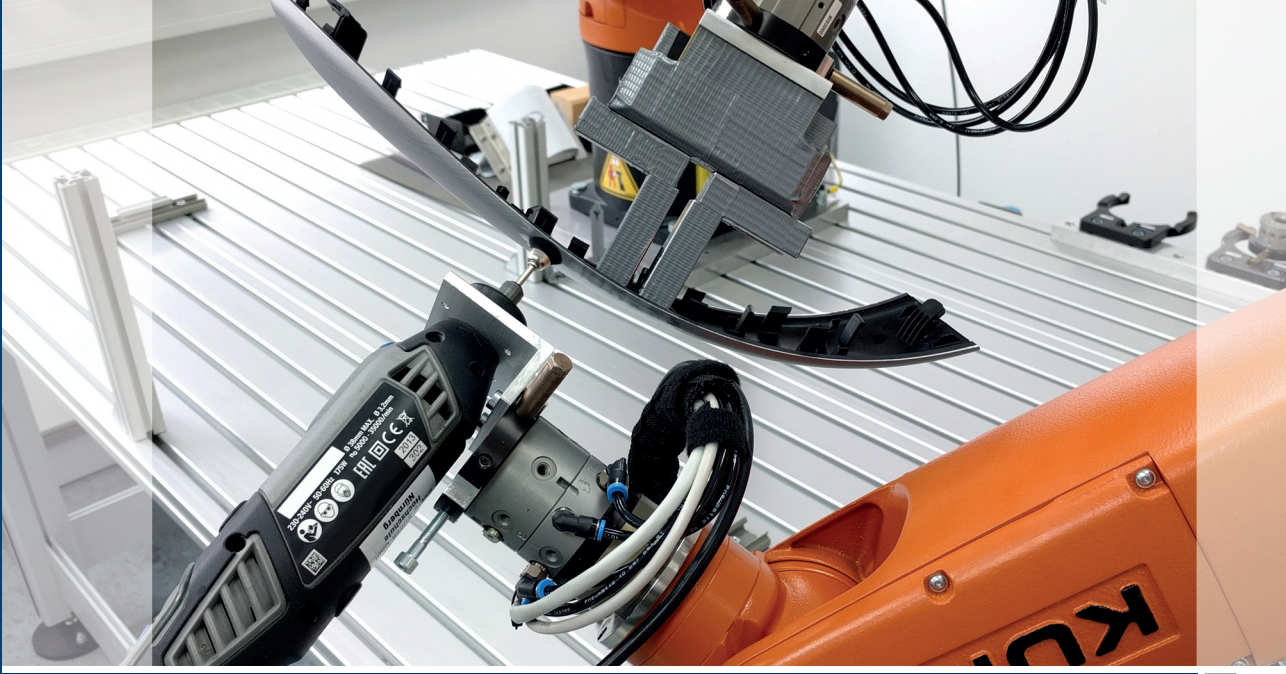


FAU Studien aus dem Maschinenbau 344

Maximilian Wagner

Automatische Bahnplanung für die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen mittels Multi-Roboter-Systemen

FAU UNIVERSITY PRESS

Maximilian Wagner

Automatische Bahnplanung für die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen mittels Multi-Roboter-Systemen

FAU Studien aus dem Maschinenbau

Band 344

Herausgeber der Reihe:

Prof. Dr.-Ing. Jörg Franke
Prof. Dr.-Ing. Nico Hanenkamp
Prof. Dr.-Ing. habil. Marion Merklein
Prof. Dr.-Ing. Michael Schmidt
Prof. Dr.-Ing. Sandro Wartzack

Maximilian Wagner

Automatische Bahnplanung für die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen mittels Multi-Roboter-Systemen

Dissertation aus dem Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik (FAPS)

Prof. Dr.-Ing. Jörg Franke

Erlangen
FAU University Press
2020

Bibliografische Information der Deutschen Nationalbibliothek:
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Bitte zitieren als
Wagner, Maximilian. 2020. *Automatische Bahnplanung für die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen mittels Multi-Roboter-Systemen.* FAU Studien aus dem Maschinenbau Band 344. Erlangen: FAU University Press.
DOI: 10.25593/978-3-96147-310-6.

Das Werk, einschließlich seiner Teile, ist urheberrechtlich geschützt.
Die Rechte an allen Inhalten liegen bei ihren jeweiligen Autoren.
Sie sind nutzbar unter der Creative Commons Lizenz BY-NC.

Der vollständige Inhalt des Buchs ist als PDF über den OPUS Server der Friedrich-Alexander-Universität Erlangen-Nürnberg abrufbar:
https://opus4.kobv.de/opus4-fau/home

Verlag und Auslieferung:
FAU University Press, Universitätsstraße 4, 91054 Erlangen

Druck: docupoint GmbH

ISBN: 978-3-96147-309-0 (Druckausgabe)
eISBN: 978-3-96147-310-6 (Online-Ausgabe)
ISSN: 2625-9974
DOI: 10.25593/978-3-96147-310-6

Automatische Bahnplanung für die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen mittels Multi-Roboter-Systemen

Der Technischen Fakultät
der Friedrich-Alexander-Universität
Erlangen-Nürnberg

zur
Erlangung des Doktorgrades Dr.-Ing.

vorgelegt von

M.Eng. Maximilian Wagner

aus Fürth

Als Dissertation genehmigt
von der Technischen Fakultät
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der mündlichen Prüfung: 19.07.2019
Vorsitzender des Promotionsorgans: Prof. Dr.-Ing. Reinhard Lerch

Gutachter: Prof. Dr.-Ing. Jörg Franke
Prof. Dr. Alois Zoitl, JKU Linz
Prof. Dr.-Ing. Peter Heß, TH Nürnberg

Vorwort

In der Industrierobotik ist es für die erfolgreiche Realisierung von Produktionsprozessen essentiell, die Kombination von Hardware und Software gut aufeinander abzustimmen. Dieses Zusammenspiel hat bereits während meines Studiums an der *Technische Hochschule Nürnberg Georg Simon Ohm* (THN) mein Interesse geweckt und ich habe mich zunehmend auf softwaretechnische Lösungen für Industrieroboteranwendungen spezialisiert. Bereits seit mehreren Jahrzehnten haben sich Industrieroboter in Produktionsprozessen bewährt. Dennoch besteht Potential für weitere Optimierungen dieser Prozesse, was der Anlass für mich war, diese Forschungsarbeit durchzuführen.

Im Rahmen dieser Forschungsarbeit und der damit verbundenen Anstellung am *Nuremberg Campus of Technology* (NCT) durfte ich mit einer Vielzahl an bemerkenswerten Menschen zusammenarbeiten. Zunächst gilt mein Dank dem Erstprüfer dieser Arbeit, Herrn Prof. Dr.-Ing. Jörg Franke vom *Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik* (FAPS) der *Friedrich-Alexander-Universität Erlangen-Nürnberg* (FAU), für die Möglichkeit zur Promotion in diesem Forschungsthema und die stets sehr konstruktive und zielführende Zusammenarbeit. Für den Vorsitz des Promotionsverfahrens bedanke ich mich bei Herrn Prof. Dr.-Ing. Klaus Feldmann, dem ehemaligen Inhaber des Lehrstuhls FAPS. Des Weiteren bedanke ich mich bei Herrn Prof. Dr. Alois Zoitl von der *Johannes Kepler Universität Linz* (JKU) für das entgegengebrachte Interesse am Forschungsthema und die Betreuung der Arbeit als Zweitprüfer sowie den interessanten fachlichen Austausch bei unseren Gesprächen. Großer Dank gilt Herrn Prof. Dr.-Ing. Peter Heß von der THN für die Schaffung der Grundlage des Forschungsthemas und der damit verbundenen Stelle am NCT sowie die Übernahme des Drittgutachten dieser Arbeit. Außerdem danke ich für die jederzeit sehr positive und hilfreiche Zusammenarbeit während meiner gesamten Zeit an der THN und dem NCT sowie darüber hinaus.

Allgemein möchte ich mich bei allen Kolleginnen und Kollegen sowie Studierenden, mit denen ich während meiner Forschungstätigkeit zusammengearbeitet habe, für die gute Zusammenarbeit bedanken. Einerseits gilt mein Dank den Kollegen des Forschungsbereichs Biomechatronik des FAPS, allen voran Arnd Buschhaus und Sebastian Reitelshöfer, für die stets aufgeschlossene Unterstützung und die konstruktiven Beiträge. Andererseits bedanke ich mich bei allen Kolleginnen und Kollegen sowie Studierenden der THN,

mit denen ich im Rahmen meiner Forschungstätigkeit an der THN zusammenarbeiten durfte. Besonderer Dank gilt den unmittelbaren Kolleginnen und Kollegen am NCT für die sehr angenehme Arbeitsumgebung und die tolle gemeinsame Zeit. Vor allem bedanke ich mich bei Christian Bergner, Christian Deuerlein und Christian Pfitzner für den anregenden fachlichen Austausch und die spannenden Gespräche.

Mein größter Dank gilt meiner Familie, sprich meinen Eltern Anne und Siegfried Wagner sowie meinem Bruder Fabian Wagner und meiner Schwägerin Claudia Wagner. Ich bedanke mich bei euch für die jederzeit bedingungslose Förderung sowie für die Rücksichtnahme und die motivierende Unterstützung während der Erstellung dieser Arbeit.

Fürth, im November 2019 Maximilian Wagner

Inhaltsverzeichnis

Abbildungsverzeichnis . . . . . **ix**

Tabellenverzeichnis . . . . . **xv**

Formelzeichen- und Abkürzungsverzeichnis . . . . . **xvii**

1 Einleitung **1**
1.1 Motivation 2
1.2 Zielsetzung 3
1.3 Vorgehensweise und Aufbau der Arbeit 4

2 Mathematische und kinematische Grundlagen der Robotik . **7**
2.1 Lage eines Körpers 7
2.2 Aufbau eines Industrieroboters 10
2.2.1 Kinematiken und Arbeitsräume 11
2.2.2 Vorwärtstransformation 12
2.2.3 Rückwärtstransformation 13
2.3 Bewegungsarten 17
2.4 Genauigkeit 18
2.4.1 Messung 19
2.4.2 Korrektur 20

3 Stand der Wissenschaft und Technik **23**
3.1 Aufteilung von Bearbeitungen auf mehrere Kinematiken . . 23
3.1.1 NC-gesteuerte Bearbeitungsmaschinen 24
3.1.2 Industrieroboter mit Rotationstisch 25
3.2 Kooperierende Industrieroboter 27
3.2.1 Koordinationsarten 27
3.2.2 Kommunikationsstrukturen 29
3.3 Programmierung von Industrierobotern 29
3.3.1 Programmierverfahren 30
3.3.2 Programmierausführung 32
3.3.3 Programmierung von kooperierenden Industrierobotern 35
3.4 Handlungsbedarf 36

4 Aufteilung von Prozessbewegungen auf mehrere Industrieroboter 39
4.1 Grundkonzept der Aufteilung in eine synchrone Bewegung von Werkzeug und Werkstück 40
4.1.1 Startposeniteration 41
4.1.2 Bahnaufteilung 43
4.1.3 Selektion 48
4.1.4 Skalierbarkeit der aufzuteilenden Bahnabschnitte . . 50
4.2 Begrenzung der Aufteilung 51
4.2.1 Verteilung der werkstückbezogenen Freiheitsgrade . 51
4.2.2 Kartesische und achsspezifische Begrenzung 52
4.3 Allgemeine Einflüsse der Aufteilung 53
4.3.1 Einfluss der kooperativen Bearbeitung auf die Bearbeitungszeit 53
4.3.2 Einfluss der kooperativen Bearbeitung auf die Erreichbarkeit 60
4.3.3 Relative Positioniergenauigkeit der beteiligten Industrieroboter 66
4.4 Zusammenfassung 72

5 Erfassung von Werkstücken mittels kooperierenden Industrierobotern 73
5.1 Scansystem und notwendige Kalibrierungen 73
5.1.1 Sensorkalibrierung 75
5.1.2 Roboterkalibrierung 77
5.1.3 Kamerakalibrierung 79
5.2 Scanvorgang 82
5.2.1 Sensorbewegung 83
5.2.2 Werkstückbewegung 84
5.2.3 Kooperative Bewegung 85
5.2.4 Farbwerterweiterung 85
5.3 Verarbeitung der erzeugten Daten 87
5.3.1 Reduzierung der Punktdaten durch Fusion 87
5.3.2 Reduzierung der Punktdaten durch Zuschnitt 88
5.3.3 Erzeugen von Flächenmodellen 88
5.4 Zusammenfassung 88

6 Implementierung **91**
6.1 Robotersystem mit dezentraler Steuerung erweitert um Werkstückerfassung 92
6.1.1 Anlagenaufbau 93
6.1.2 Softwareimplementierung 97
6.2 Robotersystem mit zentraler Steuerung 108
6.2.1 Anlagenaufbau 108
6.2.2 Softwareimplementierung 110
6.3 Zusammenfassung 114

7 Evaluation **115**
7.1 Allgemeine Untersuchungen 115
7.1.1 Auswahl einer geeigneten Diskretisierung der Startposeniteration 116
7.1.2 Grenzen der Bearbeitungszeitreduzierung der verwendeten Robotersysteme 119
7.1.3 Allgemeine Erreichbarkeit der verwendeten Robotersysteme 124
7.1.4 Kinematische Kalibrierung der verwendeten Robotersysteme 130
7.2 Anwendungsspezifische Untersuchungen 137
7.2.1 Bearbeitungszeitreduzierung durch kooperative Bearbeitung 137
7.2.2 Erschließung von Anwendungen durch verbesserte Erreichbarkeit 148
7.3 Ableitung von Bewertungskriterien für den sinnvollen Einsatz der kooperativen Bearbeitung 153
7.4 Zusammenfassung 155

8 Zusammenfassung und Ausblick **157**

9 Summary and outlook **161**

Literaturverzeichnis **165**

Abbildungsverzeichnis

1 Vorgehensweise und Aufbau der Arbeit 5
2 Objektkoordinatensystem $\{W\}$ relativ zu einem Bezugskoordinatensystem $\{O\}$ 7
3 Lage eines Koordinatensystems nach Z-Y-X-Euler-Winkeldefinition 8
4 Roboterspezifische Koordinatensysteme und Verfahren der Vorwärts- und Rückwärtstransformation 10
5 Übliche Kinematiken von Industrierobotern 11
6 Übliche Arbeitsräume von Industrierobotern 12
7 Beispiel einer kinematischen Kette mit Definition der DH-Parameter 12
8 Mehrdeutigkeit der Armstellung aufgrund einer Überkopf- oder Drehbewegung bei der Rückwärtstransformation von sechsachsigen Knickarmrobotern 14
9 Singularität der Achsorientierung Achse 4 und 6 bei der Rückwärtstransformation von sechsachsigen Knickarmrobotern . 15
10 Mehrdeutigkeit der Achsorientierung Achse 3 und 5 bei der Rückwärtstransformation von sechsachsigen Knickarmrobotern . 15
11 Bewegungsarten von Industrierobotern 17
12 Kombinationen zwischen Positionier- und Wiederholgenauigkeit 18
13 Koordinatensysteme und Transformationen für die Genauigkeitsuntersuchung 19
14 Methode der geschlossenen Korrekturschleife für Industrieroboter 21
15 Methode der offenen Korrekturschleife für Industrieroboter . . . 22
16 Exemplarische NC-gesteuerte Bearbeitungsmaschinen mit beidseitiger Anordnung von Achsen 24
17 Tiefbohrzentrum mit zwei Achsen (Z und W), die den selben Freiheitsgrad abdecken 25
18 Sechsachsiger Knickarmroboter mit Dreh- und Kipptisch zum Nahtschweißen 26
19 Koordinationsarten zwischen mehreren Industrierobotern 28
20 Kommunikationsstrukturen von kooperierenden Industrierobotern 29
21 Einteilung der Programmierverfahren von Industrierobotern kumuliert aus 30

22 Aufwand für die Programmierung von Industrierobotern in Abhängigkeit der Komplexität der realisierten Anwendung 33
23 Marktübliche Grundfunktionen für kooperierende Industrieroboter 36
24 Allgemeines Vorgehen für die Aufteilung von Bearbeitungsprozessen auf mehrere Industrieroboter 39
25 Methode für die Aufteilung von Bearbeitungsbahnen in eine synchrone Bewegung von Werkzeug und Werkstück 41
26 Algorithmus zur Startposeniteration 42
27 Algorithmus zur Bahnaufteilung 44
28 Koordinatensysteme und Transformationen für die kooperative Bearbeitung 45
29 Werkstückbezogene Koordinatensysteme und Transformationen für die kooperative Bearbeitung 45
30 Werkstückbezogene Abweichungen bei der kooperativen Bewegung 47
31 Algorithmus zur Lösungsauswahl 49
32 Abhängigkeit der Berechnungszeiten von der Anzahl an Bahnabschnitten 51
33 Verfahrprofile ohne Ruckbegrenzung 55
34 Verfahrprofile mit Ruckbegrenzung 56
35 Methode für die zeitliche Optimierung der Aufteilung von Bearbeitungsprozessen auf mehrere Industrieroboter 58
36 Abhängigkeit der beidseitigen maximalen Achswinkeländerungen von der Skalierung 59
37 Iterationsverhalten bei der Optimierung der Skalierung 60
38 Hierarchische Diskretisierung des Roboterarbeitsraums 61
39 Ausschnitt einer ohne Kollisionserkennung erstellten Reachability-Map eines KUKA KR 6 R900 sixx Knickarmroboters 63
40 Koordinatensysteme und Transformationen zur Erstellung von Reachability-Maps für die kooperative Bearbeitung 64
41 Algorithmus zur Erstellung von Reachability-Maps für die kooperative Bearbeitung 65
42 Toleranzkette bei einer kooperativen Bearbeitung 66
43 Methode der offenen Korrekturschleife für das betrachtete kooperative Robotersystem 69
44 Koordinatensysteme und Transformationen für die Genauigkeitsuntersuchung am kooperativen Robotersystem . . . 69
45 Algorithmus für das Erzeugen der Roboterbahnen für die Genauigkeitsuntersuchung am kooperativen Robotersystem . . . 70

46 Koordinatensysteme und Transformationen innerhalb des Scansystems mit kooperierenden Industrierobotern 74
47 Ablaufübersicht für den Scanvorgang mittels kooperierenden Industrierobotern inklusive optionaler Farbwerterweiterung . . . 82
48 Koordinatensysteme und Transformationen für den Scanvorgang am sensorführenden Roboter 83
49 Koordinatensysteme und Transformationen für den Scanvorgang am werkstückführenden Roboter 84
50 Ablaufübersicht für die Erweiterung der Lasersensormesswerte mit Farbinformationen 85
51 Ablauf der Roboterprogrammierung für die kooperative Bearbeitung 92
52 Anlagenaufbau des ersten Robotersystems für die Implementierung 93
53 Knickarmroboter KR 6 R900 sixx der KUKA Roboter GmbH . . . 94
54 Werkzeugwechselsystem SWS-005 der SCHUNK GmbH & Co. KG 95
55 Laser-Profilsensor scanCONTROL 2600-100 der MICRO-EPSILON MESSTECHNIK GmbH & Co. KG 96
56 HD Pro Webcam C920 von Logitech 96
57 Übersicht der Verbindungen innerhalb des ersten Robotersystems 97
58 Softwareoberfläche der entwickelten Offline-Programmiersoftware 98
59 Entwickelte Offline-Programmiersoftware während der Bahndefinition 99
60 Ablaufplan für die Bahndefinition mit der entwickelten Offline-Programmiersoftware 100
61 Entwickelte Offline-Programmiersoftware während der Bahnaufteilung 101
62 Ablaufplan für die Bahnaufteilung mit der entwickelten Offline-Programmiersoftware 102
63 Entwickelte Offline-Programmiersoftware während der Programmgenerierung und -überprüfung 102
64 Ablaufplan für die Programmgenerierung und -überprüfung mit der entwickelten Offline-Programmiersoftware 103
65 Algorithmus der Programmgenerierung für die kooperative Bearbeitung mit Multi-Roboter-Systemen 104
66 Entwickelte Offline-Programmiersoftware während der Werkstückerfassung 105

67 Ablaufplan für die Werkstückerfassung mit der entwickelten Offline-Programmiersoftware - Teil 1/2 106
68 Ablaufplan für die Werkstückerfassung mit der entwickelten Offline-Programmiersoftware - Teil 2/2 107
69 Entwickelte Offline-Programmiersoftware während der Punktwolkenbearbeitung 108
70 Anlagenaufbau des zweiten Robotersystems für die Implementierung 109
71 Zweiarmroboter IRB 14000-0.5/0.5 (YuMi) der ABB Automation GmbH 109
72 Softwareoberfläche der herstellerspezifischen Offline-Programmiersoftware mit integriertem Add-In für die kooperative Bearbeitung 111
73 Proprietäre Offline-Programmiersoftware während der Bahndefinition 112
74 Proprietäre Offline-Programmiersoftware während der Bahnaufteilung 113
75 Proprietäre Offline-Programmiersoftware während der Programmgenerierung und -überprüfung 114
76 Messwerte aus der Diskretisierung der Startposeniteration für das erste Robotersystem 117
77 Messwerte aus der Diskretisierung der Startposeniteration für das zweite Robotersystem 118
78 Synchronisationszeiten in Abhängigkeit von der Anzahl an Bewegungsbefehlen 120
79 Am Roboterflansch montierter Sensor für die Beschleunigungsmessung 121
80 Beschleunigungsprofil des ersten Robotersystems 122
81 Beschleunigungsprofil des zweiten Robotersystems 123
82 Auf den gemeinsamen Arbeitsbereich reduzierte Reachability-Maps für das erste Robotersystem 125
83 Auf den gemeinsamen Arbeitsbereich reduzierte Reachability-Maps für das zweite Robotersystem 126
84 Der Größe nach geordnete Reachability-Indizes für die untersuchten Robotersysteme 127
85 Einfluss des Abstands der Roboter auf die Erreichbarkeit 128
86 Einfluss des TCP sowie der Werkstückbasis auf die Erreichbarkeit 129
87 Einfluss der werkstückseitigen rotatorischen Freiheitsgrade auf die Erreichbarkeit 130

88 Aufbau für die Genauigkeitsuntersuchung am kooperativen Robotersystem mittels Laser-Tracker 131
89 Für die mittels Laser-Tracker durchgeführten Positionsmessungen und Kalibrierungen verwendete Reflektoren 132
90 Für die Genauigkeitsuntersuchung erzeugte Roboterbahn des werkzeugführenden Roboters 133
91 Fehlerwerte an den einzelnen Messpositionen 134
92 Karten der relativen Positioniergenauigkeit des kooperativen Robotersystems mit skalierten Abweichungen, die in Abhängigkeit des Betrags gemäßeiner Heatmap eingefärbt sind . 135
93 Box-Whisker-Plot der Positionsfehler für das kooperative Robotersystem 137
94 Komponenten für die Applikation von Klebstoff 139
95 Resultierende Bahnverläufe für die Applikation von Klebstoff . . 140
96 Bearbeitungszeiten für die Applikation von Klebstoff 141
97 Komponenten für das Zeichnen von Mäander 141
98 Resultierende Bahnverläufe für das Zeichnen von Mäander 142
99 Bearbeitungszeiten für das Zeichnen von Mäander 143
100 Komponenten für das Polieren von Zuziehgriffblenden 144
101 Resultierende Bahnverläufe für das Polieren von Zuziehgriffblenden 145
102 Bearbeitungszeiten für das Polieren von Zuziehgriffblenden . . . 145
103 Komponenten für die Montage von Sprinklersystemen 146
104 Resultierende Bahnverläufe für die Montage von Sprinklersystemen 147
105 Bearbeitungszeiten für die Montage von Sprinklersystemen . . . 148
106 Maximale Diagonalen im Arbeitsraum 148
107 Komponenten für die Inspektion von Gewindestangen 150
108 Resultierende Bahnverläufe für die Inspektion von Gewindestangen 151
109 Komponenten für das Markieren von Kunststoffleisten 152
110 Resultierende Bahnverläufe für das Markieren von Kunststoffleisten 153
111 Bewertungskriterien für den sinnvollen Einsatz der kooperativen Bearbeitung 154

Tabellenverzeichnis

1 Vergleich der Methoden zur kinematischen Berechnung einer Reachability-Map 61
2 Toleranzkette und Kompensationen bei einer kooperativen Bearbeitung 67
3 Vergleich der Kalibrierverfahren für robotergeführte Lasersensoren 75
4 Vergleich der Kalibrierverfahren für eine zusätzliche Kinematik . 78
5 Vergleich der Kalibrierverfahren zwischen Lasersensor und Kamera 81
6 Übersicht der für die Implementierung verwendeten Robotersysteme 91
7 DH-Parameter des Knickarmroboters KR 6 R900 sixx der KUKA Roboter GmbH 95
8 DH-Parameter des Zweiarmroboters IRB 14000-0.5/0.5 (YuMi) der ABB Automation GmbH 110
9 Übersicht über die allgemeinen Untersuchungen 115
10 Experimentell bestimmte Gleichungen für die Synchronisationszeiten der Robotersysteme mit und ohne Überschleifen der Bahnpunkte 120
11 Verteilung der absoluten Positionsfehler der beiden Roboterarme 133
13 Verteilung der relativen Positionsfehler für das kooperative Robotersystem 136
15 Übersicht über die anwendungsspezifischen Untersuchungen . . . 138

Formelzeichen- und Abkürzungsverzeichnis

Allgemeine Formelzeichen

Symbol	Einheit	Beschreibung
α, β, γ	rad	Raumwinkel
δ	m	Variabler Abstand
$\Delta\alpha$, $\Delta\beta$, $\Delta\gamma$	rad	Raumwinkeländerungen
$\Delta\theta$	rad	Achswinkeländerung
$\Delta\varphi$	rad	Betrag der rotatorischen Bahnabweichung
Δt	m	Betrag der translatorischen Bahnabweichung
Δx, Δy, Δz	m	Raumkoordinatenänderung
$\mathbf{e}$	m	Fehlervektor $(x_\mathrm{E}\ \ y_\mathrm{E}\ \ z_\mathrm{E})^\mathrm{T}$
$\mathbf{f}$	m, rad	Framenormvektor $(x\ \ y\ \ z\ \ \alpha\ \ \beta\ \ \gamma)^\mathrm{T}$
$\mathbf{l}$	m	Vektor zu einem Sensormesspunkt $(x_\mathrm{L}\ \ 0\ \ z_\mathrm{L})^\mathrm{T}$
$\mathbf{n}$	-	Normalenvektor $(x_\mathrm{N}\ \ y_\mathrm{N}\ \ z_\mathrm{N})^\mathrm{T}$
$\mathbf{p}$	m	Vektor zu einem Punkt $(x_\mathrm{P}\ \ y_\mathrm{P}\ \ z_\mathrm{P})^\mathrm{T}$
$\mathbf{R}$	-	Rotationsmatrix
$\mathbf{r}$	rad	Rotationsvektor $(\alpha\ \ \beta\ \ \gamma)^\mathrm{T}$
$\mathbf{s}$	-	Skalierungsvektor $\left(s_\mathrm{x}\ \ s_\mathrm{y}\ \ s_\mathrm{z}\ \ s_\alpha\ \ s_\beta\ \ s_\gamma\right)^\mathrm{T}$
$\mathbf{t}$	m	Translationsvektor $(x\ \ y\ \ z)^\mathrm{T}$
$\mathbf{x}$	-	x-Achsenvektor $(x_\mathrm{x}\ \ y_\mathrm{x}\ \ z_\mathrm{x})^\mathrm{T}$
$\mathbf{y}$	-	y-Achsenvektor $\left(x_\mathrm{y}\ \ y_\mathrm{y}\ \ z_\mathrm{y}\right)^\mathrm{T}$
$\mathbf{z}$	-	z-Achsenvektor $(x_\mathrm{z}\ \ y_\mathrm{z}\ \ z_\mathrm{z})^\mathrm{T}$
$\mathcal{P}$	-	Datenmenge
σ	-	Standardabweichung
Θ	rad	Summe der maximalen Achswinkeländerungen
θ	rad	Achswinkel
ε	m	Fehlerbetrag

${}^{A}\mathbf{T}_{B_{rot}}$	-	Transformationmatrix für die Rotation vom Koordinatensystem $\{A\}$ zum Koordinatensystem $\{B\}$
${}^{A}\mathbf{T}_{B_{trans}}$	-	Transformationmatrix für die Translation vom Koordinatensystem $\{A\}$ zum Koordinatensystem $\{B\}$
${}^{A}\mathbf{T}_{B}$	-	Transformationsmatrix für die Transformation vom Koordinatensystem $\{A\}$ zum Koordinatensystem $\{B\}$
D	%	Reachability-Index
d	m	Euklidischer Abstand
f_C	Hz	Grenzfrequenz
G	-	Güteziffer
i, j, k	-	Indizes
L	-	Lösung der Rückwärtstransformation
l	m	Länge
N	-	Anzahl
p	%	Prozentsatz
R	-	Element der Rotationsmatrix
r	m	Reichweite
s	-	Skalierungsfaktor
t	s	Zeit
X, Y, Z	-	Punktmenge
x, y, z	m	Raumkoordinaten

Bewegungsvorgang

Symbol	**Einheit**	**Beschreibung**
Δt	s	Zeitintervall
a	$\mathrm{m/s^2}$	Beschleunigung
j	$\mathrm{m/s^3}$	Ruck
v	m/s	Geschwindigkeit

Bildverarbeitung

Symbol	Einheit	Beschreibung
γ	-	Verzerrungsparameter
c	Pixel	Vektor zu einem Kamerapixel $(x_\mathrm{C} \;\; y_\mathrm{C})^\mathrm{T}$
d	-	Verzerrungsvektor $(k_1 \;\; k_2 \;\; p_1 \;\; p_2)^\mathrm{T}$
K	-	Intrinsische Matrix
c_x, c_y	Pixel	Bildhauptpunktkoordinaten
f_x, f_y	m	Brennweite

DH-Parameter

Symbol	Einheit	Beschreibung
α_i	rad	Winkel zwischen der z_{i-1}-Achse und der z_i-Achse um die x_i-Achse
θ_i	rad	Winkel zwischen der x_{i-1}-Achse und der x_i-Achse um die z_{i-1}-Achse
${}^{i-1}\mathbf{A}_i$	-	DH-Matrix für die Transformation von Achse $i-1$ zu Achse i
a_i	m	Verschiebung zwischen der z_{i-1}-Achse und der z_i-Achse in x-Richtung
d_i	m	Verschiebung zwischen der x_{i-1}-Achse und der x_i-Achse in z-Richtung

Toleranzkettenbetrachtung

Symbol	Einheit	Beschreibung
E_C	m	Toleranzmittenabmaß
M	m	Maß
N	m	Nennwert
T	m	Toleranz

Konstanten

Symbol	Einheit	Beschreibung
$\pi = 3,14159...$	-	Kreiszahl

Koordinatensysteme

Symbol	Beschreibung
$\{B\}$	Koordinatensystem in der Basis des Roboters
$\{C\}$	Koordinatensystem in der Basis der Kamera
$\{F\}$	Koordinatensystem im Flansch des Roboters
$\{H\}$	Koordinatensystem in einem Hilfspunkt
$\{K\}$	Koordinatensystem im Zentrum der Kugel
$\{O\}$	Koordinatensystem im Ursprung
$\{P\}$	Koordinatensystem in einer Bahnpose
$\{S\}$	Koordinatensystem in der Basis des Sensors
$\{T\}$	Koordinatensystem im TCP
$\{W\}$	Koordinatensystem in der Basis des Werkstückes

Abkürzungen

Symbol	Beschreibung
ABS	Acrylnitril-Butadien-Styrol-Copolymere
CAD	Computer-Aided Design
CFRP	Carbon Fiber Reinforced Polymer
CP	Continuous-Path
DA-RL	Dual-Arm Robot Programming Language
DH	Denavit-Hartenberg
FK	Forward Kinematics
HMI	Human Machine Interface
HYB	Hybrid
ICP	Iterative Closest Point
IFR	International Federation of Robotics
IK	Inverse Kinematics
KMM	Koordinatenmessmaschine
MP	Multi-Point
MRK	Mensch-Roboter-Kollaboration

MRS	Multi-Roboter-System
NC	Numerical Control
POE	Power Over Ethernet
PTFE	Polytetrafluorethylen
PTP	Point-To-Point
STL	STereoLithography
TCP	Tool Center Point
USB	Universal Serial Bus
VTK	Visualization Toolkit
WPF	Windows Presentation Foundation
XAML	Extensible Application Markup Language

1 Einleitung

Industrieroboter bieten eine Reihe von vorteilhaften Eigenschaften. Ein zunehmend wichtiger Aspekt ist die hohe Flexibilität in Bezug auf ihre Fähigkeit, sich an unterschiedliche Arbeitsprozesse anzupassen. Nicht zuletzt deshalb kommen Industrieroboter immer häufiger zum Einsatz. Wie eine aktuelle Studie der *International Federation of Robotics* (IFR) zeigt, sind die Verkaufszahlen seit 2012 so stark angestiegen wie noch nie zuvor [92]. Im Jahr 2017 erreichten sie ihr bisheriges Maximum und es wird von einem weiteren Wachstum ausgegangen. Dabei werden die Einsatzfelder von Industrierobotern zunehmend komplexer und kürzere Einsatzzeiten der Anwendungen geraten in den Fokus, was einen erhöhten Aufwand für die Programmierung der Robotersysteme bedingt. Dies geht bis hin zur Einzelstückfertigung, bei der jedes Werkstück ein neues Programm erfordert. Um derartige Anlagen wirtschaftlich zu betreiben, muss ein möglichst geringer Aufwand bei der Umstellung auf neue Werkstücke erzielt werden. Eine Studie zeigt, dass die komplexe Programmierung, neben mangelnden Genauigkeiten und ungenügenden Steifigkeiten, eine der drei Hauptherausforderungen für die roboterbasierte Bearbeitung von Werkstücken ist [53]. Eine Möglichkeit, die komplexe Programmierung zu bewältigen und zu beschleunigen, stellt die automatisierte Programmierung dar, bei der eine Software die Erstellung der Roboterbahnen und der Programme übernimmt [12]. Der Benutzer selbst muss dabei keinen expliziten Programmcode erstellen, sondern gibt lediglich einzelne Parameter vor.

Ein weiterer Trend ist der Einsatz von *Multi-Roboter-System* (MRS), bei denen mehrere Roboter kooperativ zusammenarbeiten. Vor allem im Bereich der mobilen Robotik kommen diese zunehmend zum Einsatz, um Aufgaben auf mehrere Roboter zu verteilen [93, 108]. In der Industrierobotik ist die Automobilindustrie der Vorreiter beim Einsatz von MRS [131]. Insbesondere im Karosseriebau haben sich derartige Systeme im großen Umfang etabliert. Dennoch sind gegenwärtig verhältnismäßig wenig Industrieanwendungen mittels kooperativen Industrierobotern umgesetzt. Dabei bietet der Einsatz mehrerer kooperierender Industrieroboter eine Reihe von Vorteilen. So können beispielsweise Bewegungen und Prozesse überlagert werden. Eine koordinierte Zusammenarbeit von mehreren Robotern innerhalb eines gemeinsamen Arbeitsbereiches ermöglicht es, Durchlaufzeiten zu reduzieren. Ein weiterer Vorteil, der sich durch die Überlagerung der Arbeitsbereiche ergibt, ist die Reduzierung des Platzbedarfes in der Produktionsstätte. Des Weiteren lässt sich die Flexibilität von Anlagen durch eine Kooperation

verbessern, wie beispielsweise beim Heben von Lasten, das auf mehrere kleinere Roboter aufgeteilt wird, um diese dann nur gezielt nach Lastgröße einzusetzen [77]. Zudem erschließt sich durch den Einsatz von MRS eine Reihe von Anwendungsfeldern, die erst durch kooperative Prozesse möglich werden, wie zum Beispiel zweiarmige Prozesse [102].

Die vorliegende Dissertation befasst sich mit der Aufteilung von Prozessen auf mehrere Industrieroboter. Dabei wird ein neuartiges Konzept für die Aufteilung von Bearbeitungsbahnen auf mehrere Kinematiken, durch gleichzeitige Handhabung von Werkstück und Werkzeug, vorgestellt und untersucht. Andererseits wird dieser Ansatz in einem Programmierungsautomatismus umgesetzt, um auch komplexere Programme mit geringem Zeitaufwand erzeugen zu können.

1.1 Motivation

In den meisten existierenden MRS fällt die Koordination zwischen den beteiligten Robotern verhältnismäßig einfach aus, da etwa nur einfache Signale ausgetauscht werden. MRS besitzen jedoch auch das Potenzial für Prozesse mit einer weitaus komplexeren Koordination. Ein geeignetes Vorbild hierfür bietet der Mensch, der mit seinen beiden Armen eine Vielzahl von Prozessen umsetzen kann und diese dabei auf unterschiedliche Weise koordiniert. Meist verwendet der Mensch seine dominante Hand für eine Hauptbewegung, während er die andere Hand lediglich für unterstützende Bewegungen einsetzt, was beispielsweise beim Schreiben mit einem Stift oder beim Schnitzen von Holz deutlich wird. Es gibt jedoch auch Tätigkeiten, bei denen beide Hände gleichermaßen verwendet werden. Hierbei können die Bewegungen entweder parallel, unabhängig oder gegenläufig ausgeführt werden [73]. Ein Beispiel für eine parallele Bewegung, bei der beide Hände die gleiche Bewegung absolvieren, ist das zweihändige Führen eines Werkzeugs. Bei einer unabhängigen Bewegung haben die beiden Hände keinen räumlichen Bezug zueinander, wie etwa beim Schreiben auf einer Tastatur. Gegenläufige Bewegungen treten unter anderem beim Binden von Schuhen auf. Hieraus leitet sich die Idee ab, Bearbeitungsbahnen auf zwei gegenläufige Bewegungen – Werkstück- und Werkzeugbewegung – aufzuteilen, um optimierte Prozesse zu erzielen.

Ein Grund weshalb derartige kooperative Anwendungen noch verhältnismäßig selten zum Einsatz kommen ist die aufwendige Programmierung der beteiligten Roboter. Die meisten Roboterhersteller bieten zwar grundlegende Kooperationsfunktionen (z. B. zeitliche oder räumliche

Synchronisation) für ihre Systeme an – bei komplexeren Anwendungen ist der Programmieraufwand jedoch oftmals unverhältnismäßig groß. Besonders unter dem Aspekt, dass die Flexibilität von Industrieroboteranlagen zunehmend wichtiger wird, ist es zielführend, die Programmierung von kooperierenden Industrierobotern zu vereinfachen.

1.2 Zielsetzung

Das übergeordnete Ziel dieser Arbeit ist die Erforschung einer Methode zur flexiblen Aufteilung von Bearbeitungsbahnen auf mehrere Kinematiken, die Werkstück und Werkzeug gleichzeitig handhaben. Mit dieser Art der Aufteilung trägt die vorliegende Arbeit zur Erweiterung des Standes der Wissenschaft und Technik bei. Potenzielle Vorteile, die sich daraus ergeben, sind reduzierte Bearbeitungszeiten sowie eine verbesserte werkstückbezogene Erreichbarkeit. Der Ansatz ist prinzipiell auf alle Bearbeitungssysteme, die beidseitig eine Kinematik besitzen, anwendbar. Bei *Numerical Control* (NC)-gesteuerten Bearbeitungsmaschinen kann die Aufteilung jedoch lediglich auf einzelne Achsen bezogen umgesetzt werden, da die Achsen fest einzelnen räumlichen Freiheitsgraden zugeordnet sind. Der beidseitige Einsatz von Industrierobotern hingegen ermöglicht eine freie räumliche Aufteilung der Bearbeitung und wird daher im Rahmen dieser Arbeit fokussiert.

Zentrales Forschungsziel ist es, eine geeignete Grundidee für die Aufteilung in synchron ausgeführte Bewegungen von Werkstück und Werkzeug zu entwickeln, die implementierbar für die unterschiedlichen proprietären Steuerungskonzepte der Roboterhersteller ist, ohne in deren Steuerungsebene einzugreifen. Daher wird ein universeller Ansatz erarbeitet, der sich in Offline-Programmiersoftware integrieren lässt und die Programme für die beteiligten Roboter automatisiert erzeugt. Somit ist sichergestellt, dass der Ansatz für eine Vielzahl von Robotersystemen zum Einsatz kommen kann. Der Benutzer gibt dabei lediglich den gewünschten werkstückbezogenen Bahnverlauf sowie einzelne Prozessparameter vor und die Software leitet daraus die entsprechenden Roboterprogramme ab.

Ein weiterer wichtiger Forschungsgegenstand dieser Arbeit ist die Flexibilität des Aufteilungsansatzes. Damit der Ansatz auch für die spezifischen Anforderungen unterschiedlicher Anwendungen geeignet ist, enthält er die Möglichkeit eine zweckmäßige Skalierung und Begrenzungen der Aufteilung vorzunehmen. Somit sind ein gezieltes Auslasten der einzelnen Roboter sowie die Berücksichtigung von prozessbedingten Randbedingungen, wie

beispielsweise eine senkrechte Werkzeugorientierung oder eine Einschränkung des Arbeitsraumes, realisierbar.

Damit MRS hinsichtlich ihres Potenzials für die Aufteilung von Bearbeitungsbahnen beurteilt werden können, ist die Entwicklung von Methoden zur Untersuchung der allgemeinen Einflüsse der Aufteilung ein weiteres Forschungsziel dieser Arbeit. Hierbei werden einerseits der Einfluss der Aufteilung auf die Bearbeitungszeit und auf die Erreichbarkeit erforscht, andererseits muss die Genauigkeit derartiger Systeme betrachtet werden, da aufgrund der verhältnismäßig langen kinematischen Ketten mit erhöhten Abweichungen zu rechnen ist. Auf Grundlage dieser Untersuchungen können MRS zukünftig so gestaltet werden, dass sie den individuellen Anforderungen von kooperierenden Anwendungen besser gerecht werden. Letztlich gilt es anhand von exemplarischen Anwendungen die potenziellen Vorteile des Ansatzes zu veranschaulichen. Da der Markt verstärkt zu kundenindividuellen Produkten tendiert, ist auch die Erfassung von Werkstücken mittels des Aufteilungsansatzes ein wichtiger Forschungsbeitrag dieser Arbeit.

1.3 Vorgehensweise und Aufbau der Arbeit

Diese Arbeit ist in sechs zentrale inhaltliche Kapitel untergliedert (siehe Bild 1), die sich in drei Kategorien einteilen lassen und abschließend in einem weiteren Kapitel zusammengefasst werden. Zunächst erfolgt eine Betrachtung der *theoretischen Grundlagen*. Im Anschluss daran werden die *wissenschaftlichen Ansätze* erläutert sowie die *praktische Umsetzung* dieser dargestellt.

Gemäß der Zielsetzung werden im Rahmen dieser Arbeit Ansätze für die Aufteilung von Bearbeitungsbahnen auf mehrere Industrieroboter und für eine Evaluation dieser entwickelt. Die dafür relevanten mathematischen und kinematischen Grundlagen der Industrierobotik werden in Kapitel 2 betrachtet. In Kapitel 3 wird auf den relevanten Stand der Wissenschaft und Technik eingegangen. Hierbei erfolgt zunächst eine Betrachtung der Aufteilung von Bearbeitungen auf mehrere Kinematiken. Des Weiteren wird auf den Einsatz von kooperierenden Industrierobotern und die damit verbundenen Besonderheiten eingegangen. Anschließend wird die Programmierung von Industrierobotern genauer analysiert. Der Schwerpunkt liegt dabei auf den gängigen Programmierverfahren sowie der Programmierung von kooperierenden Industrierobotern. Aufbauend hierauf werden abschließend der Handlungsbedarf in diesem Bereich aufgezeigt sowie die daraus abgeleiteten Ziele dieser Arbeit beschrieben.



Bild 1: Vorgehensweise und Aufbau der Arbeit

Wissenschaftlicher Kern dieser Arbeit ist Kapitel 4, in dem die Aufteilung von Bearbeitungsbahnen in eine synchrone Bewegung von Werkstück und Werkzeug erarbeitet wird. Hierbei erfolgt zunächst die Beschreibung des Grundkonzeptes. Anschließend wird die Begrenzung der Aufteilung mittels Verteilung der werkstückbezogenen Freiheitsgrade sowie mittels kartesischer und achsspezifischer Begrenzung betrachtet. Zudem werden in diesem Kapitel Ansätze für die Ermittlung der allgemeinen Einflüsse der Aufteilung mit Bezug auf Bearbeitungszeit, Erreichbarkeit und Genauigkeit definiert. In Kapitel 5 wird der zuvor entwickelte Ansatz für die Aufteilung von Bearbeitungsbahnen auf kooperierende Industrieroboter auf die Erfassung von Werkstücken übertragen. Dabei erfolgt zunächst eine Beschreibung des Scansystems sowie der notwendigen Kalibrierungen. Abschließend werden der Scanvorgang sowie die Verarbeitung der erzeugten Daten betrachtet.

Die Umsetzung der zuvor beschriebenen Ansätze wird anhand von zwei Robotersystemen in Kapitel 6 gezeigt. Dabei werden der Aufbau der Robotersysteme sowie die zentralen beteiligten Komponenten kurz beschrieben.

Außerdem erfolgt eine Übersicht über die softwaretechnische Implementierung. In Kapitel 7 erfolgt eine Evaluierung der beiden Robotersysteme. Im Zuge dessen werden sie zunächst allgemein gemäß der zuvor erarbeiteten Ansätze untersucht. Weitergehend folgen anwendungsspezifische Untersuchungen anhand von sechs exemplarischen Anwendungen. Daraufhin werden im Rahmen dieses Kapitels Bewertungskriterien für den sinnvollen Einsatz der kooperativen Bearbeitung abgeleitet.

Abgeschlossen wird diese Arbeit mit Kapitel 8, welches die gewonnenen Erkenntnisse zusammenfasst. Zudem werden denkbare und sinnvolle Erweiterungen für das entwickelte Konzept sowie die realisierten Robotersysteme aufgezeigt.

2 Mathematische und kinematische Grundlagen der Robotik

Roboter sind immer häufiger im Alltag der Menschen anzutreffen, in der Industrie sind sie schon längst unverzichtbar. Industrieroboter stellen einen wesentlichen Bestandteil der hierzulande hochautomatisierten Produktion dar. Aufgrund ihrer Flexibilität kommen sie für eine Vielzahl von unterschiedlichen Anwendungen in den Bereichen Manipulation, Bearbeitung und Messen zum Einsatz [45].

Im Rahmen dieses Kapitels wird zunächst auf die für diese Arbeit relevanten mathematischen und kinematischen Grundlagen der Robotik eingegangen. Unter die relevanten Grundlagen fällt zunächst einmal die Definition der Lage von Körpern gegenüber einem Bezugskoordinatensystem (siehe Abschnitt 2.1). Weitergehend wird auf den Aufbau und die kinematische Positionierung von Industrierobotern eingegangen (siehe Abschnitt 2.2), gefolgt von einer Erläuterung der relevanten Bewegungsarten (siehe Abschnitt 2.3). Für die meisten roboterbasierten Prozesse ist die Genauigkeit ein wichtiges Kriterium. Daher werden die Grundlagen hierzu abschließend aufgezeigt (siehe Abschnitt 2.4).

2.1 Lage eines Körpers

Die Lage, definiert durch Position und Orientierung, eines Körpers mit Bezug auf das Bezugskoordinatensystem $\{O\}$ kann über ein körperfestes Koordinatensystem $\{W\}$ definiert werden (siehe Bild 2). Die Lage des körperfesten Koordinatensystems wird dabei durch eine Verschiebung

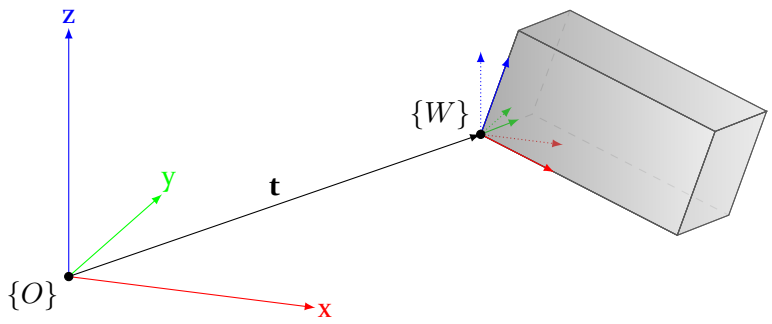


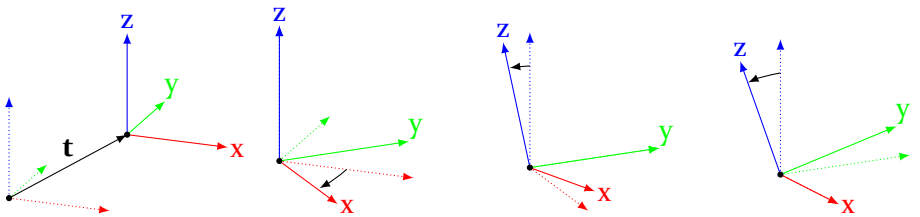
Bild 2: Objektkoordinatensystem $\{W\}$ relativ zu einem Bezugskoordinatensystem $\{O\}$

und eine Rotation gegenüber dem Bezugskoordinatensystem erreicht. Sie lässt sich entsprechend der Framenormdarstellung durch einen Vektor $\mathbf{f} = (x \;\; y \;\; z \;\; \alpha \;\; \beta \;\; \gamma)^{\mathrm{T}}$ mit $x, y, z, \alpha, \beta, \gamma \in \mathbb{R}$ ausdrücken. Dieser beinhaltet die drei Komponenten der Verschiebung in x-, y- und z-Richtung, die sich in einem Translationsvektor $\mathbf{t} = (x \;\; y \;\; z)^{\mathrm{T}}$ zusammenfassen lassen. Außerdem beinhaltet der Framenormvektor drei Komponenten der Rotation um die x-, y- und z-Achse, die sich in einem Rotationsvektor $\mathbf{r} = (\alpha \;\; \beta \;\; \gamma)^{\mathrm{T}}$ zusammenfassen lassen. Zusätzlich bedarf es noch einer Festlegung der Rotationsreihenfolge, um die Definition der Lage eindeutig zu gestalten. Im Rahmen dieser Arbeit wird die Z-Y-X-Euler-Winkeldefinition verwendet, bei der zunächst mit dem Winkel α um die z-Achse, dann mit dem Winkel β um die y-Achse und abschließend mit dem Winkel γ um die x-Achse rotiert wird (siehe Bild 3).

Eine weitere Möglichkeit die Lage eines Koordinatensystems zu definieren bieten homogene Transformationen. Hiermit lassen sich die Verschiebung und die Rotation in einer 4×4 Matrix darstellen. Eine allgemeine Transformation vom Koordinatensystem $\{A\}$ zum Koordinatensystem $\{B\}$ lässt sich durch eine Transformationsmatrix beschreiben:

$$ {}^{\mathrm{A}}\mathbf{T}_{\mathrm{B}} = \begin{bmatrix} {}^{\mathrm{A}}\mathbf{R}_{\mathrm{B}} & \mathbf{t} \\ 0_{1\times 3} & 1 \end{bmatrix} \tag{2.1} $$

Diese enthält einen Translationsvektor $\mathbf{t} = (x \, y \, z)^{\mathrm{T}}$, der die Verschiebung entlang der drei Raumachsen enthält. Des Weiteren enthält die Transformationsmatrix eine Rotationsmatrix ${}^{\mathrm{A}}\mathbf{R}_{\mathrm{B}}$, die eine Rotation zwischen den beiden Koordinatensystemen beschreibt. Diese resultiert aus der



(a) Verschiebung (b) Rotation um z-Achse (c) Rotation um y-Achse (d) Rotation um x-Achse

Bild 3: Lage eines Koordinatensystems nach Z-Y-X-Euler-Winkeldefinition

Multiplikation der drei einzelnen Matrizen der Rotationen um die jeweiligen Achsen ${}^{A}\mathbf{R}_{Bz}$, ${}^{A}\mathbf{R}_{By}$ und ${}^{A}\mathbf{R}_{Bx}$ [101] gemäß:

$$
{}^{A}\mathbf{R}_{B} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} = {}^{A}\mathbf{R}_{Bz} \cdot {}^{A}\mathbf{R}_{By} \cdot {}^{A}\mathbf{R}_{Bx} =
$$

$$
= \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) \\ 0 & \sin(\gamma) & \cos(\gamma) \end{bmatrix} = \tag{2.2}
$$

$$
= \begin{bmatrix} \cos(\alpha)\cos(\beta) & -\sin(\alpha)\cos(\gamma) + \cos(\alpha)\sin(\beta)\sin(\gamma) & \sin(\alpha)\sin(\gamma) + \cos(\alpha)\sin(\beta)\cos(\gamma) \\ \sin(\alpha)\cos(\beta) & \cos(\alpha)\cos(\gamma) + \sin(\alpha)\sin(\beta)\sin(\gamma) & -\cos(\alpha)\sin(\gamma) + \sin(\alpha)\sin(\beta)\cos(\gamma) \\ -\sin(\beta) & \cos(\beta)\sin(\gamma) & \cos(\beta)\cos(\gamma) \end{bmatrix}
$$

Sind die drei Rotationswinkel α, β und γ bei gegebener Rotationsmatrix ${}^{A}\mathbf{R}_{B}$ gesucht, so können diese für $-\pi/2 \leq \beta \leq \pi/2$ mit den folgenden Gleichungen ermittelt werden:

$$
\alpha = \operatorname{arctan2}\left(R_{21}, R_{11}\right) \tag{2.3}
$$

$$
\beta = \operatorname{arctan2}\left(-R_{31}, \sqrt{{R_{32}}^2 + {R_{33}}^2}\right) \tag{2.4}
$$

$$
\gamma = \operatorname{arctan2}\left(R_{32}, R_{33}\right) \tag{2.5}
$$

Für den Bereich $\pi/2 < \beta < 3\pi/2$ gelten die Gleichungen:

$$
\alpha = \operatorname{arctan2}\left(-R_{21}, -R_{11}\right) \tag{2.6}
$$

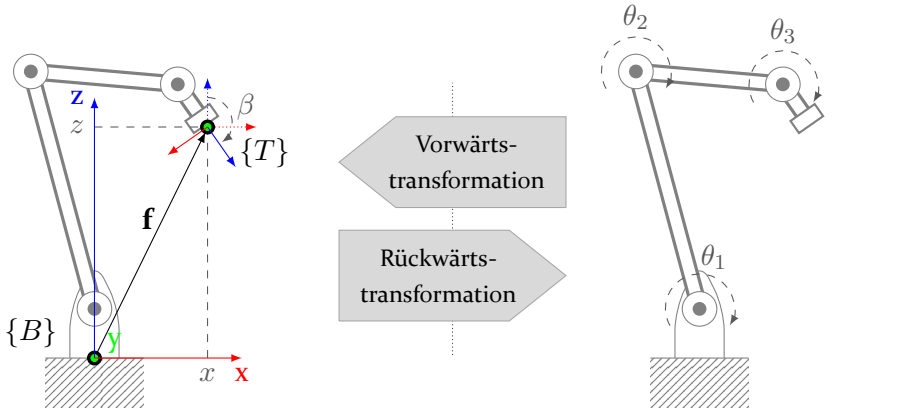
$$
\beta = \operatorname{arctan2}\left(-R_{31}, -\sqrt{{R_{32}}^2 + {R_{33}}^2}\right) \tag{2.7}
$$

$$
\gamma = \operatorname{arctan2}\left(-R_{32}, -R_{33}\right) \tag{2.8}
$$

2.2 Aufbau eines Industrieroboters

Industrieroboter lassen sich als frei programmierbare, programmgesteuerte Handhabungsgeräte definieren [119]. Prinzipiell können sie nach der Anzahl und Art der Achsen unterschieden werden, wobei primär rotatorische und translatorische Achsen zum Einsatz kommen. In Abhängigkeit von der mechanischen Struktur des Roboters, der Kinematik, kann dieser seine(n) Effektor(en) in eine definierte Pose (Position und Orientierung) gegenüber einem Bezugskoordinatensystem $\{B\}$ bringen (siehe Bild 4). Das am Effektor betrachtete Koordinatensystem wird dabei als Werkzeugkoordinatensystem $\{T\}$ und dessen Lage als *Tool Center Point* (TCP) bezeichnet. Die Pose des Werkzeugkoordinatensystems kann in Form von kartesischen Koordinaten oder in Form von Gelenkkoordinaten dargestellt werden [47, 119].

Die Berechnung der kartesischen Koordinaten des Werkzeugkoordinatensystems aus gegebenen Achsvariablen wird als *Vorwärtstransformation* oder auch direkte Kinematik bezeichnet. Im Gegensatz dazu werden bei der *Rückwärtstransformation*, auch als inverse Kinematik bezeichnet, die Gelenkkoordinaten aus einer vorgegebenen kartesischen Pose abgeleitet. Je nach Kinematik kann es dabei auch mehrere Lösungen geben. Aus der Kinematik ergibt sich darüber hinaus eine Erreichbarkeit für den Roboter. Der so aufgespannte Raum wird als Arbeitsraum bezeichnet.



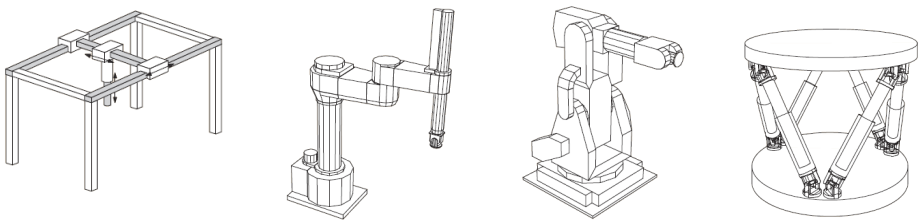
(a) Kartesisches Koordinatensystem (b) Achsspezifisches Koordinatensystem

Bild 4: Roboterspezifische Koordinatensysteme und Verfahren der Vorwärts- und Rückwärtstransformation

Folgend werden übliche Kinematiken sowie die daraus resultierenden Arbeitsräume (siehe Abschnitt 2.2.1) aufgezeigt. Des Weiteren erfolgt eine kurze Erläuterung der Grundprinzipien der beiden Transformationsrechnungen, Vorwärtstransformation (siehe Abschnitt 2.2.2) und Rückwärtstransformation (siehe Abschnitt 2.2.3).

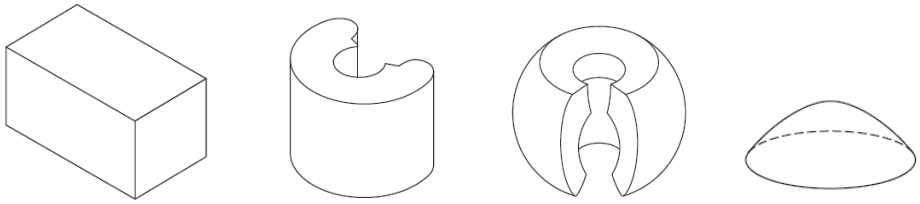
2.2.1 Kinematiken und Arbeitsräume

Die Kinematik eines Roboters ergibt sich aus der Verknüpfung mehrerer Achsen. Auf dem Markt existiert eine Vielzahl von unterschiedlichen Roboterkinematiken [46, 101], die sich in vier Grundkinematiken einteilen lassen (siehe Bild 5). Die Verknüpfung von mindestens zwei bzw. üblicherweise drei translatorischen Achsen wird als *Portalroboter* bezeichnet. Für diesen resultiert ein kartesischer Arbeitsraum (siehe Bild 6). Im Falle von drei Achsen ergibt sich ein kubischer Arbeitsraum. Als *Schwenkarm* werden Roboterkinematiken bezeichnet, die aus einer oder mehreren seriellen Rotationsachsen bestehen, die zusätzlich mit einer seriellen translatorischen Achse verknüpft sind. Der resultierende Arbeitsraum ist zylindrisch und enthält Aussparungen, da in diesem Bereich Kollisionen des Roboters mit sich selbst entstehen würden. Eine Verknüpfung von mehreren rotatorischen Achsen und das teilweise zueinander verdreht wird als *Knickarmroboter* bezeichnet. Der Arbeitsraum eines Knickarmroboters ist sphärisch und enthält ebenfalls Aussparungen. Der *Parallelroboter* besitzt mehrere parallele Achsen, die alle an beiden Enden miteinander verbunden sind. Die Achsen können dabei sowohl rotatorisch als auch translatorisch sein. Ersteres wird als Delta-Roboter bezeichnet. Im zweiten Fall spricht man von einem Hexapod-Roboter. Der resultierenden Arbeitsraum hat eine parabolische Form.



(a) Portalroboter (b) Schwenkarmroboter (c) Knickarmroboter (d) Parallelroboter

Bild 5: Übliche Kinematiken von Industrierobotern [101]



(a) Kartesisch (b) Zylindrisch (c) Sphärisch (d) Parabolisch

Bild 6: Übliche Arbeitsräume von Industrierobotern [101]

2.2.2 Vorwärtstransformation

Industrieroboter sind meist offene kinematische Ketten, bei denen jedes Armelement über eine Achse mit dem folgenden verbunden ist (siehe Bild 7). Weit verbreitet für die Definition von Industrieroboterkinematiken ist die Konvention nach *Denavit-Hartenberg* (DH) [5, 27, 119]. Bei dieser werden die N_A verknüpften Achsen jeweils durch vier Parameter ($\theta_i,\ d_i,\ a_i,\ \alpha_i$ mit $i \in \{1,\ 2,\ \ldots,\ N_\mathrm{A}\}$) beschrieben. Der Winkel θ_i spannt sich zwischen der x_{i-1}-Achse und der x_i-Achse auf. Dieser entspricht der Rotation bei einer rotatorischen Achse. Der Abstand d_i korrespondiert mit dem Abstand zwischen dem Koordinatensystem $\{i-1\}$ und dem Koordinatensystem $\{i\}$

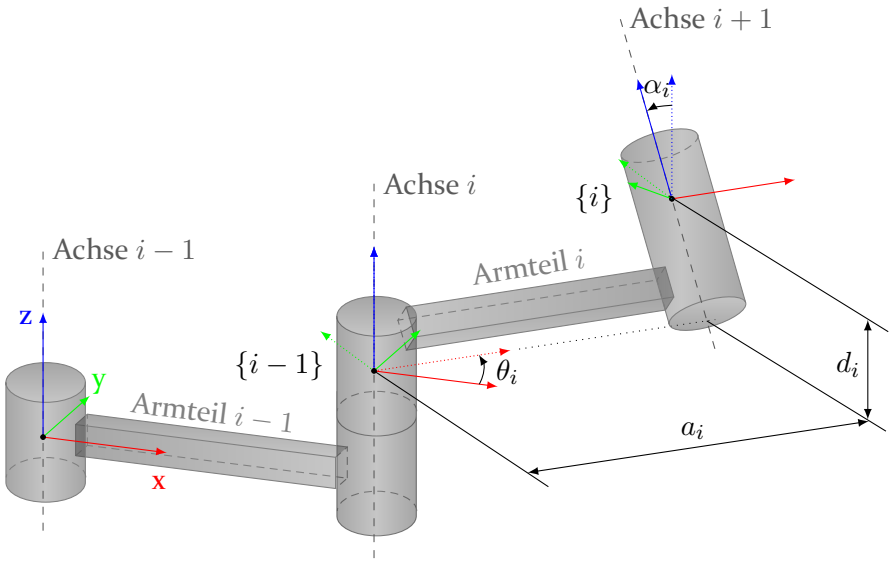


Bild 7: Beispiel einer kinematischen Kette mit Definition der DH-Parameter

entlang der z_{i-1}-Achse und entspricht der Translation bei einer translatorischen Achse. Die Distanz zwischen der z_{i-1}-Achse und der z_i-Achse in Richtung der x_i-Achse wird mit dem Parameter a_i angegeben. Dieser Parameter ist bei beiden Achstypen konstant. Die Rotation zwischen der z_{i-1}-Achse und der z_i-Achse um die x_i-Achse wird mit dem Parameter α_i bezeichnet, der ebenfalls bei beiden Achstypen konstant ist.

Die DH-Transformation ${}^{i-1}\mathbf{A}_i$ zwischen den Achskoordinatensystemen $\{i-1\}$ und $\{i\}$ lässt sich mit der folgenden Transformationsmatrix beschreiben:

$$
\begin{aligned}
{}^{i-1}\mathbf{A}_i &= \mathbf{T}_{\mathrm{rot_z}}(\theta_i) \cdot \mathbf{T}_{\mathrm{trans_z}}(d_i) \cdot \mathbf{T}_{\mathrm{trans_x}}(a_i) \cdot \mathbf{T}_{\mathrm{rot_x}}(\alpha_i) = \\
&= \begin{bmatrix}
\cos(\theta_i) & -\sin(\theta_i)\cdot\cos(\alpha_i) & \sin(\theta_i)\cdot\sin(\alpha_i) & a_i\cdot\cos(\theta_i) \\
\sin(\theta_i) & \cos(\theta_i)\cdot\cos(\alpha_i) & -\cos(\theta_i)\cdot\sin(\alpha_i) & a_i\cdot\sin(\theta_i) \\
0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\
0 & 0 & 0 & 1
\end{bmatrix}
\end{aligned}
\tag{2.9}
$$

Durch Verknüpfung der einzelnen DH-Transformationen bis zur Achse N_A kann die Transformation ${}^{B}\mathbf{T}_{N_A}$ von der Basis des Roboters zur letzten Achse bestimmt werden. Diese kann mit der Transformation ${}^{N_A}\mathbf{T}_T$ von der letzten Achse zum TCP verknüpft werden, um die Transformation ${}^{B}\mathbf{T}_T$ des Werkzeugkoordinatensystems mit Bezug auf die Basis zu beschreiben, gemäß:

$$
\begin{aligned}
{}^{B}\mathbf{T}_T &= {}^{B}\mathbf{T}_{N_A} \cdot {}^{N_A}\mathbf{T}_T = \\
&= \overbrace{{}^{B}\mathbf{A}_1 \cdot {}^{1}\mathbf{A}_2 \cdot \ldots \cdot {}^{N_A-1}\mathbf{A}_{N_A}}^{} \cdot {}^{N_A}\mathbf{T}_T
\end{aligned}
\tag{2.10}
$$

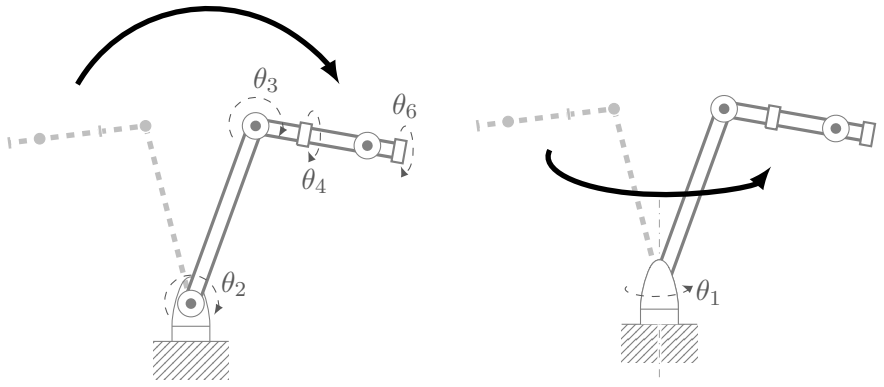
2.2.3 Rückwärtstransformation

Für Roboteranwendungen ist es oftmals erforderlich für vorgegebene Stellungen des Werkzeugkoordinatensystems Achswerte zu bestimmen. Dieser Vorgang wird auch als *inverse Kinematik* bezeichnet. Hierbei kann es zu Mehrdeutigkeiten oder sogar Singularitäten, also unendlich viele mögliche Achsstellungen, kommen. Deshalb ist in diesen Fällen das Festlegen einer Lösung erforderlich, um die Bewegung der Kinematik eindeutig zu definieren.

Bei einem sechsachsigen Knickarmroboter ergeben sich drei Mehrdeutigkeiten [121]. Zunächst gibt es eine Mehrdeutigkeit der Armstellung (siehe Bild 8), denn der Arm kann entweder auf der linken oder auf der rechten Seite des Roboter positioniert sein. Hiervon hängt ab, ob der Zielpunkt durch eine Überkopfbewegung oder eine Drehung der ersten Achse erreicht wird, woraus sich der Schwenkbereich des Roboters ergibt. Des Weiteren gibt es eine Mehrdeutigkeit für die Achsen 4 und 6, wenn diese fluchten (siehe Bild 9). In diesem Fall können die Achsen beliebig verdreht werden, es gibt immer eine Stellung der anderen Achse, die zu einer Lösung führt. Somit gibt es unendliche viele Lösungskombinationen für die beiden Achsen. Die dritte Mehrdeutigkeit liegt bei den Achsorientierungen der Achsen 3 und 5 vor (siehe Bild 10). Hierfür sind zwei Kombinationen möglich, entweder ist der Ellenbogen nach oben oder nach unten geknickt.

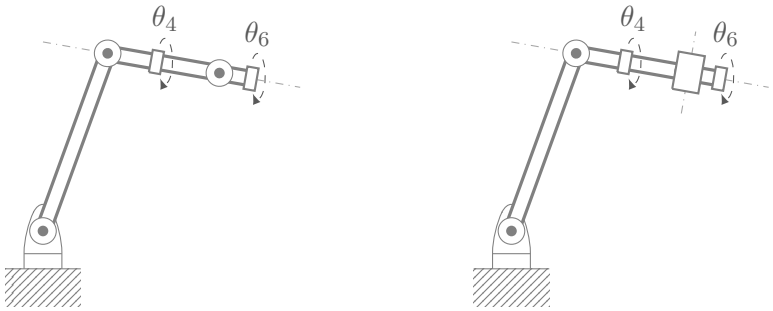
Die Mehrdeutigkeiten müssen von der Steuerungssoftware durch spezielle Auswahlkriterien geklärt werden. Mögliche Entscheidungskriterien sind der mechanisch begrenzte Achsverfahrbereich, die Kontinuität der Bewegungsbahn oder der kürzeste Weg zum Zielpunkt. Bei handelsüblichen Knickarmrobotern erfolgt meist eine analytische Lösung über die Zentralhand (Achse 4-6). Bei Roboterarmen ohne Zentralhand oder mit mehr als sechs Achsen ist nur eine numerische Lösung möglich.

Weitergehend wird eine analytische Lösung für handelsübliche Knickarmroboter mit sechs Achsen aufgezeigt [119]. Hierbei wird zunächst



(a) Arm rechts (b) Arm links

Bild 8: Mehrdeutigkeit der Armstellung aufgrund einer Überkopf- oder Drehbewegung bei der Rückwärtstransformation von sechsachsigen Knickarmrobotern



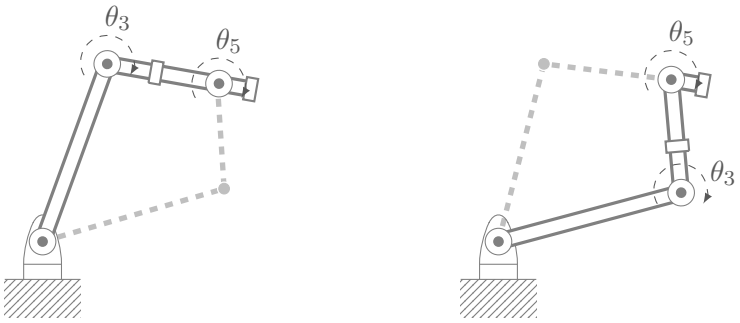
(a) Stellung N (b) Stellung N+1

Bild 9: Singularität der Achsorientierung Achse 4 und 6 bei der Rückwärtstransformation von sechsachsigen Knickarmrobotern

der Rotationswinkel der ersten Roboterachse θ_1 anhand des Ortsvektors $\mathbf{p}_{04} = \left(x_{\mathbf{p}_{04}}\ y_{\mathbf{p}_{04}}\ z_{\mathbf{p}_{04}}\right)^{\mathrm{T}}$, der vom Koordinatensystem in der ersten Roboterachse $\{0\}$ zum Koordinatensystem in der fünften Roboterachse $\{4\}$ zeigt, bestimmt durch:

$$\theta_1 = \operatorname{arctan2}\left(y_{\mathbf{p}_{04}}, x_{\mathbf{p}_{04}}\right) \tag{2.11}$$

Die Lage des Koordinatensystems in der fünften Roboterachse ergibt sich dabei aus dem Schnittpunkt der letzten drei Gelenkachsen. Basierend hierauf kann der Ortsvektor $\mathbf{p}_{14} = \left(x_{\mathbf{p}_{14}}\ y_{\mathbf{p}_{14}}\ z_{\mathbf{p}_{14}}\right)^{\mathrm{T}}$ ermittelt werden, über dessen euklidische Norm der Winkel der dritten Achse θ_3 unter Verwendung



(a) Ellenbogen oben (b) Ellenbogen unten

Bild 10: Mehrdeutigkeit der Achsorientierung Achse 3 und 5 bei der Rückwärtstransformation von sechsachsigen Knickarmrobotern

der Armlängen des zweiten und des vierten Armteils (l_2 und l_4) berechnet werden kann durch:

$$\theta_3 = \frac{3}{2} \cdot \pi - \arccos\left(\frac{{l_2}^2 + {l_4}^2 - ||\mathbf{p}_{14}||^2}{2 \cdot l_2 \cdot l_4}\right) \tag{2.12}$$

Die Berechnung des Winkels der zweiten Achse θ_2 erfolgt anhand des um θ_1 gedrehten Ortsvektor $\mathbf{p}_{14}$, der mit $\mathbf{p}'_{14} = \left(x'_{\mathbf{p}_{14}}\ y'_{\mathbf{p}_{14}}\ z'_{\mathbf{p}_{14}}\right)^{\mathrm{T}}$ bezeichnet wird, durch:

$$\theta_2 = -\arctan2\left(-y'_{\mathbf{p}_{14}}, x'_{\mathbf{p}_{14}}\right) - \arccos\left(\frac{{l_2}^2 - {l_4}^2 + ||\mathbf{p}'_{14}||^2}{2 \cdot l_2 \cdot ||\mathbf{p}'_{14}||}\right) \tag{2.13}$$

Anschließend kann der Winkel der fünften Achse θ_5 aus dem Skalarprodukt zwischen den Einheitsvektoren der z-Achsen in der vierten Roboterachse $\mathbf{z}_3$ und im Roboterflansch, der dem Normalenvektor des Flansches entspricht $\mathbf{z}_6 = \mathbf{n}$, bestimmt werden durch:

$$\theta_5 = \arccos\left(\mathbf{z}_3 \cdot \mathbf{z}_6\right) \tag{2.14}$$

Der Winkel der vierten Achse θ_4 ergibt sich aus dem Einheitsvektor der y-Achsen in der vierten Roboterachse $\mathbf{y}_3$ und den Einheitsvektoren der z-Achsen in der vierten Roboterachse $\mathbf{z}_3$ und im Roboterflansch $\mathbf{z}_6$ gemäß:

$$\theta_4 = \pi \pm \arccos\left(\mathbf{y}_3 \cdot \frac{\mathbf{z}_6 \times \mathbf{z}_3}{||\mathbf{z}_6 \times \mathbf{z}_3||}\right) \tag{2.15}$$

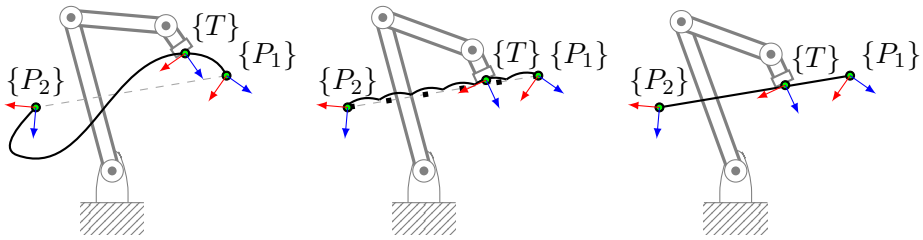
Abschließend kann der Winkel der sechsten Achse θ_6 anhand des Skalarprodukts zwischen den Einheitsvektoren der x-Achsen in der sechsten Roboterachse $\mathbf{x}_5$ und im Roboterflansch $\mathbf{x}_6$ ermittelt werden durch:

$$\theta_6 = \pm \arccos\left(\mathbf{x}_5 \cdot \mathbf{x}_6\right) \tag{2.16}$$

2.3 Bewegungsarten

Der Bahnverlauf, der mit dem TCP eines Industrieroboters abgefahren werden soll, wird durch eine Folge von Posen definiert. Der Weg zwischen diesen Posen kann durch unterschiedliche Bewegungsarten zurückgelegt werden [46, 47, 119, 121], die sich in drei Kategorien einteilen lassen (siehe Bild 11). Bei einer Bewegung vom Typ *Point-To-Point* (PTP) erfolgt das Anfahren der Posen, ohne dass der Bahnverlauf dazwischen festgelegt wird. Stattdessen wird der Weg- oder Winkelbereich der einzelnen Roboterachsen durchfahren. Die ausgeführte Bewegung ergibt somit eine nicht vorhersehbare Raumkurve. Bahnen, die nicht durch einfache geometrische Teilbahnen definiert werden können, lassen sich durch Posen in einem engen zeitlichen Raster nachbilden. In diesem Fall spricht man von einer Bewegung vom Typ *Multi-Point* (MP). Bei einem durch geometrische Teilbahnen definierten Verlauf (z. B. Linear-, Kreis- oder Splinebahn) spricht man von einer Bewegung vom Typ *Continuous-Path* (CP). Zum Abfahren dieser Bahnverläufe müssen alle Achsen kontinuierlich koordiniert werden, was meist zu größeren Bewegungszeiten führt. Dies erfolgt üblicherweise durch eine steuerungsinterne Interpolation der Bahnkurven anhand von Zwischenpunkten, die in einem engen Zeitraster abgestuft sind.

Um das Abfahren einer Bahn zeitoptimal und ruckfrei zu ermöglichen, können die Bahnpunkte überschliffen werden. Das bedeutet, die Bewegung wird vor Erreichen des Bahnpunktes bereits in Richtung des nächsten Bahnpunktes umgeleitet. Dabei wird zwar der Bahnpunkt nicht exakt angefahren, aber der Roboter kommt nicht zum Stillstand, wodurch eine flüssigere Bewegung entsteht. Das Überschleifen kann anhand zweier Kriterien definiert werden. Einerseits anhand der Bahngeschwindigkeit und andererseits anhand des noch zurückzulegenden Weges. Im ersten Fall soll eine Geschwindigkeit von



(a) PTP-Bewegung (b) MP-Bewegung (c) CP-Bewegung

Bild 11: Bewegungsarten von Industrierobotern

z. B. 70 % nicht unterschritten werden. Sobald diese Grenze erreicht ist, erfolgt das Umschwenken in Richtung des nächsten Bahnpunktes, sodass die Geschwindigkeit wieder erhöht werden kann. Das Wegkriterium verhält sich gleichermaßen, nur wird in diesem Fall die Distanz zum Bahnpunkt verwendet. Wird ein Grenzabstand von z. B. 50 mm erreicht, folgt der Richtungswechsel.

2.4 Genauigkeit

Aufgrund seiner Mechanik ist ein Industrieroboter in seiner Genauigkeit begrenzt. Hierbei lässt sich zwischen der *Positioniergenauigkeit* und der *Wiederholgenauigkeit* unterscheiden (siehe Bild 12). Die Positioniergenauigkeit ergibt sich aus dem Unterschied zwischen programmierter und tatsächlich angefahrener Position oder Bahn. Sie resultiert aus Ungenauigkeiten und Nachgiebigkeiten im System. Die Wiederholgenauigkeit definiert die

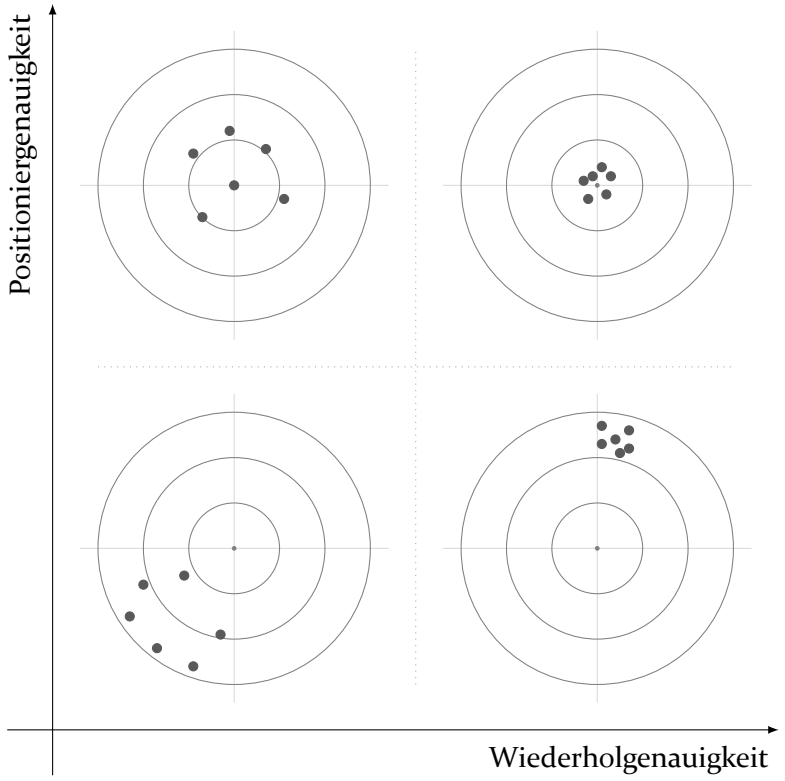


Bild 12: Kombinationen zwischen Positionier- und Wiederholgenauigkeit

Fähigkeit eine zuvor angefahrene Position oder Bahn wieder zu erreichen. Sie resultiert aus temporalen Veränderungen im System, beispielsweise verursacht durch Temperaturänderungen.

Weitergehend erfolgt eine kurze Erläuterung der Genauigkeitsmessung bei Industrierobotern einschließlich einer Beschreibung der zugehörigen Parameter (siehe Abschnitt 2.4.1). Anschließend werden die grundlegenden Möglichkeiten zur Verbesserung der Genauigkeit von Industrierobotern mittels Korrekturschleifen aufgezeigt (siehe Abschnitt 2.4.2).

2.4.1 Messung

Für die Untersuchung der Genauigkeit von Industrierobotern werden hochpräzise Messsysteme verwendet, mit denen durch den Roboter angefahrene Positionen erfasst werden können. Hierbei kommen unter anderem akustische Sensoren [122], visuelle Sensoren [24] oder Koordinatenmessmaschinen [34] zum Einsatz, deren Genauigkeit im Submillimeterbereich liegt. Die meisten der Ansätze verwenden aufgrund ihrer hohen Genauigkeit automatische Theodolite, auch bekannt als Laser-Tacker [4, 6, 80]. Hiermit können durch den Roboter angefahrene Positionen mit Genauigkeiten bis zu wenigen µm erfasst werden.

Während der Messungen wird die Position des TCP im Arbeitsraum mit Bezug auf das Sensorkoordinatensystem $\{S\}$ erfasst (siehe Bild 13). Ein untersuchter Roboter besitzt ein Basiskoordinatensystem $\{B\}$. Dieses liegt

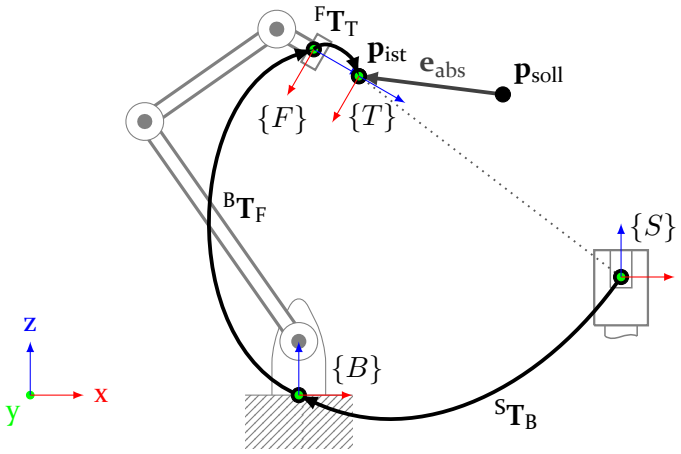


Bild 13: Koordinatensysteme und Transformationen für die Genauigkeitsuntersuchung

meist im Sockel des Roboters, die genaue Position ist jedoch unbekannt, da sie nicht zugänglich ist. Daher muss die Basis mit dem Sensor messtechnisch erfasst werden. Hierbei wird die Transformation vom Sensorkoordinatensystem $\{S\}$ zur Roboterbasis ${}^{S}\mathbf{T}_{B}$ bestimmt. Des Weiteren ist für den Roboter ein Flanschkoordinatensystem $\{F\}$ definiert, dessen Lage gegenüber des Basiskoordinatensystems mit der Transformation ${}^{B}\mathbf{T}_{F}$ beschrieben wird. Für viele Roboteranwendungen ist zudem ein Werkzeugkoordinatensystem $\{T\}$ mit Bezug auf den Flansch definiert. Die zugehörige Transformation lautet dementsprechend ${}^{F}\mathbf{T}_{T}$. Wenn die exakte Lage des Werkzeugkoordinatensystems nicht bekannt ist, muss diese mittels Kalibrierung ermittelt werden.

Die Definition sinnvoller Messpunkte hängt stark von den für das Robotersystem relevanten Anwendungen ab. So wurden beispielsweise in [4] $N_{M} = 85$ Positionen, die den Bewegungsbereich für die betrachteten Anwendungen abdecken, heuristisch festgelegt. Um die Anzahl der nötigen Messpunkte möglichst niedrig zu halten, kann des Weiteren ein generischer Algorithmus verwendet werden, mit dem effektive Messpositionen ausgewählt werden [7].

Jede der N_{M} gemessenen Positionen $\mathbf{p}_{ist}$ kann mit der zugehörigen Zielposition $\mathbf{p}_{soll}$ verglichen werden, indem ein Vektor des absoluten Fehlers $\mathbf{e}_{abs} = \begin{pmatrix} x_{E_{abs}} & y_{E_{abs}} & z_{E_{abs}} \end{pmatrix}^{T}$ berechnet wird:

$$\mathbf{e}_{abs_i} = \mathbf{p}_{ist_i} - \mathbf{p}_{soll_i}\,, \quad i \in \{1,\ 2,\ \ldots,\ N_{M}\} \quad \text{mit} \quad \mathbf{p} = \begin{pmatrix} x & y & z \end{pmatrix}^{T} \tag{2.17}$$

Des Weiteren ergibt sich für jede Messposition ein Betrag des absoluten Fehlers ϵ_{abs} anhand des euklidischen Abstandes der beiden Positionen:

$$\begin{gathered} \epsilon_{abs_i} = ||\mathbf{e}_{abs_i}|| = ||\mathbf{p}_{ist_i} - \mathbf{p}_{soll_i}|| = \\ = \sqrt{(x_{ist_i} - x_{soll_i})^2 + (y_{ist_i} - y_{soll_i})^2 + (z_{ist_i} - z_{soll_i})^2}\,, \\ i \in \{1,\ 2,\ \ldots,\ N_{M}\} \end{gathered} \tag{2.18}$$

2.4.2 Korrektur

Die Genauigkeit hat einen entscheidenden Einfluss auf die Qualität der mittels Roboter durchgeführten Prozesse. Um eine ausreichende Prozessqualität zu gewährleisten, muss daher oftmals eine Korrektur des für die

Steuerung des Roboters verwendeten kinematischen Modells durchgeführt werden. Hierfür gibt es zwei Grundprinzipien: die *geschlossene Korrekturschleife* und die *offene Korrekturschleife* [31].

Bei der geschlossenen Korrekturschleife (vgl. [22]) wird die tatsächlich angefahrene Roboterpose kontinuierlich durch ein Sensorsystem erfasst und mit der Sollpose verglichen (siehe Bild 14). Anhand des Unterschieds der beiden Posen wird eine Anpassung der Achswerte ermittelt und auf den Roboter übertragen. Somit wird die Genauigkeit im Rahmen der Messgenauigkeit des Sensorsystems kontinuierlich erhöht. Dies bedingt jedoch die permanente Einbindung des Sensorsystems in die Roboteranlage.

Bei einer offenen Korrekturschleife erfolgt eine Anpassung des für die Berechnung der inversen Kinematik verwendeten kinematischen Modells an das reale Robotersystem (siehe Bild 15). Hierbei wird durch mehrere Messungen die Abweichung des kinematischen Modells erfasst und dementsprechend das Modell angepasst. Für diese Korrektur lassen sich vier Schritte definieren [36]. Zunächst wird das kinematische Modell für das Robotersystem aufgestellt. Anschließend erfolgt die Durchführung der notwendigen Messungen. Auf Grundlage der erzeugten Messdaten kann weitergehend eine Abweichung im kinematischen Modell ermittelt werden. Anhand dieser kann das Modell abschließend angepasst werden, sodass eine Kompensation erfolgt. Wenn mit der Zeit erneut relevante Abweichungen im System auftreten, muss die Korrektur wiederholt werden, um eine

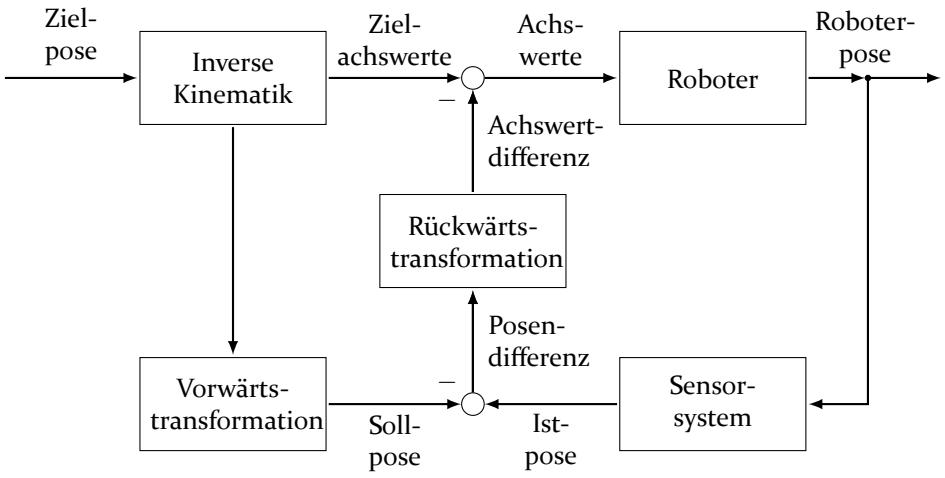


Bild 14: Methode der geschlossenen Korrekturschleife für Industrieroboter

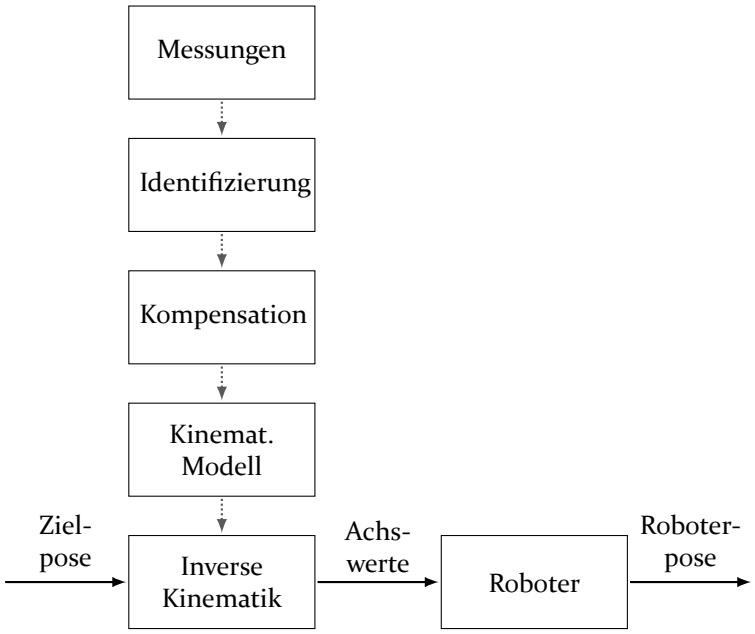


Bild 15: Methode der offenen Korrekturschleife für Industrieroboter

ausreichende Prozessgenauigkeit zu erhalten. Da die Messungen jedoch nicht kontinuierlich notwendigen sind, können hierbei auch externe Messsysteme eingesetzt werden.

3 Stand der Wissenschaft und Technik

Gegenwärtig ist die *Mensch-Roboter-Kollaboration* (MRK) – das Arbeiten von Mensch und Roboter innerhalb eines gemeinsamen Arbeitsraumes – einer der Trends im Bereich Industrierobotik. Dennoch bietet auch die Kooperation zwischen Robotern weiteres Potenzial für Optimierungen sowie die Erschließung von industriellen Prozessen. Daher liegt der Fokus dieser Arbeit auf einer neuen Art der Kooperation von Industrierobotern, bei der eine Aufteilung der Bearbeitung auf mehrere Industrieroboter erfolgt. Da dies eine verhältnismäßig komplexe Programmierung der Roboter erfordert, ist die Erzeugung der Programme ein wichtiger Bestandteil dieser Arbeit. Die Möglichkeiten der Programmerzeugung für Industrieroboter sind mittlerweile zahlreich, wobei diese zunehmend intuitiver gestaltet werden, um den Vorgang zu beschleunigen oder komplexere Programme erstellen zu können. Dies geht bis hin zur völligen Übernahme der Programmierung durch Softwarealgorithmen. Dabei kommt auch immer häufiger eine Werkstückerfassung zum Einsatz, um individuelle Werkstücke schnell und gezielt bearbeiten zu können.

Im Rahmen dieses Kapitels wird zunächst auf aktuelle Systeme mit einer Aufteilung der Bearbeitung auf mehrere Kinematiken eingegangen (siehe Abschnitt 3.1). Anschließend erfolgt die Betrachtung von Industrierobotern, die in gemeinsamen Anwendungen kooperieren (siehe Abschnitt 3.2). Eine Beschreibung der gegenwärtigen Programmiermöglichkeiten von Industrierobotern folgt im nächsten Abschnitt (siehe Abschnitt 3.3). Hierbei wird auch genauer auf die spezifische Programmierung von kooperierenden Industrierobotern eingegangen. Den Abschluss des Kapitels bildet die Darstellung des fokussierten Handlungsbedarfes, der auf der Verknüpfung der vorhergehenden Abschnitte dieses Kapitels basiert (siehe Abschnitt 3.4).

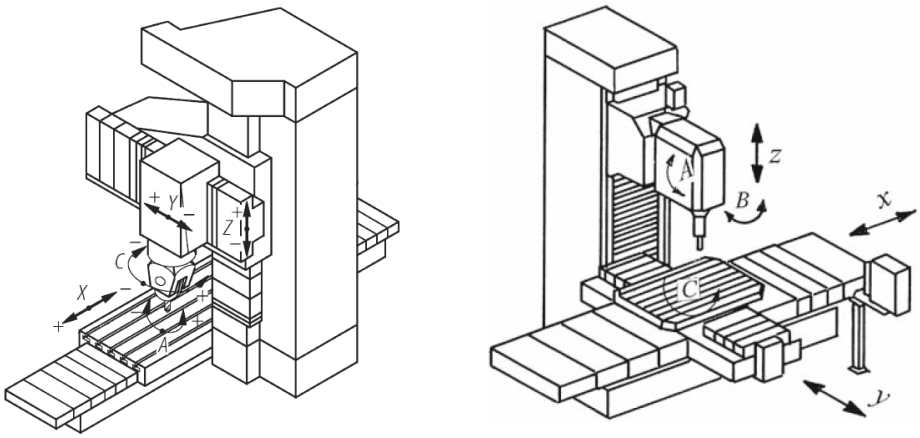
3.1 Aufteilung von Bearbeitungen auf mehrere Kinematiken

Die Aufteilung von Bearbeitungen auf mehrere Kinematiken erfolgt üblicherweise um zwei Vorteile zu erzielen. Einerseits kann dadurch die Prozessgenauigkeit verbessert werden, indem zwei kürzere kinematische Ketten zu erhöhten Steifigkeiten im System führen. Andererseits kann die Erreichbarkeit verbessert werden, indem die Achsen zielführender angeordnet werden.

Gegenwärtig existieren zwei etablierte Systeme mit einer Aufteilung der Bearbeitungsbewegung auf mehrere Kinematiken. Weit verbreitet sind NC-gesteuerte Bearbeitungsmaschinen, die im nächsten Abschnitt genauer betrachtet werden (siehe Abschnitt 3.1.1). Ein roboterbasiertes System stellt die Kombination eines Industrieroboters mit einem automatisierten Rotationstisch (Dreh-/Kipp- und/oder Schwenktisch) dar. Hierauf wird weitergehend ebenfalls genauer eingegangen (siehe Abschnitt 3.1.2).

3.1.1 NC-gesteuerte Bearbeitungsmaschinen

Bei NC-gesteuerten Bearbeitungsmaschinen gibt es eine Vielzahl an Bauformen [43, 86, 120], wobei die Anzahl, die Anordnung und die Verteilung der Achsen zwischen Werkzeugträger und Werkstückträger variiert. Alle Achsen werden dabei von einer gemeinsamen NC-Steuerung angesteuert. Es gibt Bearbeitungsmaschinen, die lediglich Achsen auf der Werkzeug- oder der Werkstückseite besitzen. Oftmals erfolgt jedoch eine beidseitige Anordnung der Achsen (siehe Bild 16), um zwei kürzere kinematische Ketten zu erhalten und somit eine erhöhte Steifigkeit im System zu erzielen. Dabei sind bei konventionellen Bearbeitungsmaschinen die beiden Kinematiken seriell angeordnet und jede Achse (X, Y, Z, A, B, C) ist fest einem räumlichen Freiheitsgrad zugeordnet. Somit können die räumlichen Bewegungen nicht individuell auf die einzelnen Achsen verteilt werden, sondern es gibt immer eine



(a) Einständer-Langfräsmaschine mit Ausleger [43]

(b) Kreuztischfräsmaschine [86]

Bild 16: Exemplarische NC-gesteuerte Bearbeitungsmaschinen mit beidseitiger Anordnung von Achsen

feste Zuordnung der Bewegungsparameter. Selbst bei Bearbeitungsmaschinen mit mehreren Achsen, die einen Freiheitsgrad abdecken (siehe Bild 17), kann lediglich eine Aufteilung entlang der Achsrichtung erfolgen und nicht entlang einer werkstückbezogenen, räumlichen Bearbeitungsbahn. Da bei NC-gesteuerten Bearbeitungsmaschinen meist die Steifigkeit im Fokus steht, um Prozesskräfte aufnehmen zu können und um die geforderte Qualität der bearbeiteten Werkstücke zu erzielen, werden tendenziell möglichst wenige Achsen auf den beiden Seiten eingesetzt.

3.1.2 Industrieroboter mit Rotationstisch

Ein weiteres System, bei dem beidseitig Kinematiken eingesetzt werden, um das Werkzeug und das Werkstück parallel zu handhaben, sind Industrieroboter in Verbindung mit einem Rotationstisch. Letzterer wird auch als Dreh-, Kipp- und/oder Schwenktisch bezeichnet, je nachdem welche räumlichen Rotationsfreiheitsgrade abgedeckt werden.

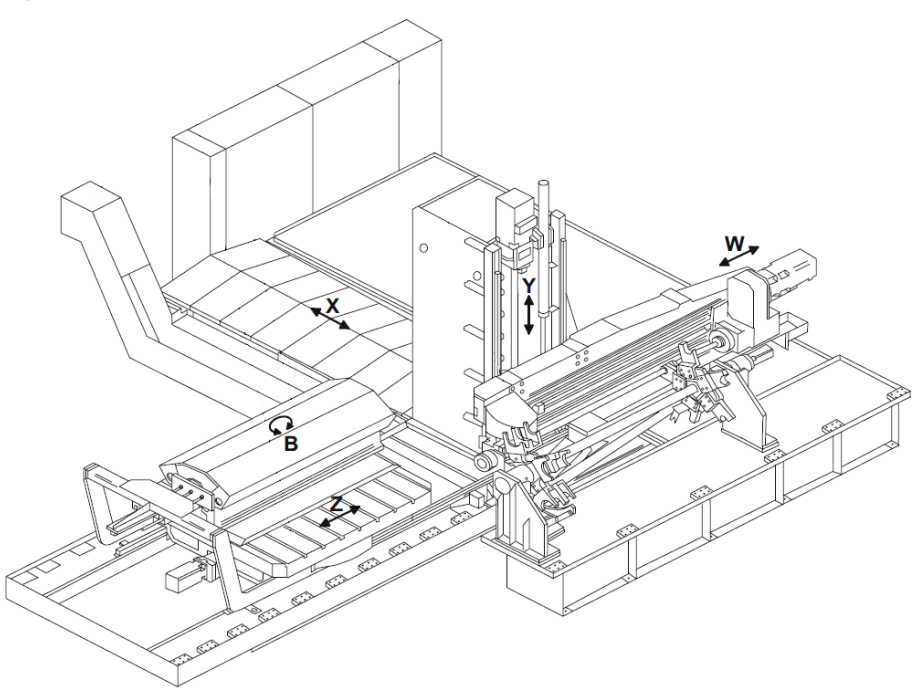


Bild 17: Tiefbohrzentrum mit zwei Achsen (Z und W), die den selben Freiheitsgrad abdecken [120]

Derartige Systeme werden meist eingesetzt, um die Zugänglichkeit am Werkstück zu erhöhen, indem dieses durch den Rotationstisch umpositioniert wird. Dabei werden die Bewegungen von Werkzeug und Werkstück meist seriell ausgeführt. So kann beispielsweise die erforderliche Erreichbarkeit beim Nahtschweißen von komplexeren Geometrien erreicht werden (siehe Bild 18). Ein weiteres Beispiel ist das roboterbasierte Scannen, bei dem ein Drehtisch eingesetzt wird, um den Sensor rundum gezielt auf der Werkstückoberfläche ausrichten zu können [16, 38] und somit die Werkstücke vollständig sowie mit ausreichender Qualität erfassen zu können. Die Bearbeitung in Vorzugslage ist ein weiterer potenzieller Vorteil derartiger Systeme. Eine exemplarische Anwendung hierfür ist die Oberflächenbearbeitung durch einen sechsachsigen Knickarmroboter in Kombination mit einem Dreh- und Kipptisch [30]. Hierbei kann das Werkzeug stets in einer gewünschten Orientierung gehalten werden.

Es zeigt sich, dass die Kombination einer Roboterkinematik mit einem Rotationstisch eine verbesserte Erreichbarkeit sowie die Bearbeitung in Vorzugslage ermöglicht. Jedoch erfolgt bei bisherigen Systemen keine freie Aufteilung der Bearbeitungsbewegungen auf die beiden Kinematiken, sondern lediglich eine zielgerichtete Veränderung der Werkstückorientierung.

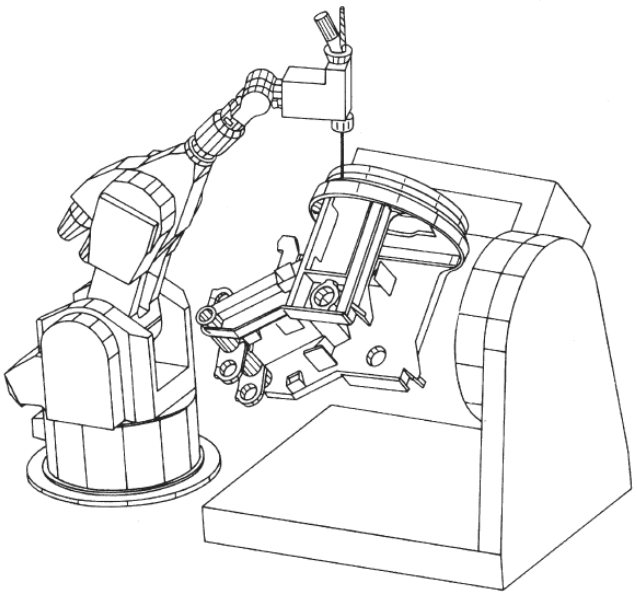


Bild 18: Sechsachsiger Knickarmroboter mit Dreh- und Kipptisch zum Nahtschweißen [43]

3.2 Kooperierende Industrieroboter

Wenn mehrere Roboter in einem gemeinsamen System zum Einsatz kommen, wird dieses als MRS bezeichnet. Vor allem in der mobilen Robotik kommen MRS zunehmend zum Einsatz, um Aufgaben auf mehrere Roboter zu verteilen [108]. Exemplarische Anwendungen hierfür sind die Exploration [21], die bildbasierte 3D-Kartierung [93], die Positionsbestimmung von Objekten [104] und die Montage [57]. Auch in der Industrierobotik werden MRS eingesetzt, um die Produktionsprozesse auf mehrere Roboter zu verteilen. Wie auch beim Einsatz von Industrierobotern allgemein, ist die Automobilindustrie beim Einsatz von MRS Vorreiter in Deutschland [131]. Der Anteil an Roboteranwendungen mit kooperierenden Industrierobotern ist dennoch verhältnismäßig gering und vor allem für kleine und mittelständische Unternehmen ist der Einsatz derartiger Systeme noch nicht attraktiv, da der Mehraufwand gegenüber herkömmlichen Roboteranlagen nicht in einem ausreichenden Verhältnis zum Nutzen steht.

Weitergehend wird auf die unterschiedlichen *Koordinationsarten* (siehe Abschnitt 3.2.1) innerhalb von MRS sowie die dabei verwendeten *Kommunikationsstrukturen* (siehe Abschnitt 3.2.2) eingegangen. Eine Zusammenfassung des aktuellen Standes der *Programmierung* von kooperierenden Industrierobotern erfolgt separat im Rahmen des nächsten übergeordneten Abschnittes (siehe Abschnitt 3.3.3).

3.2.1 Koordinationsarten

Sobald mehrere Roboter innerhalb einer Anlage zum Einsatz kommen, müssen diese aufeinander abgestimmt werden. Im einfachsten Fall findet dabei zwischen den beteiligten Robotern keinerlei aktive Koordination statt, sondern jeder Roboter führt lediglich seine eigenen Programme aus und der übergreifende Prozessablauf muss vorab beim Programmieren berücksichtigt werden. Wenn mehrere Industrieroboter koordiniert zusammenarbeiten, spricht man von kooperierenden Industrierobotern. Die Koordination zwischen den Robotern kann dabei unterschiedlich komplex ausfallen und lässt sich in drei Stufen untergliedern (siehe Bild 19). Werden einzelne Signale zwischen den Robotern ausgetauscht, wird dies als *asynchrone Koordination* bezeichnet. Ein einfaches Beispiel hierfür ist das Senden eines booleschen Signales bei der Bereitstellung durch einen Transferroboter. Exemplarische Anwendungen hierfür sind Montageanwendungen, wie beispielsweise das zweiarmige Einsetzen von Holzdübeln in Möbelteile [106] oder die Montage einer Autotür [94]. Hierbei wird zunächst der

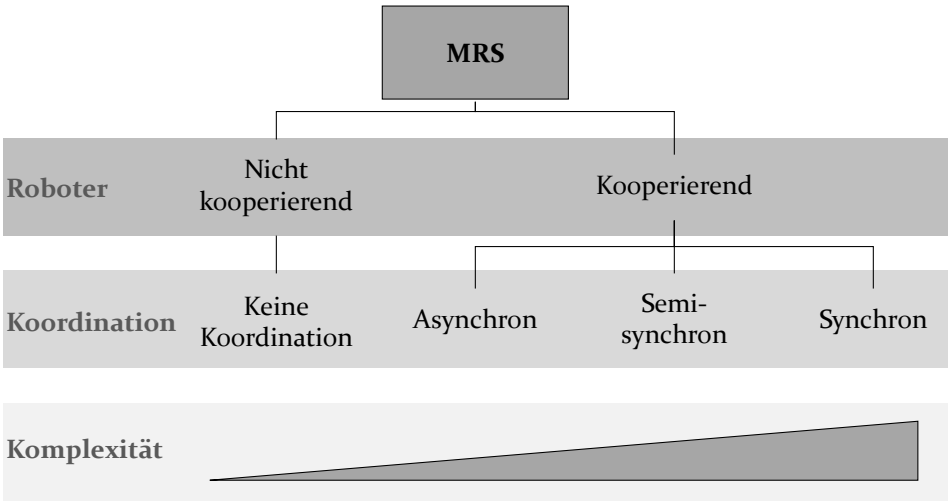


Bild 19: Koordinationsarten zwischen mehreren Industrierobotern

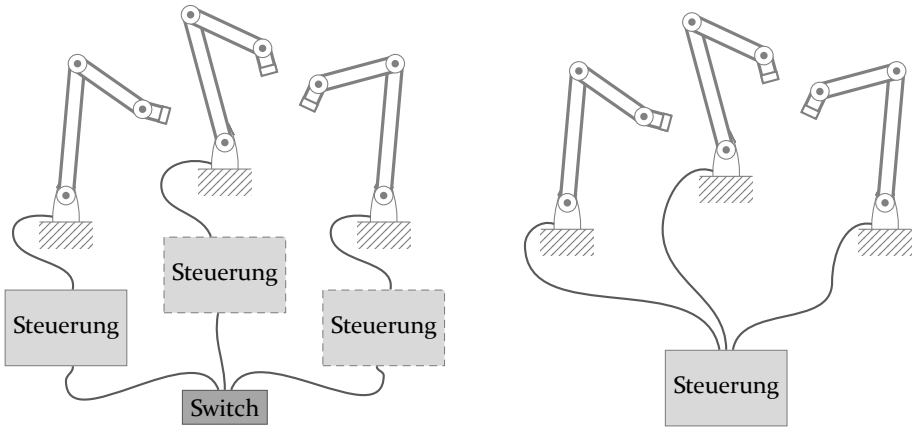
Holzdübel durch einen der beiden Roboter aufgenommen, anschließend erfolgt die Bereitstellung des Möbelteils durch den anderen Roboter und abschließend wird der Holzdübel vom erstgenannten Roboter eingefügt. Eine deutlich intensivere Kommunikation findet bei der *semi-synchronen* Koordination statt, bei der die Bewegungen systematisch (z. B. identische oder gespiegelte Bewegung) koordiniert werden. Eine häufig realisierte Anwendung mit dieser Art der Koordination ist das gemeinsame Handhaben von Lasten [57, 77, 102, 114, 130]. Dieser Ansatz wird beispielsweise in einem Assistenzsystem für Schweißprozesse [112] eingesetzt, um größere Rohr- und Rahmenkonstruktionen gezielt gegenüber dem Mitarbeiter zu positionieren. Die inkrementelle Blechumformung durch zwei Roboterarme [19, 60] ist eine exemplarische Anwendung, bei der zwei Werkzeuge definiert von den Robotern zueinander positioniert werden. Ein weiteres Beispiel ist die zweiarmige Tumorerkennung und Zerstörung [15]. Hierbei führt ein Roboter einen Ultraschallsensor für die Tumorerkennung und der andere Roboter eine Nadel für die Tumorzerstörung. Die komplexeste Stufe der Koordination ist die *synchrone Koordination*, bei der die einzelnen Bewegungen individuell verknüpft werden, wie etwa beim Montieren einer komplexen Baugruppe. Eine exemplarische Anwendung hierfür ist das kraftgeführte Fügen eines Stiftes in eine Hülse mittels zweiarmigem Roboter [59]. Auch das überlagerte Ausführen von Prozessbewegungen durch einen der Roboter mit einer gemeinsamen Grundbewegung durch alle Roboter [64, 103] ist eine Anwendung mit synchroner Koordination.

3.2.2 Kommunikationsstrukturen

In Robotersystemen mit kooperierenden Industrierobotern kommen zwei unterschiedliche Kommunikationsstrukturen zum Einsatz [94]. Entweder besitzt jeder Roboter eine eigene Steuerung und die Steuerungen kommunizieren miteinander oder es existiert eine *zentrale Steuerung*, die mit allen Robotern verbunden ist (siehe Bild 20). Bei einer *dezentralen Steuerung* der Roboter müssen die beteiligten Steuerungen direkt oder gegebenenfalls über ein Kopplungselement (z. B. *Switch*) miteinander verbunden werden. Damit die Kommunikation zwischen den Robotern koordiniert erfolgt, kommt eine hierarchische Verwaltung des Zugriffs nach dem Master/Slave-Prinzip zum Einsatz. Bei einer *zentralen Steuerung* übernimmt eine übergeordnete Steuerung die Koordination aller Roboter. Dabei kann aber auch jeder Roboter untergeordnet eine eigene Steuerung besitzen [130].

3.3 Programmierung von Industrierobotern

Robotersysteme sind als frei programmierbar definiert (siehe DIN EN ISO 10218-1 [32]) und können daher Roboterprogramme ausführen, die aus einer Reihe von Roboterbefehlen bestehen. Die Programmierung von Industrierobotern kann mittels zahlreicher unterschiedlicher Programmierverfahren erfolgen, welche sich nach diversen Kriterien gliedern lassen.



(a) Dezentrale Steuerung (b) Zentrale Steuerung

Bild 20: Kommunikationsstrukturen von kooperierenden Industrierobotern

Im Rahmen dieses Abschnittes wird ein Überblick über den aktuellen Stand der Programmierung von Industrierobotern gegeben. Dabei wird zunächst auf die gängigen Programmierverfahren eingegangen (siehe Abschnitt 3.3.1). Anschließend erfolgt eine Betrachtung der Programmierung von Industrierobotern mit Bezug auf die Ausführungsvarianten (siehe Abschnitt 3.3.2). Weitergehend wird auf die spezifische Programmierung von kooperierenden Industrierobotern eingegangen (siehe Abschnitt 3.3.3).

3.3.1 Programmierverfahren

Für die Programmierung von Industrierobotern existiert eine Vielzahl an unterschiedlichen Programmierverfahren und klassischerweise wird zwischen *Onlineprogrammierverfahren* und *Offlineprogrammierverfahren* (siehe Bild 21) unterschieden [45, 95, 116, 119, 121]. Zudem gibt es noch *hybride Programmierverfahren*, die Online- und Offlineprogrammierung vereinen. Weitergehend wird genauer auf die einzelnen Verfahren eingegangen.

Onlineprogrammierverfahren

Bei der Onlineprogrammierung erfolgt die Programmerstellung unmittelbar über die reale Robotersteuerung. Dies hat jedoch den Nachteil, dass die Roboter parallel nicht nicht aktiv produzieren können. Zudem ist die Programmierung bei komplexen Bewegungs- oder Programmabläufen meist aufwendig.

Zu den Onlineprogrammierverfahren zählen die *Teach-In-Programmierung*, die *Playback-Programmierung* und die *sensorgestützte Programmierung* (siehe Bild 21). Die ersten beiden Verfahren lassen sich unter dem Begriff Lernprogrammierverfahren zusammenfassen, da hierbei der Programmierer dem Roboter den gesamten Programmverlauf beibringt.

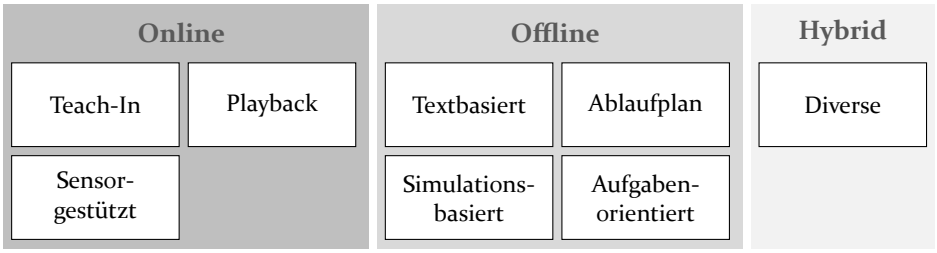


Bild 21: Einteilung der Programmierverfahren von Industrierobotern kumuliert aus [45, 95, 116, 119, 121]

Bei der Teach-In-Programmierung führt der Bediener den Roboter mittels diverser Eingabegeräte zu den gewünschten Posen und diese werden dann dem Programm verknüpft mit einer Bewegungsart hinzugefügt. Werden die kompletten Bewegungen des Roboters aufgezeichnet und in das Programm übernommen, spricht man von einer Playback-Programmierung. Im Gegensatz hierzu werden bei der sensorgestützten Programmierung lediglich grobe Bewegungsvorgaben (z. B. Startpunkt oder Endpunkt) durch den Bediener getätigt, der exakte Bahnverlauf wird durch das Robotersystem mit Hilfe eines Sensors ermittelt.

Offlineprogrammierverfahren

Die Offlineprogrammierung erfolgt basierend auf abstrahierten Modellen bzw. Vorstellungen der realen Arbeitsumgebung. Im Gegensatz zur Onlineprogrammierung sind dabei umfangreiche Vorarbeiten ohne eine Belegung des realen Roboters möglich. Es ist sogar eine Parallelisierung der Programmierung möglich. Jedoch entsprechen die Modelle, auf denen die Programmierung basiert, nie exakt der Realität. Die so resultierenden Ungenauigkeiten können zu unzureichenden Prozessen führen.

Die Offlineprogrammierung lässt sich in vier Verfahren aufgliedern: *textbasierte*, *grafische*, *simulationsbasierte* und *aufgabenorientierte Programmierung* (siehe Bild 21). Das grundlegendste Verfahren ist das explizite textuelle Schreiben des Programms. Bei komplexeren Anwendungen wird diese Programmierung jedoch unübersichtlich und ist mit einer hohen Fehlerrate verbunden. Durch Bild der Prozesse bietet die grafische Programmierung mit Ablaufdiagrammen eine Abhilfe hierfür. Dabei wird die Syntax, aufbauend auf den erstellten Ablaufdiagrammen, automatisch generiert. Noch näher an der realen Anwendung ist die simulationsbasierte Programmierung. Sie erfolgt auf Basis eines virtuellen Modells der Roboterzelle, in dem die Posen definiert werden. Die Schwierigkeit liegt jedoch in der exakten Abbildung der Roboterzelle. Bei der aufgabenorientierten Roboterprogrammierung wird die Syntax ebenfalls automatisch generiert, jedoch auf Grundlage von einfachen Beschreibungen komplexer Zusammenhänge (z. B. „Greife Werkstück“).

Hybride Programmierverfahren

Durch die Kombination von Online- und Offlineprogrammierverfahren werden die Vorteile beider Verfahren beim hybriden Programmierverfahren

vereint. Dabei sind beide Reihenfolgen möglich. Einerseits kann die grundlegende Programmerstellung zunächst offline erfolgen, wobei der reale Roboter nicht blockiert wird. Weitergehend werden die erzeugten Programme online korrigiert und an die reale Arbeitsumgebung angepasst, wodurch eine ausreichende Prozessgenauigkeit erzielt werden kann. Hierfür kommen unter anderem Kamerasysteme zum Einsatz [41, 110]. Andererseits kann vorab online ein präzises Modell der realen Arbeitsumgebung mit entsprechenden Sensoren aufgenommen werden. Anschließend erfolgt offline unmittelbar auf Grundlage dieses Modelles die Programmerzeugung, sodass eine ausreichende Genauigkeit erzielt werden kann. Jedoch muss hierbei auf eine sinnvolle zeitliche Abstimmung geachtet werden.

3.3.2 Programmierausführung

Eine weitere Möglichkeit die Programmierung von Robotern zu differenzieren ist die Unterscheidung gemäß der Ausführung der Programmierung [12]. So kann die Programmierung entweder *manuell* durch Programmierer oder *automatisch* durch Softwarealgorithmen ausgeführt werden. Bei der manuellen Programmierung von komplexeren Anwendungen wird der Programmierer häufig bei der Erstellung durch Software unterstützt, jedoch muss der Bediener bei der softwaregestützten Programmierung die Posen und Parameter weiterhin selbstständig definieren, lediglich das Erzeugen des expliziten Programmes wird ihm abgenommen. Die automatisierte Programmierung geht noch einen Schritt weiter und übernimmt sowohl die Erzeugung der Syntax als auch die Definition der exakten Roboterbahnen. Bei einer manuellen Ausführung der Programmierung steigt der Programmieraufwand schnell mit zunehmender Komplexität (siehe Bild 22), da der Mensch an seine Grenzen kommt. Der Aufwand bei der automatischen Ausführung hingegen ist nahezu unabhängig von der Komplexität der Anwendung. Nachteilig hierbei ist jedoch der vorhergehende erhöhte Aufwand für die Erstellung der Programmiersoftware, daher ist der Einsatz der automatischen Programmierung erst ab einer gewissen Komplexität sinnvoll. Weitergehend werden die beiden Arten der Programmierausführung genauer betrachtet.

Manuelle Ausführung

Bei der manuellen Programmierung von Robotern werden die Programme inklusive enthaltener Parameter manuell durch den Benutzer erstellt. Dabei lässt sich wiederum zwischen *expliziter* und *impliziter* Programmierung unterscheiden. Bei der expliziten Programmierung wird textuell in einer

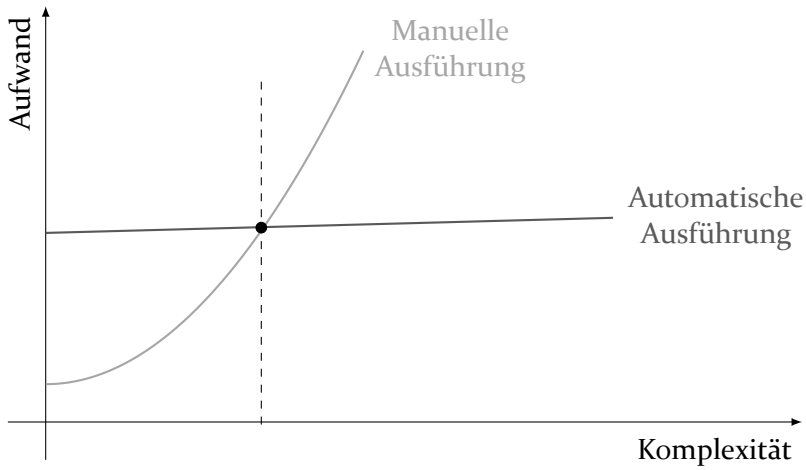


Bild 22: Aufwand für die Programmierung von Industrierobotern in Abhängigkeit der Komplexität der realisierten Anwendung

spezifischen Programmiersprache programmiert. Da das explizite Erstellen der Roboterprogramme mit zunehmender Komplexität der Anwendungen schnell unverhältnismäßig aufwendig werden kann, haben sich eine Vielzahl an impliziten Programmierverfahren etabliert, die spezifische Programmiersprachen abstrahieren, um die Programmierung zu vereinfachen.

Bei der *expliziten* Programmierung kommen meist roboterspezifische Programmiersprachen zum Einsatz, wobei zahlreiche herstellerspezifische Programmiersprachen (z. B. KAREL, KRL, RAPID oder VAL3) aber auch herstellerunabhängige Programmiersprachen (z. B. IRL) existieren [45]. Herstellerunabhängige Programmiersprachen konnten sich bis dato jedoch nicht etablieren, wie der gescheiterte Versuch einer Normung der Programmiersprache verdeutlicht. Eine Alternative zu roboterspezifischen Programmiersprachen sind generische Programmiersprachen, die um Roboterbefehle erweitert sind. Beispiele hierfür sind unter anderem in den Programmiersprachen C++ [48, 67, 113], C# [91] und XML [55] zu finden.

Die *implizite* Programmierung kann mittels verhaltensbasierten Programmiersprachen, bei denen die detaillierte Programmierung anhand von Verhaltensanweisungen abstrahiert ist, ebenfalls textuell erfolgen [20, 29, 49, 87]. Das bedeutet, es wird lediglich programmiert, was auszuführen ist und nicht wie etwas auszuführen ist, wie zum Beispiel durch die Anweisung „Kontur verfolgen". Anschließend werden die abstrahierten Befehle in einen roboterspezifischen Programmcode konvertiert. Eine weitere Möglichkeit

der Abstraktion ist die grafische Programmierung, wie beispielsweise durch Icons [13, 28] oder durch grafische Elemente in einem Hypergraphen [18]. Hierbei erstellt der Benutzer ein Programm durch Verknüpfen von grafischen Elementen und anschließend wird daraus ein roboterspezifischer Programmcode erzeugt. Eine weit verbreitete Methode der manuellen Programmierung ist die Teach-In-Programmierung, bei der einzelne Posen durch manuelles Anfahren festlegt werden. Diese Methode wird von den meisten Roboteranbietern in deren Robotersteuerung integriert zur Verfügung gestellt. Es gibt jedoch auch erweiterte Teach-In-Konzepte, wie beispielsweise ein webbasiertes Teach-In via Teleoperation [33]. Die Playback-Programmierung geht noch eine Stufe weiter, indem Aufgaben oder Teilaufgaben vollständig manuell vorgeführt, dabei aufgezeichnet und anschließend in Roboterprogramme konvertiert werden [78, 82, 83, 109, 127]. Die simulationsbasierte Programmierung ist ebenfalls weit verbreitet. So bieten die meisten Roboterhersteller eigene Simulationssysteme an [1, 37, 62], die einen klaren Fokus auf die eigenen Robotersysteme haben. Es ist aber auch herstellerunabhängige Simulationssoftware für Industrieroboter auf dem Markt vertreten [10, 35, 91], die teilweise proprietär und teilweise nicht proprietär ist. Die Auswahl an Robotersystemen und der enthaltene Funktionsumfang variieren jedoch teilweise stark, weshalb bis dato keine Simulationssoftware den Markt dominiert.

Automatische Ausführung

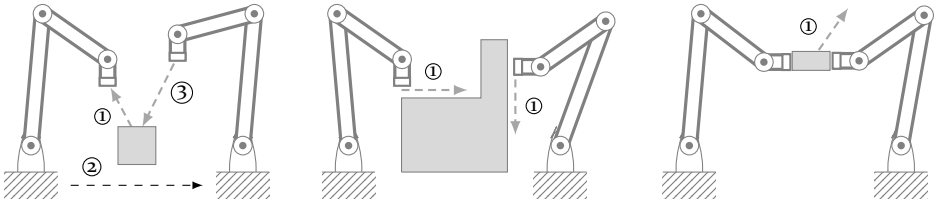
Im Rahmen der gegenwärtigen Entwicklung des Marktes hin zu individuelleren und komplexeren Produkten und dem damit verbundenen Anstieg des Anspruches bei der Roboterprogrammierung, kommt die manuelle und softwaregestützte Programmierung von Robotern teilweise an ihre Grenzen. Abhilfe hierfür bieten die semi-automatische und automatische Programmierung, bei denen ein Teil bzw. die gesamte Programmierung von Softwarealgorithmen übernommen wird. Die Idee der automatischen Roboterprogrammierung existiert bereits seit mehreren Jahrzehnten [123], bei industriellen Anwendungen kommt sie jedoch bis dato verhältnismäßig selten zum Einsatz. Ein Beispiel für eine semi-automatische Programmierung ist die in [70] präsentierte Herstellung von Bauteilen aus *Carbon Fiber Reinforced Polymer* (CFRP). Hierbei wird der Großteil der Roboterprogramme durch Algorithmen erzeugt. Die besonders anspruchsvollen Prozessschritte – das Aufnehmen und Ablegen der Kohlefaser-Schnittstücke – werden jedoch basierend auf Experteneingaben softwaregestützt programmiert. Ein robotergestütztes Frässystem [118] ist eine exemplarische Anwendung, bei

der eine automatische Programmierung des Roboters eingesetzt wird, um schichtweise Bahnverläufe zu erzeugen. Die meisten der Ansätze für eine automatische Programmierung basieren auf *Computer-Aided Design* (CAD). So werden beispielsweise in [124] die Bahnverläufe für ein roboterbasiertes Lackieren basierend auf CAD-Daten durch eine Software erzeugt. In [54] wird das Aufbringen von Klebstoff auf Schuhsohlen und -oberteile mittels CAD-basierter automatisierter Programmierung realisiert. Wenn die Genauigkeiten der CAD-Daten nicht ausreichen, um akzeptable Prozesse zu erzielen, kann die automatische Programmierung mit einer Erfassung des Werkstückes oder einer Online-Überwachung kombiniert werden. So kann unter anderem das roboterbasierte Entgraten von Werkstücken, deren Grat individuell ausfällt, ermöglicht werden [90]. Auch das roboterbasierte Lackieren mit einer vorgeschaltenen Werkstückerfassung ist ein Beispiel hierfür [88, 115].

Auch wenn bei der semi-automatischen und automatischen Programmierung die Bahn- und Programmerzeugung durch Algorithmen erfolgt, werden oftmals dennoch Benutzereingaben benötigt, um beispielsweise Parameter anpassen zu können. Hierfür ist ein *Human Machine Interface* (HMI) erforderlich, das ein wesentlicher Bestandteil des Programmiersystems ist [119], da es die Schnittstelle zwischen Anwender und Roboter darstellt und somit ausschlaggebend für die Benutzerfreundlichkeit des Robotersystems ist.

3.3.3 Programmierung von kooperierenden Industrierobotern

Die meisten der Roboteranbieter bieten Grundfunktionen für die Koordination von mehreren Robotern für ihre Robotersysteme an [1, 63]. Hierzu zählen der *Signalaustausch*, die *zeitliche Synchronisation* und das Ausführen von identischen Bewegungen, was auch als *geometrische Kopplung* bezeichnet wird (siehe Bild 23). Der Austausch von Signalen kann verwendet werden, um die Roboter asynchron zu koordinieren und somit beispielsweise die Übergabe von Werkstücken zu realisieren. Die zeitliche Synchronisierung ermöglicht das zeitgleiche Bearbeiten von Werkstücken. Mittels geometrischer Kopplung kann beispielsweise das gemeinsame Handhaben von Lasten realisiert werden. Auch das Ausführen von überlagerten Bewegungen ist bei einer Reihe der angebotenen Robotersysteme möglich. Systeme, die komplexere kooperative Funktionen mit synchroner Koordination beinhalten, werden bisher jedoch noch nicht von den Roboterherstellern angeboten.



(a) Signalaustausch (b) Zeitliche Synchronisation (c) Geometrische Kopplung

Bild 23: Marktübliche Grundfunktionen für kooperierende Industrieroboter

Im Rahmen der gegenwärtigen Entwicklung des Marktes hin zu komplexeren und kurzlebigeren Anwendungen ist es notwendig, die Programmierung von kooperativen Anwendungen weiterzuentwickeln. Hier setzen eine Reihe von Forschungsarbeiten an, in denen die Programmierung von kooperierenden Industrierobotern vereinfacht werden soll. So wird beispielsweise in [11] eine aufgabenorientierte Programmierung für MRS entwickelt. Es existieren auch Ansätze für die Abstraktion in einer eigenen Programmiersprache für MRS, wie die auf C++ basierende *Dual-Arm Robot Programming Language* (DA-RL) [59]. Für die Umsetzung von anspruchsvollen kooperativen Roboterprozessen wird oftmals eine automatische Programmierung eingesetzt. Ein Beispiel hierfür ist das gleichzeitige Handhaben von CFRP mit mehreren Industrierobotern, welches in [6] durch eine automatische Programmierung umgesetzt ist. Auch die in [94] präsentierte Montageanwendung mittels kooperierenden Industrierobotern wird durch eine CAD-gestützte automatisierte Programmierung vereinfacht. Ein weiteres Beispiel für die Vereinfachung der Programmierung von kooperativen Roboterprozessen mittels automatisierter Programmierung ist die inkrementelle Blechumformung durch zwei Roboterarme [19, 60]. Hierbei werden die Programme ebenfalls auf Grundlage von CAD-Daten generiert. Diese Anwendung, mit einer verhältnismäßig komplexen Art der Kooperation, wird erst durch die Verwendung einer automatisierten Programmierung möglich.

3.4 Handlungsbedarf

Wie im Rahmen dieses Kapitels aufgezeigt, wird bislang bei den meisten Roboteranlagen, in denen mehrere Industrieroboter eingesetzt werden, das Zusammenspiel der Abläufe im Voraus bei der Programmierung berücksichtigt. Eine koordinierte Kooperation findet in diesen Fällen nicht statt. Kooperierende Industrieroboter kommen in der Industrie gegenwärtig verhältnismäßig selten zum Einsatz. Vor allem Anlagen mit einer

komplexeren Koordination der beteiligten Roboter sind selten, da der Aufwand für die Programmierung dieser oftmals unverhältnismäßig groß ist. Doch eine komplexere Koordination der beteiligten Roboter bietet Potenzial für die Optimierung und Erschließung von neuen Prozessen. Zudem kann durch das Überlagern der Arbeitsräume der Platzbedarf der Roboter in der Produktionshalle reduziert werden, was einen Kostenvorteil mit sich bringen kann. Daher ist es sinnvoll nach neuartigen Möglichkeiten der Zusammenarbeit von mehreren Industrierobotern zu suchen und dabei stets eine möglichst einfache Programmierung der Kooperation zu fokussieren.

An dieser Stelle setzt das im Rahmen dieser Arbeit angestrebte Konzept für die kooperative Bearbeitung von Werkstücken, durch eine beidseitige Bewegung von Werkzeug und Werkstück, an. Die Aufteilung von Bearbeitungen auf mehrere Kinematiken erfolgt bis dato entweder asynchron oder gebunden an einzelne räumliche Freiheitsgrade. Eine noch flexiblere Aufteilung kann durch eine beidseitige Verwendung von Kinematiken, die alle räumlichen Freiheitsgrade abdecken, erzielt werden. Deshalb liegt der Fokus auf kooperierenden Industrierobotern. Die gleichzeitige Handhabung von Werkstück und Werkzeug durch Industrieroboter bietet dabei zwei potenzielle Vorteile:

- Prozesszeitreduzierung durch kooperative Bearbeitung
- Erschließung von Anwendungen durch verbesserte Erreichbarkeit

Daher ist das Ziel dieser Arbeit:

Die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen die von mehreren Robotern ausgeführt werden sowie die automatische Bahnplanung, die für eine derartige Aufteilung erforderlich ist.

Dabei soll eine einfache Programmierung sowie Implementierung mit kommerziellen Robotersystemen fokussiert werden, um ein leicht zu adaptierendes Konzept zu erhalten. Der Schwerpunkt liegt auf Anwendungen, die üblicherweise von Industrierobotern durchgeführt werden, wie beispielsweise Klebstoffapplikationen, Montageanwendungen oder Qualitätskontrollen. Zudem entwickelt sich der Markt immer mehr hin zur kundenindividuellen Massenproduktion [42]. Dies geht bis hin zur Einzelstückfertigung, die bedingt durch neue Trends bei den Fertigungstechnologien, wie zum

Beispiel den 3D-Druck, ebenfalls immer häufiger im Fokus stehen. Damit individuelle Werkstücke mit geringem Aufwand erfasst und anschließend ggf. bearbeitet werden können, wird im Rahmen dieser Arbeit auch ein Konzept für die Werkstückerfassung mittels kooperierenden Industrierobotern erarbeitet. Auch für die Erfassung von Werkstücken stellt die beidseitige roboterbasierte Handhabung gemäß dem in dieser Arbeit angestrebten Konzept der Aufteilung von Bearbeitungen eine Neuerung dar. Durch gleichzeitiges Handhaben von Sensor und Messobjekt, ergibt sich eine erhöhte Erreichbarkeit und Flexibilität sowie eine schnellere Positionierung für statische Messungen.

Für die Erlangung der Zielvorgaben wird das folgende Vorgehen gewählt:

- Konzept für die Aufteilung von Bearbeitungsbahnen entwickeln
- Konzept für die Untersuchung der allgemeinen Einflüsse der Aufteilung entwickeln
- Konzept für die Erfassung von Werkstücken mittels kooperierenden Industrierobotern entwickeln
- Implementierung der Konzepte mit unterschiedlichen Robotersystemen
- Allgemeine und anwendungsbezogene Evaluation der implementierten Konzepte

Die einzelnen Schritte sind in den nächsten Kapiteln beschrieben.

4 Aufteilung von Prozessbewegungen auf mehrere Industrieroboter

Im Rahmen dieses Kapitels wird das entwickelte Konzept für die Aufteilung von Prozessbewegungen auf mehrere Industrieroboter beschrieben. Das übergeordnete systemunabhängige Vorgehen besteht dabei aus fünf Schritten (siehe Bild 24). Zunächst muss der Benutzer eine Bearbeitungsbahn definieren. Anschließend kann der Benutzer Einschränkungen festlegen, um bei dem Prozess gewisse Randbedingungen einhalten zu können. Hierunter fallen zum Beispiel kartesische und achsspezifische

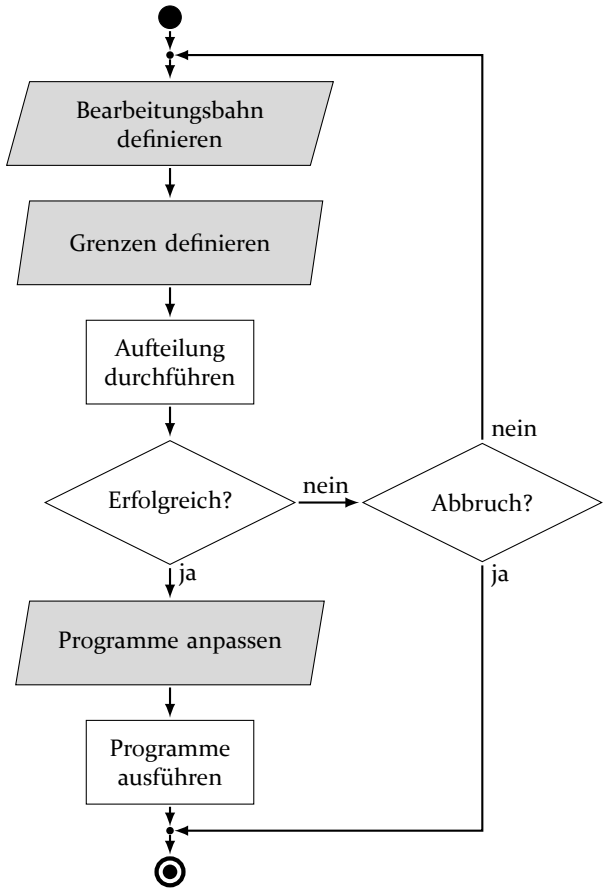


Bild 24: Allgemeines Vorgehen für die Aufteilung von Bearbeitungsprozessen auf mehrere Industrieroboter

Begrenzungen der beteiligten Roboter. Wenn alle Eingangsgrößen definiert sind, kann die Aufteilung gemäß des weitergehend erläuterten Ansatzes erfolgen. Sollte die Aufteilung zu keiner Lösung führen, kann versucht werden durch eine Anpassung der Eingangsgrößen zu einer Lösung zu gelangen. Andernfalls ist der Prozess nicht mittels der betrachteten Aufteilung möglich. Bei Erfolg können die erzeugten Programme noch um individuelle Befehle erweitert bzw. angepasst werden. Im Anschluss sollten die Programme nochmals überprüft werden, bevor sie abschließend am realen Robotersystem zum Einsatz kommen.

Nachfolgend wird zunächst die Grundidee der Aufteilung ausführlich erläutert (siehe Abschnitt 4.1). Zudem werden Möglichkeiten zur gezielten Begrenzung der Aufteilung aufgezeigt (siehe Abschnitt 4.2). Abschließend erfolgt die Erarbeitung von Ansätzen für die Untersuchung von allgemeinen Einflüssen der Aufteilung auf die Kriterien Bearbeitungszeit, Erreichbarkeit und Positioniergenauigkeit (siehe Abschnitt 4.3).

4.1 Grundkonzept der Aufteilung in eine synchrone Bewegung von Werkzeug und Werkstück

Die Aufteilung von Bearbeitungsbahnen in eine synchrone Bewegung von Werkzeug und Werkstück erfolgt anhand eines einfach strukturierten Ablaufes (siehe Bild 25). Dieser besteht aus drei zentralen Schritten: *Startposeniteration*, *Bahnaufteilung* und *Selektion*, die im Rahmen dieses Abschnittes beschrieben werden. Im ersten Schritt wird eine endliche Anzahl an gleichmäßig verteilten Startposen innerhalb des gemeinsamen Arbeitsraumes ermittelt (siehe Abschnitt 4.1.1). Eingangsgrößen, die vom Benutzer gewählt werden können, sind dabei die Diskretisierung und die kartesische Begrenzung des Startbereiches. Hierauf aufbauend erfolgt für jede der Startposen eine Aufteilung der gesamten Bearbeitungsbahn gemäß des entwickelten Ansatzes (siehe Abschnitt 4.1.2). Hierbei kann die Skalierung der Bearbeitung zwischen Werkzeug- und Werkstückseite sowie die kartesische Begrenzung vom Benutzer angepasst werden. Abschließend wird aus der Vielzahl an Lösungen die für das verwendete Robotersystem effizienteste Lösung ausgewählt (siehe Abschnitt 4.1.3), wobei die Roboter durch den Benutzer achsspezifisch begrenzt werden können. Des Weiteren wird im Rahmen dieses Abschnittes die Skalierbarkeit des Ansatzes im Bezug auf die Anzahl der aufzuteilenden Bahnabschnitte betrachtet (siehe Abschnitt 4.1.4).

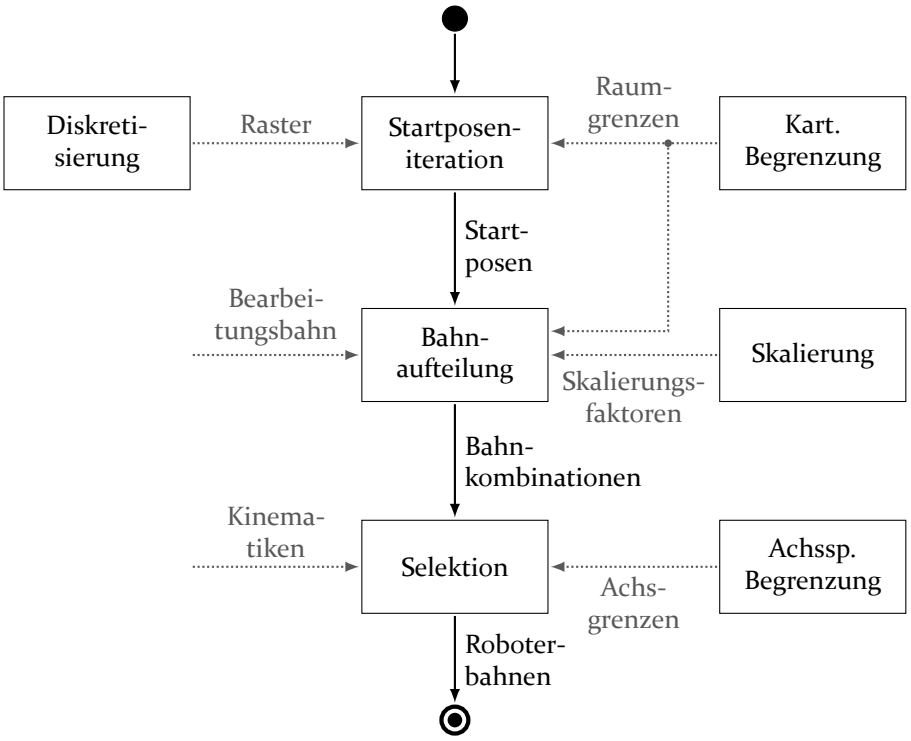


Bild 25: Methode für die Aufteilung von Bearbeitungsbahnen in eine synchrone Bewegung von Werkzeug und Werkstück

4.1.1 Startposeniteration

Ein wichtiger Schritt bei der Aufteilung der Bearbeitung auf mehrere Industrieroboter ist die Bestimmung einer effizienten Ausgangspose des Werkstückes $\{W_1\}$. Theoretisch gibt es hierfür innerhalb des gemeinsamen Arbeitsraumes unendlich viele Lösungen. Damit der Lösungsraum eingeschränkt und dennoch eine effiziente Startpose ermittelt werden kann, wird auf die Brute-Force-Methode zurückgegriffen. Diese berechnet iterativ in einem gleichmäßigen Raster durch den gemeinsamen Arbeitsraum N_W Lösungen. Die Bestimmung der potenziellen Startposen erfolgt anhand eines Algorithmus mit vier verschachtelten Schleifen, die durch die räumlichen Freiheitsgrade iterieren (siehe Bild 26). Eingangsgrößen sind dabei die kartesischen Grenzwerte ($x_{\mathrm{W}_\mathrm{min}}$, $x_{\mathrm{W}_\mathrm{max}}$, ... , $\gamma_{\mathrm{W}_\mathrm{min}}$, $\gamma_{\mathrm{W}_\mathrm{max}}$) sowie eine translatorische Rastergröße l_R und eine Anzahl an Orientierungen N_O. In jedem Durchlauf der Iteration wird eine Pose mit den sechs Parametern (x_W, y_W, z_W, α_W, β_W, γ_W) der Liste von Startposen $\mathcal{P}_{\mathrm{W}_1}$ hinzugefügt.

```
1: procedure StartPoseIteration(x_Wmin, x_Wmax, ..., γ_Wmin, γ_Wmax, l_R, N_O)
2:     P_O ← GetSphereOrientations(α_Wmin, α_Wmax, β_Wmin, β_Wmax, γ_Wmin, γ_Wmax, N_O)
3:     for x_W ← x_Wmin, x_Wmax by l_R do
4:         for y_W ← y_Wmin, y_Wmax by l_R do
5:             for z_W ← z_Wmin, z_Wmax by l_R do
6:                 for all p_O (α_W, β_W, γ_W) ∈ P_O do
7:                     P_W1 ← Pose(x_W, y_W, z_W, α_W, β_W, γ_W)
8:                 end for
9:             end for
10:        end for
11:    end for
12:    return P_W1
13: end procedure
```

Bild 26: Algorithmus zur Startposeniteration

Vorab kann es hilfreich sein, eine geeignete Diskretisierung der Startposeniteration zu bestimmen, um große Berechnungszeiten sowie ungünstige Startposen, die mit großen Bearbeitungszeiten verknüpft sind, zu vermeiden. Daher wird weitergehend ein Ansatz erarbeitet, der den Einfluss der Diskretisierung auf die Berechnungszeit t_B und auf die bei der Ausführung resultierende Summe der maximalen Achswinkeländerungen Θ, die direkt proportional zur Bearbeitungszeit ist, betrachtet und die Auswahl einer optimierten Diskretisierung ermöglicht. Die Diskretisierung setzt sich dabei aus den beiden Komponenten Translation und Rotation zusammen. Aus diesem Grund wird bei diesem Ansatz einerseits die Rastergröße l_R und andererseits die Anzahl an Orientierungen N_O stufenweise variiert. Letzteres lässt sich gemäß dem Ansatz aus [96] gleichmäßig im Raum verteilen (vergl. Abschnitt 4.3.2).

Für jede Iteration wird für den betrachteten Prozess die Aufteilung berechnet. Dabei wird einerseits die Berechnungszeit t_B erfasst. Andererseits wird die Summe der maximalen Achswinkeländerungen Θ aus den jeweiligen maximalen Achswinkeländerungen $\Delta\theta$ der N_A Roboterachsen bestimmt durch:

$$\Theta = \sum_{i=1}^{N_\mathcal{P}-1} \max(\Delta\theta_j)_i \,, \quad j \in \{1,\ 2,\ \ldots,\ N_A\} \tag{4.1}$$

Um einen geeigneten Kompromiss zwischen akzeptabler Berechnungszeit und Minimierung der Summe der maximalen Achswinkeländerungen bestimmen zu können, wird eine Güteziffer G, die beide Faktoren gleichermaßen berücksichtigt, definiert:

$$\begin{aligned} G &= 1 - 0,5 \cdot \frac{p_\mathrm{T} + p_\Theta}{100\,\%} = \\ &= 1 - 0,5 \cdot \left(\frac{t_\mathrm{B} - t_{\mathrm{B}_\mathrm{min}}}{t_{\mathrm{B}_\mathrm{max}} - t_{\mathrm{B}_\mathrm{min}}} + \frac{\Theta - \Theta_\mathrm{min}}{\Theta_\mathrm{max} - \Theta_\mathrm{min}} \right) \in \mathbb{R} \mid 0 \leq G \leq 1 \end{aligned} \tag{4.2}$$

Diese ergibt sich aus dem prozentualen Anteil der Berechnungszeit p_T und dem prozentualen Anteil der Summe der maximalen Achswinkeländerungen p_Θ, die über die jeweiligen Minimal- und Maximalwerte definiert sind. Somit kann anhand der maximalen Güteziffer, für die in Abhängigkeit der Rastergröße und für die in Abhängigkeit der Anzahl der Orientierungen ermittelten Werte, ein geeigneter Wert ausgewählt werden.

4.1.2 Bahnaufteilung

Die Bahnaufteilung erfolgt weitergehend für jede der zuvor ermittelten Startposen $\mathcal{P}_{\mathrm{W}_1}$ (siehe Bild 27). Dabei werden zunächst die jeweiligen Startposen der beteiligten Roboter ermittelt und auf Einhaltung der kartesischen Begrenzung überprüft. Wenn die Prüfung positiv ausfällt, erfolgt die Aufteilung des gesamten Bearbeitungsbahnverlaufes einschließlich einer Überprüfung der kartesischen Begrenzung. Sobald eine Begrenzung überschritten wird, erfolgt ein Abbruch. Andernfalls werden nach dem Durchlauf aller Bahnabschnitte die resultierenden Roboterbahnen in einer Liste hinterlegt. Nachdem alle Startposen durchlaufen sind, erfolgt die Ausgabe der Liste mit allen Lösungsvarianten.

Bei der grundlegenden Idee der Aufteilung einer Prozessbewegung in eine Werkstück- und eine Werkzeugbewegung können jeweils mehrere Roboter das Werkstück bzw. das Werkzeug gemeinsam führen [P6]. Aufgrund der einfacheren Formulierung, ist im Folgenden lediglich die Rede von zwei Robotern: dem *werkstückführenden Roboter* und dem *werkzeugführenden Roboter*. Für die Berechnung der jeweiligen Roboterbahnen sind eine Reihe von Transformationen notwendig (siehe Bild 28). Die beiden Roboter sind mit ihrer Basis $\{B_\mathrm{W}\}$ bzw. $\{B_\mathrm{T}\}$ im Bezug auf das Ursprungskoordinatensystem $\{O\}$ positioniert. Die jeweiligen Transformationen sind ${}^\mathrm{O}\mathbf{T}_{\mathrm{B}_\mathrm{W}}$ und ${}^\mathrm{O}\mathbf{T}_{\mathrm{B}_\mathrm{T}}$.

```
 1: procedure PathDivision(Path, P_W1, Margins, Scalings)
 2:     abort ← false
 3:     for all p_W1 ∈ P_W1 do
 4:         StartMovements ← GetStartMovements(P_W1)
 5:         if CheckMargins(StartMovements, Margins) then
 6:             CooperativePath ← StartMovements
 7:             for all Movement ∈ Path do
 8:                 CooperativeMovements ← PathDivision(Movement, Scalings)
 9:                 if CheckMargins(CooperativeMovements, Margins) then
10:                     CooperativePath ← CooperativeMovements
11:                 else
12:                     abort ← true
13:                     break
14:                 end if
15:             end for
16:             if !abort then
17:                 CooperativePaths ← CooperativePath
18:             else
19:                 abort ← false
20:             end if
21:         end if
22:     end for
23:     return CooperativePaths
24: end procedure
```

Bild 27: Algorithmus zur Bahnaufteilung

Das Ursprungskoordinatensystem kann auch in eine Roboterbasis gelegt werden, wodurch eine der Transformationen entfällt. Weitergehend lässt sich für beide Roboter die Pose des Flansches $\{F_\mathrm{W}\}$ bzw. $\{F_\mathrm{T}\}$ anhand einer Transformation von der Basis zum Flanschkoordinatensystem mit ${}^{\mathrm{B_W}}\mathbf{T}_{\mathrm{F_W}}$ bzw. ${}^{\mathrm{B_T}}\mathbf{T}_{\mathrm{F_T}}$ beschreiben. Das Werkstück besitzt ein eigenes Koordinatensystem $\{W\}$, dessen Lage anhand der Transformation ${}^{\mathrm{F_W}}\mathbf{T}_{\mathrm{W}}$ gegenüber dem Flansch definiert ist. Gleiches gilt für das Werkzeug, dessen Koordinatensystem $\{T\}$ durch die Transformation ${}^{\mathrm{F_T}}\mathbf{T}_{\mathrm{T}}$ beschrieben ist.

Die Aufteilung der Bewegung wird für jeden der $(N_\mathcal{P} - 1)$ Bahnabschnitte einzeln berechnet. Hierbei wird zunächst eine *Werkzeugbewegung* zu einer Hilfspose $\{H\}$ berechnet (siehe Bild 29a). Anschließend erfolgt die Bestimmung einer passenden *Werkstückbewegung*, sodass die angestrebte

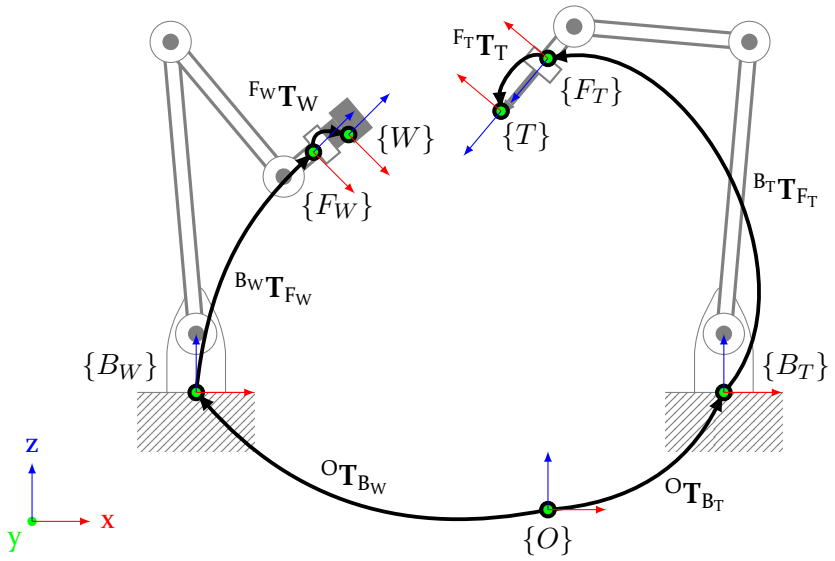
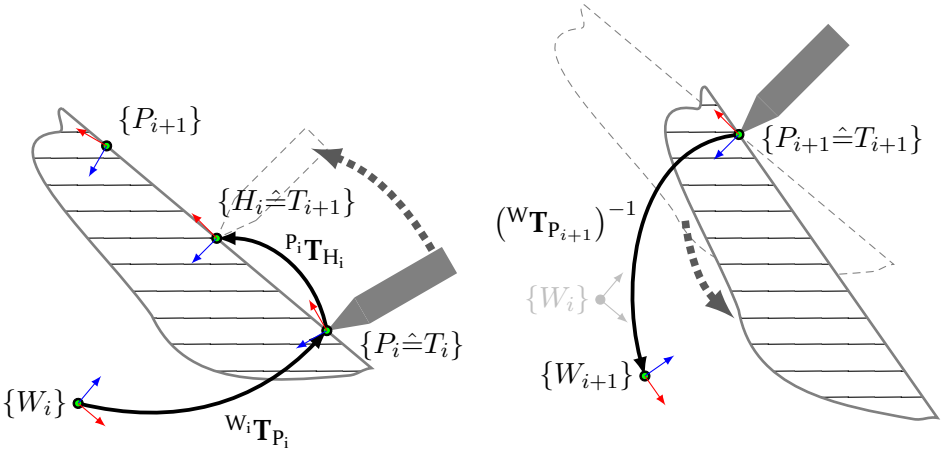


Bild 28: Koordinatensysteme und Transformationen für die kooperative Bearbeitung

Bahnpose $\{P\}$ an dem Werkzeugkoordinatensystem $\{T\}$ positioniert wird (siehe Bild 29b). Die Verteilung zwischen den beiden Robotern kann anhand eines Skalierungsfaktors $s_i \in \mathbb{R} \mid 0 \leq s_i \leq 1$ für alle



(a) Werkzeugbewegung

(b) Werkstückbewegung

Bild 29: Werkstückbezogene Koordinatensysteme und Transformationen für die kooperative Bearbeitung

Bahnabschnitte $i \in \{1,\ 2,\ \ldots,\ N_{\mathcal{P}} - 1\}$ reguliert werden. Dieser gibt an, wie viel Prozent der Bearbeitungsbewegung durch den werkzeugführenden Roboter ausgeführt wird. Folglich wird der Rest der Bearbeitungsbewegung vom werkstückführenden Roboter beigetragen. Nachfolgend werden die jeweiligen Berechnungen beschrieben.

Werkzeugbewegung

Einerseits erfolgt eine *Werkzeugbewegung* durch den werkzeugführenden Roboter zu einer Hilfspose $\{H_i\}$, deren Transformation ${}^{\mathrm{P}}\mathbf{T}_{\mathrm{H}}$ mit Bezug zur der Ausgangspose $\{P_i\}$ berechnet werden kann durch:

$$\begin{aligned} &{}^{\mathrm{P}_i}\mathbf{T}_{\mathrm{H}_i} = s_i \cdot {}^{\mathrm{P}_i}\mathbf{T}_{\mathrm{P}_{i+1}} = \mathbf{T}\left(s_i \cdot {}^{\mathrm{P}_i}\mathbf{f}_{\mathrm{P}_{i+1}}\right) \quad , \quad i \in \{1,\ 2,\ \ldots,\ N_{\mathcal{P}} - 1\} \\ &\text{mit} \quad s_i \in \mathbb{R} \mid 0 \leq s_i \leq 1 \ , \ {}^{\mathrm{P}_i}\mathbf{f}_{\mathrm{P}_{i+1}} = (\Delta x \ \ \Delta y \ \ \Delta z \ \ \Delta\alpha \ \ \Delta\beta \ \ \Delta\gamma)^{\mathrm{T}} \end{aligned} \tag{4.3}$$

Mit Hilfe des Skalierungsfaktors s kann der Anteil der Werkzeugbewegung festgelegt werden. Anschließend lässt sich die Zielpose für den werkzeugführenden Roboter $\{T_{i+1}\}$ beschreiben durch die folgende Transformation:

$$\begin{gathered} {}^{\mathrm{B_T}}\mathbf{T}_{\mathrm{T}_{i+1}} = \left({}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_T}}\right)^{-1} \cdot {}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_W}} \cdot {}^{\mathrm{B_W}}\mathbf{T}_{\mathrm{F_{W_i}}} \cdot {}^{\mathrm{F_W}}\mathbf{T}_{\mathrm{W}} \cdot {}^{\mathrm{W}}\mathbf{T}_{\mathrm{P}_i} \cdot {}^{\mathrm{P}_i}\mathbf{T}_{\mathrm{H}_i} \ , \\ i \in \{1,\ 2,\ \ldots,\ N_{\mathcal{P}} - 1\} \end{gathered} \tag{4.4}$$

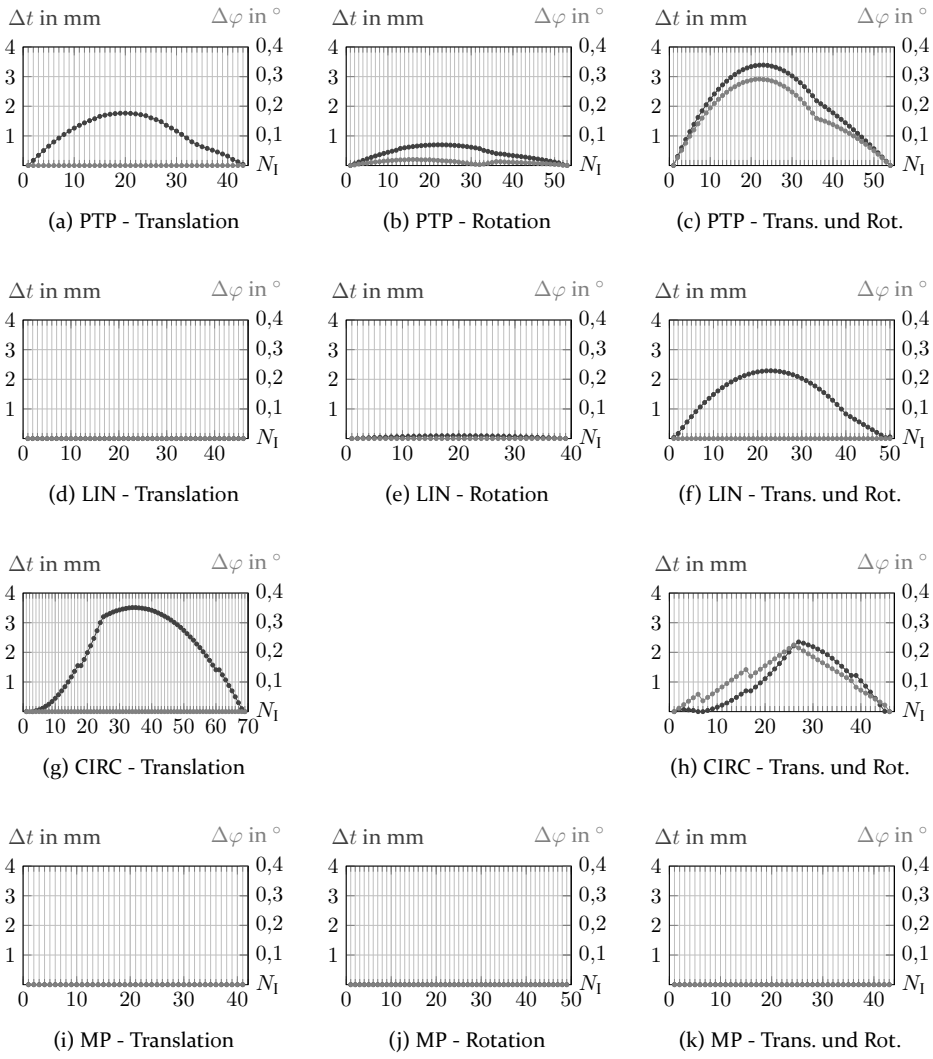
Werkstückbewegung

Andrerseits wird das Werkstück durch den werkstückführenden Roboter mit der Zielpose $\{P_{i+1}\}$ am neuen Werkzeugkoordinatensystem $\{T_{i+1}\}$ positioniert. Die Transformation ${}^{\mathrm{B_W}}\mathbf{T}_{\mathrm{W}}$ für die jeweilige Werkstückpose kann berechnet werden durch:

$$\begin{gathered} {}^{\mathrm{B_W}}\mathbf{T}_{\mathrm{W}_{i+1}} = \left({}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_W}}\right)^{-1} \cdot {}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_T}} \cdot {}^{\mathrm{B_T}}\mathbf{T}_{\mathrm{T}_{i+1}} \cdot \left({}^{\mathrm{W}}\mathbf{T}_{\mathrm{P}_{i+1}}\right)^{-1} \cdot \left({}^{\mathrm{F_W}}\mathbf{T}_{\mathrm{W}}\right)^{-1} \ , \\ i \in \{1,\ 2,\ \ldots,\ N_{\mathcal{P}} - 1\} \end{gathered} \tag{4.5}$$

Kooperative Bewegung

Werden die beiden zuvor beschriebenen Bewegungen synchron ausgeführt, führt dies je nach Bewegungsart (siehe Abschnitt 2.3) zu unterschiedlichen Bearbeitungsbahnen. Der Zusammenhang wird anhand von experimentell durchgeführten Einzelbewegungen veranschaulicht (siehe Bild 30). Der Skalierungsfaktor beträgt dabei stets $s = 0,5$ und es erfolgt jeweils eine getrennte Betrachtung von Translation und Rotation sowie eine Betrachtung



(a) PTP - Translation (b) PTP - Rotation (c) PTP - Trans. und Rot.

(d) LIN - Translation (e) LIN - Rotation (f) LIN - Trans. und Rot.

(g) CIRC - Translation (h) CIRC - Trans. und Rot.

(i) MP - Translation (j) MP - Rotation (k) MP - Trans. und Rot.

Bild 30: Werkstückbezogene Abweichungen bei der kooperativen Bewegung

der Kombination von beiden. In den Diagrammen aufgetragen ist einerseits der Betrag der translatorischen Bahnabweichung Δt in mm, der für jeden der N_I Iterationsschritte durch den euklidischen Abstand berechnet wird:

$$\Delta t = ||\mathbf{t}_{\mathrm{soll}_i} - \mathbf{t}_{\mathrm{ist}_i}|| \,, \quad i \in \{1,\ 2,\ \ldots,\ N_\mathrm{I}\} \quad \text{mit} \quad \mathbf{t} = (x\ \ y\ \ z)^\mathrm{T} \tag{4.6}$$

Andererseits ist der Betrag der rotatorischen Bahnabweichung $\Delta\varphi$ in ° aufgetragen, der nach dem selben Prinzip berechnet wird:

$$\Delta\varphi = ||\mathbf{r}_{\mathrm{soll}_i} - \mathbf{r}_{\mathrm{ist}_i}|| \,, \quad i \in \{1,\ 2,\ \ldots,\ N_\mathrm{I}\} \quad \text{mit} \quad \mathbf{r} = (\alpha\ \ \beta\ \ \gamma)^\mathrm{T} \tag{4.7}$$

Für die PTP-Bewegung ist in jedem Fall (siehe Bild 30a-30c) mit Abweichungen zu rechnen, da die Kombination von zwei freien Roboterbewegungen stets zu einer freien Relativbewegung führt. Bei der Verwendung von Linearbahnen (LIN) ist eine reine Translation (siehe Bild 30d) oder eine reine Rotation (siehe Bild 30e) nicht mit merklichen Abweichungen verbunden. Diese entstehen erst bei einer kombinierten Bewegung (siehe Bild 30f), wobei primär translatorische Abweichungen vorliegen. Bewegungstypen mit einem durch geometrische Teilbahnen definierten Verlauf, wie *Kreisbahnen* (CIRC) oder *Splinebahnen*, sind nicht mit einer reinen Rotation möglich, da sich die Positionen der Bahnabschnitte zwangsläufig unterscheiden müssen. Daher wird für diese Bahntypen ausschließlich die reine Translation (siehe Bild 30g) sowie die Kombination von Translation und Rotation (siehe Bild 30h) betrachtet. Hierbei zeigen sich beachtliche Abweichungen. Lediglich bei einer Interpolation mittels Bewegung vom Typ *Multi-Point* (MP) können die Abweichungen für eine gleichzeitige Translation und Rotation bei diesem Ansatz vermieden werden (siehe Bild 30i-30k).

4.1.3 Selektion

Aus den im vorherigen Abschnitt ermittelten Lösungen muss im letzten Schritt die am besten geeignete Variante ausgewählt werden. Hierfür wird für alle Lösungsvarianten die Gesamtbearbeitungszeit t_P berechnet und letztlich die Variante mit der geringsten Gesamtbearbeitungszeit $t_{\mathrm{P}_{\mathrm{min}}}$ ausgewählt (siehe Bild 31). Für jede Variante wird dabei die gesamte Bearbeitungsbahn Schritt für Schritt durchlaufen und für jeden Abschnitt die Erreichbarkeit durch die verwendete Kinematik mittels Rückwärtstransformation

```
1: procedure PathSelection(CooperativePaths, AxisMargins)
2:     abort ← false
3:     t_Pmin ← max
4:     for all CooperativePath ∈ CooperativePaths do
5:         for all Movements ∈ CooperativePath do
6:             if GetInverseKinematics(Movements, AxisMargins) then
7:                 t_P ← GetProzessTime(Movements)
8:             else
9:                 abort ← true
10:                t_P ← 0
11:                break
12:            end if
13:        end for
14:        if !abort and t_P < t_Pmin then
15:            SelectedPath ← CooperativePath
16:            t_Pmin ← t_P
17:        else
18:            abort ← false
19:        end if
20:    end for
21:    if t_Pmin = max then
22:        ThrowException("Process not possible")
23:    end if
24:    return SelectedPath
25: end procedure
```

Bild 31: Algorithmus zur Lösungsauswahl

(siehe Abschnitt 2.2.3) überprüft. Die Gesamtbearbeitungszeit t_P für eine Variante ergibt sich aus der Summe aller $N_\mathcal{P} - 1$ Einzelbewegungszeiten t_C gemäß:

$$\begin{gathered} t_\mathrm{P} = \sum_{i=1}^{N_\mathcal{P}-1} t_{\mathrm{C}_i} = \sum_{i=1}^{N_\mathcal{P}-1} \max\left(\max\left(f(a)_k \cdot \Delta\theta_k\right)_j\right)_i \,, \\ i \in \{1,\ 2,\ \ldots,\ N_\mathcal{P} - 1\}\,, \quad j \in \{1,\ 2,\ \ldots,\ N_\mathrm{R}\}\,, \\ k \in \{1,\ 2,\ \ldots,\ N_\mathrm{A}\} \end{gathered} \tag{4.8}$$

Die Einzelbewegungszeiten t_C sind wiederum über die zeitkritische Achse der Bewegung definiert. Sie ergeben sich aus der maximalen Bewegungszeit t_R der N_R beteiligten Roboter:

$$t_C = \max(t_{R_j}) \,, \quad j \in \{1,\ 2,\ \ldots,\ N_R\} \tag{4.9}$$

Die Bewegungszeit eines Roboters t_R ergibt sich aus der maximalen Achswinkeländerung $\Delta\theta$ der N_A Achsen des Roboters gemäß:

$$t_R = \max(f(a)_k \cdot \Delta\theta_k) \,, \quad k \in \{1,\ 2,\ \ldots,\ N_A\} \tag{4.10}$$

Gegebenenfalls kann das Beschleunigungsverhalten der einzelnen Achsen durch die Funktion $f(a)$ berücksichtigt werden. Da das Beschleunigungsverhalten der N_A Achsen oftmals nicht bekannt ist, können an dieser Stelle auch proportionale Gewichtungsfaktoren verwendet werden. Die jeweilige Achswinkeländerung $\Delta\theta$ ergibt sich aus dem Betrag der Differenz der Achswinkel:

$$\Delta\theta_i = |\theta_{i+1} - \theta_i| \,, \quad i \in \{1,\ 2,\ \ldots,\ N_\mathcal{P} - 1\} \tag{4.11}$$

Die entsprechenden Achswinkel der Roboter müssen mittels Rückwärtstransformation (siehe Abschnitt 2.2.3) bestimmt werden. Lösungsvarianten, bei denen einer der beteiligten Roboter seine Konfiguration ändert, führen aufgrund von sich somit stark ändernden einzelnen Achswinkeln zu einer verhältnismäßig großen Gesamtzeit und werden somit nicht durch den Algorithmus ausgewählt.

4.1.4 Skalierbarkeit der aufzuteilenden Bahnabschnitte

Die Skalierbarkeit des Ansatzes im Bezug auf die Anzahl der aufzuteilenden Bahnabschnitte wird bestimmt, indem die Berechnungszeiten t_B für eine unterschiedliche Anzahl an Bahnabschnitten $N_B = N_\mathcal{P} - 1$ erfasst werden (siehe Bild 32). Dabei werden die Berechnungszeiten der Startposeniteration, der Aufteilung und der Selektion sowie die Summe dieser getrennt voneinander betrachtet. Die Berechnungszeit der Startposeniteration ist unabhängig von der Anzahl der Bahnabschnitte und hat keinen merklichen Einfluss auf die Gesamtzeit, weshalb diese nicht dargestellt wird. Weitergehend zeigt sich sowohl für die Berechnungszeit der Aufteilung als auch für die Berechnungszeit der Selektion ein näherungsweise lineares Verhalten. Dementsprechend besitzt auch die Gesamtzeit dieses Verhalten.

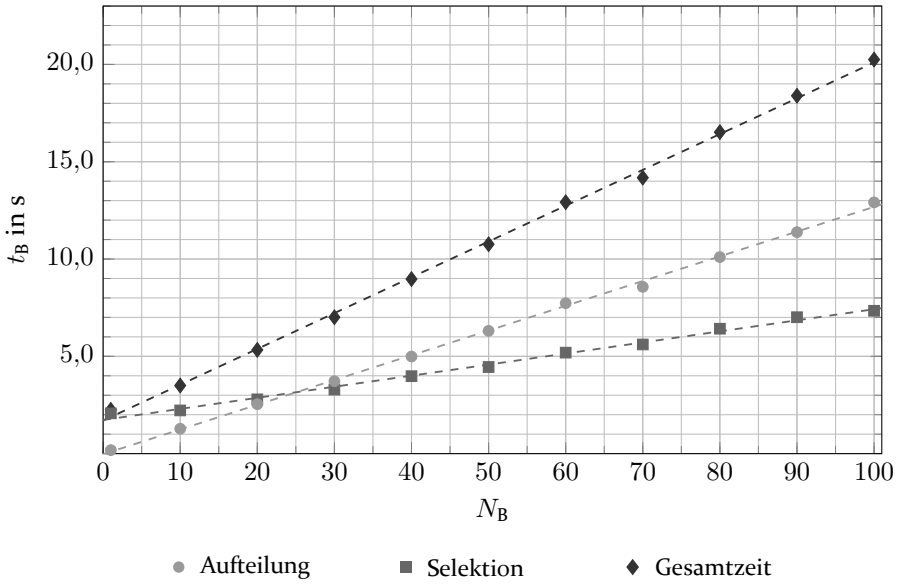


Bild 32: Abhängigkeit der Berechnungszeiten von der Anzahl an Bahnabschnitten

4.2 Begrenzung der Aufteilung

Bedingt durch das verwendete System oder den zu realisierenden Prozess ist es oftmals erforderlich gewisse Einschränkungen vorzunehmen. Daher wird in diesem Abschnitt zunächst ein Ansatz für die Verteilung der werkstückbezogenen Freiheitsgrade zwischen der Werkstück- und der Werkzeugseite erarbeitet (siehe Abschnitt 4.2.1). Dies stellt eine wichtige Erweiterung des zuvor beschriebenen Ansatzes dar, denn es ermöglicht eine Bearbeitung mit räumlicher Vorzugslage. Anschießend erfolgt die Betrachtung einer kartesischen sowie einer achsspezifischen Begrenzung der beteiligten Roboter (siehe Abschnitt 4.2.2). Hiermit können Grenzbereiche der Kinematiken gezielt ausgeschlossen werden.

4.2.1 Verteilung der werkstückbezogenen Freiheitsgrade

Für die Verteilung der werkstückbezogenen Freiheitsgrade muss die Berechnung der Hilfsposition $\{H\}$ (siehe Abschnitt 4.1.2) erweitert werden. So erfolgt die Skalierung getrennt nach Freiheitsgrad, indem jeweils ein eigener Skalierungsfaktor (s_x, s_y, s_z, s_α, s_β, s_γ) verwendet wird. Die Transformation

zur angepassten Hilfsposition ${}^{\mathrm{P}}\mathbf{T}_{\mathrm{H}}$ mit Bezug zur Ausgangspose $\{P_i\}$ berechnet sich für die $N_{\mathcal{P}} - 1$ Bahnposen gemäß:

$$\begin{aligned} {}^{\mathrm{P}_i}\mathbf{T}_{\mathrm{H}_i} &= \mathbf{T}\left(\mathbf{s}_i \cdot {}^{\mathrm{P}_i}\mathbf{f}_{\mathrm{P}_{i+1}}\right) \quad , \quad i \in \{1,\ 2,\ \ldots,\ N_{\mathcal{P}} - 1\} \\ \text{mit} \quad \mathbf{s}_i &= \left(s_{\mathrm{x}}\ \ s_{\mathrm{y}}\ \ s_{\mathrm{z}}\ \ s_\alpha\ \ s_\beta\ \ s_\gamma\right)^{\mathrm{T}} \quad , \\ {}^{\mathrm{P}_i}\mathbf{f}_{\mathrm{P}_{i+1}} &= \left(\Delta x\ \ \Delta y\ \ \Delta z\ \ \Delta\alpha\ \ \Delta\beta\ \ \Delta\gamma\right)^{\mathrm{T}} \end{aligned} \tag{4.12}$$

Somit können über die Skalierungsfaktoren s_α, s_β und s_γ die drei rotatorischen Freiheitsgrade und über die Skalierungsfaktoren s_{x}, s_{y} und s_{z} die drei translatorischen Freiheitsgrade verteilt werden. Dabei gilt für alle Skalierungsfaktoren $s_i \in \mathbb{R} \mid 0 \leq s_i \leq 1$. Wird einer dieser Skalierungsfaktoren auf $s = 0$ gesetzt, so übernimmt der werkstückführende Roboter die gesamte Bewegung dieses Freiheitsgrades. Das Gegenteil, also eine reine Werkzeugbewegung für diesen Freiheitsgrad, ist der Fall wenn der Skalierungsfaktor auf einen Wert von $s = 1$ festgelegt wird.

4.2.2 Kartesische und achsspezifische Begrenzung

Eine kartesische Begrenzung kann für jeden der beteiligten Roboter anhand von sechs kartesischen Grenzwerten ($x_{\mathrm{T_{min}}}$, $x_{\mathrm{T_{max}}}$, ... , $\gamma_{\mathrm{T_{min}}}$, $\gamma_{\mathrm{T_{max}}}$) erfolgen, indem die Zielposen des im TCP liegenden Koordinatensystems nur zulässig sind, wenn diese innerhalb der Grenzen liegen. Die Zielposen werden dabei durch die Transformation ${}^{\mathrm{O}}\mathbf{T}_{\mathrm{T}}$ beschrieben:

$${}^{\mathrm{O}}\mathbf{T}_{\mathrm{T}} = \begin{bmatrix} {}^{\mathrm{O}}\mathbf{R}_{\mathrm{T}} & {}^{\mathrm{O}}\mathbf{t}_{\mathrm{T}} \\ 0_{1\times 3} & 1 \end{bmatrix} \tag{4.13}$$

Hierbei lässt sich einerseits der Translationsvektor ${}^{\mathrm{O}}\mathbf{t}_{\mathrm{T}}$ in den drei translatorischen Richtungen durch jeweils einen Minimal- und einen Maximalwert begrenzen gemäß:

$$\begin{aligned} {}^{\mathrm{O}}\mathbf{t}_{\mathrm{T}} \mid {}^{\mathrm{O}}\mathbf{t}_{\mathrm{T}} \rightarrow\ & x_{\mathrm{T_{min}}} < x_{\mathrm{T}} < x_{\mathrm{T_{max}}} \\ \wedge\ & y_{\mathrm{T_{min}}} < y_{\mathrm{T}} < y_{\mathrm{T_{max}}} \\ \wedge\ & z_{\mathrm{T_{min}}} < z_{\mathrm{T}} < z_{\mathrm{T_{max}}} \end{aligned} \tag{4.14}$$

Andererseits kann die Rotationsmatrix ${}^{\mathrm{O}}\mathbf{R}_{\mathrm{T}}$ angepasst werden, indem die drei rotatorischen Parameter durch jeweils einen Minimal- und einen Maximalwert begrenzt werden:

$$\begin{aligned} {}^{\mathrm{O}}\mathbf{R}_{\mathrm{T}} \mid {}^{\mathrm{O}}\mathbf{R}_{\mathrm{T}} \rightarrow \alpha_{\mathrm{T}_{\mathrm{min}}} < \alpha_{\mathrm{T}} < \alpha_{\mathrm{T}_{\mathrm{max}}} \\ \wedge\ \beta_{\mathrm{T}_{\mathrm{min}}} < \beta_{\mathrm{T}} < \beta_{\mathrm{T}_{\mathrm{max}}} \\ \wedge\ \gamma_{\mathrm{T}_{\mathrm{min}}} < \gamma_{\mathrm{T}} < \gamma_{\mathrm{T}_{\mathrm{max}}} \end{aligned} \tag{4.15}$$

Es folgt also eine Roboterpose mit sechs Parametern (x_{T}, y_{T}, z_{T}, a_{T}, b_{T}, c_{T}), die innerhalb der kartesischen Begrenzung liegt.

Des Weiteren können die beteiligten Roboter auch achsspezifisch begrenzt werden, indem nur Zielposen zulässig sind, bei denen alle N_{A} Achswerte θ_j in der Lösung L der Rückwärtstransformation (siehe Abschnitt 2.2.3) zwischen einem jeweiligen Minimalwert $\theta_{j_{\mathrm{min}}}$ und einem Maximalwerten $\theta_{j_{\mathrm{max}}}$ liegen:

$$L = \{(\theta_j)\} = \{(\theta_1; \theta_2; \ \ldots\ ; \theta_{N_{\mathrm{A}}})\} \ \text{ mit } \ \theta_{j_{\mathrm{min}}} < \theta_j < \theta_{j_{\mathrm{max}}} \tag{4.16}$$

4.3 Allgemeine Einflüsse der Aufteilung

In diesem Abschnitt werden Ansätze für die Untersuchung der allgemeinen Einflüsse der Aufteilung aufgezeigt. Dabei wird zunächst der Einfluss der kooperativen Bearbeitung auf die Bearbeitungszeit (siehe Abschnitt 4.3.1) sowie auf die Erreichbarkeit (siehe Abschnitt 4.3.2) betrachtet. Außerdem wird auf die relative Positioniergenauigkeit der beteiligten Industrieroboter (siehe Abschnitt 4.3.3) eingegangen.

4.3.1 Einfluss der kooperativen Bearbeitung auf die Bearbeitungszeit

Als Basis für die Betrachtung der Bearbeitungszeit erfolgt im Rahmen dieses Abschnittes zunächst die Definition eines Ansatzes für die *Bestimmung der Bearbeitungszeitreduzierung*. Da der Prozessweg beim realisierten Ansatz der Aufteilung halbiert wird, ist der Verdacht naheliegend, dass auch die Bearbeitungszeit halbiert wird. Um den genauen Zusammenhang ermitteln zu können, schließt dem eine detaillierte Analyse der theoretischen

Grenzen der Bearbeitungszeitreduzierung an. Des Weiteren liegt beim zuvor beschriebenen Ansatz der Fokus lediglich auf einer werkstückbezogenen Aufteilung der Bearbeitungsbahnen. Inwieweit die beteiligten Roboter ausgelastet werden, wird dabei nicht betrachtet. Eine gleichmäßige Auslastung kann jedoch zu einer *zeitlichen Optimierung* führen und wird daher im Rahmen dieses Abschnittes ebenfalls erarbeitet.

Bestimmung der Bearbeitungszeitreduzierung

Der resultierende prozentuale Zeiteinsparung p ergibt sich aus dem Verhältnis zwischen der Zeit für die Bearbeitung durch eine kooperative Bewegung (t_C) und der Zeit für Bearbeitung durch eine ausschließliche Werkstück- bzw. Werkzeugbewegung ($t_{W/T}$) gemäß:

$$p = \left(1 - \frac{t_C}{t_{W/T}}\right) \cdot 100\,\% \in \mathbb{R} \mid p < 100\,\% \tag{4.17}$$

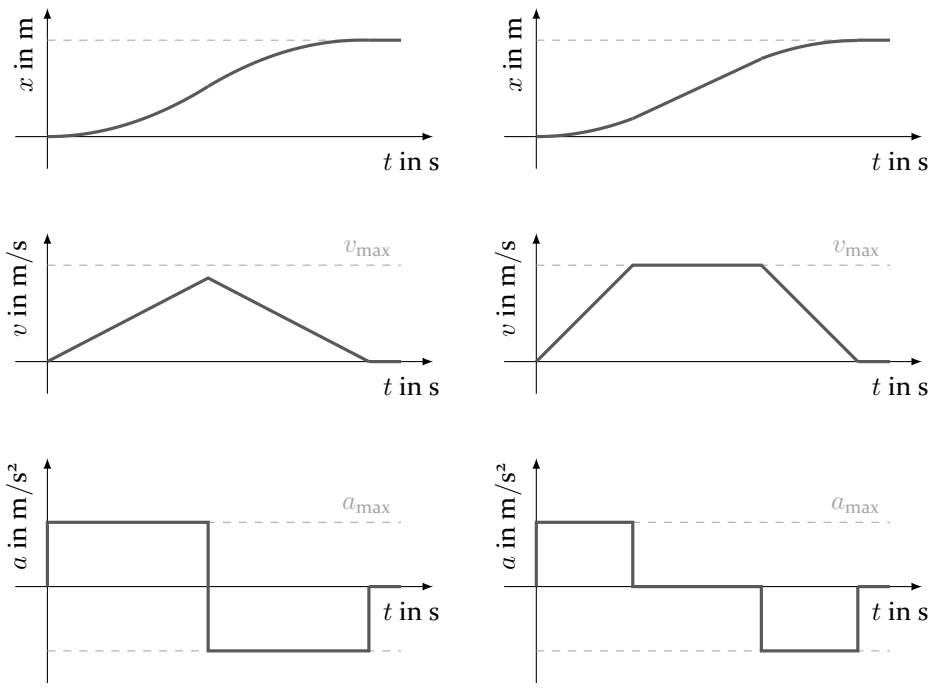
Bei der Zeit für die Bearbeitung durch eine kooperative Bewegung kommt zudem noch die Synchronisationszeit t_S zur von der Strecke x abhängigen reinen Bewegungszeit t_M hinzu:

$$t_C\,(x) = t_S + t_M\,(x) \tag{4.18}$$

Die Synchronisationszeit ist die Zeit, die für eine Synchronisation der Ausführung der beidseitigen Bewegungen erforderlich ist. Wenn diese verhältnismäßig groß und somit die Zeit für eine kooperative Bewegung größer als für eine Werkstück- bzw. Werkzeugbewegung ist, kann die prozentuale Zeiteinsparung auch negativ ausfallen.

Grenzen der Bearbeitungszeitreduzierung

Ein reales Robotersystem kann nicht schlagartig eine vorgegebene Geschwindigkeit erreichen, sondern es liegt stets eine endliche Beschleunigung vor. Daher ist es unmöglich bei der kooperativen Bearbeitung durch die Halbierung der Bearbeitungsstrecke eine Halbierung der Bearbeitungszeit zu erzielen. Auch ein Verfahrprofil ohne Ruckbegrenzung (siehe Bild 33) ist mit derartigen Systemen nur theoretisch möglich. In diesem Fall liegt eine gleichmäßige Beschleunigung vor. Die hierbei theoretisch mögliche



(a) Maximalgeschwindigkeit nicht erreicht

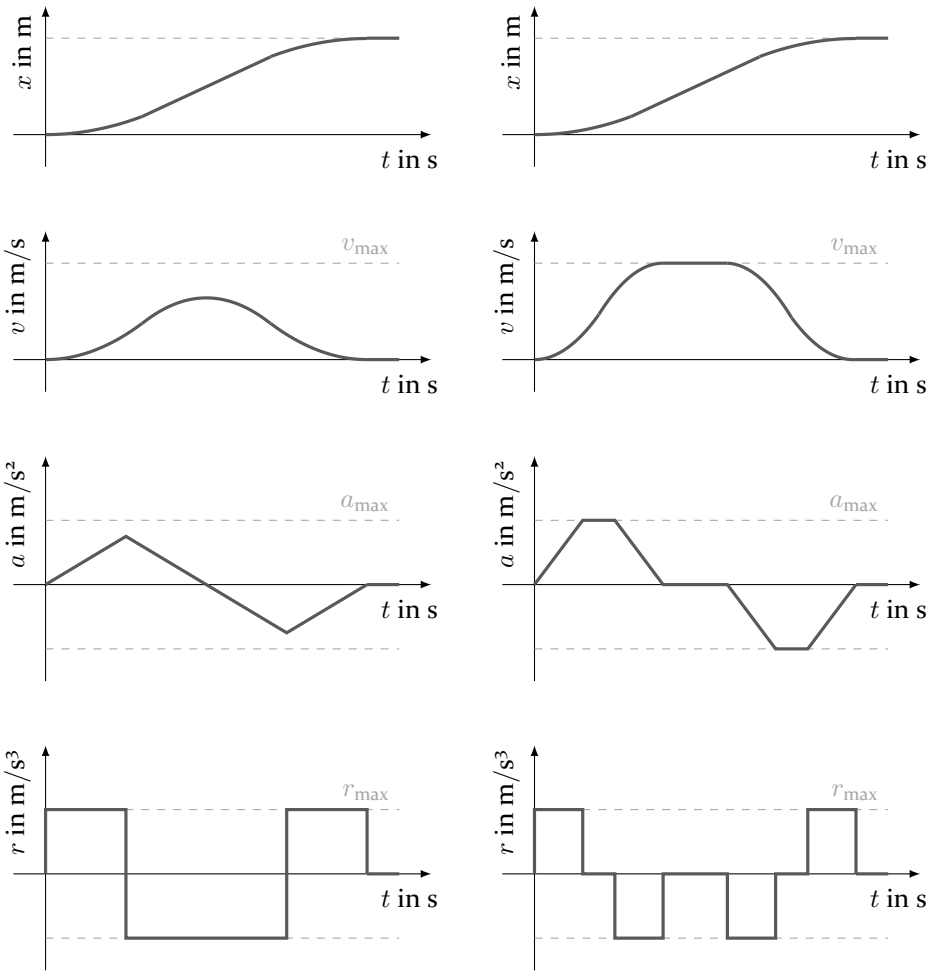
(b) Maximalgeschwindigkeit erreicht

Bild 33: Verfahrprofile ohne Ruckbegrenzung

prozentuale Zeiteinsparung p bei einer Halbierung der Bearbeitungsstrecke x beträgt $29,3\,\%$ und lässt sich bestimmen durch:

$$p_{a=\text{const}} = \left(1 - \frac{\sqrt{2\frac{\frac{x}{2}}{a}}}{\sqrt{2\frac{x}{a}}}\right) \cdot 100\,\% = \left(1 - \sqrt{\frac{1}{2}}\right) \cdot 100\,\% \approx 29,3\,\% \quad (4.19)$$

Im Realfall liegt jedoch eine ungleichmäßig beschleunigte Bewegung vor (siehe Bild 34), da die Antriebe des Roboters eine gewisse Zeit benötigen, um die maximale Beschleunigung zu erreichen. Man spricht in diesem Fall von einem Verfahrprofil mit Ruckbegrenzung, das bedeutet, dass die Beschleunigung abschnittsweise reduziert ist. Die resultierende Zeiteinsparung bei einer Halbierung der Bearbeitungsstrecke fällt somit geringer aus als im zuvor beschriebenen Idealfall. Wird die Maximalbeschleunigung und -geschwindigkeit bei beiden Bewegungen nicht erreicht, beträgt die



(a) Maximalbeschleunigung und -geschwindigkeit nicht erreicht

(b) Maximalbeschleunigung und -geschwindigkeit erreicht

Bild 34: Verfahrprofile mit Ruckbegrenzung

prozentuale Zeiteinsparung $20,6\,\%$ bei einer Halbierung der Bearbeitungsstrecke x gemäß:

$$p_{a\neq\text{const}} = \left(1 - \frac{\sqrt[3]{6\frac{\frac{x}{2}}{r}}}{\sqrt[3]{6\frac{x}{r}}}\right) \cdot 100\,\% = \left(1 - \sqrt[3]{\frac{1}{2}}\right) \cdot 100\,\% \approx 20,6\,\% \quad (4.20)$$

Bei vergleichbaren Verfahrprofilen liegt die für ein reales Robotersystem vorliegende prozentuale Bewegungszeiteinsparung p_{M} zwischen den beiden zuvor beschriebenen Fällen und lässt sich in Abhängigkeit von der zurückgelegten Bearbeitungsstrecke x beschreiben durch:

$$p_{\mathrm{M}} = \left(1 - \frac{t_{\mathrm{M}}\left(\frac{x}{2}\right)}{t_{\mathrm{M}}\left(x\right)}\right) \cdot 100\,\% \in \mathbb{R} \mid 20,6\,\% \leq p_{\mathrm{M}} < 29,3\,\% \tag{4.21}$$

Bei einem Wechsel des Verfahrprofils können auch höhere prozentuale Bewegungszeiteinsparung entstehen. Die Bewegungszeiten t_{M} ergeben sich in Abhängigkeit von der zurückgelegten Bearbeitungsstrecke x aus unterschiedlichen Gleichungen, je nachdem ob die Maximalgeschwindigkeit und -beschleunigung erreicht werden oder nicht. Um diese beiden Fälle unterscheiden zu können, kann eine Grenzstrecke x_{G}, ab der eine konstante Beschleunigung erfolgt (Grenzfall zwischen Bild 34a und 34b), über die Summe der Teilstrecken für die drei Beschleunigungsabschnitte Δx_{J} berechnet werden durch:

$$\begin{gathered} x_{\mathrm{G}} = \Delta x_{\mathrm{J}_1} + \Delta x_{\mathrm{J}_2} + \Delta x_{\mathrm{J}_3} \\ \text{mit } \Delta x_{\mathrm{J}_i} = \frac{1}{6} \cdot r_i \cdot \Delta t^3 + \frac{1}{2} \cdot a_{i-1} \cdot \Delta t^2 + v_{i-1} \cdot \Delta t\,, \\ i \in \{1,\ 2,\ 3\} \end{gathered} \tag{4.22}$$

Für den Fall, dass die Maximalgeschwindigkeit und -beschleunigung nicht erreicht wird, folgt die Bewegungszeit t_{M} aus:

$$t_{\mathrm{M}}\left(x\right) = t_{a \neq \mathrm{const}}\left(x\right) \quad \text{für } x \leq x_{\mathrm{G}} \tag{4.23}$$

Wenn die Maximalgeschwindigkeit und -beschleunigung erreicht werden, setzt sich die Bewegungszeit t_{M} aus einer wegunabhängigen Komponente $t_{a \neq \mathrm{const}}$ und einer wegabhängigen Komponente $t_{a = \mathrm{const}}$ zusammen:

$$t_{\mathrm{M}}\left(x\right) = t_{a \neq \mathrm{const}} + t_{a = \mathrm{const}}\left(x\right) \quad \text{für } x > x_{\mathrm{G}} \tag{4.24}$$

Zeitliche Optimierung der Aufteilung durch gleichmäßige Auslastung

Für die zeitliche Optimierung der Aufteilung durch gleichmäßige Auslastung der beteiligten Roboter muss die zuvor beschriebene Methode für die Aufteilung (siehe Abschnitt 4.1) erweitert werden (siehe Bild 35). Hierbei erfolgt eine Optimierungsschleife, bis diese anhand von Abbruchkriterien verlassen wird. Nach dem ersten Durchlauf der Berechnung folgt eine Optimierungsberechnung, die angepasste Skalierungsfaktoren ausgibt. Mit den optimierten Skalierungsfaktoren erfolgt dann eine erneute Bahnaufteilung gefolgt von einer Selektion der am besten geeigneten Lösung.

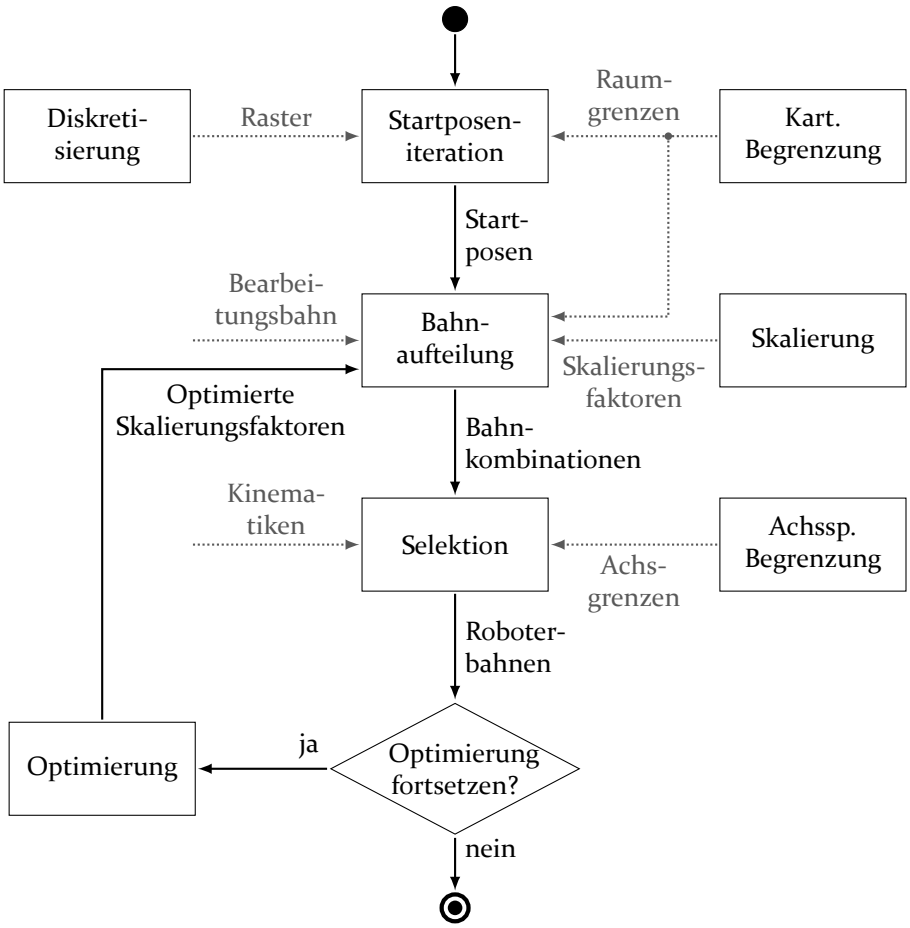


Bild 35: Methode für die zeitliche Optimierung der Aufteilung von Bearbeitungsprozessen auf mehrere Industrieroboter

Für die Bestimmung eines optimierten Skalierungsfaktors s_{opt}, bei dem die Gesamtzeit der parallel ausgeführten Bewegungen minimal ist, werden die Bewegungszeiten der beiden Roboter t_{T} und t_{W} für eine aufgeteilte Bewegung herangezogen und in ein Verhältnis zueinander gesetzt:

$$s_{\text{opt}} = \left(1 - \frac{t_{\text{T}} - t_{\text{W}}}{t_{\text{T}} + t_{\text{W}}}\right) \cdot s = \frac{2\,t_{\text{W}}}{t_{\text{T}} + t_{\text{W}}} \cdot s \;\in\; \mathbb{R} \mid 0 \leq s_{\text{opt}} \leq 1 \tag{4.25}$$

Angestrebt wird eine gleichmäßige Aufteilung, das bedeutet $t_{\text{T}} = t_{\text{W}}$ bzw. $\Theta_{\text{T}} = \Theta_{\text{W}}$, wenn das Beschleunigungsverhalten unbekannt ist oder als gleichwertig eingestuft wird. Der Zusammenhang lässt sich anhand einer einfachen linearen Bearbeitungsbahn, die mit unterschiedlichen Skalierungsfaktoren s durchgerechnet wird, darstellen (siehe Bild 36). Der Startpunkt der Optimierung liegt mittig im Intervall, also bei $s = 0,5$. Ausgehend hiervon nähert sich die Iteration immer mehr dem Schnittpunkt der beiden maximalen Achswinkeländerungen. Das Iterationsverhalten lässt sich ebenfalls anhand einer exemplarischen Bearbeitungsbahn veranschaulichen, indem diese durchgerechnet wird und für jeden Optimierungsschritt die Summe der maximalen Achswinkeländerungen Θ erfasst wird (siehe Bild 37). Als Abbruchkriterien für die Iteration können Grenzwerte für die maximale Achswinkeländerung Θ_{max}, dessen Veränderung je Iterationsschritt $\Delta\Theta$ oder die Anzahl der Durchläufe N_{I} verwendet werden. Im

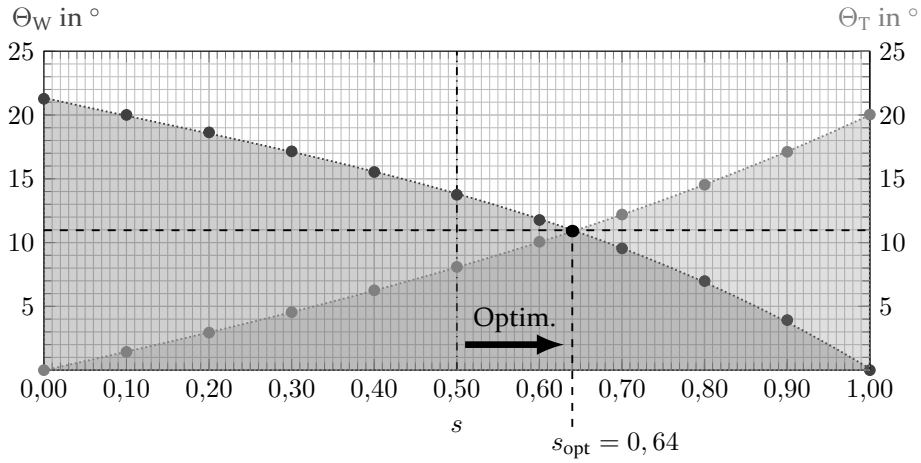
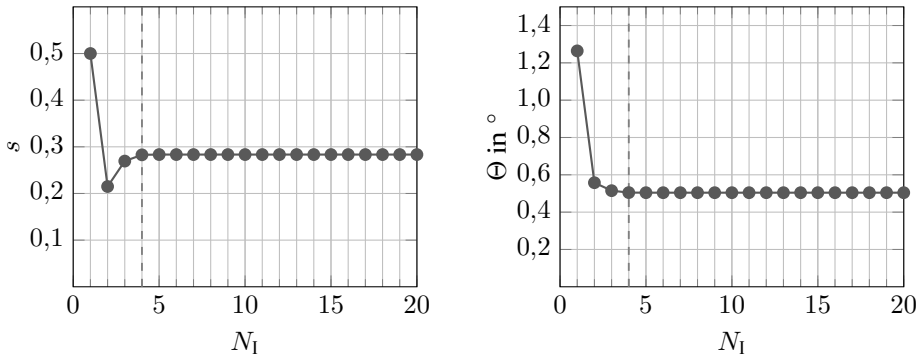


Bild 36: Abhängigkeit der beidseitigen maximalen Achswinkeländerungen von der Skalierung



(a) Skalierungsfaktor

(b) Summer der maximalen Achswinkeländerungen

Bild 37: Iterationsverhalten bei der Optimierung der Skalierung

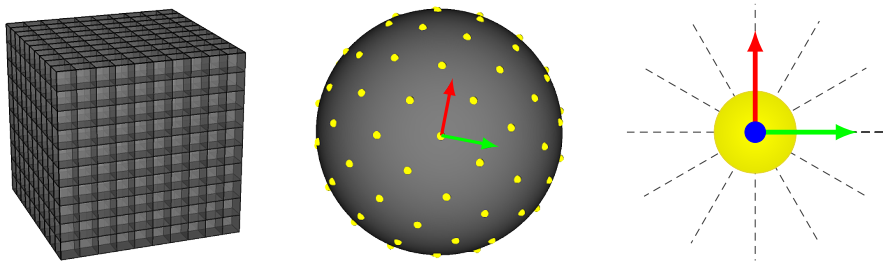
betrachteten Beispiel erfolgt der Abbruch nach $N_\mathrm{I} = 4$ Iterationsschritten, da die Veränderung der maximalen Achswinkeländerung $\Delta\Theta$ gegen Null geht.

4.3.2 Einfluss der kooperativen Bearbeitung auf die Erreichbarkeit

Hinsichtlich der Untersuchung der Erreichbarkeit bei der kooperativen Bearbeitung von Werkstücken wird zunächst ein allgemeiner Ansatz für die Untersuchung der Erreichbarkeit von Industrierobotern betrachtet und anschließend wird dieser Ansatz auf die kooperative Bearbeitung erweitert. Ziel ist es einerseits eine Methode zu entwickeln, durch welche der gemeinsame Arbeitsraum für die kooperative Bearbeitung bestimmt werden kann. Andererseits soll nicht nur die Erreichbarkeit von Positionen untersucht werden, sondern auch inwieweit diese Positionen mit bestimmten Orientierungen angefahren werden können.

Untersuchung der Erreichbarkeit von Industrierobotern

Für die Analyse der Erreichbarkeiten von Industrierobotern kommen *Reachability-Maps* zum Einsatz [89, 129], mit denen der Arbeitsraum der Roboter sowie die lokal vorherrschenden Erreichbarkeiten abgebildet werden. *Reachability-Maps* repräsentieren somit alle Posen, die mit dem im TCP liegende Koordinatensystem des Roboters erreichbar sind. Für die Generierung einer *Reachability-Map* wird zunächst ein Kubus definiert, der den Arbeitsraum des Roboters umfasst (siehe Bild 38a). Dieser wird



(a) Voxelraster ($10 \times 10 \times 10$) (b) Kugelpunkte ($N_P = 100$) (c) Drehrichtungen ($N_D = 12$)

Bild 38: Hierarchische Diskretisierung des Roboterarbeitsraums

anschließend in gleichmäßig große Voxel mit der Kantenlänge l_R aufgeteilt. Für jeden der N_K Voxel wird eine Kugel definiert, welche den Voxel füllt. Auf der Oberfläche der Kugel werden N_P Punkte gemäß dem Ansatz aus [96] gleichmäßig verteilt (siehe Bild 38b). Für jeden Punkt wird eine Pose mit der z-Achse durch das Zentrum der Kugel definiert. Hiervon ausgehend werden N_D Posen durch eine gleichmäßige Rotation um die z-Achse festgelegt (siehe Bild 38c). Somit ergeben sich für jede Kugel $N_P \cdot N_D$ Posen, die alle Orientierungen in diesem Bereich des Voxel-Gitters repräsentieren.

Ob eine Pose erreichbar ist oder nicht, wird mittels kinematischer Berechnungen bestimmt. Gemäß [89] gibt es hierfür drei unterschiedliche Methoden (siehe Tabelle 1). Die *Forward Kinematics* (FK)-Methode berechnet die erreichbaren Positionen anhand von zufällig generierten Achswerten. Im Gegensatz dazu überprüft die *Inverse Kinematics* (IK)-Methode die Erreichbarkeit einer Position durch Lösen der Rückwärtstransformation. Die FK-Methode benötigt im Vergleich zwar geringe Berechnungszeiten, eine

Tabelle 1: Vergleich der Methoden zur kinematischen Berechnung einer Reachability-Map

Methode	Zeit	Vollständigkeit	Genauigkeit
FK	++	−	−
IK	−	++	++
HYB	+	++	+

Vollständigkeit ist jedoch nicht gewährleistet und die Genauigkeit der erzeugten *Reachability-Map* ist verhältnismäßig schlecht. Demgegenüber liefert die IK-Methode vollständige und genaue *Reachability-Maps*. Die benötigte Berechnungszeit hingegen ist verhältnismäßig groß. Durch die Kombination beider Ansätze in der *Hybrid* (HYB)-Methode werden die Vorteile vereint. Hierbei wird zunächst die FK-Methode angewandt bis ein zeitliches Abbruchkriterium eintritt. Anschließend werden alle noch fehlenden Posen mittels der IK-Methode überprüft. Somit können vollständige *Reachability-Maps* schneller als mit der IK-Methode erzeugt werden, die Genauigkeit ist jedoch geringer.

Für jede der N_S definierten Kugeln wird anhand der Anzahl an erreichten Posen N_E ein *Reachability-Index* D_k gemäß der folgenden Formel berechnet:

$$D_k = \frac{N_{\mathrm{E}_k}}{N_\mathrm{P} \cdot N_\mathrm{D}} \cdot 100\,\% \in \mathbb{R} \mid 0\,\% \leq D_k \leq 100\,\%\,, \qquad k \in \{1,\ 2,\ \ldots,\ N_\mathrm{K}\} \tag{4.26}$$

Der Index ergibt sich aus dem Verhältnis von erreichbaren Posen zu überprüften Posen. Somit liegt der Wert zwischen $D = 0\,\%$ für keine erreichbare Pose und $D = 100\,\%$ wenn alle Posen erreichbar sind. Für die Visualisierung werden die Kugeln entsprechend des zugehörigen Index eingefärbt dargestellt (siehe Bild 39). Üblicherweise wird hierfür eine *Heatmap* verwendet, die von rot für $D = 0\,\%$ über grün für $D = 50\,\%$ bis blau für $D = 100\,\%$ definiert ist.

Die Genauigkeit einer *Reachability-Map* hängt neben der kinematischen Berechnung vor allem von der Auflösung ab. Je feiner die Unterteilung bezüglich Voxelkantenlänge l_R, Anzahl an Kugelpunkten N_P und Anzahl an Drehrichtungen N_D ist, desto höher ist die Genauigkeit. Dies ist jedoch mit einer Zunahme der Berechnungszeit verbunden.

Der bisher beschriebene Ansatz ermittelt die Erreichbarkeit einer Pose lediglich über das kinematische Modell. Weiterführend lässt sich die Repräsentativität durch Implementierung einer Kollisionserkennung verbessern, da der reale Roboter durch Kontakt mit seiner Umgebung und sich selbst in seiner Bewegung eingeschränkt wird. Hierzu werden der Roboter und alle weiteren relevanten Komponenten modelliert und auf räumliche Überschneidungen überprüft. Auch hierbei gilt: Je genauer das Kollisionsmodell, desto größer die notwendigen Berechnungszeiten.

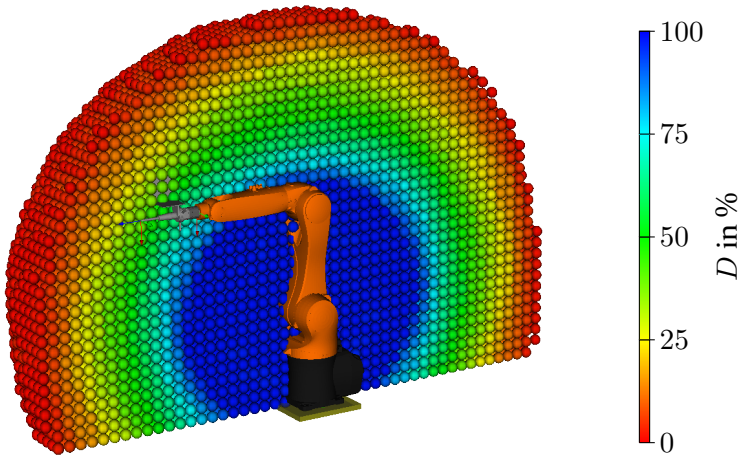


Bild 39: Ausschnitt einer ohne Kollisionserkennung erstellten Reachability-Map eines KUKA KR 6 R900 sixx Knickarmroboters

Erweiterung des Ansatzes auf die kooperative Bearbeitung

Es existiert bereits eine Reihe von Ansätzen, die *Reachability-Maps* auf zweiarmige Prozesse zu erweitern [25, 50, 107]. Dabei wird aber lediglich eine feste Positionierung der beiden Arme zueinander berücksichtigt. Deshalb erfolgt eine Erweiterung des Ansatzes auf die kooperative Bearbeitung durch gegenläufiges Bewegen von Werkzeug und Werkstück [P7]. Hierbei werden mehrere Posen und Transformationen verwendet (siehe Bild 40), die im Folgenden beschrieben sind. Jeder der beiden Roboter hat eine Basistransformation vom Ursprung zu seiner Basis, die mit ${}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_T}}$ und ${}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_W}}$ bezeichnet werden. Ebenso ist für beide Roboter eine Transformation von deren Basis zu deren Flansch definiert, die mit ${}^{\mathrm{B_T}}\mathbf{T}_{\mathrm{F_T}}$ und ${}^{\mathrm{B_W}}\mathbf{T}_{\mathrm{F_W}}$ benannt sind. Die Transformation vom Flansch des werkzeugführenden Roboters zu dessen Koordinatensystem im TCP wird mit ${}^{\mathrm{F_T}}\mathbf{T}_{\mathrm{T}}$ angegeben. Der TCP des werkstückführenden Roboters liegt im Zentrum der Kugel, die entsprechende Transformation lautet ${}^{\mathrm{F_W}}\mathbf{T}_{\mathrm{K}}$. Das Zentrum der Kugel mit Bezug zum Ursprung wird mit der Transformation ${}^{\mathrm{O}}\mathbf{T}_{\mathrm{K}}$ beschrieben. Ausgehend vom Zentrum werden die auf der Kugel verteilten Posen $\mathcal{P}$ durch die Transformation ${}^{\mathrm{K}}\mathbf{T}_{\mathcal{P}}$ beschrieben.

Gemäß der kooperativen Bearbeitung wird bei diesem Ansatz zusätzlich zur Bewegung des Werkzeugkoordinatensystems auch die Kugel bewegt. Prinzipiell sind also drei Bewegungen möglich: eine reine *Werkzeugbewegung*, eine reine *Werkstückbewegung* und eine *kooperative Bewegung*. Weitergehend werden die jeweiligen Transformationsberechnungen beschrieben.

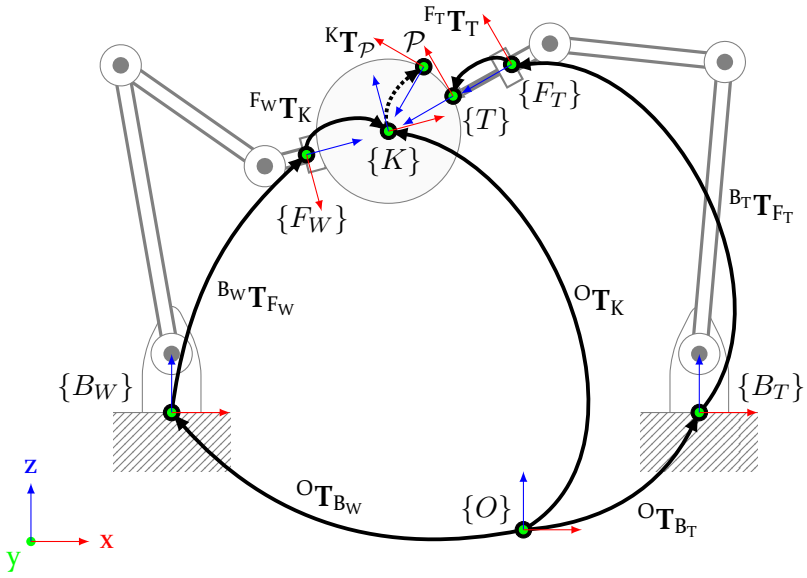


Bild 40: Koordinatensysteme und Transformationen zur Erstellung von Reachability-Maps für die kooperative Bearbeitung

Bei einer reinen Werkzeugbewegung wird für jede der N_K Kugeln das Koordinatensystem im TCP des werkzeugführenden Roboters auf den $N_P \cdot N_D$ Posen auf der Kugel $\mathcal{P}$ positioniert. Die Ziel-Posen des Roboters ergeben sich aus:

$$\begin{aligned} {}^{B_T}\mathbf{T}_{T_i} &= \left({}^{O}\mathbf{T}_{B_T}\right)^{-1} \cdot {}^{O}\mathbf{T}_{S_k} \cdot {}^{S_k}\mathbf{T}_{\mathcal{P}_i} \,, \\ i \in \{1,\ 2,\ \ldots,\ N_P \cdot N_D\},\, k &\in \{1,\ 2,\ \ldots,\ N_K\} \end{aligned} \tag{4.27}$$

Bei einer reinen Werkstückbewegung wird jede der N_K Kugeln mit ihren $N_P \cdot N_D$ Posen $\mathcal{P}$ an einem fixen Werkzeugkoordinatensystem positioniert. Die zugehörige Transformation ergibt sich aus:

$$\begin{aligned} {}^{B_W}\mathbf{T}_{S_i} &= \left({}^{O}\mathbf{T}_{B_W}\right)^{-1} \cdot {}^{O}\mathbf{T}_{B_T} \cdot {}^{B_T}\mathbf{T}_{T} \cdot \left({}^{S_k}\mathbf{T}_{\mathcal{P}_i}\right)^{-1} \,, \\ i \in \{1,\ 2,\ \ldots,\ N_P \cdot N_D\},\, k &\in \{1,\ 2,\ \ldots,\ N_K\} \end{aligned} \tag{4.28}$$

Für die kooperative Bewegung werden die beiden zuvor beschriebenen Bewegungen kombiniert. Zunächst werden alle Posen auf der Kugel $\mathcal{P}$ auf

Erreichbarkeit durch eine Werkzeugbewegung überprüft. Alle so nicht erreichbaren Posen werden weitergehend auf Erreichbarkeit durch eine zusätzliche Werkstückbewegung überprüft. So kann sich maximal ein Reachability-Index von $D = 100\,\%$ ergeben. Der Algorithmus für das Erzeugen der Reachability-Maps für die kooperative Bearbeitung (siehe Bild 41) iteriert über alle der N_K Elemente des Voxel-Gitters. Nur die durch den werkstückführenden Roboter erreichbaren Zentren S_k werden weitergehend berücksichtigt. Für jede der Kugeln wird zunächst die Erreichbarkeit aller Kugel-Posen $\mathcal{P}$ mittels Werkzeugbewegung überprüft. Alle erreichbaren Posen $\mathcal{P}_\mathrm{R} \subseteq \mathcal{P}$ und alle nicht erreichbaren Posen $\mathcal{P}_\mathrm{N} \subseteq \mathcal{P}$

```
1:  procedure GenerateMap(S_k)
2:      for S_k ← 1, N_K do
3:          D ← 0
4:          if Reachable(S_k) then
5:              for all p ∈ P do
6:                  if Reachable(S_k, p) then
7:                      p → P_R
8:                      D ← D + 100 %/(N_P · N_D)
9:                  else
10:                     p → P_N
11:                 end if
12:             end for
13:             if D ≠ 0 then
14:                 for all p_N ∈ P_N do
15:                     for all p_R ∈ P_R do
16:                         if Reachable(S_k, p_N, p_R) then
17:                             D ← D + 100 %/(N_P · N_D)
18:                             break
19:                         end if
20:                     end for
21:                 end for
22:                 Save(S_k, D)
23:             end if
24:         end if
25:     end for
26: end procedure
```

Bild 41: Algorithmus zur Erstellung von Reachability-Maps für die kooperative Bearbeitung

werden separat abgespeichert. Im Falle einer erreichbaren Pose wird der *Reachability-Index* D um $100\,\%/(N_\mathrm{P} \cdot N_\mathrm{D})$ vergrößert. Sollte keine der Posen erreichbar sein, wird der Index auf $D = 0\,\%$ gesetzt und die Berechnung mit der nächsten Kugel fortgesetzt. Andernfalls werden alle noch nicht erreichten Posen $\mathcal{P}_\mathrm{N}$ auf Erreichbarkeit durch eine Werkstückbewegung überprüft. Wenn sich eine erreichbare Kombination aus einer bereits erreichten Werkzeugpose und einer noch fehlenden Kugel-Pose ergibt, wird der Index weiter erhöht. Sobald alle Kugel-Posen durchlaufen sind, wird der berechnete *Reachabiliy-Index* inklusive der Zentrumskoordinaten der Kugel abgespeichert. Die Indizes mit einem Wert von $D = 0\,\%$ werden nicht abgespeichert, um die Datenmenge so gering wie möglich zu halten.

4.3.3 Relative Positioniergenauigkeit der beteiligten Industrieroboter

Ziel dieses Ansatzes ist die Ermittlung und Minimierung der relativen Positionsabweichungen zwischen den beiden Robotern bei einer werkstückbezogenen Positionierung zueinander. Zunächst erfolgt die Betrachtung der Toleranzkette bei der kooperativen Bearbeitung. Hierauf aufbauend werden ein Ansatz für die Erfassung der relativen Abweichungen innerhalb des gemeinsamen Arbeitsraumes der kooperierenden Industrieroboter sowie eine Kompensation dieser aufgezeigt.

Toleranzkettenbetrachtung für die kooperative Bearbeitung

Ausschlaggebend für die Bearbeitungsgenauigkeit bei einer kooperativen Bearbeitung ist die relative Positionierung von Werkstück und Werkzeug. Die dabei resultierenden Toleranzen ergeben sich aus einer Reihe von Maßen M, die aus den einzelnen Komponenten des Systems resultieren (siehe Bild 42). Die betrachtete Toleranzkette geht über den werkstückführenden

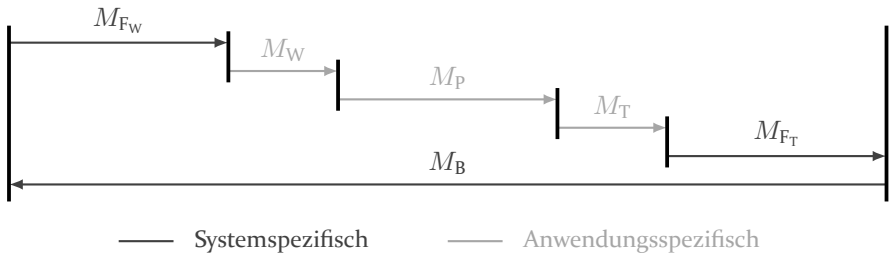


Bild 42: Toleranzkette bei einer kooperativen Bearbeitung

Roboter M_{F_W}, die Fixierung des Werkstücks an dessen Flansch M_W, das Werkstück selbst M_P, das am anderen Roboter fixierte Werkzeug M_T bis zum werkzeugführenden Roboter M_{F_T} und wieder zurück zum werkstückführenden Roboter M_B. Um eine Aussage über die Bearbeitungsgenauigkeit treffen zu können, wird das Maß des Werkstücks M_P gesucht. Für die Ermittlung der erzielbaren Bearbeitungsgenauigkeit kann die Maximum-Minimum-Methode verwendet werden [97]. Jedes Maß M wird dabei in drei Komponenten zerlegt:

$$M = N + E_C \pm T/2 \tag{4.29}$$

Es besteht dabei aus einem Nennwert N, einer Toleranz T sowie einem Toleranzmittenabmaß E_C. Die drei Komponenten werden in der Berechnung unabhängig voneinander betrachtet. Für das Nennmaß N und das Toleranzmittenabmaß E_C gilt, dass die Summe aller Maße jeweils Null ergibt. Die Summe aller Toleranzen T ist gleich der Schlusstoleranz T_0:

$$T_0 = \sum_{i=1}^{n} (T_i) \tag{4.30}$$

Aufgrund der vielen mit Abweichungen behafteten Komponenten des Systems, kann die resultierende Schlusstoleranz groß ausfallen (siehe Tabelle 2). Deshalb kann eine Kompensation der Abweichungen erforderlich sein, um eine ausreichende Bearbeitungsgenauigkeit erzielen zu

Tabelle 2: Toleranzkette und Kompensationen bei einer kooperativen Bearbeitung

Maß	Abhängigkeit	Status	Kompensation
M_{F_W}	System	Variabel	Kinematische Kalibrierung
M_W	Anwendung	Feststehend	Fertigungsgenauigkeit
M_P	Anwendung	Feststehend/Variabel	Ggf. Werkstückerfassung
M_T	Anwendung	Feststehend	Fertigungsgenauigkeit
M_{F_T}	System	Variabel	Kinematische Kalibrierung
M_B	System	Feststehend	Kinematische Kalibrierung

können. Toleranzen bei der Fixierung des Werkstücks, dem Werkzeug oder der Befestigung der Roboter können durch entsprechende Fertigungsgenauigkeiten gering gehalten werden. Die Positioniergenauigkeit von Robotern kann, aufgrund von Auswirkungen wie Reibung und Spiel, mit der Position im Arbeitsraum stark variieren. Um dem entgegen zu wirken, kann eine *kinematische Kalibrierung* erfolgen, wie sie weitergehend vorgestellt wird. Zudem soll im Rahmen dieser Arbeit auch berücksichtigt werden, dass die Toleranzen der Werkstückgeometrie verhältnismäßig groß ausfallen. Daher wird im nächsten Kapitel eine *prozessspezifische Werkstückerfassung* erarbeitet. Eine Alternative hierzu ist der *sensorgeführte Prozess*. Wird hierbei auch das Werkzeug durch den Sensor erfasst, kann die komplette Toleranzkette in einem Schritt berücksichtigt werden. Ein vergleichbares Konzept, ein Berücksichtigen der kompletten Toleranzkette durch einen werkstückbezogenen Sensor, liegt bei der gemeinsamen Handhabung von Lasten mit Hilfe von Kraft- und Drehmomentsensoren [130] vor.

Kinematische Kalibrierung

Anhand der in den Grundlagen beschriebenen Methode zur Messung der Genauigkeit von Industrierobotern (siehe Abschnitt 2.4.1) können durch den einzelnen Industrieroboter angefahrene Positionen präzise erfasst werden. Anschließend lassen sich anhand der gemessenen Positionen die jeweiligen Abweichungen sowie eine zweckmäßige Kompensation berechnen. Dieser Ablauf wird auch als *kinematische Kalibrierung* bezeichnet und kann gemäß zweier grundsätzlicher Prinzipien erfolgen (siehe Abschnitt 2.4.2): Entweder erfolgt die Kalibrierung in einer *geschlossenen Korrekturschleife*, in welcher der lokal auftretende Fehler während des Prozesses erfasst und kompensiert wird oder sie erfolgt in einer *offenen Korrekturschleife*, bei der die Abweichungen vor dem Prozess erfasst und durch Anpassung des kinematischen Modells kompensiert werden. Aus letzterem wird die Korrekturmethode für das kooperative Robotersystem abgeleitet (siehe Bild 43), bei der nicht das kinematische Modell angepasst wird, sondern die Zielpose.

Gemäß [36] lässt sich die kinematische Kalibrierung in vier Schritte einteilen. Der erste Schritt ist die kinematische *Modellierung* des Roboters. Als zweites werden die für die Kalibrierung erforderlichen *Messungen* durchgeführt. Der dritte Schritt ist die *Identifizierung* der Abweichungen und im letzten Schritt erfolgt eine *Kompensation* der Abweichungen. Im Folgenden werden diese vier Schritte mit Bezug auf die durchgeführte Untersuchung genauer erläutert.

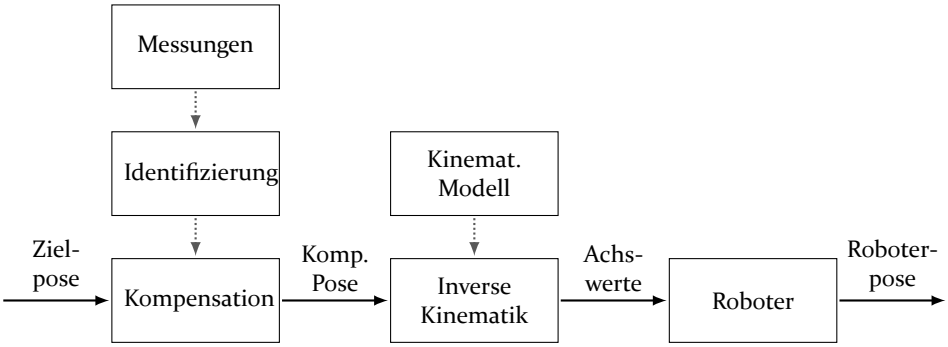


Bild 43: Methode der offenen Korrekturschleife für das betrachtete kooperative Robotersystem

Modellierung Für die Erweiterung der Untersuchung auf das kooperative Robotersystem sind weitere Transformationen notwendig (siehe Bild 44). Neben den jeweiligen Transformationen zum Flansch ${}^{B_T}\mathbf{T}_{F_T}$ und ${}^{B_W}\mathbf{T}_{F_W}$ sowie den Transformationen zu den Koordinatensystemen im jeweiligen TCP ${}^{F_T}\mathbf{T}_{T_T}$ und ${}^{F_W}\mathbf{T}_{T_W}$, gibt es eine Transformation zwischen den beiden Basiskoordinatensystemen der Roboter $\{B_T\}$ und $\{B_W\}$, die mit ${}^{B_T}\mathbf{T}_{B_W}$ bezeichnet wird. Die Lage des jeweiligen Basiskoordinatensystems mit Bezug

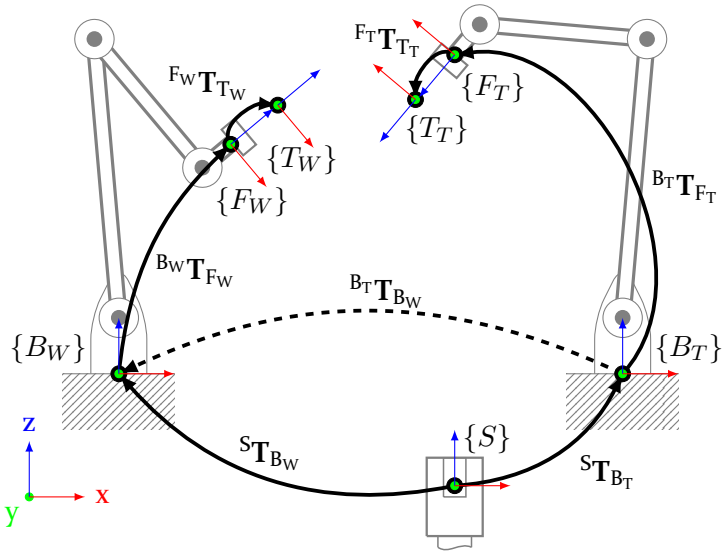


Bild 44: Koordinatensysteme und Transformationen für die Genauigkeitsuntersuchung am kooperativen Robotersystem

auf den Sensor wird durch die Transformation ${}^{S}\mathbf{T}_{B_T}$ bzw. ${}^{S}\mathbf{T}_{B_W}$ beschrieben. Aus den Basiskoordinatensystemen kann die Basistransformation zwischen beiden Robotern berechnet werden durch:

$$ {}^{B_T}\mathbf{T}_{B_W} = \left({}^{S}\mathbf{T}_{B_T}\right)^{-1} \cdot {}^{S}\mathbf{T}_{B_W} \tag{4.31} $$

Messungen Die hier beschriebene Untersuchung beschränkt sich auf den gemeinsamen Arbeitsraum der kooperierenden Roboter. Um die Zahl der Messungen zu begrenzen, aber dennoch den gesamten gemeinsamen Arbeitsraum abzudecken, wird ein Algorithmus entwickelt (siehe Bild 45), der gleichmäßig verteilte Messpositionen im Arbeitsraum ermittelt. Der Algorithmus läuft mit einer konstanten Schrittweite entlang der z-Achse durch einen Quader, welcher den gemeinsamen Arbeitsbereich umfasst. In jeder Ebene iteriert der Algorithmus in Form einer Rechteckfunktion

```
 1: procedure GeneratePaths(S_k)
 2:     for z ← z_min, z_max do
 3:         for y ← y_min, y_max do
 4:             if toggle then
 5:                 for x ← x_min, x_max do
 6:                     if Reachable(x, y, z) then
 7:                         commands → programs
 8:                     end if
 9:                 end for
10:             else
11:                 for x ← x_max, x_min do
12:                     if Reachable(x, y, z) then
13:                         commands → programs
14:                     end if
15:                 end for
16:             end if
17:             toggle ←!toggle
18:         end for
19:     end for
20: end procedure
```

Bild 45: Algorithmus für das Erzeugen der Roboterbahnen für die Genauigkeitsuntersuchung am kooperativen Robotersystem

entlang der x-Achse, wobei die Richtung abwechselt, um die Roboterbewegungen minimal zu halten. Jede der iterierten Positionen wird durch Berechnung der inversen Kinematik (siehe Abschnitt 2.2) auf Erreichbarkeit überprüft. Wenn eine Position erreichbar ist, wird diese der Liste der Messpositionen hinzugefügt.

Auf diese Weise wird für beide Roboter jeweils ein Programm erzeugt, mit dem der Roboter den Reflektor der Reihe nach an alle raumbezogenen Messpositionen bewegt. Der resultierende Bahnverlauf deckt den gemeinsamen Arbeitsraum der beiden Roboter ab. Nach jedem Bewegungsbefehl wird dem Programm eine Zeitverzögerung hinzugefügt, um eine positionsgetreue Messung zu gewährleisten.

Identifizierung Für das betrachtete System ist die relative Positionierung der beiden Roboter zueinander von wesentlichem Interesse, weshalb Gleichung 2.17 und Gleichung 2.18 (siehe Abschnitt 2.4.1) entsprechend angepasst werden. Somit ergibt sich der Vektor des relativen Fehlers $\mathbf{e}_{\text{rel}} = \begin{pmatrix} x_{\text{E}_{\text{rel}}} & y_{\text{E}_{\text{rel}}} & z_{\text{E}_{\text{rel}}} \end{pmatrix}^{\text{T}}$ aus dem Unterschied zwischen der gemessen Werkzeugposition $\mathbf{p}_{\text{T}}$ und der gemessenen Werkstückposition $\mathbf{p}_{\text{W}}$ gemäß:

$$\mathbf{e}_{\text{rel}_i} = \mathbf{p}_{\text{T}_i} - \mathbf{p}_{\text{W}_i} \,, \quad i \in \{1,\ 2,\ \ldots,\ N\} \quad \text{mit} \quad \mathbf{p} = \begin{pmatrix} x & y & z \end{pmatrix}^{\text{T}} \tag{4.32}$$

Der Betrag des relativen Fehlers ϵ_{rel} an einer Position resultiert aus dem entsprechenden Abstand zwischen Werkzeugposition $\mathbf{p}_{\text{T}}$ und Werkstückposition $\mathbf{p}_{\text{W}}$ gemäß:

$$\begin{gathered} \epsilon_{\text{rel}_i} = ||\mathbf{e}_{\text{rel}_i}|| = ||\mathbf{p}_{\text{T}_i} - \mathbf{p}_{\text{W}_i}|| = \\ = \sqrt{(x_{\text{T}_i} - x_{\text{W}_i})^2 + (y_{\text{T}_i} - y_{\text{W}_i})^2 + (z_{\text{T}_i} - z_{\text{W}_i})^2} \,, \\ i \in \{1,\ 2,\ \ldots,\ N\} \end{gathered} \tag{4.33}$$

Kompensation Aus den zuvor bestimmten Abweichungen kann mittels des *Iterative Closest Point* (ICP) Algorithmus [133] durch iterative Minimierung der Abstände zwischen den beiden Punktwolken eine Kompensationstransformation $\mathbf{T}_{\text{C}}$ berechnet werden. Die Kompensationstransformation besteht aus einer Translation $\mathbf{t}_{\text{C}} = \begin{pmatrix} x_{\text{C}} & y_{\text{C}} & z_{\text{C}} \end{pmatrix}^{\text{T}}$ und einer Rotation

$\mathbf{r}_C = (\alpha_C \;\; \beta_C \;\; \gamma_C)^T$. Anhand dieser Parameter kann die Basistransformation zwischen den beiden Robotern ${}^{B_T}\mathbf{T}_{B_W}$ angepasst werden gemäß:

$$ {}^{B_T}\mathbf{T}'_{B_W} = {}^{B_T}\mathbf{T}_{B_W} \cdot \mathbf{T}_C \tag{4.34} $$

Somit lässt sich die Zielpose des werkstückführenden Roboters ${}^{B_W}\mathbf{T}_P$ an die Zielposition des werkzeugführenden Roboters ${}^{B_T}\mathbf{T}_P$ anpassen durch:

$$ {}^{B_W}\mathbf{T}_P = (\mathbf{T}_C)^{-1} \cdot \left({}^{B_T}\mathbf{T}_{B_W}\right)^{-1} \cdot {}^{B_T}\mathbf{T}_P = \left({}^{B_T}\mathbf{T}'_{B_W}\right)^{-1} \cdot {}^{B_T}\mathbf{T}_P \tag{4.35} $$

4.4 Zusammenfassung

Der in diesem Kapitel beschriebene Aufteilungsansatz bildet die theoretische Grundlage, um einzelne Prozessbewegungen oder ganze Prozesse in synchron ausgeführte Werkstück- und Werkzeugbewegungen aufzuteilen und enthält zudem die Möglichkeit Randbedingungen zu berücksichtigen. Bei Bedarf kann dabei die kooperative Bearbeitung gezielt anhand der werkstückbezogenen Freiheitsgrade verteilt und gegebenenfalls nur auf einzelne Freiheitsgrade reduziert werden. Zudem können alle beteiligten Kinematiken kartesisch und achsspezifisch begrenzt werden. In Ergänzung zum Aufteilungsansatz sind im Rahmen dieses Kapitels Untersuchungsansätze für die allgemeinen Einflüsse der Aufteilung auf die Bearbeitungszeit, die Erreichbarkeit und die relative Positioniergenauigkeit beschrieben. Diese ermöglichen es, Robotersysteme gezielt hinsichtlich ihrer Eignung für die kooperative Bearbeitung zu beurteilen und auszulegen. Weitergehend gilt es die Funktionalität dieser Ansätze anhand von realen Robotersystemen und Anwendungen unter Beweis zu stellen. Hierfür müssen die Ansätze zunächst in eine entsprechende Offline-Programmiersoftware implementiert werden. Für gewisse Anwendungen können dabei zudem noch spezifische Erweiterungen erforderlich sein, wie für die im nächsten Kapitel dargestellte Erfassung von Werkstücken.

5 Erfassung von Werkstücken mittels kooperierenden Industrierobotern

Der zuvor beschriebene Ansatz für die Aufteilung von Prozessbewegungen auf mehrere Industrieroboter (siehe Kapitel 4) kann auch bei der Erfassung von Werkstücken Anwendung finden. Sei es einerseits als Prozess selbst oder andererseits, um das Werkstück vor dem eigentlichen Prozess zu erfassen. Aus diesem Grund wird im Rahmen dieses Kapitels der entsprechende Ansatz erarbeitet. Analog zum kooperativen Bearbeitungsprozess führt bei diesem Scansystem ein Teil der Roboter das Werkstück und ein weiterer Teil der Roboter den Sensor. Dementsprechend werden weitergehend die Begriffe *werkstückführender Roboter* und *sensorführender Roboter* verwendet.

Als exemplarischer Sensor wird ein Laser-Profilsensor gewählt, der Ansatz ist jedoch auch mit anderen Sensoren anwendbar. Der gewählte Sensor liefert lediglich ein 2D-Profil, durch Verknüpfung mit der Pose des ihn haltenden Roboters können die Profilpunkte im dreidimensionalen Raum platziert und somit eine Punktwolke erzeugt werden. Die Bewegung des Werkstückes kann über die Pose des anderen Roboters auf die bereits vorhandene Punktwolke übertragen werden. Da Laser-Profilsensoren lediglich geometrische Daten erfassen, wird das 3D-Scansystem mit Hilfe einer Kamera um Farbinformationen erweitert, um die erzeugten 3D-Modelle realitätsnäher zu gestalten.

Der Ablauf für eine Werkstückerfassung mit dem erarbeiteten System setzt sich aus drei Teilschritten zusammen, die in den folgenden Abschnitten erläutert werden. Zunächst müssen die unbekannten Transformationen im System mittels Kalibrierung bestimmt werden (siehe Abschnitt 5.1). Anschließend kann der Scanvorgang durchgeführt werden (siehe Abschnitt 5.2), wobei die Messdaten erfasst und transformiert werden. Die resultierenden Rohdaten müssen abschließend noch bearbeitet werden, um daraus letztlich repräsentative Flächenmodelle zu erzeugen (siehe Abschnitt 5.3).

5.1 Scansystem und notwendige Kalibrierungen

Bevor die Roboterposen mit den Messwerten verknüpft werden können, müssen zwei unbekannte Transformationen (siehe Bild 46) ermittelt werden. Einerseits ist die Transformation ${}^{F_S}\mathbf{T}_S$ zwischen Roboterflansch $\{F_S\}$ und Sensorbasis $\{S\}$ unbekannt. Andererseits ist die Transformation

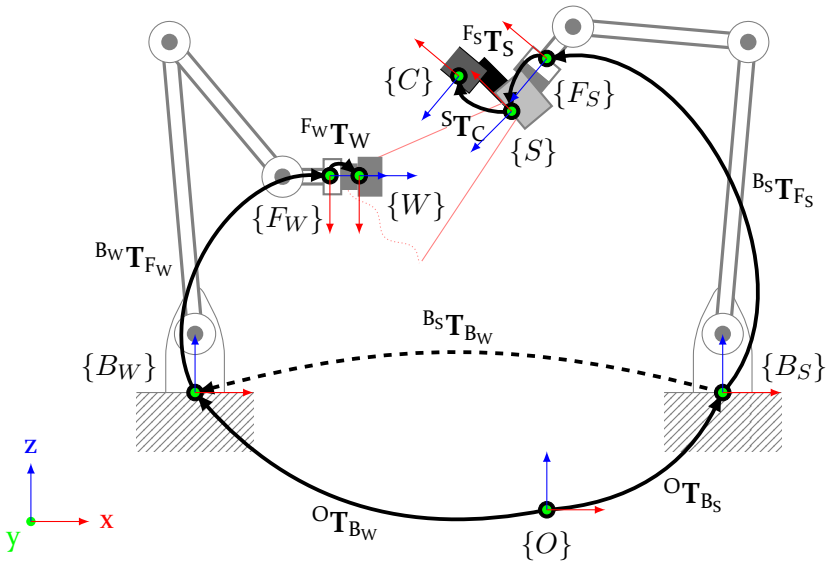


Bild 46: Koordinatensysteme und Transformationen innerhalb des Scansystems mit kooperierenden Industrierobotern

${}^{B_S}\mathbf{T}_{B_W}$ zwischen der Basis des sensorführenden Roboters $\{B_S\}$ und der Basis des werkstückführenden Roboters $\{B_W\}$ unbekannt. Da das unmittelbare Erfassen der Transformationen durch Messgeräte aufgrund der begrenzten Zugänglichkeit nicht möglich ist, erfolgt die Bestimmung mittels Kalibrierung. Des Weiteren muss auch eine Kalibrierung der für eine Farberweiterung verwendeten Kamera erfolgen. Diese muss einerseits intrinsisch und andererseits extrinsisch kalibriert werden. Letzteres dient zur Ermittlung der Transformation ${}^{S}\mathbf{T}_{C}$ zwischen Sensorbasis $\{S\}$ und Kamerabasis $\{C\}$.

Für eine Kalibrierung werden stets Referenzpunkte benötigt. Prinzipiell können diese entweder manuell festgelegt oder automatisch detektiert werden. Bei letzterem kann die Detektion entweder auf Grundlage eines bekannten oder eines unbekannten Referenzobjektes geschehen. Da eine manuelle Kalibrierung oftmals zu aufwendig und zeitintensiv ist, werden im industriellen Bereich immer häufiger automatische Routinen eingesetzt. Diese werden im Rahmen des dieser Arbeit übergeordneten Forschungsprojektes ebenfalls angestrebt, weshalb dabei alternative Kalibrierverfahren entwickelt worden sind [P3, P4]. Da dies jedoch nicht der Schwerpunkt dieser Arbeit ist und der Aufteilungsansatz unabhängig von der Art der Kalibrierung ist, werden die erarbeiteten Kalibrierverfahren hier nicht im

Detail betrachtet, sondern lediglich die drei Kalibriervorgänge, *Sensor-*, *Roboter-* und *Kamerakalibrierung* im Allgemeinen (siehe Abschnitt 5.1.1 bis Abschnitt 5.1.3).

5.1.1 Sensorkalibrierung

Die Transformation zwischen Roboterflansch und Sensorbasis ist nach der Montage des Sensors nicht exakt bekannt. Da das unmittelbare Erfassen der Transformationen durch Messgeräte aufgrund der begrenzten Zugänglichkeit nicht möglich ist, muss die Bestimmung mittels Kalibrierung erfolgen. Bei konventionellen Kalibrierverfahren für im TCP liegende Koordinatensysteme wird dieser mindestens drei mal mit unterschiedlichen Orientierungen an einem fixen Bezugspunkt positioniert [44]. Basierend auf den Posen kann ein Gleichungssystem aufgestellt und mittels der linearen Methode der kleinsten Quadrate eine Lösung angenähert werden. Dabei sind Messspitzen ein üblicherweise in der Robotik eingesetztes Hilfsmittel [51]. Für berührungslose Sensoren ist diese Methode jedoch nicht praktikabel, da die Zugänglichkeit der Basis nicht gegeben ist. Deshalb kommen für die Kalibrierung von Lasersensoren oftmals Kalibrierverfahren zum Einsatz, bei denen der Sensor mittels automatischer Routinen gegenüber fixierten Bezugspunkten positioniert wird (siehe Tabelle 3). Hierbei dienen

Tabelle 3: Vergleich der Kalibrierverfahren für robotergeführte Lasersensoren

Sensor	Kalibrierobjekt	Genauigkeit	Aufwand	Quelle
Linienlaser	Pyramide	−−	−−	[125]
Linienlaser	Marker	−	−	[126]
Punktlaser	Kugel	+	++	[135]
Linienlaser	Kugel	++	+	[66]
Linienlaser	Kugel	−−	+	[17]
Linienlaser	Kugel	++	+	[100]
Linienlaser	Messspitze	+	++	[P4]

beispielsweise Pyramiden mit Aussparungen [125] oder optische Marker [126] als Kalibrierobjekte. Bei den meisten Ansätzen werden Kugeln als Kalibrierobjekte verwendet [17, 66, 100, 135]. Im Rahmen des dieser Arbeit übergeordneten Forschungsprojektes wird ein Kalibrierverfahren entwickelt [P4], bei dem eine Messspitze als fixer Referenzpunkt dient.

Die extrinsische Kalibrierung von an Industrierobotern eingesetzten Lasersensoren erfolgt oftmals in zwei Schritten: Bestimmung der translatorischen Komponente und Bestimmung der rotatorischen Komponente. Exemplarisch hierfür ist die in [135] präsentierte Kalibrierung eines eindimensionalen Lasersensors. Hierbei wird zunächst die rotatorische Komponente der Transformation bestimmt, indem die Projektion des Lasers auf eine planare Oberfläche mit einer stationären Kamera erfasst wird. Dabei wird der Roboter senkrecht zur Oberfläche bewegt und wenn eine rotatorische Abweichung vorliegt verändert sich die Position des projizierten Laserpunktes. Aus der Veränderung der Position können die Rotationen um die beiden Querachsen berechnet werden. Die Rotation um die Laserachse ist für den eindimensionalen Lasersensor nicht relevant. Die translatorische Komponente der Transformation wird mit Hilfe einer Kugel mit einem bekannten Durchmesser bestimmt. Hierfür muss der Laserstrahl aus mindestens sechs Richtungen auf der Kugeloberfläche positioniert werden. Weitergehend kann aus den Messwerten der Mittelpunkt der Kugel ermittelt werden. Letztlich kann aus der Flanschpose des Roboters, der rotatorischen Komponente der Transformation und dem Mittelpunkt der Kugel die Translation berechnet werden. Die mit diesem Ansatz erzielte durchschnittliche Abweichung liegt laut den Autoren bei ungefähr $0,07$ mm und die vorhandene maximale Abweichung liegt bei etwa $0,12$ mm. Die für die Berechnung der Genauigkeitsparameter angegebene Formel deutet jedoch darauf hin, dass diese Parameter nicht repräsentativ und somit auch nicht mit den anderen Verfahren vergleichbar sind.

Ein Kalibrierverfahren für einen Laser-Profilsensor wird in [17] dargestellt. Hierbei wird das Laserprofil aus mindestens drei unterschiedlichen Richtungen auf einer im Arbeitsraum fixierten Kugel mit bekanntem Durchmesser positioniert. Anschließend wird das Zentrum der Kugel mittels Regression bestimmt. Hierauf aufbauend kann ein Gleichungssystem aufgestellt werden, aus dessen Lösung die gesuchte Transformation resultiert. Die mit diesem Kalibrierverfahren erzielten maximalen Abweichungen liegen laut den Autoren bei ungefähr $0,50$ mm. Dieser Wert wurde anhand von exemplarischen Scans ermittelt und stellt die maximale Abweichung zwischen den einzelnen Aufnahmen dar. In [66] und in [100] werden vergleichbare

Ansätze präsentiert. Die erzielte maximale Abweichung wird in [66] mittels sphärischer Regression bestimmt und liegt bei etwa $0,10$ mm. Hierbei wird mit der ermittelten Transformation die Ist-Kugel aus einem Scan berechnet und mit der Soll-Kugel verglichen. Der maximale Abstand zwischen den Kugeln dient als Anhaltspunkt für die Genauigkeit der durchgeführten Kalibrierung. Bei der in [126] präsentierten optischen Methode wird das Laserprofil in unterschiedlichen Orientierungen mit Hilfe einer Kamera auf einem fixierten x-förmigen Marker positioniert. Vergleichbar mit dem zuvor beschriebenen Verfahren, kann aus mindestens drei ausgerichteten Roboterposen ein Gleichungssystem aufgestellt werden, dessen Lösung die gesuchte Transformation ergibt. Für die Berechnung einer Genauigkeitsangabe wird die Position des Bezugspunktes mit der ermittelten Transformation für mehrere Messungen berechnet und die Abweichungen zueinander aufsummiert. Die erzielte durchschnittliche Abweichung liegt mit ungefähr $0,07$ mm nahe der Genauigkeit des verwendeten Lasersensors. Die vorhandene maximale Abweichung liegt bei etwa $0,39$ mm. Das in [P4] dargestellte Kalibrierverfahren verwendet eine Messspitze als fixen Referenzpunkt. Diese ist vertikal innerhalb des Arbeitsbereiches des entsprechenden Roboterarmes montiert, die exakte Position der Messspitze muss jedoch nicht bekannt sein. Die Kalibrierung basiert auf dem automatischen Zentrieren des Sensors gegenüber der Referenzspitze ausgehend von fünf unterschiedlichen Startpositionen. Mit diesem Ansatz wird eine durchschnittliche Abweichung von ungefähr $0,27$ mm erzielt.

Bei den Kalibrierverfahren, die mehrere Messungen verwenden, kann die Genauigkeit verbessert werden, indem mehr als die minimale Menge an Messungen verwendet wird. Aus dem resultierenden Pool von Lösungen kann die beste Lösung durch die nichtlineare Methode der kleinsten Quadrate ausgewählt werden. Weitergehend kann die Genauigkeit des Scansystems auch erhöht werden, indem die einzelnen Aufnahmen durch beispielsweise den ICP-Algorithmus [17] oder die Punktwolkenentropie [71] korrigiert werden. Für die im Rahmen dieser Arbeitet entwickelte Anwendung ist dies jedoch nicht zielführend, da größtenteils keinerlei Überschneidungen zwischen den einzelnen Aufnahmen besteht.

5.1.2 Roboterkalibrierung

Wenn in einem robotergeführten Scansystem mehrere Kinematiken beteiligt sind, muss die Transformation zwischen den Basiskoordinatensystemen der beteiligten Kinematiken $\{B_S\}$ und $\{B_W\}$ bestimmt werden (siehe Bild 46). Das direkte Erfassen der Transformation ${}^{B_S}\mathbf{T}_{B_W}$ durch Messgeräte

ist ebenfalls nicht möglich, da die Basen der Kinematiken an einer nicht zugänglichen Position innerhalb derer liegen. Deshalb wird für die Bestimmung der Transformation üblicherweise ein Kalibrierverfahren verwendet (siehe Tabelle 4).

Bei einer Reihe von roboterbasierten 3D-Scansystemen wird ein Rotationstisch für das Bewegen des Werkstückes eingesetzt [16, 66, 100]. Hierbei muss die Basistransformation zwischen Roboter und Drehtisch ermittelt werden, um die Bewegungen mit den Sensoraufnahmen abstimmen zu können. In [16] wird hierfür der Drehtisch eingescannt und die Exzentrizität und der Versatz aus den Messdaten berechnet. Weitergehend werden unterschiedliche Methoden zur Verifikation der Kalibrierung beschrieben, genaue Zahlen zu den erzielten Genauigkeiten fehlen jedoch. Ein weiteres Verfahren für die Kalibrierung eines Drehtisches gegenüber eines Industrieroboters wird in [66] und in [100] verwendet. Hierbei dient eine exzentrisch auf dem Drehtisch montierte Kugel als Referenzpunkt. Diese wird an mehreren Positionen rotiert und gescannt. Für jeden Scan wird der Mittelpunkt der Kugel berechnet und anhand von mindestens drei Mittelpunkten kann eine Kreisbahn ermittelt werden, durch deren Zentrum die Rotationsachse des Drehtisches verläuft. Durch Wiederholen dieses Vorgangs mit der Kugel in einer anderen Höhe, kann ein zweiter Punkt auf der Drehachse ermittelt und somit die Drehachse selbst beschrieben werden. Die mit dieser Methode erzielte Genauigkeit liegt bei $0,15$ mm, was durch sphärische Regression überprüft wurde [66].

Tabelle 4: Vergleich der Kalibrierverfahren für eine zusätzliche Kinematik

Kinematik	Kalibrierobjekt	Genauigkeit	Aufwand	Quelle
Drehtisch	Drehtisch	+	++	[16]
Drehtisch	Kugel	++	+	[66, 100]
Roboter	Messspitzen	−	−−	[39, 63]
Roboter	Kugel und KMM	+	−−	[117]
Roboter	Messspitze	++	++	[P4]

Beim konventionellen Kalibrierverfahren für MRS werden zwei Messspitzen verwendet, die jeweils an den Flansch der beiden Roboter montiert sind [39, 63]. Diese werden üblicherweise viermal mit jeweils unterschiedlicher Orientierung und Spitze an Spitze innerhalb des gemeinsamen Arbeitsraumes positioniert. Aus den dabei aufgenommen Roboterposen kann ein Gleichungssystem aufgestellt werden, dessen Lösung die Berechnung der Transformation ermöglicht. Zudem wird meist noch ein Schätzwert für die bei der Kalibrierung erzielte Genauigkeit berechnet, die dem Benutzer eine schnelle Einschätzung der Verwendbarkeit der durchgeführten Kalibrierung ermöglicht. Nachteilig bei diesem Kalibrierverfahren ist jedoch das manuelle Positionieren durch den Benutzer, da dies nur bedingt genau und zeitaufwendig ist. Ein alternativer Ansatz verwendet eine *Koordinatenmessmaschine* (KMM) und eine am Roboterflansch montierte Kugel als Referenzpunkt [117]. Für die Kalibrierung wird eine Vielzahl von Punkten auf der Oberfläche der Kugel mit der KMM eingemessen. Ausgehend von mehreren Kugelpositionen kann ein Gleichungssystem aufgestellt und gelöst werden, um die Basistransformation zu berechnen. Der mit diesem Ansatz erzielte mittlere Fehler liegt bei $0,65$ mm. Der in [P4] präsentierte Ansatz baut auf der zugehörigen Sensorkalibrierung auf. Hierbei wird die Messspitze am werkstückführenden Roboter montiert. Die Kalibrierung erfolgt in zwei Schritten, indem zunächst die rotatorische Komponente und dann die translatorische Komponente bestimmt wird. Zur Bestimmung der rotatorischen Komponente wird die Messspitze durch den werkstückführenden Roboter vertikal an drei Positionen im gemeinsamem Arbeitsraum positioniert. Als zweites wird die translatorische Komponente durch Schließen einer Transformationskette über beide Roboter bestimmt. Die mit dieser Methode erzielte durchschnittliche Abweichung beträgt etwa $0,29$ mm.

5.1.3 Kamerakalibrierung

Die Kalibrierung von Kameras setzt sich aus zwei Teilen, der *intrinsischen* und der *extrinsischen Kalibrierung*, zusammen. Die Grundlage für Kameras sind die technischen Prinzipien der Lochkamera. In modernen Kameras werden zusätzlich Linsen eingesetzt, um diese kompakter gestalten sowie hellere und schärfere Bilder erzeugen zu können. Dabei werden die resultierenden Bilder jedoch durch die Linsen verzerrt. Daher ist zunächst eine *intrinsische Kalibrierung* zur Bestimmung der intrinsischen Parameter der Kamera notwendig, sodass die interne Verzerrung kompensiert werden kann. Um eine Zuordnung der Kamerapixel zu den Positionsdaten zu ermöglichen, muss weitergehend die Transformation ${}^{S}\mathbf{T}_{C}$ zwischen der Basis des

Sensors $\{S\}$ und der Basis der Kamera $\{C\}$ (siehe Bild 46) bestimmt werden. Dieser Vorgang wird als *extrinsische Kalibrierung* bezeichnet. Weitergehend werden Verfahren für diese beiden Kalibrierungen beschrieben.

Klassische Verfahren für die intrinsische Kalibrierung von Kameras verwenden komplexe Kalibrierobjekte mit bekannten 3D-Koordinaten [111]. Moderne Verfahren verwenden meist einfachere Kalibrierobjekte, die in den Sensormessdaten erkannt werden, oder ermitteln Bezugspunkte unmittelbar aus den aufgenommenen Szenen. Weit verbreitet ist die Kalibrierung mittels planarem Muster (z. B. Schachbrett), welche in [134] präsentiert wird. Hierbei wird in mehreren Aufnahmen, bei denen die Kamera unterschiedlich gegenüber dem Muster ausgerichtet ist, das Muster erkannt und daraus anschließend die intrinsischen Parameter berechnet. Hierzu zählen der Scherungsparameter γ, die Brennweite, angegeben durch f_x und f_y, sowie der Bildhauptpunkt, angegeben durch c_x und c_y. Diese Parameter lassen sich durch die intrinsische Matrix $\mathbf{K}$ zusammenfassen gemäß:

$$\mathbf{K} = \begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{5.1}$$

Zudem wird noch die Verzerrung des Kamerabildes ermittelt. Diese lässt sich mit dem Verzerrungsvektor $\mathbf{d} = (k_1 \;\; k_2 \;\; p_1 \;\; p_2)^T$ beschreiben, wobei k_1 und k_2 radiale sowie p_1 und p_2 tangentiale Verzerrungskoeffizienten sind.

Auch bei der extrinsischen Kalibrierung ist es schwierig die Transformation mechanisch zu messen, da die Nullpunkte nicht zugänglich sind. Deshalb muss wiederum ein Kalibrierverfahren angewandt werden (siehe Tabelle 5). Hierbei werden Referenzpunkte benötigt, die in beiden Sensoraufnahmen eindeutig erkennbar sind. Eine Möglichkeit diese Punkte zu bestimmen ist das manuelle Auswählen durch den Benutzer, wie exemplarisch in [26] und in [98] umgesetzt. Dies bedeutet jedoch einen erhöhten Aufwand für den Benutzer und zudem ist dieser Vorgang anfällig für Fehler. Alternative Ansätze verwenden Kalibrierobjekte und eine automatische Erkennung derer. Dabei gibt es Herangehensweisen, die bekannte Kalibrierobjekte voraussetzen, aber auch welche, bei denen das Kalibrierobjekt nicht im Detail bekannt sein muss. Eine Reihe von Ansätzen verwenden mehrere kleine Reflektoren als Bezugspunkte. In [84] werden zum Beispiel kleine weiße Scheiben verwendet, die aufgrund von Reflexionen im Kamerabild erkannt werden können.

Tabelle 5: Vergleich der Kalibrierverfahren zwischen Lasersensor und Kamera

Kalibrierobjekt	Kalibriermuster	Manuell	Automatisch	Genauigkeit	Aufwand	Quelle
Platte	Kreise	✓	✓	−−	−	[26]
Umgebung	Umgebung	✓	–	+	−−	[98]
Reflektoren	Reflexion	–	✓	+	++	[79, 84]
Platte	Schachbrett	–	✓	−	++	[76, 81, 132]
Rechtw. Dreieck	Geometrie	✓	✓	−	−	[38]
Spez. Geometrie	Farbe	–	✓	−	−	[56]
Umgebung	Umgebung	–	✓	+	++	[74]
Platte	Farbe	–	✓	++	++	[P3]

Bei dem in [79] präsentierten Ansatz kommen kleine Spiegel zum Einsatz und die Erkennung in der Laseraufnahme erfolgt anhand der Intensität des reflektierten Lasers. Die meisten der Kalibrierverfahren verwenden Ebenen um Bezugspunkte zu bestimmen. In [132], [76] und [81] wird mit Hilfe eines Schachbrettmusters und der Kontur der Platte, auf der es aufgebracht ist, kalibriert. Ein anderer Ansatz verwendt den Rand eines rechtwinkligen Dreiecks [38]. Auch Muster mit geometrischen Erhebungen kommen zum Einsatz, wie in [56] anhand eines Schachbrettmusters mit Erhebungen gezeigt. Bei entsprechenden Messbereichen, können auch die Wände und Ecken eines Raumes für die Bestimmung von Bezugspunkten verwendet werden [74, 98]. Für die extrinsische Kalibrierung wird im Rahmen des dieser Arbeit übergeordneten Forschungsprojektes ein adaptierter Ansatz aus den zuvor beschriebenen Kalibrierverfahren abgleitet [P3]. Bei diesem können jegliche planare Objekte, die in den Messbereich der Sensoren passen, als Kalibrierobjekt verwendet werden. Die genaue Größe oder Farbe des Objektes ist nicht relevant. Als Bezugspunkte werden die Schnittpunkte der Profillinie mit den Kanten der planaren Fläche verwendet. Die mittlere Abweichung

dieser Kalibrierung liegt unterhalb der Auflösung der verwendeten Kamera und ist somit besser als die zuvor aufgezeigten Kalibrierverfahren.

5.2 Scanvorgang

Beim kooperativen Scanvorgang können sowohl der Sensor als auch das Werkstück bewegt werden. Die Daten müssen anhand der Pose des jeweiligen Roboters transformiert werden (siehe Bild 47). Die mit dem Lasersensor erzeugten Punktwolken enthalten lediglich geometrische Informationen, das bedeutet Punkte mit jeweils einer x-, y- und z-Koordinate. Um die Werkstücke noch realitätsnäher abbilden zu können, kann das Scansystem um weitere Sensoren ergänzt werden. So lassen sich unter anderem Farb- [3] oder Temperaturinformationen [65] zu den Punkten hinzufügen. Für die Erweiterung von Lasersensormesswerten mit Farbinformationen werden üblicherweise Kameras verwendet. Auch für das im Rahmen dieser Arbeit entwickelte 3D-Scansystem wird eine Erweiterung um Farbinformationen unter Verwendung einer Kamera entwickelt [P3], die optional verwendet werden kann (siehe Bild 47). Dies ermöglicht das Erkennen von optischen

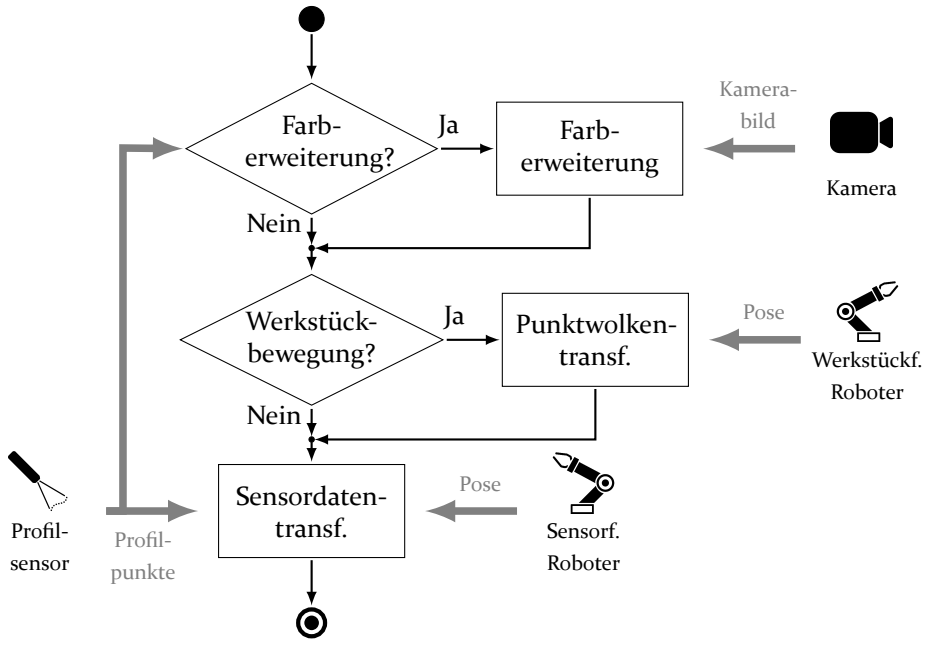


Bild 47: Ablaufübersicht für den Scanvorgang mittels kooperierenden Industrierobotern inklusive optionaler Farbwerterweiterung

Defekten am Werkstück und das erzeugte Modell ist zudem realitätsnäher, was die Benutzerfreundlichkeit unterstützt.

Die Bewegungen bei diesem 3D-Scansystem können jeweils unabhängig voneinander oder gleichzeitig erfolgen. Weitergehend werden die drei Bewegungsmöglichkeiten: *Sensorbewegung*, *Werkstückbewegung* und *kooperative Bewegung* beschrieben (siehe Abschnitt 5.2.1 bis Abschnitt 5.2.3). Der Ansatz für die *Farbwerterweiterung* wird im Rahmen dieses Abschnittes ebenfalls erläutert (siehe Abschnitt 5.2.4).

5.2.1 Sensorbewegung

Wenn ausschließlich der Sensor bewegt wird, spielt der werkstückführende Roboter keine Rolle (siehe Bild 48). Hierbei müssen die N_L Profilpunkte des Lasersensors $\mathbf{l} = (x_\mathrm{L}\ 0\ z_\mathrm{L})^\mathrm{T}$ mit Hilfe der Flanschpose des sensorführenden Roboters $\{F_\mathrm{S}\}$ in das Ursprungskoordinatensystem $\{O\}$ transformiert werden. Unter Verwendung der durch Kalibrierung bestimmten Transformation des Sensors ${}^{\mathrm{F_S}}\mathbf{T}_\mathrm{S}$ können die N_L auf den Ursprung bezogenen Profilpunkte $\mathbf{p} = (x_\mathrm{P}\ y_\mathrm{P}\ z_\mathrm{P})^\mathrm{T}$ ermittelt werden durch:

$$\begin{pmatrix}\mathbf{p}_j\\1\end{pmatrix} = {}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_S}} \cdot {}^{\mathrm{B_S}}\mathbf{T}_{\mathrm{F}_{\mathrm{S}i}} \cdot {}^{\mathrm{F_S}}\mathbf{T}_\mathrm{S} \cdot \begin{pmatrix}\mathbf{l}_j\\1\end{pmatrix}, \qquad (5.2)$$

$$i \in \{1,\ 2,\ \ldots,\ N_\mathrm{M}\}\,, \quad j \in \{1,\ 2,\ \ldots,\ N_\mathrm{M} \cdot N_\mathrm{L}\}$$

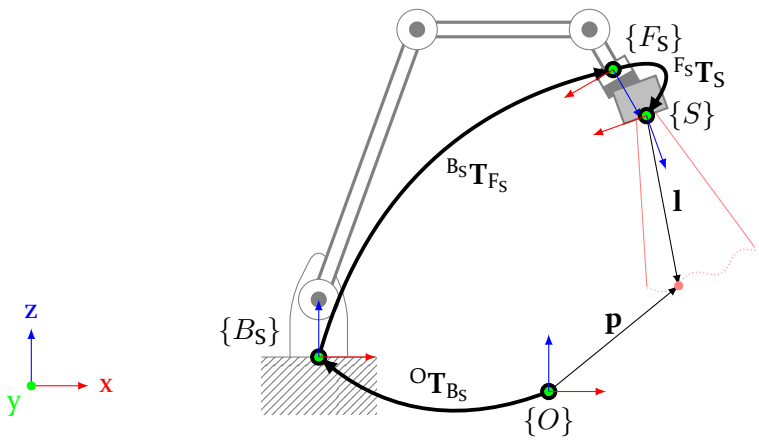


Bild 48: Koordinatensysteme und Transformationen für den Scanvorgang am sensorführenden Roboter

Diese Berechnung wird für jede der N_M Roboterposen wiederholt und anschließend können die N_L Profilpunkte den Daten der Punktwolke hinzugefügt werden.

5.2.2 Werkstückbewegung

Bei einer reinen Werkstückbewegung muss die bereits vorhandene Punktwolke genauso wie das reale Werkstück bewegt werden (siehe Bild 49). Hierfür werden die Punkte $\mathbf{p}$ zunächst in das Ursprungskoordinatensystem $\{O\}$ und dann entsprechend der Bewegung des werkstückführenden Roboters ${}^{\mathrm{F_W}}\mathbf{T}_\Delta$ transformiert. Anschließend werden sie wieder zurück in das Flanschkoordinatensystem des Roboters $\{F_W\}$ transformiert. Die Transformationskette lässt sich in einer Formel zusammenfassen durch:

$$\begin{pmatrix}\mathbf{p}'_j\\1\end{pmatrix} = \left({}^{\mathrm{B_W}}\mathbf{T}_{\mathrm{F_W}i}\right)^{-1} \cdot \left({}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_W}}\right)^{-1} \cdot {}^{\mathrm{F_W}}\mathbf{T}_{\Delta_i} \cdot {}^{\mathrm{O}}\mathbf{T}_{\mathrm{B_W}} \cdot {}^{\mathrm{B_W}}\mathbf{T}_{\mathrm{F_W}i} \cdot \begin{pmatrix}\mathbf{p}_j\\1\end{pmatrix}, \tag{5.3}$$
$$i \in \{1,\ 2,\ \ldots,\ N_\mathrm{M}\}\,, \quad j \in \{1,\ 2,\ \ldots,\ N_\mathrm{M} \cdot N_\mathrm{L}\}$$

Nach der Transformation der bereits vorhandenen Punktdaten können die neuen Profilpunkte gemäß Gleichung 5.2 berechnet und hinzugefügt werden.

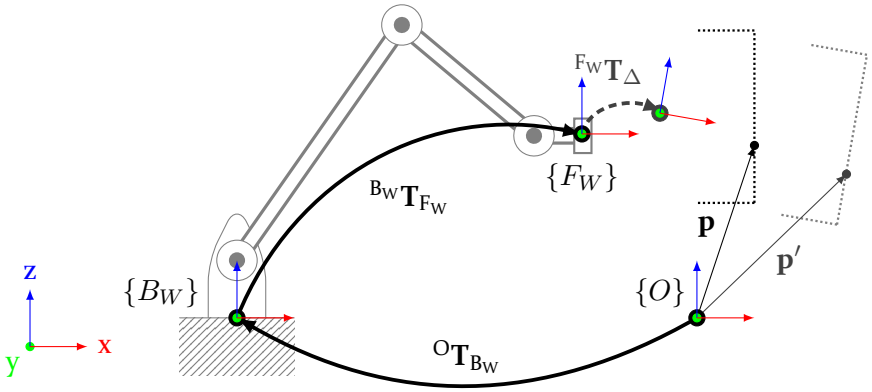


Bild 49: Koordinatensysteme und Transformationen für den Scanvorgang am werkstückführenden Roboter

5.2.3 Kooperative Bewegung

Die Abläufe beim kooperativen Scanvorgang sind vergleichbar mit denen der reinen Werkstückbewegung. Zunächst werden die bereits vorhandenen Punktdaten gemäß Gleichung 5.3, gleichermaßen wie das reale Werkstück bewegt wird, transformiert. Anschließend werden die neuen Profilpunkte gemäß Gleichung 5.2 und unter Verwendung der neuen Pose des sensorführenden Roboters berechnet und den Daten der Punktwolke hinzugefügt.

5.2.4 Farbwerterweiterung

Die entwickelte Routine für die Erweiterung der Lasersensormesswerte mit den Farbinformationen aus einem Kamerabild setzt sich aus vier zentralen Prozessschritten zusammen (siehe Bild 50). Zunächst sind zwei Kalibrierungen notwendig (siehe Abschnitt 5.1.3). Anschließend kann

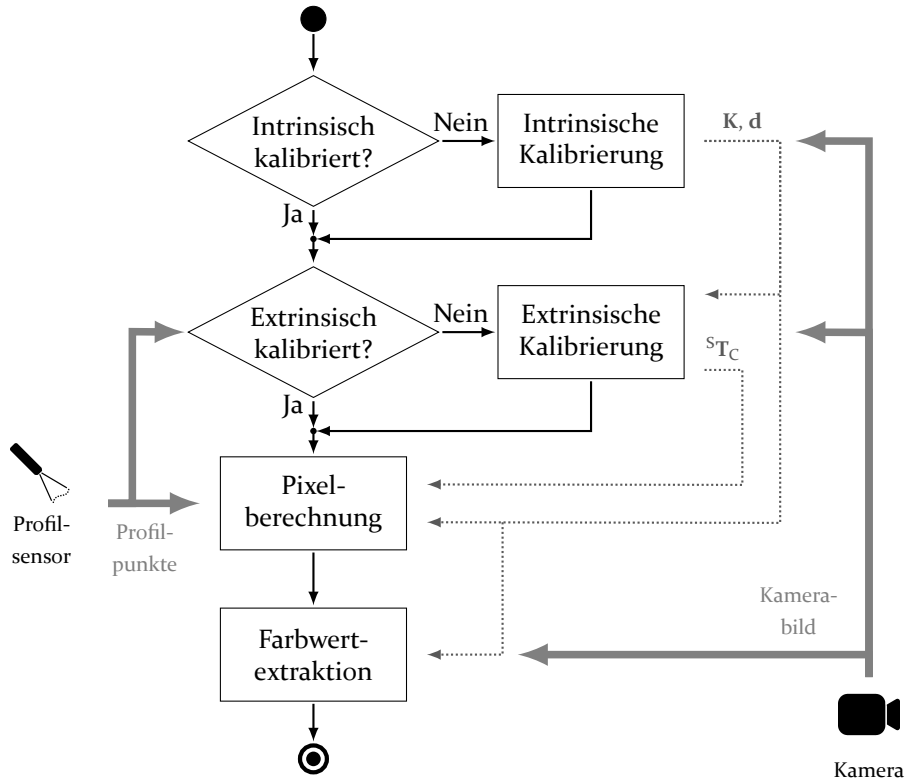


Bild 50: Ablaufübersicht für die Erweiterung der Lasersensormesswerte mit Farbinformationen

unter Verwendung der durch die Kalibrierung bestimmten Parameter eine *Pixelberechnung* erfolgen, bei der für jeden Profilpunkt ein zugehöriger Kamerapixel berechnet wird. Aus diesem Pixel kann letztlich durch *Farbwertextraktion* der korrespondierende Farbwert gewonnen werden. Weitergehend werden die zwei Verarbeitungsschritte ausführlicher beschrieben.

Pixelberechnung

Mit der durch Kalibrierung ermittelten Transformation zwischen der Basis des Lasersensors und der Kamera ${}^{\mathrm{S}}\mathbf{T}_{\mathrm{C}}$ kann nun für jeden der N_{L} Sensormesspunkte $\mathbf{l} = (x_{\mathrm{L}}\ 0\ z_{\mathrm{L}})^{\mathrm{T}}$ ein zugehöriger Kamerabildpunkt $\mathbf{c} = (x_{\mathrm{C}}\ y_{\mathrm{C}})^{\mathrm{T}}$ berechnet werden durch:

$$\begin{pmatrix} \mathbf{c}_i \\ 0 \\ 1 \end{pmatrix} = {}^{\mathrm{S}}\mathbf{T}_{\mathrm{C}} \cdot \begin{pmatrix} \mathbf{l}_i \\ 1 \end{pmatrix}, \quad i \in \{1,\ 2,\ \ldots,\ N_{\mathrm{L}}\} \tag{5.4}$$

Der resultierende Kamerabildpunkt entspricht der Projektion des Sensormesspunktes auf die Bildebene.

Farbwertextraktion

Für die Extraktion eines zugehörigen Farbwertes aus dem Kamerabild gibt es mehrere Ansätze. Da der hier beschriebene Ansatz für Laser-Profilsensoren mit sichtbarer Laserlinie konzipiert ist und somit das Objekt an den Messpunkten durch den Laser verfärbt wird, ist ein direktes Entnehmen aus dem Kamerabild nicht zielführend. Eine mögliche Lösung ist das Ausschalten des Lasers für diesen Prozessschritt, was jedoch einen erhöhten Zeitaufwand bedeutet. Alternativ könnte die gesamte Punktwolke erst im Anschluss an den Scanvorgang eingefärbt werden. Hierfür könnten jedoch mehrere Aufnahmen notwendig sein, um alle Seiten des Objektes abzudecken. Um geeignete Farbwerte bereits während des Scanvorganges zu ermitteln, wird ein Ansatz entwickelt, der die Farbwerte anhand der benachbarten Pixel näherungsweise ermittelt. Hierfür werden für jeden der N_{L} Sensormesspunkte zwei in y-Richtung benachbarte Punkte $\mathbf{p} = (x_{\mathrm{L}}\ \pm\delta\ z_{\mathrm{L}})^{\mathrm{T}}$ in einem variablen Abstand von $\pm\delta$ verwendet, um zugehörige Bildpunkte $\mathbf{c}$ zu berechnen gemäß:

$$\begin{pmatrix} \mathbf{c}_{j_i} \\ 0 \\ 1 \end{pmatrix} = {}^{\mathrm{S}}\mathbf{T}_{\mathrm{C}} \cdot \begin{pmatrix} \mathbf{p}_{j_i} \\ 1 \end{pmatrix}, \quad i \in \{1,\ 2,\ \ldots,\ N_{\mathrm{L}}\}, \quad j \in \{1,\ 2\} \tag{5.5}$$

Der Versatz kann an die Breite der Laserlinie angepasst werden. Der zugehörige Farbwert kann weitergehend durch den Mittelwerte der beiden benachbarten Farbwerte angenähert werden.

5.3 Verarbeitung der erzeugten Daten

Mit dem zuvor beschriebenen 3D-Scansystem können eine Vielzahl von Punkten auf der Werkstückoberfläche erfasst werden. Als Grundlage für die werkstückbezogene Bahndefinition werden jedoch üblicherweise Flächenmodelle verwendet. Daher müssen die erfassten Punktwolken weitergehend bearbeitet werden, mit dem Ziel ein auf den relevanten Teil reduziertes Flächenmodell des Werkstückes zu erzeugen. Hierzu wird die Punktwolke durch *Fusion* (siehe Abschnitt 5.3.1) und *Zuschnitt* (siehe Abschnitt 5.3.2) zunächst auf die relevanten Daten reduziert. Anschließend kann basierend auf der reduzierten Punktwolke ein *Flächenmodell* erzeugt werden (siehe Abschnitt 5.3.3).

5.3.1 Reduzierung der Punktdaten durch Fusion

In den erzeugten Punktwolken kann es vorkommen, dass mehrere Punkte nahe zueinander liegen. Der Gewinn an Informationen fällt dabei minimal aus, wohingegen die Dateigröße zunimmt. Deshalb ist es sinnvoll Punkte, die unmittelbar benachbart sind, zusammenzufassen. Dies kann durch Verwendung des Mittelwertes aller Punkte innerhalb eines Voxels erfolgen gemäß:

$$\overline{\mathbf{p}}_j = \frac{\sum X \cap Y \cap Z}{|X \cap Y \cap Z|} \quad \text{mit} \quad \begin{aligned} X &= \{\mathbf{p}_i \mid \mathbf{p}_i \to x_{\nu_{\min}} < x_{\mathrm{P}} < x_{\nu_{\max}}\} \\ Y &= \{\mathbf{p}_i \mid \mathbf{p}_i \to y_{\nu_{\min}} < y_{\mathrm{P}} < y_{\nu_{\max}}\} \\ Z &= \{\mathbf{p}_i \mid \mathbf{p}_i \to z_{\nu_{\min}} < z_{\mathrm{P}} < z_{\nu_{\max}}\} \\ i &\in \{1,\ 2,\ \ldots,\ N_{\mathrm{P}}\} \\ j &\in \{1,\ 2,\ \ldots,\ N_{\nu}\} \end{aligned} \tag{5.6}$$

Dabei werden alle der N_{P} Punkte, die innerhalb eines der N_{ν} Voxel liegen, zusammengefasst. Die Auflösung der Fusion und somit von der resultierenden Punktwolke hängt von der gewählten Voxelgröße, aus der sich die Grenzwerte $x_{\nu_{\min}}$, $x_{\nu_{\max}}$, $y_{\nu_{\min}}$, $y_{\nu_{\max}}$, $z_{\nu_{\min}}$ sowie $z_{\nu_{\max}}$ ergeben, ab.

5.3.2 Reduzierung der Punktdaten durch Zuschnitt

Außerdem können gewisse Bereiche der erstellten Punktwolke nicht zugehörig zum Werkstück bzw. nicht von Interesse sein. Um auch in diesem Fall die Datenmenge so gering wie möglich zu halten, kann eine Volumenreduzierung gemäß der folgenden Bedingung erfolgen:

$$\mathcal{P}^p = X \cap Y \cap Z \quad \text{mit} \quad \begin{aligned} X &= \{\mathbf{p}_i \mid \mathbf{p}_i \rightarrow x_{\text{min}} < x_{\text{P}} < x_{\text{max}}\} \\ Y &= \{\mathbf{p}_i \mid \mathbf{p}_i \rightarrow y_{\text{min}} < y_{\text{P}} < y_{\text{max}}\} \\ Z &= \{\mathbf{p}_i \mid \mathbf{p}_i \rightarrow z_{\text{min}} < z_{\text{P}} < z_{\text{max}}\} \\ i &\in \{1,\ 2,\ \ldots,\ N_{\text{P}}\} \end{aligned} \tag{5.7}$$

Somit werden lediglich diejenigen der N_{P} Punkte verwendet, die innerhalb des durch die Grenzwerte x_{min}, x_{max}, y_{min}, y_{max}, z_{min} sowie z_{max} aufgespannt Quaders liegen.

5.3.3 Erzeugen von Flächenmodellen

Ein weit verbreitetes Datei-Format für Flächenmodelle ist das *STereoLithography* (STL)-Format, das in den meisten CAD-Systemen zur Verfügung steht, weshalb es weitergehend verwendet wird. Das Format beschreibt die Oberflächen der Modelle durch Dreiecksfacetten, die jeweils durch drei Eckpunkte sowie eine Flächennormale beschrieben sind. Die durch den Scan erzeugten Punktwolken müssen daher weitergehend vernetzt werden, um die Oberflächen dementsprechend abzubilden. Für die Vernetzung von Punktwolken existieren gegenwärtig mehrere etablierte Ansätze. Im Rahmen dieser Arbeit wird der Ansatz für die schnelle Vernetzung von ungeordneten Punktwolken aus [72] verwendet. Hierbei erfolgt das Vernetzen durch ein inkrementelles Wachstum der Oberfläche und eine Suche nach dem nächsten Nachbar mittels k-d-Baum.

5.4 Zusammenfassung

Anhand des zuvor beschriebenen Ansatzes für eine Werkstückerfassung mittels kooperierenden Industrierobotern kann bei der Erfassung sowohl das Werkstück als auch der Sensor bewegt werden, unabhängig davon ob die Bewegungen synchron oder asynchron erfolgen. Zudem ist eine Erweiterung des Scansystems um Farbinformationen beschrieben, welche eine realitätsnähere Abbildung der Werkstücke ermöglicht. Bei der Realisierung

eines derartigen Scansystems ist es erforderlich, dass die beteiligten Komponenten – alle Roboter und Sensoren – miteinander vernetzt sind, sodass die Daten verknüpft werden können. Nachdem bei der Umsetzung von Anwendungen aus den Sensordaten gemäß des Ansatzes Punktwolken erzeugt worden sind, kann es erforderlich sein, dass diese weiterverarbeitet werden müssen. Die im Rahmen dieses Kapitels beschriebenen Funktionen bieten die Möglichkeit erzeugte Punktwolken zu bearbeiten und daraus Flächenmodelle abzuleiten.

6 Implementierung

In diesem Kapitel erfolgt die Implementierung der zuvor erarbeiteten Ansätze anhand von realen Robotersystemen. Hierbei werden zwei Robotersysteme verwendet, die sich in mehreren Aspekten unterscheiden (siehe Tabelle 6). Einerseits wird ein Robotersystem, das aus zwei Knickarmrobotern besteht und die *dezentrale Steuerung* als Kommunikationsstruktur (siehe Abschnitt 3.2.2) verwendet, herangezogen. Dieses ist zudem um eine Werkstückerfassung gemäß dem zuvor beschriebenen Ansatz (siehe Kapitel 5) erweitert. Andererseits erfolgt die Implementierung des Aufteilungsansatzes für einen Zweiarmroboter mit *zentraler Steuerung*. Ein Vorteil hierbei ist, dass die beiden Kinematiken bereits werksseitig auf einer gemeinsamen Plattform befestigt und somit zueinander kalibriert sind.

Die softwaretechnische Implementierung für das erstgenannte System erfolgt innerhalb einer selbstentwickelten Offline-Programmiersoftware, da hiermit die für die Werkstückerfassung notwendigen Schnittstellen sowie tiefgreifendere Untersuchungen möglich sind. Für das zweite System werden die Ansätze unmittelbar in die herstellerspezifische Offline-Programmiersoftware integriert, um zu verdeutlichen, dass der erarbeitete Aufteilungsansatz ohne großen Mehraufwand mit einer Vielzahl von auf dem Markt verfügbaren Simulationstools umgesetzt werden kann. Mit beiden Robotersystemen erfolgt die anwendungsbezogene Programmierung gemäß eines Ablaufes, der aus drei zentralen Teilschritten besteht (siehe

Tabelle 6: Übersicht der für die Implementierung verwendeten Robotersysteme

Kriterium	1. Robotersystem	2. Robotersystem
Roboterhersteller	KUKA Roboter GmbH	ABB Automation GmbH
Robotertyp	2 × Knickarmroboter	1 × Zweiarmroboter
Roboterachsen	2 × 6 rot. Achsen	2 × 7 rot. Achsen
Kommunikationsstruktur	Dezentrale Steuerung	Zentrale Steuerung
Vorkalibriert	–	✓
Werkstückerfassung	✓	–
Offline-Programmiersoftware	Eigenentwicklung	Proprietär

Bild 51). Hierbei erfolgt zunächst die *Bahndefinition* von werkstückbezogenen Bearbeitungsbahnen inklusive der Konfiguration von Prozessparametern und Randbedingungen durch den Benutzer. Diese wird üblicherweise auf Grundlage von Flächenmodellen der Werkstücke durchgeführt. Anschließend wird die *Bahnaufteilung* in eine Werkstück- und eine Werkzeugbewegung gemäß des entwickelten Ansatzes durchgeführt, die entsprechende Bahnen für die beteiligten Industrieroboter ausgibt. Die *Programmgenerierung und -überprüfung* schließt den Ablauf mit der Ausgabe der Roboterprogramme ab, sodass diese am realen Robotersystem ausgeführt werden können.

Weitergehend werden die beiden Robotersysteme ausführlicher erläutert (siehe Abschnitt 6.1 und Abschnitt 6.2). Dabei wird jeweils der Anlagenaufbau und die Softwareimplementierung betrachtet.

6.1 Robotersystem mit dezentraler Steuerung erweitert um Werkstückerfassung

Im Rahmen dieses Abschnittes wird zunächst die für die Umsetzung gewählte Hardware des ersten Robotersystems beschrieben (siehe Abschnitt 6.1.1). Der Fokus liegt dabei auf den zentralen Komponenten der Anlage sowie deren Vernetzung. Weitergehend wird die entwickelte

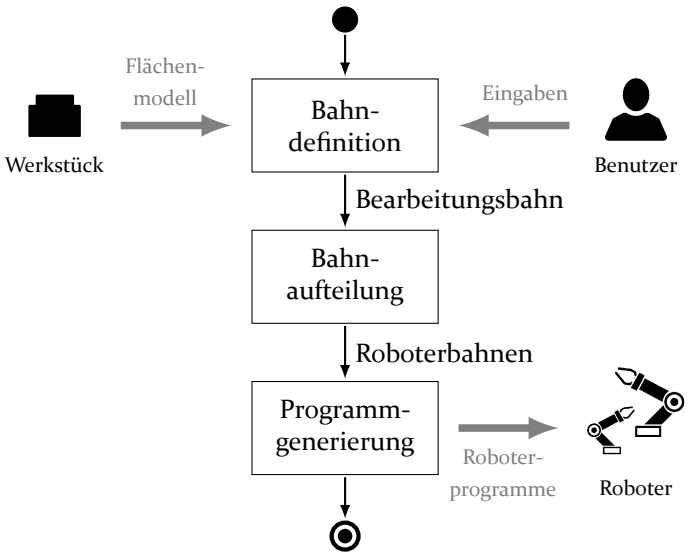


Bild 51: Ablauf der Roboterprogrammierung für die kooperative Bearbeitung

Offline-Programmiersoftware dargestellt (siehe Abschnitt 6.1.2). Dabei erfolgt eine Betrachtung der drei für die Roboterprogrammierung vorgesehenen Ablaufschritte (siehe Bild 51). Zudem wird die Werkstückerfassung mit diesem System erläutert.

6.1.1 Anlagenaufbau

Bei der Realisierung der Roboterzelle wird auf eine hohe Flexibilität geachtet, um diese mit möglichst wenig Aufwand an neue Werkstücke und Prozesse anpassen zu können. Im Zentrum der Anlage stehen zwei *Knickarmroboter* mit jeweils sechs rotatorischen Achsen (siehe Bild 52). Diese sind jeweils auf einem Profiltisch montiert. Die beiden Tische wiederum sind miteinander verbunden und am Boden der Zelle fixiert. Durch die Profiloberfläche der Tische können Komponenten flexibel darauf positioniert werden. Damit auch die in der Anlage eingesetzten Endeffektoren (z. B. Greifer oder Sensorplattform) schnell gewechselt werden können, wird ein *Werkzeugwechselsystem* an beiden Roboterarmen eingesetzt. Die entsprechende Werkzeughalterung ist zwischen den zwei Robotern an beiden Tischen montiert, sodass die vorhandenen Endeffektoren von beiden Robotern aufgenommen werden können. Für die Umsetzung des zuvor beschriebenen kooperativen 3D-Scansystems (siehe Kapitel 5) werden zwei Sensoren benötigt. Einerseits wird ein *Laser-Profilsensor* für das Erfassen der Werkstückgeometrie eingesetzt.

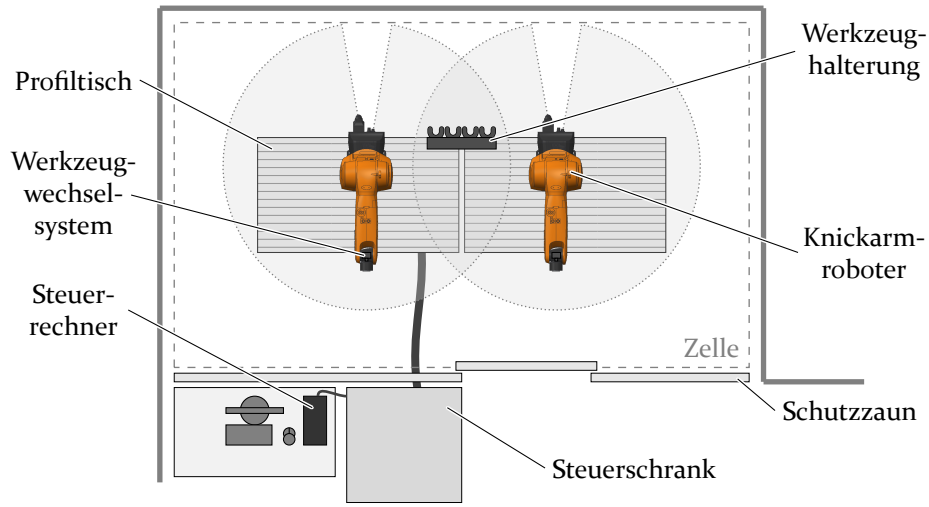
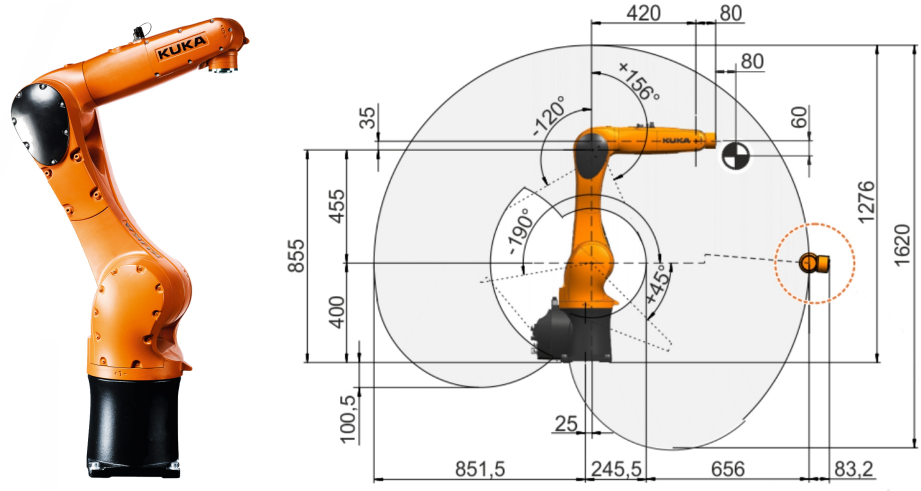


Bild 52: Anlagenaufbau des ersten Robotersystems für die Implementierung

Andererseits wird eine *Kamera* für die Erweiterung des Systems um Farbinformationen benötigt. Gegenüber dem Bediener ist die Zelle durch einen Schutzzaun abgetrennt, um ungewollten Kontakt zu vermeiden. Außerhalb des Zaunes befindet sich der Steuerschrank, in dem die Steuerungen und weitere Komponenten für die *Vernetzung* der Roboterzelle enthalten sind. Daneben befindet sich der Steuerrechner für die Ansteuerung einzelner Zellkomponenten und die Bedienung der Software. Weitergehend werden die zentralen Bestandteile der Anlage und deren Vernetzung detaillierter beschrieben.

Knickarmroboter

Für die Implementierung werden zwei Knickarmroboter vom Typ *KR 6 R900 sixx* der *KUKA Roboter GmbH* (siehe Bild 53) verwendet. Dieser besitzt sechs rotatorische Achsen und hat eine gesteckte Armlänge von 901 mm [58, 61]. Der Hersteller gibt eine Positionswiederholgenauigkeit von $\pm 0,03$ mm sowie eine maximale Traglast von 6 kg an. Basierend auf den Herstellerangaben können die DH-Parameter gemäß der DH-Konvention (siehe Abschnitt 2.2) für die $N_A = 6$ Achsen aufgestellt werden (siehe Tabelle 7). Angesteuert werden die Roboter jeweils über eine zugehörige Steuerung vom Typ *KR C4 compact*. Auf diesen sind die Funktions- und Technologiepakete *KUKA.EtherNet KRL* und *KUKA.RoboTeam* installiert.



(a) Roboter [61] (b) Arbeitsbereich [58]

Bild 53: Knickarmroboter KR 6 R900 sixx der KUKA Roboter GmbH

Tabelle 7: DH-Parameter des Knickarmroboters KR 6 R900 sixx der KUKA Roboter GmbH

Achse i	θ_i in rad	d_i in mm	a_i in mm	α_i in rad
1	θ_1	400	25	$\pi/2$
2	θ_2	0	455	0
3	$\theta_3 - \pi/2$	0	35	$\pi/2$
4	θ_4	420	0	$-\pi/2$
5	θ_5	0	0	$\pi/2$
6	$\theta_6 + \pi$	80	0	0

Das erste Paket dient zur Kommunikation mit dem Steuerrechner, das zweite zur Kommunikation zwischen den beiden Roboterarmen, wobei die Kommunikationsstruktur der dezentralen Steuerung (siehe Abschnitt 3.2) verwendet wird. Für die Synchronisation von Programmpunkten ist die Funktion *ProgSync* und für die Synchronisation mit überschliffenen Bahnpunkten die Funktion *MotionSync* enthalten.

Werkzeugwechselsystem

Da an dieser Anlage eine Vielzahl unterschiedlicher Anwendungen umgesetzt werden, wird sie mit einem Werkzeugwechselsystem ausgestattet. An beiden Roboterarmen kommt das Werkzeugwechselsystem *SWS-005* der *SCHUNK GmbH & Co. KG* zum Einsatz (siehe Bild 54), welches einen Wechsel der Werkzeuge bzw. Sensoren ohne manuellen



Bild 54: Werkzeugwechselsystem SWS-005 der SCHUNK GmbH & Co. KG [99]

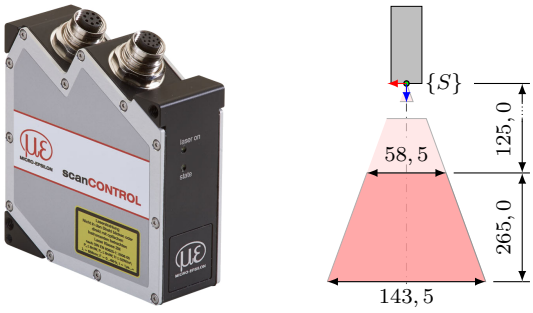
Eingriff ermöglicht. Es besteht jeweils aus einem Kopfteil auf der Seite des Roboters und einem Adapterteil auf der Seite des Endeffektors. Die an den Endeffektoren verwendeten Daten- und Druckluftleitungen werden über Kopplungen durch das Werkzeugwechselsystem geführt und können somit ebenfalls abgekoppelt werden. Betätigt wird die Kopplung des Werkzeugwechselsystems per Druckluft, die in diesem Fall über ein im jeweiligen Roboter integriertes Ventil gesteuert wird.

Laser-Profilsensor

Aufgrund seiner kompakten Bauweise wird der Sensor *scanCONTROL 2600-100* der *MICRO-EPSILON MESSTECHNIK GmbH & Co. KG* (siehe Bild 55a) als Laser-Profilsensor gewählt. Dieser misst 640 Punkte pro Profil mit einer Profilfrequenz von 200 Hz und einer Referenzauflösung von 12 µm [75]. Der Messbereich des Sensors (siehe Bild 55b) reicht von einer Entfernung von $125,0$ mm bis zu $390,0$ mm, wobei er sich von einer Breite von $58,5$ mm auf $143,5$ mm ausweitet.

Kamera

Des Weiteren wird im 3D-Scansystem eine Kamera eingesetzt, um die erzeugten 3D-Modelle mit Farbinformationen zu ergänzen (siehe Abschnitt 5.2.4). Zum Einsatz kommt die Kamera *Logitech HD Pro Webcam C920* (siehe Bild 56) mit einer Auflösung von 1920×1080 Pixeln bei 30 Bildern pro Sekunde. Diese enthält eine Autofokus-Funktion, weshalb ein manuelles Einstellen des Fokus nicht notwendig ist.



(a) Sensor (b) Messbereich

Bild 55: Laser-Profilsensor scanCONTROL 2600-100 der MICRO-EPSILON MESSTECHNIK GmbH & Co. KG [75]



Bild 56: HD Pro Webcam C920 von Logitech [68]

Vernetzung

Zentraler Knotenpunkt der Vernetzung innerhalb der Anlage ist ein Ethernet-Switch (siehe Bild 57). Hieran angeschlossen sind der Steuerrechner, der Profilsensor via *Power Over Ethernet* (POE)-Modul sowie die beiden Robotersteuerungen. Jeder der beiden Roboter ist mit einer Datenleitung und einer Steuerungsleitung an die jeweilige Robotersteuerung angeschlossen und wird mit Druckluft über einen in die Zelle integrierten Druckluftanschluss versorgt. Außerdem sind die beiden Robotersteuerungen via *EtherCAT* miteinander verbunden, um miteinander kommunizieren zu können. Der Profilsensor verwendet Ethernet für die Datenübertragung und ein zwischengeschaltetes POE-Modul für Spannungsversorgung mit 24 V. Die integrierte Kamera ist via *Universal Serial Bus* (USB) für Datenübertragung und Spannungsversorgung an den Steuerrechner angeschlossen.

6.1.2 Softwareimplementierung

Da der Einsatz von CAD-basierter Offline-Roboterprogrammierung immer mehr zunimmt, gibt es mittlerweile eine Vielzahl von proprietären Programmen für die Offline-Programmierung von Industrierobotern, wie

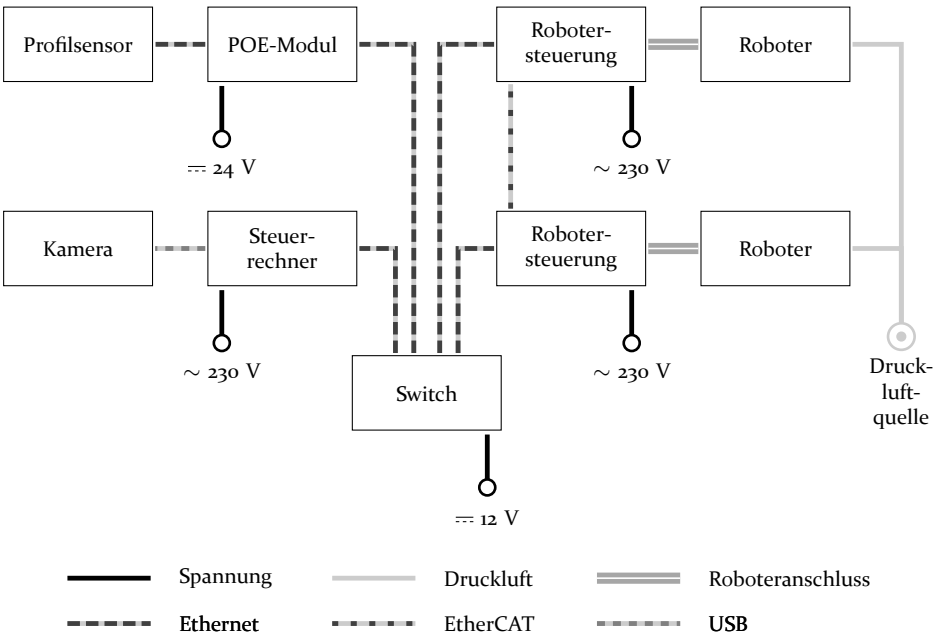


Bild 57: Übersicht der Verbindungen innerhalb des ersten Robotersystems

z. B. *Visual Motion Planner* von *Arotec* [9] oder *RobotMotionCenter* von *Blackbird Robotersysteme* [14]. In den verfügbaren Programmen sind jedoch nur begrenzte Schnittstellen vorhanden und sie können nicht flexibel um beliebige Softwarekomponenten erweitert werden. Daher wird im Rahmen dieser Arbeit eine eigene Offline-Programmiersoftware für Industrieroboter entwickelt, in der alle für das Gesamtsystem vorgesehenen Ablaufschritte (siehe Bild 51) sowie die für die Werkstückerfassung relevanten Komponenten implementiert werden. Im Zentrum der Softwareoberfläche steht die 3D-Visualisierung (siehe Bild 58). Über eine Menüleiste auf der rechten Seite kann eines der fünf Hardware-Interfaces (Roboter, Sensor, Werkstück, Dosiergerät und Kamera) geöffnet werden. Die Software wird basierend auf dem *.NET Framework* erstellt, da hierfür eine Vielzahl von objektorientierten Klassenbibliotheken zur Verfügung steht. Dabei kommen die Programmiersprachen *C++* und *C#* zum Einsatz. Die Gestaltung der grafischen Benutzeroberfläche erfolgt unter der Verwendung der Beschreibungssprache *Extensible Application Markup Language* (XAML) mittels der Klassenbibliothek *Windows Presentation Foundation* (WPF), einem Grafik-Framework der im *.NET Framework* ab Version 3.0 zur Verfügung steht. Die dreidimensionale Simulationsumgebung basiert auf der freien Klassenbibliothek *Visualization Toolkit* (VTK), die aufgrund der enthaltenen *Frameworks* und *Widgets* besonders für wissenschaftliche 3D-Visualisierung geeignet ist.

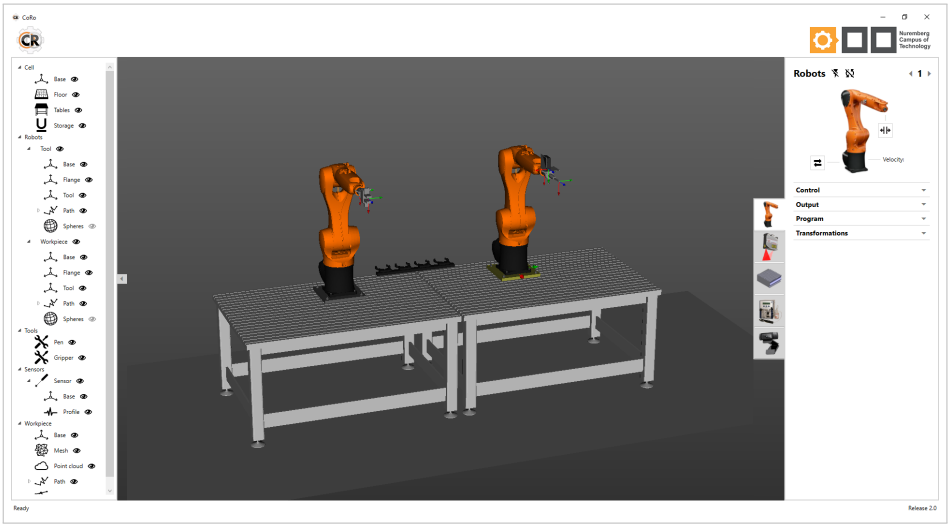


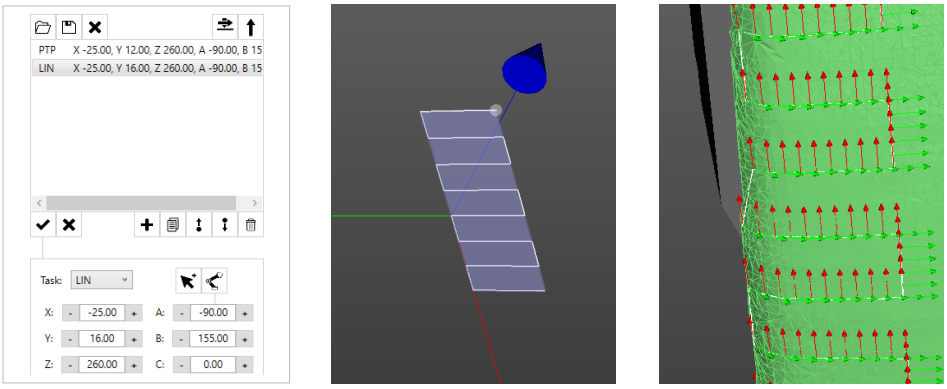
Bild 58: Softwareoberfläche der entwickelten Offline-Programmiersoftware

Die Einbindung in die erstellte Software erfolgt mit Hilfe des *C#-Wrapper ActiViz.Net* in der Version 5.8.0.607.

Nachfolgend wird die Durchführung der drei zentralen Teilschritte des Programmierablaufes: *Bahndefinition*, *Bahnaufteilung* sowie *Programmgenerierung und -überprüfung* mit der entwickelten Software beschrieben. Zudem wird auf den Ablauf für die *Werkstückerfassung* eingegangen.

Bahndefinition

Die Bahndefinition kann über das *Workpiece-Interface* durch den Benutzer vorgenommen werden (siehe Bild 59a). Hierbei ist einerseits eine vereinfachte Eingabe von auf gekrümmte Flächen bezogene Bahnen durch senkrechte Projektion von 2D-Skizzen möglich (siehe Bild 60). Hierbei muss der Benutzer zunächst die Projektionsskizze definieren (siehe Bild 59b). Dies kann durch manuelles Zeichnen oder durch Laden einer Skizze erfolgen. Anschließend kann die Skizzenebene gegenüber dem Werkstückmodell positioniert werden. Dabei wird das resultierende Projektionsergebnis kontinuierlich angezeigt, sodass das Resultat unmittelbar kontrolliert werden kann. Sobald der Benutzer mit dem Projektionsergebnis zufrieden ist, wird die Projektionsbahn als Bearbeitungsbahn übernommen (siehe Bild 59c). Andererseits kann der Bahnverlauf durch manuelle Festlegung von Bahnposen und Bahnverläufen erfolgen. Hierfür kann der Benutzer über das Interface explizite Werte eingeben oder Oberflächenpunkte in der Simulationsumgebung manuell durch Klicken auswählen.



(a) Workpiece-Interface (b) Projektionsskizze (c) Bearbeitungsbahn

Bild 59: Entwickelte Offline-Programmiersoftware während der Bahndefinition

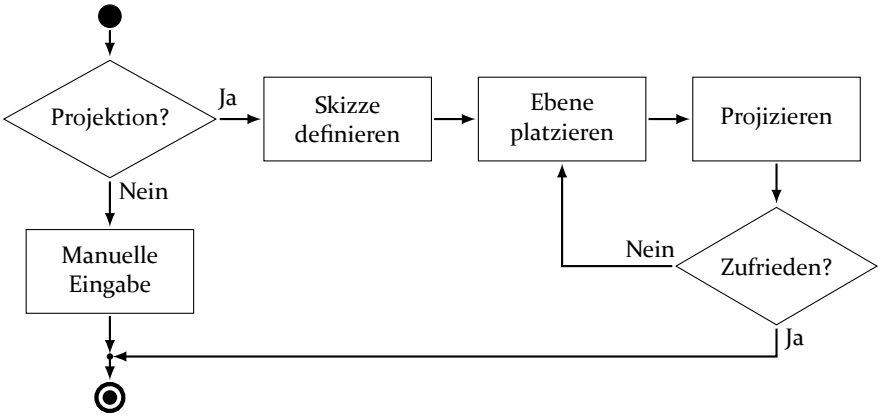


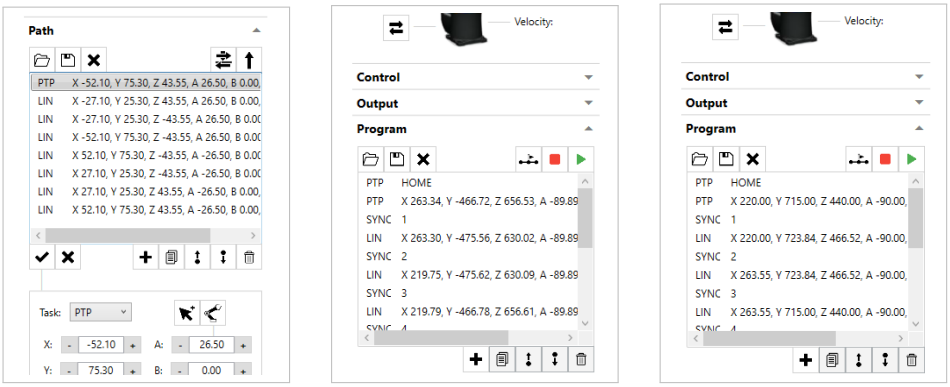
Bild 60: Ablaufplan für die Bahndefinition mit der entwickelten Offline-Programmiersoftware

Bahnaufteilung

Des Weiteren kann die Bahnaufteilung über das *Workpiece-Interface* vorgenommen werden (siehe Bild 61a). Hierbei muss der Benutzer zunächst wählen, welche Art von Bewegung durchgeführt werden soll (siehe Bild 62). Neben der kooperativen Bearbeitung ($0 < s < 1$) stehen auch die Bearbeitung mittels reiner Werkstück- ($s = 0$) und Werkzeugbewegung ($s = 1$) zur Verfügung. Durch Betätigen der Übertragungsschaltfläche wird die zuvor definierte werkstückbezogene Bearbeitungsbahn (siehe Bild 61d) gemäß des entwickelten Ansatzes (siehe Kapitel 4) in Roboterbahnen (siehe Bild 61e und Bild 61f) umgerechnet und den Programmen der jeweiligen Roboter hinzugefügt (siehe Bild 61b und Bild 61c). Hierbei iteriert der entwickelte Algorithmus durch den gemeinsamen Arbeitsraum der beiden Roboter und findet die Startposition mit der geringsten Gesamtzeit für den kooperativen Prozess. Anschließend kann der Benutzer bei Bedarf die Roboterbahnen über das *Robot-Interface* ergänzen bzw. bearbeiten und so beispielsweise nicht-kooperative Programmabschnitte hinzufügen.

Programmgenerierung und -überprüfung

Im Anschluss an die Erzeugung der beidseitigen Roboterbahnen können diese über das *Robot-Interface* (siehe Bild 63a) in Roboterprogramme konvertiert werden. Zuvor sollten das Ausführen der Bahnverläufe jedoch noch in der Simulation auf Kollisionen überprüft werden, um eine kollisionsfreie Ausführung am realen Robotersystem sicherzustellen.



(a) Bearbeitungsbahn-Liste (b) Werkzeugbahn-Liste (c) Werkstückbahn-Liste

(d) Bearbeitungsbahn (e) Werkzeugbahn (f) Werkstückbahn

Bild 61: Entwickelte Offline-Programmiersoftware während der Bahnaufteilung

Hierfür muss zunächst die Kollisionsüberprüfung über das *Robot-Interface* aktiviert werden (siehe Bild 64). Für die kooperative Ausführung der Programme muss des Weiteren die Synchronisation über das Interface aktiviert werden. Bei Bedarf kann die Synchronisation auch deaktiviert sein, um lediglich einzelne Roboter verwenden bzw. überprüfen zu können. Anschließend kann das Ausführen der Programme in der Simulation gestartet werden. Sollten beim Ausführen Kollisionen auftreten, verfärbt sich das Kollisionsmodell an der entsprechenden Stelle rot (siehe Bild 63b) und es wird eine Meldung ausgegeben. Gegebenenfalls müssen die Bahnverläufe bearbeitet oder anhand von neuen Randbedingungen generiert werden, um einen kollisionsfreien Programmablauf zu gewährleisten. Wenn keine Kollision auftritt, können die Programmdateien (siehe Bild 63c) durch einen Algorithmus (siehe

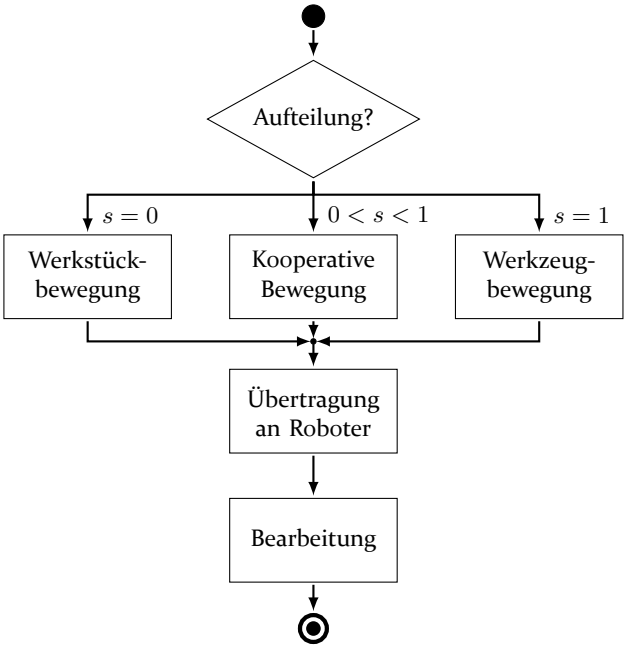
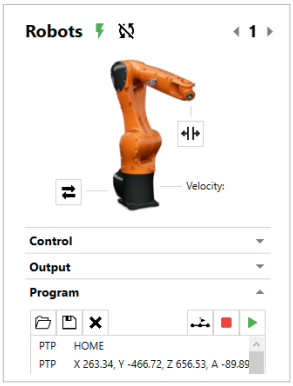
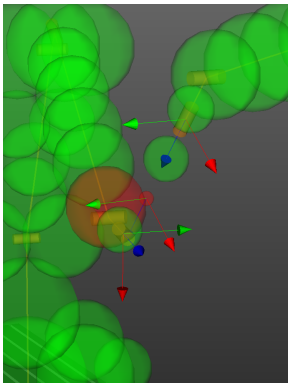


Bild 62: Ablaufplan für die Bahnaufteilung mit der entwickelten Offline-Programmiersoftware

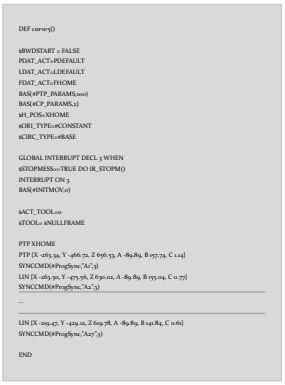
Bild 65) erzeugt werden. Hierbei wird zunächst der Programmkopf geschrieben. In die Programme des werkzeugführenden Roboters wird dabei die Abstimmung mit dem verwendeten Werkzeug integriert. Anschließend



(a) Robot-Interface



(b) Kollision



(c) Programmdatei

Bild 63: Entwickelte Offline-Programmiersoftware während der Programmgenerierung und -überprüfung

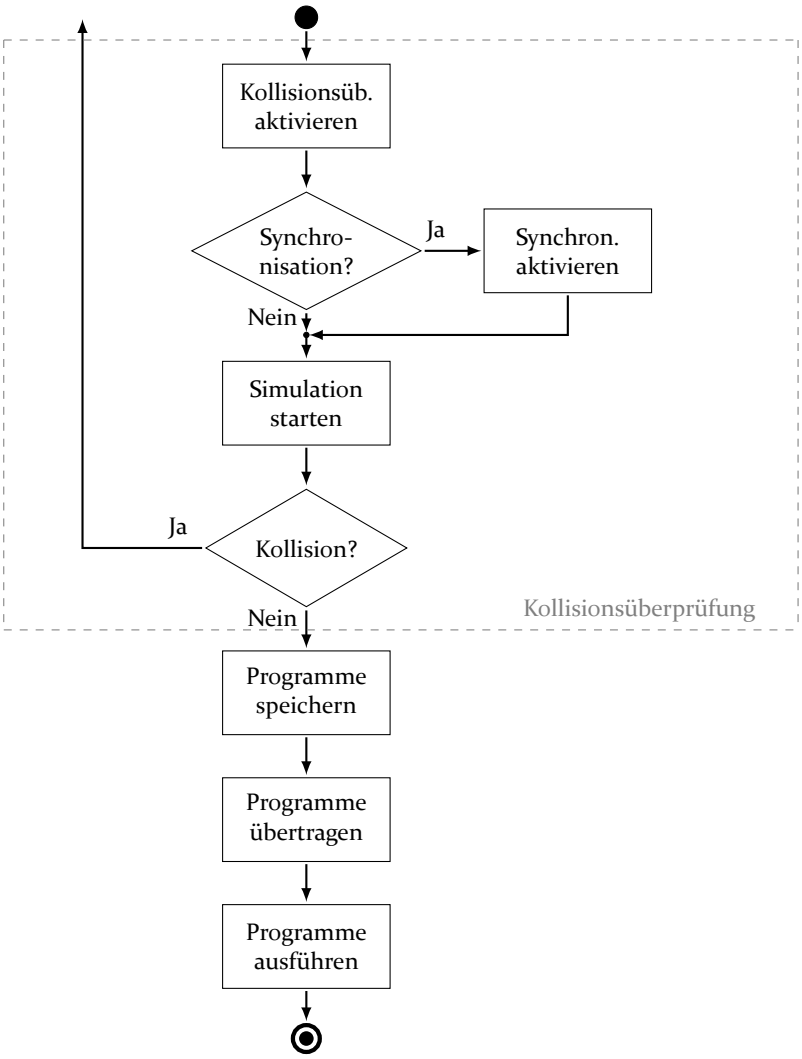


Bild 64: Ablaufplan für die Programmgenerierung und -überprüfung mit der entwickelten Offline-Programmiersoftware

wird für jede der Roboterposen $\mathcal{P}_R$ der zugehörige Bewegungsbefehl erzeugt. Da eine exakte Übereinstimmung der für die Berechnungen verwendeten Kinematikmodelle mit den in den Robotersteuerungen verwendeten Kinematikmodellen nicht gewährleistet werden kann, muss zudem jeweils ein Synchronisationsbefehl hinzugefügt werden, um ein synchrones Ausführen der Bewegungen zu gewährleisten. Letztlich wird der Programmcode noch durch eine Endzeile abgeschlossen. Anschließend

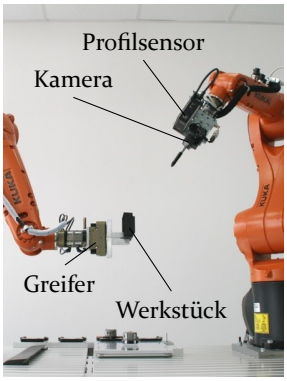
```
1: procedure GeneratePrograM(P_R)
2:     i ← 0
3:     WriteHeader( )
4:     for all p_R ∈ P_R do
5:         WriteSync(i)
6:         WriteMovement(p_R)
7:         i ← i + 1
8:     end for
9:     WriteEnd( )
10: end procedure
```

Bild 65: Algorithmus der Programmgenerierung für die kooperative Bearbeitung mit Multi-Roboter-Systemen

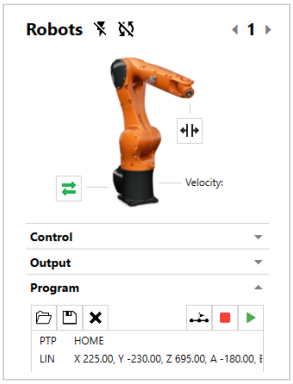
können die Programme auf die jeweilige Robotersteuerung übertragen und auf dem realen Robotersystem ausgeführt werden, sodass der angedachte Prozess ausgeführt wird.

Werkstückerfassung

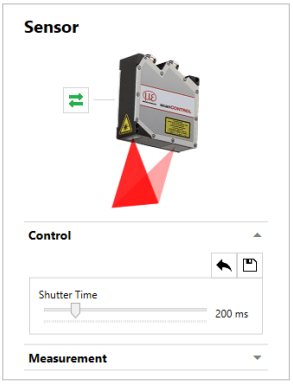
Für die Werkstückerfassung fixiert der werkstückführende Roboter das Werkstück mit einem Greifer und der werkzeugführende Roboter hat den Profilsensor sowie die Kamera an seinem Flansch montiert (siehe Bild 66a). Nachdem das 3D-Scansystem aufgebaut ist, müssen zunächst alle notwendigen Kalibrierungen (siehe Abschnitt 5.1) durchgeführt werden (siehe Bild 67). Hierunter fallen die Roboterkalibrierung, die Sensorkalibrierung sowie die intrinsische und die extrinsische Kamerakalibrierung. Die Kalibrierung der Kamera ist nur notwendig, wenn die Farberweiterung (siehe Abschnitt 5.2.4) verwendet werden soll. Das Ausführen der einzelnen Kalibrierungen kann über das jeweilige Interface ausgelöst werden. Vor dem Ausführen eines Scanvorgangs müssen alle benötigten Komponenten, sprich die Roboter, der Sensor und ggf. die Kamera, mit dem Steuerrechner verbunden werden. Die einzelnen Verbindungen können über das jeweilige Interface aufgebaut werden (siehe Bild 66b-66d). Hierfür enthalten diese eine Verbindungsschaltfläche, die durch zwei gegenläufige Pfeile symbolisiert ist und der Verbindungsstatus wird anhand der Farbe der Pfeile visualisiert. Vor dem Scanvorgang können des Weiteren noch Einstellungen an den beteiligten Komponenten vorgenommen werden (siehe Bild 68). So kann beispielsweise je nach Werkstückoberfläche die Belichtungszeit des Sensors über das *Sensor-Interface* manuell angepasst werden (siehe Bild 66c).



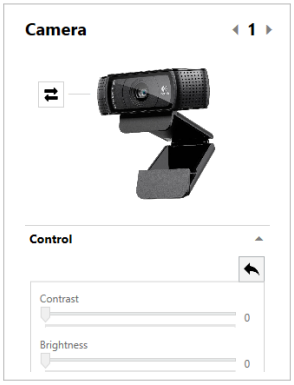
(a) Roboterzelle



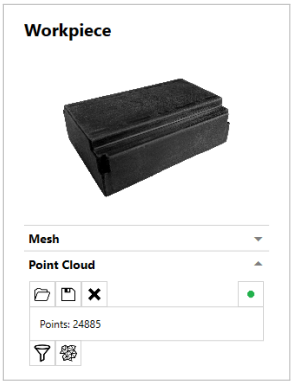
(b) Robot-Interface



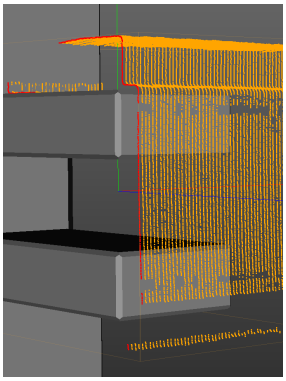
(c) Sensor-Interface



(d) Camera-Interface



(e) Workpiece-Interface



(f) Simulation-Interface

Bild 66: Entwickelte Offline-Programmiersoftware während der Werkstückerfassung

Der Scanvorgang selbst kann über das *Workpiece-Interface* gestartet und gestoppt werden (siehe Bild 66e). Sobald die Aufnahme gestartet ist, werden alle vom Sensor erfassten Messpunkte mit der aktuellen Kinematik des Robotersystems gemäß dem entwickelten Ansatz (siehe Abschnitt 5.2) verrechnet und den Punktwolkendaten des Werkstückes hinzugefügt. Somit kann das Werkstück durch schrittweises Abfahren der Werkstückoberfläche erfasst werden (siehe Bild 66f). Beim Scannen können entsprechend dem entwickelten Ansatz sowohl der Sensor als auch das Werkstück bewegt werden. Die Scanbewegung kann einerseits über das *Robot-Interface* durch Laden oder Erstellen von Programmen definiert werden (siehe Bild 66b). Andererseits ist über das Interface ein manuelles Bewegen der beiden Roboter während des Scanvorgangs möglich. Abschließend kann der Scanvorgang über das *Workpiece-Interface* durch Stoppen der Aufnahme beendet werden.

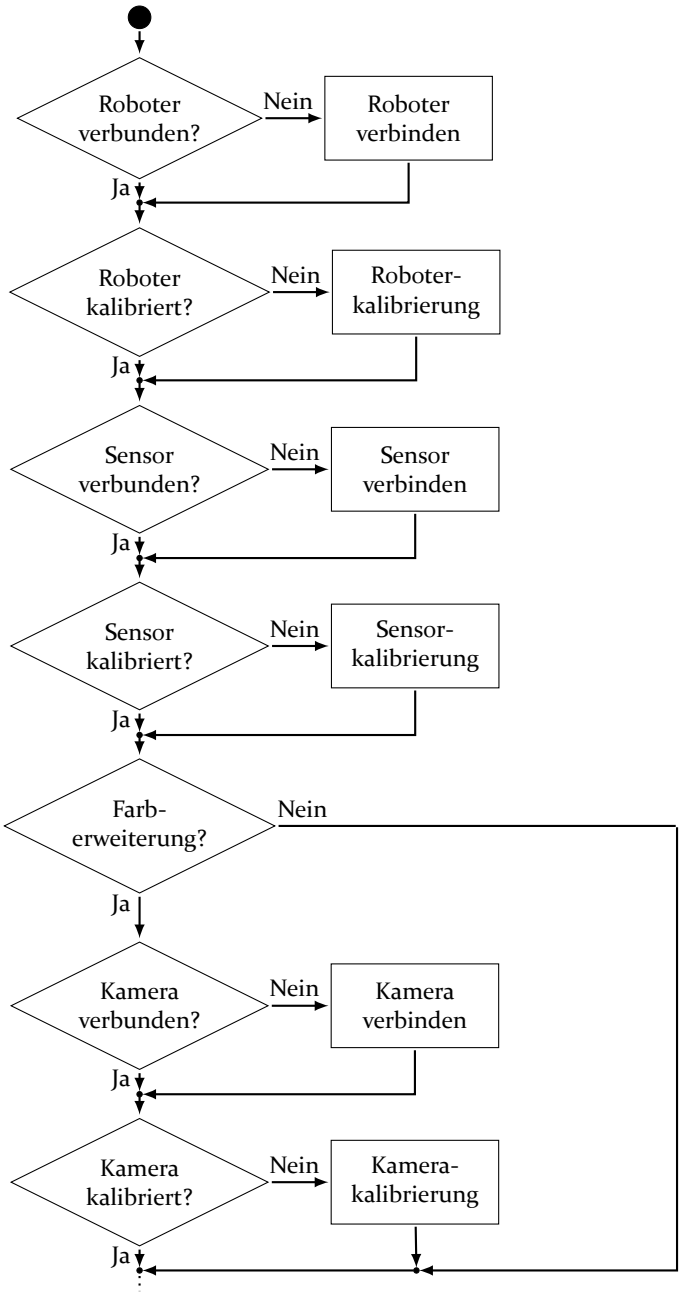


Bild 67: Ablaufplan für die Werkstückerfassung mit der entwickelten Offline-Programmiersoftware - Teil 1/2

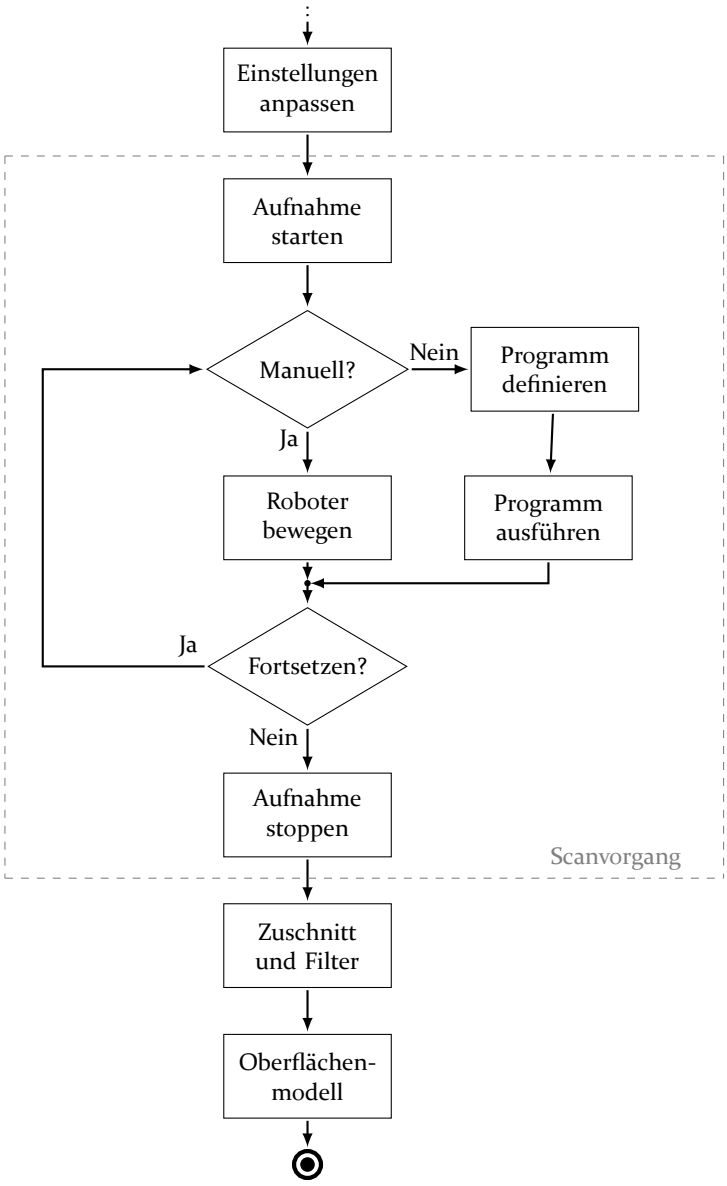


Bild 68: Ablaufplan für die Werkstückerfassung mit der entwickelten Offline-Programmiersoftware - Teil 2/2

Im Anschluss an den Scanvorgang kann die resultierende Punktwolke mittels Zuschnitt- und Fusionsfunktion (siehe Abschnitt 5.3) bearbeitet werden, bevor durch Vernetzen ein Flächenmodell erzeugt wird (siehe Bild 69a-69d). Für das Zuschneiden der Punktwolke können die Grenzen des

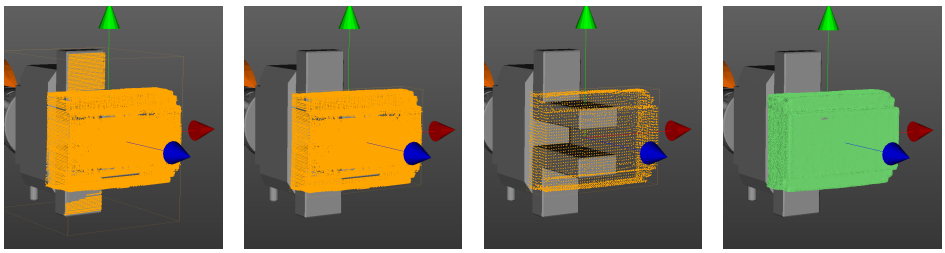
umfassenden Quaders angepasst werden. Die Filterfunktion basiert auf einem Voxelgitter, dessen Raster durch den Benutzer variiert werden kann. Bei der implementierten Vernetzung zur Erzeugung eines Flächenmodells kann der Suchradius und die Anzahl an einbezogenen Nachbarn gewählt werden.

6.2 Robotersystem mit zentraler Steuerung

Im Gegensatz zum zuvor beschriebenen System wird in dieser Implementierung ein MRS mit einer zentralen Steuerung verwendet. Es handelt sich dabei um ein Robotersystem, das für Anwendungen mit MRK ausgelegt ist. Primäres Ziel ist es, den zuvor erarbeiteten Ansatz für die Aufteilung von Bearbeitungsbahnen auf mehrere Industrieroboter in die herstellerspezifische Offline-Programmiersoftware des Robotersystems zu integrieren. Weitergehend erfolgt eine kurze Beschreibung der verwendeten Hardware (siehe Abschnitt 6.2.1) sowie der softwaretechnischen Implementierung (siehe Abschnitt 6.2.2) mit Bezug auf die drei für die Roboterprogrammierung vorgesehenen Ablaufschritte (siehe Bild 51).

6.2.1 Anlagenaufbau

Im Vergleich zum ersten Robotersystem ist diese Anlage vom Aufbau her deutlich einfacher gehalten. Zentrale Komponente dieser Anlage ist ein zweiarmiger Industrieroboter, der für MRK geeignet ist (siehe Bild 70). Dieser ist mittig im hinteren Bereich auf einen Profiltisch, auf dem für unterschiedliche Anwendungen flexibel Komponenten positioniert werden können, montiert. Da der direkte Kontakt zum Menschen bei dieser Anlage erwünscht ist, entfällt hier der Schutzzaun. Die Sicherheit muss jedoch durch den Roboter selbst und eine entsprechende Auslegung der gesamten



(a) Rohdaten (b) Zuschnitt (c) Fusion (d) Flächenmodell

Bild 69: Entwickelte Offline-Programmiersoftware während der Punktwolkenbearbeitung

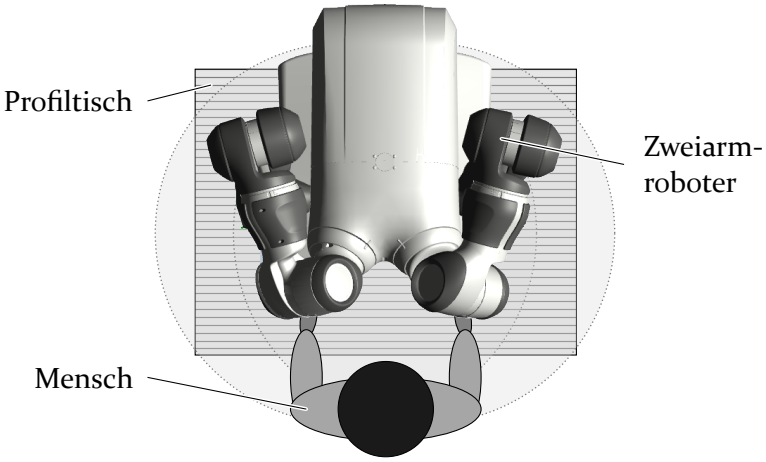
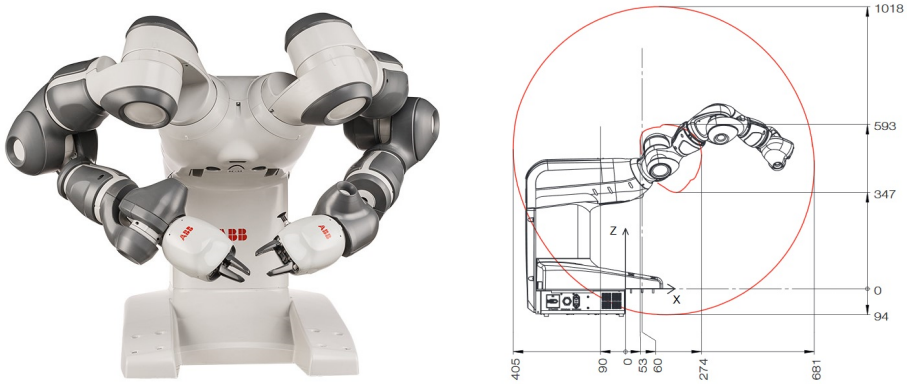


Bild 70: Anlagenaufbau des zweiten Robotersystems für die Implementierung

Anlage erzielt werden. Bedingt durch den verwendeten Roboter ist in dieser Anlage die Positionierung der beiden Arme zueinander und somit der gemeinsame Arbeitsraum der Roboterarme nicht variabel. Es folgt eine kurze Beschreibung der technischen Details des Robotersystems.

Industrieroboter

Zum Einsatz kommt ein zweiarmiger Roboter vom Typ *IRB 14000-0.5/0.5* von *ABB Automation GmbH* (siehe Bild 71), der auch als *YuMi* bezeichnet wird. Dieser ist von seinem Aufbau her dem Menschen nachempfunden, wodurch



(a) Roboter [2] (b) Arbeitsbereich [128]

Bild 71: Zweiarmroboter IRB 14000-0.5/0.5 (YuMi) der ABB Automation GmbH

eine menschenähnliche Beweglichkeit erzielt wird. Zudem sind die Antriebe in den Roboterarmen inhärent sicher dimensioniert, sodass sie eine Person nicht verletzen können, was den Einsatz des Roboters für MRK ermöglicht. Dies bedingt jedoch eine verhältnismäßig geringe maximale Traglast von 500 g pro Arm [128]. Jeder Arm besitzt sieben rotatorische Achsen, hat eine gesteckte Armlänge von ca. 559 mm und erzielt laut Hersteller eine Positionswiederholgenauigkeit von $\pm 0,02$ mm. Basierend auf den Herstellerangaben können die DH-Parameter gemäß der DH-Konvention (siehe Abschnitt 2.2) für die $N_\mathrm{A} = 7$ Achsen aufgestellt werden (siehe Tabelle 8).

Der Roboter besitzt eine *zentrale Steuerung* vom Typ *IRC5*, die im Sockel integriert ist. Diese steuert beide Roboterarme an und übernimmt die Koordination zwischen den Roboterarmen, welche mit dem enthaltenen Funktionspaket *Coordinated Robots* ermöglicht wird. Für die Synchronisation von Programmpunkten ist die Funktion *WaitSyncTask* und für die Synchronisation mit überschliffenen Bahnpunkten die Funktion *SyncMove* enthalten.

6.2.2 Softwareimplementierung

Für dieses Robotersystem wird die herstellerspezifische Offline-Programmiersoftware *RobotStudio* in der Version *6.06.01* herangezogen und um die für die kooperative Bearbeitung erforderlichen Funktionen erweitert (siehe Bild 72). Dies erfolgt mit Hilfe des *RobotStudio SDK*, welches das Erstellen von individuellen Add-Ins in der Programmiersprache *C#* ermöglicht. Das

Tabelle 8: DH-Parameter des Zweiarmroboters IRB 14000-0.5/0.5 (YuMi) der ABB Automation GmbH

Achse i	θ_i in rad	d_i in mm	a_i in mm	α_i in rad
1	$\theta_1 - \pi$	$110,0$	$30,0$	$\pi/2$
2	$\theta_2 - \pi$	$0,0$	$30,0$	$\pi/2$
3	θ_3	$256,5$	$40,5$	$-\pi/2$
4	$\theta_4 - \pi/2$	$0,0$	$40,5$	$-\pi/2$
5	$\theta_5 + \pi$	$265,0$	$13,5$	$-\pi/2$
6	$\theta_6 + \pi$	$0,0$	$27,0$	$-\pi/2$
7	θ_7	$32,0$	$0,0$	0

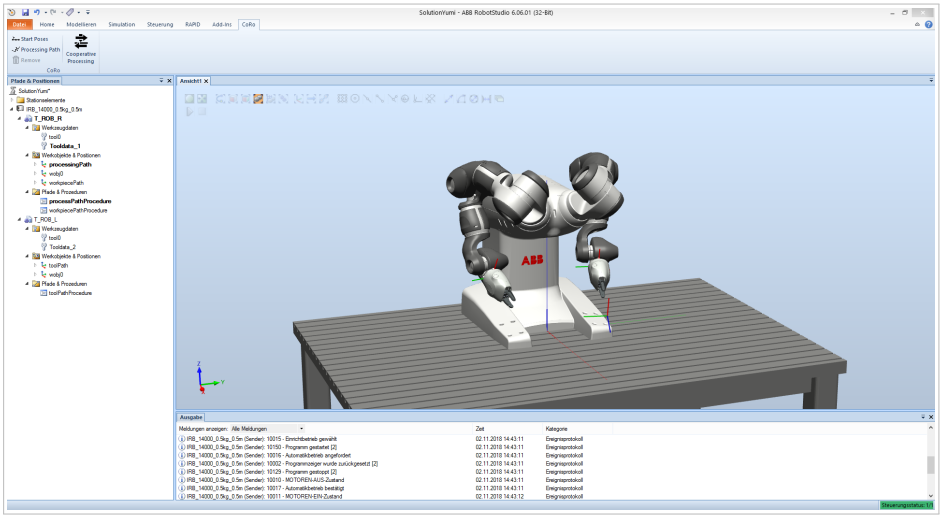


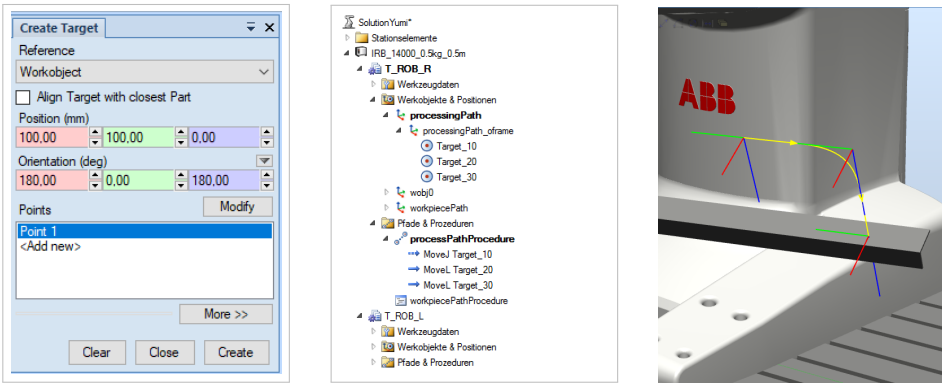
Bild 72: Softwareoberfläche der herstellerspezifischen Offline-Programmiersoftware mit integriertem Add-In für die kooperative Bearbeitung

Add-In erscheint als zusätzlicher Reiter in der Hauptmenüleiste im oberen Bereich der Softwareoberfläche. Über die unter diesem Reiter hinterlegten Schaltflächen können alle für die Aufteilung einer Prozessbahn relevanten Funktionen vom Bediener ausgeführt werden.

Weitergehend werden die drei zentralen Teilschritte des hier umgesetzten Konzeptes dargestellt. Zunächst erfolgt eine Beschreibung der *Bahndefinition*, welche mit Hilfe der Standardfunktionen der herstellerspezifischen Offline-Programmiersoftware erzeugt werden. Anschließend wird ausführlich auf den Ablauf bei der *Bahnaufteilung* eingegangen. Zuletzt folgt die *Programmgenerierung und -überprüfung*, die wiederum auf die Standardfunktionen der herstellerspezifische Offline-Programmiersoftware zurückgreift.

Bahndefinition

Die Bahndefinition kann über die Standardfunktionen der herstellerspezifischen Offline-Programmiersoftware erfolgen, indem die gewünschten Zielposen über das *Target-Interface* (siehe Bild 73a) der Bearbeitungsbahn hinzugefügt werden. Auch alternative Eingabemethoden, wie beispielsweise die enthaltene Möglichkeit der Definition von Bahnposen anhand der Ränder von Oberflächen, sind anwendbar. Unabhängig von der Art der Eingabe werden die Zielposen auf das Werkzeugkoordinatensystem des



(a) Target-Interface (b) Bearbeitungsbahn-Liste (c) Bearbeitungsbahn

Bild 73: Proprietäre Offline-Programmiersoftware während der Bahndefinition

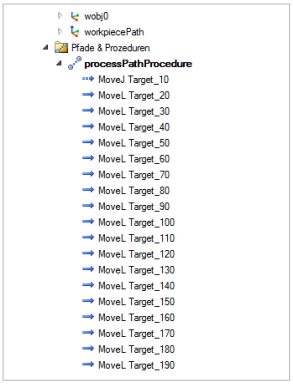
werkstückführenden Roboters bezogen erstellt, mit dem das Werkstück verknüpft ist. Alle Bahnposen sind in einer Liste (siehe Bild 73b) aufgeführt. Des Weiteren können sie in eine Bearbeitungsbahn überführt werden. Anschließend resultiert der werkstückbezogene Verlauf der Bearbeitungsbahn, welcher in der Simulationsumgebung in gelber Farbe dargestellt ist (siehe Bild 73c).

Bahnaufteilung

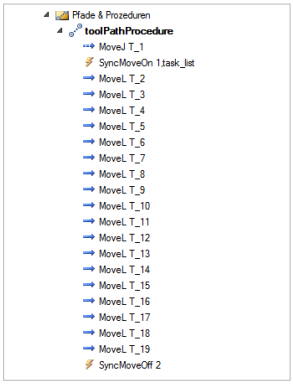
Die Bahnaufteilung für das zweite Robotersystem erfolgt nach dem identischen Ablauf wie beim ersten Robotersystem (siehe Bild 62). Das Initiieren der Aufteilung erfolgt über eine Schaltfläche des Add-In. Anschließend verarbeitet im Hintergrund die Software die zuvor definierte Bearbeitungsbahn (siehe Bild 74a und Bild 74d) gemäß dem Aufteilungsansatzes und erzeugt eine Werkzeug- und eine Werkstückbahn, die ebenfalls jeweils in einer Liste (siehe Bild 74b und Bild 74c) und in der Simulationsumgebung (siehe Bild 74e und Bild 74f) eingeblendet werden. In den beiden Listen sind die notwendigen Synchronisationsbefehle bereits enthalten. Zudem können an dieser Stelle die roboterbezogenen Abläufe mit allen in der Software enthaltenen Funktionen durch den Benutzer angepasst werden, um so beispielsweise das Ansteuern von Signalausgängen oder Einzelbewegungen hinzuzufügen.

Programmgenerierung und -überprüfung

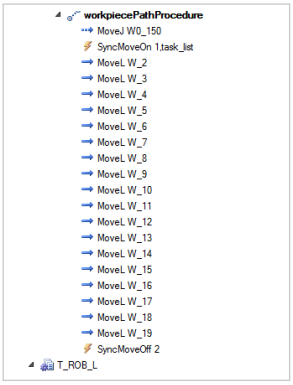
Der Ablauf der Programmgenerierung und -überprüfung orientiert sich am zuvor erläuterten Ablauf des ersten Robotersystems (siehe Bild 64), fällt



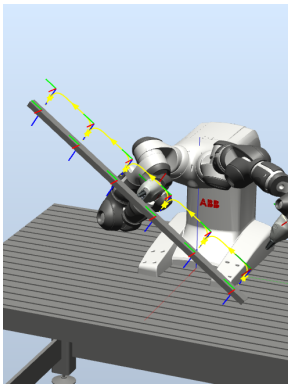
(a) Bearbeitungsbahn-Liste



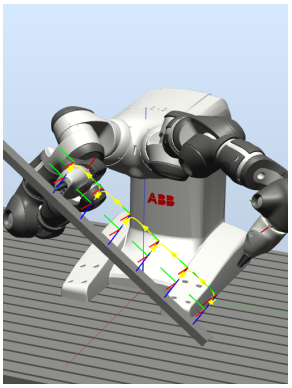
(b) Werkzeugbahn-Liste



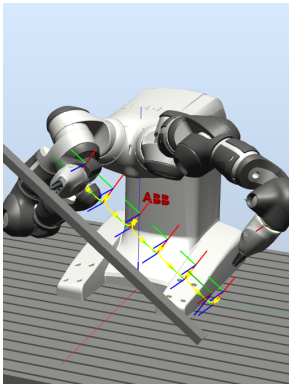
(c) Werkstückbahn-Liste



(d) Bearbeitungsbahn



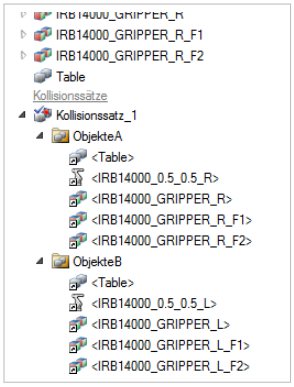
(e) Werkzeugbahn



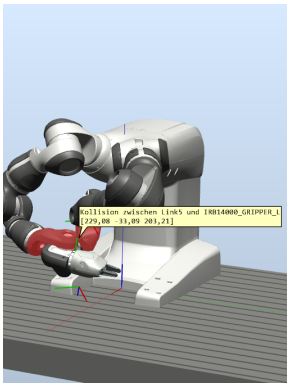
(f) Werkstückbahn

Bild 74: Proprietäre Offline-Programmiersoftware während der Bahnaufteilung

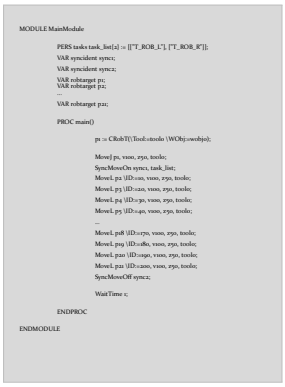
jedoch etwas vereinfacht aus. Die Programmabläufe können ebenfalls mittels der Simulationsumgebung auf Kollisionen überprüft werden. Hierfür ist in das Robotermodell ein Kollisionsmodell integriert (siehe Bild 75a), das aber wie die Synchronisation der Roboterarme standardmäßig bereits aktiviert ist. Sobald eine Kollision im System auftritt, werden die beteiligten Komponenten rot eingefärbt (siehe Bild 75b) und die Simulation kann bei Bedarf automatisch angehalten werden. Gegebenenfalls gilt es die Programmabläufe an dieser Stelle nochmals anzupassen. Damit abschließend die Anwendung am realen Robotersystem umgesetzt werden kann, müssen die Roboterprogramme (siehe Bild 75c) mit der entsprechenden Funktion der Offline-Programmiersoftware abgespeichert und auf die Steuerung des Robotersystems übertragen werden.



(a) Kollisionsmodell



(b) Kollision



```
MODULE MainModule

    PERS tasks task_list[2] := [["T_ROB_L"], ["T_ROB_R"]];
    VAR syncident sync1;
    VAR syncident sync2;
    VAR robtarget p1;
    VAR robtarget p2;
    ...
    VAR robtarget p21;

    PROC main()

        p1 := CRobT(\Tool:=tool0 \WObj:=wobj0);

        MoveJ p1, v1000, z50, tool0;
        SyncMoveOn sync1, task_list;
        MoveL p2 \ID:=10, v1000, z50, tool0;
        MoveL p3 \ID:=20, v1000, z50, tool0;
        MoveL p4 \ID:=30, v1000, z50, tool0;
        MoveL p5 \ID:=40, v1000, z50, tool0;
        ...
        MoveL p18 \ID:=170, v1000, z50, tool0;
        MoveL p19 \ID:=180, v1000, z50, tool0;
        MoveL p20 \ID:=190, v1000, z50, tool0;
        MoveL p21 \ID:=200, v1000, z50, tool0;
        SyncMoveOff sync2;

        WaitTime 1;

    ENDPROC
ENDMODULE
```

(c) Programmdatei

Bild 75: Proprietäre Offline-Programmiersoftware während der Programmgenerierung und -überprüfung

6.3 Zusammenfassung

Die beiden im Rahmen dieser Arbeit realisierten Robotersysteme können flexibel für unterschiedliche Anwendungen eingesetzt werden. Wesentlicher Bestandteil ist jeweils die in die Offline-Programmiersoftware integrierte Möglichkeit einzelne Prozessbahnen oder ganze Prozesse gemäß des entwickelten Ansatzes auf die beiden Roboterarme aufzuteilen. Das erste Robotersystem ist zudem um eine Werkstückerfassung erweitert, mit der Flächenmodelle von Werkstücken erzeugt werden können. Diese können anschließend beispielsweise auch als Grundlage für eine Offline-Programmierung verwendet werden, wie im weiteren Verlauf dieser Arbeit anhand einer exemplarischen Anwendung gezeigt wird. Des Weiteren sind die zuvor erarbeiteten Ansätze für die Untersuchung von MRS hinsichtlich der kooperativen Bearbeitung ebenfalls softwaretechnisch implementiert, sodass die Robotersysteme weitergehend evaluiert werden können.

7 Evaluation

Die im Rahmen dieses Kapitels durchgeführte Evaluation dient einer technischen und wirtschaftlichen Bewertung der zuvor entwickelten Ansätze und gliedert sich in zwei Bereiche. Zunächst erfolgen allgemeine Untersuchungen anhand der für die Implementierung gewählten Robotersysteme (siehe Abschnitt 7.1), welche die Grenzen der Anlagen aufzeigen. Dem schließen sich anwendungsspezifische Untersuchungen an (siehe Abschnitt 7.2), bei denen mit den realisierten Robotersystemen eine Reihe von exemplarischen Anwendungen umgesetzt und analysiert werden.

7.1 Allgemeine Untersuchungen

Die allgemeine Untersuchung der für die Implementierung gewählten Robotersysteme beinhaltet vier Komponenten (siehe Tabelle 9). Bevor mit einem Robotersystem der erarbeitete Ansatz für die kooperative Bearbeitung (siehe Abschnitt 4) durchgeführt wird, sollte ein geeignetes Raster für die Startposeniteration gewählt werden, um keine unnötig großen Berechnungszeiten oder ungünstige Aufteilungsvarianten zu erhalten. Daher wird zunächst für beide Robotersysteme die Auswahl einer geeigneten Diskretisierung der Startposeniteration (siehe Abschnitt 7.1.1) durchgeführt. Des Weiteren werden im Rahmen dieses Abschnittes die Grenzen der Bearbeitungszeitreduzierung für die verwendeten Robotersysteme ermittelt (siehe Abschnitt 7.1.2). Ebenso erfolgt eine Betrachtung der allgemeinen Erreichbarkeit der verwendeten Robotersysteme (siehe Abschnitt 7.1.3). Für das erste Robotersystem, das aus zwei frei zueinander positionierten Knickarmrobotern besteht, erfolgt zudem eine kinematische Kalibrierung (siehe Abschnitt 7.1.4). Beim zweiten Robotersystem entfällt diese, da es sich um

Tabelle 9: Übersicht über die allgemeinen Untersuchungen

Untersuchung	1. Robotersystem	2. Robotersystem
Diskretisierung der Startposeniteration	✓	✓
Grenzen der Bearbeitungszeitreduzierung	✓	✓
Allgemeine Erreichbarkeit	✓	✓
Kinematische Kalibrierung	✓	–

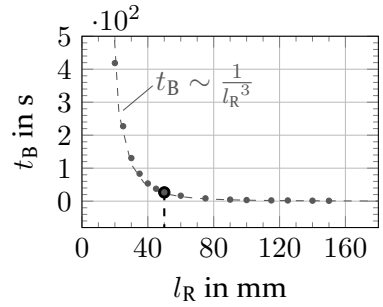
ein gebundenes System handelt und dieses eine ausreichende relative Bearbeitungsgenauigkeit mit sich bringt.

7.1.1 Auswahl einer geeigneten Diskretisierung der Startposeniteration

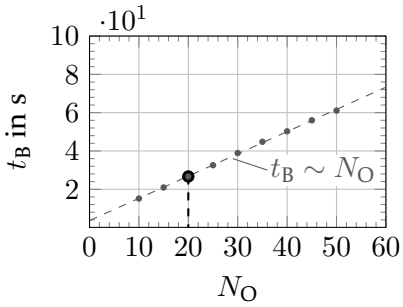
Die Auswahl einer geeigneten Diskretisierung der Startposeniteration erfolgt für beide Robotersysteme gemäß dem zuvor beschriebenen Ansatz (siehe Abschnitt 4.1.1) und auf Grundlage einer Bewegung in Form von einer einzelnen linearen Bahn von 200 mm Länge. Dabei wird zunächst die Rastergröße zwischen $l_{\mathrm{R}} = 20$ mm und $l_{\mathrm{R}} = 150$ mm variiert. Des Weiteren wird die Anzahl der Orientierungen zwischen $N_{\mathrm{O}} = 10$ und $N_{\mathrm{O}} = 50$ variiert. Für jede Variante wird sowohl die Berechnungszeit t_{B} als auch die resultierende Summe der maximalen Achswinkeländerungen Θ erfasst. Anschließend kann hieraus gemäß der zuvor erarbeiten Formel eine Güteziffer G berechnet werden, anhand derer sich eine geeignete Rastergröße sowie eine geeignete Orientierungsanzahl ablesen lassen.

Die für das erste Robotersystem resultierende Berechnungszeit (siehe Bild 76a) reduziert sich mit zunehmender Rastergröße in der dritten Potenz. Bei der resultierenden Summe der maximalen Achswinkeländerungen (siehe Bild 76c) ist kein eindeutiger Zusammenhang zu erkennen. Dennoch zeigt sich, dass bei einem großen Raster deutlich größere Werte resultieren. Unterhalb einer Rastergröße von $l_{\mathrm{R}} = 60$ mm variiert der Wert nur noch geringfügig. Für die in Abhängigkeit der Rastergröße ermittelten Werte ergibt sich somit eine maximale Güteziffer G bei einer Rastergröße von $l_{\mathrm{R}} = 50$ mm (siehe Bild 76e). Daher wird dieser Wert im Weiteren verwendet. Bei der Variation der Orientierungsanzahl zeigt sich ein linearer Zusammenhang zwischen der Berechnungszeit und der Anzahl an Orientierungen (siehe Bild 76b). Die resultierenden Summen der maximalen Achswinkeländerungen (siehe Bild 76d) schwanken stark und es ist ebenfalls kein klarer Verlauf zu erkennen. Eine geringe Anzahl an Orientierungen führt jedoch zu deutlich größeren Achswinkeländerungen. Für die Wahl einer geeigneten Anzahl an Orientierungen wird wiederum die Güteziffer G verwendet (siehe Bild 76f). Die höchste Güteziffer ergibt sich für eine Anzahl an Orientierungen von $N_{\mathrm{O}} = 20$.

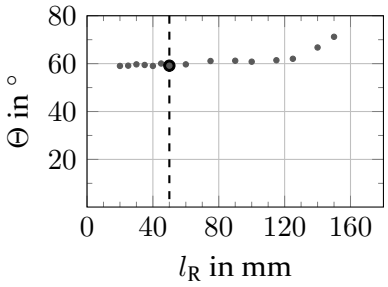
Bei der Betrachtung des zweiten Robotersystems ergibt sich ein vergleichbares Verhalten. So reduziert sich die resultierende Berechnungszeit mit zunehmender Rastergröße ebenfalls in der dritten Potenz (siehe Bild 77a). Bei der resultierenden Summe der maximalen Achswinkeländerungen (siehe



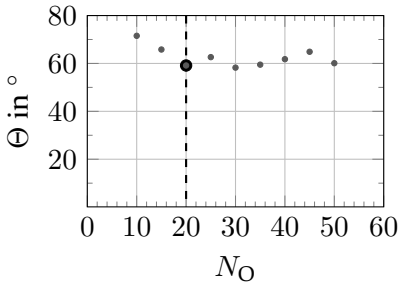
(a) Berechnungszeit in Abhängigkeit der Rastergröße



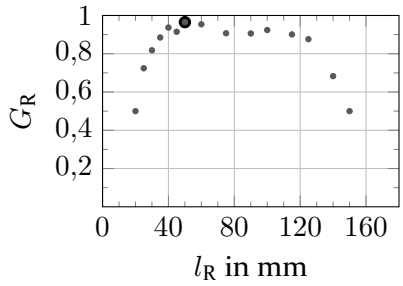
(b) Berechnungszeit in Abhängigkeit der Orientierungsanzahl



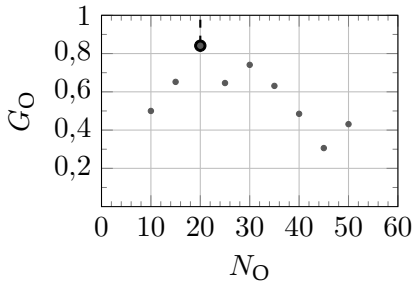
(c) Summe der maximalen Achswinkeländerungen in Abhängigkeit der Rastergröße



(d) Summe der maximalen Achswinkeländerungen in Abhängigkeit der Orientierungsanzahl



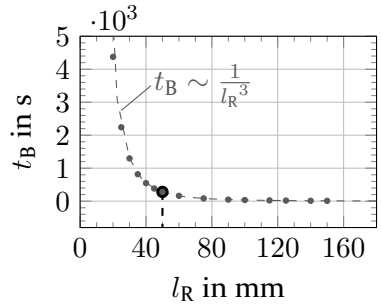
(e) Güteziffer in Abhängigkeit der Rastergröße



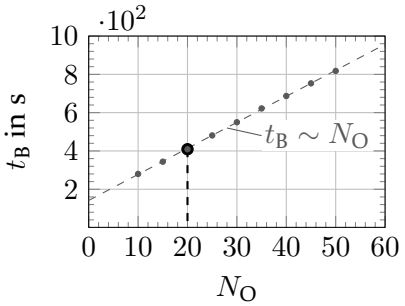
(f) Güteziffer in Abhängigkeit der Orientierungsanzahl

Bild 76: Messwerte aus der Diskretisierung der Startposeniteration für das erste Robotersystem

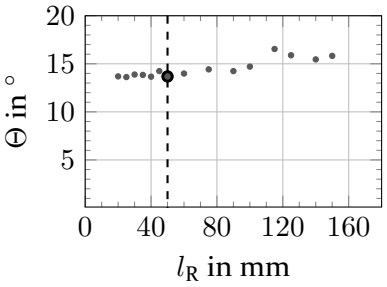
Bild 77c) zeigt sich ebenfalls, dass sich bei großen Rastergrößen deutlich größere Werte ergeben. Für die in Abhängigkeit der Rastergröße ermittelten Werte folgt somit eine maximale Güteziffer G bei einer Rastergröße



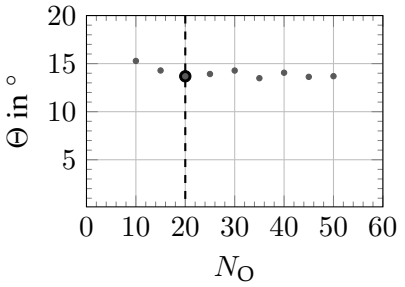
(a) Berechnungszeit in Abhängigkeit der Rastergröße



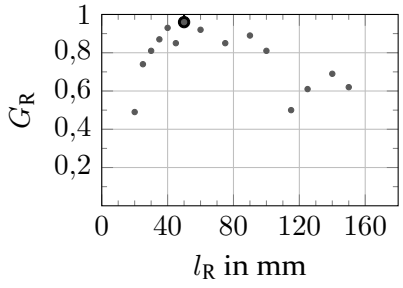
(b) Berechnungszeit in Abhängigkeit der Orientierungsanzahl



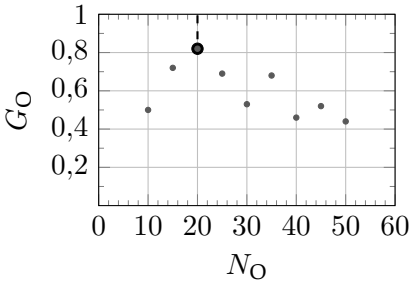
(c) Summe der maximalen Achswinkeländerungen in Abhängigkeit der Rastergröße



(d) Summe der maximalen Achswinkeländerungen in Abhängigkeit der Orientierungsanzahl



(e) Güteziffer in Abhängigkeit der Rastergröße



(f) Güteziffer in Abhängigkeit der Orientierungsanzahl

Bild 77: Messwerte aus der Diskretisierung der Startposeniteration für das zweite Robotersystem

von $l_R = 50$ mm (siehe Bild 77e). Wie beim ersten Robotersystem zeigt sich bei der Variation der Orientierungsanzahl ein linearer Zusammenhang zwischen der Berechnungszeit und der Anzahl an Orientierungen (siehe

Bild 77b). Die resultierenden Summen der maximalen Achswinkeländerungen (siehe Bild 77d) schwanken stark und es ist ebenfalls kein klarer Verlauf identifizierbar. Eine erkennbare Tendenz ist, dass eine geringe Anzahl an Orientierungen zu größeren Achswinkeländerungen führt. Die höchste Güteziffer G ergibt sich für eine Anzahl an Orientierungen von $N_{\mathrm{O}} = 20$ (siehe Bild 77f).

7.1.2 Grenzen der Bearbeitungszeitreduzierung der verwendeten Robotersysteme

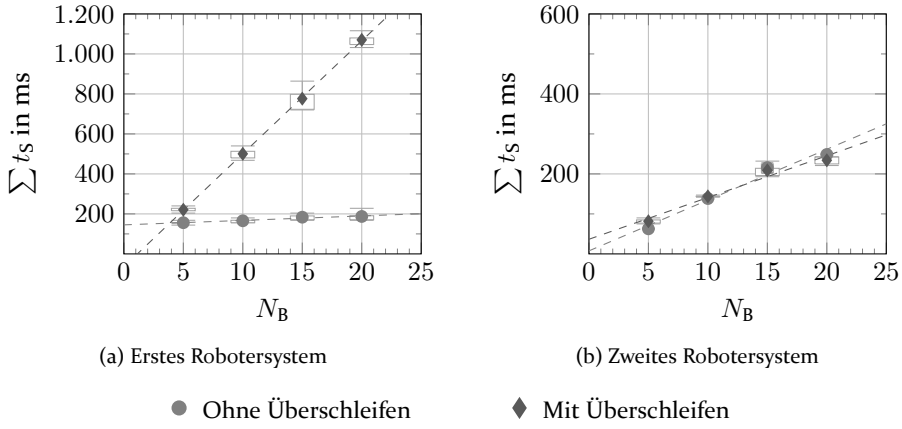
Im Rahmen dieses Abschnittes werden die allgemeinen *Synchronisationszeiten* sowie das *Beschleunigungsverhalten* der verwendeten Robotersysteme ermittelt und die damit verbundenen *Grenzen der Bearbeitungszeitreduzierung* entsprechend dem im Voraus beschriebenen Ansatz (siehe Abschnitt 4.3.1) abgeleitet.

Synchronisationszeit

Für die Ermittlung der Synchronisationszeit werden jeweils Programme erstellt, die $N_{\mathrm{B}} = 5, 10, 15$ oder 20 Bewegungsbefehle zur selben Pose sowie die zugehörigen Synchronisationsbefehle enthalten. Außerdem werden die Zeiten vor und nach den Bewegungssetzen gespeichert, sodass aus der Differenz der beiden Zeiten die jeweilige Durchlaufzeit berechnet werden kann. Des Weiteren werden die Programme ohne Synchronisationsbefehle erstellt, sodass aus der Differenz der beiden zugehörigen Programmzeiten die jeweilige Gesamtsynchronisationszeit $\sum t_{\mathrm{S}}$ abgeleitet werden kann (siehe Bild 78). Das Erfassen der Synchronisationszeit erfolgt für beide Robotersysteme sowohl mit als auch ohne Überschleifen der Bahnpunkte und wird in jeder Variation zehnmal wiederholt, sodass anschließend die Mittelwerte gebildet werden können, um Messfehler zu minimieren. Wie die jeweiligen Box-Plots verdeutlichen, sind die Schwankungen der Wiederholungsmessungen verhältnismäßig gering. Für die Abhängigkeit der Synchronisationszeit gegenüber der Anzahl an Bewegungsbefehlen zeigt sich ein annähernd linearer Zusammenhang. Anhand der approximierten Geraden lassen sich Gleichungen für die Gesamtsynchronisationszeiten $\sum t_{\mathrm{S}}$ in Abhängigkeit der Anzahl an Bahnelementen N_{B} ableiten (siehe Tabelle 10).

Beschleunigungsverhalten

Das Beschleunigungsverhalten eines realen Robotersystems kann stark vom theoretischen Beschleunigungsverhalten abweichen. Um die genauen



(a) Erstes Robotersystem

(b) Zweites Robotersystem

Bild 78: Synchronisationszeiten in Abhängigkeit von der Anzahl an Bewegungsbefehlen

Zusammenhänge besser nachvollziehen zu können, wird das Beschleunigungsverhalten im Folgenden genauer untersucht. Hierfür werden Messungen der am TCP vorliegenden Beschleunigung bei einer linearen Bewegung innerhalb des kooperativen Arbeitsbereiches durchgeführt. Für die Messungen wird ein 3-Achsen-Beschleunigungssensor vom Typ *MPU-6050* in Kombination mit einem Mikrocontroller-Board *Arduino Nano* verwendet (siehe Bild 79). Diese sind auf einer gemeinsamen Grundplatte befestigt, welche über einen Dämpfer an den Flansch des verwendeten Roboters montiert ist, um die Übertragung von Schwingungen, die das Messergebnis negativ beeinflussen könnten, zu reduzieren. Die Datenerfassung erfolgt mit Hilfe der Software *Processing 3.3.5* über einen separaten Rechner, der per USB an das Mikrocontroller-Board angeschlossen ist. Um Störgrößen weiter zu reduzieren, werden die Rohdaten durch einen Tiefpassfilter (Grenzfrequenz

Tabelle 10: Experimentell bestimmte Gleichungen für die Synchronisationszeiten der Robotersysteme mit und ohne Überschleifen der Bahnpunkte

	$\sum t_S\,(N_B)$ in ms	
	ohne Überschleifen	mit Überschleifen
1. Robotersystem	$2,2 \cdot N_B + 145,2$	$56,5 \cdot N_B - 64,2$
2. Robotersystem	$12,7 \cdot N_B + 7,6$	$10,4 \cdot N_B + 36,6$

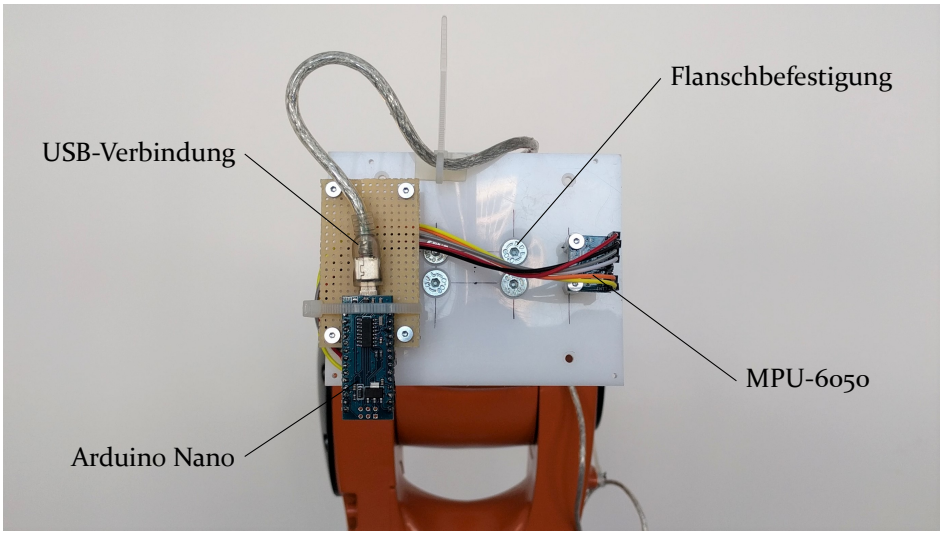


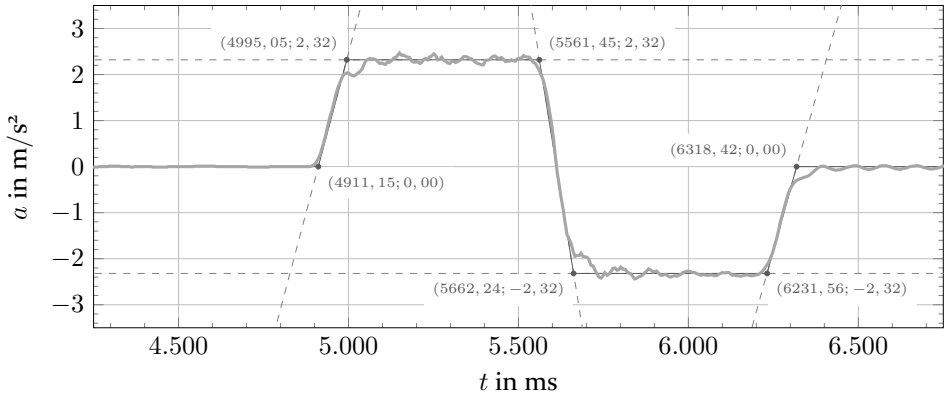
Bild 79: Am Roboterflansch montierter Sensor für die Beschleunigungsmessung

$f_C = 5$ Hz) sowie eine Glättung mittels gleitendem Mittelwert (Ordnung 7) bearbeitet, der sich berechnet durch:

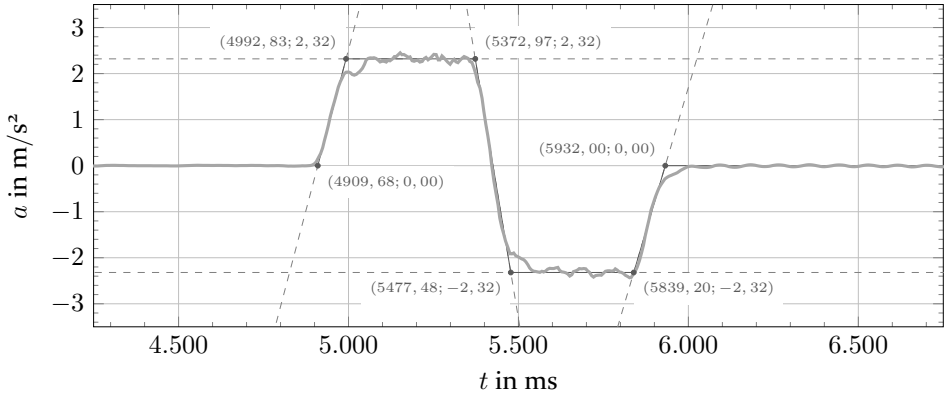
$$\overline{a}_i^{(7)} = \frac{1}{7} \sum_{j=-3}^{3} a\left(t_{i+j}\right) \,, \qquad (7.1)$$
$$i \in \{1,\, 2,\, \ldots,\, N_M\} \,, \quad j \in \{-3,\, -2,\, \ldots,\, 3\}$$

Des Weiteren erfolgt die Montage des Sensors so, dass der Einfluss der Beschleunigung in Querrichtung zur Bewegung vernachlässigt werden kann. Daher wird für die Beschreibung der Daten die Skalarform verwendet.

Für das erste Robotersystem erfolgen die Bewegungen entlang der x-Achse des Basiskoordinatensystems und über Distanzen von 500 mm sowie 1000 mm. Die Ausgangspose ist in beiden Fällen $\mathbf{f}_S = (-250\,\text{mm} \;\; 500\,\text{mm} \;\; 500\,\text{mm} \;\; 90° \;\; 90° \;\; 0°)^T$. Die resultierenden Daten (siehe Bild 80) entsprechen annähernd dem Beschleunigungsprofil mit Ruckbegrenzung, wobei die Maximalbeschleunigung erreicht wird. Trotz der im Verhältnis zur Robotergröße langen Strecken wird die Maximalgeschwindigkeit nicht erreicht. Für die weiteren Berechnungen werden die Verläufe durch abschnittsweise Linearisierung



(a) Volle Bearbeitungsstrecke ($x = 1000$ mm)



(b) Halbe Bearbeitungsstrecke ($x = 500$ mm)

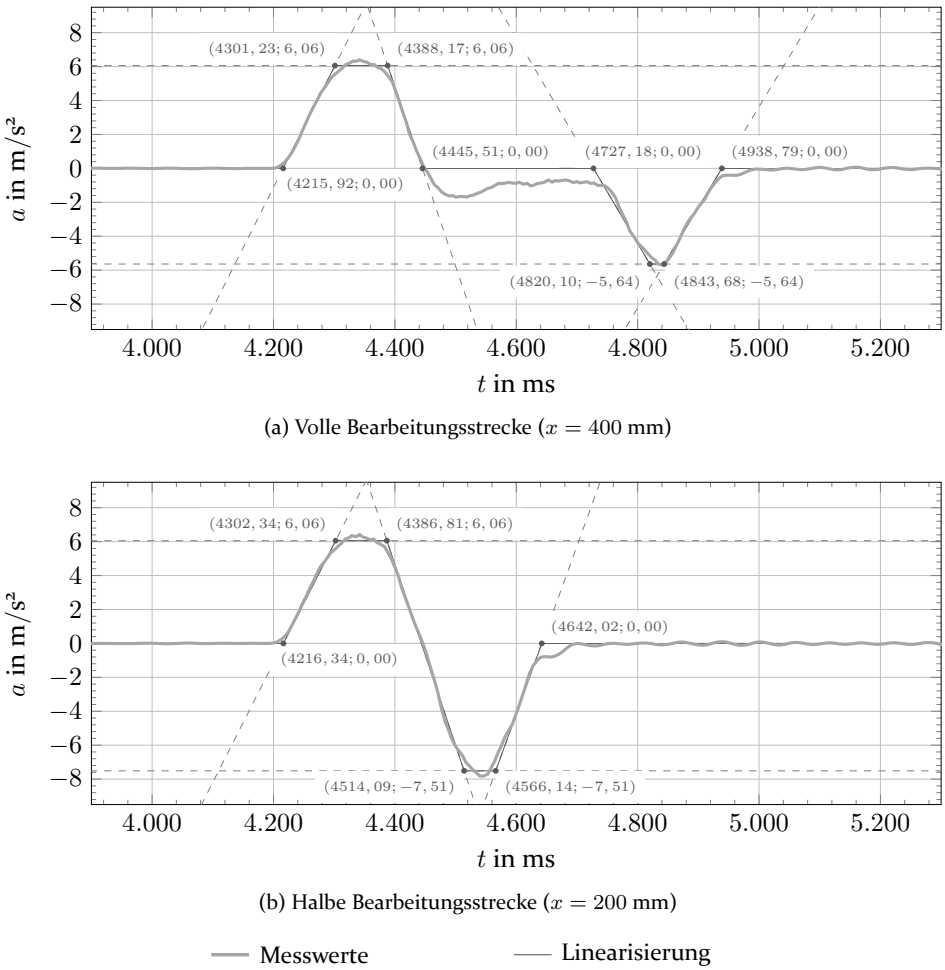
Messwerte — Linearisierung

Bild 80: Beschleunigungsprofil des ersten Robotersystems

abgebildet. Basierend hierauf können die angenäherten Eckpunkte des Beschleunigungsprofils anhand der Schnittpunkte der Geraden berechnet werden. Es ergibt sich somit eine Bewegungszeit von $t_{\mathrm{W/T}} = 1407,3$ ms für die nicht aufgeteilte Bewegung und eine Bewegungszeit $t_{\mathrm{M}} = 1022,3$ ms für die kooperative Bewegung.

Für das zweite Robotersystem werden die Bewegungen ebenfalls entlang der x-Achse des Basiskoordinatensystems durchgeführt, aufgrund der geringeren Reichweite jedoch lediglich über Distanzen von 200 mm sowie 400 mm. In beiden Fällen ist die Ausgangspose $\mathbf{f}_{\mathrm{S}} = (600\,\mathrm{mm} \quad -200\,\mathrm{mm} \quad 400\,\mathrm{mm} \quad 90° \quad 0° \quad 90°)^{\mathrm{T}}$. Die resultierenden

Beschleunigungsprofile (siehe Bild 81) entsprechen ebenfalls annähernd dem Beschleunigungsprofil mit Ruckbegrenzung, wobei in beiden Fällen die Maximalbeschleunigung erreicht wird. Bei der längeren Distanz wird zudem die Maximalgeschwindigkeit erreicht, wobei in diesem Bewegungsabschnitt systembedingt eine geringfügige negative Beschleunigung vorliegt. Auch bei diesen Messungen erfolgt für die weiteren Berechnungen eine Abbildung der Verläufe durch abschnittsweise Linearisierung. Es resultiert eine Bewegungszeit von $t_{\mathrm{W/T}} = 722,9$ ms für die nicht aufgeteilte Bewegung und eine Bewegungszeit $t_{\mathrm{M}} = 425,7$ ms für die kooperative Bewegung.



(a) Volle Bearbeitungsstrecke ($x = 400$ mm)

(b) Halbe Bearbeitungsstrecke ($x = 200$ mm)

Bild 81: Beschleunigungsprofil des zweiten Robotersystems

Grenzen der Bearbeitungszeitreduzierung

Aus dem Verhältnis der beiden Gesamtbewegungszeiten resultiert gemäß Formel 4.21 eine Bewegungszeiteinsparung von $p_M \approx 27,4\,\%$ für das erste Robotersystem sowie $p_M \approx 41,1\,\%$ für das zweite Robotersystem. Letztere liegt über der zuvor ermittelten theoretischen Grenze (siehe Abschnitt 4.3.1), da ein Wechsel des Verfahrprofils vorliegt. Da Prozessbewegungen meist jedoch deutlich kürzer sind als die hier angesetzten Distanzen, fällt die Zeiteinsparung hierbei auch geringer aus, wie im weiteren Verlauf dieser Arbeit noch dargestellte Beispielanwendungen bestätigen. Eine Bewegung ohne Erreichen der maximalen Beschleunigung liegt unterhalb einer Grenzstrecke von $x_G \approx 19,02$ mm sowie $x_G \approx 72,65$ mm, die mittels Formel 4.22 berechnet sind, vor. Hierbei nimmt die Zeiteinsparung weiter ab, wie exemplarische Anwendungen weitergehend ebenfalls bestätigen.

Zudem müssen noch die für eine Synchronisation notwendigen Zeiten gemäß Formel 4.18 berücksichtigt werden. Somit ergibt sich für das erste Robotersystem eine Zeit von $t_C = 1169,7$ ms für eine kooperative Bewegung ohne Überschleifen und eine Zeit von $t_{C_{Cont}} = 1014,6$ ms für eine kooperative Bewegung mit Überschleifen. Die tatsächliche prozentuale Zeiteinsparung nach Formel 4.17 beträgt dementsprechend $p \approx 16,9\,\%$ bzw. $p \approx 27,9\,\%$. Für das zweite Robotersystem resultiert eine Zeit von $t_C = 446,0$ ms für eine kooperative Bewegung ohne Überschleifen und eine Zeit von $t_{C_{Cont}} = 472,7$ ms für eine kooperative Bewegung mit Überschleifen. Die prozentuale Zeiteinsparung beträgt $p \approx 38,3\,\%$ bzw. $p \approx 34,6\,\%$.

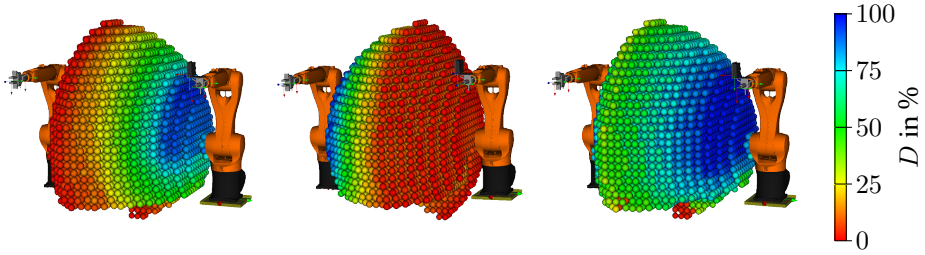
7.1.3 Allgemeine Erreichbarkeit der verwendeten Robotersysteme

Gemäß dem zuvor beschriebenen Ansatz (siehe Abschnitt 4.3.2) erfolgt eine Untersuchung der allgemeinen Erreichbarkeit innerhalb des gemeinsamen Arbeitsraumes der verwendeten Robotersysteme. Der werkstückführende Roboter befindet sich dabei in beiden Robotersystemen auf der linken Seite und der werkzeugführende Roboter auf der rechten Seite. Entsprechend des zuvor erläuterten Ansatzes wird jeweils für alle drei Bewegungsprinzipien eine *Reachability-Map* erzeugt. Um die erzeugten *Reachability-Maps* besser vergleichen zu können, werden die Resultate für eine reine Werkzeugbewegung und eine reine Werkstückbewegung auf den gemeinsamen Arbeitsbereich der beiden Roboter reduziert. Des Weiteren erfolgt eine numerische *Auswertung* der erzeugten Daten. Abschließend wird zudem der Effekt von relevanten *Einflussgrößen* auf die Erreichbarkeit anhand des ersten Robotersystems veranschaulicht.

Reachability-Maps

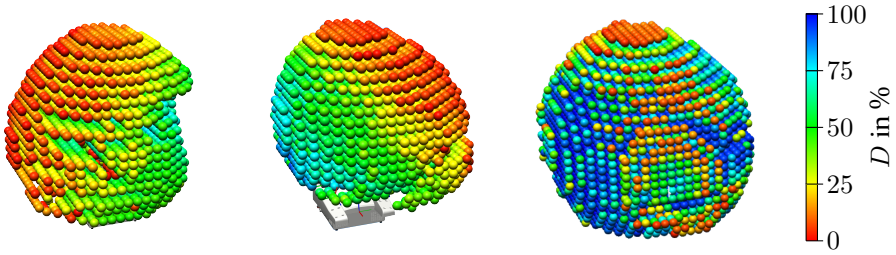
Die Berechnungen der *Reachability-Maps* für das erste Robotersystem werden in die zuvor selbst entwickelte Offline-Programmiersoftware (siehe Abschnitt 6.1.2) integriert. Dies erfolgt unter Verwendung des im Rahmen dieser Arbeit realisierten vereinfachten Kollisionsmodels, um eine möglichst große Realitätsnähe der erzeugten *Reachability-Maps* (siehe Bild 82) zu erzielen. Das Kollisionsmodel nähert die Roboter samt Peripherie als Kugelmodelle und die Arbeitstische als Quader an. Aufgrund des Kollisionsmodels entfällt einerseits der Bereich, in dem sich die Tische befinden, andererseits ist vor allem an den Rändern des gemeinsamem Arbeitsraumes die Erreichbarkeit reduziert. Als Parameter für die Erstellung wird eine Voxelkantenlänge von $l_\mathrm{R} = 50$ mm, eine Anzahl an Kugelpunkten von $N_\mathrm{P} = 100$ und eine Anzahl an Drehrichtungen von $N_\mathrm{D} = 12$ als Kompromiss zwischen Genauigkeit und Berechnungszeit gewählt. Die durchschnittliche Berechnungszeit pro Voxel beträgt $\bar{t}_{\mathrm{V_{Coll}}} = 5,13$ s. Im Vergleich dazu liegt die durchschnittliche Berechnungszeit bei einer Berechnung ohne Kollisionsmodel bei $\bar{t}_\mathrm{V} = 5,99$ s pro Voxel und ist somit trotz des entfallenden Aufwandes für die Kollisionsberechnungen größer, was durch die Erhöhung der Anzahl an zu überprüfenden Posen begründet ist.

Die Berechnungen der *Reachability-Maps* für das zweite Robotersystem werden in die Offline-Programmiersoftware des Herstellers (siehe Abschnitt 6.2.2) integriert. Bei der Erzeugung der *Reachability-Maps* (siehe Bild 83) kommt das in der Software enthaltene Kollisionsmodell zum Einsatz. Aufgrund des Kollisionsmodels entfällt einerseits der Bereich unterhalb der Tischebene, andererseits ist vor allem in den durch die Roboterarme verdeckten Bereichen des gemeinsamem Arbeitsraumes



(a) Werkzeugbewegung (b) Werkstückbewegung (c) Koop. Bewegung

Bild 82: Auf den gemeinsamen Arbeitsbereich reduzierte Reachability-Maps für das erste Robotersystem



(a) Werkzeugbewegung (b) Werkstückbewegung (c) Koop. Bewegung

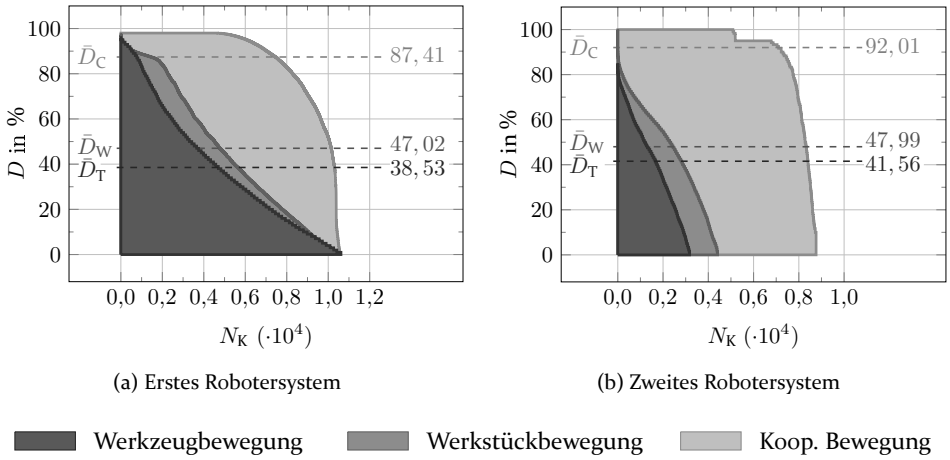
Bild 83: Auf den gemeinsamen Arbeitsbereich reduzierte Reachability-Maps für das zweite Robotersystem

die Erreichbarkeit reduziert. Die für die Untersuchung gewählte Voxelkantenlänge beträgt $l_R = 50$ mm, die Anzahl an Kugelpunkten $N_P = 20$ und die Anzahl an Drehrichtungen $N_D = 6$. Die durchschnittliche Berechnungszeit pro Voxel beträgt $\bar{t}_{V_{Coll}} = 20,39$ s. Demgegenüber liegt die durchschnittliche Berechnungszeit bei einer Berechnung ohne Kollisionsmodel bei $\bar{t}_V = 9,17$ s pro Voxel und ist somit trotz einer reduzierten Anzahl an Posen geringer. Das zur Untersuchung des ersten Robotersystems gegenläufige Verhalten liegt in der Verwendung eines detaillierteren Kollisionsmodelles begründet, das mit zeitaufwendigeren Kollisionsberechnungen verknüpft ist.

Auswertung

Im Sinne einer besseren Gegenüberstellung werden die Daten der generierten *Reachability-Maps* numerisch dargestellt. Hierfür werden die *Reachability-Indizes* für die drei Bewegungsarten der Größe nach gestaffelt in einem gemeinsamen Diagramm (siehe Bild 84) aufgetragen.

Die für das erste Robotersystem erzeugten *Reachability-Maps* enthalten etwa $N_K \approx 10.500$ erreichbare Kugeln. Die Mittelwerte für den *Reachability-Index* liegen bei $\bar{D}_T \approx 39\,\%$ bei einer Werkzeugbewegung, $\bar{D}_W \approx 47\,\%$ bei einer Werkstückbewegung und $\bar{D}_C \approx 87\,\%$ bei einer kooperativen Bewegung. Auffällig ist eine vergleichsweise große Anzahl von hohen Indizes für die kooperative Bewegung. Des Weiteren ist die Streuung der Reachability-Indizes bei einer kooperativen Bewegung geringer. So beträgt die Standardabweichung bei einer Werkzeugbewegung $\sigma(D_T) \approx 27,0\,\%$, bei einer Werkstückbewegung $\sigma(D_W) \approx 29,4\,\%$ und bei einer kooperativen Bewegung $\sigma(D_C) \approx 18,1\,\%$. Zusammenfassend lässt sich sagen, dass für



(a) Erstes Robotersystem

(b) Zweites Robotersystem

Bild 84: Der Größe nach geordnete Reachability-Indizes für die untersuchten Robotersysteme

den betrachteten Roboteraufbau die Erreichbarkeit bei einer kooperativen Bewegung um etwa 127 % besser ist als bei einer reinen Werkzeugbewegung. Gegenüber der reinen Werkstückbewegung ist die Erreichbarkeit um etwa 86 % verbessert.

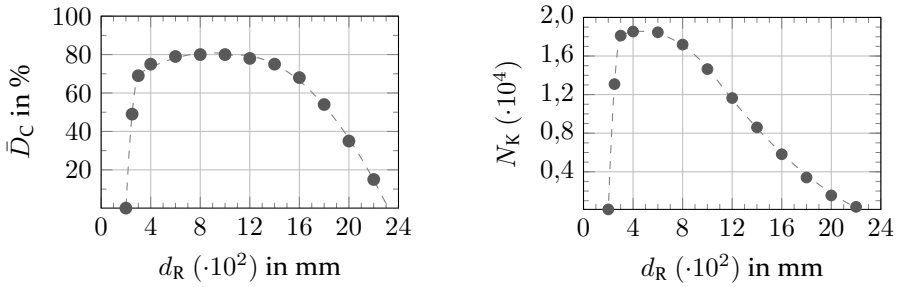
Im Gegensatz zum ersten Robotersystem variieren die *Reachability-Maps* für das zweite Robotersystem in der Anzahl an erreichbaren Kugeln signifikant. So beträgt $N_K \approx 8.800$ bei einer kooperativen Bewegung und 3.200 bzw. 4.400 bei einer reinen Werkzeug- bzw. Werkstückbewegung. Dies entspricht einer Erhöhung von 175 % bzw. 100 %. Dieser deutliche Unterschied begründet sich darin, dass in der unmittelbaren Nähe der beiden Roboterarme die Grundstellung des statischen Roboterarms die jeweilige Erreichbarkeit des anderen Roboterarms stark einschränkt. Die Mittelwerte für den *Reachability-Index* sind jedoch mit $\bar{D}_T \approx 42\,\%$ bei einer Werkzeugbewegung, $\bar{D}_W \approx 48\,\%$ bei einer Werkstückbewegung und $\bar{D}_C \approx 92\,\%$ bei einer kooperativen Bewegung vergleichbar zu den Werten des ersten Robotersystems. Auch hier zeigt sich deutlich eine vergleichsweise große Anzahl von hohen Indizes für die kooperative Bewegung. Zudem ist die Streuung der Reachability-Indizes bei einer kooperativen Bewegung geringer als bei den Einzelbewegungen. So beträgt die Standardabweichung bei einer Werkzeugbewegung $\sigma(D_T) \approx 21,1\,\%$, bei einer Werkstückbewegung $\sigma(D_W) \approx 21,9\,\%$ und bei einer kooperativen Bewegung $\sigma(D_C) \approx 17,6\,\%$. Mit Bezug auf die Orientierung ist die Erreichbarkeit bei einer kooperativen Bewegung etwa 119 % besser ist

als bei einer reinen Werkzeugbewegung. Gegenüber der reinen Werkstückbewegung ist die Erreichbarkeit um etwa 92 % verbessert.

Einflussgrößen

Im Rahmen dieser Untersuchung wird der grundsätzliche Einfluss der wesentlichen Variationsfaktoren auf die Erreichbarkeit ermittelt. Hierzu zählen die Variation des Abstandes zwischen den beiden Roboterarmen und des Abstandes des TCP sowie der Werkstückbasis gegenüber dem jeweiligen Flansch. Zudem wird die Anzahl der werkstückseitigen rotatorischen Freiheitsgrade als Variationsfaktor betrachtet. Für die Berechnung der entsprechenden Reachability-Maps wird eine Voxelkantenlänge von $l_\mathrm{R} = 50$ mm, eine Anzahl an Kugelpunkten $N_\mathrm{P} = 20$ und eine Anzahl an Drehrichtungen $N_\mathrm{D} = 6$ gewählt.

Der Abstand zwischen den beiden Robotern d_R wird in mehreren Stufen variiert und aus den resultierenden Reachability-Maps der Mittelwert des Reachability-Index $\bar{D}_\mathrm{C}$ sowie die Summe der erreichbaren Kugeln N_K ermittelt (siehe Bild 85). Der minimale Abstand, der mit dem System möglich ist, liegt über einem Wert von $d_\mathrm{R} = 200$ mm. Eine Überschneidung des Arbeitsbereiches liegt bis zu einem Abstand von $d_\mathrm{R} = 2200$ mm vor. Mit abnehmendem Abstand steigt der Mittelwert des Reachability-Index $\bar{D}_\mathrm{C}$ schnell an, bis der Anstieg ab einem Wert von etwa $d_\mathrm{R} = 1400$ mm stagniert. Ab einem Abstand von $d_\mathrm{R} = 800$ mm nimmt der Mittelwert wieder ab und bei $d_\mathrm{R} = 300$ mm knickt er schlagartig ein. Auch für die Anzahl der erreichbaren Kugeln N_K in Abhängigkeit vom Roboterabstand zeigt sich unterhalb dieser Distanz ein ähnliches verhalten. Der Grund hierfür ist, dass sich die beiden Roboterarme bei einer verhältnismäßig nahen Positionierung zueinander



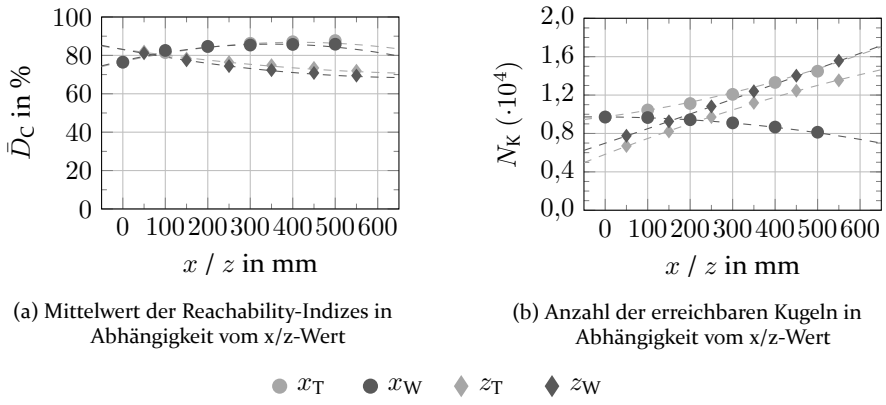
(a) Mittelwert der Reachability-Indizes in Abhängigkeit vom Roboterabstand

(b) Anzahl der erreichbaren Kugeln in Abhängigkeit vom Roboterabstand

Bild 85: Einfluss des Abstands der Roboter auf die Erreichbarkeit

gegenseitig in ihrer Bewegungsfreiheit einschränken. Das Maximum der Anzahl an erreichbaren Kugeln liegt in einem Bereich zwischen $d_R = 400$ mm und 600 mm. Mit zunehmendem Abstand nimmt die Anzahl zunächst schnell und dann immer weniger schnell ab, bis sie bei einem Abstand von etwa $d_R = 2200$ mm ihren Tiefpunkt erreicht. Zusammenfassend lässt sich sagen, dass das reale Robotersystem mit Bezug auf die theoretisch mögliche Erreichbarkeit nicht optimal ausgelegt ist. Der ideale Abstand der beiden Roboter liegt in einem Bereich um $d_R = 600$ mm. Die Auslegung des Systems kann aber durch andere Prozesse und Randbedingungen begrenzt sein, was bei diesem Wert nicht berücksichtigt wird.

Für die Ermittlung des Einflusses des TCP sowie der Werkstückbasis auf die Erreichbarkeit werden jeweils die x und die z-Werte variiert (siehe Bild 86). Dabei wird der x-Wert mit einer Schrittweite von 100 mm von $x = 0$ mm bis $z = 500$ mm erhöht. Der z-Wert hingegen beginnt mit einem Wert von $z = 50$ mm, da ein Werkzeug mit einem TCP im Nullpunkt der z-Achse technisch nicht möglich ist. Die Schrittweite beträgt ebenfalls 100 mm und der z-Wert variiert bis zu einem Wert von $z = 550$ mm. Der Mittelwert des Reachability-Index $\bar{D}_C$ zeigt sowohl bei einer Veränderung des x-Werts als auch bei einer Veränderung des z-Werts für Werkzeug und Werkstück ein vergleichbares und nicht stark ausgeprägtes Verhalten. Mit zunehmendem x-Wert steigt der Index langsam an bis zu einem Wert von etwa $x = 400$ mm und fällt dann wieder langsam ab. Die Vergrößerung des z-Wertes hingegen führt unmittelbar zu einer leichten Reduzierung des Indizes. Der Einfluss auf die Summe der erreichbaren Kugeln N_K hingegen fällt signifikanter und teilweise auch nicht vergleichbar aus. Mit steigendem x-Wert des TCP



(a) Mittelwert der Reachability-Indizes in Abhängigkeit vom x/z-Wert

(b) Anzahl der erreichbaren Kugeln in Abhängigkeit vom x/z-Wert

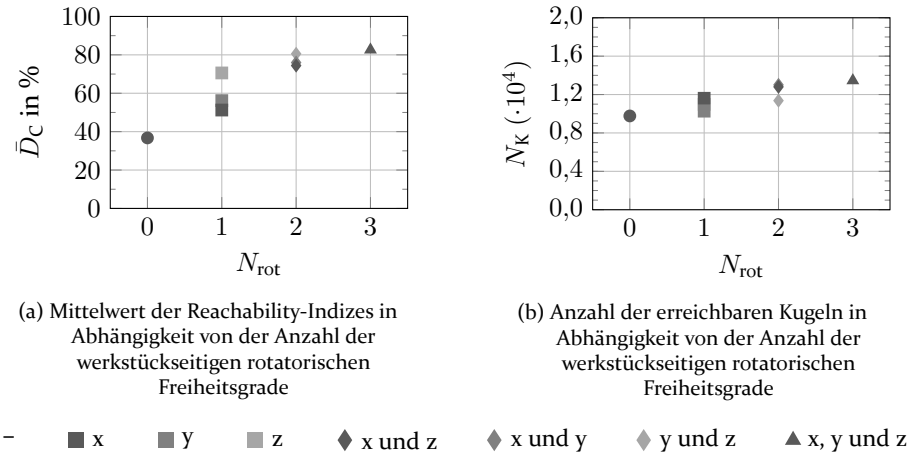
Bild 86: Einfluss des TCP sowie der Werkstückbasis auf die Erreichbarkeit

nimmt die Anzahl stark zu, beim Werkstück hingegen nimmt sie stark ab. Die Erhöhung des z-Werts führt in beiden Fällen zu einer stark steigenden Anzahl.

In einer dritten Untersuchungsreihe wird die Anzahl der werkstückseitigen rotatorischen Freiheitsgrade variiert (siehe Bild 87). Dabei gibt es den Fall, dass werkstückseitig keine Rotation eingesetzt wird, was sowohl zu einem deutlich niedrigeren Mittelwert des Reachability-Index $\bar{D}_C$ als auch zu einer reduzierten Anzahl der erreichbaren Kugeln N_K führt. Kommt ein rotatorischer Freiheitsgrad zum Einsatz, so steigt beides an. Auffällig ist der deutlich größere Anstieg des durchschnittlichen Reachability-Index bei einer Rotation um die z-Achse. Die Erweiterung auf zwei rotatorische Freiheitsgrade erhöht beide Kenngrößen, das Maximum wird aber in beiden Fällen mit dem Einsatz aller Freiheitsgrade erreicht. Dies verdeutlicht den Vorteil des im Rahmen dieser Arbeit entwickelten Ansatzes.

7.1.4 Kinematische Kalibrierung der verwendeten Robotersysteme

Das um die Werkstückerfassung erweiterte Robotersystem wird des Weiteren anhand des erarbeiteten Ansatzes (siehe Abschnitt 4.3.3) auf seine relative Positioniergenauigkeit hin untersucht. Die anwendungsübergreifenden Komponenten sind die beteiligten Roboter und deren Positionierung. Hierfür erfolgt eine *kinematische Kalibrierung* des verwendeten Robotersystems, mit der die Bearbeitungsgenauigkeit signifikant erhöht wird



(a) Mittelwert der Reachability-Indizes in Abhängigkeit von der Anzahl der werkstückseitigen rotatorischen Freiheitsgrade

(b) Anzahl der erreichbaren Kugeln in Abhängigkeit von der Anzahl der werkstückseitigen rotatorischen Freiheitsgrade

Bild 87: Einfluss der werkstückseitigen rotatorischen Freiheitsgrade auf die Erreichbarkeit

[P1]. Diese erfolgt gemäß [36] in vier chronologischen Schritten: *Modellierung*, *Messungen*, *Identifizierung* und *Kompensation*, deren Umsetzung weitergehend beschrieben ist.

Modellierung

Für die Untersuchung des betrachteten Robotersystems wird ein Laser-Tracker des Typs *API R-20 Radian* verwendet (siehe Bild 88), mit dem laut Herstellerangaben Messungen mit einer Genauigkeit von ± 10 µm möglich sind [8]. Das Messgerät wird für die Messungen mittig vor die beiden Roboter positioniert, um den gesamten gemeinsamen Arbeitsbereich abzudecken. Beide Roboter haben für die Messung einen Reflektor mittels Adapterplatte am Flansch montiert. Da die Adapterplatte als Drehteil ausgeführt ist, wird von einer verhältnismäßig hohen Fertigungsgenauigkeit in radialer Richtung ausgegangen. Bezüglich des Versatzes in z-Richtung ist jedoch mit einer geringeren Fertigungsgenauigkeit zu rechnen, weshalb dieser mittels Kalibrierung bestimmt werden muss, um eine ausreichende Genauigkeit der Untersuchung zu gewährleisten. Der Versatz des Reflektors in z-Richtung z_R wird bestimmt, indem der Reflektor durch den Roboter um dessen Flansch geneigt wird (siehe Bild 89a). Hierdurch wird der Reflektor entlang einer zirkularen Bahn bewegt, die mit dem Laser-Tracker



Bild 88: Aufbau für die Genauigkeitsuntersuchung am kooperativen Robotersystem mittels Laser-Tracker



(a) Flansch (b) 1. Achse (c) 2. Achse

Bild 89: Für die mittels Laser-Tracker durchgeführten Positionsmessungen und Kalibrierungen verwendete Reflektoren

aufgenommen werden kann. Der Versatz in z-Richtung entspricht dem Radius der resultierenden Kreisbahn. Für die an die beiden Roboter montierten Reflektoren ergeben sich somit die Werte $z_{R_T} \approx 36,79$ mm und $z_{R_W} \approx 36,71$ mm.

Des Weiteren müssen die Basiskoordinatensysteme der beiden Roboter erfasst werden. Hierfür werden Reflektoren an den rotierenden Teilen der ersten (siehe Bild 89b) und der zweiten Achse der Roboter (siehe Bild 89c) montiert. Für beide Achsen wird jeweils eine Messung während einer Achsrotation durchgeführt und die resultierenden Messdaten bilden eine zirkulare Bahn, deren Mittelpunkt auf der jeweiligen Rotationsachse liegt. Die Lage der Roboterbasis kann anhand der Schnittpunkte der beiden Achsen ermittelt werden. Aus den so ermittelten Basiskoordinatensystemen kann die Basistransformation zwischen beiden Robotern berechnet werden (siehe Abschnitt 4.3.3). Es resultiert für das betrachtete Robotersystem eine Translation der Basis von $\mathbf{t}_B \approx (3,18 \quad -1319,28 \quad 0,17)^T$ in mm sowie eine Rotation der Basis von $\mathbf{r}_B \approx (0,00 \quad 0,01 \quad 0,26)^T$ in °.

Messungen

Auf die zuvor definierte Weise (siehe Abschnitt 4.3.3) wird für beide Roboter jeweils ein Programm erzeugt, mit dem der Roboter den Reflektor der Reihe nach an alle raumbezogenen Messpositionen bewegt. Der resultierende Bahnverlauf deckt den gemeinsamen Arbeitsraum der beiden Roboter ab und enthält 472 Messpositionen (siehe Bild 90). Nach jedem Bewegungsbefehl wird dem Programm noch eine Zeitverzögerung hinzugefügt, um eine positionsgetreue Messung zu gewährleisten.

Aus den so erfassten Positionsdaten können gemäß Gleichung 4.32 und Gleichung 4.33 die Absolutfehler berechnet werden. Um eine bessere Übersicht über die Absolutgenauigkeit der Roboter zu erhalten, werden die Absolutfehler der beiden Roboter in Diagrammen sortiert

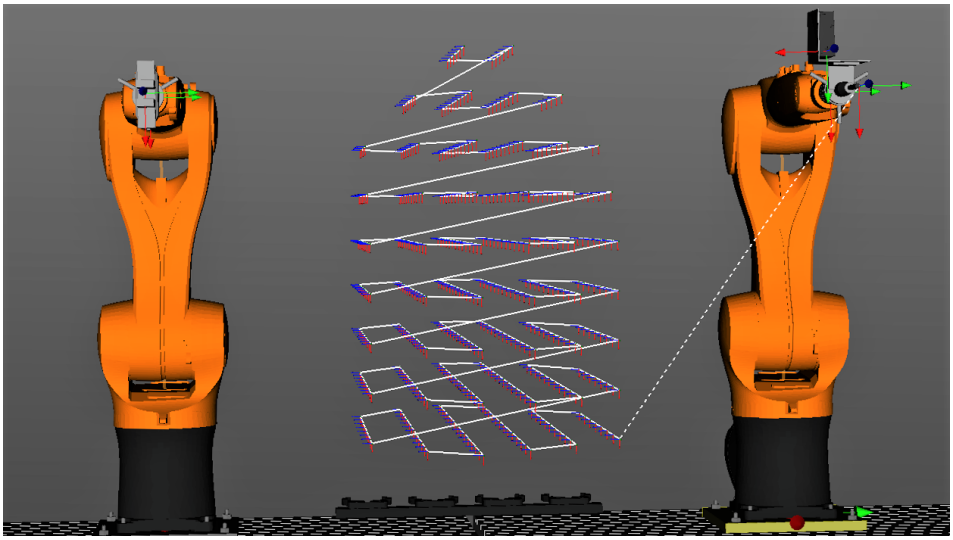


Bild 90: Für die Genauigkeitsuntersuchung erzeugte Roboterbahn des werkzeugführenden Roboters

nach Positionsnummer dargestellt (siehe Bild 91). Hierbei werden die drei Achsrichtungen $(x_{\mathrm{E}}, y_{\mathrm{E}}, z_{\mathrm{E}})$ sowie der Fehlerbetrag ϵ separat abgebildet. Die Verteilung der resultierenden Absolutabweichungen wird tabellarisch zusammengefasst (siehe Tabelle 11). Für beide Roboter zeigen sich vergleichbare Abweichungen der Absolutgenauigkeit, die verhältnismäßig groß sind und um mehrere Millimeter innerhalb des betrachteten Arbeitsraumes schwanken. Die Abweichungen in y-Richtung sind dabei am signifikantesten.

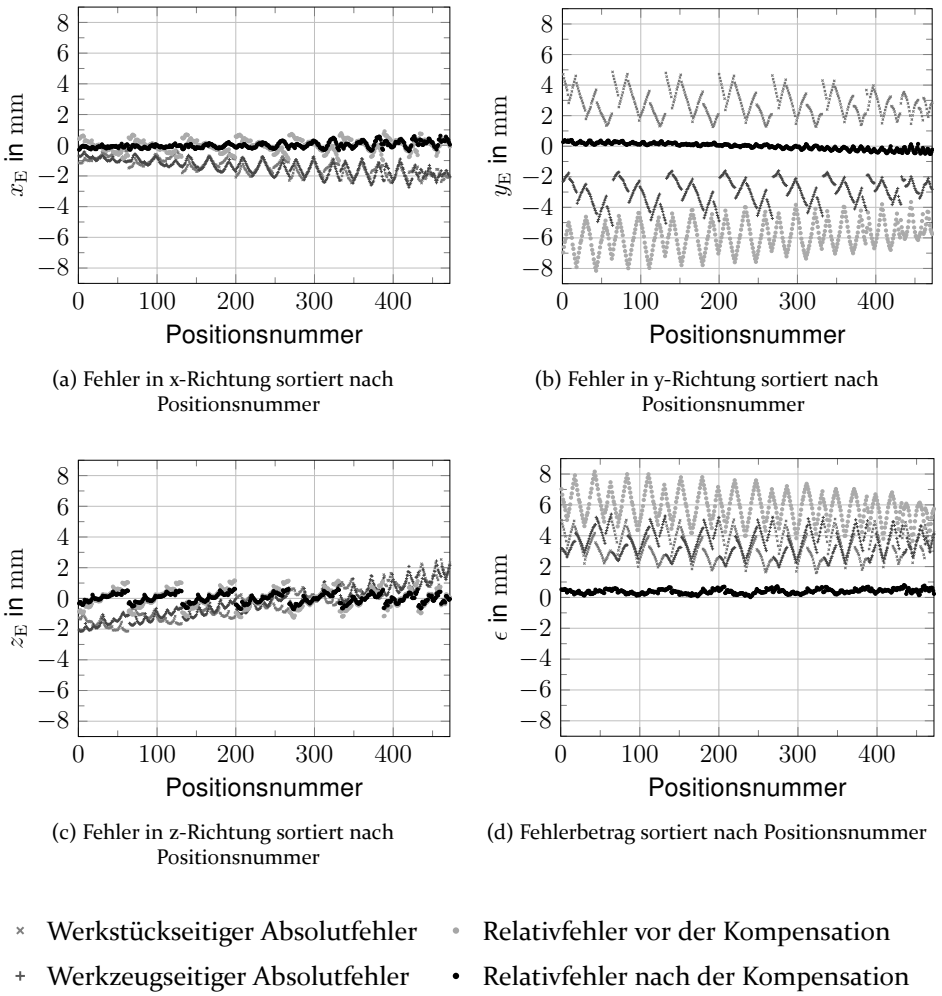
Tabelle 11: Verteilung der absoluten Positionsfehler der beiden Roboterarme

(a) Werkstückführender Roboter

	$x_{\mathrm{E}_{\mathrm{abs}}}$ in mm	$y_{\mathrm{E}_{\mathrm{abs}}}$ in mm	$z_{\mathrm{E}_{\mathrm{abs}}}$ in mm	ϵ_{abs} in mm
$\bar{\mathbf{e}}_{\mathrm{abs}}$	$-1,3763$	$2,8081$	$-0,2560$	$3,3459$
$\min(\mathbf{e}_{\mathrm{abs}})$	$-2,4671$	$1,2709$	$-2,1381$	$1,5914$
$\max(\mathbf{e}_{\mathrm{abs}})$	$-0,0739$	$4,8651$	$2,5341$	$5,2143$
$\sigma(\mathbf{e}_{\mathrm{abs}})$	$0,5322$	$0,8458$	$1,0446$	$0,8587$

(b) Werkzeugführender Roboter

	$x_{\mathrm{E}_{\mathrm{abs}}}$ in mm	$y_{\mathrm{E}_{\mathrm{abs}}}$ in mm	$z_{\mathrm{E}_{\mathrm{abs}}}$ in mm	ϵ_{abs} in mm
$\bar{\mathbf{e}}_{\mathrm{abs}}$	$-1,4241$	$-2,9930$	$-0,2557$	$3,5153$
$\min(\mathbf{e}_{\mathrm{abs}})$	$-2,7368$	$-5,1784$	$-2,1482$	$1,9679$
$\max(\mathbf{e}_{\mathrm{abs}})$	$-0,4436$	$-1,6072$	$2,4348$	$5,3951$
$\sigma(\mathbf{e}_{\mathrm{abs}})$	$0,4782$	$0,8052$	$1,0273$	$0,7900$



(a) Fehler in x-Richtung sortiert nach Positionsnummer

(b) Fehler in y-Richtung sortiert nach Positionsnummer

(c) Fehler in z-Richtung sortiert nach Positionsnummer

(d) Fehlerbetrag sortiert nach Positionsnummer

Bild 91: Fehlerwerte an den einzelnen Messpositionen

Identifizierung

Anhand der für beide Roboter gemessenen Positionen können die Abweichungen zueinander, gemäß Gleichung 4.32 und Gleichung 4.33, bestimmt werden. Die Abweichungen an allen Messpositionen werden separiert nach Achsrichtung und entsprechend einer Heatmap eingefärbt visualisiert (siehe Bild 92). Hierin lassen sich bereits allgemeine Tendenzen der Abweichungen erkennen. In x- und z-Richtung sind leichte Rotationen zu erkennen, wohingegen in y-Richtung ein zusätzlicher großer Versatz von mehreren Millimetern markant ist.

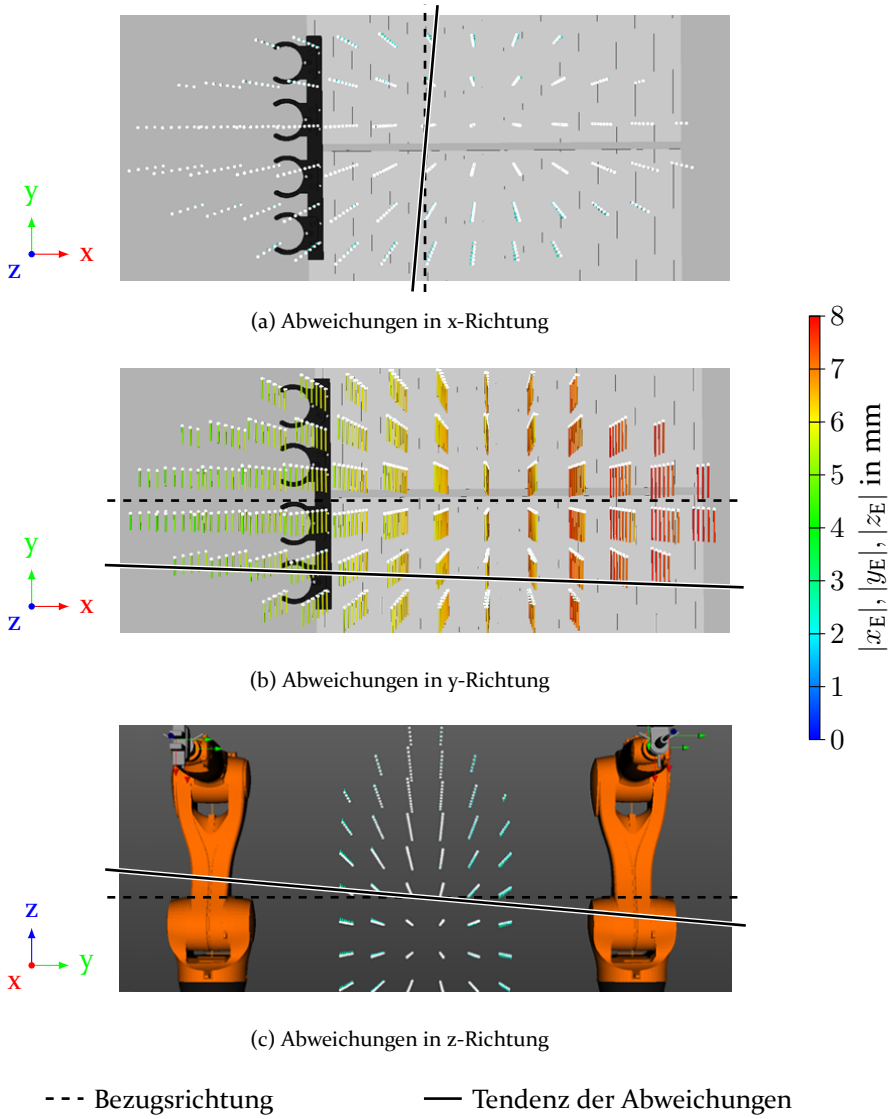


Bild 92: Karten der relativen Positioniergenauigkeit des kooperativen Robotersystems mit skalierten Abweichungen, die in Abhängigkeit des Betrags gemäßeiner Heatmap eingefärbt sind

Um eine bessere Übersicht über die Abweichungen zu erhalten, werden die Fehlerwerte ebenfalls in den Diagrammen sortiert nach Positionsnummer dargestellt (siehe Bild 91). Die Verteilung der resultierenden Fehler vor der Kompensation wird in einer Tabelle zusammengefasst (siehe

Tabelle 13a). Die Abweichungen in x- und z-Richtung liegen in einem Bereich zwischen etwa $-1,00$ mm und $1,00$ mm und sind somit geringer als die Absolutabweichungen der beiden Roboter, was in der Gleichläufigkeit der Absolutabweichungen der beiden Roboter in dieser Richtung begründet ist. In y-Richtung hingegen sind die Absolutabweichungen gegenläufig, weshalb mit Werten von bis zu über $8,00$ mm eine deutlich größere Abweichung in dieser Richtung vorliegt. Diese ist zudem maßgeblich für den verhältnismäßig großen Fehlerbetrag. Die Standardabweichungen vor der Kompensation liegen in einem Bereich zwischen $0,40$ mm und $0,96$ mm.

Kompensation

Gemäß dem zuvor beschriebenen Ansatz (siehe Abschnitt 4.3.3) wird für das betrachtete Robotersystem eine Transformation zur Kompensation der Abweichungen ermittelt. Diese besteht aus einer Translation $\mathbf{t}_C \approx (-1,81 \quad 6,96 \quad 1,41)^T$ in mm und einer Rotation $\mathbf{r}_C \approx (0,16 \quad 0,00 \quad 0,12)^T$ in °. Mit diesen Paramtern kann die zuvor bestimmte Basistransformation angepasst werden. Die Verteilung der resultierenden Abweichungen nach der Kompensation ist ebenfalls in den Diagrammen sortiert nach Positionsnummer (siehe Bild 91) sowie in einer Tabelle (siehe Tabelle 13b) zusammengefasst. Der gemittelte Fehlervektor tendiert in Richtung Nullvektor und alle Abweichungen sind kleiner als $0,80$ mm. Die Standardabweichungen nach der Kompensation liegen alle zwischen $0,14$ mm und $0,31$ mm. Für einen besseren Vergleich der Verteilungen vor und nach der Kompensation werden diese in einem Box-Whisker-Plot dargestellt (siehe Bild 93). Es ist eine signifikante Reduzierung der Abweichungen durch die Kompensation zu erkennen.

Tabelle 13: Verteilung der relativen Positionsfehler für das kooperative Robotersystem

(a) Vor der Kompensation

	$x_{E_{rel}}$ in mm	$y_{E_{rel}}$ in mm	$z_{E_{rel}}$ in mm	ϵ_{rel} in mm
$\bar{\mathbf{e}}_{rel}$	$-0,0262$	$-5,8465$	$0,0087$	$5,8925$
$\min(\mathbf{e}_{rel})$	$-1,0007$	$-8,1463$	$-1,2508$	$3,6713$
$\max(\mathbf{e}_{rel})$	$0,9037$	$-3,6525$	$1,1629$	$8,1544$
$\sigma(\mathbf{e}_{rel})$	$0,4044$	$0,9559$	$0,6119$	$0,9547$

(b) Nach der Kompensation

	$x_{E_{rel}}$ in mm	$y_{E_{rel}}$ in mm	$z_{E_{rel}}$ in mm	ϵ_{rel} in mm
$\bar{\mathbf{e}}_{rel}$	$-0,0001$	$0,0001$	$0,0000$	$0,3905$
$\min(\mathbf{e}_{rel})$	$-0,3687$	$-0,5031$	$-0,7086$	$0,0223$
$\max(\mathbf{e}_{rel})$	$0,6136$	$0,4041$	$0,6213$	$0,7938$
$\sigma(\mathbf{e}_{rel})$	$0,1889$	$0,2088$	$0,3077$	$0,1456$

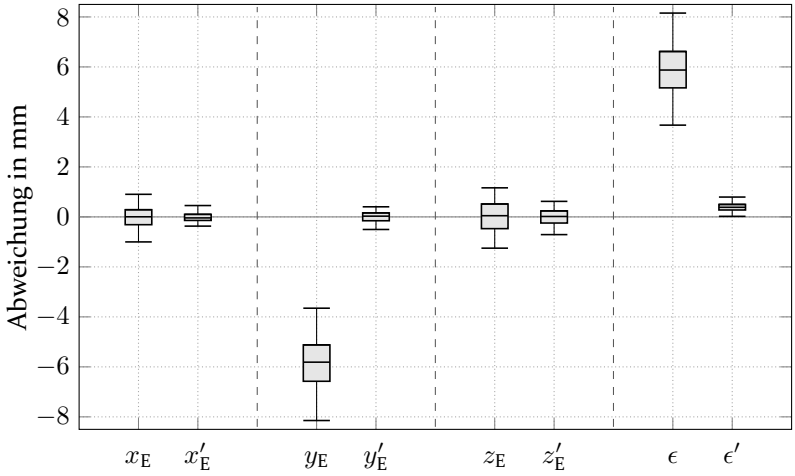


Bild 93: Box-Whisker-Plot der Positionsfehler für das kooperative Robotersystem

7.2 Anwendungsspezifische Untersuchungen

Die bisher in diesem Kapitel durchgeführten Untersuchungen beziehen sich allgemein auf die Robotersysteme. Weitergehend wird der Fokus auf Anwendungen, bei denen die entwickelten Ansätze konkrete Vorteile ermöglichen, gelegt (siehe Tabelle 15). Dabei kommen die beiden im vorherigen Kapitel beschriebenen Robotersysteme sowie die zugehörige Offline-Programmiersoftware zum Einsatz. Zunächst wird der Einfluss der kooperativen Bearbeitung auf die Bearbeitungszeit betrachtet (siehe Abschnitt 7.2.1). Hierfür werden mehrere exemplarische Prozesse anhand der unterschiedlichen Bearbeitungsprinzipien realisiert und anschließend miteinander verglichen. Im nächsten Abschnitt erfolgt eine Erschließung von Anwendungen durch die verbesserte Erreichbarkeit (siehe Abschnitt 7.2.2).

7.2.1 Bearbeitungszeitreduzierung durch kooperative Bearbeitung

Die in diesem Abschnitt betrachteten exemplarischen Anwendungen verdeutlichen, wie sich Prozesse durch die zusätzliche Einbindung eines zweiten Roboters hinsichtlich Bearbeitungszeit reduzieren lassen. Wenn eine Limitierung durch die Geschwindigkeit oder Achsreichweiten des einzelnen Bearbeitungsroboters gegeben ist, kann somit eine Produktivitätssteigerung erzielt werden. Im Falle, dass der zweite Roboter bereits als Transferroboter in der Anlage eingesetzt wird und folglich vorab bereits

Tabelle 15: Übersicht über die anwendungsspezifischen Untersuchungen

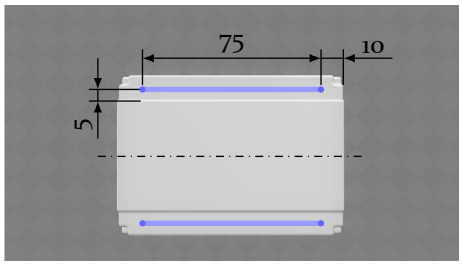
	1. Robotersystem				2. Robotersystem	
Anwendung	Applikation von Klebstoff	Zeichnen von Mäander	Polieren von Zuziehgriffblenden	Inspektion von Gewindestangen	Montage von Sprinklersystemen	Markieren von Kunststoffleisten
Bearbeitungszeitreduzierung	✓	✓	✓	–	✓	–
Verbesserte Erreichbarkeit	–	–	–	✓	–	✓

vorhanden ist, entstehen hierfür keine zusätzlichen Investitionen, sondern lediglich der Programmablauf muss dementsprechend angepasst werden.

Weitergehend werden vier Anwendungen betrachtet: *Applikation von Klebstoff*, *Zeichnen von Mäander*, *Polieren von Zuziehgriffblenden* sowie *Montage von Sprinklersystemen*. Diese werden mit dem entsprechenden Robotersystemen jeweils mittels reiner Werkzeugbewegung, reiner Werkstückbewegung und als kooperativer Prozess realisiert. Anschließend erfolgt das Ausführen der erzeugten Roboterprogramme am verwendeten Robotersystem sowie die Erfassung der dabei resultierenden Bearbeitungszeiten, die letztlich für einen Vergleich herangezogen werden.

Applikation von Klebstoff

Ziel dieser Anwendung ist es, zwei gleichförmige Klebstoffraupen auf zwei gegenüberliegende Absätze eines Kunststoffgehäuses zu applizieren (siehe Bild 94a). Das Gehäuse ist als Spritzgussteil produziert und besteht aus schwarzem *Acrylnitril-Butadien-Styrol-Copolymere* (ABS) (siehe Bild 94b). Es ist ein CAD-Modell des Kunststoffgehäuses vorhanden, auf dessen Grundlage eine Programmierung erfolgen kann.



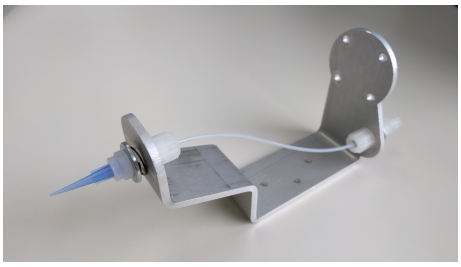
(a) Klebstoffauftrag



(b) Kunststoffgehäuse



(c) Dosiergerät



(d) Dosierwerkzeug mit Düse

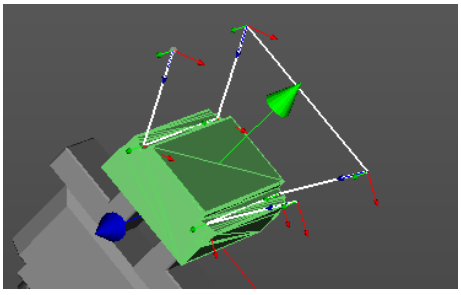
Bild 94: Komponenten für die Applikation von Klebstoff

Für diese Anwendung wird ein Dosiersystem in das erste Robotersystem integriert (siehe Bild 94c). Dieses basiert auf einem peristaltischen Dosiergerät vom Typ *PPD-2005* des Herstellers *Globaco GmbH*, mit dem Fluide mit Viskositäten von bis zu 10.000 mPa · s gezielt dosiert werden können [40]. Angesteuert wird das Dosiergerät über eine integrierte RS232-Schnittstelle. Via *Polytetrafluorethylen* (PTFE)-Schlauchsystem wird das Fluid an den Endeffektor des Roboters transportiert. Dort endet das Schlauchsystem in einer mittels Luer-Lock-System befestigten Düse (siehe Bild 94d). Aufgrund des verwendeten Schnellverschlusssystems können die Düsen flexibel an die geforderte Dosierung angepasst werden.

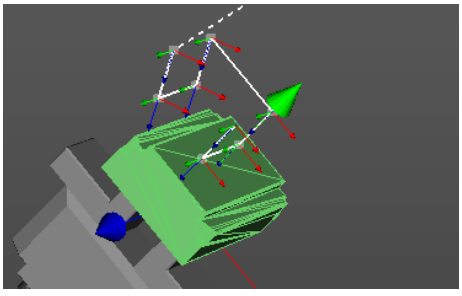
Während des Dosierprozesses sollte keinerlei Kontakt zwischen Werkzeug und Werkstück auftreten. Um eine gleichmäßige Klebstoffraupe garantieren zu können, ist es jedoch wichtig, einen gleichmäßigen Abstand der Düse gegenüber der Werkstückoberfläche einzuhalten. Das Werkstück muss während des Prozesses mit der Applikationsfläche stets waagerecht gehalten werden, damit der Klebstoff nicht übermäßig und ungleichmäßig verläuft. Des Weiteren ist es aus prozesstechnischen Gründen notwendig die Dosiernadel bei der Dosierung um $45°$ seitlich geneigt gegenüber der Oberfläche zu halten. Vor und nach den Applikationsbahnen muss die Düse entlang

der z-Achse zum Werkstück geführt bzw. vom Werkstück entfernt werden. Ferner wird das Dosiersystem zu Beginn der Applikationsbahnen aktiviert und am Ende wieder deaktiviert.

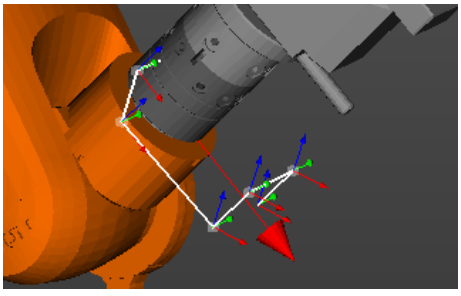
Aus den Prozessbedingungen wird ein werkstückbezogener Bahnverlauf für den Prozess abgeleitet (siehe Bild 95a). Hieraus werden jeweils sowohl die Programme für eine kooperierende Bearbeitung (siehe Bild 95b und Bild 95c) als auch die Programme für eine reine Werkzeug- und eine reine Werkstückbewegung mit der entwickelten Offline-Programmiersoftware erzeugt. Beim Ausführen der Programme am realen Robotersystem werden die zugehörigen Zeiten erfasst (siehe Bild 96). Für die Einzelbewegungen ergeben sich vergleichbare Bearbeitungszeiten (t_W und t_T). Der Vergleich zwischen Einzelbewegung und kooperativem Prozess (t_C) zeigt eine Reduzierung der Bearbeitungszeit von ca. 25 %. Hiermit liegt die Reduzierung der Bearbeitungszeit verhältnismäßig weit entfernt von der theoretisch möglichen Halbierung der Bearbeitungszeit. Dies liegt im Wesentlichen an den begrenzten Beschleunigungen der einzelnen Roboterachsen während des kooperativen Prozesses.



(a) Werkstückbezogener Bahnverlauf



(b) Bahn des werkzeugführenden Roboters



(c) Bahn des werkstückführenden Roboters

Bild 95: Resultierende Bahnverläufe für die Applikation von Klebstoff

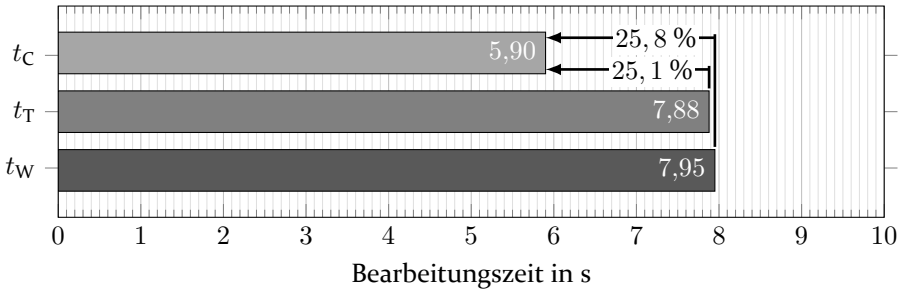
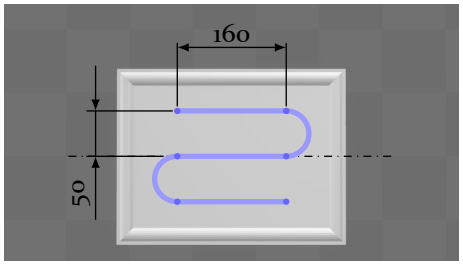


Bild 96: Bearbeitungszeiten für die Applikation von Klebstoff

Zeichnen von Mäander

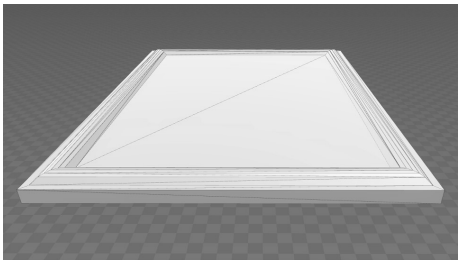
Bei der zweiten exemplarischen Anwendung soll eine mäanderförmige Linie auf ein Blatt Papier im Format DIN-A4 gezeichnet werden (siehe Bild 97a). Hierzu wird der werkstückführende Roboter mit einer Papierhalterung als Werkzeug ausgestattet (siehe Bild 97b), in die entsprechende Papierblätter eingespannt werden können. Aus der Konstruktion der Halterung existiert ein CAD-Modell (siehe Bild 97c), welches für die weitergehende Umsetzung verwendet werden kann.



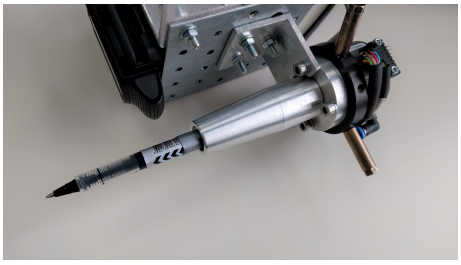
(a) Zeichnungsbahn



(b) Papier in Halterung



(c) CAD-Modell



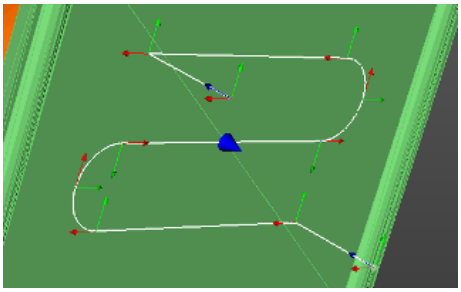
(d) Stifthalterung mit Stift

Bild 97: Komponenten für das Zeichnen von Mäander

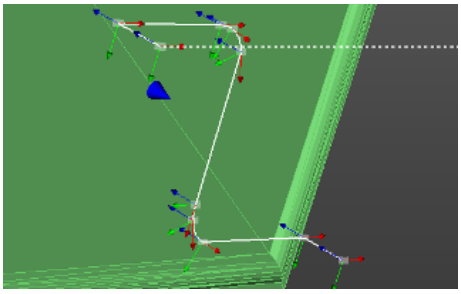
Für diese Anwendung wird als Werkzeug eine Stifthalterung entwickelt, mit der handelsübliche Stifte aufgenommen werden können (siehe Bild 97d). Diese ist über das Werkzeugwechselsystem am werkzeugführenden Roboter des ersten Robotersystems fixiert. Des Weiteren ist der Stift darin entlang der z-Achse gefedert gelagert, wodurch Ungenauigkeiten im Prozess kompensiert werden können.

Damit die für das Zeichnen geforderte mäanderförmige Linie ohne Verkanten des Stiftes gezeichnet werden kann, muss der Stift mit seiner Orientierung entlang der Linie geführt werden. Vor und nach dem Zeichnen muss der Stift senkrecht zur Zeichenfläche geführt bzw. von ihr entfernt werden. Während des Prozesses stellt sich durch Auslenkung der Feder ein gewisser Anpressdruck des Stiftes ein. Dieser kann durch eine Verschiebung des TCP entlang der z-Achse ggf. angepasst werden.

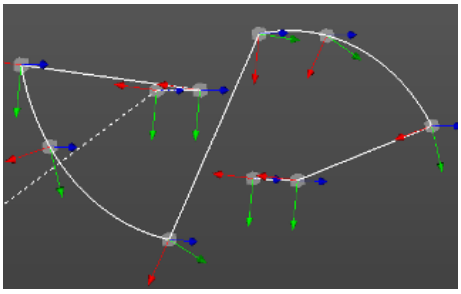
Gemäß den Prozessbedingungen erfolgt die Erstellung eines werkstückbezogenen Bahnverlaufes für den Prozess (siehe Bild 98a). Darauf aufbauend werden die Programme für eine kooperierende Bearbeitung (siehe Bild 98b und Bild 98c) sowie durch entsprechende Einzelbearbeitungen mittels



(a) Werkstückbezogener Bahnverlauf



(b) Bahn des werkzeugführenden Roboters



(c) Bahn des werkstückführenden Roboters

Bild 98: Resultierende Bahnverläufe für das Zeichnen von Mäander

der implementierten Ansätze erzeugt und am realen Robotersystem ausgeführt. Die dabei erfassten Zeiten (siehe Bild 99) zeigen ein vergleichbares Verhalten wie für die erste exemplarische Anwendung. Die Einzelbewegungen benötigen vergleichbare Bearbeitungszeiten (t_W und t_T) und der kooperative Prozess (t_C) zeigt eine Reduzierung der Bearbeitungszeit von etwa 25 %.

Polieren von Zuziehgriffblenden

Bei dieser Anwendung sollen Zuziehgriffblenden mit Aluminiumoberfläche (siehe Bild 100a) zielgerichtet poliert werden, wenn diese Oberflächendefekte aufweisen (siehe Bild 100b). Daher wird die im Rahmen dieser Arbeit entwickelte Werkstückerfassung (siehe Kapitel 5) eingesetzt, um das Erkennen von ggf. enthaltenen Defekten zu ermöglichen. Basierend auf den so erzeugten Daten (siehe Bild 100c) können individuelle Bahnverläufe für die Prozessbewegungen erzeugt werden, um die Defekt zu entfernen.

Für das Polieren der Aluminiumoberfläche wird am werkzeugführenden Roboter ein Multifunktionswerkzeug des Typs *Dremel Model 4000* montiert (siehe Bild 100d). An diesem können unterschiedliche Werkzeugaufsätze, wie beispielsweise für diese Anwendung eine Polierbürste, verwendet werden.

Die Oberfläche im Bereich eines Defektes muss mit gleichmäßiger Geschwindigkeit abgefahren werden, um diesen zu entfernen. Durch die Nachgiebigkeit der Bürste ist eine gewisse Ungenauigkeit unkritisch, jedoch gilt es zu beachten, ob der variierende Druck zu unzureichenden Prozessergebnissen führt. Der werkstückbezogene Bahnverlauf muss mit Bezug auf die Freiformoberfläche des Werkstückes erzeugt werden. Hierfür wird die in die Offline-Programmiersoftware implementierte Projektionsmethode

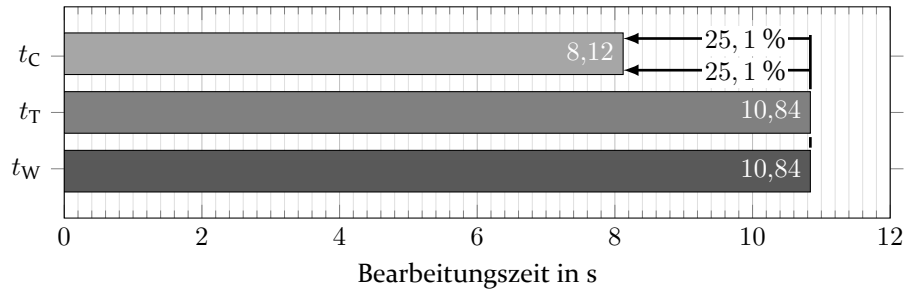
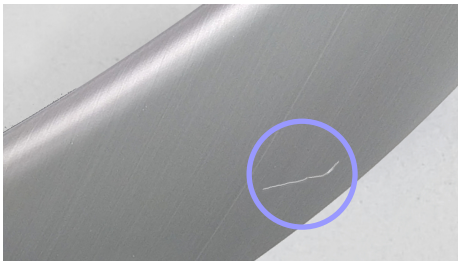


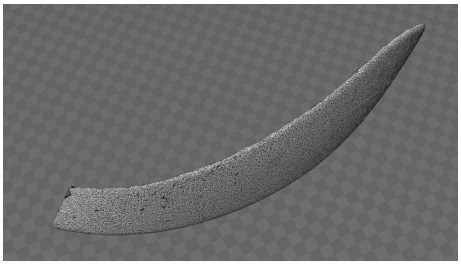
Bild 99: Bearbeitungszeiten für das Zeichnen von Mäander



(a) Zuziehgriffblende



(b) Defekt in der Oberfläche



(c) 3D-Scan

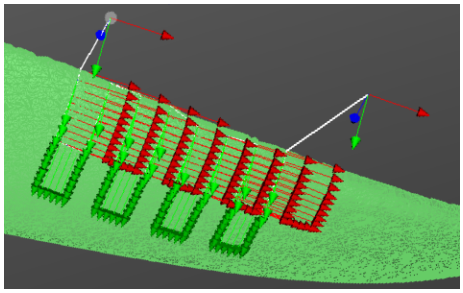


(d) Multifunktionswerkzeug mit Polierbürste

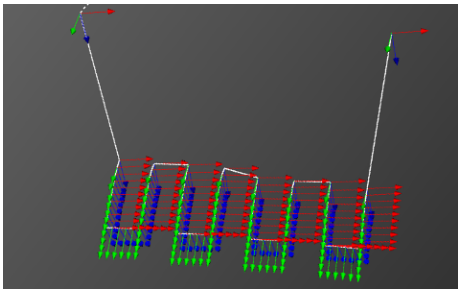
Bild 100: Komponenten für das Polieren von Zuziehgriffblenden

verwendet, mit der ein mäanderförmiger 2D-Bahnverlauf, der die ausgewählte Fläche ausfüllt, auf die Bauteiloberfläche projiziert wird. Als Fläche wird ein Bereich ausgewählt, welcher den Oberflächendefekt mit einem Abstand von etwa 10 mm umschließt. Auch bei diesem Prozess ist das Werkzeug vor der eigentlichen Prozessbahn dem Werkstück senkrecht zuzuführen sowie danach wiederum zu entfernen. Das Multifunktionswerkzeug wird während des Prozesses kontinuierlich mit konstanter Drehzahl betrieben.

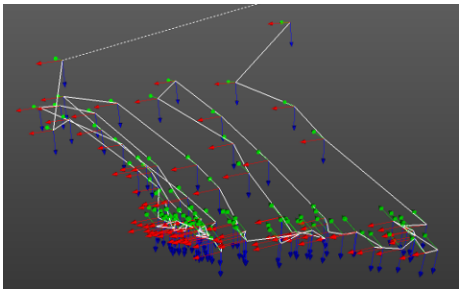
Auch für diese Anwendung erfolgt die Erstellung eines werkstückbezogenen Bahnverlaufes (siehe Bild 101a) sowie die Ableitung der Programme für eine kooperierende Bearbeitung (siehe Bild 101b und Bild 101c) als auch die Programme für eine reine Werkzeug- und eine reine Werkstückbewegung mit der entwickelten Offline-Programmiersoftware. Bei den beim Ausführen der Programme am realen Robotersystem bestimmten Bearbeitungszeiten (siehe Bild 102) zeigt sich erneut, dass für die Einzelbewegungen vergleichbare Bearbeitungszeiten (t_W und t_T) resultieren. Der Vergleich zwischen Einzelbewegung und kooperativem Prozess (t_C) zeigt eine Reduzierung der Bearbeitungszeit von 21 %. Hiermit liegt die Reduzierung der Bearbeitungszeit etwas unterhalb der Reduzierungen in den ersten beiden exemplarischen Anwendungen, was auf die Vielzahl an verhältnismäßig kurzen Bahnabschnitten zurückzuführen ist.



(a) Werkstückbezogener Bahnverlauf



(b) Bahn des werkzeugführenden Roboters



(c) Bahn des werkstückführenden Roboters

Bild 101: Resultierende Bahnverläufe für das Polieren von Zuziehgriffblenden

Montage von Sprinklersystemen

In dieser exemplarischen Anwendung erfolgt eine kollaborierende Montage eines Sprinklersystems (siehe Bild 103a) mit dem zweiten Robotersystem, das für MRK geeignet ist. Innerhalb dieser Anwendung wird das Aufschrauben der Abdeckkappe (siehe Bild 103b) durch den Roboter ausgeführt. Hierbei gilt es die Abdeckkappe mit zwei vollständigen Umdrehungen, also einer Drehung von 720°, auf das Gehäuse zu schrauben. Da die vorderste

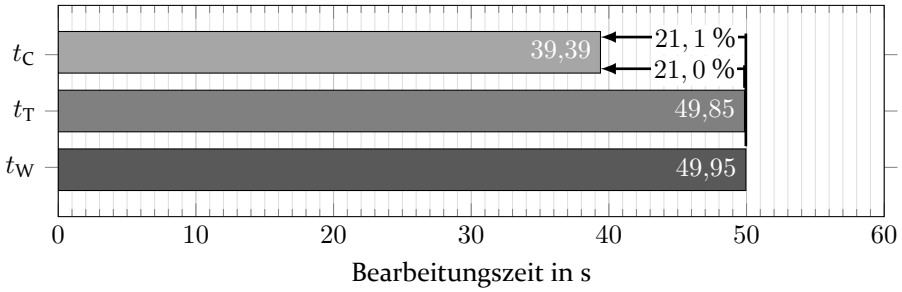


Bild 102: Bearbeitungszeiten für das Polieren von Zuziehgriffblenden



(a) Sprinklersystem



(b) Abdeckkappe



(c) Greifer für Gehäuse



(d) Greifer für Abdeckkappen

Bild 103: Komponenten für die Montage von Sprinklersystemen

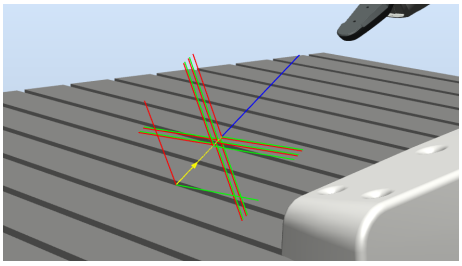
Roboterachse der verwendeten Roboterarme nur einen Arbeitsbereich von $\pm$ 229° hat, also eine Reichweite von 458°, muss der Roboter bei einer einseitigen Bewegung umgreifen, was mit entsprechenden Nebenzeiten verbunden ist. Durch den beidseitigen Einsatz der Roboterarme gemäß des Aufteilungsansatzes summiert sich die Reichweite für Prozessbewegung auf einen Wert von 916°, was die Montage ohne Umgreifen ermöglicht.

Bei dieser Anwendung kommen die Standardgreifer des verwendeten Robotersystems zum Einsatz. Diese werden mit Greiferbacken ausgerüstet (siehe Bild 103c und Bild 103d), mit denen auf der einen Seite die Gehäuse und auf der anderen Seite die Abdeckkappen aufgenommen werden können. Der Roboter, der das Gehäuse handhabt, wird als werkstückführender Roboter bezeichnet. Dementsprechend ist der anderer Roboter der werkzeugführende Roboter. Um ein Verdrehen der Bauteile innerhalb der geschlossenen Greiferbacken zu verhindern, sind diese mit zur Außenkontur der Bauteile passenden Kerben versehen.

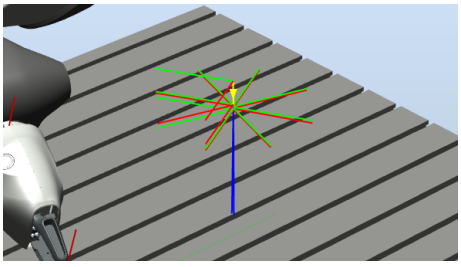
Ein wichtiger Aspekt bei dem Montagevorgang ist, dass der Inhalt des Sprinklersystems nicht fixiert ist. Daher ist es zwingend erforderlich, dass das Gehäuse stets aufrecht gehalten wird. Zudem muss bei der Montage der Abdeckkappen parallel zur Drehbewegung eine Translation von etwa

6 mm durchgeführt werden, um der Steigung des enthaltenen Gewindes zu folgen.

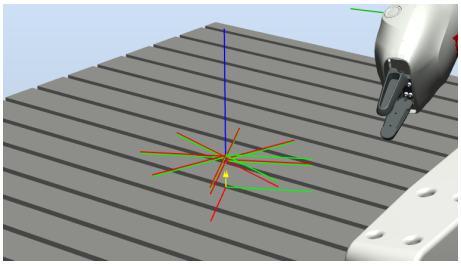
Für diese Anwendung erfolgt die Programmerzeugung mit dem in die herstellerspezifische Offline-Programmiersoftware integrierten Add-In (siehe Bild 104). Hierfür wird einerseits die werkstückbezogene Bahnverlauf definiert, der aus einer längeren Aufsetzbewegung sowie acht um jeweils 90° verdrehten Aufschraubbewegungen besteht. Bei der darauf folgenden Generierung der beidseitigen Bahnen gemäß des Aufteilungsansatzes wird die Werkstückorientierung gemäß der Vorgaben begrenzt, sodass die resultierenden Bahnen stets eine vertikale Orientierung des Sprinklersystems beinhalten. Die zudem erzeugten Programme für die Einzelbewegungen werden um entsprechende Umgriffbewegungen sowie Greiferbefehle erweitert. Beim anschließenden Ausführen der Roboterprogramme am realen Robotersystem werden die Bearbeitungszeiten erfasst (siehe Bild 105). Die Einzelbewegungen ergeben Bearbeitungszeiten (t_W und t_T), die in einer ähnlichen Größenordnung liegen. Der Vergleich zwischen Einzelbewegungen und kooperativem Prozess (t_C) zeigt jedoch signifikante Reduzierungen der Bearbeitungszeit von etwa 62 % bzw. 63 %. Die im Vergleich zu den anderen exemplarischen Anwendungen deutlich größeren Zeiteinsparungen sind durch den Wegfall des Umgreifens beim kooperativen Prozess begründet.



(a) Werkstückbezogener Bahnverlauf



(b) Bahn des werkzeugführenden Roboters



(c) Bahn des werkstückführenden Roboters

Bild 104: Resultierende Bahnverläufe für die Montage von Sprinklersystemen

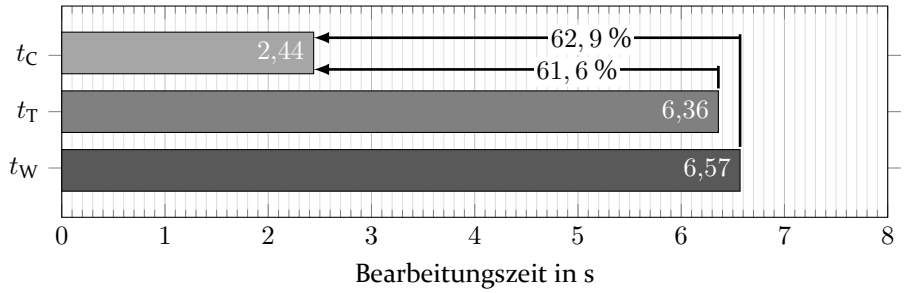


Bild 105: Bearbeitungszeiten für die Montage von Sprinklersystemen

7.2.2 Erschließung von Anwendungen durch verbesserte Erreichbarkeit

Die hier betrachteten exemplarischen Anwendungen zeigen, wie ein verhältnismäßig großes Werkstück komplett bearbeitet werden kann, indem die Erreichbarkeit durch den zusätzlichen Einsatz eines weiteren Roboters erhöht wird. Diese Anwendungen sind mit einem einzelnen Roboter nicht unter den geforderten Randbedingungen umsetzbar. Auch der Einsatz eines zusätzlichen Drehtisches genügt nicht, um die Anwendungen umzusetzen. Die Grenzen des Systems, die maximal zu bearbeitende Länge des Werkstücks, können anhand der längsten durchgehenden Diagonalen innerhalb des jeweiligen Arbeitsraumes (siehe Bild 106) abgeschätzt werden. Für einen

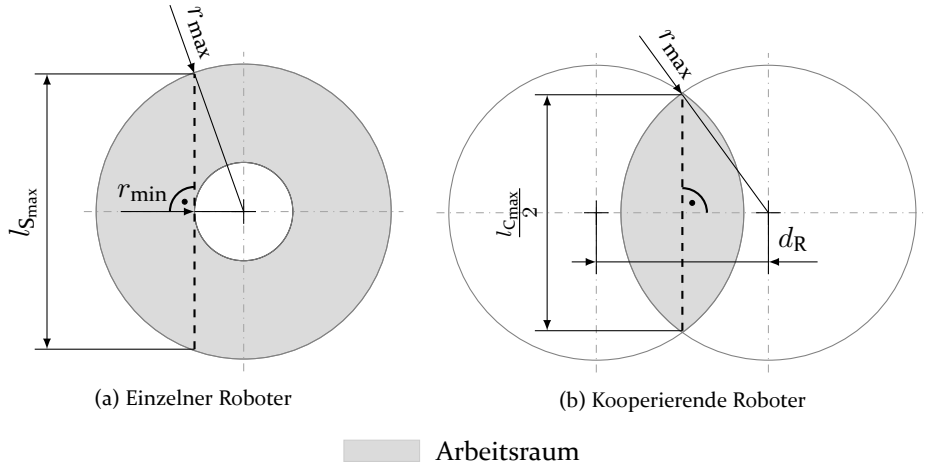


Bild 106: Maximale Diagonalen im Arbeitsraum

einzelnen Roboter ergibt sich ein Schätzwert für die maximale Bauteillänge $l_{S_{max}}$ über den Satz des Pythagoras gemäß:

$$l_{S_{max}} = 2 \cdot \sqrt{{r_{max}}^2 - {r_{min}}^2} \tag{7.2}$$

Diese wird durch die maximalen Reichweite des Roboters r_{max} und die minimale Reichweite r_{min} begrenzt. Beim zweiarmigen System kann die längste durchgehende Diagonale innerhalb des gemeinsamen Arbeitsraumes der beiden kooperierenden Roboter ebenfalls über den Satz des Pythagoras abgeschätzt werden. Der Schätzwert für die maximale Bauteillänge bei einer kooperativen Bearbeitung $l_{C_{max}}$ entspricht aber der doppelten Diagonalen, da eine entgegengesetzte Bewegung mit beiden Robotern erfolgt. Deshalb lautet die entsprechende Formel:

$$l_{C_{max}} = 4 \cdot \sqrt{{r_{max}}^2 - \left(\frac{d_R}{2}\right)^2} \tag{7.3}$$

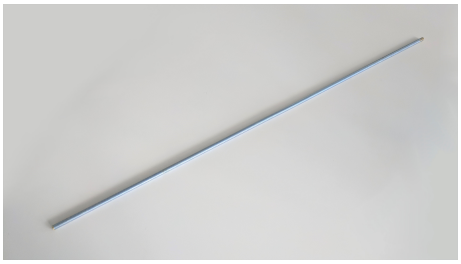
Die maximale Bauteillänge ist in diesem Fall nicht nur abhängig von der maximalen Reichweite r_{max} der eingesetzten Roboter, sondern auch vom Abstand d_R der beiden Roboter zueinander.

Weitergehend werden zwei exemplarische Anwendungen aufgezeigt, bei denen eine Erschließung durch die verbesserte Erreichbarkeit erfolgt. Hierbei handelt es sich um eine *Inspektion von Gewindestangen* und um ein *Markieren von Kunststoffleisten*.

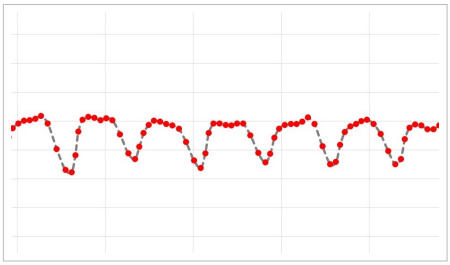
Inspektion von Gewindestangen

Ziel der hier betrachteten Anwendung ist die Überprüfung des Gewindeprofils von Gewindestangen mit einer Länge von $l = 2000$ mm (siehe Bild 107a). Hierfür muss das Gewindeprofil (siehe Bild 107b) mittels Laser-Profilsensor auf zwei Seiten erfasst und mit dem Sollprofil, das über ein CAD-Modell (siehe Bild 107c) definiert ist, verglichen werden. Anschließend können die Gewindestangen entsprechend ihrer Qualität sortiert werden.

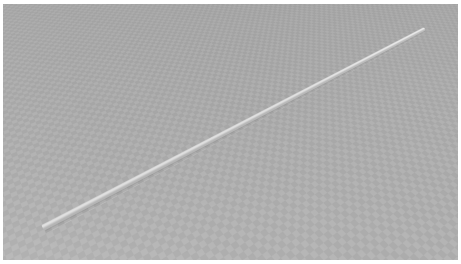
Für diese Anwendung kommt das erste Robotersystem inklusive der integrierten Werkstückerfassung zum Einsatz. Der werkstückseitige Roboter nimmt die Gewindestangen mit einem pneumatischen Greifer auf. Auf der anderen Seite führt der zweite Roboter die Sensorplattform mit einem Laser-Profilsensor (siehe Bild 107d). Mit der minimalen Reichweite $r_{min} = 245,5$ mm und der maximalen Reichweite $r_{max} = 901,5$ mm des



(a) Gewindestange



(b) Gewindeprofil



(c) CAD-Modell



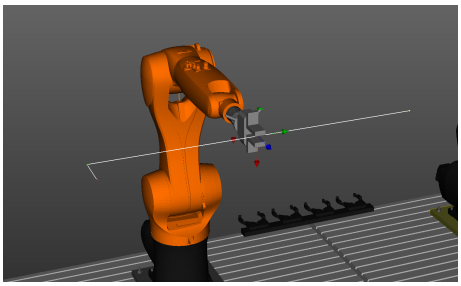
(d) Sensorplattform

Bild 107: Komponenten für die Inspektion von Gewindestangen

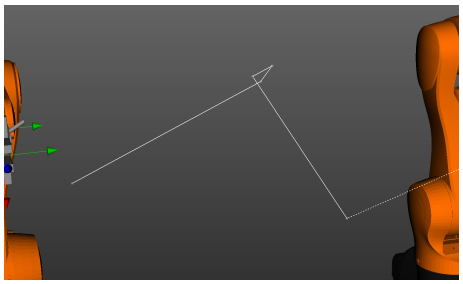
Robotersystems (siehe Abschnitt 6.1.1) resultiert gemäß Gleichung 7.2 eine maximale Bauteillänge für einen einzelnen Roboter von $l_{S_{max}} = 1734,9$ mm. Diese liegt unter der geforderten Bauteillänge. Für die Berechnung der maximalen Bauteillänge bei einer kooperativen Bearbeitung durch Gleichung 7.3 wird der zuvor bei der kinematischen Kalibrierung (siehe Abschnitt 7.1.4) bestimmte Roboterabstand $d_R = 1319,3$ mm verwendet. Die resultierende maximale Bauteillänge von $l_{C_{max}} = 2457,9$ mm zeigt, dass die geforderte Anwendung mit dem System möglich ist.

Damit der Sensor das Gewindeprofil mit einer hohen Genauigkeit erfassen kann, muss dieser senkrecht und äquidistant zur Gewindestange geführt werden. Der für die Anwendung gewählte Messabstand beträgt 200 mm und liegt damit ungefähr mittig innerhalb des Messbereichs des Sensors (siehe Abschnitt 6.1.1). Zudem muss die Gewindestange mittig gegriffen werden, um eine ausreichende Stabilität zu erzielen.

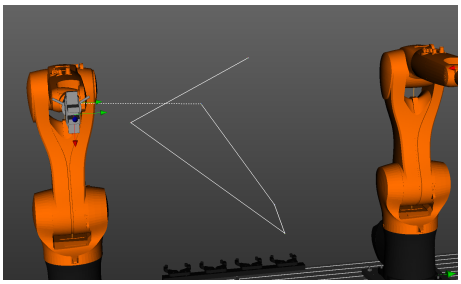
Gemäß dieser Vorgaben wird der werkstückbezogene Bahnverlauf in der entwickelten Offline-Programmiersoftware definiert (siehe Bild 108a). Anschließend erfolgt die Erstellung der Roboterprogramme für den aufgeteilten Prozess (siehe Bild 108b und Bild 108c) sowie das erfolgreiche Ausführen dieser mit dem realen Robotersystem.



(a) Werkstückbezogener Bahnverlauf



(b) Bahn des werkzeugführenden Roboters



(c) Bahn des werkstückführenden Roboters

Bild 108: Resultierende Bahnverläufe für die Inspektion von Gewindestangen

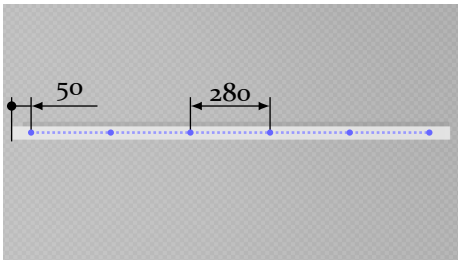
Markieren von Kunststoffleisten

Bei dieser exemplarischen Anwendung soll der Mensch bei der Verarbeitung von Kunststoffleisten (siehe Bild 109a) durch ein kollaborierendes Robotersystem unterstützt werden. Ziel ist es, dass der Roboter Markierungen im äquidistanten Abstand (siehe Bild 109b) auf der Oberfläche der Leisten aufbringt, damit der Mensch anschließend diese zielgerichtet weiterverarbeiten kann. Die Kunststoffleisten haben eine L-Profil mit einer Dicke von 2 mm, einem Querschnitt von 20 mm $\times$ 30 mm und einer Länge von $l = 1500$ mm (siehe Bild 109c). Das Muster der Markierungen besteht aus sechs äquidistant Punkten, die mittig auf der kurzen Seite des L-Profils platziert sind.

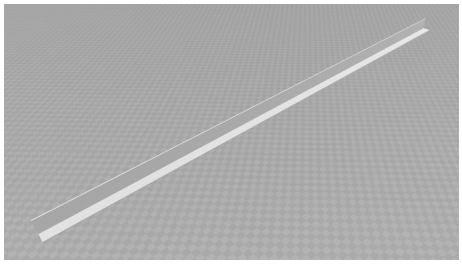
Für diese Anwendung wird das für MRK konzipierte zweite Robotersystem sowie dessen Standardgreifer verwendet. Einer der Greifer ist mit Greiferbacken versehen, die es ermöglichen einen handelsüblichen Stift zu fixieren (siehe Bild 109d). Auf der anderen Seite wird der Greifer mit Greiferbacken ausgestattet, die das Handhaben der Kunststoffleisten ermöglichen. Für das hier verwendete Robotersystem lässt sich mit der minimalen Reichweite $r_{\text{min}} = 124,5$ mm und der maximalen Reichweite $r_{\text{max}} = 559,0$ mm



(a) Kunststoffleiste



(b) Markierungen



(c) CAD-Modell



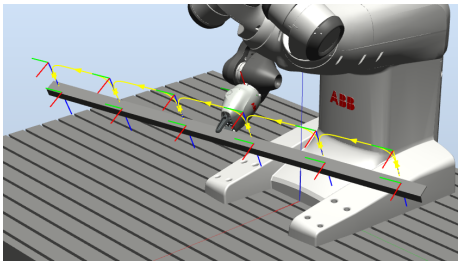
(d) Stifthalterung mit Stift

Bild 109: Komponenten für das Markieren von Kunststoffleisten

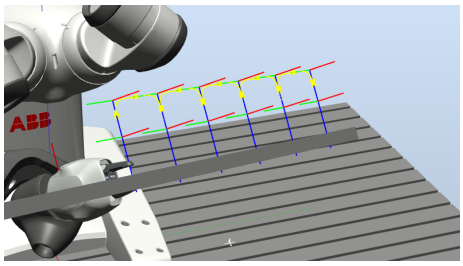
(siehe Abschnitt 6.2.1) anhand von Gleichung 7.2 eine maximale Bauteillänge für den Einsatz eines einzelnen Roboterarms abschätzen. Diese liegt mit $l_{S_{max}} = 1089,9\,\text{mm}$ unter der geforderten Bauteillänge. Die durch Gleichung 7.3 und einem Roboterabstand von $d_R = 220,0\,\text{mm}$ resultierende maximale Bauteillänge bei einer kooperativen Bearbeitung von $l_{C_{max}} = 2192,3\,\text{mm}$ zeigt, dass die geforderte Anwendung unter Verwendung beider Roboterarme möglich ist.

Als Randbedingung für den Prozess ist vorgegeben, dass der Stift senkrecht auf die Oberfläche der Kunststoffleiste aufgesetzt werden muss, um eine gleichmäßige Markierung zu erzeugen. Die Orientierung um die z-Achse des Stiftes ist aus Prozesssicht nicht relevant, wird aber dennoch beibehalten, um unnötige Bewegungen zu vermeiden. Die Gewindestange muss mittig gegriffen werden, um eine ausreichende Stabilität zu gewährleisten.

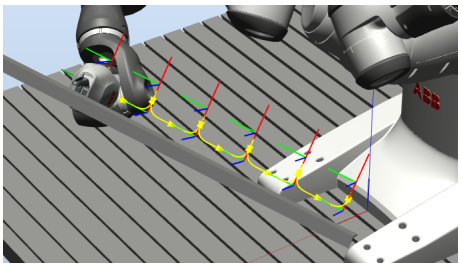
Für diese Anwendung wird gemäß der Randbedingungen mit der herstellerspezifischen Offline-Programmiersoftware der werkstückbezogene Bahnverlauf erstellt (siehe Bild 110a). Anschließend erfolgt mit dem in die Software integrierten Add-In die Ableitung der Programme für eine kooperative Bearbeitung (siehe Bild 110b und Bild 110c) sowie das Ausführen der Programme am realen Robotersystem.



(a) Werkstückbezogener Bahnverlauf



(b) Bahn des werkzeugführenden Roboters



(c) Bahn des werkstückführenden Roboters

Bild 110: Resultierende Bahnverläufe für das Markieren von Kunststoffleisten

7.3 Ableitung von Bewertungskriterien für den sinnvollen Einsatz der kooperativen Bearbeitung

Ob der Einsatz der kooperativen Bearbeitung bei einer Anwendung sinnvoll ist oder nicht, lässt sich anhand einer Reihe von evaluierbaren Bewertungskriterien beurteilen (siehe Bild 111). Zunächst wird unterschieden, ob es sich bei dem betrachteten Robotersystem bereits um ein MRS handelt oder nicht und wenn dies der Fall ist, ob Werkstück und Werkzeug bereits getrennt gehandhabt werden. Trifft beides zu, gilt es weitergehend zu hinterfragen, ob alle Roboter zum Prozess beitragen und wenn ja, ob diese gleichmäßig ausgelastet sind. Wenn einer dieser Punkte nicht zutrifft, ist die kooperative Bearbeitung gemäß des entwickelten Aufteilungsansatzes eine sinnvolle Erweiterung, um den Prozess gleichmäßig auf die Roboter zu verteilen und somit zu optimieren. Auf der anderen Seite gibt es Kriterien, die entscheiden, ob es sinnvoll ist ein Robotersystem um einen zusätzlichen Roboter zu erweitern und mit diesem eine kooperative Bearbeitung umzusetzen. Hierzu zählt das Erreichen der maximal möglichen Prozessgeschwindigkeit. Kann diese nicht durch eine einseitige Bewegung erreicht werden, ist es sinnvoll durch eine beidseitige Bewegung die relative Prozessgeschwindigkeit zu

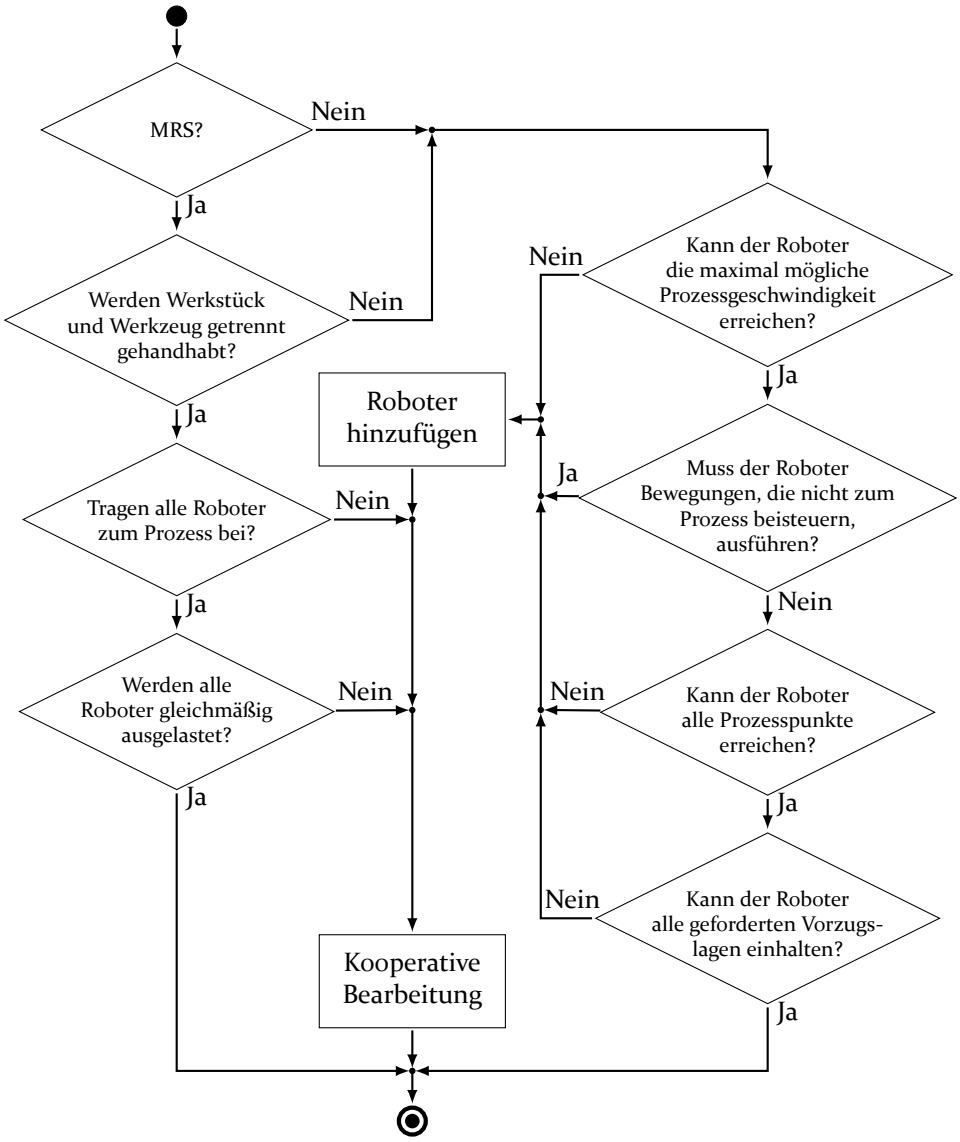


Bild 111: Bewertungskriterien für den sinnvollen Einsatz der kooperativen Bearbeitung

erhöhen und somit die Bearbeitungszeit zu reduzieren. Auch wenn nicht wertschöpfende Roboterbewegungen während des Prozesses ausgeführt werden, bietet die kooperative Bearbeitung eine Möglichkeit die Bearbeitungszeit zu reduzieren. Weitere Kriterien sind das Erreichen aller Prozesspunkte sowie das Einhalten der geforderten Orientierung. Kann das

betrachtete Robotersystem einzelne Prozesspunkte oder -posen nicht erreichen, erschließt sich dies eventuell durch die kooperative Bearbeitung mittels synchroner Bewegung von Werkzeug und Werkstück.

7.4 Zusammenfassung

Die in diesem Kapitel durchgeführte Evaluation zeigt die Grenzen der realisierten Robotersysteme hinsichtlich Bearbeitungszeitreduzierung, Erreichbarkeit und relativer Positioniergenauigkeit. Die reale Reduzierung der Bearbeitungszeit schwankt bei den Robotersystemen merklich und weicht teilweise von den theoretischen Werten ab, vor allem wenn ein Wechsel des Verfahrprofils vorliegt. Beim Großteil der mit den Robotersystemen realisierten Anwendungen liegt die erzielte Reduzierung der Bearbeitungszeit im Bereich von $25\,\%$. Wie anhand einer exemplarischen Anwendung mit einer Bearbeitungszeitreduzierung von etwa $60\,\%$ gezeigt, kann die Reduzierung aber auch größer ausfallen, wenn durch die Kooperation Bewegungen, die nicht zum Prozess beisteuern, vermieden werden. Die Erreichbarkeit der Robotersysteme ist durch die Kooperation annähernd verdoppelt. Es zeigt sich zudem, dass der Abstand der Roboter einen entscheidenden Einfluss auf die Erreichbarkeit hat. Mit zunehmendem Abstand sinkt die Erreichbarkeit. Eine zu nahe Positionierung der Roboter zueinander reduziert die Erreichbarkeit jedoch ebenfalls, da die Roboter sich gegenseitig in ihrer Bewegung einschränken. Zwei exemplarische Anwendungen zeigen, wie die erhöhte Erreichbarkeit das Bearbeiten bzw. Erfassen von größeren Werkstücken ermöglicht. Für eines der Robotersysteme erfolgt zudem eine kinematische Kalibrierung, bei der signifikante Abweichungen ermittelt werden. Mit Hilfe einer Kompensation können die Abweichungen deutlich reduziert werden, sodass eine Positioniergenauigkeit innerhalb eines Bereiches von ± 1 mm resultiert, was in den betrachteten Anwendungen zu einer ausreichenden Genauigkeit führt. Die auf Grundlage dieser Ergebnisse gebildeten Bewertungskriterien können zukünftig herangezogen werden, um Anwendungen bezüglich dem sinnvollen Einsatz der kooperativen Bearbeitung zu analysieren.

8 Zusammenfassung und Ausblick

Das wissenschaftliche Kernthema dieser Arbeit ist ein neuartiger Ansatz für die Kooperation von Industrierobotern – die kooperative Bearbeitung durch eine synchrone Bewegung von Werkstück und Werkzeug. Der entwickelte Ansatz bietet die Möglichkeit Prozesse auf mehrere Kinematiken aufzuteilen. Im Gegensatz zu NC-gesteuerten Bearbeitungsmaschinen wird durch den beidseitigen Einsatz von Industrierobotern eine freie räumliche Aufteilung möglich, was die Optimierung von komplexen Prozessbewegungen ermöglicht. Dies stellt den zentralen wissenschaftlichen Zugewinn dieser Forschungsarbeit dar. Da die manuelle Programmierung derartiger Anwendungen mit großem Aufwand verbunden ist, erfolgt eine automatisierte Programmierung auf Grundlage von Offline-Programmiersoftware, die eine schnelle und flexible Generierung der aufeinander abgestimmten Roboterprogramme ermöglicht. Aufgrund des zunehmenden Fokus der Industrie auf eine hohe Flexibilität bei Roboteranlagen, wird des Weiteren eine neuartige Werkstückerfassung unter Verwendung der kooperativen Bearbeitung entwickelt.

Der erarbeitete Ansatz für die Aufteilung von Prozessen ist herstellerunabhängig und somit mit einer Vielzahl von Industrierobotern umsetzbar. Im Rahmen dieser Arbeit wird er für einen Aufbau mit zwei Knickarmrobotern sowie einen Aufbau mit einem Zweiarmroboter implementiert und evaluiert. Dabei wird der Einfluss der Kooperation auf die Prozesszeit untersucht. Für Anwendungen, die komplett aus Prozessbewegungen bestehen, ist die theoretisch mögliche Halbierung der Prozesszeiten aufgrund der notwendigen Beschleunigungen in der Realität nicht erreichbar. Im Sinne einer möglichst hohen Stückzahl ist es in diesen Fällen attraktiver die zwei Industrieroboter parallel mit einer reinen Werkstück- oder Werkzeugbewegung einzusetzen. Einsparungspotenzial bietet das Konzept für Anwendungen mit Bewegungen, die nicht zum Prozess beisteuern. Durch Vermeidung dieser Bewegungen kann die Prozesszeit mehr als halbiert werden. Auch in Anlagen, bei denen Werkstück und Werkzeug aus anderen Gründen bereits getrennt von mehreren Industrierobotern gehandhabt werden, kann die Prozesszeit verbessert werden, indem die Roboter beidseitig einen Beitrag zum Prozess leisten. Voraussetzung ist jedoch stets, dass eine Steigerung der Prozessgeschwindigkeit prozesstechnisch möglich ist. Umgekehrt kann aber auch einen Steigerung der Prozessgeschwindigkeit durch das gleichzeitige Bewegen von Werkstück und Werkzeug durch den Ansatz erzielt werden. Des Weiteren erfolgt eine Untersuchung des Einflusses der Kooperation

auf die werkstückbezogene Erreichbarkeit. Es zeigt sich, dass die Erreichbarkeit innerhalb des gemeinsamen Arbeitsraumes im Vergleich zur reinen Werkstück- oder Werkzeugbewegung annähernd verdoppelt ist. Die kooperative Bearbeitung ermöglicht also eine deutlich gesteigerte Erreichbarkeit, wodurch neue Anwendungen erschlossen werden können. Ein weiteres Kriterium, das untersucht wird, ist die Positioniergenauigkeit der Industrieroboter zueinander. Eine messtechnische Überprüfung eines Robotersystems mit zwei Knickarmrobotern zeigt, dass die Abweichungen vor allem in Richtung der beiden Industrieroboter zueinander markant sind. Mit Hilfe einer Kompensation können die Abweichungen deutlich reduziert werden, sodass für die betrachteten Anwendungen eine ausreichende Genauigkeit erzielt wird.

Der Aufteilungsansatz und die zugehörigen Untersuchungsmethoden stellen einen Erkenntnisgewinn in der Wissenschaft dar, bieten aber dennoch Potenzial für weitere Verbesserungen. Vor allem hinsichtlich der Optimierung der Aufteilung sind alternative Lösungswege denkbar. Die in dieser Arbeit präsentierte Aufteilung der Prozessbahnen ist hinsichtlich der Prozesszeit optimiert, gegenwärtig gerät jedoch auch die Reduzierung des Energieverbrauches zunehmend in den Fokus der Industrie [85]. Daher ist eine Erweiterung des Konzeptes um eine Optimierung des Energieverbrauches sinnvoll. Ein möglicher Ansatz hierfür ist die Anpassung der Roboterbewegungen durch energetisch günstigere Zwischenpunkte [136]. Ebenso ist eine inverse Berechnung der optimalen Startpose, beispielsweise durch das Gradientenverfahren oder das Simplex-Verfahren, denkbar.

Auch hinsichtlich der Auslegung von MRS für eine kooperative Bearbeitung sind ergänzende Ansätze erstrebenswert. Wie die Untersuchungen am ersten Robotersystem zeigen, ist die Positionierung der beiden Industrieroboter zueinander ein starker Einflussfaktor auf die Erreichbarkeit. Eine zielgerichtete Positionierung der Industrieroboter zueinander bietet Potenzial für eine zielgerichtete Steigerung der Erreichbarkeit sowie eine Vergrößerung des gemeinsamen Arbeitsraumes. So ist eine Bestimmung der optimalen Roboterpositionierung, beispielsweise auf Grundlage der Reaktionsflächenmethode [52] oder von Optimierungsalgorithmen [69], eine weitere sinnvolle Ergänzung.

Für die Umsetzung von Roboteranwendungen mittels kooperativer Bearbeitung ist eine ausreichende Prozessgenauigkeit essentiell. Die durchgeführte Untersuchung der Positioniergenauigkeit zeigt deutlich die Grenzen der verwendeten Systeme. Wenn Anwendungen mit einer erhöhten Anforderung

an die Prozessgenauigkeit realisiert werden sollen, bietet eine Erhöhung der Genauigkeit durch iteratives Lernen [105] Abhilfe. Hierbei wird die Position beim Verfahren im Arbeitsraum kontinuierlich durch ein externes Messsystem erfasst und aus der Vielzahl an Abweichungen wird iterativ das für die Steuerung verwendete Robotermodell angepasst. Auch das dynamische Verhalten bei der kooperativen Bearbeitung kann einen Einfluss auf die Verwendbarkeit der erzeugten Anwendung haben, weshalb eine weitergehende Untersuchung dieser Zusammenhänge sinnvoll ist. Ein möglicher Ansatz zur Sicherstellung von geeigneten Prozessen ist die sensorgestützte Korrektur der Bewegung der Industrieroboter während des Prozesses [90]. Eine geeignete Erweiterung zur Vermeidung von ruckartigen Werkstückbewegungen bei Rotationen um Bahnpunkte kann mittels automatisierter Bahnoptimierung [23] erfolgen.

9 Summary and outlook

The scientific key issue of this work is a new approach for the cooperation of industrial robots—the cooperative processing by a synchronous movement of workpiece and tool. The developed approach offers the possibility of dividing processes into several kinematics. In contrast to NC-controlled processing machines, the use of industrial robots on both sides allows a free spatial division, which enables the optimization of complex process movements. This represents the central scientific gain of this research work. Since the manual programming of such applications involves a great effort, automated programming is performed with offline programming software, which enables fast and flexible generation of the coordinated robot programs. Furthermore, due to the industry's increasing focus on high flexibility in robotic systems, a new type of workpiece scanning using the cooperative processing is developed.

The developed approach for the division of processes is vendor-independent and thus feasible with a variety of industrial robots. Within the scope of this thesis, the approach is implemented and evaluated for a setup with two articulated robot arms as well as for a setup with a dual-arm robot. In the course of this, the influence of the cooperation on the process time is examined. For applications that consist entirely of process movements, the theoretically possible halving of process times cannot be achieved in reality due to the necessary accelerations. In terms of the highest possible number of pieces, it is more attractive to use the two industrial robots in parallel with a pure workpiece or tool movement. The concept offers savings potential for applications with movements that do not contribute to the process. By avoiding these movements, the process time can be more than halved. Likewise, in systems where the workpiece and tool are already handled separately by several industrial robots for other reasons, the process time can be improved by the robots making a contribution to the process on both sides. However, the prerequisite is always that an increase in the process speed is technically possible. Conversely, an increase in process speed can also be achieved by simultaneously moving the workpiece and the tool using the approach. Furthermore, the influence of the cooperation on the workpiece-related reachability is examined. It turns out that the reachability within the common working space is almost doubled in comparison to the pure workpiece or tool movement. Thus, the cooperative processing allows a significantly increased reachability, whereby new applications can be acquired. Another criterion that is investigated is the positioning accuracy of

the industrial robots relative to each other. A metrological investigation of the first robot system shows that the deviations are particularly striking in the direction of the two industrial robots towards each other. With the aid of a compensation, the deviations can be significantly reduced, so that a sufficient accuracy is achieved for the considered applications.

The division approach and the associated investigation methods represent a gain in knowledge in science, but still offer potential for further improvements. Alternative solutions are conceivable, especially regarding the optimization of the distribution. The division of the process paths presented in this thesis is optimized with regard to the process time, but currently the reduction of energy consumption is increasingly coming into focus of the industry [85]. Therefore, an extension of the approach to optimize the energy consumption makes sense. A possible approach for this is the adaptation of the robot movements by energetically favorable intermediate points [136]. An inverse calculation of the optimal start pose, for example, by gradient descent or simplex method, is also conceivable.

Also, regarding the interpretation of an MRS for cooperative processing, complementary approaches are desirable. As the investigations on the first robot system show, the positioning of the two industrial robots in relation to each other is a strong influence factor on the reachability. A targeted positioning of the industrial robots to each other offers potential for a purposeful increase in reachability and an expansion of the shared workspace. So a determination of the optimal robot positioning, for example based on the reaction surface method [52] or optimization algorithms [69], is another useful extension.

For the implementation of robotic applications by means of cooperative processing, sufficient process accuracy is essential. The investigation of the positioning accuracy clearly shows the limits of the systems used. If applications with increased process accuracy requirements need to be implemented, increasing the accuracy through iterative learning [105] is a possible aid. Here, the position during the procedure in the working area is continuously recorded by an external measuring system and from the large number of deviations, the robot model used for the control is iteratively adapted. The dynamic behavior during cooperative processing can also have an influence on the usability of the generated application, which is why further investigation of these relationships is useful. One possible approach for ensuring suitable processes is the sensor-guided adjustment of the movement of industrial robots during the process [90]. A suitable extension for avoiding

jerky movements of workpieces during rotations around waypoints can be done by automated path optimization [23].

Literaturverzeichnis

[1] ABB Automation GmbH. Website. RobotStudio. 2018. URL: https://new.abb.com/products/robotics/de/robotstudio (aufgerufen am 11. 11. 2018).

[2] ABB Automation GmbH. Website. YUMI (IRB 14000). 2018. URL: https://new.abb.com/products/robotics/de/industrieroboter/yumi (aufgerufen am 22. 05. 2018).

[3] Abdelhafiz, A., B. Riedel und W. Niemeier. Towards a 3D true colored space by the fusion of laser scanner point cloud and digital photos. In: *Proceedings of the ISPRS Working Group V/4 Workshop (3D-ARCH)*. 2005.

[4] Alicia, G. und B. Shirinzadeh. A systematic technique to estimate positioning errors for robot accuracy improvement using laser interferometry based sensing. In: *Mechanism and Machine Theory*. Bd. 40. 8. Elsevier, 2005, S. 879–906.

[5] Angeles, J. Fundamentals of Robotic Mechanical Systems. Cham: Springer-Verlag, 2014. ISBN: 978-3-319-01850-8.

[6] Angerer, A., A. Hoffmann, L. Larsen, M. Vistein, J. Kim, M. Kupke und W. Reif. Planning and Execution of Collision-free Multi-robot Trajectories in Industrial Applications. In: *Proceedings of the 47st International Symposium on Robotics (ISR 2016)*. VDE-Verlag, 2016.

[7] Aoyagi, S., A. Kohama, Y. Nakata, Y. Hayano und M. Suzuki. Improvement of robot accuracy by calibrating kinematic model using a laser tracking system-compensation of non-geometric errors using neural networks and selection of optimal measuring points using genetic algorithm. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)*. IEEE, 2010, S. 5660–5665.

[8] API Product Specifications. Radian. Automated Precision Inc. 2012.

[9] AROTEC Automation und Robotik GmbH. Website. Offene Planungs- und Prozesssoftware VMP. 2018. URL: https://www.arotec-robotik.de/webpage/produkte/vmp.php (aufgerufen am 14. 11. 2018).

[10] ArtiMinds Robotics GmbH. Website. ArtiMinds RPS. 2018. URL: https://www.artiminds.com/?page_id=2412&lang=de (aufgerufen am 05. 05. 2018).

[11] Basile, F., F. Caccavale, P. Chiacchio, J. Coppola und C. Curatella. Task-oriented motion planning for multi-arm robotic systems. In: *Robotics and Computer-Integrated Manufacturing* 28.5 (2012), S. 569–582. DOI: 10.1016/j.rcim.2012.02.007.

[12] Biggs, G. und B. MacDonald. A Survey of Robot Programming Systems. In: *Proceedings of the Australasian Conference on Robotics and Automation (CSIRO 2003)*. 2003.

[13] Bischoff, R., A. Kazi und M. Seyfarth. The MORPHA style guide for icon-based programming. In: *Proceedings of the 11th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN 2002)*. 2002, S. 482–487.

[14] Blackbird Robotersysteme GmbH. Poduktblatt. RobotMotionCenter. 2014. URL: http : / / www . blackbird - robotics . de / fileadmin / user _ upload/pdf/Blackbird_Produktblatt_RMC_2_180614.pdf (aufgerufen am 14. 11. 2018).

[15] Boctor, E. M., G. Fischer, M. A. Choti, G. Fichtinger und R. H. Taylor. A dual-armed robotic system for intraoperative ultrasound guided hepatic ablative therapy: a prospective study. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2004)*. IEEE, 2004, S. 2517–2522.

[16] Borangiu, T., A. Dogar und A. Dumitrache. Integrating a Short Range Laser Probe with a 6-DOF Vertical Robot Arm and a Rotary Table. In: *Proceedings of the 17th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD 2008)*. Ancona, Italy, 2008.

[17] Borangiu, T., A. Dogar und A. Dumitrache. Calibration of Wrist-Mounted Profile Laser Scanning Probe using a Tool Transformation Approach. In: *Proceedings of the 18th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD 2009)*. Brasov, Romania, 2009.

[18] Bredenfeld, A. und G. Indiveri. Robot behavior engineering using DD-Designer. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2001)*. IEEE, 2001, S. 205–210.

[19] Brüninghaus, J., A. Hypki und A. Schyja. Bahnplanung für kooperierende Robotersysteme zur Umsetzung der roboterbasierten inkrementellen Blechumformung von freigeformten Bauteilen. In: *Robotik 2008: Leistungsstand, Anwendungen, Visionen, Trends*. VDI-Verlag, 2008.

[20] Buchmann, T., J. Baumgartl, D. Henrich und B. Westfechtel. Towards A Domain-specific Language For Pick-And-Place Applications. In: *Proceedings of the 4th International Workshop on Domain-Specific Languages and Models for Robotic Systems (DSLRob 2013)*. 2013.

[21] Burgard, W., M. Moors, D. Fox, R. Simmons und S. Thrun. Collaborative multi-robot exploration. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2000)*. IEEE, 2000, S. 476–481.

[22] Buschhaus, A., H. Grünsteudel und J. Franke. Geometry-based 6D-pose visual servoing system enabling accuracy improvements of industrial robots. In: *Proceedings of the International Conference on Advanced Mechatronic Systems (ICAMechS 2016)*. IEEE, 2016, S. 195–200.

[23] Buschhaus, A., P. Reitinger und J. Franke. Automated optimization of complex three-dimensional robot trajectories. In: *Proceedings of the 2nd International Conference on Control, Automation and Robotics (ICCAR 2016)*. 2016, S. 3–10.

[24] Chaumette, F., P. Rives und B. Espiau. Positioning of a robot with respect to an object, tracking it and estimating its velocity by visual servoing. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 1991)*. IEEE, 1991, S. 2248–2253.

[25] Chen, C.-P., P.-J. Wang, H.-P. W. W.-H. Wang und C.-C. Chen. Developing industrial dual arm robot for flexible assembly through reachability map. In: *Technical Report* (2015).

[26] Cobzas, D., H. Zhang und M. Jagersand. A comparative analysis of geometric and image-based volumetric and intensity data registration algorithms. In: *Proceedings of the International Conference on Robotics and Automation*. Bd. 3. Washington, DC, USA: IEEE, 2002.

[27] Corke, P. Robotics, Vision and Control. Berlin: Springer-Verlag, 2011.

[28] Dai, W. und M. Kampker. User oriented integration of sensor operations in a offline programming system for welding robots. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2000)*. IEEE, 2000, S. 1563–1567.

[29] Dai, X., G. Hager und J. Peterson. Specifying behavior in C++. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2002)*. 2002, S. 153–160.

[30] Dani, J., A. Winkler und J. Suchy. Robot Programming for Surface Finishing based on CAD Model Including External Axes. In: *Proceedings of the 7th German Conference on Robotics (ROBOTIK 2012)*. VDE-Verlag, 2012.

[31] Day, C. P. Robot Accuracy Issues and Methods of Improvement. In: *Progress in Robotics and Intelligent Systems*. Bd. 2. Intellect Books, 1996, S. 90–108.

[32] DIN EN ISO 10218-1: Industrieroboter - Sicherheitanforderungen - Teil 1: Roboter. Norm. 2011.

[33] Doulgeri, Z. und T. Matiakis. A web telerobotic system to teach industrial robot path planning and control. In: *IEEE Transactions on Education*. Bd. 49. 2. IEEE, 2006, S. 263–270. DOI: 10.1109/TE.2006.873975.

[34] Driels, M. R., W. Swayze und S. Potter. Full-pose calibration of a robot manipulator using a coordinate-measuring machine. In: *The International Journal of Advanced Manufacturing Technology* 8.1 (1993), S. 34–41. ISSN: 1433-3015. DOI: 10.1007/BF01756635.

[35] EASY-ROB. Website. EASY-ROB 3D Robot Simulation Tool. 2017. URL: http : / / www . easy - rob . com / easy - rob (aufgerufen am 24. 05. 2017).

[36] Elatta, A., L. Gen, F. Zhi, Y. Daoyuan und L. Fei. An Overview of Robot Calibration. In: *Information Technology Journal*. Bd. 3. 1. Asian Network for Scientific Information, 2004, S. 74–78.

[37] FANUC Deutschland GmbH. Website. Simulation Software ROBOGUIDE. 2018. URL: https://www.fanuc.eu/de/en/robots/accessories/roboguide (aufgerufen am 11. 11. 2018).

[38] G. Li, L. D., L. Pan und F. Henghai. The Calibration Algorithm of a 3D Color Measurement System based on the Line Feature. In: *International Journal of Image, Graphics and Signal Processing* (2009).

[39] Gan, Y. und X. Dai. Base frame calibration for coordinated industrial robots. In: *Robotics and Autonomous Systems* 59 (2011), S. 563–570.

[40] Globaco GmbH. Website. Dosiergeräte und Ventilsteuergeräte. 2018. URL: https://www.globaco.de/produkte/dosieren/dosiergeraete-volumetrisch (aufgerufen am 16. 09. 2018).

[41] Gonzalez-Galvan, E. J., A. Loredo-Flores, E. D. Laborico-Aviles, F. Pazos-Flores und J. J. Cervantes-Sanchez. An algorithm for optimal closed-path generation over arbitrary surfaces using uncalibrated vision. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2007)*. 2007, S. 2465–2470. DOI: 10.1109/ROBOT.2007.363689.

[42] Gräßler, I. Kundenindividuelle Massenproduktion - Entwicklung, Vorbereitung der Herstellung, Veränderungsmanagement. Berlin: Springer-Verlag, 2004. ISBN: 978-3-540-20554-8.

[43] Grote, K.-H. und J. Feldhusen. Dubbel - Taschenbuch für den Maschinenbau. Berlin: Springer-Verlag, 2007. ISBN: 978-3-540-49714-1.

[44] Hallenberg, J. Robot Tool Center Point Calibration using Computer Vision. Linköping, Sweden: Linköpings University, 2007.

[45] Haun, M. Handbuch Robotik - Programmieren und Einsatz intelligenter Roboter. Berlin: Springer-Verlag, 2013. ISBN: 978-3-642-39857-5.

[46] Hesse, S. Grundlagen der Handhabungstechnik. München: Carl Hanser Verlag, 2016. ISBN: 978-3-446-44432-4.

[47] Hesse, S. und V. Malisa. Taschenbuch Robotik - Montage - Handhabung. München: Carl Hanser Verlag, 2010. ISBN: 978-3-446-41969-8.

[48] Höpler, R. und M. Otter. A Versatile C++ Toolbox for Model Based, Real Time Control Systems of Robotic Manipulators. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001)*. IEEE, 2001, S. 2208–2214.

[49] Hudak, P., A. Courtney, H. Nilsson und J. Peterson. Arrows, Robots, and Functional Reactive Programming. In: *Summer School on Advanced Functional Programming 2002, Oxford University*. Bd. 2638. Lecture Notes in Computer Science. Springer-Verlag, 2003, S. 159–187.

[50] Jamone, L., L. Natale, G. Sandini und A. Takanishi. Interactive online learning of the kinematic workspace of a humanoid robot. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*. IEEE, 2012, S. 2606–2612.

[51] Kalibrierspitzen. einfach aber effektiv. YASKAWA Europe GmbH. 2017.

[52] Kamrani, B., V. Berbyuk, D. Wäppling, U. Stickelmann und X. Feng. Optimal robot placement using response surface method. In: *The International Journal of Advanced Manufacturing Technology*. Bd. 44. Berlin: Springer-Verlag, 2009, S. 201–210.

[53] Karim, A. und A. Verl. Challenges and obstacles in robot-machining. In: *Proceedings of the 44th International Symposium on Robotics (ISR 2013)*. IEEE, 2014.

[54] Kim, J. Y. CAD-Based Automated Robot Programming in Adhesive Spray Systems for Shoe Outsoles and Uppers. In: *Journal of Robotic Systems*. Wiley Periodicals, 2004, S. 625–634.

[55] Kitagishi, I., T. Machino, A. Nakayama, S. Iwaki und M. Okudaira. Development of motion data description language for robots based on eXtensible Markup Language - realization of better understanding and communication via networks. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2002)*. IEEE, 2002, S. 1145–1151.

[56] Klimentjew, D., N. Hendrich und J. Zhang. Multi sensor fusion of camera and 3D laser range finder for object recognition. In: *Proceedings of the Conference on Multisensor Fusion and Integration for Intelligent Systems*. Salt Lake City, UT, USA: IEEE, 2010.

[57] Knepper, R. A., T. Layton, J. Romanishin und D. Rus. IkeaBot: An Autonomous Multi-Robot Coordinated Furniture Assembly System. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2013)*. IEEE, 2013, S. 855–862.

[58] KR AGILUS sixx. Spez KR AGILUS sixx V13. KUKA Deutschland GmbH. 2018.

[59] Krüger, J., G. Schreck und D. Surdilovic. Dual arm robot for flexible and cooperative assembly. In: *CIRP Annals - Manufacturing Technology* 60 (2011), S. 5–8. DOI: 10.1016/j.cirp.2011.03.017.

[60] Kuhlenkötter, B., J. Brüninghaus und A. Hypki. Roboterbasierte inkrementelle Blechumformung: Bahnplanung für die Gegenhalterbewegung des kooperierenden Robotersystems. In: *Automation 2010: der 11. Branchentreff der Mess- und Automatisierungstechnik*. VDI-Verlag, 2010, S. 305–308.

[61] KUKA Roboter GmbH. Website. KR AGILUS. 2015. URL: https://www.kuka.com/de-de/produkte-leistungen/robotersysteme/industrieroboter/kr-agilus (aufgerufen am 17. 09. 2015).

[62] KUKA Roboter GmbH. Website. Software KUKA.Sim. 2018. URL: https://www.kuka.com/de-de/produkte-leistungen/robotersysteme/software/planung-projektierung-service-sicherheit/kuka_sim (aufgerufen am 25. 11. 2018).

[63] KUKA.RoboTeam 1.0. KST RoboTeam 1.0 V2 de. KUKA Roboter GmbH. 2012.

[64] Lee, J. und P. H. Chang. Redundancy Resolution for Dual-Arm Robots Inspired by Human Asymmetric Bimanual Action: Formulation and Experiments. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2015)*. IEEE, 2015, S. 6058–6065.

[65] Leutert, F., D. Borrmann, K. Schilling und A. Nüchter. Spatial projection of thermal data for visual inspection. In: *Proceedings of the 14th International Conference on Control, Automation, Robotics and Vision (ICARCV 2016)*. Phuket, Thailand, 2016.

[66] Li, J., M. Chen, X. Jin, C. Xuebi, D. Yu, Z.Dai, Z. Ou und Q. Tang. Calibration of a multiple axes 3-D laser scanning system consisting of robot, portable laser scanner and turntable. In: *Optik - International Journal for Light and Electron Optics*. Bd. 122. Amsterdam, Netherlands: Elsevier B.V., 2011.

[67] Loffler, M., D. Dawson, E. Zergeroglu und N. Costescu. Object-oriented techniques in robot manipulator control software development. In: *Proceedings of the American Control Conferenc (ACC 2001)*. IEEE, 2001, S. 4520–4525.

[68] Logitech. Website. C920 HD Pro Webcam. 2015. URL: https://www.logitech.com/de-de/product/hd-pro-webcam-c920 (aufgerufen am 17. 09. 2015).

[69] Lopes, A. M. und E. S. Pires. Optimization of the Workpiece Location in a Machining Robotic Cell. In: *International Journal of Advanced Robotic Systems*. Bd. 8. Vukovar, Croatia: InTech, 2011, S. 37–46.

[70] Macho, M., L. Nägele, A. Hoffmann, A. Angerer und W. Reif. A Flexible Architecture for Automatically Generating Robot Applications based on Expert Knowledge. In: *Proceedings of the 47st International Symposium on Robotics (ISR 2016)*. VDE-Verlag, 2016.

[71] Maddern, W., A. Harrison und P. Newman. Lost in Translation (and Rotation): Rapid Extrinsic Calibration for 2D and 3D LIDARs. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2012)*. IEEE, 2012, S. 3096–3102.

[72] Marton, Z. C., R. B. Rusu und M. Beetz. On Fast Surface Reconstruction Methods for Large and Noisy Datasets. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2009)*. Kobe, Japan, 2009.

[73] Mechsner, F., D. Kerzel, G. Knoblich und W. Prinz. Perceptual basis of bimanual coordination. In: *Nature* 414 (2001), S. 69–73.

[74] Mei, C. und P. Rives. Calibration between a central catadioptric camera and a laser range finder for robotic applications. In: *Proceedings of the International Conference on Robotics and Automation*. Orlando, FL, USA: IEEE, 2006.

[75] MICRO-EPSILON MESSTECHNIK GmbH & Co. KG. Website. Kompakte 2D/3D Laser-Profil-Sensoren. 2015. URL: https://www.micro-epsilon.de/sensors/laser-scanner (aufgerufen am 17. 09. 2015).

[76] Mohottala, S., S. Ono, M.Kagesawa und K. Ikeuchi. Fusion of a camera and a laser range sensor for vehicle recognition. In: *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. Miami, FL, USA: IEEE, 2009.

[77] Müller, R., M. Esser und M. Janssen. Integrative Path Planning and Motion Control for Handling Large Components. In: *Proceedings of the 4th International Conference on Intelligent Robotics and Applications (ICIRA 2011)*. Berlin: Springer-Verlag, 2011, S. 93–101.

[78] Myers, D., M. Pritchard und M. Brown. Automated programming of an industrial robot through teach-by showing. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2001)*. IEEE, 2001, S. 4078–4083.

[79] Navarrete, J., D. Viejo und M. Cazorla. Portable 3D laser-camera calibration system with color fusion for SLAM. In: *International Journal of Automation and Smart Technology*. Bd. 3. Taipei, Taiwan: AUSMT, 2013.

[80] Nguyena, H.-N., J. Zhoua und H.-J. Kang. A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network. In: *Neurocomputing*. Bd. 151. 3. Elsevier, 2015, S. 996–1005.

[81] Nunez, P., P. Drews, R. Rocha und J. Dias. Data fusion calibration for a 3d laser range finder and a camera using inertial data. In: *Proceedings of the 4th European Conference on Mobile Robots*. Barcelona, Spain: IEEE, 2009.

[82] Ogawara, K., J. Takamatsu, H. Kimura und K. Ikeuchi. Generation of a task model by integrating multiple observations of human demonstrations. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2002)*. IEEE, 2002, S. 1545–1550.

[83] Orendt, E. M., M. Fichtner und D. Henrich. Robot Programming by Non-Experts: Intuitiveness and Robustness of One-Shot Robot Programming. In: *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN 2016)*. IEEE, 2016.

[84] Osgood, T. J. und Y. Huang. Calibration of laser scanner and camera fusion system for intelligent vehicles using Nelder-Mead optimization. In: *Measurement Science and Technology*. Bd. 24. IOP Publishing Ltd, 2013.

[85] Paryanto, M. Brossog, M. Bornschlegl und J. Franke. Reducing the energy consumption of industrial robots in manufacturing systems. In: *The International Journal of Advanced Manufacturing Technology* 78 (2015), S. 1315–1328. DOI: 10.1007/s00170-014-6737-z.

[86] Perovic, B. Spanende Werkzeugmaschinen. Berlin: Springer-Verlag, 2009. ISBN: 978-3-540-89951-8.

[87] Peterson, J., G. Hager und P. Hudak. A Language for Declarative Robotic Programming. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 1999)*. IEEE, 1999.

[88] Pichler, A., M. Vincze, H. Andersen, O. Madsen und K. Häusler. A Method for Automatic Spray Painting of Unknown Parts. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2002)*. IEEE, 2002.

[89] Porges, O., R. Lampariello, J. Artigas, A. Wedler, C. Borst und M. A. Roa. Reachability and Dexterity: Analysis and Applications for Space Robotics. In: *Proceedings of the Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA)*. Noordwijk, Netherlands, 2015.

[90] Posada, J. R. D., S. Kumar, A. Kuss, U. Schneider, M. Drust, T. Dietz und A. Verl. Automatic Programming and Control for Robotic Deburring. In: *Proceedings of the 47st International Symposium on Robotics (ISR 2016)*. VDE-Verlag, 2016, S. 688–695.

[91] RoboDK. Website. Offline Programming. 2015. URL: https://robodk.com/offline-programming (aufgerufen am 23. 05. 2017).

[92] Robotics (IFR) Statistical Department, I. F. of. World Robotics 2018 Industrial Robots. Frankfrut: VDMA-Verlag, 2018.

[93] Rocha, R., J. Dias und A. Carvalho. Cooperative Multi-Robot Systems: A study of Vision-based 3-D Mapping using Information Theory. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2005)*. IEEE, 2005, S. 384–389.

[94] Rückel, V. Rechnergestützte Ablaufplanung und Bahngenerierung für kooperierende Industrieroboter. Bamberg: Meisenbach, 2006. ISBN: 978-3-87525-250-7.

[95] Rudloff, D. Ein Beitrag zur Automatisierung der Programmierung von Industrierobotern. Herzogenrath: Shaker Verlag, 1995. ISBN: 978-3-82650-700-7.

[96] Saff, E. B. und A. B. J. Kuijlaars. Distributing Many Points on a Sphere. In: *Mathematical Intelligencer* 19.1 (1997), S. 5–11.

[97] Sauer, B. Konstruktionselemente des Maschinenbaus 1: Grundlagen der Berechnung und Gestaltung von Maschinenelementen. Berlin: Springer-Verlag, 2016. ISBN: 978-3-642-39500-0.

[98] Scaramuzza, D., A. Harati und R. Siegwart. Extrinsic self calibration of a camera and a 3D laser range finder from natural scenes. In: *Proceedings of the International Conference on Intelligent Robots and Systems*. San Diego, CA, USA: IEEE, 2007.

[99] SCHUNK GmbH & Co. KG. Website. SWA-005-000-000. 2015. URL: https : / / schunk . com / de _ en / gripping - systems / product / 17975 - 0302308-swa-005-000-000 (aufgerufen am 17. 09. 2015).

[100] Shen, C. und S. Zhu. A Robotic System for Surface Measurement Via 3D Laser Scanner. In: *Proceedings of the 2nd International Conference on Computer Application and System Modeling*. Bd. 21. Paris, France: Atlantis Press, 2012.

[101] Siciliano, B. und O. Khatib. Springer Handbook of Robotics. Berlin: Springer-Verlag, 2008. ISBN: 978-3-540-23957-4.

[102] Smith, C., Y. Karayiannidis, L. Nalpantidis, X. Gratal, P. Qi, D. V. Dimarogonas und D. Kragic. Dual arm manipulation-A survey. In: *Robotics and Autonomous Systems*. Bd. 60. Elsevier, 2012, S. 1340–1353.

[103] Smits, R., T. D. Laet, K. Claes, H. Bruyninckx und J. Schutter. iTASC: a tool for multi-sensor integration in robot manipulation. In: *Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2008)*. IEEE, 2008, S. 426–433.

[104] Stroupe, A., M. Martin und T. Balch. Distributed sensor fusion for object position estimation by multi-robot systems. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2001)*. IEEE, 2001, S. 1092–1098.

[105] Stückelmaier, P., M. Grotjahn und C. Fräger. Iterative Improvement of Path Accuracy of Industrial Robots Using External Measurements. In: *Proceedings of the IEEE International Conference on Advanced Intelligent Mechatronics (AIM 2017)*. IEEE, 2017.

[106] Súarez-Ruiz, F. und Q.-C. Pham. A Framework for Fine Robotic Assembly. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2016)*. IEEE, 2016, S. 421–426.

[107] Sundaram, M. A. R. A. M. und O. Porges. Planning realistic interactions for bimanual grasping and manipulation. In: *Proceedings of the IEEE/RAS 16th International Conference on Humanoid Robots (Humanoids 2016)*. IEEE, 2016, S. 987–994.

[108] Sung, C., N. Ayanian und D. Rus. Improving the Performance of Multi-Robot Systems by Task Switching. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2013)*. IEEE, 2013, S. 2999–3006.

[109] Takamatsu, J., K. Ogawara, H. Kimura und K. Ikeuchi. Correcting observation errors for assembly task recognition. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2002)*. IEEE, 2002, S. 232–237.

[110] Takarics, B., P. T. Szemes, G. Nemeth und P. Korondi. Welding trajectory reconstruction based on the Intelligent Space concept. In: *Proceedings of the Conference on Human System Interactions (HSI 2008)*. 2008, S. 791–796. DOI: 10.1109/HSI.2008.4581542.

[111] Tan, T. N., G. D. Sullivan und K. D. Baker. Recovery of Intrinsic and Extrinsic Camera Parameters Using Perspective Views of Rectangles. In: *Proceedings of the 1995 British Conference on Machine Vision (Vol. 1)*. BMVC '95. Surrey, UK: BMVA Press, 1995, S. 177–186.

[112] Thomas, C., F. Busch, B. Kuhlenkötter und J. Deusei. Ensuring Human Safety with Offline Simulation and Real-time Workspace Surveillance to Develop a Hybrid Robot Assistance System for Welding of Assemblies. In: *Proceedings of the 4th International Conference on Changeable, Agile, Reconfigurable and Virtual Production(CARV 2011)*. Springer-Verlag, 2011, S. 465–470.

[113] Thrun, S. Towards programming tools for robots that integrate probabilistic computation and learning. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2000)*. IEEE, 2000, S. 306–312.

[114] Tsarouchi, P., S. Makris, G. Michalos, M. Stefos, K. Fourtakas, K. Kaltsoukalas, D. Kontrovrakis und G. Chryssolouris. Robotized Assembly Process Using Dual Arm Robot. In: *Procedia CIRP* 23 (2014), S. 47–52. DOI: 10.1016/j.procir.2014.10.078.

[115] Vincze, M., A. Pichler, G. Biegelbauer, K. Häusler, H. Andersen, O. Madsen und M. Kristiansen. Automatic Robotic Spray Painting of Low Volume High Variant Parts. In: *Proceedings of the 33th International Symposium on Robotics (ISR 2002)*. 2002.

[116] Vogl, W. Eine interaktive räumliche Benutzerschnittstelle für die Programmierung von Industrieroboter. München: Herbert Utz Verlag, 2009. ISBN: 978-3-8316-0869-0.

[117] Wang, W., F. Liu und C. Yun. Calibration method of robot base frame using unit quaternion form. In: *Precision Engineering* 41 (2015), S. 47–54.

[118] Waringo, M. und D. Henrich. 3-dimensionale schichtweise Bahnplanung für Any-Time-Fräsanwendungen. In: *Robotik 2004*. VDI-Verlag, 2004, S. 781–788.

[119] Weber, W. Industrieroboter - Methoden der Steuerung und Regelung. München: Carl Hanser Verlag, 2009. ISBN: 978-3-446-41031-2.

[120] Weck, M. und C. Brecher. Werkzeugmaschinen - Maschinenarten und Anwendungsbereiche. Berlin: Springer-Verlag, 2005. ISBN: 978-3-540-22504-1.

[121] Weck, M. und C. Brecher. Werkzeugmaschinen 4 - Automatisierung von Maschinen und Anlagen. Berlin: Springer-Verlag, 2006. ISBN: 978-3-540-22507-2.

[122] Wehn, H. W. und P. R. Belanger. Ultrasound-based robot position estimation. In: *IEEE Transactions on Robotics and Automation* 13.5 (1997), S. 682–692. DOI: 10.1109/70.631230.

[123] Wörn, H., C. Wurll und D. Henrich. Automatic off-line programming and motion planning for industrial robots. In: *Proceedings of the 29th International Symposium on Robotics (ISR 1998)*. 1998.

[124] Xiongzi, L., A. L. Øyvind, C. Heping, M.-V. Sudarshan, T. A. Fuhlbrigge und M.-A. Rege. Automatic Trajectory Generation for Robotic Painting Application. In: *Proceedings for the joint conference of 41st International Symposium on Robotics (ISR 2010) und 6th German Conference on Robotics (ROBOTIK 2010)*. VDE-Verlag, 2010.

[125] Yang, G., D. Zhengchun und Y. Zhenqiang. Calibration Method of Three Dimensional (3D) Laser Measurement System Based on Projective Transformation. In: *Proceedings of the Third International Conference on Measuring Technology and Mechatronics Automation (ICMTMA 2011)*. IEEE, 2011, S. 666–671.

[126] Yin, S., Y. Ren, J. Zhu, S. Yang und S. Ye. A Vision-Based Self-Calibration Method for Robotic Visual Inspection Systems. In: *Sensors* 13 (2013), S. 16565–16582.

[127] Yokokohji, Y., Y. Kitaoka und T. Yoshikawa. Motion capture from demonstrator's viewpoint and its application to robot teaching. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2002)*. IEEE, 2002, S. 1551–1558.

[128] YuMi: Zusammen in die Zukunft der Automatisierung. You and Me. ABB Automation GmbH. 2017.

[129] Zacharias, F., C. Brost und G. Hirzinger. Capturing Robot Workspace Structure: Representing Robot Capabilities. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007)*. IEEE, 2007, S. 3229–3236.

[130] Zaidan, S. A Work-Piece Based Approach for Programming Cooperating Industrial Robots. München: Herbert Utz Verlag, 2012. ISBN: 978-3-8316-4175-8.

[131] Zeilhofer-Ficker, I. Kooperierende Roboter: Die kommende Revolution für Produktion und Dienstleistung. München: GBI-Genios Verlag, 2006.

[132] Zhang, Q. und R. Pless. Extrinsic calibration of a camera and laser range finder (improves camera calibration). In: *Proceedings of the International Conference on Intelligent Robots and Systems*. Bd. 3. Sendai, Japan: IEEE, 2004.

[133] Zhang, Z. Iterative Point Matching for Registration of Free-form Curves. In: *Technical Report RR-1658*. Valbonne Cedex, France: INRIA Sophia Antipolis, 1992.

[134] Zhang, Z. A flexible new technique for camera calibration. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (2000), S. 1330–1334.

[135] Zhu, Z., Q. Tang, J. Li und Z. Gan. Calibration of laser displacement sensor used by industrial robots. In: *Optical Engineering* 43 (2004), S. 12–13.

[136] Ziaukas, Z., K. Eggers, J. Kotlarski und T. Ortmaier. Optimizing PTP Motions of Industrial Robots through Addition of Via-points. In: *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2017)*. SCITEPRESS, 2017.

Verzeichnis promotionsbezogener, eigener Publikationen

[P1] Wagner, M., A. Buschhaus, S. Reitelshöfer, P. Heß und J. Franke. Accuracy Analysis and Improvement for Cooperative Industrial Robots. In: *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2017)*. SCITEPRESS, 2017.

[P2] Wagner, M., P. Heß und S. Reitelshöfer. Automated programming of cooperating industrial robots. In: *Proceedings of the 45st International Symposium on Robotics (ISR 2014)*. VDE-Verlag, 2014.

[P3] Wagner, M., P. Heß, S. Reitelshöfer und J. Franke. Data Fusion between a 2D Laser Profile Sensor and a Camera. In: *Proceedings of the 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2015)*. SCITEPRESS, 2015.

[P4] Wagner, M., P. Heß, S. Reitelshöfer und J. Franke. Self-Calibration Method for a Robotic Based 3D Scanning System. In: *Proceedings of the IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA 2015)*. IEEE, 2015.

[P5] Wagner, M., P. Heß, S. Reitelshöfer und J. Franke. 3D Scanning of Workpieces with Cooperative Industrial Robot Arms. In: *Proceedings of the 47st International Symposium on Robotics (ISR 2016)*. VDE-Verlag, 2016.

[P6] Wagner, M., P. Heß, S. Reitelshöfer und J. Franke. Cooperative Processing with Multi-Robot Systems. In: *Proceedings of the International Conference on Advanced Intelligent Mechatronics (AIM 2017)*. IEEE, 2017.

[P7] Wagner, M., P. Heß, S. Reitelshöfer und J. Franke. Reachability Analysis for Cooperative Processing with Industrial Robots. In: *Proceedings of the IEEE 22nd Conference on Emerging Technologies & Factory Automation (ETFA 2017)*. IEEE, 2017.

Verzeichnis promotionsbezogener, studentischer Arbeiten*

[S1] Balling, C., M. Wagner und P. Heß. Optimierung einer roboterbasierten Applikation von Klebstoffen. Bachelorarbeit. Nürnberg, 2016.

[S2] Beez, C., M. Wagner und P. Heß. Handskizzenbasierte Roboterprogrammierung. Bachelorarbeit. Nürnberg, 2015.

[S3] Brand, C., F. Schemberer und M. Wagner. Aufbau und Inbetriebnahme einer Multi-Roboter-Anlage. Projektarbeit. Nürnberg, 2014.

[S4] Bruckner, J. und M. Wagner. Entwicklung eines roboterbasierten Scan-Algorithmus zur Erfassung unbekannter Objekte. Projektarbeit. Nürnberg, 2017.

[S5] Harrer, A., M. Menßen, C. Bergner und M. Wagner. Tassenbeschriftung mit kooperierenden Robotern. Projektarbeit. Nürnberg, 2017.

[S6] Kittler, A., T. Bögelein, B. Höfler und M. Wagner. Entwicklung eines 3D-Druckers basierend auf einem Industrieroboterarm. Projektarbeit. Nürnberg, 2014.

[S7] Kögel, F. und M. Wagner. Untersuchung und Optimierung der Benutzerfreundlichkeit einer Gamepad-Steuerung für Industrieroboter. Projektarbeit. Nürnberg, 2016.

[S8] Langer, M., A. Ressel und M. Wagner. Roboterprogrammierung auf Grundlage von Punktwolken. Projektarbeit. Nürnberg, 2016.

[S9] Lehnert, R., O. Licegevic und M. Wagner. Entwicklung eines roboterbasierten 3D-Scanners. Projektarbeit. Nürnberg, 2014.

[S10] Lenda, J. und M. Wagner. Application of a 3D scanner. Projektarbeit. Nürnberg, 2013.

[S11] Merchel, C., M. Wagner, P. Heß und B. von Rymon Lipinski. Entwicklung einer Simulationsumgebung für kooperierende Industrieroboter. Bachelorarbeit. Nürnberg, 2016.

[S12] Merz, M., F. Palau und M. Wagner. Automatisierte Bauteilerkennung und Greifvorgänge. Projektarbeit. Nürnberg, 2015.

[S13] Morgenroth, M., P. Surm und M. Wagner. Erstellung eines 2D-Pfadplaners in C#. Projektarbeit. Nürnberg, 2016.

———————————

* Der Autor an zweiter (ggf. dritter, vierter oder fünfter) Stelle bezeichnet den Betreuer der studentischen Arbeit. Bei Abschlussarbeiten bezeichnet der an vorletzter Stelle genannte Autor den Zweitprüfer und der an letztgenannter Stelle den Erstprüfer.

[S14] Morgenroth, M., P. Surm, A. Winzer und M. Wagner. Optimierung eines 2D-Skizziertools für Industrieroboter-Anwendungen. Projektarbeit. Nürnberg, 2017.

[S15] Neumeier, S., H. Oeder und M. Wagner. Konstruktion einer roboterbasierten Klebstoff-Applikations-Anlage. Projektarbeit. Nürnberg, 2015.

[S16] Scharpf, F. und M. Wagner. Entwicklung eines Low-Cost-Dosiersystems für den Einsatz an Industrierobotern. Projektarbeit. Nürnberg, 2017.

[S17] Stangl, C. und M. Wagner. Optimierung einer roboterbasierten Klebstoff-Applikations-Anlage. Projektarbeit. Nürnberg, 2015.

[S18] Wodtke, D. und M. Wagner. Entwicklung einer roboterbasierten Klebstoff-Applikations-Anlage. Projektarbeit. Nürnberg, 2014.

Reihenübersicht

Koordination der Reihe (Stand 2020):
Geschäftsstelle Maschinenbau, Dr.-Ing. Oliver Kreis, www.mb.fau.de/diss/

Im Rahmen der Reihe sind bisher die nachfolgenden Bände erschienen.

Band 1 – 52
Fertigungstechnik – Erlangen
ISSN 1431-6226
Carl Hanser Verlag, München

Band 53 – 307
Fertigungstechnik – Erlangen
ISSN 1431-6226
Meisenbach Verlag, Bamberg

ab Band 308
FAU Studien aus dem Maschinenbau
ISSN 2625-9974
FAU University Press, Erlangen

Die Zugehörigkeit zu den jeweiligen Lehrstühlen ist wie folgt gekennzeichnet:

Lehrstühle:

FAPS	Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik
KTmfk	Lehrstuhl für Konstruktionstechnik
LFT	Lehrstuhl für Fertigungstechnologie
LPT	Lehrstuhl für Photonische Technologien

Band 1: Andreas Hemberger
Innovationspotentiale in der rechnerintegrierten Produktion durch wissensbasierte Systeme
FAPS, 208 Seiten, 107 Bilder. 1988.
ISBN 3-446-15234-2.

Band 2: Detlef Classe
Beitrag zur Steigerung der Flexibilität automatisierter Montagesysteme durch Sensorintegration und erweiterte Steuerungskonzepte
FAPS, 194 Seiten, 70 Bilder. 1988.
ISBN 3-446-15529-5.

Band 3: Friedrich-Wilhelm Nolting
Projektierung von Montagesystemen
FAPS, 201 Seiten, 107 Bilder, 1 Tab. 1989.
ISBN 3-446-15541-4.

Band 4: Karsten Schlüter
Nutzungsgradsteigerung von Montagesystemen durch den Einsatz der Simulationstechnik
FAPS, 177 Seiten, 97 Bilder. 1989.
ISBN 3-446-15542-2.

Band 5: Shir-Kuan Lin
Aufbau von Modellen zur Lageregelung von Industrierobotern
FAPS, 168 Seiten, 46 Bilder. 1989.
ISBN 3-446-15546-5.

Band 6: Rudolf Nuss
Untersuchungen zur Bearbeitungsqualität im Fertigungssystem Laserstrahlschneiden
LFT, 206 Seiten, 115 Bilder, 6 Tab. 1989.
ISBN 3-446-15783-2.

Band 7: Wolfgang Scholz
Modell zur datenbankgestützten Planung automatisierter Montageanlagen
FAPS, 194 Seiten, 89 Bilder. 1989.
ISBN 3-446-15825-1.

Band 8: Hans-Jürgen Wißmeier
Beitrag zur Beurteilung des Bruchverhaltens von Hartmetall-Fließpreßmatrizen
LFT, 179 Seiten, 99 Bilder, 9 Tab. 1989.
ISBN 3-446-15921-5.

Band 9: Rainer Eisele
Konzeption und Wirtschaftlichkeit von Planungssystemen in der Produktion
FAPS, 183 Seiten, 86 Bilder. 1990.
ISBN 3-446-16107-4.

Band 10: Rolf Pfeiffer
Technologisch orientierte Montageplanung am Beispiel der Schraubtechnik
FAPS, 216 Seiten, 102 Bilder, 16 Tab. 1990.
ISBN 3-446-16161-9.

Band 11: Herbert Fischer
Verteilte Planungssysteme zur Flexibilitätssteigerung der rechnerintegrierten Teilefertigung
FAPS, 201 Seiten, 82 Bilder. 1990.
ISBN 3-446-16105-8.

Band 12: Gerhard Kleineidam
CAD/CAP: Rechnergestützte Montagefeinplanung
FAPS, 203 Seiten, 107 Bilder. 1990.
ISBN 3-446-16112-0.

Band 13: Frank Vollertsen
Pulvermetallurgische Verarbeitung eines übereutektoiden verschleißfesten Stahls
LFT, XIII u. 217 Seiten, 67 Bilder, 34 Tab. 1990. ISBN 3-446-16133-3.

Band 14: Stephan Biermann
Untersuchungen zur Anlagen- und Prozeßdiagnostik für das Schneiden mit CO_2-Hochleistungslasern
LFT, VIII u. 170 Seiten, 93 Bilder, 4 Tab. 1991. ISBN 3-446-16269-0.

Band 15: Uwe Geißler
Material- und Datenfluß in einer flexiblen Blechbearbeitungszelle
LFT, 124 Seiten, 41 Bilder, 7 Tab. 1991.
ISBN 3-446-16358-1.

Band 16: Frank Oswald Hake
Entwicklung eines rechnergestützten Diagnosesystems für automatisierte Montagezellen
FAPS, XIV u. 166 Seiten, 77 Bilder. 1991.
ISBN 3-446-16428-6.

Band 17: Herbert Reichel
Optimierung der Werkzeugbereitstellung durch rechnergestützte Arbeitsfolgenbestimmung
FAPS, 198 Seiten, 73 Bilder, 2 Tab. 1991.
ISBN 3-446-16453-7.

Band 18: Josef Scheller
Modellierung und Einsatz von Softwaresystemen für rechnergeführte Montagezellen
FAPS, 198 Seiten, 65 Bilder. 1991.
ISBN 3-446-16454-5.

Band 19: Arnold vom Ende
Untersuchungen zum Biegeumforme mit elastischer Matrize
LFT, 166 Seiten, 55 Bilder, 13 Tab. 1991.
ISBN 3-446-16493-6.

Band 20: Joachim Schmid
Beitrag zum automatisierten Bearbeiten von Keramikguß mit Industrierobotern
FAPS, XIV u. 176 Seiten, 111 Bilder, 6 Tab. 1991. ISBN 3-446-16560-6.

Band 21: Egon Sommer
Multiprozessorsteuerung für kooperierende Industrieroboter in Montagezellen
FAPS, 188 Seiten, 102 Bilder. 1991.
ISBN 3-446-17062-6.

Band 22: Georg Geyer
Entwicklung problemspezifischer Verfahrensketten in der Montage
FAPS, 192 Seiten, 112 Bilder. 1991.
ISBN 3-446-16552-5.

Band 23: Rainer Flohr
Beitrag zur optimalen Verbindungstechnik in der Oberflächenmontage (SMT)
FAPS, 186 Seiten, 79 Bilder. 1991.
ISBN 3-446-16568-1.

Band 24: Alfons Rief
Untersuchungen zur Verfahrensfolge Laserstrahlschneiden und -schweißen in der Rohkarosseriefertigung
LFT, VI u. 145 Seiten, 58 Bilder, 5 Tab. 1991. ISBN 3-446-16593-2.

Band 25: Christoph Thim
Rechnerunterstützte Optimierung von Materialflußstrukturen in der Elektronikmontage durch Simulation
FAPS, 188 Seiten, 74 Bilder. 1992.
ISBN 3-446-17118-5.

Band 26: Roland Müller
CO_2 -Laserstrahlschneiden von kurzglasverstärkten Verbundwerkstoffen
LFT, 141 Seiten, 107 Bilder, 4 Tab. 1992.
ISBN 3-446-17104-5.

Band 27: Günther Schäfer
Integrierte Informationsverarbeitung bei der Montageplanung
FAPS, 195 Seiten, 76 Bilder. 1992.
ISBN 3-446-17117-7.

Band 28: Martin Hoffmann
Entwicklung einer CAD/CAM-Prozeßkette für die Herstellung von Blechbiegeteilen
LFT, 149 Seiten, 89 Bilder. 1992.
ISBN 3-446-17154-1.

Band 29: Peter Hoffmann
Verfahrensfolge Laserstrahlschneiden und –schweißen: Prozeßführung und Systemtechnik in der 3D-Laserstrahlbearbeitung von Blechformteilen
LFT, 186 Seiten, 92 Bilder, 10 Tab. 1992.
ISBN 3-446-17153-3.

Band 30: Olaf Schrödel
Flexible Werkstattsteuerung mit objektorientierten Softwarestrukturen
FAPS, 180 Seiten, 84 Bilder. 1992.
ISBN 3-446-17242-4.

Band 31: Hubert Reinisch
Planungs- und Steuerungswerkzeuge zur impliziten Geräteprogrammierung in Roboterzellen
FAPS, XI u. 212 Seiten, 112 Bilder. 1992.
ISBN 3-446-17380-3.

Band 32: Brigitte Bärnreuther
Ein Beitrag zur Bewertung des Kommunikationsverhaltens von Automatisierungsgeräten in flexiblen Produktionszellen
FAPS, XI u. 179 Seiten, 71 Bilder. 1992.
ISBN 3-446-17451-6.

Band 33: Joachim Hutfless
Laserstrahlregelung und Optikdiagnostik in der Strahlführung einer CO_2-Hochleistungslaseranlage
LFT, 175 Seiten, 70 Bilder, 17 Tab. 1993.
ISBN 3-446-17532-6.

Band 34: Uwe Günzel
Entwicklung und Einsatz eines Simulationsverfahrens für operative und strategische Probleme der Produktionsplanung und –steuerung
FAPS, XIV u. 170 Seiten, 66 Bilder, 5 Tab. 1993. ISBN 3-446-17604-7.

Band 35: Bertram Ehmann
Operatives Fertigungscontrolling durch Optimierung auftragsbezogener Bearbeitungsabläufe in der Elektronikfertigung
FAPS, XV u. 167 Seiten, 114 Bilder. 1993.
ISBN 3-446-17658-6.

Band 36: Harald Kolléra
Entwicklung eines benutzerorientierten Werkstattprogrammiersystems für das Laserstrahlschneiden
LFT, 129 Seiten, 66 Bilder, 1 Tab. 1993.
ISBN 3-446-17719-1.

Band 37: Stephanie Abels
Modellierung und Optimierung von Montageanlagen in einem integrierten Simulationssystem
FAPS, 188 Seiten, 88 Bilder. 1993.
ISBN 3-446-17731-0.

Band 38: Robert Schmidt-Hebbel
Laserstrahlbohren durchflußbestimmender Durchgangslöcher
LFT, 145 Seiten, 63 Bilder, 11 Tab. 1993.
ISBN 3-446-17778-7.

Band 39: Norbert Lutz
Oberflächenfeinbearbeitung keramischer Werkstoffe mit XeCl-Excimerlaserstrahlung
LFT, 187 Seiten, 98 Bilder, 29 Tab. 1994.
ISBN 3-446-17970-4.

Band 40: Konrad Grampp
Rechnerunterstützung bei Test und Schulung an Steuerungssoftware von SMD-Bestücklinien
FAPS, 178 Seiten, 88 Bilder. 1995.
ISBN 3-446-18173-3.

Band 41: Martin Koch
Wissensbasierte Unterstützung der Angebotsbearbeitung in der Investitionsgüterindustrie
FAPS, 169 Seiten, 68 Bilder. 1995.
ISBN 3-446-18174-1.

Band 42: Armin Gropp
Anlagen- und Prozeßdiagnostik beim Schneiden mit einem gepulsten Nd:YAG-Laser
LFT, 160 Seiten, 88 Bilder, 7 Tab. 1995.
ISBN 3-446-18241-1.

Band 43: Werner Heckel
Optische 3D-Konturerfassung und on-line Biegewinkelmessung mit dem Lichtschnittverfahren
LFT, 149 Seiten, 43 Bilder, 11 Tab. 1995.
ISBN 3-446-18243-8.

Band 44: Armin Rothhaupt
Modulares Planungssystem zur Optimierung der Elektronikfertigung
FAPS, 180 Seiten, 101 Bilder. 1995.
ISBN 3-446-18307-8.

Band 45: Bernd Zöllner
Adaptive Diagnose in der Elektronikproduktion
FAPS, 195 Seiten, 74 Bilder, 3 Tab. 1995.
ISBN 3-446-18308-6.

Band 46: Bodo Vormann
Beitrag zur automatisierten Handhabungsplanung komplexer Blechbiegeteile
LFT, 126 Seiten, 89 Bilder, 3 Tab. 1995.
ISBN 3-446-18345-0.

Band 47: Peter Schnepf
Zielkostenorientierte Montageplanung
FAPS, 144 Seiten, 75 Bilder. 1995.
ISBN 3-446-18397-3.

Band 48: Rainer Klotzbücher
Konzept zur rechnerintegrierten Materialversorgung in flexiblen Fertigungssystemen
FAPS, 156 Seiten, 62 Bilder. 1995.
ISBN 3-446-18412-0.

Band 49: Wolfgang Greska
Wissensbasierte Analyse und Klassifizierung von Blechteilen
LFT, 144 Seiten, 96 Bilder. 1995.
ISBN 3-446-18462-7.

Band 50: Jörg Franke
Integrierte Entwicklung neuer Produkt- und Produktionstechnologien für räumliche spritzgegossene Schaltungsträger (3-D MID)
FAPS, 196 Seiten, 86 Bilder, 4 Tab. 1995.
ISBN 3-446-18448-1.

Band 51: Franz-Josef Zeller
Sensorplanung und schnelle Sensorregelung für Industrieroboter
FAPS, 190 Seiten, 102 Bilder, 9 Tab. 1995.
ISBN 3-446-18601-8.

Band 52: Michael Solvie
Zeitbehandlung und Multimedia-Unterstützung in Feldkommunikationssystemen
FAPS, 200 Seiten, 87 Bilder, 35 Tab. 1996.
ISBN 3-446-18607-7.

Band 53: Robert Hopperdietzel
Reengineering in der Elektro- und Elektronikindustrie
FAPS, 180 Seiten, 109 Bilder, 1 Tab. 1996.
ISBN 3-87525-070-2.

Band 54: Thomas Rebhahn
Beitrag zur Mikromaterialbearbeitung mit Excimerlasern - Systemkomponenten und Verfahrensoptimierungen
LFT, 148 Seiten, 61 Bilder, 10 Tab. 1996.
ISBN 3-87525-075-3.

Band 55: Henning Hanebuth
Laserstrahlhartlöten mit Zweistrahltechnik
LFT, 157 Seiten, 58 Bilder, 11 Tab. 1996.
ISBN 3-87525-074-5.

Band 56: Uwe Schönherr
Steuerung und Sensordatenintegration für flexible Fertigungszellen mit kooperierenden Robotern
FAPS, 188 Seiten, 116 Bilder, 3 Tab. 1996.
ISBN 3-87525-076-1.

Band 57: Stefan Holzer
Berührungslose Formgebung mit Laserstrahlung
LFT, 162 Seiten, 69 Bilder, 11 Tab. 1996.
ISBN 3-87525-079-6.

Band 58: Markus Schultz
Fertigungsqualität beim 3D-Laserstrahlschweißen von Blechformteilen
LFT, 165 Seiten, 88 Bilder, 9 Tab. 1997.
ISBN 3-87525-080-X.

Band 59: Thomas Krebs
Integration elektromechanischer CA-Anwendungen über einem STEP-Produktmodell
FAPS, 198 Seiten, 58 Bilder, 8 Tab. 1997.
ISBN 3-87525-081-8.

Band 60: Jürgen Sturm
Prozeßintegrierte Qualitätssicherung in der Elektronikproduktion
FAPS, 167 Seiten, 112 Bilder, 5 Tab. 1997.
ISBN 3-87525-082-6.

Band 61: Andreas Brand
Prozesse und Systeme zur Bestückung räumlicher elektronischer Baugruppen (3D-MID)
FAPS, 182 Seiten, 100 Bilder. 1997.
ISBN 3-87525-087-7.

Band 62: Michael Kauf
Regelung der Laserstrahlleistung und der Fokusparameter einer CO_2-Hochleistungslaseranlage
LFT, 140 Seiten, 70 Bilder, 5 Tab. 1997.
ISBN 3-87525-083-4.

Band 63: Peter Steinwasser
Modulares Informationsmanagement in der integrierten Produkt- und Prozeßplanung
FAPS, 190 Seiten, 87 Bilder. 1997.
ISBN 3-87525-084-2.

Band 64: Georg Liedl
Integriertes Automatisierungskonzept für den flexiblen Materialfluß in der Elektronikproduktion
FAPS, 196 Seiten, 96 Bilder, 3 Tab. 1997.
ISBN 3-87525-086-9.

Band 65: Andreas Otto
Transiente Prozesse beim Laserstrahlschweißen
LFT, 132 Seiten, 62 Bilder, 1 Tab. 1997.
ISBN 3-87525-089-3.

Band 66: Wolfgang Blöchl
Erweiterte Informationsbereitstellung an offenen CNC-Steuerungen zur Prozeß- und Programmoptimierung
FAPS, 168 Seiten, 96 Bilder. 1997.
ISBN 3-87525-091-5.

Band 67: Klaus-Uwe Wolf
Verbesserte Prozeßführung und Prozeßplanung zur Leistungs- und Qualitätssteigerung beim Spulenwickeln
FAPS, 186 Seiten, 125 Bilder. 1997.
ISBN 3-87525-092-3.

Band 68: Frank Backes
Technologieorientierte Bahnplanung für die 3D-Laserstrahlbearbeitung
LFT, 138 Seiten, 71 Bilder, 2 Tab. 1997.
ISBN 3-87525-093-1.

Band 69: Jürgen Kraus
Laserstrahlumformen von Profilen
LFT, 137 Seiten, 72 Bilder, 8 Tab. 1997.
ISBN 3-87525-094-X.

Band 70: Norbert Neubauer
Adaptive Strahlführungen für CO_2-Laseranlagen
LFT, 120 Seiten, 50 Bilder, 3 Tab. 1997.
ISBN 3-87525-095-8.

Band 71: Michael Steber
Prozeßoptimierter Betrieb flexibler Schraubstationen in der automatisierten Montage
FAPS, 168 Seiten, 78 Bilder, 3 Tab. 1997.
ISBN 3-87525-096-6.

Band 72: Markus Pfestorf
Funktionale 3D-Oberflächenkenngrößen in der Umformtechnik
LFT, 162 Seiten, 84 Bilder, 15 Tab. 1997.
ISBN 3-87525-097-4.

Band 73: Volker Franke
Integrierte Planung und Konstruktion von Werkzeugen für die Biegebearbeitung
LFT, 143 Seiten, 81 Bilder. 1998.
ISBN 3-87525-098-2.

Band 74: Herbert Scheller
Automatisierte Demontagesysteme und recyclinggerechte Produktgestaltung elektronischer Baugruppen
FAPS, 184 Seiten, 104 Bilder, 17 Tab. 1998.
ISBN 3-87525-099-0.

Band 75: Arthur Meßner
Kaltmassivumformung metallischer Kleinstteile - Werkstoffverhalten, Wirkflächenreibung, Prozeßauslegung
LFT, 164 Seiten, 92 Bilder, 14 Tab. 1998.
ISBN 3-87525-100-8.

Band 76: Mathias Glasmacher
Prozeß- und Systemtechnik zum Laserstrahl-Mikroschweißen
LFT, 184 Seiten, 104 Bilder, 12 Tab. 1998.
ISBN 3-87525-101-6.

Band 77: Michael Schwind
Zerstörungsfreie Ermittlung mechanischer Eigenschaften von Feinblechen mit dem Wirbelstromverfahren
LFT, 124 Seiten, 68 Bilder, 8 Tab. 1998.
ISBN 3-87525-102-4.

Band 78: Manfred Gerhard
Qualitätssteigerung in der Elektronikproduktion durch Optimierung der Prozeßführung beim Löten komplexer Baugruppen
FAPS, 179 Seiten, 113 Bilder, 7 Tab. 1998.
ISBN 3-87525-103-2.

Band 79: Elke Rauh
Methodische Einbindung der Simulation in die betrieblichen Planungs- und Entscheidungsabläufe
FAPS, 192 Seiten, 114 Bilder, 4 Tab. 1998.
ISBN 3-87525-104-0.

Band 80: Sorin Niederkorn
Meßeinrichtung zur Untersuchung der Wirkflächenreibung bei umformtechnischen Prozessen
LFT, 99 Seiten, 46 Bilder, 6 Tab. 1998.
ISBN 3-87525-105-9.

Band 81: Stefan Schuberth
Regelung der Fokuslage beim Schweißen mit CO2-Hochleistungslasern unter Einsatz von adaptiven Optiken
LFT, 140 Seiten, 64 Bilder, 3 Tab. 1998.
ISBN 3-87525-106-7.

Band 82: Armando Walter Colombo
Development and Implementation of Hierarchical Control Structures of Flexible Production Systems Using High Level Petri Nets
FAPS, 216 Seiten, 86 Bilder. 1998.
ISBN 3-87525-109-1.

Band 83: Otto Meedt
Effizienzsteigerung bei Demontage und Recycling durch flexible Demontagetechnologien und optimierte Produktgestaltung
FAPS, 186 Seiten, 103 Bilder. 1998.
ISBN 3-87525-108-3.

Band 84: Knuth Götz
Modelle und effiziente Modellbildung zur Qualitätssicherung in der Elektronikproduktion
FAPS, 212 Seiten, 129 Bilder, 24 Tab. 1998.
ISBN 3-87525-112-1.

Band 85: Ralf Luchs
Einsatzmöglichkeiten leitender Klebstoffe zur zuverlässigen Kontaktierung elektronischer Bauelemente in der SMT
FAPS, 176 Seiten, 126 Bilder, 30 Tab. 1998.
ISBN 3-87525-113-7.

Band 86: Frank Pöhlau
Entscheidungsgrundlagen zur Einführung räumlicher spritzgegossener Schaltungsträger (3-D MID)
FAPS, 144 Seiten, 99 Bilder. 1999.
ISBN 3-87525-114-8.

Band 87: Roland T. A. Kals
Fundamentals on the miniaturization of sheet metal working processes
LFT, 128 Seiten, 58 Bilder, 11 Tab. 1999.
ISBN 3-87525-115-6.

Band 88: Gerhard Luhn
Implizites Wissen und technisches Handeln am Beispiel der Elektronikproduktion
FAPS, 252 Seiten, 61 Bilder, 1 Tab. 1999.
ISBN 3-87525-116-4.

Band 89: Axel Sprenger
Adaptives Streckbiegen von Aluminium-Strangpreßprofilen
LFT, 114 Seiten, 63 Bilder, 4 Tab. 1999.
ISBN 3-87525-117-2.

Band 90: Hans-Jörg Pucher
Untersuchungen zur Prozeßfolge Umformen, Bestücken und Laserstrahllöten von Mikrokontakten
LFT, 158 Seiten, 69 Bilder, 9 Tab. 1999.
ISBN 3-87525-119-9.

Band 91: Horst Arnet
Profilbiegen mit kinematischer Gestalterzeugung
LFT, 128 Seiten, 67 Bilder, 7 Tab. 1999.
ISBN 3-87525-120-2.

Band 92: Doris Schubart
Prozeßmodellierung und Technologieentwicklung beim Abtragen mit CO_2-Laserstrahlung
LFT, 133 Seiten, 57 Bilder, 13 Tab. 1999.
ISBN 3-87525-122-9.

Band 93: Adrianus L. P. Coremans
Laserstrahlsintern von Metallpulver - Prozeßmodellierung, Systemtechnik, Eigenschaften laserstrahlgesinterter Metallkörper
LFT, 184 Seiten, 108 Bilder, 12 Tab. 1999.
ISBN 3-87525-124-5.

Band 94: Hans-Martin Biehler
Optimierungskonzepte für Qualitätsdatenverarbeitung und Informationsbereitstellung in der Elektronikfertigung
FAPS, 194 Seiten, 105 Bilder. 1999.
ISBN 3-87525-126-1.

Band 95: Wolfgang Becker
Oberflächenausbildung und tribologische Eigenschaften excimerlaserstrahlbearbeiteter Hochleistungskeramiken
LFT, 175 Seiten, 71 Bilder, 3 Tab. 1999.
ISBN 3-87525-127-X.

Band 96: Philipp Hein
Innenhochdruck-Umformen von Blechpaaren: Modellierung, Prozeßauslegung und Prozeßführung
LFT, 129 Seiten, 57 Bilder, 7 Tab. 1999.
ISBN 3-87525-128-8.

Band 97: Gunter Beitinger
Herstellungs- und Prüfverfahren für thermoplastische Schaltungsträger
FAPS, 169 Seiten, 92 Bilder, 20 Tab. 1999.
ISBN 3-87525-129-6.

Band 98: Jürgen Knoblach
Beitrag zur rechnerunterstützten verursachungsgerechten Angebotskalkulation von Blechteilen mit Hilfe wissensbasierter Methoden
LFT, 155 Seiten, 53 Bilder, 26 Tab. 1999.
ISBN 3-87525-130-X.

Band 99: Frank Breitenbach
Bildverarbeitungssystem zur Erfassung der Anschlußgeometrie elektronischer SMT-Bauelemente
LFT, 147 Seiten, 92 Bilder, 12 Tab. 2000.
ISBN 3-87525-131-8.

Band 100: Bernd Falk
Simulationsbasierte Lebensdauervorhersage für Werkzeuge der Kaltmassivumformung
LFT, 134 Seiten, 44 Bilder, 15 Tab. 2000.
ISBN 3-87525-136-9.

Band 101: Wolfgang Schlögl
Integriertes Simulationsdaten-Management für Maschinenentwicklung und Anlagenplanung
FAPS, 169 Seiten, 101 Bilder, 20 Tab. 2000.
ISBN 3-87525-137-7.

Band 102: Christian Hinsel
Ermüdungsbruchversagen hartstoffbeschichteter Werkzeugstähle in der Kaltmassivumformung
LFT, 130 Seiten, 80 Bilder, 14 Tab. 2000.
ISBN 3-87525-138-5.

Band 103: Stefan Bobbert
Simulationsgestützte Prozessauslegung für das Innenhochdruck-Umformen von Blechpaaren
LFT, 123 Seiten, 77 Bilder. 2000.
ISBN 3-87525-145-8.

Band 104: Harald Rottbauer
Modulares Planungswerkzeug zum Produktionsmanagement in der Elektronikproduktion
FAPS, 166 Seiten, 106 Bilder. 2001.
ISBN 3-87525-139-3.

Band 111: Jürgen Göhringer
Integrierte Telediagnose via Internet zum effizienten Service von Produktionssystemen
FAPS, 178 Seiten, 98 Bilder, 5 Tab. 2001.
ISBN 3-87525-147-4.

Band 105: Thomas Hennige
Flexible Formgebung von Blechen durch Laserstrahlumformen
LFT, 119 Seiten, 50 Bilder. 2001.
ISBN 3-87525-140-7.

Band 106: Thomas Menzel
Wissensbasierte Methoden für die rechnergestützte Charakterisierung und Bewertung innovativer Fertigungsprozesse
LFT, 152 Seiten, 71 Bilder. 2001.
ISBN 3-87525-142-3.

Band 107: Thomas Stöckel
Kommunikationstechnische Integration der Prozeßebene in Produktionssysteme durch Middleware-Frameworks
FAPS, 147 Seiten, 65 Bilder, 5 Tab. 2001.
ISBN 3-87525-143-1.

Band 108: Frank Pitter
Verfügbarkeitssteigerung von Werkzeugmaschinen durch Einsatz mechatronischer Sensorlösungen
FAPS, 158 Seiten, 131 Bilder, 8 Tab. 2001.
ISBN 3-87525-144-X.

Band 109: Markus Korneli
Integration lokaler CAP-Systeme in einen globalen Fertigungsdatenverbund
FAPS, 121 Seiten, 53 Bilder, 11 Tab. 2001.
ISBN 3-87525-146-6.

Band 110: Burkhard Müller
Laserstrahljustieren mit Excimer-Lasern - Prozeßparameter und Modelle zur Aktorkonstruktion
LFT, 128 Seiten, 36 Bilder, 9 Tab. 2001.
ISBN 3-87525-159-8.

Band 111: Jürgen Göhringer
Integrierte Telediagnose via Internet zum effizienten Service von Produktionssystemen
FAPS, 178 Seiten, 98 Bilder, 5 Tab. 2001.
ISBN 3-87525-147-4.

Band 112: Robert Feuerstein
Qualitäts- und kosteneffiziente Integration neuer Bauelementetechnologien in die Flachbaugruppenfertigung
FAPS, 161 Seiten, 99 Bilder, 10 Tab. 2001.
ISBN 3-87525-151-2.

Band 113: Marcus Reichenberger
Eigenschaften und Einsatzmöglichkeiten alternativer Elektroniklote in der Oberflächenmontage (SMT)
FAPS, 165 Seiten, 97 Bilder, 18 Tab. 2001.
ISBN 3-87525-152-0.

Band 114: Alexander Huber
Justieren vormontierter Systeme mit dem Nd:YAG-Laser unter Einsatz von Aktoren
LFT, 122 Seiten, 58 Bilder, 5 Tab. 2001.
ISBN 3-87525-153-9.

Band 115: Sami Krimi
Analyse und Optimierung von Montagesystemen in der Elektronikproduktion
FAPS, 155 Seiten, 88 Bilder, 3 Tab. 2001.
ISBN 3-87525-157-1.

Band 116: Marion Merklein
Laserstrahlumformen von Aluminiumwerkstoffen - Beeinflussung der Mikrostruktur und der mechanischen Eigenschaften
LFT, 122 Seiten, 65 Bilder, 15 Tab. 2001.
ISBN 3-87525-156-3.

Band 117: Thomas Collisi
Ein informationslogistisches Architekturkonzept zur Akquisition simulationsrelevanter Daten
FAPS, 181 Seiten, 105 Bilder, 7 Tab. 2002.
ISBN 3-87525-164-4.

Band 118: Markus Koch
Rationalisierung und ergonomische Optimierung im Innenausbau durch den Einsatz moderner Automatisierungstechnik
FAPS, 176 Seiten, 98 Bilder, 9 Tab. 2002.
ISBN 3-87525-165-2.

Band 119: Michael Schmidt
Prozeßregelung für das Laserstrahl-Punktschweißen in der Elektronikproduktion
LFT, 152 Seiten, 71 Bilder, 3 Tab. 2002.
ISBN 3-87525-166-0.

Band 120: Nicolas Tiesler
Grundlegende Untersuchungen zum Fließpressen metallischer Kleinstteile
LFT, 126 Seiten, 78 Bilder, 12 Tab. 2002.
ISBN 3-87525-175-X.

Band 121: Lars Pursche
Methoden zur technologieorientierten Programmierung für die 3D-Lasermikrobearbeitung
LFT, 111 Seiten, 39 Bilder, 0 Tab. 2002.
ISBN 3-87525-183-0.

Band 122: Jan-Oliver Brassel
Prozeßkontrolle beim Laserstrahl-Mikroschweißen
LFT, 148 Seiten, 72 Bilder, 12 Tab. 2002.
ISBN 3-87525-181-4.

Band 123: Mark Geisel
Prozeßkontrolle und -steuerung beim Laserstrahlschweißen mit den Methoden der nichtlinearen Dynamik
LFT, 135 Seiten, 46 Bilder, 2 Tab. 2002.
ISBN 3-87525-180-6.

Band 124: Gerd Eßer
Laserstrahlunterstützte Erzeugung metallischer Leiterstrukturen auf Thermoplastsubstraten für die MID-Technik
LFT, 148 Seiten, 60 Bilder, 6 Tab. 2002.
ISBN 3-87525-171-7.

Band 125: Marc Fleckenstein
Qualität laserstrahl-gefügter Mikroverbindungen elektronischer Kontakte
LFT, 159 Seiten, 77 Bilder, 7 Tab. 2002.
ISBN 3-87525-170-9.

Band 126: Stefan Kaufmann
Grundlegende Untersuchungen zum Nd:YAG- Laserstrahlfügen von Silizium für Komponenten der Optoelektronik
LFT, 159 Seiten, 100 Bilder, 6 Tab. 2002.
ISBN 3-87525-172-5.

Band 127: Thomas Fröhlich
Simultanes Löten von Anschlußkontakten elektronischer Bauelemente mit Diodenlaserstrahlung
LFT, 143 Seiten, 75 Bilder, 6 Tab. 2002.
ISBN 3-87525-186-5.

Band 128: Achim Hofmann
Erweiterung der Formgebungsgrenzen beim Umformen von Aluminiumwerkstoffen durch den Einsatz prozessangepasster Platinen
LFT, 113 Seiten, 58 Bilder, 4 Tab. 2002.
ISBN 3-87525-182-2.

Band 129: Ingo Kriebitzsch
3 - D MID Technologie in der Automobilelektronik
FAPS, 129 Seiten, 102 Bilder, 10 Tab. 2002.
ISBN 3-87525-169-5.

Band 130: Thomas Pohl
Fertigungsqualität und Umformbarkeit laserstrahlgeschweißter Formplatinen aus Aluminiumlegierungen
LFT, 133 Seiten, 93 Bilder, 12 Tab. 2002.
ISBN 3-87525-173-3.

Band 131: Matthias Wenk
Entwicklung eines konfigurierbaren Steuerungssystems für die flexible Sensorführung von Industrierobotern
FAPS, 167 Seiten, 85 Bilder, 1 Tab. 2002.
ISBN 3-87525-174-1.

Band 132: Matthias Negendanck
Neue Sensorik und Aktorik für Bearbeitungsköpfe zum Laserstrahlschweißen
LFT, 116 Seiten, 60 Bilder, 14 Tab. 2002.
ISBN 3-87525-184-9.

Band 133: Oliver Kreis
Integrierte Fertigung - Verfahrensintegration durch Innenhochdruck-Umformen, Trennen und Laserstrahlschweißen in einem Werkzeug sowie ihre tele- und multimediale Präsentation
LFT, 167 Seiten, 90 Bilder, 43 Tab. 2002.
ISBN 3-87525-176-8.

Band 134: Stefan Trautner
Technische Umsetzung produktbezogener Instrumente der Umweltpolitik bei Elektro- und Elektronikgeräten
FAPS, 179 Seiten, 92 Bilder, 11 Tab. 2002.
ISBN 3-87525-177-6.

Band 135: Roland Meier
Strategien für einen produktorientierten Einsatz räumlicher spritzgegossener Schaltungsträger (3-D MID)
FAPS, 155 Seiten, 88 Bilder, 14 Tab. 2002.
ISBN 3-87525-178-4.

Band 136: Jürgen Wunderlich
Kostensimulation - Simulationsbasierte Wirtschaftlichkeitsregelung komplexer Produktionssysteme
FAPS, 202 Seiten, 119 Bilder, 17 Tab. 2002.
ISBN 3-87525-179-2.

Band 137: Stefan Novotny
Innenhochdruck-Umformen von Blechen aus Aluminium- und Magnesiumlegierungen bei erhöhter Temperatur
LFT, 132 Seiten, 82 Bilder, 6 Tab. 2002.
ISBN 3-87525-185-7.

Band 138: Andreas Licha
Flexible Montageautomatisierung zur Komplettmontage flächenhafter Produktstrukturen durch kooperierende Industrieroboter
FAPS, 158 Seiten, 87 Bilder, 8 Tab. 2003.
ISBN 3-87525-189-X.

Band 139: Michael Eisenbarth
Beitrag zur Optimierung der Aufbau- und Verbindungstechnik für mechatronische Baugruppen
FAPS, 207 Seiten, 141 Bilder, 9 Tab. 2003.
ISBN 3-87525-190-3.

Band 140: Frank Christoph
Durchgängige simulationsgestützte Planung von Fertigungseinrichtungen der Elektronikproduktion
FAPS, 187 Seiten, 107 Bilder, 9 Tab. 2003.
ISBN 3-87525-191-1.

Band 141: Hinnerk Hagenah
Simulationsbasierte Bestimmung der zu erwartenden Maßhaltigkeit für das Blechbiegen
LFT, 131 Seiten, 36 Bilder, 26 Tab. 2003.
ISBN 3-87525-192-X.

Band 142: Ralf Eckstein
Scherschneiden und Biegen metallischer Kleinstteile - Materialeinfluss und Materialverhalten
LFT, 148 Seiten, 71 Bilder, 19 Tab. 2003.
ISBN 3-87525-193-8.

Band 143: Frank H. Meyer-Pittroff
Excimerlaserstrahlbiegen dünner metallischer Folien mit homogener Lichtlinie
LFT, 138 Seiten, 60 Bilder, 16 Tab. 2003.
ISBN 3-87525-196-2.

Band 144: Andreas Kach
Rechnergestützte Anpassung von Laserstrahlschneidbahnen an Bauteilabweichungen
LFT, 139 Seiten, 69 Bilder, 11 Tab. 2004.
ISBN 3-87525-197-0.

Band 145: Stefan Hierl
System- und Prozeßtechnik für das simultane Löten mit Diodenlaserstrahlung von elektronischen Bauelementen
LFT, 124 Seiten, 66 Bilder, 4 Tab. 2004.
ISBN 3-87525-198-9.

Band 146: Thomas Neudecker
Tribologische Eigenschaften keramischer Blechumformwerkzeuge- Einfluss einer Oberflächenendbearbeitung mittels Excimerlaserstrahlung
LFT, 166 Seiten, 75 Bilder, 26 Tab. 2004.
ISBN 3-87525-200-4.

Band 147: Ulrich Wenger
Prozessoptimierung in der Wickeltechnik durch innovative maschinenbauliche und regelungstechnische Ansätze
FAPS, 132 Seiten, 88 Bilder, 0 Tab. 2004.
ISBN 3-87525-203-9.

Band 148: Stefan Slama
Effizienzsteigerung in der Montage durch marktorientierte Montagestrukturen und erweiterte Mitarbeiterkompetenz
FAPS, 188 Seiten, 125 Bilder, 0 Tab. 2004.
ISBN 3-87525-204-7.

Band 149: Thomas Wurm
Laserstrahljustieren mittels Aktoren-Entwicklung von Konzepten und Methoden für die rechnerunterstützte Modellierung und Optimierung von komplexen Aktorsystemen in der Mikrotechnik
LFT, 122 Seiten, 51 Bilder, 9 Tab. 2004.
ISBN 3-87525-206-3.

Band 150: Martino Celeghini
Wirkmedienbasierte Blechumformung: Grundlagenuntersuchungen zum Einfluss von Werkstoff und Bauteilgeometrie
LFT, 146 Seiten, 77 Bilder, 6 Tab. 2004.
ISBN 3-87525-207-1.

Band 151: Ralph Hohenstein
Entwurf hochdynamischer Sensor- und Regelsysteme für die adaptive Laserbearbeitung
LFT, 282 Seiten, 63 Bilder, 16 Tab. 2004.
ISBN 3-87525-210-1.

Band 152: Angelika Hutterer
Entwicklung prozessüberwachender Regelkreise für flexible Formgebungsprozesse
LFT, 149 Seiten, 57 Bilder, 2 Tab. 2005.
ISBN 3-87525-212-8.

Band 153: Emil Egerer
Massivumformen metallischer Kleinstteile bei erhöhter Prozesstemperatur
LFT, 158 Seiten, 87 Bilder, 10 Tab. 2005.
ISBN 3-87525-213-6.

Band 154: Rüdiger Holzmann
Strategien zur nachhaltigen Optimierung von Qualität und Zuverlässigkeit in der Fertigung hochintegrierter Flachbaugruppen
FAPS, 186 Seiten, 99 Bilder, 19 Tab. 2005.
ISBN 3-87525-217-9.

Band 155: Marco Nock
Biegeumformen mit Elastomerwerkzeugen Modellierung, Prozessauslegung und Abgrenzung des Verfahrens am Beispiel des Rohrbiegens
LFT, 164 Seiten, 85 Bilder, 13 Tab. 2005.
ISBN 3-87525-218-7.

Band 156: Frank Niebling
Qualifizierung einer Prozesskette zum Laserstrahlsintern metallischer Bauteile
LFT, 148 Seiten, 89 Bilder, 3 Tab. 2005.
ISBN 3-87525-219-5.

Band 157: Markus Meiler
Großserientauglichkeit trockenschmierstoffbeschichteter Aluminiumbleche im Presswerk Grundlegende Untersuchungen zur Tribologie, zum Umformverhalten und Bauteilversuche
LFT, 104 Seiten, 57 Bilder, 21 Tab. 2005.
ISBN 3-87525-221-7.

Band 158: Agus Sutanto
Solution Approaches for Planning of Assembly Systems in Three-Dimensional Virtual Environments
FAPS, 169 Seiten, 98 Bilder, 3 Tab. 2005.
ISBN 3-87525-220-9.

Band 159: Matthias Boiger
Hochleistungssysteme für die Fertigung elektronischer Baugruppen auf der Basis flexibler Schaltungsträger
FAPS, 175 Seiten, 111 Bilder, 8 Tab. 2005.
ISBN 3-87525-222-5.

Band 160: Matthias Pitz
Laserunterstütztes Biegen höchstfester Mehrphasenstähle
LFT, 120 Seiten, 73 Bilder, 11 Tab. 2005.
ISBN 3-87525-223-3.

Band 161: Meik Vahl
Beitrag zur gezielten Beeinflussung des Werkstoffflusses beim Innenhochdruck-Umformen von Blechen
LFT, 165 Seiten, 94 Bilder, 15 Tab. 2005.
ISBN 3-87525-224-1.

Band 162: Peter K. Kraus
Plattformstrategien - Realisierung einer varianz- und kostenoptimierten Wertschöpfung
FAPS, 181 Seiten, 95 Bilder, 0 Tab. 2005.
ISBN 3-87525-226-8.

Band 163: Adrienn Cser
Laserstrahlschmelzabtrag - Prozessanalyse und -modellierung
LFT, 146 Seiten, 79 Bilder, 3 Tab. 2005.
ISBN 3-87525-227-6.

Band 164: Markus C. Hahn
Grundlegende Untersuchungen zur Herstellung von Leichtbauverbundstrukturen mit Aluminiumschaumkern
LFT, 143 Seiten, 60 Bilder, 16 Tab. 2005.
ISBN 3-87525-228-4.

Band 165: Gordana Michos
Mechatronische Ansätze zur Optimierung von Vorschubachsen
FAPS, 146 Seiten, 87 Bilder, 17 Tab. 2005.
ISBN 3-87525-230-6.

Band 166: Markus Stark
Auslegung und Fertigung hochpräziser Faser-Kollimator-Arrays
LFT, 158 Seiten, 115 Bilder, 11 Tab. 2005.
ISBN 3-87525-231-4.

Band 167: Yurong Zhou
Kollaboratives Engineering Management in der integrierten virtuellen Entwicklung der Anlagen für die Elektronikproduktion
FAPS, 156 Seiten, 84 Bilder, 6 Tab. 2005.
ISBN 3-87525-232-2.

Band 168: Werner Enser
Neue Formen permanenter und lösbarer elektrischer Kontaktierungen für mechatronische Baugruppen
FAPS, 190 Seiten, 112 Bilder, 5 Tab. 2005.
ISBN 3-87525-233-0.

Band 169: Katrin Melzer
Integrierte Produktpolitik bei elektrischen und elektronischen Geräten zur Optimierung des Product-Life-Cycle
FAPS, 155 Seiten, 91 Bilder, 17 Tab. 2005.
ISBN 3-87525-234-9.

Band 170: Alexander Putz
Grundlegende Untersuchungen zur Erfassung der realen Vorspannung von armierten Kaltfließpresswerkzeugen mittels Ultraschall
LFT, 137 Seiten, 71 Bilder, 15 Tab. 2006.
ISBN 3-87525-237-3.

Band 171: Martin Prechtl
Automatisiertes Schichtverfahren für metallische Folien - System- und Prozesstechnik
LFT, 154 Seiten, 45 Bilder, 7 Tab. 2006.
ISBN 3-87525-238-1.

Band 172: Markus Meidert
Beitrag zur deterministischen Lebensdauerabschätzung von Werkzeugen der Kaltmassivumformung
LFT, 131 Seiten, 78 Bilder, 9 Tab. 2006.
ISBN 3-87525-239-X.

Band 173: Bernd Müller
Robuste, automatisierte Montagesysteme durch adaptive Prozessführung und montageübergreifende Fehlerprävention am Beispiel flächiger Leichtbauteile
FAPS, 147 Seiten, 77 Bilder, 0 Tab. 2006.
ISBN 3-87525-240-3.

Band 174: Alexander Hofmann
Hybrides Laserdurchstrahlschweißen von Kunststoffen
LFT, 136 Seiten, 72 Bilder, 4 Tab. 2006.
ISBN 978-3-87525-243-9.

Band 175: Peter Wölflick
Innovative Substrate und Prozesse mit feinsten Strukturen für bleifreie Mechatronik-Anwendungen
FAPS, 177 Seiten, 148 Bilder, 24 Tab. 2006.
ISBN 978-3-87525-246-0.

Band 176: Attila Komlodi
Detection and Prevention of Hot Cracks during Laser Welding of Aluminium Alloys Using Advanced Simulation Methods
LFT, 155 Seiten, 89 Bilder, 14 Tab. 2006.
ISBN 978-3-87525-248-4.

Band 177: Uwe Popp
Grundlegende Untersuchungen zum Laserstrahlstrukturieren von Kaltmassivumformwerkzeugen
LFT, 140 Seiten, 67 Bilder, 16 Tab. 2006.
ISBN 978-3-87525-249-1.

Band 178: Veit Rückel
Rechnergestützte Ablaufplanung und Bahngenerierung Für kooperierende Industrieroboter
FAPS, 148 Seiten, 75 Bilder, 7 Tab. 2006.
ISBN 978-3-87525-250-7.

Band 179: Manfred Dirscherl
Nicht-thermische Mikrojustiertechnik mittels ultrakurzer Laserpulse
LFT, 154 Seiten, 69 Bilder, 10 Tab. 2007.
ISBN 978-3-87525-251-4.

Band 180: Yong Zhuo
Entwurf eines rechnergestützten integrierten Systems für Konstruktion und Fertigungsplanung räumlicher spritzgegossener Schaltungsträger (3D-MID)
FAPS, 181 Seiten, 95 Bilder, 5 Tab. 2007.
ISBN 978-3-87525-253-8.

Band 181: Stefan Lang
Durchgängige Mitarbeiterinformation zur Steigerung von Effizienz und Prozesssicherheit in der Produktion
FAPS, 172 Seiten, 93 Bilder. 2007.
ISBN 978-3-87525-257-6.

Band 182: Hans-Joachim Krauß
Laserstrahlinduzierte Pyrolyse präkeramischer Polymere
LFT, 171 Seiten, 100 Bilder. 2007.
ISBN 978-3-87525-258-3.

Band 183: Stefan Junker
Technologien und Systemlösungen für die flexibel automatisierte Bestückung permanent erregter Läufer mit oberflächenmontierten Dauermagneten
FAPS, 173 Seiten, 75 Bilder. 2007.
ISBN 978-3-87525-259-0.

Band 184: Rainer Kohlbauer
Wissensbasierte Methoden für die simulationsgestützte Auslegung wirkmedienbasierter Blechumformprozesse
LFT, 135 Seiten, 50 Bilder. 2007.
ISBN 978-3-87525-260-6.

Band 185: Klaus Lamprecht
Wirkmedienbasierte Umformung tiefgezogener Vorformen unter besonderer Berücksichtigung maßgeschneiderter Halbzeuge
LFT, 137 Seiten, 81 Bilder. 2007.
ISBN 978-3-87525-265-1.

Band 186: Bernd Zolleiß
Optimierte Prozesse und Systeme für die Bestückung mechatronischer Baugruppen
FAPS, 180 Seiten, 117 Bilder. 2007.
ISBN 978-3-87525-266-8.

Band 187: Michael Kerausch
Simulationsgestützte Prozessauslegung für das Umformen lokal wärmebehandelter Aluminiumplatinen
LFT, 146 Seiten, 76 Bilder, 7 Tab. 2007.
ISBN 978-3-87525-267-5.

Band 188: Matthias Weber
Unterstützung der Wandlungsfähigkeit von Produktionsanlagen durch innovative Softwaresysteme
FAPS, 183 Seiten, 122 Bilder, 3 Tab. 2007.
ISBN 978-3-87525-269-9.

Band 189: Thomas Frick
Untersuchung der prozessbestimmenden Strahl-Stoff-Wechselwirkungen beim Laserstrahlschweißen von Kunststoffen
LFT, 104 Seiten, 62 Bilder, 8 Tab. 2007.
ISBN 978-3-87525-268-2.

Band 190: Joachim Hecht
Werkstoffcharakterisierung und Prozessauslegung für die wirkmedienbasierte Doppelblech-Umformung von Magnesiumlegierungen
LFT, 107 Seiten, 91 Bilder, 2 Tab. 2007.
ISBN 978-3-87525-270-5.

Band 191: Ralf Völkl
Stochastische Simulation zur Werkzeuglebensdaueroptimierung und Präzisionsfertigung in der Kaltmassivumformung
LFT, 178 Seiten, 75 Bilder, 12 Tab. 2008.
ISBN 978-3-87525-272-9.

Band 192: Massimo Tolazzi
Innenhochdruck-Umformen verstärkter Blech-Rahmenstrukturen
LFT, 164 Seiten, 85 Bilder, 7 Tab. 2008.
ISBN 978-3-87525-273-6.

Band 193: Cornelia Hoff
Untersuchung der Prozesseinflussgrößen beim Presshärten des höchstfesten Vergütungsstahls 22MnB5
LFT, 133 Seiten, 92 Bilder, 5 Tab. 2008.
ISBN 978-3-87525-275-0.

Band 194: Christian Alvarez
Simulationsgestützte Methoden zur effizienten Gestaltung von Lötprozessen in der Elektronikproduktion
FAPS, 149 Seiten, 86 Bilder, 8 Tab. 2008.
ISBN 978-3-87525-277-4.

Band 195: Andreas Kunze
Automatisierte Montage von makromechatronischen Modulen zur flexiblen Integration in hybride Pkw-Bordnetzsysteme
FAPS, 160 Seiten, 90 Bilder, 14 Tab. 2008.
ISBN 978-3-87525-278-1.

Band 196: Wolfgang Hußnätter
Grundlegende Untersuchungen zur experimentellen Ermittlung und zur Modellierung von Fließortkurven bei erhöhten Temperaturen
LFT, 152 Seiten, 73 Bilder, 21 Tab. 2008.
ISBN 978-3-87525-279-8.

Band 197: Thomas Bigl
Entwicklung, angepasste Herstellungsverfahren und erweiterte Qualitätssicherung von einsatzgerechten elektronischen Baugruppen
FAPS, 175 Seiten, 107 Bilder, 14 Tab. 2008.
ISBN 978-3-87525-280-4.

Band 198: Stephan Roth
Grundlegende Untersuchungen zum Excimerlaserstrahl-Abtragen unter Flüssigkeitsfilmen
LFT, 113 Seiten, 47 Bilder, 14 Tab. 2008.
ISBN 978-3-87525-281-1.

Band 199: Artur Giera
Prozesstechnische Untersuchungen zum Rührreibschweißen metallischer Werkstoffe
LFT, 179 Seiten, 104 Bilder, 36 Tab. 2008.
ISBN 978-3-87525-282-8.

Band 200: Jürgen Lechler
Beschreibung und Modellierung des Werkstoffverhaltens von presshärtbaren Bor-Manganstählen
LFT, 154 Seiten, 75 Bilder, 12 Tab. 2009.
ISBN 978-3-87525-286-6.

Band 201: Andreas Blankl
Untersuchungen zur Erhöhung der Prozessrobustheit bei der Innenhochdruck-Umformung von flächigen Halbzeugen mit vor- bzw. nachgeschalteten Laserstrahlfügeoperationen
LFT, 120 Seiten, 68 Bilder, 9 Tab. 2009.
ISBN 978-3-87525-287-3.

Band 202: Andreas Schaller
Modellierung eines nachfrageorientierten Produktionskonzeptes für mobile Telekommunikationsgeräte
FAPS, 120 Seiten, 79 Bilder, 0 Tab. 2009.
ISBN 978-3-87525-289-7.

Band 203: Claudius Schimpf
Optimierung von Zuverlässigkeitsuntersuchungen, Prüfabläufen und Nacharbeitsprozessen in der Elektronikproduktion
FAPS, 162 Seiten, 90 Bilder, 14 Tab. 2009.
ISBN 978-3-87525-290-3.

Band 204: Simon Dietrich
Sensoriken zur Schwerpunktslagebestimmung der optischen Prozessemissionen beim Laserstrahltiefschweißen
LFT, 138 Seiten, 70 Bilder, 5 Tab. 2009.
ISBN 978-3-87525-292-7.

Band 205: Wolfgang Wolf
Entwicklung eines agentenbasierten Steuerungssystems zur Materialflussorganisation im wandelbaren Produktionsumfeld
FAPS, 167 Seiten, 98 Bilder. 2009.
ISBN 978-3-87525-293-4.

Band 206: Steffen Polster
Laserdurchstrahlschweißen transparenter Polymerbauteile
LFT, 160 Seiten, 92 Bilder, 13 Tab. 2009.
ISBN 978-3-87525-294-1.

Band 207: Stephan Manuel Dörfler
Rührreibschweißen von walzplattiertem Halbzeug und Aluminiumblech zur Herstellung flächiger Aluminiumschaum-Sandwich-Verbundstrukturen
LFT, 190 Seiten, 98 Bilder, 5 Tab. 2009.
ISBN 978-3-87525-295-8.

Band 208: Uwe Vogt
Seriennahe Auslegung von Aluminium Tailored Heat Treated Blanks
LFT, 151 Seiten, 68 Bilder, 26 Tab. 2009.
ISBN 978-3-87525-296-5.

Band 209: Till Laumann
Qualitative und quantitative Bewertung der Crashtauglichkeit von höchstfesten Stählen
LFT, 117 Seiten, 69 Bilder, 7 Tab. 2009.
ISBN 978-3-87525-299-6.

Band 210: Alexander Diehl
Größeneffekte bei Biegeprozessen- Entwicklung einer Methodik zur Identifikation und Quantifizierung
LFT, 180 Seiten, 92 Bilder, 12 Tab. 2010.
ISBN 978-3-87525-302-3.

Band 211: Detlev Staud
Effiziente Prozesskettenauslegung für das Umformen lokal wärmebehandelter und geschweißter Aluminiumbleche
LFT, 164 Seiten, 72 Bilder, 12 Tab. 2010.
ISBN 978-3-87525-303-0.

Band 212: Jens Ackermann
Prozesssicherung beim Laserdurchstrahlschweißen thermoplastischer Kunststoffe
LPT, 129 Seiten, 74 Bilder, 13 Tab. 2010.
ISBN 978-3-87525-305-4.

Band 213: Stephan Weidel
Grundlegende Untersuchungen zum Kontaktzustand zwischen Werkstück und Werkzeug bei umformtechnischen Prozessen unter tribologischen Gesichtspunkten
LFT, 144 Seiten, 67 Bilder, 11 Tab. 2010.
ISBN 978-3-87525-307-8.

Band 214: Stefan Geißdörfer
Entwicklung eines mesoskopischen Modells zur Abbildung von Größeneffekten in der Kaltmassivumformung mit Methoden der FE-Simulation
LFT, 133 Seiten, 83 Bilder, 11 Tab. 2010.
ISBN 978-3-87525-308-5.

Band 215: Christian Matzner
Konzeption produktspezifischer Lösungen zur Robustheitssteigerung elektronischer Systeme gegen die Einwirkung von Betauung im Automobil
FAPS, 165 Seiten, 93 Bilder, 14 Tab. 2010.
ISBN 978-3-87525-309-2.

Band 216: Florian Schüßler
Verbindungs- und Systemtechnik für thermisch hochbeanspruchte und miniaturisierte elektronische Baugruppen
FAPS, 184 Seiten, 93 Bilder, 18 Tab. 2010.
ISBN 978-3-87525-310-8.

Band 217: Massimo Cojutti
Strategien zur Erweiterung der Prozessgrenzen bei der Innhochdruck-Umformung von Rohren und Blechpaaren
LFT, 125 Seiten, 56 Bilder, 9 Tab. 2010.
ISBN 978-3-87525-312-2.

Band 218: Raoul Plettke
Mehrkriterielle Optimierung komplexer Aktorsysteme für das Laserstrahljustieren
LFT, 152 Seiten, 25 Bilder, 3 Tab. 2010.
ISBN 978-3-87525-315-3.

Band 219: Andreas Dobroschke
Flexible Automatisierungslösungen für die Fertigung wickeltechnischer Produkte
FAPS, 184 Seiten, 109 Bilder, 18 Tab. 2011.
ISBN 978-3-87525-317-7.

Band 220: Azhar Zam
Optical Tissue Differentiation for Sensor-Controlled Tissue-Specific Laser Surgery
LPT, 99 Seiten, 45 Bilder, 8 Tab. 2011.
ISBN 978-3-87525-318-4.

Band 221: Michael Rösch
Potenziale und Strategien zur Optimierung des Schablonendruckprozesses in der Elektronikproduktion
FAPS, 192 Seiten, 127 Bilder, 19 Tab. 2011.
ISBN 978-3-87525-319-1.

Band 222: Thomas Rechtenwald
Quasi-isothermes Laserstrahlsintern von Hochtemperatur-Thermoplasten - Eine Betrachtung werkstoff-prozessspezifischer Aspekte am Beispiel PEEK
LPT, 150 Seiten, 62 Bilder, 8 Tab. 2011.
ISBN 978-3-87525-320-7.

Band 223: Daniel Craiovan
Prozesse und Systemlösungen für die SMT-Montage optischer Bauelemente auf Substrate mit integrierten Lichtwellenleitern
FAPS, 165 Seiten, 85 Bilder, 8 Tab. 2011.
ISBN 978-3-87525-324-5.

Band 224: Kay Wagner
Beanspruchungsangepasste Kaltmassivumformwerkzeuge durch lokal optimierte Werkzeugoberflächen
LFT, 147 Seiten, 103 Bilder, 17 Tab. 2011.
ISBN 978-3-87525-325-2.

Band 225: Martin Brandhuber
Verbesserung der Prognosegüte des Versagens von Punktschweißverbindungen bei höchstfesten Stahlgüten
LFT, 155 Seiten, 91 Bilder, 19 Tab. 2011.
ISBN 978-3-87525-327-6.

Band 226: Peter Sebastian Feuser
Ein Ansatz zur Herstellung von pressgehärteten Karosseriekomponenten mit maßgeschneiderten mechanischen Eigenschaften: Temperierte Umformwerkzeuge. Prozessfenster, Prozesssimulation und funktionale Untersuchung
LFT, 195 Seiten, 97 Bilder, 60 Tab. 2012.
ISBN 978-3-87525-328-3.

Band 227: Murat Arbak
Material Adapted Design of Cold Forging Tools Exemplified by Powder Metallurgical Tool Steels and Ceramics
LFT, 109 Seiten, 56 Bilder, 8 Tab. 2012.
ISBN 978-3-87525-330-6.

Band 228: Indra Pitz
Beschleunigte Simulation des Laserstrahlumformens von Aluminiumblechen
LPT, 137 Seiten, 45 Bilder, 27 Tab. 2012.
ISBN 978-3-87525-333-7.

Band 229: Alexander Grimm
Prozessanalyse und -überwachung des Laserstrahlhartlötens mittels optischer Sensorik
LPT, 125 Seiten, 61 Bilder, 5 Tab. 2012.
ISBN 978-3-87525-334-4.

Band 230: Markus Kaupper
Biegen von höhenfesten Stahlblechwerkstoffen - Umformverhalten und Grenzen der Biegbarkeit
LFT, 160 Seiten, 57 Bilder, 10 Tab. 2012.
ISBN 978-3-87525-339-9.

Band 231: Thomas Kroiß
Modellbasierte Prozessauslegung für die Kaltmassivumformung unter Brücksichtigung der Werkzeug- und Pressenauffederung
LFT, 169 Seiten, 50 Bilder, 19 Tab. 2012.
ISBN 978-3-87525-341-2.

Band 232: Christian Goth
Analyse und Optimierung der Entwicklung und Zuverlässigkeit räumlicher Schaltungsträger (3D-MID)
FAPS, 176 Seiten, 102 Bilder, 22 Tab. 2012.
ISBN 978-3-87525-340-5.

Band 233: Christian Ziegler
Ganzheitliche Automatisierung mechatronischer Systeme in der Medizin am Beispiel Strahlentherapie
FAPS, 170 Seiten, 71 Bilder, 19 Tab. 2012.
ISBN 978-3-87525-342-9.

Band 234: Florian Albert
Automatisiertes Laserstrahllöten und -reparaturlöten elektronischer Baugruppen
LPT, 127 Seiten, 78 Bilder, 11 Tab. 2012.
ISBN 978-3-87525-344-3.

Band 235: Thomas Stöhr
Analyse und Beschreibung des mechanischen Werkstoffverhaltens von presshärtbaren Bor-Manganstählen
LFT, 118 Seiten, 74 Bilder, 18 Tab. 2013.
ISBN 978-3-87525-346-7.

Band 236: Christian Kägeler
Prozessdynamik beim Laserstrahlschweißen verzinkter Stahlbleche im Überlappstoß
LPT, 145 Seiten, 80 Bilder, 3 Tab. 2013.
ISBN 978-3-87525-347-4.

Band 237: Andreas Sulzberger
Seriennahe Auslegung der Prozesskette zur wärmeunterstützten Umformung von Aluminiumblechwerkstoffen
LFT, 153 Seiten, 87 Bilder, 17 Tab. 2013.
ISBN 978-3-87525-349-8.

Band 238: Simon Opel
Herstellung prozessangepasster Halbzeuge mit variabler Blechdicke durch die Anwendung von Verfahren der Blechmassivumformung
LFT, 165 Seiten, 108 Bilder, 27 Tab. 2013.
ISBN 978-3-87525-350-4.

Band 239: Rajesh Kanawade
In-vivo Monitoring of Epithelium Vessel and Capillary Density for the Application of Detection of Clinical Shock and Early Signs of Cancer Development
LPT, 124 Seiten, 58 Bilder, 15 Tab. 2013.
ISBN 978-3-87525-351-1.

Band 240: Stephan Busse
Entwicklung und Qualifizierung eines Schneidclinchverfahrens
LFT, 119 Seiten, 86 Bilder, 20 Tab. 2013.
ISBN 978-3-87525-352-8.

Band 241: Karl-Heinz Leitz
Mikro- und Nanostrukturierung mit kurz und ultrakurz gepulster Laserstrahlung
LPT, 154 Seiten, 71 Bilder, 9 Tab. 2013.
ISBN 978-3-87525-355-9.

Band 242: Markus Michl
Webbasierte Ansätze zur ganzheitlichen technischen Diagnose
FAPS, 182 Seiten, 62 Bilder, 20 Tab. 2013.
ISBN 978-3-87525-356-6.

Band 243: Vera Sturm
Einfluss von Chargenschwankungen auf die Verarbeitungsgrenzen von Stahlwerkstoffen
LFT, 113 Seiten, 58 Bilder, 9 Tab. 2013.
ISBN 978-3-87525-357-3.

Band 244: Christian Neudel
Mikrostrukturelle und mechanisch-technologische Eigenschaften widerstandspunktgeschweißter Aluminium-Stahl-Verbindungen für den Fahrzeugbau
LFT, 178 Seiten, 171 Bilder, 31 Tab. 2014.
ISBN 978-3-87525-358-0.

Band 245: Anja Neumann
Konzept zur Beherrschung der Prozessschwankungen im Presswerk
LFT, 162 Seiten, 68 Bilder, 15 Tab. 2014.
ISBN 978-3-87525-360-3.

Band 246: Ulf-Hermann Quentin
Laserbasierte Nanostrukturierung mit optisch positionierten Mikrolinsen
LPT, 137 Seiten, 89 Bilder, 6 Tab. 2014.
ISBN 978-3-87525-361-0.

Band 247: Erik Lamprecht
Der Einfluss der Fertigungsverfahren auf die Wirbelstromverluste von Stator-Einzelzahnblechpaketen für den Einsatz in Hybrid- und Elektrofahrzeugen
FAPS, 148 Seiten, 138 Bilder, 4 Tab. 2014.
ISBN 978-3-87525-362-7.

Band 248: Sebastian Rösel
Wirkmedienbasierte Umformung von Blechhalbzeugen unter Anwendung magnetorheologischer Flüssigkeiten als kombiniertes Wirk- und Dichtmedium
LFT, 148 Seiten, 61 Bilder, 12 Tab. 2014.
ISBN 978-3-87525-363-4.

Band 249: Paul Hippchen
Simulative Prognose der Geometrie indirekt pressgehärteter Karosseriebauteile für die industrielle Anwendung
LFT, 163 Seiten, 89 Bilder, 12 Tab. 2014.
ISBN 978-3-87525-364-1.

Band 250: Martin Zubeil
Versagensprognose bei der Prozesssimulation von Biegeumform- und Falzverfahren
LFT, 171 Seiten, 90 Bilder, 5 Tab. 2014.
ISBN 978-3-87525-365-8.

Band 251: Alexander Kühl
Flexible Automatisierung der Statorenmontage mit Hilfe einer universellen ambidexteren Kinematik
FAPS, 142 Seiten, 60 Bilder, 26 Tab. 2014.
ISBN 978-3-87525-367-2.

Band 252: Thomas Albrecht
Optimierte Fertigungstechnologien für Rotoren getriebeintegrierter PM-Synchronmotoren von Hybridfahrzeugen
FAPS, 198 Seiten, 130 Bilder, 38 Tab. 2014.
ISBN 978-3-87525-368-9.

Band 253: Florian Risch
Planning and Production Concepts for Contactless Power Transfer Systems for Electric Vehicles
FAPS, 185 Seiten, 125 Bilder, 13 Tab. 2014.
ISBN 978-3-87525-369-6.

Band 254: Markus Weigl
Laserstrahlschweißen von Mischverbindungen aus austenitischen und ferritischen korrosionsbeständigen Stahlwerkstoffen
LPT, 184 Seiten, 110 Bilder, 6 Tab. 2014.
ISBN 978-3-87525-370-2.

Band 255: Johannes Noneder
Beanspruchungserfassung für die Validierung von FE-Modellen zur Auslegung von Massivumformwerkzeugen
LFT, 161 Seiten, 65 Bilder, 14 Tab. 2014.
ISBN 978-3-87525-371-9.

Band 256: Andreas Reinhardt
Ressourceneffiziente Prozess- und Produktionstechnologie für flexible Schaltungsträger
FAPS, 123 Seiten, 69 Bilder, 19 Tab. 2014.
ISBN 978-3-87525-373-3.

Band 257: Tobias Schmuck
Ein Beitrag zur effizienten Gestaltung globaler Produktions- und Logistiknetzwerke mittels Simulation
FAPS, 151 Seiten, 74 Bilder. 2014.
ISBN 978-3-87525-374-0.

Band 258: Bernd Eichenhüller
Untersuchungen der Effekte und Wechselwirkungen charakteristischer Einflussgrößen auf das Umformverhalten bei Mikroumformprozessen
LFT, 127 Seiten, 29 Bilder, 9 Tab. 2014.
ISBN 978-3-87525-375-7.

Band 259: Felix Lütteke
Vielseitiges autonomes Transportsystem basierend auf Weltmodellerstellung mittels Datenfusion von Deckenkameras und Fahrzeugsensoren
FAPS, 152 Seiten, 54 Bilder, 20 Tab. 2014.
ISBN 978-3-87525-376-4.

Band 260: Martin Grüner
Hochdruck-Blechumformung mit formlos festen Stoffen als Wirkmedium
LFT, 144 Seiten, 66 Bilder, 29 Tab. 2014.
ISBN 978-3-87525-379-5.

Band 261: Christian Brock
Analyse und Regelung des Laserstrahltiefschweißprozesses durch Detektion der Metalldampffackelposition
LPT, 126 Seiten, 65 Bilder, 3 Tab. 2015.
ISBN 978-3-87525-380-1.

Band 262: Peter Vatter
Sensitivitätsanalyse des 3-Rollen-Schubbiegens auf Basis der Finite Elemente Methode
LFT, 145 Seiten, 57 Bilder, 26 Tab. 2015.
ISBN 978-3-87525-381-8.

Band 263: Florian Klämpfl
Planung von Laserbestrahlungen durch simulationsbasierte Optimierung
LPT, 169 Seiten, 78 Bilder, 32 Tab. 2015.
ISBN 978-3-87525-384-9.

Band 264: Matthias Domke
Transiente physikalische Mechanismen bei der Laserablation von dünnen Metallschichten
LPT, 133 Seiten, 43 Bilder, 3 Tab. 2015.
ISBN 978-3-87525-385-6.

Band 265: Johannes Götz
Community-basierte Optimierung des Anlagenengineerings
FAPS, 177 Seiten, 80 Bilder, 30 Tab. 2015.
ISBN 978-3-87525-386-3.

Band 266: Hung Nguyen
Qualifizierung des Potentials von Verfestigungseffekten zur Erweiterung des Umformvermögens aushärtbarer Aluminiumlegierungen
LFT, 137 Seiten, 57 Bilder, 16 Tab. 2015.
ISBN 978-3-87525-387-0.

Band 267: Andreas Kuppert
Erweiterung und Verbesserung von Versuchs- und Auswertetechniken für die Bestimmung von Grenzformänderungskurven
LFT, 138 Seiten, 82 Bilder, 2 Tab. 2015.
ISBN 978-3-87525-388-7.

Band 268: Kathleen Klaus
Erstellung eines Werkstofforientierten Fertigungsprozessfensters zur Steigerung des Formgebungsvermögens von Aluminiumlegierungen unter Anwendung einer zwischengeschalteten Wärmebehandlung
LFT, 154 Seiten, 70 Bilder, 8 Tab. 2015.
ISBN 978-3-87525-391-7.

Band 269: Thomas Svec
Untersuchungen zur Herstellung von funktionsoptimierten Bauteilen im partiellen Presshärtprozess mittels lokal unterschiedlich temperierter Werkzeuge
LFT, 166 Seiten, 87 Bilder, 15 Tab. 2015.
ISBN 978-3-87525-392-4.

Band 270: Tobias Schrader
Grundlegende Untersuchungen zur Verschleißcharakterisierung beschichteter Kaltmassivumformwerkzeuge
LFT, 164 Seiten, 55 Bilder, 11 Tab. 2015.
ISBN 978-3-87525-393-1.

Band 271: Matthäus Brela
Untersuchung von Magnetfeld-Messmethoden zur ganzheitlichen Wertschöpfungsoptimierung und Fehlerdetektion an magnetischen Aktoren
FAPS, 170 Seiten, 97 Bilder, 4 Tab. 2015.
ISBN 978-3-87525-394-8.

Band 272: Michael Wieland
Entwicklung einer Methode zur Prognose adhäsiven Verschleißes an Werkzeugen für das direkte Presshärten
LFT, 156 Seiten, 84 Bilder, 9 Tab. 2015.
ISBN 978-3-87525-395-5.

Band 273: René Schramm
Strukturierte additive Metallisierung durch kaltaktives Atmosphärendruckplasma
FAPS, 136 Seiten, 62 Bilder, 15 Tab. 2015.
ISBN 978-3-87525-396-2.

Band 274: Michael Lechner
Herstellung beanspruchungsangepasster Aluminiumblechhalbzeuge durch eine maßgeschneiderte Variation der Abkühlgeschwindigkeit nach Lösungsglühen
LFT, 136 Seiten, 62 Bilder, 15 Tab. 2015.
ISBN 978-3-87525-397-9.

Band 275: Kolja Andreas
Einfluss der Oberflächenbeschaffenheit auf das Werkzeugeinsatzverhalten beim Kaltfließpressen
LFT, 169 Seiten, 76 Bilder, 4 Tab. 2015.
ISBN 978-3-87525-398-6.

Band 276: Marcus Baum
Laser Consolidation of ITO Nanoparticles for the Generation of Thin Conductive Layers on Transparent Substrates
LPT, 158 Seiten, 75 Bilder, 3 Tab. 2015.
ISBN 978-3-87525-399-3.

Band 277: Thomas Schneider
Umformtechnische Herstellung dünnwandiger Funktionsbauteile aus Feinblech durch Verfahren der Blechmassivumformung
LFT, 188 Seiten, 95 Bilder, 7 Tab. 2015.
ISBN 978-3-87525-401-3.

Band 278: Jochen Merhof
Sematische Modellierung automatisierter Produktionssysteme zur Verbesserung der IT-Integration zwischen Anlagen-Engineering und Steuerungsebene
FAPS, 157 Seiten, 88 Bilder, 8 Tab. 2015.
ISBN 978-3-87525-402-0.

Band 279: Fabian Zöller
Erarbeitung von Grundlagen zur Abbildung des tribologischen Systems in der Umformsimulation
LFT, 126 Seiten, 51 Bilder, 3 Tab. 2016.
ISBN 978-3-87525-403-7.

Band 280: Christian Hezler
Einsatz technologischer Versuche zur Erweiterung der Versagensvorhersage bei Karosseriebauteilen aus höchstfesten Stählen
LFT, 147 Seiten, 63 Bilder, 44 Tab. 2016.
ISBN 978-3-87525-404-4.

Band 281: Jochen Bönig
Integration des Systemverhaltens von Automobil-Hochvoltleitungen in die virtuelle Absicherung durch strukturmechanische Simulation
FAPS, 177 Seiten, 107 Bilder, 17 Tab. 2016.
ISBN 978-3-87525-405-1.

Band 282: Johannes Kohl
Automatisierte Datenerfassung für diskret ereignisorientierte Simulationen in der energieflexibelen Fabrik
FAPS, 160 Seiten, 80 Bilder, 27 Tab. 2016.
ISBN 978-3-87525-406-8.

Band 283: Peter Bechtold
Mikroschockwellenumformung mittels ultrakurzer Laserpulse
LPT, 155 Seiten, 59 Bilder, 10 Tab. 2016.
ISBN 978-3-87525-407-5.

Band 284: Stefan Berger
Laserstrahlschweißen thermoplastischer Kohlenstofffaserverbundwerkstoffe mit spezifischem Zusatzdraht
LPT, 118 Seiten, 68 Bilder, 9 Tab. 2016.
ISBN 978-3-87525-408-2.

Band 285: Martin Bornschlegl
Methods-Energy Measurement - Eine Methode zur Energieplanung für Fügeverfahren im Karosseriebau
FAPS, 136 Seiten, 72 Bilder, 46 Tab. 2016.
ISBN 978-3-87525-409-9.

Band 286: Tobias Rackow
Erweiterung des Unternehmenscontrollings um die Dimension Energie
FAPS, 164 Seiten, 82 Bilder, 29 Tab. 2016.
ISBN 978-3-87525-410-5.

Band 287: Johannes Koch
Grundlegende Untersuchungen zur Herstellung zyklisch-symmetrischer Bauteile mit Nebenformelementen durch Blechmassivumformung
LFT, 125 Seiten, 49 Bilder, 17 Tab. 2016.
ISBN 978-3-87525-411-2.

Band 288: Hans Ulrich Vierzigmann
Beitrag zur Untersuchung der tribologischen Bedingungen in der Blechmassivumformung - Bereitstellung von tribologischen Modellversuchen und Realisierung von Tailored Surfaces
LFT, 174 Seiten, 102 Bilder, 34 Tab. 2016.
ISBN 978-3-87525-412-9.

Band 289: Thomas Senner
Methodik zur virtuellen Absicherung der formgebenden Operation des Nasspressprozesses von Gelege-Mehrschichtverbunden
LFT, 156 Seiten, 96 Bilder, 21 Tab. 2016.
ISBN 978-3-87525-414-3.

Band 290: Sven Kreitlein
Der grundoperationsspezifische Mindestenergiebedarf als Referenzwert zur Bewertung der Energieeffizienz in der Produktion
FAPS, 185 Seiten, 64 Bilder, 30 Tab. 2016.
ISBN 978-3-87525-415-0.

Band 291: Christian Roos
Remote-Laserstrahlschweißen verzinkter Stahlbleche in Kehlnahtgeometrie
LPT, 123 Seiten, 52 Bilder, 0 Tab. 2016.
ISBN 978-3-87525-416-7.

Band 292: Alexander Kahrimanidis
Thermisch unterstützte Umformung von Aluminiumblechen
LFT, 165 Seiten, 103 Bilder, 18 Tab. 2016.
ISBN 978-3-87525-417-4.

Band 293: Jan Tremel
Flexible Systems for Permanent Magnet Assembly and Magnetic Rotor Measurement / Flexible Systeme zur Montage von Permanentmagneten und zur Messung magnetischer Rotoren
FAPS, 152 Seiten, 91 Bilder, 12 Tab. 2016.
ISBN 978-3-87525-419-8.

Band 294: Ioannis Tsoupis
Schädigungs- und Versagensverhalten hochfester Leichtbauwerkstoffe unter Biegebeanspruchung
LFT, 176 Seiten, 51 Bilder, 6 Tab. 2017.
ISBN 978-3-87525-420-4.

Band 295: Sven Hildering
Grundlegende Untersuchungen zum Prozessverhalten von Silizium als Werkzeugwerkstoff für das Mikroscherschneiden metallischer Folien
LFT, 177 Seiten, 74 Bilder, 17 Tab. 2017.
ISBN 978-3-87525-422-8.

Band 296: Sasia Mareike Hertweck
Zeitliche Pulsformung in der Lasermikromaterialbearbeitung – Grundlegende Untersuchungen und Anwendungen
LPT, 146 Seiten, 67 Bilder, 5 Tab. 2017.
ISBN 978-3-87525-423-5.

Band 297: Paryanto
Mechatronic Simulation Approach for the Process Planning of Energy-Efficient Handling Systems
FAPS, 162 Seiten, 86 Bilder, 13 Tab. 2017.
ISBN 978-3-87525-424-2.

Band 298: Peer Stenzel
Großserientaugliche Nadelwickeltechnik für verteilte Wicklungen im Anwendungsfall der E-Traktionsantriebe
FAPS, 239 Seiten, 147 Bilder, 20 Tab. 2017.
ISBN 978-3-87525-425-9.

Band 299: Mario Lušić
Ein Vorgehensmodell zur Erstellung montageführender Werkerinformationssysteme simultan zum Produktentstehungsprozess
FAPS, 174 Seiten, 79 Bilder, 22 Tab. 2017.
ISBN 978-3-87525-426-6.

Band 300: Arnd Buschhaus
Hochpräzise adaptive Steuerung und Regelung robotergeführter Prozesse
FAPS, 202 Seiten, 96 Bilder, 4 Tab. 2017.
ISBN 978-3-87525-427-3.

Band 301: Tobias Laumer
Erzeugung von thermoplastischen Werkstoffverbunden mittels simultanem, intensitätsselektivem Laserstrahlschmelzen
LPT, 140 Seiten, 82 Bilder, 0 Tab. 2017.
ISBN 978-3-87525-428-0.

Band 302: Nora Unger
Untersuchung einer thermisch unterstützten Fertigungskette zur Herstellung umgeformter Bauteile aus der höherfesten Aluminiumlegierung EN AW-7020
LFT, 142 Seiten, 53 Bilder, 8 Tab. 2017.
ISBN 978-3-87525-429-7.

Band 303: Tommaso Stellin
Design of Manufacturing Processes for the Cold Bulk Forming of Small Metal Components from Metal Strip
LFT, 146 Seiten, 67 Bilder, 7 Tab. 2017.
ISBN 978-3-87525-430-3.

Band 304: Bassim Bachy
Experimental Investigation, Modeling, Simulation and Optimization of Molded Interconnect Devices (MID) Based on Laser Direct Structuring (LDS) / Experimentelle Untersuchung, Modellierung, Simulation und Optimierung von Molded Interconnect Devices (MID) basierend auf Laser Direktstrukturierung (LDS)
FAPS, 168 Seiten, 120 Bilder, 26 Tab. 2017.
ISBN 978-3-87525-431-0.

Band 305: Michael Spahr
Automatisierte Kontaktierungsverfahren für flachleiterbasierte Pkw-Bordnetzsysteme
FAPS, 197 Seiten, 98 Bilder, 17 Tab. 2017.
ISBN 978-3-87525-432-7.

Band 306: Sebastian Suttner
Charakterisierung und Modellierung des spannungszustandsabhängigen Werkstoffverhaltens der Magnesiumlegierung AZ31B für die numerische Prozessauslegung
LFT, 150 Seiten, 84 Bilder, 19 Tab. 2017.
ISBN 978-3-87525-433-4.

Band 307: Bhargav Potdar
A reliable methodology to deduce thermo-mechanical flow behaviour of hot stamping steels
LFT, 203 Seiten, 98 Bilder, 27 Tab. 2017.
ISBN 978-3-87525-436-5.

Band 308: Maria Löffler
Steuerung von Blechmassivumformprozessen durch maßgeschneiderte tribologische Systeme
LFT, viii u. 166 Seiten, 90 Bilder, 5 Tab. 2018. ISBN 978-3-96147-133-1.

Band 309: Martin Müller
Untersuchung des kombinierten Trenn- und Umformprozesses beim Fügen artungleicher Werkstoffe mittels Schneidclinchverfahren
LFT, xi u. 149 Seiten, 89 Bilder, 6 Tab. 2018. ISBN: 978-3-96147-135-5.

Band 310: Christopher Kästle
Qualifizierung der Kupfer-Drahtbondtechnologie für integrierte Leistungsmodule in harschen Umgebungsbedingungen
FAPS, xii u. 167 Seiten, 70 Bilder, 18 Tab. 2018. ISBN 978-3-96147-145-4.

Band 311: Daniel Vipavc
Eine Simulationsmethode für das 3-Rollen-Schubbiegen
LFT, xiii u. 121 Seiten, 56 Bilder, 17 Tab. 2018. ISBN 978-3-96147-147-8.

Band 312: Christina Ramer
Arbeitsraumüberwachung und autonome Bahnplanung für ein sicheres und flexibles Roboter-Assistenzsystem in der Fertigung
FAPS, xiv u. 188 Seiten, 57 Bilder, 9 Tab. 2018. ISBN 978-3-96147-153-9.

Band 313: Miriam Rauer
Der Einfluss von Poren auf die Zuverlässigkeit der Lötverbindungen von Hochleistungs-Leuchtdioden
FAPS, xii u. 209 Seiten, 108 Bilder, 21 Tab. 2018. ISBN 978-3-96147-157-7.

Band 314: Felix Tenner

Kamerabasierte Untersuchungen der Schmelze und Gasströmungen beim Laserstrahlschweißen verzinkter Stahlbleche

LPT, xxiii u. 184 Seiten, 94 Bilder, 7 Tab.
2018. ISBN 978-3-96147-160-7.

Band 315: Aarief Syed-Khaja

Diffusion Soldering for High-temperature Packaging of Power Electronics

FAPS, x u. 202 Seiten, 144 Bilder, 32 Tab.
2018. ISBN 978-3-87525-162-1.

Band 316: Adam Schaub

Grundlagenwissenschaftliche Untersuchung der kombinierten Prozesskette aus Umformen und Additive Fertigung

LFT, xi u. 192 Seiten, 72 Bilder, 27 Tab.
2019. ISBN 978-3-96147-166-9.

Band 317: Daniel Gröbel

Herstellung von Nebenformelementen unterschiedlicher Geometrie an Blechen mittels Fließpressverfahren der Blechmassivumformung

LFT, x u. 165 Seiten, 96 Bilder, 13 Tab.
2019. ISBN 978-3-96147-168-3.

Band 318: Philipp Hildenbrand

Entwicklung einer Methodik zur Herstellung von Tailored Blanks mit definierten Halbzeugeigenschaften durch einen Taumelprozess

LFT, ix u. 153 Seiten, 77 Bilder, 4 Tab.
2019. ISBN 978-3-96147-174-4.

Band 319: Tobias Konrad

Simulative Auslegung der Spann- und Fixierkonzepte im Karosserierohbau: Bewertung der Baugruppenmaßhaltigkeit unter Berücksichtigung schwankender Einflussgrößen

LFT, x u. 203 Seiten, 134 Bilder, 32 Tab.
2019. ISBN 978-3-96147-176-8.

Band 320: David Meinel

Architektur applikationsspezifischer Multi-Physics-Simulationskonfiguratoren am Beispiel modularer Triebzüge

FAPS, xii u. 166 Seiten, 82 Bilder, 25 Tab.
2019. ISBN 978-3-96147-184-3.

Band 321: Andrea Zimmermann

Grundlegende Untersuchungen zum Einfluss fertigungsbedingter Eigenschaften auf die Ermüdungsfestigkeit kaltmassivumgeformter Bauteile

LFT, ix u. 160 Seiten, 66 Bilder, 5 Tab.
2019. ISBN 978-3-96147-190-4.

Band 322: Christoph Amann

Simulative Prognose der Geometrie nassgepresster Karosseriebauteile aus Gelege-Mehrschichtverbunden

LFT, xvi u. 169 Seiten, 80 Bilder, 13 Tab.
2019. ISBN 978-3-96147-194-2.

Band 323: Jennifer Tenner

Realisierung schmierstofffreier Tiefziehprozesse durch maßgeschneiderte Werkzeugoberflächen

LFT, x u. 187 Seiten, 68 Bilder, 13 Tab.
2019. ISBN 978-3-96147-196-6.

Band 324: Susan Zöller

Mapping Individual Subjective Values to Product Design

KTmfk, xi u. 223 Seiten, 81 Bilder, 25 Tab.
2019. ISBN 978-3-96147-202-4.

Band 325: Stefan Lutz

Erarbeitung einer Methodik zur semiempirischen Ermittlung der Umwandlungskinetik durchhärtender Wälzlagerstähle für die Wärmebehandlungssimulation

LFT, xiv u. 189 Seiten, 75 Bilder, 32 Tab. 2019. ISBN 978-3-96147-209-3.

Band 326: Tobias Gnibl

Modellbasierte Prozesskettenabbildung rührreibgeschweißter Aluminiumhalbzeuge zur umformtechnischen Herstellung höchstfester Leichtbaustrukturteile

LFT, xii u. 167 Seiten, 68 Bilder, 17 Tab. 2019. ISBN 978-3-96147-217-8.

Band 327: Johannes Bürner

Technisch-wirtschaftliche Optionen zur Lastflexibilisierung durch intelligente elektrische Wärmespeicher

FAPS, xiv u. 233 Seiten, 89 Bilder, 27 Tab. 2019. ISBN 978-3-96147-219-2.

Band 328: Wolfgang Böhm

Verbesserung des Umformverhaltens von mehrlagigen Aluminiumblechwerkstoffen mit ultrafeinkörnigem Gefüge

LFT, ix u. 160 Seiten, 88 Bilder, 14 Tab. 2019. ISBN 978-3-96147-227-7.

Band 329: Stefan Landkammer

Grundsatzuntersuchungen, mathematische Modellierung und Ableitung einer Auslegungsmethodik für Gelenkantriebe nach dem Spinnenbeinprinzip

LFT, xii u. 200 Seiten, 83 Bilder, 13 Tab. 2019. ISBN 978-3-96147-229-1.

Band 330: Stephan Rapp

Pump-Probe-Ellipsometrie zur Messung transienter optischer Materialeigenschaften bei der Ultrakurzpuls-Lasermaterialbearbeitung

LPT, xi u. 143 Seiten, 49 Bilder, 2 Tab. 2019. ISBN 978-3-96147-235-2.

Band 331: Michael Scholz

Intralogistics Execution System mit integrierten autonomen, servicebasierten Transportentitäten

FAPS, xi u. 195 Seiten, 55 Bilder, 11 Tab. 2019. ISBN 978-3-96147-237-6.

Band 332: Eva Bogner

Strategien der Produktindividualisierung in der produzierenden Industrie im Kontext der Digitalisierung

FAPS, ix u. 201 Seiten, 55 Bilder, 28 Tab. 2019. ISBN 978-3-96147-246-8.

Band 333: Daniel Benjamin Krüger

Ein Ansatz zur CAD-integrierten muskuloskelettalen Analyse der Mensch-Maschine-Interaktion

KTmfk, x u. 217 Seiten, 102 Bilder, 7 Tab. 2019. ISBN 978-3-96147-250-5.

Band 334: Thomas Kuhn

Qualität und Zuverlässigkeit laserdirektstrukturierter mechatronisch integrierter Baugruppen (LDS-MID)

FAPS, ix u. 152 Seiten, 69 Bilder, 12 Tab. 2019. ISBN: 978-3-96147-252-9.

Band 335: Hans Fleischmann
Modellbasierte Zustands- und Prozessüberwachung auf Basis sozio-cyber-physischer Systeme
FAPS, xi u. 214 Seiten, 111 Bilder, 18 Tab.
2019. ISBN: 978-3-96147-256-7.

Band 336: Markus Michalski
Grundlegende Untersuchungen zum Prozess- und Werkstoffverhalten bei schwingungsüberlagerter Umformung
LFT, xii u. 197 Seiten, 93 Bilder, 11 Tab.
2019. ISBN: 978-3-96147-270-3.

Band 337: Markus Brandmeier
Ganzheitliches ontologiebasiertes Wissensmanagement im Umfeld der industriellen Produktion
FAPS, xi u. 255 Seiten, 77 Bilder, 33 Tab.
2020. ISBN: 978-3-96147-275-8.

Band 338: Stephan Purr
Datenerfassung für die Anwendung lernender Algorithmen bei der Herstellung von Blechformteilen
LFT, ix u. 165 Seiten, 48 Bilder, 4 Tab.
2020. ISBN: 978-3-96147-281-9.

Band 339: Christoph Kiener
Kaltfließpressen von gerad- und schrägverzahnten Zahnrädern
LFT, viii u. 151 Seiten, 81 Bilder, 3 Tab.
2020. ISBN 978-3-96147-287-1.

Band 340: Simon Spreng
Numerische, analytische und empirische Modellierung des Heißcrimpprozesses
FAPS, xix u. 204 Seiten, 91 Bilder, 27 Tab.
2020. ISBN 978-3-96147-293-2.

Band 341: Patrik Schwingenschlögl
Erarbeitung eines Prozessverständnisses zur Verbesserung der tribologischen Bedingungen beim Presshärten
LFT, x u. 177 Seiten, 81 Bilder, 8 Tab.
2020. ISBN 978-3-96147-297-0.

Band 342: Emanuela Affronti
Evaluation of failure behaviour of sheet metals
LFT, ix u. 136 Seiten, 57 Bilder, 20 Tab.
2020. ISBN 978-3-96147-303-8.

Band 343: Julia Degner
Grundlegende Untersuchungen zur Herstellung hochfester Aluminiumblechbauteile in einem kombinierten Umform- und Abschreckprozess
LFT, x u. 172 Seiten, 61 Bilder, 9 Tab.
2020. ISBN 978-3-96147-307-6.

Band 344: Maximilian Wagner
Automatische Bahnplanung für die Aufteilung von Prozessbewegungen in synchrone Werkstück- und Werkzeugbewegungen mittels Multi-Roboter-Systemen
FAPS, xxi u. 181 Seiten, 111 Bilder, 15 Tab.
2020. ISBN 978-3-96147-309-0.

Abstract

The research focus of this thesis is a concept for the division into synchronously executed movements of workpiece and tool that can be implemented for different proprietary control concepts without intervening in their control level. For this purpose, a universal approach is being developed that can be integrated into offline programming software and automatically generates the programs for the participating robots. The approach also includes the ability to scale and limit the division to suit the specific requirements of different applications. Consequently, a targeted division to the individual robots as well as the consideration of process-related boundary conditions, such as a constant workpiece orientation or a restriction of the Cartesian workspace, can be implemented.

The division approach is implemented as an example for a set-up with two articulated arm robots and a set-up with a two-arm robot and evaluated extensively. General and application-specific investigations are carried out with regard to the influence of the cooperation on the processing accuracy, the process time and on the workpiece-related reachability.

Forschungsschwerpunkt dieser Arbeit ist ein Konzept für die Aufteilung in synchron ausgeführte Bewegungen von Werkstück und Werkzeug, das für unterschiedliche proprietäre Steuerungskonzepte implementierbar ist, ohne in deren Steuerungsebene einzugreifen. Hierfür wird ein universeller Ansatz erarbeitet, der sich in Offline-Programmiersoftware integrieren lässt und die Programme für die beteiligten Roboter automatisiert generiert. Der Ansatz enthält zudem die Möglichkeit, eine zweckmäßige Skalierung und Begrenzungen der Aufteilung vorzunehmen, damit er für die spezifischen Anforderungen unterschiedlicher Anwendungen geeignet ist. Folglich sind eine zielgerichtete Verteilung auf die einzelnen Roboter sowie die Berücksichtigung von prozessbedingten Randbedingungen, wie beispielsweise eine konstante Werkstückorientierung oder eine Einschränkung des kartesischen Arbeitsraumes, umsetzbar.

Der Aufteilungsansatz wird exemplarisch für einen Aufbau mit zwei Knickarmrobotern und einen Aufbau mit einem Zweiarmroboter implementiert und umfassend evaluiert. Dabei erfolgen allgemeine sowie anwendungsspezifische Untersuchungen hinsichtlich des Einflusses der Kooperation auf die Bearbeitungsgenauigkeit, die Prozesszeit sowie auf die werkstückbezogene Erreichbarkeit.

ISBN 978-3-96147-309-0
9 783961 473090

FAU UNIVERSITY PRESS 2020