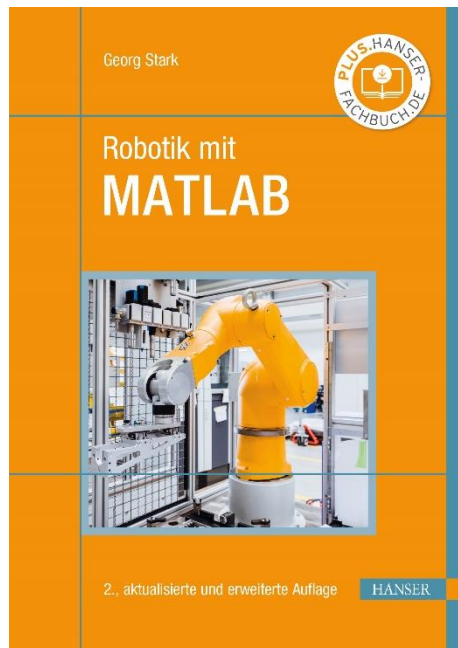


# HANSER



## Leseprobe

zu

## Robotik mit MATLAB

von Georg Stark

Print-ISBN: 978-3-446-47195-5  
E-Book-ISBN: 978-3-446-47362-1

Weitere Informationen und Bestellungen unter  
<https://www.hanser-kundencenter.de/fachbuch/artikel/9783446471955>

sowie im Buchhandel

© Carl Hanser Verlag, München

# Vorwort

## **Ein besonderer Ansatz!**

Den Anstoß für dieses Lehrbuch gaben eigene Erfahrungen bei der Entwicklung einer Robotersteuerung. Mit einer technikorientierten Programmiersprache und umfangreicher Funktionsbibliothek, wie die technische Software MATLAB sie bereitstellt, verläuft der Entwicklungsprozess wesentlich schneller und problemloser als mit allgemeinen Programmiersprachen. Die formalen Modelle können sehr direkt in Software umgesetzt werden. Dies führt zu übersichtlichem und leicht wartbarem Programmcode.

Ein weiterer, wichtiger Ansatz besteht darin, alle benötigten Wissensgebiete für die Entwicklung von Robotersoftware, aufeinander abgestimmt, mit einem einzigen Buch abzudecken. Hierzu gehören die mathematischen Grundlagen, die Programmiersprache, die Verfahren der Robotik und die Roboteranwendung. Aber dieser Ansatz bedeutet auch, dass die behandelten Wissensgebiete in ihrem Umfang beschränkt und auf die zu lösenden Probleme ausgerichtet sein müssen.

## **Für wen?**

Das Buch ist konzipiert für Studierende in den ersten Semestern der Ingenieurwissenschaften und technikorientierten Informatik an Universitäten und Hochschulen. Es kann als Leitfaden für einführende Lehrveranstaltungen zum Thema „Softwareentwicklung für mechanische Systeme, Robotik“ eingesetzt werden. Sowohl für Studierende als auch für Berufstätige ist es zum Selbststudium und zur Weiterbildung geeignet.

## **Mit welchem Ziel?**

Inhaltlich besteht das Ziel darin, die Grundlagen der Robotik mit dem Schwerpunkt Software zu vermitteln. Dabei stehen moderne Methoden wie modellbasierte und komponentenorientierte Softwareentwicklung im Vordergrund. Methodisch wird angestrebt, vertieftes Lernen durch eine enge Verzahnung von Wissensvermittlung und unmittelbarer Umsetzung in Software zu erreichen. Durch diese Verknüpfung wird der Lernende zu einer intensiven Auseinandersetzung mit den In-

halten herausgefordert. Darüber hinaus wird angestrebt, das Wissen auch auf verwandte Problembereiche außerhalb der Robotik zu transferieren, um so eine umfassende Sicht zu gewinnen.

### **Mit welchem Inhalt und Aufbau?**

Als Einstieg dient ein kurzer historischer Rückblick und die zentralen Begriffe *Roboter* und *Robotik* werden definiert. Ausgehend von den Anforderungen werden zunächst die Industrieroboter dargestellt, bevor auf die anderen Roboterarten eingegangen wird. Abschließend werden die gemeinsamen Grundprinzipien von Robotern aufgezeigt (Kapitel 1).

Der Ausgangspunkt für die Programmierung sind formale, mathematische Modelle. Als Voraussetzung für deren Realisierung werden die Grundlagen der Robotermathematik behandelt (Kapitel 2). Die technische Software MATLAB wird ausführlich erklärt. Sie umfasst die Bedienungsumgebung, die Funktionsbibliothek und die darin eingebettete Programmiersprache. Darauf aufbauend zeigt das Buch, wie die benötigte Robotermathematik durch MATLAB-Funktionen realisiert wird. Das Ergebnis ist die ROBOMATS-Funktionsbibliothek, die als Grundlage für die Realisierung der Robotersoftware verwendet wird (Kapitel 3).

Die beiden zentralen Kapitel 4 und 5 behandeln die Verfahren zur Modellierung der kinematischen Struktur und zum Entwurf von Bahnsteuerungen. Schrittweise werden die Softwareteile für einen kompletten Robotersimulator mit Grafikausgabe entwickelt. Die Realisierung von größeren Softwarepaketen erfordert zusätzliche Softwaretechniken, die in einem eigenen Kapitel behandelt werden (Kapitel 6). Im Mittelpunkt steht dabei die komponentenorientierte Programmierung, basierend auf der weitverbreiteten COM-Schnittstelle. Sie ermöglicht die Aufteilung der Gesamtsoftware auf mehrere Komponenten, die auch mit unterschiedlichen Sprachen programmiert sein können. Auf diese Weise können die einzelnen Softwareteile flexibel zu einem Robotersimulator mit Grafikausgabe integriert werden.

Entscheidend für jede Steuerungssoftware ist, dass sie für die beabsichtigten Anwendungen tauglich ist. Aus diesem Grunde wird die nun entwickelte Software für zwei typische Anwendungen getestet (Kapitel 7). Beim Palettieren wird das Ausführungsprogramm automatisch vom Palettiermuster abgeleitet. Die Bearbeitung eines Langlochs benötigt exakte Bahnen, die auch die Ausführung von Halbkreisen erfordern. Zum Abschluss wird aufgezeigt, wie mithilfe der MATLAB-Entwicklungsumgebung Fehler analysiert und Programme optimiert werden können (Kapitel 8).

### **Welche didaktische Unterstützung gibt es dabei?**

Am Anfang eines jeden Kapitels wird die Zielsetzung formuliert und ein kurzer Abriss gegeben, der die wesentlichen Leitgedanken herausstellt. Am Ende erfolgt eine Zusammenfassung und Auflistung der wichtigsten Begriffe und Methoden. Die Ausführungen werden ergänzt durch die Kommentierungen *Wichtig* und *Hinweis*.

Am Ende der Kapitel 2 bis 8 finden Sie Aufgaben. Sie sollen zur Vertiefung des Gelernten dienen und anregen, das Wissen auf andere Themengebiete zu transferieren. Lösungen für die Aufgaben und Zusatzinformationen werden über das Internet bereitgestellt.

### **Wie soll das Buch benutzt werden?**

Kapitel 1 vermittelt Hintergrundwissen und stellt wichtige Begriffe vor, die im weiteren Verlauf verwendet werden. Die Beherrschung der Inhalte von Kapitel 2 und 3 ist Voraussetzung für die Folgenden. Dies sollte durch gewissenhafte Bearbeitung der Aufgaben sichergestellt werden. Die Kapitel 4 und 5 behandeln die wesentlichen Grundlagen für die Entwicklung von Robotersoftware. Die dargelegten mathematischen Verfahren werden unmittelbar in Programme umgesetzt. Die gezeigten Listings haben beispielhaften Charakter. Dem Lernenden wird ausdrücklich empfohlen, zunächst eine eigene Implementierung zu versuchen und diese dann mit der vorgestellten zu vergleichen. Die Konzepte für das Programmieren im Großen in Kapitel 6 sollen erst nach sicherer Beherrschung des vorausgehenden Stoffes bearbeitet werden. Dieses Kapitel kann auch übersprungen werden. Die beiden Beispiele in Kapitel 7 können nur Anregung sein, sich mit weiteren Roboteranwendungen zu befassen. Mit den Methoden von Kapitel 8 soll im Nachhinein die bereits entwickelte Software analysiert und gegebenenfalls optimiert werden.

### **Zu guter Letzt!**

Mein besonderer Dank gilt Frau Dipl.-Ing. Erika Hotho und Frau Mirja Werner, M.A., die mich als Lektorinnen begleitet haben, und Frau Dipl.-Ing. Franziska Kaufmann, die mir beim Layout zur Seite stand. Mein Kollege Prof. Dr. Christian Martin hat mich in vielen Gesprächen ermutigt, dieses Buch zu schreiben. Ich danke Frau Prof. Dr. Anja Schanzenberger und Herrn Prof. Dr. Michael Lutz von der Hochschule Augsburg, sowie Herrn Prof. Dr.-Ing. Heinz Wörn von der Universität Karlsruhe für viele wertvolle Hinweise. Besonders danke ich meiner langjährigen Mitarbeiterin Frau Dipl.-Inf. Gertraud Matzke für die geduldige und gewissenhafte Durchsicht des Manuskripts. Eine große Hilfe waren meine wunderbaren Kinder Christina, Katja und Andreas, die mir viele nützliche Korrekturhinweise aus der Sicht von Studenten und Berufsanfängern gegeben haben. Den Firmen und Forschungsinstituten danke ich für die Unterstützung und das Überlassen von Bildmaterial.

Die vorgestellten Programmbeispiele wurden mit verschiedenen Versionen von MATLAB 7 getestet. Programmcode, Zusatzinformationen und die Lösungen der Aufgaben finden Sie auf meiner Homepage:

*[www.hs-augsburg.de/homes/stark/Buch\\_Robotik\\_mit\\_Matlab](http://www.hs-augsburg.de/homes/stark/Buch_Robotik_mit_Matlab)*

Möge dieses Lehrbuch dazu beitragen, eine engere Verzahnung der Informatik mit den Ingenieurwissenschaften zu fördern und das Programmieren als einen Weg zu einem vertieften Verständnis von Lehrinhalten zu praktizieren.

Friedberg, im Juli 2009

Georg Stark

## ■ Vorwort zur 2. Auflage

Die Ausgabe einer 2. Auflage legt es nahe, die Inhalte an die in der Zwischenzeit erfolgten Fortschritte anzupassen. Im Bereich der Robotik können die Fortschritte mit den beiden Schlüsselwörtern *kollaborativ* und *kognitiv* beschrieben werden.

Kollaborative Roboter (Abkürzung *Cobot*) werden verstärkt als Assistenten des Menschen eingesetzt. Solche Anwendungen gibt es zunehmend im Bereich der Industrie, z. B. in der Montage, und in der Medizin, z. B. als Chirurgieroboter. Aber auch im Bereich der Servicerobotik, beispielsweise in der Pflege, gibt es viele neue Anwendungen für Cobots.

Die kognitive Robotik beschäftigt sich mit dem intelligenten Verhalten von Robotern auf einer hohen Ebene. Dieses beruht auf Wissen, Erinnern, Verstehen, Lernen, Planen und sogar Kreativität. Kognitive Roboter sind zunächst eine Weiterentwicklung der humanoiden Roboter. Aber auch alle anderen Roboterarten können mit kognitiven Fähigkeiten ausgestattet werden.

Die vergangenen Jahre waren auch durch Veränderungen auf dem Markt der Roboterhersteller gekennzeichnet. Viele kleinere Roboterhersteller sind verschwunden oder wurden von größeren übernommen. Andererseits gibt es vor allem in den Bereichen der kollaborativen und kognitiven Robotik neue, sehr innovative und erfolgreiche Hersteller.

Der zweite Schwerpunkt meines Buches, die Anwendung von technisch-wissenschaftlicher Standardsoftware, insbesondere MATLAB mit der aktuellen Version R2021b, ist ebenfalls durch große Fortschritte geprägt. Eine neue Bedienoberfläche berücksichtigt, dass wesentlich umfangreichere Bedienfunktionen zur Verfügung stehen. Daneben gibt es neue, leistungsfähige Verfahren, um Programmcode besser zu dokumentieren, und um MATLAB-Programme als eigenständige Anwendungen für nahezu alle Rechnerplattformen und das Internet bereitzustellen. Die Funktionalität von MATLAB wird durch zusätzliche Toolboxen wesentlich erweitert. Diese beziehen sich sowohl auf neue, verbesserte mathematische und grafische Verfahren, aber auch auf neue Anwendungsbereiche wie Robotik, Kommunikationssysteme, Bioinformatik oder Finanzwirtschaft. Auch die Möglichkeiten zur Fehleranalyse und Programmoptimierung werden weiterentwickelt. Für Neueinsteiger ist besonders wertvoll, dass für MATLAB nun auch eine Online-Version zur Verfügung steht, die mit einem Standard-Browser benutzt werden kann. Die Installation auf einem lokalen Rechner entfällt somit.

Lösungen zu den Aufgaben, Vorgehensweise bei der Lösungserstellung, Zusatzsoftware, ergänzende und weiterführende Literatur sind auf [plus.hanser-fachbuch.de](http://plus.hanser-fachbuch.de) und [www.hs-augsburg.de/homes/stark/Buch\\_Robotik\\_mit\\_Matlab](http://www.hs-augsburg.de/homes/stark/Buch_Robotik_mit_Matlab) abrufbar.

Danken möchte ich wieder den Firmen und Instituten, die mich mit Bildern, Zusatzinformationen und Ratschlägen versorgt haben. Meine Tochter Christina Stark und Herr Dietmar Steidl von KUKA Deutschland haben mir daneben auch noch interessante Einblicke in Struktur und Funktion einer modernen Robotersteuerung ermöglicht. Herr Oliver Jählig von Mathworks Deutschland hat mir viele wichtige Hinweise bezüglich der neu hinzugekommenen Funktionen von MATLAB gegeben. Mein besonderer Dank geht ganz besonders auch an Frau Silakova und Frau Kubiak vom Carl Hanser Verlag, die immer mit viel Geduld und Engagement an meiner Seite standen.

Dieses Buch möchte ich meinen wunderbaren Enkeln Gabriel, Karla, Juliane und Jaron widmen. Sie sind für mich eine unschätzbare Quelle an Lebensfreude, Inspiration und Zuversicht. Mögen sie in eine Welt hineinwachsen, die diese fantastischen Möglichkeiten der Technik und der Informatik verantwortlich nutzt, zum Wohle der gesamten Menschheit und ihrer Lebensgrundlagen.

Friedberg, im Februar 2022

Georg Stark

# Inhalt

<b>Vorwort</b> .....	<b>V</b>
<b>1 Einführung in die Robotik</b> .....	<b>1</b>
1.1 Historie .....	1
1.2 Definition und Klassifikation .....	5
1.2.1 Roboter .....	5
1.2.2 Robotik als Wissenschaft .....	7
1.3 Industrieroboter .....	8
1.3.1 Mechanischer Aufbau .....	9
1.3.2 Komponenten eines Robotersystems .....	17
1.3.3 Steuerung und Systemsoftware .....	19
1.3.4 Anwendungsprogrammierung und Bedienung .....	21
1.4 Andere Roboterklassen .....	24
1.4.1 Serviceroboter und mobile Roboter .....	24
1.4.2 Mikroroboter .....	26
1.4.3 Humanoide und kognitive Roboter .....	27
1.5 Datenfluss in einem Robotersystem .....	29
1.6 Zusammenfassung .....	31
<b>2 Grundlagen der Robotermathematik</b> .....	<b>33</b>
2.1 Formale Modelle .....	33
2.2 Punkt, Gerade, Ebene .....	35
2.2.1 Punkt und Koordinaten .....	35
2.2.2 Vektoren .....	36
2.2.3 Gerade in Ebene und Raum .....	37
2.2.4 Schnittpunkt zweier Geraden .....	39
2.2.5 Ebene im Raum .....	40
2.2.6 Schnittgerade zweier Ebenen .....	40



2.3	Trigonometrische Funktionen .....	41
2.3.1	Gradmaß, Bogenmaß, Einheitskreis .....	41
2.3.2	Inverse trigonometrische Funktionen .....	42
2.3.3	Winkel in einem allgemeinen Dreieck, Kosinussatz ....	43
2.4	Lineare Algebra .....	43
2.4.1	Vektoren und Matrizen .....	43
2.4.2	Lineare Gleichungssysteme .....	47
2.4.3	Lineare Abbildung .....	51
2.4.4	Darstellung der Orientierung durch Eulerwinkel .....	58
2.5	Polynome .....	63
2.6	Differentielle Zusammenhänge .....	64
2.6.1	Ableitung von Funktionen .....	64
2.6.2	Berechnung der Jacobimatrix .....	65
2.7	Zusammenfassung .....	67
2.8	Aufgaben .....	68
<b>3</b>	<b>Programmieren mit MATLAB .....</b>	<b>73</b>
3.1	Erste Schritte .....	73
3.1.1	Was ist MATLAB? .....	73
3.1.2	Bedienoberfläche .....	75
3.1.3	Variablen in MATLAB .....	78
3.1.4	Arithmetische Operationen .....	84
3.1.5	Behandlung von Zeichenketten .....	87
3.1.6	Programmstrukturen .....	90
3.1.7	Ein-/Ausgabe und Dateioperationen .....	97
3.1.8	Grafik .....	99
3.2	Unterstützung der Robotermathematik .....	101
3.2.1	Schnitt, Abstand von Ebene und Gerade .....	101
3.2.2	Winkelberechnung .....	105
3.2.3	Koordinatentransformationen .....	106
3.2.4	Inverse Eulertransformation .....	108
3.2.5	Synthese von Polynomen .....	109
3.2.6	Berechnung der Jacobimatrix .....	111
3.2.7	Funktionsbibliothek ROBOMATS .....	112
3.3	MATLAB als technische Programmiersprache .....	115
3.3.1	Besondere Eigenschaften .....	115
3.3.2	Softwareentwurf .....	117
3.4	Zusammenfassung .....	121
3.5	Aufgaben .....	122

<b>4</b>	<b>Modellierung der kinematischen Struktur</b>	<b>127</b>
4.1	Einführung	127
4.1.1	Freiheitsgrad und kinematische Kette	127
4.1.2	Geeignetes Robotermodell	129
4.2	Kinematikmodell nach Denavit-Hartenberg	130
4.2.1	Konzept	130
4.2.2	Kinematische Strukturen in der Praxis	133
4.3	Programm zur Berechnung der DH-Parameter	138
4.3.1	Softwareentwicklung	138
4.3.2	Implementierung und Test	140
4.4	Transformationen zwischen Roboter- und Weltkoordinaten	144
4.4.1	Vorüberlegungen	144
4.4.2	Vorwärtstransformation	146
4.4.3	Rücktransformation	148
4.5	Zusammenfassung	164
4.6	Aufgaben	165
<b>5</b>	<b>Entwurf von Bahnsteuerungen</b>	<b>169</b>
5.1	Prinzipien	169
5.1.1	Einbindung der Bahnsteuerung in den Datenfluss	169
5.1.2	Verfahrenarten	171
5.2	Bahnplanung	173
5.2.1	Trajektorie	173
5.2.2	Geschwindigkeitsprofil	180
5.2.3	Synchronisation und Anpassung an den Interpolationstakt	185
5.3	Interpolation	186
5.3.1	Echtzeitanforderungen	186
5.3.2	Interpolationsvektor und Geschwindigkeitsprofil	187
5.3.3	Trajektorie	188
5.4	Programmierung	190
5.4.1	Umsetzung des Bewegungsmodells	190
5.4.2	Modellzustand und Modellparameter	193
5.4.3	Globale Modellfunktionen	194
5.4.4	Hilfsfunktionen	203
5.5	Test und Visualisierung	205
5.5.1	Testanwendung	205
5.5.2	Visualisierung und Auswertung	207

5.6	Zusammenfassung .....	209
5.7	Aufgaben .....	211
<b>6</b>	<b>Programmieren im Großen .....</b>	<b>215</b>
6.1	Geeignete Softwarekonzepte .....	216
6.1.1	Allgemeine Anforderungen .....	216
6.1.2	Unterstützung durch MATLAB .....	217
6.2	Komponentenorientierte Programmierung in MATLAB .....	220
6.2.1	Konzept .....	220
6.2.2	Programmierung .....	224
6.3	Integration der Serverkomponente Echtzeitinterpolator .....	230
6.3.1	Systemarchitektur .....	230
6.3.2	Funktionsumfang des Echtzeitinterpolators .....	232
6.3.3	Exportierte Methoden und Ereignisse .....	234
6.3.4	Programmierung der Client-Schnittstelle .....	235
6.3.5	Programmierung der Server-Schnittstelle und Registrierung .....	237
6.4	Realisierung eines Robotersimulators mit Grafikmodell .....	239
6.4.1	Beschreibung des Grafikmodells .....	239
6.4.2	Initialisierung und Ausführung der Gesamtsoftware ...	240
6.5	Zusammenfassung .....	241
6.6	Aufgaben .....	243
<b>7</b>	<b>Anwendungen .....</b>	<b>245</b>
7.1	Grundsätze .....	245
7.2	Beispiele .....	247
7.2.1	Palettieren .....	248
7.2.2	Bearbeiten Langloch .....	253
7.3	Zusammenfassung .....	258
7.4	Aufgaben .....	259
<b>8</b>	<b>Fehlerbehandlung und Optimierung .....</b>	<b>261</b>
8.1	Fehler im Programmcode .....	261
8.1.1	Syntax- und Laufzeitfehler .....	261
8.1.2	Debugger .....	262
8.2	Behandlung externer Fehler .....	266
8.2.1	Ausgabe von Meldungen .....	267

8.2.2	Überprüfung fehlerhafter Daten .....	267
8.2.3	Try-Catch-Konstrukt .....	268
8.3	Programmoptimierung .....	269
8.3.1	Rechenzeit .....	270
8.3.2	Programmstruktur und Quellcode .....	273
8.4	Beispiel – Verbesserung der Bahnsteuerung .....	277
8.4.1	Codeanalyse .....	277
8.4.2	Fehlerüberwachung .....	278
8.4.3	Optimierung der Rechenzeit .....	279
8.5	Zusammenfassung .....	279
8.6	Aufgaben .....	281
<b>9</b>	<b>Literaturverzeichnis .....</b>	<b>283</b>
	<b>Index .....</b>	<b>285</b>

Ergänzendes Material auf [plus.hanser-fachbuch.de](http://plus.hanser-fachbuch.de):

- Lösungen zu den Aufgaben
- Vorgehensweise bei der Lösungserstellung
- Zusatzsoftware
- ergänzende und weiterführende Literatur



# 1

# Einführung in die Robotik

## Zielsetzung

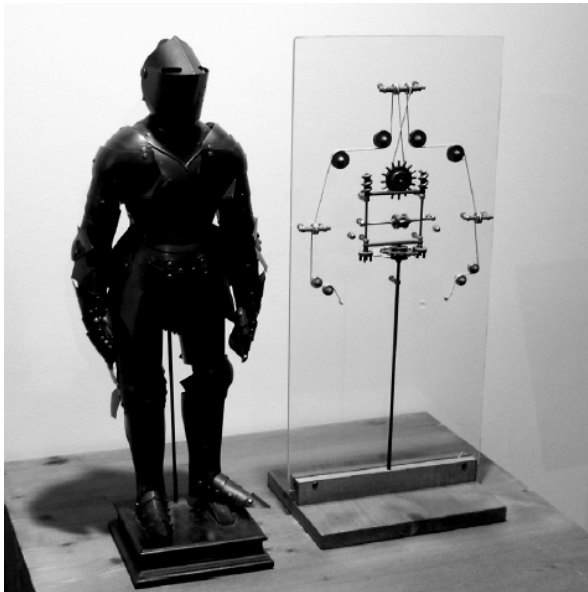
Sie können nur dann erfolgreich Software für Roboter entwickeln, wenn Sie ein Grundwissen auf diesem Fachgebiet haben. Ein Blick auf die Historie und ihre auslösenden Impulse hilft, auch die aktuelle Robotertechnik besser zu verstehen. Die Schlüsselbegriffe *Roboter* und *Robotik* werden in diesem Kapitel erklärt.

Die *Industrieroboter* als wichtigste Klasse werden ausgehend von den Anforderungen und den wesentlichen Systemparametern *Arbeitsraum*, *Traglast*, *Genauigkeit* und *Geschwindigkeit* dargestellt. Die weiteren Roboterklassen *Serviceroboter*, *Mikroroboter* und *humanoide, kognitive Roboter* werden nur kurz gestreift, da die bereits ausgereifte Technik der Industrieroboter auch deren technologische Basis bildet.

Zum Abschluss werden die gemeinsamen Grundprinzipien der verschiedenen Roboterklassen aufgezeigt, die im integrierten Material-, Energie- und Datenfluss eines Robotersystems zum Ausdruck kommen.

## ■ 1.1 Historie

Roboter werden zuerst in der Literatur erwähnt, in Karel Capeks satirischem Stück „Rossum’s Universal Robots“ von 1922. Seitdem werden sie mit intelligenten, selbstständig agierenden Automaten gleichgesetzt. Sie sind so zu einem Synonym für hoch entwickelte Technik geworden. Mit dieser Idee verbunden sind Hoffnungen auf mehr Wohlstand, Komfort und humane Arbeitsplätze, aber auch Ängste vor Arbeitsplatzabbau, Armut und Fremdbestimmung.

**Bild 1.1**

Nachbau einer roboterähnlichen Maschine von Leonardo da Vinci (Foto: Erik Möller)

Die heutige Robotertechnik ist das Ergebnis eines langen Entwicklungsprozesses. Erste Entwürfe von roboterähnlichen Maschinen stammen aus dem 16. Jahrhundert von dem Universalgenie Leonardo da Vinci. Ein Nachbau ist in Bild 1.1 dargestellt. Um 1700 werden musikspielende Puppen und damit die ersten Vorläufer von Robotern gebaut. Die historische Entwicklung der Robotertechnik ist in Tabelle 1.1 wiedergegeben. Aufgeteilt nach Dekaden werden zunächst die Fortschritte der Rechnertechnik als wichtigster Motor der Robotertechnik beschrieben. Es folgen die wesentlichen Entwicklungsschritte bei den Robotern selbst. Dabei wird zwischen Industrierobotern und den anderen Roboterklassen unterschieden. Erstere werden bereits ab Mitte der 1970er-Jahre in größeren Stückzahlen eingesetzt, sind aber mit einer konservativeren Technik ausgestattet. Letztere repräsentieren zwar den jeweils neuesten Stand der Technik, sind dadurch aber aufwendiger und noch nicht so ausgereift.

**Tabelle 1.1** Geschichtliche Entwicklung der Robotertechnik

Um 1700	Technik der mechanischen Uhren Entwicklung von musikspielenden Puppen
1950 – 60	Die Elektronik, basierend auf Elektronenröhren, später Transistoren, wird eingeführt. Die Rechnertechnik wird kommerziell eingesetzt. Basierend auf der Technik der numerisch gesteuerten Werkzeugmaschinen entwickeln George Devol und Joe Engelberger den ersten programmierbaren Roboter.

1960 – 70	<p>Informatik und künstliche Intelligenz als wichtige Teilwissenschaft werden eingeführt.</p> <p>Robotikinstitute werden gegründet und die Robotikforschung kommt in Schwung.</p> <p>Der erste kommerzielle Roboter wird an eine Automobilfabrik ausgeliefert.</p>
1970 – 80	<p>Der Mikroprozessor wird erfunden und in Steuerungen eingesetzt.</p> <p>Ab der zweiten Hälfte des Jahrzehnts werden Roboter in der Automobilindustrie in größerem Umfang für Schweißen und Handhabung verwendet.</p> <p>Der erste anthropomorphe Roboter mit zwei Beinen wird entwickelt.</p>
1980 – 90	<p>Die Leistungsfähigkeit der Mikrocomputertechnik steigt rasant.</p> <p>Die Schnelligkeit und Genauigkeit von Robotersystemen werden verbessert.</p> <p>Die Anzahl der eingesetzten Systeme nimmt stark zu.</p> <p>Roboter werden mit Sensoren und Kameras ausgestattet, z. B. zur Nahtverfolgung beim Schweißen.</p>
1990 – 00	<p>Steuerungen werden auf die PC-Technik<sup>1</sup> umgestellt. Die Vernetzung, auch über das Internet, nimmt zu.</p> <p>Viele neue Anwendungsbereiche für Industrieroboter außerhalb der Automobilindustrie werden erschlossen.</p> <p>Serviceroboter und mobile Roboter werden entwickelt und in ersten Anwendungen eingesetzt. Großen Anklang findet der <i>Robocup</i>, eine weltweite Bewegung, mobile und intelligente Roboter für kooperative Ballspiele werden entwickelt.</p>
2000 – 09	<p>Die zunehmende Leistungsfähigkeit der Rechner erlaubt die Realisierung von intelligenten Systemen.</p> <p>Medizinroboter werden zur Praxisreife gebracht, roboterähnliche Bewegungsstrukturen als künstliche Glieder und künstliche Sinnesorgane entwickelt.</p> <p>Kooperative und humanoide Roboter werden entwickelt.</p> <p>Mikro- und Makroroboter werden entwickelt.</p>
2010 – 19	<p>Aufgrund der steigenden Integrationsdichte der Rechnerbausteine werden kompakte Rechnersysteme mit einer früher nicht vorstellbaren Leistungsfähigkeit ermöglicht. Auch die Techniken und Technologien der künstlichen Intelligenz, z. B. neuronale Netze und wissensbasierte, lernende Systeme, werden weiterentwickelt.</p> <p>Autonome Roboter, ausgestattet mit Multisensoren und Multikinematiken, kombiniert mit unterschiedlichen mobilen Systemen, kommen auf den Markt. Kollaborative Roboter (Cobot) und autonome Systeme kommen in steigendem Umfang zur Anwendung. Autonome Roboter werden als Schwarm eingesetzt.</p> <p>Roboter finden als Assistenten, Chirurgieroboter und Mikroroboter zur Untersuchung von Körperhöhlen ihren festen Platz in der Medizin. Der Einsatz von Cobots in der Industrie und in neuen Anwendungsgebieten, z. B. in der Pflege, nimmt zu. Roboter mit immer umfangreicheren kognitiven Fähigkeiten werden entwickelt.</p>

<sup>1)</sup> Die in Personal Computern verwendete Technik stellt inzwischen einen weltweiten Standard dar.



Die geschichtliche Entwicklung zeigt, dass die entscheidenden Impulse für die Weiterentwicklung der Roboter durch die Rechner-technik erfolgten. So war z. B. im Jahr 1983 eine Robotersteuerung mit zwei Mikroprozessoren mit 15 MHz Taktfrequenz und einem Speicher mit 256 KB ausgestattet. Heutige Steuerungen arbeiten mit mehreren Prozessoren mit Taktfrequenzen von über 5 GHz und Speichern mit weit mehr als 10 GB.

Ein weiterer wichtiger Aspekt ist die Vernetzung. Sie begann zwar bereits in den 80er-Jahren. Mit der Anwendung der für Personal Computer entwickelten Standards für Hardware und Software in der Steuerungstechnik wurden jedoch die Möglichkeiten zur Vernetzung wesentlich verbessert. Heute kann fast jede Industriesteuerung für Wartungs- und Diagnosezwecke über das Internet erreicht werden.

Eine wichtige Rolle spielte auch die Weiterentwicklung der elektrischen Antriebe durch den Einzug der Halbleitertechnik. Nur durch die Verwendung von Leistungstransistoren ist es möglich, bürstenlose und damit wartungsarme Motoren einzusetzen. Bezüglich der Werkstoffe war der Übergang von geschweißtem Stahl auf Aluminiumguss ein wichtiger Schritt. Dieser kann kostengünstiger gefertigt werden und ist bei gleicher Steifigkeit wesentlich leichter. Inzwischen werden auch vermehrt Kunststoffe eingesetzt.

Sensoren und Kamerasysteme finden bei den Industrierobotern nur zögerlich Eingang, etwa ab Mitte der 1980er-Jahre. Dies liegt daran, dass durch Sensoren die Komplexität und damit auch die Anfälligkeit wesentlich steigen. In vielen Fällen kann durch hohe Fertigungsgenauigkeit der Vorprodukte und durch genaue Positionierung der Einsatz von Sensoren umgangen werden. Da Serviceroboter und mobile Roboter in wesentlich unstrukturierteren Umgebungen eingesetzt werden, sind für sie Sensoren unerlässlich. Jedoch sind die Genauigkeitsanforderungen auch kleiner als in der Industrie.

Die ersten bedeutenden Anwendungsgebiete für die Industrieroboter sind Schweißen und Handhaben in der Automobilindustrie gewesen. Sehr bald hatten auch Roboter und roboterähnliche Bestückungsautomaten in der Elektronikindustrie ihren festen Platz. Hinzu kamen Anwendungen in anderen Industrien wie der Metall-, Glas-, Nahrungsmittel- und Getränkeindustrie. Bisher wurden weltweit mehr als 3 Mio. Industrieroboter installiert.

Serviceroboter werden in rasant steigendem Umfang für Dienstleistungen eingesetzt. Seit einigen Jahren gibt es einen Trend, in der Produktion Roboter auch in direkter Kooperation mit Personen zu verwenden. Dies stellt sehr hohe Anforderungen an ein abgesichertes, adaptives, intelligentes Verhalten. Als höchste Form gelten humanoide Roboter mit Armen und Beinen und zunehmend kognitiven Fähigkeiten, die als Assistenten des Menschen wirken.

In eine völlig andere Richtung geht die Entwicklung von einfachen, aber kostengünstigen Robotern für den privaten Gebrauch. Verwendet werden sie vorwiegend für einfache Dienste wie Staubsaugen, Rasenmähen oder für Hobby und Unterhaltung, zunehmend auch für Servicedienstleistungen.

## ■ 1.2 Definition und Klassifikation

### 1.2.1 Roboter

Am Anfang der Entwicklung standen die Industrieroboter. Eine allgemein anerkannte Definition, festgelegt durch den VDI<sup>2</sup>, lautet:

#### **Definition – Industrieroboter**

Industrieroboter sind universell einsetzbare Bewegungsautomaten mit mehreren Achsen, deren Bewegungen hinsichtlich Bewegungsfolge und Wegen bzw. Winkeln frei programmierbar und gegebenenfalls sensorgeführt sind. Sie sind mit Greifern, Werkzeugen oder anderen Fertigungsmitteln ausrüstbar und können Handhabungs- und/oder Fertigungsaufgaben ausführen.

Eine weitere Definition wurde von der *International Organization for Standardization*<sup>3</sup> (ISO 8373) festgelegt. Diese wird auch von der internationalen Robotervereinigung *International Federation of Robotics*<sup>4</sup> (IFR) benutzt.

Oft wird der Begriff Roboter aber viel weiter gefasst. So werden zum Teil ganz allgemein Systeme, die über Sensoren Informationen aufnehmen, diese weiterverarbeiten und daraus Aktionen ableiten, ebenfalls als Roboter bezeichnet. Nach dieser Definition lassen sich dann auch autonome Fahrzeuge mit unterschiedlichen Bewegungsvorrichtungen und mit Sensoren ausgerüstete Bau-, Baumfäll- und Erntemaschinen einordnen.

In Tabelle 1.2 wird der Versuch einer Klassifikation unternommen.

---

<sup>2)</sup> Verein Deutscher Ingenieure, VDI-Richtlinie 2860, Montage- und Handhabungstechnik

<sup>3)</sup> <https://committee.iso.org/home/tc299>

<sup>4)</sup> <https://ifr.org/industrial-robots>

**Tabelle 1.2** Klassifikation von Robotern

Roboterart	Merkmale und Anwendungsbereiche
Industrieroboter <ul style="list-style-type: none"> <li>▪ Gelenkarm<sup>5</sup></li> <li>▪ Portal</li> <li>▪ SCARA</li> <li>▪ Cobot</li> <li>▪ paralleler Roboter</li> </ul>	<p>Diese Roboter sind für den industriellen Einsatz entwickelt. Sie zeichnen sich durch Robustheit, Schnelligkeit, Genauigkeit und hohe Traglast aus. Unterteilt werden sie nach Art und Anordnung der Bewegungsachsen. Der neue Begriff <i>Cobot</i> bezeichnet Roboter, die auch in direktem Kontakt zu Personen betrieben werden können.</p> <p>Industrieroboter handhaben sowohl Werkstücke als auch Werkzeuge und führen dabei auch Fertigungsprozesse aus. Typische Fertigungsprozesse sind Schweißen, Kleben, Schneiden, Lackieren, Handhaben, Montieren.</p>
Serviceroboter <ul style="list-style-type: none"> <li>▪ professioneller Einsatz</li> <li>▪ privater Einsatz</li> </ul>	<p>Serviceroboter erbringen Dienstleistungen für den Menschen. Sie müssen sich autonom auch in unstrukturierter Umgebung bewegen können. Deshalb sind Sie mit Sensoren zur Erfassung der Umwelt und mit Navigationseinrichtungen ausgestattet. Weitere wichtige Merkmale sind eine leicht und sicher zu handhabende Bedienoberfläche, sowie ein für den Menschen jederzeit ungefährliches Verhalten.</p> <p>Serviceroboter werden in zwei Klassen unterteilt. Solche für den professionellen Einsatz genügen hohen Ansprüchen. Die wichtigsten Anwendungsbereiche sind Logistik, Reinigung, gefährliche Umgebung, Hotel und Landwirtschaft.</p> <p>Immer häufiger werden auch kostengünstige Serviceroboter für den privaten Bereich angeboten, z. B. für Rasenmähen, Staubsaugen oder Transport im Haushalt.</p>
Mobile Roboter	<p>Mobile, autonome Roboter haben viele Gemeinsamkeiten mit den Servicerobotern.</p> <p>Typische Einsatzgebiete sind Logistik, Produktion, Labore und Hotels. Daneben werden sie auch in gefährlichen Umgebungen, unter Wasser oder extraterrestrisch auf fremden Himmelskörpern eingesetzt.</p>
Mikro-, Nanoroboter	<p>Mikroroboter haben nur eine Größe von wenigen Millimetern und können sich autonom in kleinen Strukturen bewegen und dort Aktionen durchführen, z. B. im Inneren des Körpers. Eine andere Entwicklungsrichtung zielt darauf ab, viele Mikroroboter als <i>Schwarm</i> agieren zu lassen, z. B. zur Erkundung.</p> <p>Unter Nanorobotern versteht man autonome Maschinen und Strukturen im Kleinstformat, bis hinunter zur Größe von Molekülen.</p>

<sup>5)</sup> Ein synonymer Begriff ist *Knickarmroboter*.

Roboterart	Merkmale und Anwendungsbereiche
Humanoider, kognitiver Roboter	<p>Humanoide Roboter haben menschenähnliches Aussehen mit Armen und Beinen. Sie sollen vergleichbare kognitive, sensorische und motorische Fähigkeiten haben, um mit Menschen direkt kommunizieren und interagieren zu können. Dies ist jedoch noch weitgehend Gegenstand der Forschung. Kognitive Roboter stellen eine intelligente Weiterentwicklung der bisherigen humanoiden Roboter dar.</p> <p>Langfristig wird angestrebt, humanoide Roboter als multifunktionale Arbeitsmaschine und Assistent des Menschen einzusetzen.</p>

### 1.2.2 Robotik als Wissenschaft

Der Begriff *Robotik* ist nicht so allgemeingültig definiert wie *Roboter*. Er beschreibt das Bestreben, eine Wissenschaft der Roboter zu definieren. Der Versuch einer Definition lautet:

#### Definition – Robotik

Die Robotik ist ein interdisziplinäres Wissensgebiet, das sich umfassend mit der Realisierung und Anwendung von Robotersystemen beschäftigt. Der Mensch mit seinen manuell-motorischen, sensorischen und kognitiven Fähigkeiten soll immer stärker unterstützt und ersetzt werden. Starken Einfluss auf die Robotik haben Maschinenbau, Werkstoffkunde, Elektrotechnik, Mathematik, Informatik und für zukünftige, hoch entwickelte Systeme auch die Kognitionswissenschaften, die Psychologie und die Biologie.

Die *Robotik* als Wissenschaft kann auch in Anlehnung an die *Informatik* definiert und strukturiert werden, da es viele Parallelen gibt. Befasst sich die Informatik nur mit der digitalen Welt in Rechnern, so stellt die Robotik deren Erweiterung hin zur realen, physisch erfahrbaren Welt dar. Über die *Sensorik* werden Phänomene und Zustände der realen Welt erfasst und digitalisiert. Ähnlich der *Kognition* beim Menschen werden die aufgenommenen Informationen mit der internen, digitalen Modellwelt abgeglichen und weiterverarbeitet. Die Folge sind Entscheidungen und Aktionen, die über die *Aktorik* auf die reale Welt zurückwirken und dort Zustände verändern.

Deshalb macht es auch Sinn, die Robotik ähnlich der Informatik zu strukturieren:

- Theoretische Robotik

beschäftigt sich mit der Abstraktion, Modellbildung und grundlegenden Fragestellungen, die sich mit der sensorischen Erfassung und Verarbeitung von Information, sowie deren Umsetzung in Aktionen befassen.

- **Praktische Robotik**  
entwickelt Methoden, um umfangreiche Programmsysteme (Software) für die Realisierung von Robotern erstellen zu können.
- **Technische Robotik**  
befasst sich mit der technischen Realisierung von Robotern.
- **Angewandte Robotik**  
untersucht den Einsatz von Robotersystemen in den verschiedenen Anwendungsgebieten.



#### **Hinweis – Praktische Robotik**

Im vorliegenden Lehrbuch liegt das Hauptgewicht auf der Praktischen Robotik. Die Theoretische Robotik liefert die dafür benötigten formalen mathematischen Modelle. Auf die Themen der Technischen Robotik wird direkt im Anschluss eingegangen, bei der Erörterung der Anforderungen an Roboter und der daraus folgenden Roboterarten. Das große Gebiet der Angewandten Robotik wird in einem eigenen Kapitel (Kapitel 7) anhand zweier Beispiele ansatzweise behandelt.

## ■ 1.3 Industrieroboter

Die längste praktische Erfahrung besteht mit Industrierobotern, die zunächst hauptsächlich in der Automobilindustrie eingesetzt worden sind. Inzwischen gibt es viele neue Anwendungsbereiche in anderen Industrien, aber auch außerhalb, z. B. in Laboren, in der Medizin, auf dem Bau und in der Landwirtschaft. Ihre Technik ist inzwischen ausgereift und stellt so einen guten Ausgangspunkt für andere Roboterarten dar. Der Blick auf die Roboter erfolgt aus Sicht der Anwendung, d. h. ihre besonderen Anforderungen bestimmen hauptsächlich die zu realisierenden technischen Merkmale. Zunächst werden in diesem Abschnitt die verschiedenen Ausprägungen des mechanischen Aufbaus vorgestellt. Es wird betrachtet, welche weiteren Komponenten zu einem vollständigen Robotersystem gehören. Anschließend wird die Steuerung mit ihren Funktionen dargelegt. Ihre *Systemsoftware* ist ausschlaggebend für die Intelligenz und Mächtigkeit der Roboteroperationen. Abschließend wird auf die Themen Bedienung und *Anwendungssoftware* eingegangen.

### 1.3.1 Mechanischer Aufbau

Der mechanische Aufbau beschreibt, aus welchen mechanischen Komponenten und mit welcher Struktur ein Roboter aufgebaut ist. Dies betrifft hauptsächlich die Art und die Anordnung der Bewegungsachsen. Dies wird als *Roboterkinematik* bezeichnet. Die wesentlichen mechanischen Eigenschaften eines Roboters sind dadurch bereits festgelegt.

#### Anforderungen und Kriterien

Industrieroboter führen entweder Transportaufgaben aus oder sie bearbeiten und analysieren Werkstücke und Maschinen. Daraus folgen die beiden prinzipiellen Bewegungsarten. Sie unterscheiden sich bezüglich Geschwindigkeit und Genauigkeit der Bewegungsbahn.

##### 1. Transport

Beim Transport müssen große Entfernungen überbrückt werden, oft auch mit großen Traglasten. Beispiele dafür sind das Beladen von Paletten oder der Transport von Karosserieteilen in der Automobilfertigung. Angestrebt wird eine hohe Geschwindigkeit bei gleichzeitig geringem Verschleiß der Mechanik. Das exakte Einhalten eines vorgegebenen Bahnverlaufs wird in der Regel nicht gefordert, jedoch eine hohe Genauigkeit der Position am Ende der Bewegungsbahn.

##### 2. Bearbeitung

Bei einem Bearbeitungsprozess muss ein Werkzeug mit hoher Bahngenauigkeit und gleichbleibender Geschwindigkeit bezüglich eines Werkstücks bewegt werden. Die Genauigkeitsanforderungen haben Vorrang gegenüber einer hohen Geschwindigkeit. Typische Beispiele sind das Schweißen von Nähten oder das Schneiden von Blechen.

Die dritte denkbare Bewegungsvariante – bahngenau und schnell – kommt nur selten vor, z. B. beim Laserstrahlschneiden. Dabei spielen dann die Massen von Roboter und Traglast und ihre räumliche Verteilung eine große Rolle.

Eine weitere wichtige Eigenschaft, bedingt durch die Mechanik, ist der Arbeitsraum des Roboters. Dieser Begriff bezieht sich nicht nur auf die Raumkoordinaten, die angefahren werden können, sondern auch auf die Orientierung, die räumliche Ausrichtung des Werkzeugs oder Greifers. Die Anforderungen an den verfügbaren Arbeitsraum sind eine Folge von Größe und Form des zu bearbeitenden Werkstücks. Dazu zählen nicht nur die äußeren Abmessungen, sondern auch die Form der Innenräume, beispielsweise bei einer Autokarosserie. Zusammengefasst sind die folgenden Parameter für die Gestaltung der Mechanik, einschließlich Antriebssystem, ausschlaggebend:

- Arbeitsraum,
- Traglast,
- Genauigkeit,
- Geschwindigkeit/Beschleunigung.

Ein Blick in die Prospekte der Roboterhersteller zeigt, welche Lösungswege gegangen worden sind. Die Parameter Genauigkeit und Arbeitsraum werden hauptsächlich durch Art, Anzahl und Anordnung der Bewegungsachsen bestimmt. Beim Begriff Genauigkeiten müssen drei Arten unterschieden werden:

- Wiederholgenauigkeit

Die Wiederholgenauigkeit beschreibt, wie genau ein Punkt im Raum bei Wiederholungen angefahren wird. Sie wird beeinträchtigt durch nicht-deterministische, wahrscheinlichkeitsbehaftete Prozesse, z. B. hervorgerufen durch Reibung und Getriebeispiel. Die Wiederholgenauigkeit ist vor allem ein Qualitätsmerkmal der Mechanik.

- absolute Genauigkeit

Die absolute Genauigkeit definiert, wie genau der durch ein Bewegungsprogramm numerisch vorgegebene Zielpunkt in Wirklichkeit erreicht wird. Eine hohe absolute Genauigkeit wird verfehlt, wenn das mathematische Steuerungsmodell nicht alle wesentlichen mechanischen Effekte berücksichtigt. Ein Beispiel dafür ist die Getriebeelastizität. Sie führt dazu, dass ein Roboterarm bei hoher Traglast etwas absinkt.

- Bahngenauigkeit

Die beiden ersten Arten von Genauigkeit beziehen sich auf den ruhenden Roboter. Die Bahngenauigkeit beschreibt, wie genau die durch das Anwendungsprogramm vorgegebene Bahnkurve ausgeführt wird. Dazu braucht es leistungsstarke Antriebe, um die dynamischen Kräfte, hervorgerufen durch Beschleunigung, zu kompensieren. Weiter sind hoch entwickelte Regelungsverfahren erforderlich, um während der Bewegung die Abweichung zwischen Soll- und Istwert klein zu halten. Schließlich müssen die Sollwerte für die Achsen durch einen kleinen Interpolationstakt in möglichst kurzen Zeitabständen vorgegeben werden.

# Index

## Symbole

4GL-Sprache 22

## A

Abfahrbarkeit 247  
Abhängigkeitsbericht 275  
Ablaufsteuerung 22  
Ableitung 87  
– von Funktionen 64  
absolute Genauigkeit 10  
absolute Transformation 53  
Achse 11  
A-Kompilierung 219  
aktives Gelenk 129  
Aktorik 7, 27, 32  
aktueller Parameter 94  
Analyse 117, 138  
analytische Lösung 149  
anonyme Funktion 93  
Anwendungsdaten 248 ff., 254 f.  
Anwendungsprogrammierung 21  
Anwendungssoftware 32, 245  
A-Programmierung 219  
Arbeitsraum 9, 32, 180  
Architekturdiagramm 119  
asynchrone Kommunikation 227  
Ausgabeparameter 94  
Ausnahme 91  
Ausnahmebehandlung 95  
Automatisierungsschnittstelle 221  
autonome Roboter 3

## B

Bahnframe 175, 188  
Bahngenaugigkeit 10, 186  
Bahngeschwindigkeit 249, 255  
Bahnsatz 170  
Bahnschaltbedingungen 171  
Bahnsteuerung 130, 169 f., 187, 190, 197  
Bahnvektor 177, 188 f.  
Base Workspace 78, 81, 97  
Bearbeitungsprozess 22, 255  
Bediengeräte 19  
Bedienhandgerät 23  
beliebige Drehung 55, 106  
Bewegung mit Sensoreingabe 195  
Bewegungsachse 11  
Bewegungsarten 9  
Bewegungsmodell 190  
Bewegungssegment 183  
BHG 19  
Bogenmaß 41

## C

C++ 215, 217, 230, 237  
Callback 226 f.  
catch-Block 95  
Cell Array 83  
Character Array 88  
Client 221  
Client-Schnittstelle 235  
Client-Server-Konfigurationen 223  
Client-Server-Modell 218



Cobot 3, 15  
Codeanalyse 277  
Code Analyzer Report 273  
COM (Component Object Model) 220, 222  
Command History 78  
Computeralgebra-Systeme 74  
COM-Schnittstelle 220  
CP 172  
CPCIR 173, 257  
CPLIN 173, 249, 256 f.  
Cramer'sche Regel 49, 68  
Current Directory 77  
Current Folder 78

## D

Datenfluss 29 f., 186  
DCOM 222  
Debugger 76, 126, 261 f.  
Definitionsbereich 142  
deklarative Ebene 246  
Delta-Roboter 16  
Dependency Report 274  
DH-Parameter 139  
DH-Transformation 130, 140  
Differenzorientierung 159, 162  
Differenztransformation 107, 113, 122  
D-Interpretation 219  
D-Kompilierung 219  
Domäne 22  
D-Programmierung 219  
Drehachse 11  
Drehgelenk 127  
Drehung, beliebige 55, 106  
Drehung in der Ebene 55  
Drehvektor 175, 188  
Dreiecksprofil 180 f., 183  
Dreiecksungleichung 152  
Dynamikmodell 129

## E

Echtzeit 19  
Echtzeitanforderungen 186

Echtzeitinterpolator 215, 230, 242, 250, 256  
Effektor 18  
Effektorkoordinatensystem 146  
Effektortransformation 159  
Eigenvektor 50  
Eigenwert 50  
eingebettete Funktion 81, 93  
Einheitskreis 41  
einschleifiger Regelkreis 11  
endlicher Automat 190  
Endlosschleife 265  
Energiefluss 30  
Entwurf 117, 119, 139, 197, 247  
erweiterte Systemmatrix 47  
Eulertransformation 58, 68, 70, 108, 112, 122, 149  
Eulerwinkel 149  
Exception 267  
Exception Handling 91  
explizite Darstellung 37  
externe Fehlerquellen 262

## F

Fehlerbehandlung 261, 269, 278  
Fehlererkennung 267  
Fehlerreaktion 267  
Fehlerüberwachung 278  
Feininterpolation 187  
Fernsensor 18  
Flansch 18  
Flanschkoordinatensystem 137, 143, 146, 159  
flexible Fehlerreaktion 269  
formale Parameter 94  
formale Sprache 34  
Formalisierung 118  
Formatspezifikation 88  
Frame 51, 53, 174  
freier Vektor 52  
Freiheitsgrad 24, 37, 133  
Fundamentalsatz der Algebra 63  
Funktionsbeschreibung 95

**G**

Gelenk 11, 129  
Gelenkachse 130  
Gelenkarmroboter 12, 134, 143, 197  
geometrisches Modell 131  
Geschwindigkeitsfaktor 182  
Geschwindigkeitsprofil 169, 180, 187, 204  
Gewichtsausgleich 12  
Gleichzeitigkeit 19  
Globalbereich 81, 122  
globale Variable 192  
Gradient 87  
Gradmaß 41  
Grafikfunktionen 99  
Grafikmodell 230, 239  
Grübler Formel 128  
Grundachsen 11, 133, 157  
GUID 220  
GUI (Graphical User Interface) 74

**H**

Haltepunkt 263 f.  
Handachsen 133, 137, 149, 159  
Handverfahren 19, 24  
Handwurzelpunkt 133, 137, 149, 159, 161  
Hauptbereich 81  
Hauptdiagonale 45  
Hauptfunktion 93  
Hauptkomponente 230  
Hesse-Normalenform 38, 67, 103  
Hilfetext 95  
Hilfsfunktion 196, 203  
Hinführungsvektor 38  
homogene Matrix 51, 53  
homogenes Gleichungssystem 47  
humanoide Roboter 4, 7, 27

**I**

IDispatch 221, 223, 228, 242  
Implementierung 117, 140, 160, 199, 201, 247, 249, 254 f.  
implizite Darstellung 37

Implizite Deklaration 78  
implizite For-Schleife 82  
Industrieroboter 3, 5 f., 8, 32  
inhomogenes Gleichungssystem 47  
interne Fehlerquellen 262  
Interpolation 197  
Interpolationstakt 10, 185 f., 197, 199, 250, 256  
Interpolationsvektor 173, 187  
Interpolatorkomponente 230  
inverse Eulertransformation 159  
IPO-Takt 197, 233  
IUnknown 223

**J**

Jacobimatrix 65, 111 f., 114, 122  
Java 217  
Just-in-Time-Kompilierung 271

**K**

Kettenregel 64  
Kinematikmodell 129  
Kinematikparameter 145  
kinematische Kette 128  
Knickarm 12  
Kognition 7, 28, 32  
Kognitive Abstrakte Robotermaschine 28  
kognitive Roboter 7, 27  
kollaborative Roboter 3, 15  
Kommandofenster 78  
Komponentendarstellung 38  
Komponentenschnittstelle 217  
Konfiguration 145, 154, 160, 162  
Konkretisierung 118  
Körper 130  
Kosinussatz 43, 67, 105, 152 f.  
Kreisbogen 177  
Kreisebene 177  
Kreisinterpolation 180  
Kreiskoordinatensystem 189, 199  
kritische Bahnlänge 182  
künstliche Intelligenz 3

## L

- Laufzeitfehler 262, 265
- Leistungsteil 18
- Leistungstest 270
- Linearisierung 148
- Linearkombination 44
- Linksdivision 85, 102
- lokale Funktion 93
- lokale Variable 94
- Lösungsraum für die Rücktransformation 145
- Lösungsvektor 47
- Lösungsverfahren, allgemein 47

## M

- Materialfluss 29
- Mathematik 33
- MATLAB 73 f.
  - Coder 218
  - Compiler 218
  - Compiler SDK 218
  - Externschnittstelle 217
- Matrix 81
- Mechanik 9
- Medizinroboter 3
- Mehrdeutigkeiten 149
- Mehrpunktbahn 186
- Meldungsbezeichner 267
- Meldungstext 267
- MException 95
- MEX-Compiler 217
- Mikroroboter 6, 26
- mobile Roboter 3, 6, 24
- Modell 34
- modellbasierte Programmierung 116
- modellbasierter Entwurf 217
- Modellfunktion 191, 194
- Modellparameter 191, 193
- Modellzustand 191, 193

## N

- Nahsensor 18
- Nanoroboter 6
- Navigation 27, 32
- Nebenachse 11
- Notaus-Taster 19
- Numerisches System 74

## O

- objektorientierte Programmierung 76, 217
- objektorientierter Entwurf 217
- Offline-Programmierung 23, 32
- Optimierung 245, 247, 252, 255, 257, 261
- Optimierungskriterien 180
- Orientierung des Koordinatensystems 53
- Orientierungsinterpolation 175, 180, 201
- Orientierungsmatrix 176
- Ortskoordinatensystem 131
- Ortsvektor 52

## P

- parallele Roboter 16
- Parameterdatei 145, 192, 236, 254
- Parameter-Syntax 224
- Parameterüberprüfung 268
- passives Gelenk 129
- Polynome 63, 101, 109, 112, 181
- Portalroboter 13, 136
- Pose 135
- praktische Robotik 8
- private Funktion 93
- Profiler 271, 273, 279
- Programmidentifizierer 221
- Programmieren im Großen 74, 116, 215
- Programmieren im Kleinen 74, 216
- Programmoptimierung 269
- prozedurale Ebene 246
- PTP 171, 173, 249, 251 f., 256
- Punkt-Syntax 224
- Python 217

**Q**

Quadranten 41  
quadratische Matrix 46

**R**

Rang einer Matrix 47  
räumliche und funktionelle Integration 245  
Reaktionsfunktion 225 f., 228 f., 242  
Rechenzeit 261, 269 ff., 273, 279  
Rechtzeitigkeit 19  
Regelung 11, 32, 170  
Regelungsverfahren 10  
Registrierung 219, 221, 238  
relative Transformation 53  
Richtgeschwindigkeit 182  
Richtungsvektor 38  
Robocup 3  
ROBOMATS 73, 101, 112, 268  
Roboter  
– Vektor 144  
– Wissen 34  
Roboterbasis 146  
Roboterbewegung 22  
Roboterhand 16  
Roboterkinematik 9  
Roboterkoordinaten 144, 171  
Robotermodell 129  
Roboterzeit 273  
Robotik 1, 7, 32  
Rücktransformation 144

**S**

SCARA-Roboter 14, 136  
Schnittgerade 40, 104  
Schubachse 11  
Schubgelenk 127  
Segmentart 183, 186  
Segmentlänge 186  
Segmentliste 186, 199  
Segmentzeit 185  
Sektion 75, 276

Sensoren 4, 170  
Sensorik 7, 27, 32  
Server 215, 221  
Serviceroboter 3, 6, 24  
Set Path 76  
Short-Circuit-Operator 279  
Sichtbarkeit 80, 92, 122  
Simulation 245, 247, 252, 257  
SIMULINK 73  
singuläre Matrix 46  
Singularität 62, 145, 193  
Skriptsprache 22  
Softwareentwurf 115, 117, 122  
Softwarekomponente 220  
Softwarequalität 216  
Steuerprogramm 248, 250, 255 f., 258  
Steuerungsmodul 20  
Steuerungsrechner 18  
Strichmodell 240  
string 88, 90  
String Array 88  
struct 83  
Strukturdiagramm 119, 193, 197  
Stützpunkt 22  
Suchpfad 76  
Summenregel 64  
synchrone Kommunikation 227  
Synchronisation 183 f.  
Syntaxfehler 261  
Systemkomponente 245  
Systemmatrix 47 ff.  
Systemsoftware 8, 19, 32

**T**

Tab 75  
TCP (Tool Centre Point) 23, 32  
Teachen 23, 32  
Teilbewegung 180, 182, 199, 206, 209  
Testbetrieb 24  
totales Differential 65  
Trajektorie 170, 173, 188, 199  
Transformationsmatrix 52  
Trapezprofil 181, 183  
try-catch 91, 95, 268

## U

- überbestimmtes, lineares Gleichungssystem 86
- Überschleifen 171, 186

## V

- Vektor 36
  - freier 52
- Vektordarstellung 38, 67
- Vektorisierung 83, 274, 275
- Vektorraum 43
- Verfahrenart 171, 173
- Verifikation 117
- Vernetzung 4
- Visualisierungsmodus 207
- Vorwärtstransformation 144, 146

## W

- Weltkoordinaten 144, 171
- Werkzeughandhabung 18
- Wiederholgenauigkeit 10
- Wiederverwendbarkeit 216
- Winkelberechnung 105, 153
- Winkelhand 133, 137
- Wissensbasis 29

## Z

- Zeichenketten 87
- Zeitmessung 271, 275
- Zellen in M-Dateien 276
- Zentralhand 133, 137f.
- Zustandsdaten 190
- Zykluszeit 247, 249f., 253, 255f.
- Zylinderkoordinatensystem 36
- ZYZ-Eulertransformation 58, 159
- ZYZ-Eulerwinkel 173