mit den aktivierten virtuellen Achsen und einmal nur mit den realen Roboterachsen durchgeführt. Dieser Einfluss ist in den Auswertungen mit einem weißen Balken kenntlich gemacht. Der Vergleich der Werte einmal mit und einmal ohne Kippsteifigkeiten lässt erkennen, dass die Modellierung der virtuellen Achsen signifikant dazu beigetragen hat um das reale Roboterverhalten präzise zu modellieren. Vor allem in y-Richtung haben die Kippsteifigkeiten einen sehr hohen Einfluss. Hier hat sich der Steifigkeitswert etwa gedrittelt mit dem Einsatz der virtuellen Achsen. Dies lässt sich mit dem Zusammenwirken der Nachgiebigkeiten in Kipprichtung der Achsen 2, 3, 4 und 5 erklären. Im Allgemeinen lässt sich ein korrespondierendes Verhalten der Simulation mit dem des realen Roboters feststellen. Auch die Werte aus der Simulation mit aktiven Kippachsen liegen nahe bei den realen Messwerten. An den mittleren Messpunkten 2, 5 und 8 liegt in x- und z-Richtung ein höherer Wert vor als an den jeweils benachbarten äußeren Punkten. Die hängt damit zusammen, dass an den mittleren Punkten ein geringerer Einfluss der Kippsteifigkeiten wirkt. Anhand der Auswertung in z- und y-Richtung erkennt man die immer kleiner werdenden Steifigkeiten je weiter der Messpunkt von der Roboterbasis entfernt ist, was sich durch den größeren Hebelarm begründen lässt.

Abschließend lässt sich festhalten, dass das Verhalten des Roboters in die Simulation mit einer gewissen Genauigkeit abgebildet wurde. Eine exaktere Parameteridentifikation wäre zum Beispiel mit Optimierungsverfahren möglich. Hierbei können die ermittelten Parameter als Startwerte verwendet werden. Mit den identifizierten Parametern ist es nun mög-

lich, die für die modellbasierte Offline-Kompensation notwendige Simulation der Abdrängung (vgl. 2.1 Schritt 4) für den Roboter KUKA KR 210 durchzuführen.

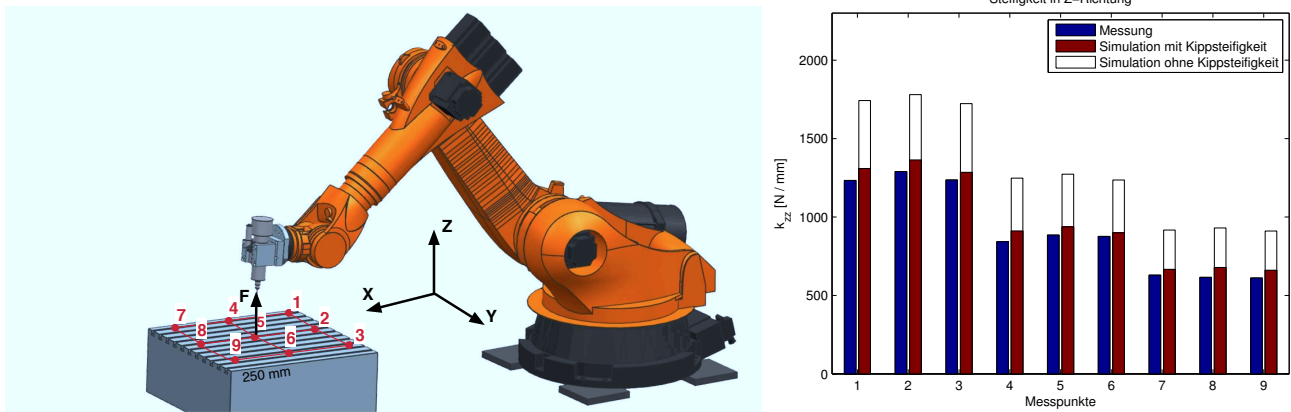


Abbildung 3.5: Messpunkte im relevanten Arbeitsraum (links) und Auswertung der Steifigkeiten in z-Richtung (rechts)

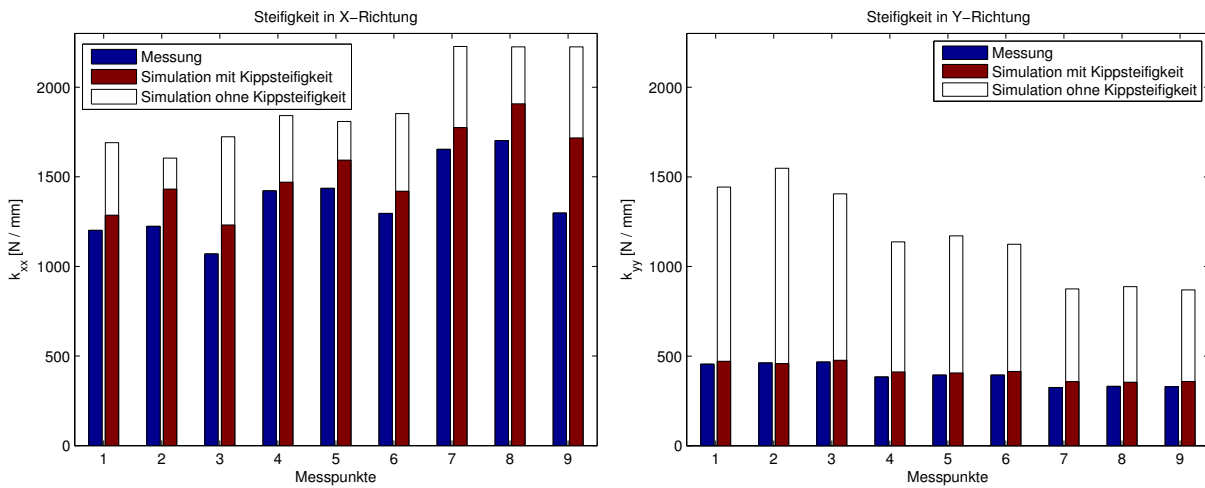


Abbildung 3.6: Auswertung der Steifigkeiten in x- (links) und y-Richtung (rechts)

4 Trajektorienplanung

Unter Trajektorienplanung versteht man die Generierung von kontinuierlichen zeitabhängigen Bewegungsbahnen mit zeitlichen Beschränkungen. Sie ist ein wesentlicher Prozessbaustein in der modellbasierten Offline-Kompensation (vgl. Abb. 2.1). Die Hauptverantwortung liegt bei dem betrachteten Anwendungsgebiet darin eine Trajektorie zu erstellen auf der die Prozesskräfte simuliert werden können. Da hierbei sehr große Bahnen zum Einsatz kommen steht eine stabile und automatische Generierung der Trajektorie im Vordergrund. Im folgenden wird zunächst der Stand der Technik beim Planen von Trajektorien vorgestellt. Dabei werden verschiedene Möglichkeiten betrachtet, die bei der Trajektorienplanung verwendet werden. Anschließend wird aus den vorgestellten Varianten eine für das Einsatzgebiet passende ausgewählt und implementiert.

4.1 Stand der Technik

Das Erstellen einer Trajektorie lässt sich in mehrere Bausteine unterteilen die verschieden miteinander verknüpft werden können. Dabei hat man jeweils immer mehrere Varianten zur Auswahl um die Trajektorie an die jeweiligen Anforderungen anzupassen. Den Prozess der Trajektorienplanung (vgl. Abb. 4.1) kann man in zwei wesentliche Schritte aufteilen. Zu Beginn wird eine kontinuierliche räumliche Punktfolge erstellt, die der TCP des Roboters abfahren soll. Dabei beschränkt man sich auf die rein geometrische Beschreibung der Bahn. Die Punkte definieren dabei sowohl die Position als auch die zugehörige Orientierung des TCP. Im nächsten Schritt wird dann aus der Bahn durch die Festlegung von zeitlichen Regulierungen eine Trajektorie erstellt. Nach Durchführung dieser Schritte erhält man eine Trajektorie bei der zu jedem Zeitpunkt die Position, Geschwindigkeit und Beschleunigung des TCP definiert sind.



Abbildung 4.1: Ein- und Ausgabedaten der Trajektorienplanung

In den folgenden Unterkapiteln wird der aktuelle Stand der Technik der einzelnen Möglichkeiten zur Spezifikation einer Trajektorie näher vorgestellt. Zuerst werden Methoden zur geometrischen Beschreibung der Bahn betrachtet und anschließend wird auf die zeitliche Regulierung der Bewegung eingegangen.

4.1.1 Geometrie der Bahn

Eine Bahn ist eine kontinuierliche räumliche Punktfolge mit Position und Orientierung. Im Gegensatz zur Trajektorie spielen zeitliche Abhängigkeiten hierbei keine Rolle. Als Ausgangssituation ist eine vorgegebene Anzahl an Punkten gegeben, die vom TCP des Roboters nacheinander abgefahren werden sollen. Das Ziel ist es, aus diesen Punkten eine geeignete mathematische Beschreibung für die Roboterbahn zu finden, so dass die Position und Orientierung des TCP kontinuierlich entlang der Bahn fest definiert ist. Hierzu werden die Roboterbahnen mit geometrischen Funktionen beschrieben die im folgenden näher vorgestellt werden.

Koordinatensysteme

Die Planung von Bahnen für einen Roboter lässt sich in verschiedenen Koordinatensystemen durchführen. Die am weitesten gebräuchlichen sind dabei die Gelenkwinkelkoordinaten des Roboters und die kartesischen Koordinaten (z.B. die des Werkstücks). Häufig ist es der Fall das beide Koordinatensysteme kombiniert zum Einsatz kommen. Die Randpunkte werden dann zum Beispiel in kartesischen Koordinaten vorgegeben und der Weg zwischen den Punkten wird anschließend in Gelenkwinkel interpoliert. In Gelenkwinkelkoordinaten wird eine Roboterstellung mit den Gelenkwinkeln

$$[q_1, q_2, q_3, q_4, q_5, q_6] \quad (4.1)$$

beschrieben. Hierbei hat man eine sehr gute Kontrolle über die Geschwindigkeiten und Beschleunigungen der Gelenke. Dadurch kann sehr einfach vermieden werden, dass die maximalen Gelenkgrenzen überschritten werden. Ein weiterer

Vorteil ist der niedrigere Berechnungsaufwand. Die inverse Kinematik muss nur für die Randpunkte berechnet werden. Die weiteren Zwischenpunkte können in Gelenkkoordinaten interpoliert werden, die der Roboter direkt ausführen kann. Deswegen sind sie auch für Anwendungen geeignet, in denen die Trajektorie in Echtzeit berechnet werden soll. Ein wesentlicher Nachteil hierbei ist das es nicht möglich ist eine fest vorgeschriebene Kontur abzufahren. Dadurch das die Interpolation in Gelenkwinkeln stattfindet kann die Position der Zwischenpunkte im Raum schwer kontrolliert werden. Das Abfahren einer Geraden oder eines Kreises ist hiermit nicht möglich. Ebenfalls ist die Kollisionserkennung mit Gegenständen nicht ohne weiteres umsetzbar.

In kartesischen Koordinaten wird eine Roboterstellung durch

$$[x, y, z, \phi, \theta, \psi] \quad (4.2)$$

beschrieben. Hierbei entspricht x, y, z den drei Raumkoordinaten und ϕ, θ, ψ den Eulerwinkeln, mit denen die Orientierung des TCP beschrieben wird (siehe Anhang A). Der größte Nachteil hierbei ist der sich ergebende höhere Berechnungsaufwand. Zum einen müssen die vielen Punkte des Pfades mit der inversen Kinematik in Gelenkwinkel umgerechnet werden. Des Weiteren müssen gegebenenfalls auftretende Singularitäten abgefangen werden und es muss sichergestellt werden das die erstellte Trajektorie keine Geschwindigkeits- und Beschleunigungsgrenzen der Gelenke überschreitet. Vorteile sind hingegen die einfache Visualisierung und Kollisionskontrolle der Trajektorie. Des Weiteren kann die Trajektorie zum großen Teil unabhängig vom speziellen Roboter erstellt werden und dieselbe Trajektorie kann von unterschiedlichen Robotern ausgeführt werden. Das Haupteinsatzgebiet von kartesischen Koordinaten besteht darin, wenn die feste Einhaltung einer Bahn beim abfahren nötig ist und man genügend Rechenaufwand zur Verfügung hat. Viele der im folgenden vorgestellten Methoden und Varianten können sowohl für die Planung in Gelenkwinkelkoordinaten als auch für kartesische Koordinaten verwendet werden.

Geometrische Elemente

Die geometrische Beschreibung einer Bahn wird im folgenden durch die Funktion $p(u)$ mit der unabhängigen Variable u beschrieben. Bei der Betrachtung von größeren Bahnen ist es sinnvoll die Funktion in mehrere Segmente aufzuteilen [Biagiotti, 2009]

$$p(u) = p_k(u), \quad k = 0, \dots, n - 1. \quad (4.3)$$

Die einzelnen Abschnitte können dabei durch beliebige Polynome beschrieben werden. Diese können einfache geometrische Grundelemente sein, wie etwa Geraden oder Kreise. Zur flexibleren Beschreibung kommen häufig Kurven aus der Computergrafik, wie etwa Bézierkurven oder B-Splines zum Einsatz. Dabei ist es wichtig auf die geometrische Stetigkeit zwischen den Segmenten zu achten, um einen sanften Übergang zu gewährleisten. Die Ordnung der Polynome wird dabei abhängig vom Bahnverlauf und den gewünschten Übergangsbedingungen zwischen den Abschnitten gewählt.

Überschleifen

Das Überschleifen wird häufig in der Bahnplanung eingesetzt damit der Roboter an den Übergangspunkten nicht zum Ruhestand kommt und ein sanfter Übergang an den Ecken entsteht [Weber, 2007]. Dabei werden die vorgegeben Punkte nicht exakt angefahren (vgl. Abb. 4.2 links), sondern die Ecken werden mit einer Kurve abgerundet. Dadurch muss der Roboter bei Ecken nicht auf Null abgebremst werden, wodurch eine ruckartige Bewegung vermieden wird. Die Auswirkungen auf das Geschwindigkeitsprofil kann man Abbildung 4.2 rechts entnehmen. Dies ist vor allem dann von Vorteil, wenn für ein Anwendungsgebiet eine flüssige Bewegung des Roboters notwendig ist. Ebenfalls wird dadurch die Dauer der Bewegung und auch die Länge der Bahn verringert. Man unterscheidet im Allgemeinen zwischen dem Geschwindigkeitsüberschleifen und dem Positionsüberschleifen. Beim Geschwindigkeitsüberschleifen wird eine Geschwindigkeit vorgegeben, bei deren Unterschreiten das Überschleifen beginnt. Beim Positionsüberschleifen hingegen ist eine Distanz vom Eckpunkt festgelegt, der so genannte Überschleifradius, der den Beginn des Überschleifprozesses eindeutig definiert. Ebenfalls ist auch eine Kombination beider Varianten möglich. Durch das Überschleifen wird die vorgegebene Bahn nicht mehr exakt abgefahren wodurch ein Fehler entsteht. Hierbei gilt es zwischen dem Ausmaß der entstehenden Bahnabweichung und der Notwendigkeit einer flüssigen TCP Bewegung abzuwägen.

Orientierung

Neben der Lage einer Position im Raum spielt bei Robotern auch immer die Orientierung in der diese Position angefahren wird eine wichtige Rolle. Diese werden in kartesischen Koordinaten mit Eulerwinkeln angegeben oder können bei einer Planung in Gelenkkoordinaten mit der inversen Kinematik in Eulerwinkel umgerechnet werden. Da die Orientierung nur an den Eckpunkten gegeben ist, muss sie dazwischen interpoliert werden um einen sanften Übergang zu gewährleisten. Im wesentlichen unterscheidet man hierbei 3 Interpolationsvarianten:

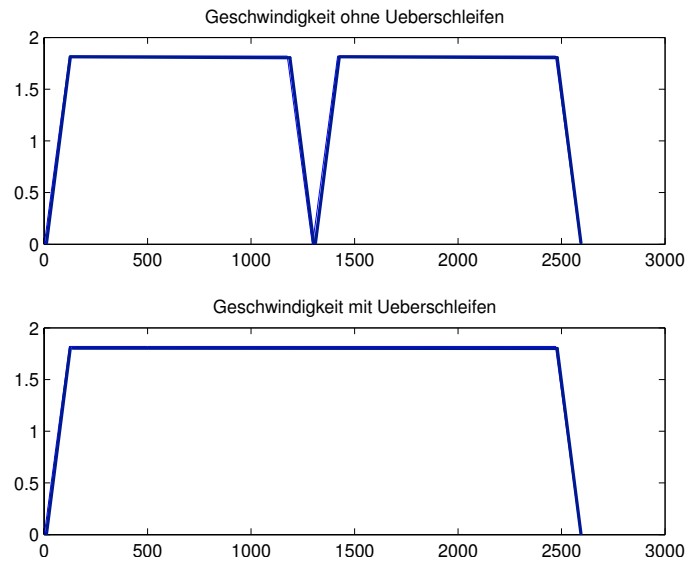
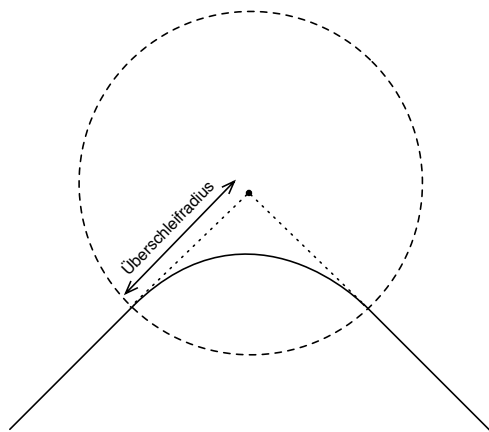


Abbildung 4.2: Überschleifen an einem Übergang zwischen zwei Geradenstücken (links) und der Vergleich zwischen eines Geschwindigkeitsprofils ohne Überschleifen (rechts oben) und mit Überschleifen (rechts unten)

- Interpolation nur in dem Überschleifradius an den Ecken,
- Erstellen eines eigenen Radius in dem die Orientierung interpoliert wird,
- Interpolation der Orientierung über die gesamte Bahn.

Die einfachste Möglichkeit dies umzusetzen besteht darin, die Eulerwinkel wie bei der Position abschnittsweise mit Polynomen zu interpolieren. Eine weitere Variante die Orientierungsänderung zwischen zwei Positionen zu beschreiben besteht darin eine Rotationsmatrix $R(\theta)$ aufzustellen die zwischen zwei Orientierung interpoliert. θ ist dabei ebenfalls wie u eine unabhängige Variable, die später mit einem Bewegungsgesetz berechnet wird. Das Konzept dieser Methode besteht im wesentlichen darin, die Orientierungsänderung von der Matrix R_1 nach R_2 durch eine Winkelveränderung um einen festen Vektor im Raum zu beschreiben (siehe [Biagiotti, 2009]).

4.1.2 Zeitliche Regulierung der Bewegung

Nachdem die Geometrie für die Bahn erstellt wurde erfolgt anschließend eine Reparametrisierung der Funktion $p(u)$ in der die Zeit t eingeführt wird:

$$\tilde{p}(t) = (p \circ u)(t) \quad (4.4)$$

Dazu wird der bisherige Parameter u abhängig von der Zeit t und man erhält eine Funktion $u=u(t)$ die den Zusammenhang der fortlaufenden Zeit t und der bisherigen Parametrisierung u angibt (vgl. Abb. 4.3). Bei der Planung in Gelenkwinkelkoordinaten wird häufig auch die Beziehung $u = t$ verwendet, um den Verlauf der Gelenkwinkel direkt mit der Zeit zu erstellen. Bei der Wahl der Funktion $u(t)$ werden Bedingungen und Bewegungsprofile berücksichtigt, denen die Trajektorie genügen soll. Diese werden im folgenden näher betrachtet.

Bedingungen und Beschränkungen

Bei der Erstellung von Trajektorien können Restriktionen zur zeitlichen Regulierung der Bewegung festgelegt werden. Jede dieser Vorschriften ist dabei an eine der folgenden Größe geknüpft:

- Geschwindigkeit $\dot{\tilde{p}}(t)$, $[\frac{m}{s}]$
- Beschleunigung $\ddot{\tilde{p}}(t)$, $[\frac{m}{s^2}]$
- Ruck $\tilde{p}^{(3)}(t)$, $[\frac{m}{s^3}]$

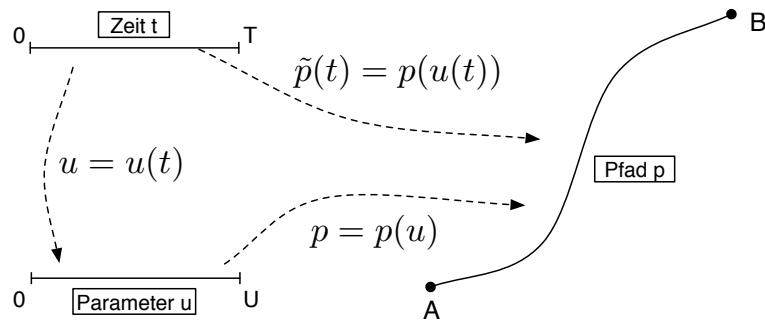


Abbildung 4.3: Parametrisierung der Trajektorie

Diese 3 Größen können jeweils durch ableiten nach der Zeit nacheinander ermittelt werden. Als Beschränkungen werden die kinematischen Grenzen des Roboters festgelegt. Da jeder Roboter nur einen bestimmten Bereich an Geschwindigkeiten und Beschleunigungen erreichen kann werden dafür obere und untere Grenzen festgelegt, die auf keinen Fall verletzt werden dürfen. Häufig sind diese Grenzen nur in Gelenkwinkelkoordinaten in der Form $\dot{q}_{min} \leq \dot{q}(t) \leq \dot{q}_{max}$ festgelegt. Das heißt bei einer Planung im kartesischen Raum benötigt man eine Umrechnung mit der inversen Jacobi-Matrix durch

$$\dot{q} = J^{-1} \cdot \begin{pmatrix} v \\ \omega \end{pmatrix}. \quad (4.5)$$

In den meisten Fällen werden neben den Beschränkungen je nach Anwendungsfall noch Bedingungen definiert die eingehalten werden sollen. Diese sind niedriger als die Beschränkungen des Roboters. Hier werden Soll- und Richtwerte definiert, die bei der Generierung der Trajektorie berücksichtigt werden sollen.

Bewegungsprofile

Um die Bedingungen der Trajektorie nicht einzeln für jeden Punkt festlegen zu müssen werden häufig allgemeine Vorschriften verwendet, die für immer wieder auftretende Situationen gelten. Diese Situationen können zum Beispiel bestimmte Roboterpositionen oder geometrische Eigenschaften der Bahn sein. Dies ist vor allem bei der automatischen Generierung von Trajektorien wichtig, wo man nicht jeden einzelnen Punkt manuell festlegen möchte. Häufig greift man dabei auf die im vorigen Kapitel definierten Restriktionen zurück, die als Randbedingungen zum Einsatz kommen. Mit diesen Vorschriften wird beschreiben, wie der Geschwindigkeits- und Beschleunigungsverlauf an bestimmten Stellen generiert werden soll. Je nach Anforderungsgebiet gilt es dafür geeignete Profile auszuwählen. In der Praxis häufig verwendete Profile sind:

- **Konstanter Geschwindigkeitsverlauf:** Die Bahn soll mit konstanter Geschwindigkeit $|\dot{p}(t)| = v_c$ abgefahren werden. Dies ist jedoch nur bei geringeren Geschwindigkeiten möglich.
- **Schnelle Bewegungsdurchführung:** Der Pfad soll mit der minimalen Gesamtzeit abgefahren werden. Dazu werden die maximale Geschwindigkeit v_{max} , maximale Beschleunigung a_{max} und das maximalen Abbremsen a_{min} verwendet.
- **Ruckfreie Bewegung:** Eine sanfte Bewegung des TCP soll erreicht werden. Bei einem linearen Geschwindigkeitsprofil kann es beim Wechsel von der Beschleunigungsphase zur konstanten Geschwindigkeitsphase zu ruckartigen Beschleunigungen kommen (vgl. Abb. 4.4 links). Diese lassen sich zum Beispiel mit double-S Geschwindigkeitsprofilen vermeiden (vgl. Abb. 4.4 rechts).

Zu beachten ist hierbei das die definierten Vorschriften fortlaufend über die Bahn definiert sind und je nach Anwendung keinen Sprung in Geschwindigkeit, Beschleunigung oder Ruck ergeben.

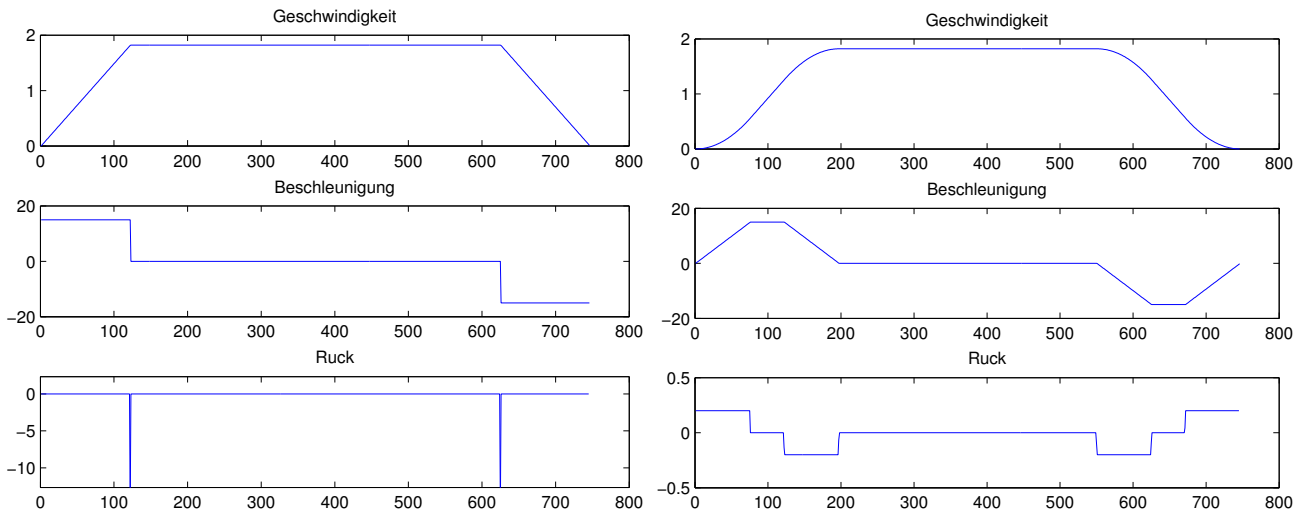


Abbildung 4.4: Vergleich eines Linearen Geschwindigkeitsprofils (links) mit einem double-S Geschwindigkeitsprofil (rechts)

4.2 Erstellung einer Trajektorienplanung

Für die modellbasierte Offline-Kompensation ist die Trajektorienplanung ein wesentlicher Grundbaustein, der zum Durchführen der Strategie notwendig ist. Nach der Generierung der Trajektorie im Vorfeld der Simulation kommt sie anschließend bei der Berechnung der Fräskräfte, der anschließenden Kompensation und beim Export an den realen Roboter zum Einsatz. Mit dem Ziel eine hohe Bearbeitungsgenauigkeit durch die Offline-Kompensation zu erreichen muss dabei gewährleistet werden, dass keine zusätzliche Fehlerabweichung durch die Trajektorienplanung entsteht. Im folgenden Abschnitt wird aus den im vorigem Kapitel genannten Möglichkeiten eine geeignete Variante für das Anwendungsgebiet ausgewählt. Danach wird auf die Implementierung und Umsetzung dieser Variante eingegangen. Zum Schluss wird diese an einem Referenzdatensatz getestet.

4.2.1 Auswahl geeigneter Variante

Beim Fräsen mit dem Industrieroboter muss die verwendete Trajektorie gewissen Anforderungen entsprechen, damit eine hohe Bearbeitungsgenauigkeit erreicht werden kann. Die wichtigsten Anforderungen beim Fräsen sind

- das Abfahren der exakten Bahnkonturen,
- die Vorschubgeschwindigkeit beim Fräsen möglichst konstant zu halten, damit ein gleichmäßiges Abtragen des Materials erreicht wird und
- eine sanfte Bewegung zu gewährleisten, da durch ruckartige Beschleunigungen zusätzliche Schwingungen die Bearbeitungsqualität reduzieren können.

Aus den hier genannten Anforderungen und den in Abschnitt 4.1 vorgestellten Möglichkeiten zur Trajektorienplanung wurde eine geeignete Kombination zur Erstellung von Trajektorien ausgewählt. Zu Beginn wurde dabei das Koordinatensystem ausgewählt. Da exakte Bahnkonturen abzufahren sind und dies in Gelenkwinkelkoordinaten nicht möglich ist, wird die Planung in kartesischen Koordinaten durchgeführt. Der dadurch entstehende höhere Rechenaufwand hat keine weiteren Folgen, da die Trajektorienplanung nicht in Echtzeit durchgeführt werden muss. Der Ablauf der Trajektorienplanung lässt sich nun wie in Abbildung 4.5 beschreiben. Als Eingabedaten werden dazu die aus dem CAM Programm generierten Randpunkte der Bahn verwendet. Diese bestehen sowohl aus der Position und Orientierung des TCP. Aus diesen wird im nächsten Schritt eine kontinuierliche geometrische Beschreibung der Bahn erstellt. Dazu wird die Bahn, wie unter Abschnitt 4.1.1 beschrieben, in mehrere Segmente aufgeteilt, um die geometrische Beschreibung zu vereinfachen. In jedem Segment wird dabei ein geometrisches Element der Bahn beschrieben. Als geometrische Elemente werden dazu Gerade und Bézierkurven verwendet, wodurch ein Großteil aller relevanten Bahnverläufe im Raum abgedeckt wird. Dabei wird darauf geachtet das ein sanfter Übergang zwischen den geometrischen Elementen entsteht, damit ein flüssiger Bewegungsverlauf darauf möglich ist. Die Bézierkurve wird ebenfalls zur Beschreibung des Überschleifabschnittes

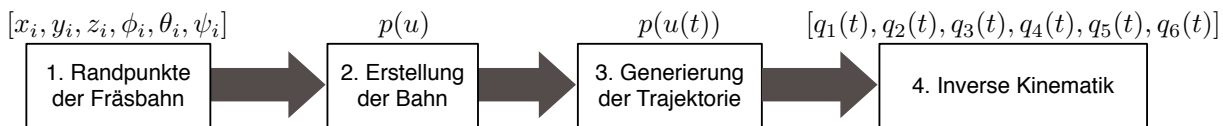


Abbildung 4.5: Schritte bei der Trajektorienplanung

verwendet. Hierbei kann vom Anwender spezifiziert werden, ob das Überschleifen verwendet werden soll und, falls ja, wie hoch der Überschleifradius dabei sein soll. Hierbei wurde das reine Positionsüberschleifen, wie in Abschnitt 4.1.1 beschrieben, verwendet. Dadurch wird gewährleistet, dass an jeder Ecke der gleiche Überschleifradius benutzt wird und das Bauteil gleichmäßig bearbeitet wird. Eine weitere Spezifikation, die vom jeweiligen Anwender durchgeführt werden kann, ist die Auswahl der Variante zur Orientierungsinterpolation. Da dies vom Bauteil abhängig ist, wurde hier eine Auswahlmöglichkeit zwischen einer Interpolation nur an den Eckpunkten und einer Interpolation über die gesamte Bahn integriert. Im nächsten Schritt wurde die Bahn mit der Zeit reparametrisiert (siehe 4.1.2). Dabei wurde ein konstantes Geschwindigkeitsprofil für das Fräsen gewählt. Damit schließt man aus, dass durch Brems- oder Beschleunigungsvorgänge weitere Ungenauigkeiten bei der Bearbeitung entstehen. Ein wichtiger Punkt den es dabei zu beachten galt, war die Möglichkeit einer Überschreitung der Gelenkgrenzwerte zu untersuchen. Hierzu wurde die Jacobi Matrix des Roboters nach [Spong, 2006] aufgestellt und die Gelenkgeschwindigkeit mit Gleichung 4.5 berechnet. Die Berechnungen haben ergeben das bei dem verwendeten Arbeitsraum und den eher geringen Geschwindigkeiten im Bereich von $0.05 \frac{m}{s}$ keine Überschreitungen der vom Hersteller vorgegebenen Grenzen auftreten. Anhand dieser Ergebnisse wurde auf eine weitere Implementierung einer Abstufung der Geschwindigkeit verzichtet. Nach der Reparametrisierung ist zu jedem Zeitpunkt die Position und Orientierung des TCP definiert. Da die Simulation und Kompensation der Strategie auf Gelenkwinkel basiert, wird im letzten Schritt eine Umrechnung mit der inversen Kinematik in analytischer Form durchgeführt (siehe Anhang B).

4.2.2 Implementierung

Das im vorigen Kapitel beschriebene Verfahren wurde zur Erweiterung der unter Kapitel 2 vorgestellten modellbasierten Offline-Kompensation in eine vorhandene Programmstruktur implementiert. Die zu fräsenden Bauteile wurden dabei mit der CAD/CAM Software Unigraphics NX¹ erstellt. Damit ist eine einfache Konstruktion von Bauteilen möglich. Des Weiteren ist durch Festlegung des Ausgangsbauteils und Bearbeitungsparametern das Erstellen einer Abtragsbahn möglich. Deren Randpunkte werden in der abzufahrenden Reihenfolge exportiert und in MATLAB² eingelesen, wo die Trajektorienplanung, Simulation und Kompensation stattfindet. Da bei der Simulation rechenintensive Berechnungen durchgeführt werden wurde mit einer MEX-Schnittstelle eine C++-Bibliothek zur Simulation von Mehrkörpersystemen integriert. Vor der Trajektoriengenerierung müssen noch wichtige Prozessparameter für die aktuelle Trajektorie festgelegt werden. Wichtige Prozessparameter die individuell verändert werden können sind dabei:

- Größe des Radius R mit dem an den Ecken die Überschleifkurve erstellt wird.
- Interpolationstakt T_{IPO} der Robotersteuerung.
- Konstante Geschwindigkeit v_c mit der die Bahn abgefahren werden soll.

Zu Beginn der Trajektorienplanung findet eine Aufteilung der Bahn in einzelne Segmente statt. In jedem Segment wird dabei entweder eine Gerade oder eine Bézierkurve (siehe Abb. 4.6) beschrieben. Diese können beliebig angeordnet werden, womit zum Beispiel auch Kreise oder Ellipsen möglich sind. Die Segmente mit Geraden zwischen zwei Punkten werden mit

$$p(u) = p_0 + (p_1 - p_0)u, \quad 0 \leq u \leq 1 \quad (4.6)$$

beschrieben. Die Bézierkurven 5. Grades lassen sich mit

$$b(u) = \sum_{j=0}^5 B_j^5(u) p_j, \quad 0 \leq u \leq 1 \quad (4.7)$$

und dem zugehörigen Bernsteinpolynom

$$B_j^5(u) = \binom{5}{j} u^j (1-u)^{5-j} \quad (4.8)$$

¹ <http://www.Siemens.com/NX>

² <http://www.mathworks.de/>

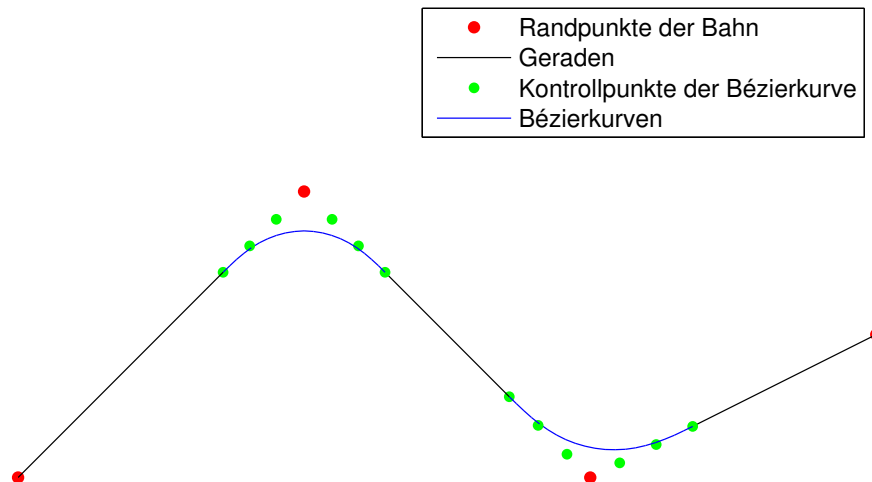


Abbildung 4.6: Aufbau einer Trajektorie

darstellen. Die zugehörigen Kontrollpunkte p_j (siehe Grafik 4.6) lassen sich mit der Bedingung an eine C^2 Stetigkeit zwischen den Segmenten aus den Übergangsbedingungen bestimmen (siehe dazu [Biagiotti, 2009]). Zur Parametrisierung der ganzen Bahn wird die Variable \hat{u} verwendet die von 0 bis zur Länge der Bahn verläuft. Daraus kann die Variable u für das aktuelle Segment durch

$$u = \frac{\hat{u} - \hat{u}_k}{\lambda_k} \quad (4.9)$$

berechnet werden. λ_k entspricht dabei der Länge des betrachteten Segments und \hat{u}_k ist die Gesamtlänge der Bahn bis zu Beginn des aktuellen Segments. Die Reparametrisierung der Variable \hat{u} gemäß Abbildung 4.3 erfolgt in dem Fall bei einer konstanten Geschwindigkeit v_c mit:

$$\hat{u}_{k+1} = \hat{u}_k + v_c \cdot T_{IPO}. \quad (4.10)$$

Da nur Beschleunigungs- und Abbremsvorgänge zu Beginn und Ende einer Trajektorie auftreten werden diese gesondert mit einem double-S Geschwindigkeitsprofil erstellt. Hiermit ist es nun möglich aus konstruierten Bauteilen eine geometrische Abtragsbahn zu erstellen und im nächsten Schritt diese mit einem konstanten Geschwindigkeitsprofil in eine Trajektorie zu reparametrisieren auf der die weiteren Schritte der modellbasierten Offline-Kompensation stattfinden.

4.2.3 Auswertung

Um die Trajektorienplanung auf ihre Anwendbarkeit zu testen, wurde sie mit einem realen Bauteil durchgeführt. Das Bauteil enthält dabei drei Quadrate und einen Kreis, die jeweils in zwei Vertiefungen nacheinander aus einem quaderförmigen Aluminiumblock heraus gefräst werden sollen. Dabei wurde mit einem CAD/CAM System das Bauteil erstellt und anschließend mit oben beschriebenem Verfahren eine Trajektorie generiert. Die einzelnen Zwischenschritte wurden in Abbildung 4.7 visualisiert. In Abbildung 4.8 sieht man das dazugehörige Geschwindigkeitsprofil aufgeteilt in die Raumrichtungen mit einer konstanten Geschwindigkeit von $0.1 \frac{m}{s}$.

Wie das Beispiel zeigt, ist es nun möglich die modellbasierte Offline-Kompensation auf komplexen Bauteilen anzuwenden. Die implementierte Trajektorienplanung bietet eine automatische und robuste Möglichkeit, um Trajektorien unter Berücksichtigung wichtiger Prozessparameter zu generieren. Des Weiteren ist es jetzt möglich, die Orientierung des Fräasers über die Bahn zu verändern, womit sich neue Frässtrategien ermöglichen.

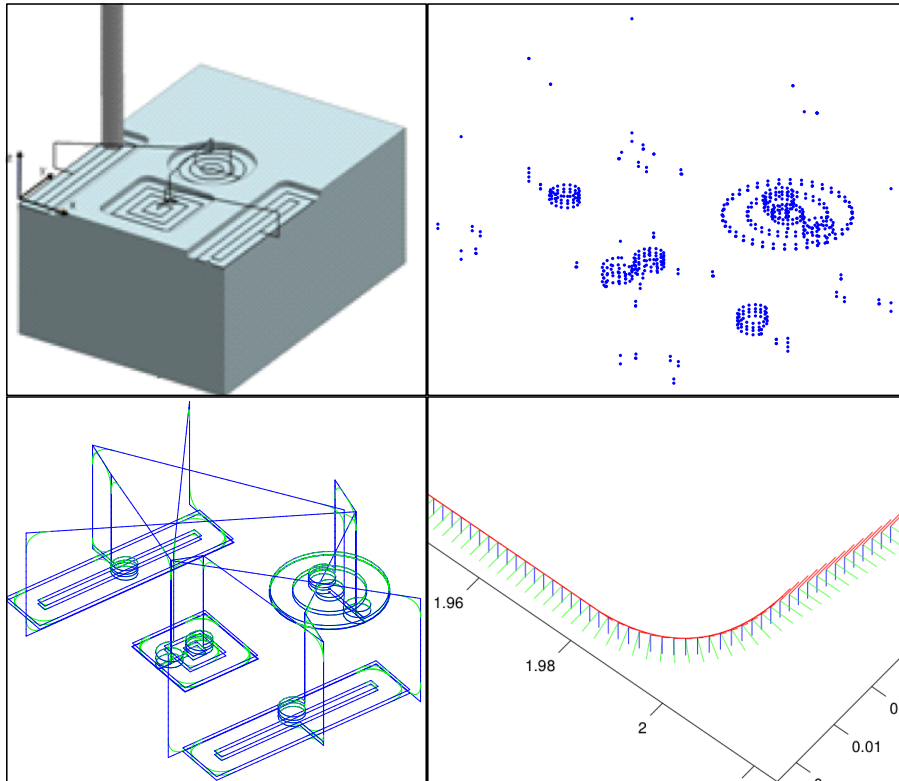


Abbildung 4.7: CAD Modell des Bauteils (links oben), Punktwolke der zu durchfahrenden Eckpunkte (rechts oben), Geometrie der Bahn mit blauen Linien und grünen Bézierkurven (links unten) und die Orientierungsinterpolation (rechts unten)

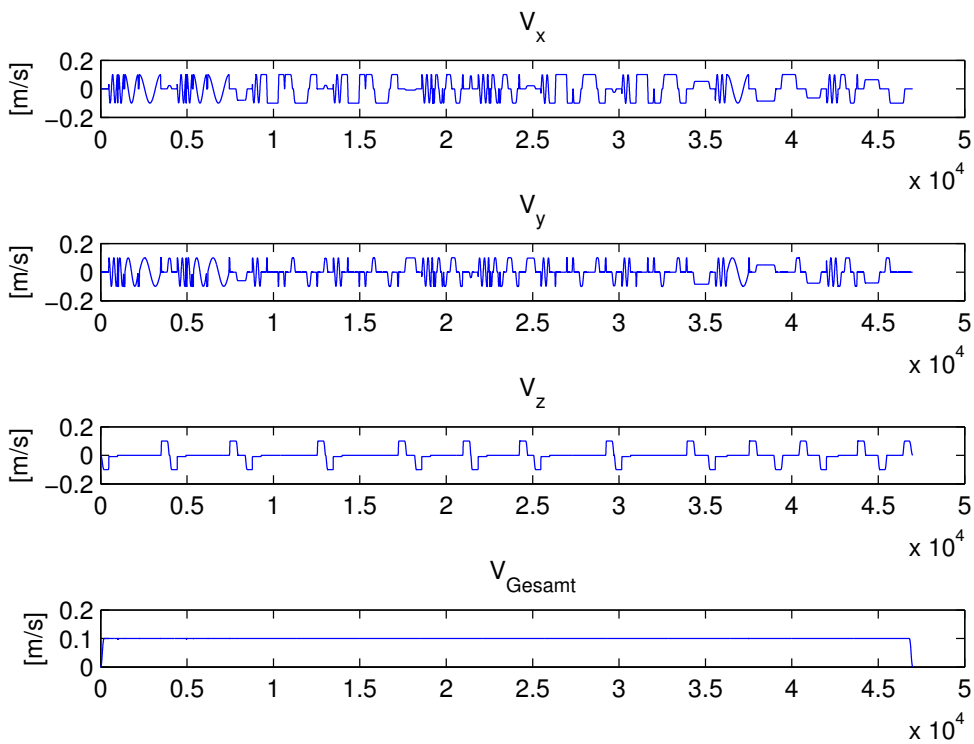


Abbildung 4.8: Geschwindigkeitsverlauf in x-, y- und z-Richtung und die Gesamtgeschwindigkeit

5 Zusammenfassung

Im Rahmen der vorliegenden Arbeit wurde eine Strategie der modellbasierten Offline-Kompensation beim Fräsen mit Industrierobotern betrachtet und im Hinblick auf ihre Anwendbarkeit mit einem realen Roboter erweitert. Zum einen wurden Steifigkeits- und Dämpfungsparameter für einen KUKA Roboter mit einem experimentellen Ansatz ermittelt. Mit diesem Verfahren konnte erfolgreich das Steifigkeitsverhalten des Roboters in die Simulation übertragen werden. Die Verwendung von virtuellen Achsen spielte dabei eine entscheidende Rolle. Die Elastizitäten, die durch die Lagerung der Roboterachsen orthogonal zu den Antriebsachsen auftreten, hatten einen signifikanten Einfluss auf die präzise Modellierung des Roboters. Des Weiteren wurde das in den Experimenten gemessene Hystereseverhalten und Spiel der Getriebe zum Anlass genommen eine weitere Modellverfeinerung für die Simulation durchzuführen. Auch die Dämpfungsparameter konnten mit dem heuristischen Ansatz mit einer akzeptablen Genauigkeit bestimmt werden. Jedoch konnte hierbei durch Betrachtung des Gesamtsystems nicht die hohe Präzision wie bei der separaten Betrachtung der einzelnen Komponenten erreicht werden. Die ermittelten Werte dienen als Startschätzungen für iterative Optimierungsverfahren, welche das Potential haben die Modellgenauigkeit weiter zu erhöhen.

Des Weiteren wurde im Rahmen der vorliegenden Arbeit die vorhandene Simulationsstruktur um eine Trajektorienplanung für komplexe Bauteile erweitert. Dabei stand eine robuste und automatische Planung im Vordergrund. Diese ermöglicht es für CAD-Bauteile eine Trajektorie zu erstellen, auf der die Simulation und Kompensation durchgeführt werden kann. Durch die Planung in kartesischen Koordinaten kann die Trajektorie zunächst unabhängig von einem speziellen Roboter erstellt werden. Zur Gewährleistung einer guten Bearbeitungsqualität wurde ein Überschleifen an den Ecken der Bahn verwendet, damit eine flüssige Bewegung des Roboters möglich ist. Dieses wurde mit Bézierkurven 5. Grades implementiert, damit eine C^2 -Stetigkeit an den Übergängen erreicht wird. Des Weiteren wurde eine Interpolation der Orientierung des Endeffektors implementiert. Damit ist es nun möglich Trajektorien zu erstellen, an denen die Punkte der Bahn mit unterschiedlicher Orientierung angefahren und gefräst werden können. Dies ist vor allem für größere und komplexere Bauteile von Bedeutung. Um durch Brems- und Beschleunigungsvorgänge nicht noch zusätzliche Abdrängungen zu erzeugen, wurde ein konstanter Geschwindigkeitsverlauf angestrebt. Hierbei ist auf die Gelenkbegrenzungen zu achten, damit diese bei höheren Geschwindigkeiten nicht verletzt werden.

Mit der Parameteridentifikation und der Trajektorienplanung sind für die modellbasierte Offline-Kompensation zwei wesentliche Module für weitere Untersuchungen und für ihre Anwendbarkeit durchgeführt worden.

Anhang

A Roll-Pitch-Yaw-Winkel

Roll-Pitch-Yaw-Winkel sind eine Möglichkeit die Orientierung eines Objektes im dreidimensionalen Raum zu beschreiben. Die Rotationsmatrix R kann dabei mit den drei Winkeln roll, pitch und yaw errechnet werden. Wie in Abbildung A.1

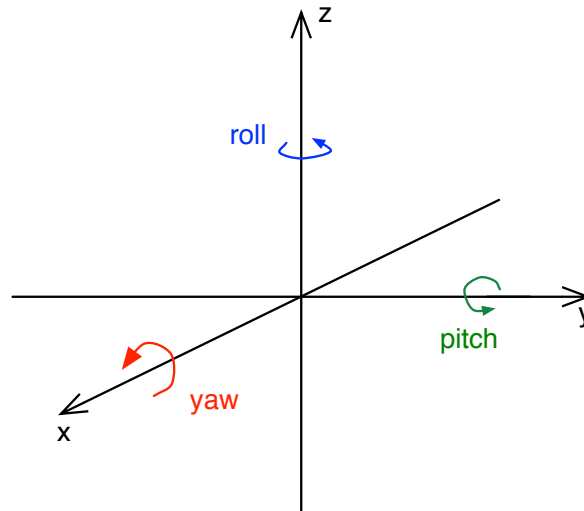


Abbildung A.1: Roll-Pitch-Yaw-Winkel

dargestellt beschreibt Yaw eine Drehung um x mit dem Winkel ψ , pitch eine Drehung um y mit dem Winkel θ und roll um z mit dem Winkel ϕ . Die Rotationen werden dabei in dieser Reihenfolge um die raumfeste Orthogonalbasis durchgeführt. Daraus resultierend erhält man folgende Rotationsmatrix [Spong, 2006]:

$$\begin{aligned}
 R(\phi, \theta, \psi) &= R_{z,\phi} R_{y,\theta} R_{x,\psi} \\
 &= \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi) & -\sin(\psi) \\ 0 & \sin(\psi) & \cos(\psi) \end{bmatrix} \\
 &= \begin{bmatrix} \cos(\phi)\cos(\theta) & -\sin(\phi) * \cos(\psi) + \cos(\phi)\sin(\theta)\sin(\psi) & \sin(\phi)\sin(\psi) + \cos(\phi)\sin(\theta)\cos(\psi) \\ \sin(\phi)\cos(\theta) & \cos(\phi) * \cos(\psi) + \sin(\phi)\sin(\theta)\sin(\psi) & -\cos(\phi)\sin(\psi) + \sin(\phi)\sin(\theta)\cos(\psi) \\ -\sin(\theta) & \cos(\theta)\sin(\psi) & \cos(\theta)\cos(\psi) \end{bmatrix}
 \end{aligned}$$

B Analytische Berechnung der inversen Kinematik

Mit der inversen Kinematik können von Roboterpositionen in kartesischen Koordinaten die zugehörigen Gelenkwinkel des Roboters berechnet werden. Eingabedaten sind dabei die kartesischen Koordinaten x, y, z und die Orientierung des TCP in Eulerwinkel ϕ, θ, ψ . Des Weiteren sind noch die Denavit-Hartenberg Parameter des Roboters bekannt. Anhand dieser Informationen werden die notwendigen Gelenkwinkel berechnet. Das angewendete Verfahren basiert dabei auf der Vorgehensweise wie sie in [Spong, 2006] erläutert wird. Die Berechnung der Gelenkwinkel wurde dabei in zwei Schritte entkoppelt. Dieses Verfahren ist nur möglich, wenn sich die letzten drei Rotationsachsen schneiden. Im ersten Schritt werden die Gelenkwinkel θ_1, θ_2 und θ_3 durch einen geometrischen Ansatz berechnet. Mit Hilfe der daraus resultierenden Matrix 0R_3 und der bekannten Rotationsmatrix 0R_6 werden dann die Gelenkwinkel θ_4, θ_5 und θ_6 mit dem Zusammenhang

$${}^0R_6 = {}^0R_3 {}^3R_6 \quad (\text{B.1})$$

berechnet. Der Schnittpunkt der letzten drei Achsen wird im folgenden mit $p = (p_x, p_y, p_z)$ bezeichnet. Durch Projektion des Punktes in die x-y Ebene, wie in Abbildung B.1 links dargestellt, kann man daraus den ersten Gelenkwinkel bestimmen.

$$\theta_1 = \text{Atan2}(p_y, p_x) \quad (\text{B.2})$$

Für den zweiten und dritten Gelenkwinkel spannen wir ein Dreieck vom zweiten Gelenk bis zum Punkt p auf (siehe

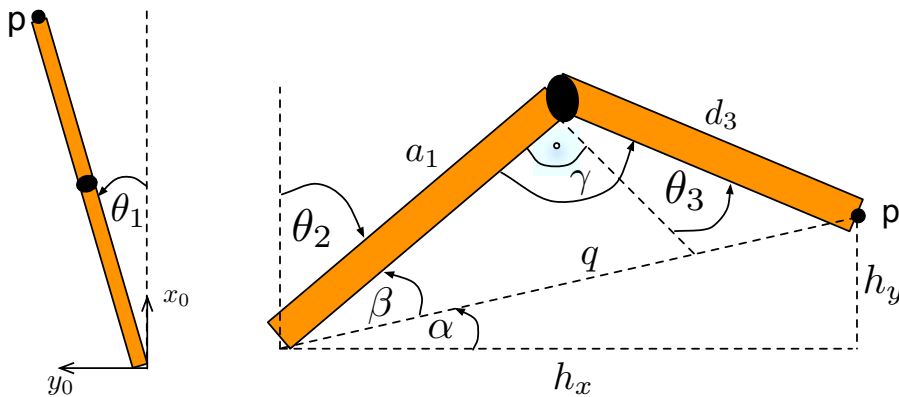


Abbildung B.1: Hilfsskizze zur Berechnung der Winkel θ_1, θ_2 und θ_3

Abb. B.1 rechts). Durch die Hilfswinkel α, β und γ und den Hilfsgeraden h_x, h_y und q können wir die Gelenkwinkel θ_2 und θ_3 herleiten. Dabei kommen wir durch den Einsatz von geometrischen Beziehungen, Additionstheoremen und dem Kosinussatz auf folgende Gleichungen für θ_2 :

$$\theta_2 = \frac{\pi}{2} - \beta - \alpha \quad (\text{B.3})$$

$$\cos(\alpha) = \frac{h_x}{q} \quad \sin(\alpha) = \frac{h_y}{q} \quad (\text{B.4})$$

$$\cos(\beta) = \frac{a_1^2 + q^2 - d_3^2}{2a_1q} \quad \sin(\beta) = \sqrt{1 - \cos^2(\beta)} \quad (\text{B.5})$$

$$\sin(\theta_2) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta) \quad (\text{B.6})$$

$$\cos(\theta_2) = \sin(\alpha)\cos(\beta) + \cos(\alpha)\sin(\beta) \quad (\text{B.7})$$

$$h_y = p_z - d_0 \quad h_x = \frac{p_x}{\cos(\theta_1)} - a_0 \quad (\text{B.8})$$

$$\theta_2 = \text{atan2}(\sin(\theta_2), \cos(\theta_2)) \quad (\text{B.9})$$

Analog ergibt sich für θ_3

$$\theta_3 = \frac{\pi}{2} - \gamma \quad (\text{B.10})$$

$$\cos(\gamma) = \frac{a_1^2 + d_3^2 - q^2}{2a_1d_3} \quad \sin(\gamma) = \sqrt{1 - \cos^2(\gamma)} \quad (\text{B.11})$$

$$\sin(\theta_3) = \sin\left(\frac{\pi}{2} - \gamma\right) = \cos(\gamma) \quad (\text{B.12})$$

$$\cos(\theta_3) = \cos\left(\frac{\pi}{2} - \gamma\right) = \sin(\gamma) \quad (\text{B.13})$$

$$\theta_3 = \text{atan2}(\sin(\theta_3), \cos(\theta_3)) \quad (\text{B.14})$$

Im zweiten Schritt werden dann die verbleibenden Gelenkwinkel θ_4 , θ_5 und θ_6 mit Hilfe der Formel B.1 bestimmt. Die Matrix 0R_6 kann aus den übergebenen Eulerwinkel bestimmt werden und die Matrix 3R_6 lässt sich mit den vorhandenen Denavit-Hartenberg Parametern aus dem Resultat der Matrizenmultiplikation ${}^3T_4 {}^4T_5 {}^5T_6$ ermitteln. Durch Vergleich der einzelnen Komponenten der Gleichung B.1 erhält man damit die restlichen Gelenkwinkel.

Abbildungsverzeichnis

2.1	Vorgehensweise der modellbasierten Offline-Kompensation	5
2.2	Überblick über die Kopplung des Roboterstrukturmodells mit dem Fräskraftmodell, um die Wechselwirkung der Roboterstruktur mit dem Fräsprozess zu simulieren	6
2.3	Kuka Roboter und die Achsen zur Modellierung von Elastizitäten bei der Gelenklagerung	7
3.1	Kraft-Verformungs-Kennlinie eines Belastungs- und Entlastungszyklus mit Spiel und Hysterese	9
3.2	Versuchsaufbau zur Messung der Steifigkeit der 6ten Achse (links) und die zugehörige Auswertung (rechts)	9
3.3	Umrechnung der Messdaten (links) und das entstehende Momenten/Winkeldiagramm (rechts)	10
3.4	Experimenteller und simulierter Übertragungsfrequenzgang	11
3.5	Messpunkte im relevanten Arbeitsraum (links) und Auswertung der Steifigkeiten in z-Richtung (rechts)	12
3.6	Auswertung der Steifigkeiten in x- (links) und y-Richtung (rechts)	12
4.1	Ein- und Ausgabedaten der Trajektorienplanung	13
4.2	Überschleifen an einem Übergang zwischen zwei Geradenstücken (links) und der Vergleich zwischen eines Geschwindigkeitsprofils ohne Überschleifen (rechts oben) und mit Überschleifen (rechts unten)	15
4.3	Parametrisierung der Trajektorie	16
4.4	Vergleich eines Linearen Geschwindigkeitsprofils (links) mit einem double-S Geschwindigkeitsprofil (rechts)	17
4.5	Schritte bei der Trajektorienplanung	18
4.6	Aufbau einer Trajektorie	19
4.7	CAD Modell des Bauteils (links oben), Punktwolke der zu durchfahrenden Eckpunkte (rechts oben), Geometrie der Bahn mit blauen Linien und grünen Bézierkurven (links unten) und die Orientierungsinterpolation (rechts unten)	20
4.8	Geschwindigkeitsverlauf in x-,y- und z-Richtung und die Gesamtgeschwindigkeit	20
A.1	Roll-Pitch-Yaw-Winkel	23
B.1	Hilfsskizze zur Berechnung der Winkel θ_1 , θ_2 und θ_3	24

Literaturverzeichnis

- [Abele, 2008] Abele, E., Bauer, J., Stelzer, M., and von Stryk, O. (2008). Wechselwirkung von Fräsprozess und Maschinenstruktur am Beispiel des Industrieroboters. *Werkstattstechnik online : wt, Springer VDI Verlag, Düsseldorf*, 98(9):733–737.
- [Altintas and Ber, 2001] Altintas, Y. and Ber, A. (2001). Manufacturing automation: metal cutting mechanics, machine tool vibrations, and CNC design. *Applied Mechanics Reviews*, 54.
- [Biagiotti, 2009] Biagiotti, L. (2009). Trajectory Planning. *Trajectory Planning for Automatic Machines and Robots*.
- [Craig, 1989] Craig, J. (1989). *Introduction to robotics*, volume 7. Addison-Wesley Reading, MA.
- [Hözl, 1994] Hözl, J. (1994). *Modellierung, Identifikation und Simulation der Dynamik von Industrierobotern*. VDI-Verlag.
- [Spong, 2006] Spong, M., Hutchinson, S., and Vidyasagar, M. (2006). *Robot modeling and control*. Wiley New Jersey.
- [Stelzer, 2008] Stelzer, M., von Stryk, O., Abele, E., Bauer, J., and Weigold, M. (2008). High speed cutting with industrial robots: Towards model based compensation of deviations. In *Proceedings of Robotik 2008, 11-12 June, Munich, Germany*, pages 143–146.
- [Weber, 2007] Weber, W. (2007). *Industrieroboter: Methoden der Steuerung und Regelung*. Hanser Verlag.
- [Weigold, 2008] Weigold, M. (2008). *Kompensation der Werkzeugabdrängung bei der spanenden Bearbeitung mit Industrierobotern*. Shaker.