

Detlef Classe

*Beitrag zur Steigerung der Flexibilität automatisierter  
Montagesysteme durch Sensorintegration und  
erweiterte Steuerungskonzepte*



Detlef Classe

*Beitrag zur Steigerung der Flexibilität  
automatisierter Montagesysteme durch  
Sensorintegration und erweiterte  
Steuerungskonzepte*

Herausgegeben von  
Professor Dr.-Ing. Klaus Feldmann,  
Lehrstuhl für  
Fertigungsautomatisierung und Produktionssystematik

**FAPS**



Carl Hanser Verlag München Wien

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

Tag der Einreichung:	10. Juni 1988
Tag der Promotion:	26. September 1988
Dekan:	Prof. Dr. F. Hofmann
Berichterstatter:	Prof. Dr.-Ing. K. Feldmann Prof. Dr.-Ing. D. Seitzer

CIP-Titelaufnahme der Deutschen Bibliothek

Classe, Detlef:  
Beitrag zur Steigerung der Flexibilität automatisierter  
Montagesysteme durch Sensorintegration und erweiterte  
Steuerungskonzepte / Detlef Classe. Hrsg. von Klaus  
Feldmann. - München ; Wien : Hanser, 1988.  
(Fertigungstechnik Erlangen)  
Zugl.: Erlangen, Nürnberg, Univ., Diss., 1988  
ISBN 3-446-15529-5

Dieses Werk ist urheberrechtlich geschützt.  
Alle Rechte, auch die der Übersetzung, des Nachdrucks  
und der Vervielfältigung des Buches oder Teilen daraus,  
vorbehalten.  
Kein Teil des Werkes darf ohne schriftliche Genehmigung  
des Verlages in irgendeiner Form (Fotokopie, Mikrofilm  
oder ein anderes Verfahren), auch nicht für Zwecke der  
Unterrichtsgestaltung - mit Ausnahme der in den §§ 53,  
54 URG ausdrücklich genannten Sonderfällen -, reproduziert  
oder unter Verwendung elektronischer Systeme verarbeitet,  
vervielfältigt oder verbreitet werden.

(C) Carl Hanser Verlag München, Wien 1988  
Herstellung: Bresler Druckerei, Erlangen  
Printed in Germany

## V O R W O R T

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Assistent am Lehrstuhl für Fertigungsautomatisierung und Produktionssystematik der Universität Erlangen-Nürnberg.

Herrn Professor Dr.-Ing. K. Feldmann, dem Leiter des Lehrstuhls für Fertigungsautomatisierung und Produktionssystematik am Institut für Fertigungstechnik, danke ich für die großzügige Förderung bei der Durchführung dieser Arbeit.

Herrn Professor Dr.-Ing. D. Seitzer, dem Leiter des Lehrstuhls für Technische Elektronik am Institut für Elektrotechnik, danke ich für die Übernahme des Korreferates und für seine wohlwollende Unterstützung.

Ferner gilt mein Dank Herrn Priv.-Doz. Dr.-Ing. Bär und Herrn Dr.-Ing. Wurmthaler, die durch Diskussionen und Anregungen die Arbeit gefördert haben, sowie allen Studenten, die durch ihr Mitwirken zum Gelingen der Arbeit beigetragen haben.

Detlef Classe

BEITRAG ZUR STEIGERUNG DER FLEXIBILITÄT AUTOMATISierter  
MONTAGESYSTEME DURCH SENSORINTEGRATION UND ERWEITERTE  
STEUERUNGSKONZEPTE

- INHALTSVERZEICHNIS -

	Seite
Verzeichnis der wichtigsten Formelzeichen und Indizes	ix
1. <u>EINFÜHRUNG</u>	1
1.1 Montageautomatisierung unter Flexibilitäts- gesichtspunkten	1
1.2 Zielsetzung und Vorgehensweise	6
2. <u>SENSOREN IN AUTOMATISIERTEN MONTAGESYSTEMEN</u>	7
2.1 Aufgaben und Funktionen von Sensoren	8
2.1.1 Aufnahme technischer Daten	9
2.1.2 Aufnahme organisatorischer Daten	11
2.1.3 Zum Funktionsbegriff	13
2.1.4 Sensorfunktionen	14
2.2 Ziele des Sensoreinsatzes	18
2.3 Die Planung des Sensorbedarfs als Teilaspekt der Montageplanung	19
2.3.1 Planungsablauf	19
2.3.2 Einbeziehung des Sensoraspekts in die Montageplanung	20
2.4 Auswahl von Sensoren	23
2.4.1 Arbeitsweise von Sensoren	23
2.4.2 Systematische Sensorauswahl	29
2.4.3 Rechnerunterstützung bei der Sensorauswahl	41

3.	<u>SENSORGESTÜTZTE ROBOTERPROGRAMMIERUNG</u>	46
3.1	Methoden der Bewegungsprogrammerstellung	46
3.1.1	Textuelle Programmierung	48
3.1.2	Sensorgestützte Programmierung	51
3.2	Entwicklung eines neuen sensorgestützten Programmierverfahrens	53
3.2.1	Auswahl eines Meßkonzepts	56
3.2.2	Prüfung marktüblicher Sensoren	60
3.2.3	Entwicklung eines neuen Sensors	61
3.2.4	Berechnung der Kontur	66
3.2.5	Regelungskonzept	70
3.2.6	Integration in ein Programmiersystem	77
3.3	Simulation des Programmiervorgangs	80
3.3.1	Modellansatz für die kontinuierliche Komponente des Roboters mit Sensor	82
3.3.2	Ableitung der Bewegungsgleichungen	91
3.3.3	Umweltmodell	97
3.3.4	Ergebnisse des Simulation	103
3.4	Datenreduktion bei sensorgestützter Programmierung	108
3.4.1	Bestimmungsmethoden für Ersatzkurven	109
3.4.2	Meßpunktselektion durch Überprüfen der Bahntoleranz	113
3.4.3	Kubische Spline-Interpolation	115
3.4.4	HERMITESche Spline-Interpolation	118
3.4.5	Vergleich der Verfahren	120
4.	<u>MONTAGE MIT DUPLEXROBOTER</u>	126
4.1	Notwendigkeit und Nutzen der Duplexmontage	126
4.2	Die Bewegungskordinierung zweier Roboter als Lösungsansatz für spezielle Montageprobleme	130
4.2.1	Problemstellung für die Montage form- instabiler Werkstücke und Baugruppen	130
4.2.2	Typen von Bewegungen	131

4.2.3	Realisierungskonzepte für koordinierte Bewegungen	135
4.3	Ermittlung abgestimmter Bahnkurven	138
4.3.1	Einführung eines globalen Koordinatensystems	141
4.3.2	Bahngekoppelte Bewegungen mit konstantem Abstandsvektor	143
4.3.3	Bahngekoppelte Drehbewegungen	145
4.3.4	Zeitgekoppelte Bewegungen	149
4.4	Steuerungskonzepte	149
4.4.1	Glatter Übergang zwischen zwei Bewegungsabschnitten	150
4.4.2	Koordiniertes Überschleifen bei bahngekoppelten Bewegungen	153
4.4.3	Grundstruktur eines Interpolators	159
4.4.4	Linearinterpolation	164
4.4.5	Zirkularinterpolation	166
4.4.6	Satzweise Synchronisation	169
4.5	Sprachkonzept für die Duplexmontage	172
4.6	Struktur eines Testsystemaufbaus	176
5.	<u>ZUSAMMENFASSUNG</u>	180
6.	<u>LITERATURVERZEICHNIS</u>	183

Verzeichnis der wichtigsten Formelzeichen und Indizes

Formelzeichen

<u>a</u>	Abstandsvektor
<u>b</u>	Koeffizientenvektor
<u>c</u>	Elastizität
<u>C</u>	Elastizitätsmatrix
<u>d</u>	Dämpfung
<u>d<sub>m</sub></u>	Drehmoment
<u>D</u>	Dämpfungsmatrix
<u>e<sub>a</sub></u>	Einheitsvektor der Korrekturrichtung
<u>e<sub>f</sub></u>	Einheitsvektor der Kraft
<u>e<sub>g</sub></u>	Einheitsvektor der Drehachse
<u>e<sub>t</sub></u>	Einheitsvektor der Vorschubrichtung
<u>e<sub>0</sub></u>	Einheitsnormalenvektor der Werkstückoberfläche
<u>E</u>	Koordinatensystem
<u>E<sub>st</sub></u>	E-Modul
<u>f</u>	Kraftvektor
<u>F</u>	Übertragungsfunktion
<u>g</u>	Drehachse
<u>h</u>	Verbiegungsvektor, Abstand zweier Drehebene
<u>h<sub>t</sub></u>	Abstandsvektor
<u>H</u>	Koeffizientenmatrix
<u>k</u>	Randbedingungsvektor
<u>K</u>	Entkoppelmatrix
<u>l</u>	Länge
<u>L</u>	LAGRANGE-Funktion
<u>m</u>	Masse
<u>M</u>	Massenmatrix, Überschleifmatrix
<u>n</u>	Tangentenvektor
<u>n<sub>0</sub></u>	Einheitsnormalenvektor der Schnittebene
<u>N</u>	Interpolationsschrittzahl
<u>o</u>	verallgemeinerte Koordinate
<u>p</u>	Positionsvektor
<u>q</u>	Vektor der freien Koordinaten

$Q^*$	potentialfreie Kräfte
$r$	Radius
$\underline{R}$	Drehmatrix
$s$	LAPLACE-Operator
$t$	Zeit
$\Delta t$	Zeitintervall
$\underline{t}$	Zeitvektor
$T$	Kinetische Energie
$\underline{T}$	Transformationsmatrix
$T_a$	Überschleifzeit
$\underline{T}_t$	Totzeit
$\underline{u}$	Spannungsvektor, Stellvektor
$u_b$	Geschwindigkeitswert
$v_b$	Bahngeschwindigkeit
$V$	potentielle Energie
$W_b$	Widerstandsmoment
$x$	kartesische Koordinate
$y$	kartesische Koordinate
$z$	kartesische Koordinate
$\alpha$	Drehwinkel, Verbiegungswinkel
$\beta$	Drehwinkel
$\gamma$	Drehwinkel
$\delta$	Winkelschritt, Abklingkonstante
$\epsilon$	Dehnung
$\vartheta$	Steigungswinkel der Verbiegungsgeraden
$\lambda$	Reibwinkel
$\mu$	Drehwinkel
$\rho$	Steigungswinkel der Kraftgeraden
$\varphi$	Drehwinkel
$\omega$	Kreisfrequenz

Mehrfach benutzte Indizes

a	abhängig
b	Bahn-
e	Ersatz-
f	Führungsgröße
i	Zählvariable
ink	Inkrement
j	Zählvariable
k	Zählvariable, Kontur-
m	Mittelpunkt-, Meß-
r	Roboter-
s	Start-, Sensor-, Sollwert
u	unabhängig
x	kartesische Koordinate
y	kartesische Koordinate
z	kartesische Koordinate, Ziel-



## 1. Einführung

Der Anteil manueller Tätigkeiten bei der Montage von Serienprodukten fällt branchenspezifisch unterschiedlich aus, liegt aber häufig so hoch, daß die Personalkosten einen wesentlichen Anteil der Montagekosten ausmachen [1,2]. Unter dem Druck einer verschärften Wettbewerbssituation bemühen sich die Hersteller um fortschreitende Automatisierung der Arbeitsgänge in der Montage, ein hoher Automatisierungsgrad ist aber bisher vorwiegend auf Montagesysteme für Massenprodukte beschränkt geblieben. Zusätzlich erschweren veränderte Bedingungen auf den Absatzmärkten die Automatisierungsbemühungen, indem sie die Hersteller zu steigender Produkt- und Variantenvielfalt zwingen. Dies führt bei konventionellen Montageeinrichtungen zu abnehmender Kapitalproduktivität, da die abnehmenden Losgrößen entweder die Auslastung der Produktionsmittel verhindern oder zu höheren Umrüstzeiten führen. In beiden Fällen sinkt die tägliche Einsatzzeit. Eine Verbesserung der Situation ist mit flexiblen Montageeinrichtungen zu erwarten, die durch Fortschritte bei der Steuerungstechnik, der Handhabungstechnik und der Sensortechnik auf Sicht eine wirtschaftliche Montageautomatisierung auch außerhalb der Massenfertigung ermöglichen werden.

### 1.1 Montageautomatisierung unter Flexibilitätsgesichtspunkten

Gebräuchliche Montageeinrichtungen sind heute Einzelvorrichtungen, mit denen die Montageautomatisierung einst begonnen hat, Rundtakt- und Längstransferautomaten für Produkte, die geringen Änderungsforderungen unterliegen, und zunehmend universell programmierbare Einrichtungen. Rundtaktautomaten zeichnen sich durch kurze Taktzeiten aus, die begrenzte Zahl von Operationen pro Einheit und ihre starre Verkettung wirken allerdings nachteilig. Bei komplizierten Produkten und einem geringen Anteil manueller Montageopera-

tionen werden meist Längstransfersysteme bevorzugt. Bei dieser Konstruktion sind einzelne Montagestationen durch ein Transportsystem miteinander verbunden, wobei durch Zwischenpuffer die Einzelstationen entkoppelt werden können. Sofern sich definierte Teilaufgaben abgrenzen lassen, faßt man eine abgeschlossene Einheit mit meist einem Fügegerät und zugehörigen Peripherieeinrichtungen zu einer Montagezelle zusammen. Das vereinfacht Inbetriebnahmen und Wartung.

Das Angebot programmierbarer Geräte ist in den letzten Jahren stetig größer geworden, es erleichtert die Berücksichtigung von Flexibilitätsforderungen. Zunächst haben modulare, freiprogrammierbare Bewegungsachsen, die als Bausteine eines Baukastensystems beliebig zusammengesetzt werden können, als Ersatz für konventionelle Fügeeinrichtungen schnelle Verbreitung gefunden [7]. Inzwischen hat aber auch die Akzeptanz bei mehrachsigen Robotern zugenommen. Darauf deuten die Veränderungen der Einsatzzahlen von Robotern in den Jahren 1986 und 1987 hin (Bild 1),

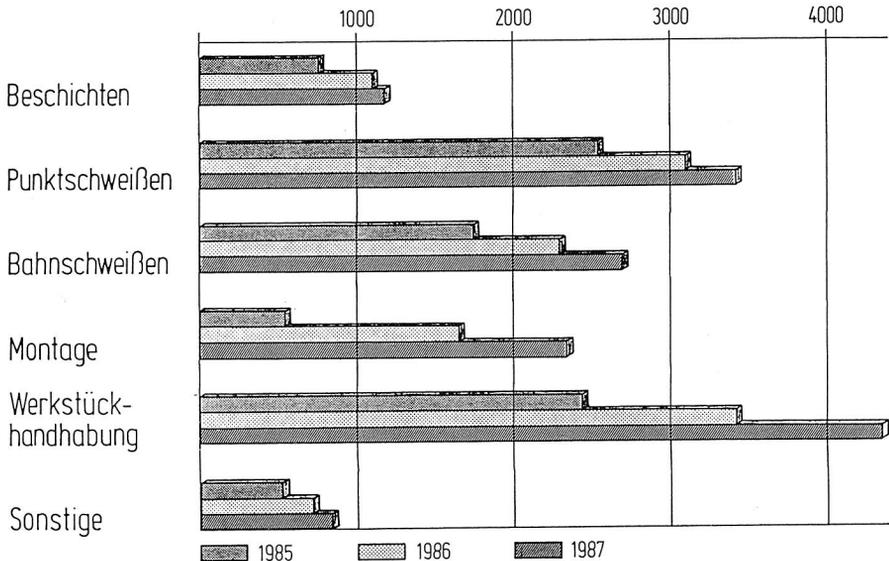


Bild 1: Industrieroboter in der BRD nach [4]

wonach in der Bundesrepublik das Marktsegment für Anwendungen in der Montage überdurchschnittlich gewachsen ist [4]. Prinzipiell sind die meisten Roboter für Montageaufgaben verwendbar; in der Regel werden aber weniger als sechs Freiheitsgrade benötigt, da die Fügebewegung fast immer senkrecht von oben erfolgt.

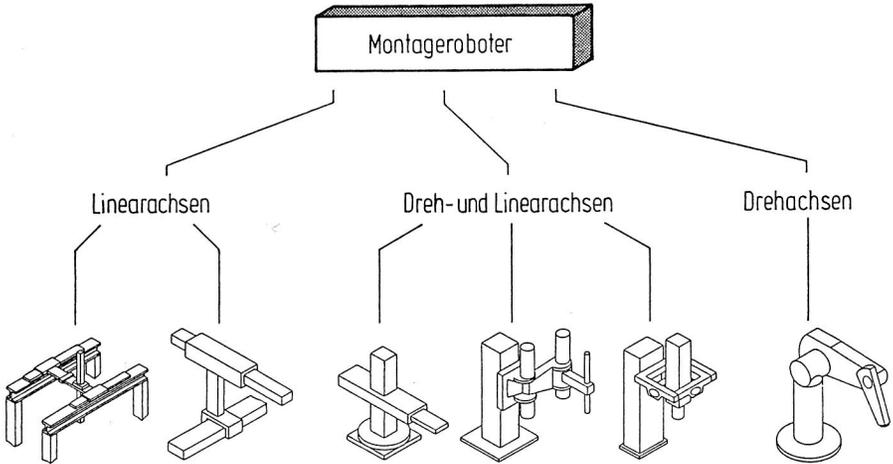


Bild 2: Grundbauarten von Montagerobotern nach [2]

Häufig werden allerdings harte Forderungen an die Positioniergenauigkeit der Geräte gestellt. Sie sind besonders bei Aufgaben der elektrischen Kleinteilmontage und der Leiterplattenbestückung zum Teil schwer zu erfüllen [8]. Probleme treten vor allem bei Geräten in Gelenkbauweise auf, da bei ihnen zur Positionierung eine Koordinatentransformation durchgeführt werden muß, deren Berechnung auf Konstruktionsmaßen beruht, die reale Fertigungsabweichungen der mechanischen Gerätekomponenten nicht berücksichtigen. Die Genauigkeitsfrage erhält besonderes Gewicht, da für einen wirtschaftlichen Betrieb mit sehr kleinen Losgrößen die Programmerstellung möglichst vollständig vom Ausführungsgerät weg in die Arbeitsvorbereitung verlegt wird. Dadurch entfallen die Korrekturmöglichkeiten beim Einlernen von Positionen

unter Sichtkontrolle (Teach-In Programmierung). Etwas entschärft wird die Problematik durch Kalibrierhilfsmittel, die einige Roboterhersteller neuerdings anbieten. Man kann mit ihnen per Software innerhalb eines definierten rechtwinkligen Feldes im Arbeitsraum die erreichbare Genauigkeit verbessern [3].

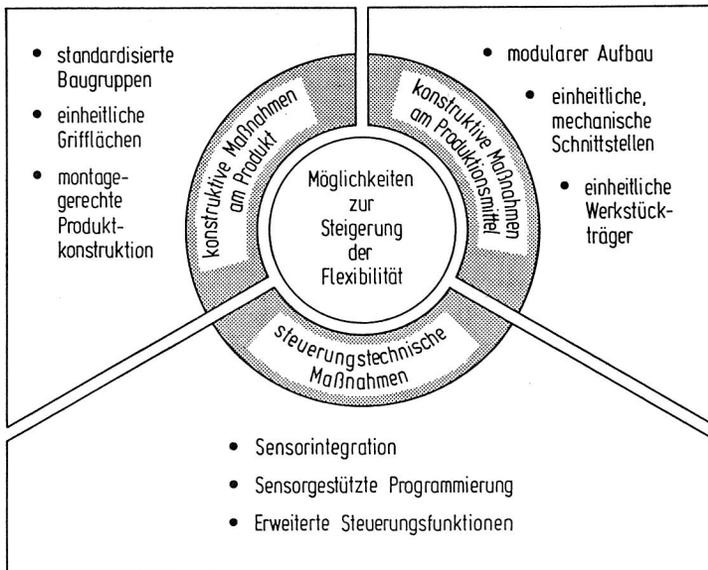
Wegen der eingeschränkten Kinematikforderungen in Bezug auf die Anzahl der geforderten Freiheitsgrade, dem Wunsch nach hoher Arbeitsgeschwindigkeit und der Forderung nach sehr hoher Positioniergenauigkeit sind inzwischen einige Spezialgeräte entwickelt worden, die die Bezeichnung 'Montageroboter' tragen (Bild 2). Mit Ausnahme des Aufbaus als Vertikalnickarm ist bei diesen Bauarten in z-Richtung eine lineare Achse vorgesehen, mit der sehr genaue und schnelle Bahnbewegungen in Fugerichtung möglich sind.

Bedingt durch die schnelle Weiterentwicklung der Produkte wie auch der technischen Einrichtungen, bevorzugt man heute einen modularen Aufbau der Systemkomponenten. Dabei wird die Systemflexibilität durch einfache Austauschbarkeit und Umbaubarkeit der Elemente, ergänzt durch die Programmierbarkeit der Einrichtungen, erreicht [5]. Sie läßt sich steigern durch konsequente Auslegung der Einrichtungen anhand einer Baukastensystematik, Schaffung einheitlicher Schnittstellen zwischen den einzelnen Komponenten und Verwendung einheitlicher Werkstückträger (Bild 3). Systemlösungen mit Robotern haben konventionellen Anlagen gegenüber den Vorteil, häufigen Produktwechsel und sehr kleine Losgrößen beherrschen zu können [3]; höhere Anlagenkosten und niedrigere erreichbare Taktzeiten sind bei entsprechendem Auftragspektrum tolerierbar.

Da automatisierte Montagesysteme auf die Montageprodukte mit ihren verschiedenen Varianten sehr fein abgestimmt sein müssen, beschränken sich die Möglichkeiten zur Steigerung der Systemflexibilität nicht allein auf Entwurf und Aus-

legung von Systemkonzepten, sie können durch produktseitige Maßnahmen ergänzt werden. Um das Aufgabenspektrum eines Montagesystems mit wirtschaftlich sinnvollem Aufwand möglichst breit auszulegen, strebt man eine umfassende Standardisierung von Baugruppen und eine einheitliche Griffflächen-gestaltung an (Bild 3). Generell steht die Produktkonstruktion unter der Forderung, die Montierbarkeit in automatischen Montagesystemen zu berücksichtigen, und die erforderlichen Handhabungs- und Montageoperationen durch konstruktive Maßnahmen so weit als möglich zu vereinfachen.

Als dritte Möglichkeit zur Steigerung der Flexibilität von Montagesystemen kommen steuerungstechnische Maßnahmen in Betracht (Bild 3). Da sie durch den Einfluß der schnellen Entwicklung auf dem Gebiet der Mikroelektronik für die nahe Zukunft besondere Wirksamkeit erwarten lassen, wurden sie als Gegenstand der vorliegenden Arbeit gewählt.



**Bild 3:** Möglichkeiten zur Steigerung der Flexibilität automatisierter Montagesysteme

## 1.2 Zielsetzung und Vorgehensweise

Es ist Ziel der Arbeit, Wege aufzuzeigen, wie durch steuerungstechnische Maßnahmen die Flexibilität automatisierter Montagesysteme gesteigert und dadurch einer wachsenden Produkt- und Variantenvielfalt Rechnung getragen werden kann. Dazu bieten sich insbesondere drei Möglichkeiten an: Durch die Integration von Sensoren können Abweichungen vom geplanten Anlagenverhalten ausgeglichen werden, die sensorgestützte Roboterprogrammierung erleichtert und beschleunigt die Bewegungsprogrammerstellung, und erweiterte Steuerungskonzepte ermöglichen den Einsatz von Robotern für spezielle Anwendungen.

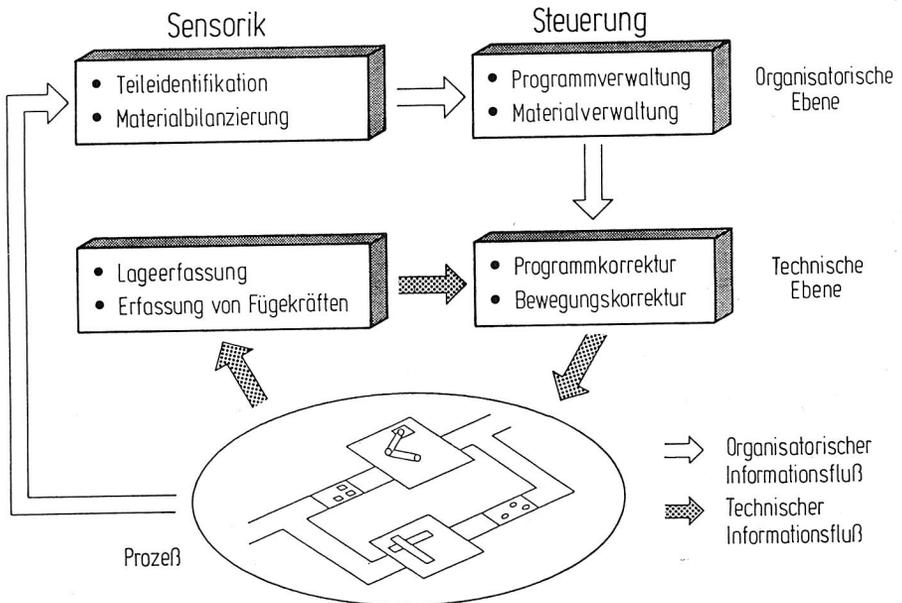
In Kapitel zwei wird die Integration von Sensoren in automatisierten Montagesystemen behandelt. Zu Beginn werden Aufgaben und Funktionen zusammengestellt, von denen ausgehend eine systematische Planungsmethodik für den Sensorbedarf entworfen wird.

Als exemplarische Sensoranwendung im Montagebereich stellt das dritte Kapitel ein neues sensorgestütztes Programmierverfahren vor, das sich durch besonders gute Bedienbarkeit auszeichnet. Empfindlichkeiten des Verfahrens gegen Störeinflüsse werden in einer Simulationsstudie untersucht und anschließend verschiedene Methoden zur Reduktion der aufgenommenen Meßdaten miteinander verglichen.

Das vierte Kapitel behandelt die Erweiterung des Funktionsumfangs von Robotersteuerungen für den Duplexbetrieb. Ausgehend vom Spektrum der Bewegungen, die für die Montage forminstabiler Werkstücke und Baugruppen mit zwei Roboterarmen erforderlich sind, werden Steuerungsmechanismen abgeleitet und Realisierungsmöglichkeiten aufgezeigt.

## 2. Sensoren in automatisierten Montagesystemen

Nach Vorbildern aus der Vorfertigung [6] beginnen sich gegenwärtig auch in der Montage hierarchische Steuerungssysteme durchzusetzen. Diese Art der Gliederung erlaubt es, Teilsysteme für sich in Betrieb zu nehmen, wodurch die Übersichtlichkeit und Verfügbarkeit der Gesamtsysteme verbessert werden. Die unterste Ebene in der Hierarchie bilden Einzelsteuerungen für Handhabungseinrichtungen, Fügeeinrichtungen und Transportsysteme. Diesen Maschinen- und Gerätesteuerungen sind auf der nächsten Ebene Rechner übergeordnet, welche die Steuerung abgegrenzter Teilsysteme übernehmen. Sie verwalten die von der darüber angeordnete Ebene abgesetzten Aufträge und steuern entsprechend dieser Aufträge den Arbeitsablauf innerhalb der Teilsysteme. Während das Schwergewicht der 'organisatorischen' Steuerungsaufgaben den höheren Hierarchieebenen zugeordnet ist,



**Bild 4:** Einsatz von Sensoren in flexiblen Montagesystemen

sind die 'technischen' Steuerungsaufgaben, zu denen alle stellgliednahen Aufgaben und vor allem die Bewegungssteuerung gehören, weitgehend auf der Ebene der Gerätesteuerungen lokalisiert.

Um die Flexibilität der automatisierten Montagesysteme zu erhöhen und auf auftragsbedingte Änderungen in den Montagevorgängen ebenso wie auf Störungen automatisch reagieren zu können, wird ein möglichst umfassender Datenrückfluß aus dem Montageprozeß angestrebt. Sensoren liefern diese Daten, indem sie die organisatorische Steuerung mit Betriebs-, Qualitäts- und Identifikationsdaten und die technische Steuerung mit ausgewählten Meßgrößen versorgen. Bild 4 zeigt die grundsätzlichen Wege der sich einstellenden Informationsflüsse. Der organisatorische Informationsfluß endet beim Eintritt in die technische Steuerung; die organisatorische Steuerung hat also keinen direkten Zugriff auf die Stellglieder, sie gibt der technischen Steuerung lediglich Daten zur Führung der Montagevorgänge vor.

## 2.1 Aufgaben und Funktionen von Sensoren

Gegenwärtig vollzieht sich unter dem Einfluß der Mikroelektronik ein Wandel des Sensorbegriffs. Die hohe Integrationsdichte, die bei der Halbleiterherstellung inzwischen erreicht wird, macht es möglich, umfangreiche elektronische Schaltungen zur Informationsverarbeitung in sehr kleiner Baugröße herzustellen und im Gehäuse von Meßwandlern unterzubringen. Viele Sensoren umfassen heute neben dem Aufnehmer auch die Teilsysteme Verstärkung und Ausgabe und stellen damit eine kompakte, d.h. auf kleinem Raum untergebrachte, klassische Meßkette dar. In einigen Fällen gehen die Möglichkeiten moderner Sensoren aber auch über das reine 'Messen' physikalischer Größen hinaus. Unter 'Messen' versteht DIN 1319 [9], in der die Grundbegriffe der Meßtechnik festgelegt sind, nämlich denjenigen experimentellen Vorgang,

durch den ein spezieller Wert einer physikalischen Größe als Vielfaches einer Einheit oder eines Bezugswertes ermittelt wird. Im engen Sinne ist deshalb die Erfassung nichtphysikalischer Größen in Chemie und Medizin oder bei der Werkstoffprüfung ebensowenig eine Messung wie die mehrdimensionale Aufnahme von Information bei der Analyse von Bild- und Schallmustern. In beiden Fällen kann das erfaßte Merkmal nicht mehr, wie bei Meßgrößen üblich, als Teil oder Vielfaches einer physikalischen Einheit beschrieben werden. Entsprechend komplexe Zusammenhänge sind nur über mehrdimensionale Vergleiche erfaßbar und durch eine Reihe von Merkmalen beschreibbar. Es handelt sich dabei um eine symbolische Zuordnung zwischen einem beobachteten Zusammenhang und der dazu vom Sensor gelieferten Nutzinformation.

Unabhängig von der Komplexität der Zusammenhänge dient die Informationserfassung in Montagesystemen in jedem Fall der Beobachtung charakteristischer Merkmale eines Prozesses oder seines Umfeldes. Da auch die Bedingungen, unter denen die Informationsaufnahme erfolgt, einander sehr ähnlich sind, und sich die Vorgänge bei der Erfassung komplexer Zusammenhänge meist als Messungen nach DIN 1319 [9] mit nachfolgender mehrdimensionaler Auswertung der Meßergebnisse interpretieren lassen, wird hier die Bedeutung des Meßbegriffes weiter gefaßt als in der DIN. Vorgänge der Informationserfassung werden auch dann als Messungen bezeichnet, wenn das Ergebnis nicht als Zahlenwert mit Einheit angebbar ist.

### 2.1.1 Aufnahme technischer Daten

Montagegeräte sind Teil der umfangreichen Einrichtungen, die zur Herstellung von Produkten erforderlich sind. Sie sind in ihrer räumlichen Arbeitsumwelt umgeben von anderen Maschinen und Einrichtungen sowie von den Werkstücken und Baugruppen, die schrittweise zum Produkt zusammengebaut werden sollen. Ebenso wie konventionelle sind programmierbare Montagegeräte

von sich aus lediglich in der Lage, vorgegebene Bewegungsverläufe im Rahmen ihrer Arbeitsgenauigkeit wiederzugeben. Veränderungen innerhalb der Arbeitsumwelt machen deshalb hin und wieder Korrekturen am Bewegungsverhalten erforderlich, was bei den programmierbaren Geräten manuell in Form von Programmänderungen oder automatisch durch Einbeziehung von Sensorinformation erfolgen kann. Die Sensoren nehmen dazu signifikante Umweltausschnitte auf und liefern den Gerätesteuerungen die zur Generierung von Bewegungsänderungen erforderlichen Daten. Da sie für den Betrieb der Geräte selbst nicht erforderlich sind, sondern ausschließlich zur Erfassung der Geräteumwelt dienen, heißen sie 'externe' Sensoren. Sie spielen bei der Planung des Sensorbedarfs die wesentliche Rolle (siehe Abschnitt 2.3). 'Interne' Sensoren dagegen sind integraler Bestandteil von Montagegeräten, sie sind zum Teil für die Gerätegrundfunktionen erforderlich oder dienen der Geräteüberwachung. Weggeber, Referenzpunkt- und Endschalter gehören heute zur Grundausstattung von NC-Achsen und Robotern, auf ihre Auswahl haben die Geräteanwender praktisch keinen Einfluß. Zusätzliche spezielle Gerätesensoren zur Spannungs-, Strom- und Temperaturüberwachung sind dem Montagesystemplaner zumeist unbekannt.

Da Umwelteinflüsse den automatischen Systembetrieb derzeit noch sehr erschweren, behandelt die Fachliteratur vorrangig den Themenkreis der 'externen' Sensoren. Neben der sensorgestützten Roboterprogrammierung [10,11] und der automatischen Überwachung der Montagesysteme [12-17] wird dabei der Verbesserung der Teilebereitstellung große Aufmerksamkeit gewidmet. Die gegenwärtig verfügbaren Bereitstellungseinrichtungen stellen einen gravierenden Flexibilitätsengpaß innerhalb von Montagesystemen dar. Sie sind in konventioneller Ausführung meist auf ein Werkstück ausgelegt und müssen beim Wechsel von Produktvarianten relativ häufig ausgetauscht werden. Die Verfügbarkeit komplexer Sensoren zur Erfassung formbezogener Werkstückgeometrie Größen sowie von Werkstückpositionen und -orientierungen erlaubt es, Teilauf-

gaben der Bereitstellungseinrichtungen auf flexible Geräte, z.B. Roboter, zu übertragen [15-22], wodurch ein Teil der bisherigen, restriktiven Anforderungen an den Ordnungszustand der Werkstücke entfällt, und durch die Verringerung der Anforderungen an die Bereitstellungseinrichtungen deren universelle Verwendbarkeit beträchtlich gesteigert werden kann.

Die Schwerpunktsetzung auf externe Sensoren erschwert allerdings die Weiterentwicklung der automatischen Anlagenüberwachung. Über die reine Störungserkennung hinaus liegt im Hinblick auf einen vollautomatischen Betrieb größerer Einheiten ein besonderer Entwicklungsbedarf bei der automatischen Analyse von Störungen. Für diese Aufgabe sind aber neben den umweltbeschreibenden Zustandsgrößen, die externe Sensoren liefern, auch diejenigen Zustandsgrößen erforderlich, die die geräteinternen Sensoren erfassen. Diese Zustandsinformation müßte zur Weiterverarbeitung von einem übergeordneten Analysesystem abgerufen werden können.

### 2.1.2 Aufnahme organisatorischer Daten

Die Verwendung von werkstückbegleitenden Informationsträgern reduziert den Aufwand zur Datenhaltung und -verwaltung im Rahmen der organisatorischen Steuerung erheblich [23,24]. Durch die Identifikation von Werkstücken oder Baugruppen sind Fehloperationen vermeidbar, was sich positiv auf das Störverhalten des Montagesystems auswirkt. Die Informationsträger werden mit verschiedenen Arbeitsprinzipien angeboten und können Auftragsnummern oder auftrags- bzw. werkstückbezogene Kenndaten enthalten. Sind Schreib-Lese-Stationen installiert, kann die Information nicht nur ausgelesen werden, zusätzlich ist es möglich, den Speicherraum zu nutzen, um Montagefortschritte und Prüfergebnisse zu dokumentieren. Grundsätzlich lassen sich ähnliche Informationsträger auch zur Kennzeichnung von Greifern oder Vorrichtungen verwenden.

Für den störungsfreien Betrieb spielen Bauelementetoleranzen bei der automatischen Montage eine besonders große Rolle. Bereits Toleranzen, die für die Funktion der Bauelemente bedeutungslos sind, können entscheidend dafür sein, ob die automatische Montage reibungslos gelingt. Die Toleranzforderungen an die zu montierenden Bauteile müssen deshalb ausreichend eng angesetzt und höhere Fertigungskosten in Kauf genommen werden. Außerdem muß eine ausreichende Anzahl von Prüfoperationen vorgesehen werden, um die Einhaltung der engen Toleranzmaße zu überwachen. Die verringerte Fehlerrate, die sich aufgrund eng tolerierter Bauteilemaße einstellt, macht dann allerdings weniger manuelle Eingriffe notwendig und erhöht die Verfügbarkeit der Montagesysteme.

Die Qualitätsprüfung hat festzustellen, inwieweit Werkstücke oder Produkte die an sie gestellten Qualitätsforderungen erfüllen. Dabei kann es sich um das Feststellen von Meßwerten handeln (objektives Prüfen) oder von Eigenschaften, die meßtechnisch nicht erfaßbar sind (subjektives Prüfen). Gegenwärtig nimmt die Bedeutung der Qualitätsprüfung in der Montage stark zu, wofür neben der Empfindlichkeit gegenüber Bauteiletoleranzen vor allem die Nachfrage nach hoher Produktqualität verantwortlich ist. Ein wichtiges Ziel bei der Entwicklung neuer Prüfverfahren und -geräte liegt deshalb darin, das Spektrum der objektiven Prüfvorgänge zu erweitern und damit Grundvoraussetzungen für die automatische Qualitätsprüfung zu schaffen. Soweit die Prüfspezifikationen es zulassen und geeignete Prüfgeräte existieren, sind die Prüfoperationen in den Montageablauf integrierbar, was den Materialfluß vereinfachen und überflüssige Montageoperationen mit Ausschussteilen vermeiden hilft.

Um die Bedeutung einzelner Sensorfunktionen für den Betrieb des Gesamtsystems einschätzen zu können, werden im weiteren typische Sensorfunktionen benannt und nach dem Zweck ihres Einsatzes in verschiedene Gruppen eingeordnet. Dazu wird zunächst ein klarer Funktionsbegriff eingeführt.

### 2.1.3 Zum Funktionsbegriff

Montagesysteme sind technische Gebilde; sie bestehen aus einer Anzahl geordneter Elemente, die aufgrund ihrer Eigenschaften miteinander durch Relationen verknüpft sind. Allgemein müssen Systeme von ihrer Umwelt abgrenzbar sein, wobei die Begrenzung die Verbindungen zur Umgebung schneidet. Diese Schnittstellen gegenseitigen Wirkungs austauschs zwischen System und Umgebung bestimmen das Systemverhalten nach außen. Zur Beschreibung des Verhaltens ist, nach der Arbeitsmethode der Systemtechnik, keine Kenntnis der inneren Struktur erforderlich. Allein die Verbindungsgrößen zwischen Umgebung und Systemelementen sind dazu ausreichend [25]. Je nach Wirkungsrichtung bezeichnet man sie als Ein- oder Ausgangsgrößen. Sie kennzeichnen die Funktion des Systems unabhängig davon, um welche Art von Größen es sich handelt. Für die Funktion, wie für das System selbst, spielt die Einordnung in den Gesamtzusammenhang des technischen Gebildes eine wichtige Rolle. Allgemein kann man deshalb nach BROCKHAUS [26] als Funktion die Aufgabe, Verrichtung oder Obliegenheit von Teilen im Rahmen eines Ganzen bezeichnen. WIENDAHL [27] greift diese Definition auf und konkretisiert sie für den technischen Bereich, indem er als Ganzes das Gebilde ansieht, in das das Teil eingeht. Mit den Begriffen 'Gebilde niederer Ordnung' und 'Gebilde höherer Ordnung', die diesen Sachverhalt beschreiben, kommt er zu folgender Definition:

Als Funktion einer Sache wird die Aufgabe dieser Sache im Rahmen einer höheren Ordnung bezeichnet.

Aus dieser Definition wird deutlich, daß sowohl eine Sache selbst, als auch die Sache, die in die erste eingeht, eine Funktion erfüllt. Daraus läßt sich eine umfassende Hierarchie von Funktionen und Teilfunktionen entwickeln, wobei Funktionen ganz unabhängig von der Komplexität eines Systems festgelegt werden können. Analog der Gliederung eines Systems in Teilsysteme läßt sich also die Funktion eines

Systems bis in die gewünschte Detaillierung von Teilfunktionen zerlegen.

Innerhalb der Funktionsstruktur, die sich für ein Montagesystem angeben läßt, sind bestimmte Teilfunktionen enthalten, nämlich die Erfassung von Prozeßinformation, für die Sensoren als Funktionsträger in Frage kommen. Sie werden im weiteren als Sensorfunktionen bezeichnet. Häufig sind für Sensorfunktionen auch mechanische Einrichtungen als Funktionsträger geeignet, die nachfolgende Betrachtung beschränkt sich aber auf 'Sensoren', die in der Lage sind, die aufgenommene Information an die Montagesteuerung weiterzugeben. Der Begriff der Sensorfunktion wird bei der Sensorauswahl u.a. in Abschnitt 2.4.2 wieder aufgenommen und dient dort als abstrakte Charakterisierung der vorliegenden Problemstellung.

#### 2.1.4 Sensorfunktionen

Nach dem Charakter der Messung, für die ein Sensor verwendet wird, unterscheiden RICHTER [28] und NAUMANN [29] drei Typen von Messungen:

- Forschungsmessungen,
- Testmessungen und
- Prozeßmessungen.

'Forschungsmessungen' haben die Analyse eines unbekanntes Sachverhaltes zum Ziel, und der Kenntnisstand über das Meßobjekt ist sehr unvollständig. Bei 'Testmessungen' sind die Eigenschaften des Meßobjektes prinzipiell bekannt, es können aber noch unvorhergesehene Ereignisse eintreten, die Berücksichtigung finden müssen. 'Prozeßmessungen' schließlich liefern in definierter Weise Prozeßinformation, die für den automatischen Anlagenbetrieb benötigt wird. Anstelle des Begriffs 'Prozeßmessung' wird hier der Begriff 'Betriebs-

messung' benutzt, da die Arbeit von einem allgemeineren Prozeßbegriff ausgeht, als er bei RICHTER [28] und NAUMANN [29] verwendet wird (vgl. Abschnitt 2.4.1). Bemerkenswert ist, daß der Charakter der Messung ganz wesentlich die Anforderungen an die Meßeinrichtung beeinflußt und sich auf die erforderliche Vorbereitung und Auswertung der Messungen auswirkt. Das Meßobjekt muß bei 'Betriebsmessungen' so genau bekannt sein, daß die Interpretation der Meßsignale in jedem Fall eindeutig ist, und Sensoren ausgewählt werden können, die möglichst genau das vorliegende Anforderungsspektrum bezüglich Meßbereich, Genauigkeit, Umgebungsempfindlichkeit u.a.m. erfüllen.

Auch wenn der Kenntnisstand über eine Reihe von Montageverfahren noch unzureichend ist, sind 'Forschungsmessungen' an automatisierten Montagesystemen die Ausnahme, da sie mit den Bedingungen der Produktion schlecht vereinbar sind. Der größte Teil der Funktionen, die Sensoren in Montagesystemen übernehmen, dient der Unterstützung des Betriebs der Systeme. Ihrem Charakter nach handelt es sich um 'Betriebsmessungen'. Ein kleinerer Teil von Sensorfunktionen hilft, die Vorbereitungen zum Betrieb zu vereinfachen. Da die dazu nötigen Arbeiten nicht in ihrem vollen Umfang übersehen werden können, sind die erforderlichen Messungen als 'Testmessungen' charakterisierbar. Bild 5 greift diesen Gedanken auf und zeigt, geordnet nach dem Zweck und dem Charakter der Messung, typische Sensorfunktionen in automatisierten Montagesystemen.

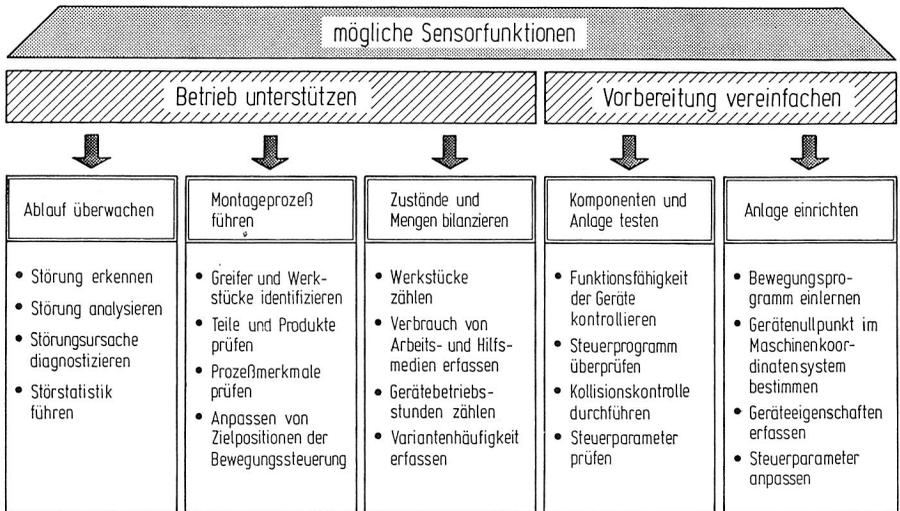
Zu den wichtigsten Sensorfunktionen gehören die verschiedenen Aspekte der Prozeßüberwachung. Zunächst müssen Abweichungen vom Normalverlauf eines Prozesses als Störungen erkannt werden; gelingt dies bereits während des Entstehens, lassen sich möglicherweise Gegenmaßnahmen einleiten, bevor ein Schaden an Werkstück oder Maschine entstehen kann. Ist der Störfall dagegen bereits eingetreten, müssen Sicherheitsabschaltungen durchgeführt und Meldungen ausgegeben

werden. Die Störungsanalyse verfolgt dann das Ziel, das ursächliche Ereignis für die Störung zu finden. Ausgehend von Modellvorstellungen über den Prozeß kann als Methode zum Auffinden von Störungsursachen die Entscheidungstabellentechnik benutzt werden [30]. Als letzter Schritt der Störungsbehandlung kann die Störungsursache diagnostiziert werden, um Hinweise für die Störungsbehebung abzuleiten. Die Störstatistik gibt schließlich Auskunft über die Anfälligkeit von Geräten und Systemteilen und stellt eine wichtige Bewertungsgrundlage bei der Auswahl neu zu beschaffender Komponenten dar. In der betrieblichen Praxis der Montage-technik stehen Störungsanalyse und Ursachendiagnostik noch am Anfang, sie sind im wesentlichen auf Steuerungskomponenten und andere elektronische Systemelemente beschränkt. Gerade die gegenseitigen Einflüsse der Komponenten des Gesamtsystems bleiben noch weitgehend unbeachtet, der Kenntnisstand über die vielfältigen Verfahren der Montage-technik ist dafür noch zu gering.

Steuerungen von Montagesystemen sind üblicherweise als Kombination von Ablaufsteuerung und numerischer Bewegungssteuerung ausgelegt. Zur Prozeßführung müssen Sensoren deshalb Weiterschaltbedingungen für die Schrittsteuerung und Positionswerte für die Bewegungssteuerung bereitstellen. Darüber hinaus kann der Einsatz von Identifikationssystemen die organisatorische Steuerung verbessern und die Einbeziehung von Prüfoperationen die Störhäufigkeit vermindern (s. Bild 5).

Die dritte Gruppe von Sensorfunktionen, die zur Aufrechterhaltung des Systembetriebs erforderlich sind, dient der Bilanzierung von Zuständen und Mengen (s. Bild 5). Das anfallende Zahlenmaterial erlaubt, die Arbeitsweise des Montagesystems und seiner Teilsysteme vergleichend zu anderen zu beurteilen, und erleichtert die Abschätzung anstehender Wartungsmaßnahmen.

Die beiden letzten Gruppen von Sensorfunktionen dienen zur Inbetriebnahme der Anlage, sie unterstützen den Test und vereinfachen das Einrichten der Anlage (s. Bild 5). Umrüst- bzw. Einrichtarbeiten fallen bei Auftragsänderungen an, ihr Umfang hängt von der Flexibilität des Montagesystems ab. Viele dieser Arbeiten sind automatisierbar oder können durch teilautomatische Einstellhilfen stark vereinfacht werden. Hierzu gehört auch die Anwenderprogrammierung von Robotern, soweit sie vor Ort vorgenommen wird. Dabei können sensorunterstützte Methoden zur Bewegungsprogrammerstellung den Programmieraufwand stark verringern (s. Kapitel 3).



**Bild 5:** Typische Sensorfunktionen in Montagesystemen

Schließlich muß zum Abschluß der Vorbereitungen, vor Beginn des Anlagenbetriebes, die Funktionstüchtigkeit der Geräte und die richtige Einstellung kritischer Geräteparameter kontrolliert sowie die Richtigkeit der Anwenderprogramme überprüft werden.

## 2.2 Ziele des Sensoreinsatzes

Sensoren und ihre Signalverarbeitung in elektronischen Steuerungen können den finanziellen Aufwand für Montagesysteme beträchtlich erhöhen. Ihr Einsatz muß sich deshalb an unternehmerischen Zielen messen lassen. Bild 6 zeigt Ziele des Sensoreinsatzes, aufgegliedert nach den drei Aspekten 'Sicherheit', 'Wirtschaftlichkeit' und 'Qualität'.

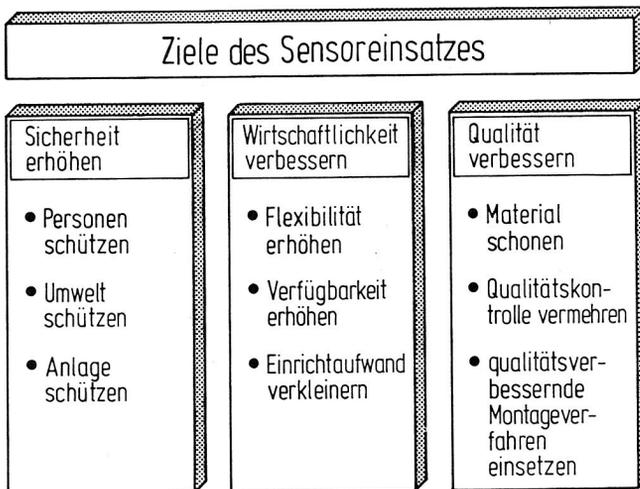


Bild 6: Ziele des Sensoreinsatzes in Montagesystemen

Unternehmen stehen im engen Beziehungsgeflecht des Marktes, wo sie ebenso ihre Produkte anbieten wie Rohstoffe, Zukaufteile und Ausrüstung beschaffen müssen. Bedingt durch starke Konkurrenz im internationalen Wettbewerb ist eine Erhöhung der Wirtschaftlichkeit produktseitig vor allem durch verbesserte Produktqualität und produktionsseitig durch Automatisierungsmaßnahmen zu erreichen. Wachsende Auflagen zum Schutz von Personen und Umwelt stellen dagegen grundsätzliche Voraussetzungen für die Produktionsgenehmigung dar.

### 2.3 Die Planung des Sensorbedarfs als Teilaspekt der Montageplanung

Die Planung automatisierter Montageanlagen stellt einen komplexen Vorgang dar; das Planungsrisiko liegt im allgemeinen deutlich höher als bei Anlagen für die Vorfertigung. Neben den Anforderungen aus der konkreten Aufgabenstellung wie Teilezahl, erforderliche Ausbringung, Umrüstaufwand usw. haben eine ganze Reihe zusätzlicher Randbedingungen Einfluß auf Konzeption und Auslegung des Montagesystems. Hierzu gehören u.a. die Qualität der Einzelteile, die Art des Anlieferungszustandes der zu montierenden Teile, die Einbindung manueller Arbeitsplätze und nicht zuletzt die erwartete Entwicklung des Produktabsatzes sowie der Variantenverteilung. Um all diesen Aspekten gerecht zu werden und das Risiko einer Fehlplanung zu verkleinern, beginnen sich Methoden zur Systematisierung der Planungsarbeiten auch in der Praxis durchzusetzen. Mit der steigenden Zahl eingesetzter Sensoren und der Einbeziehung kritischer Prüfvorgänge in den automatisierten Ablauf gewinnt die Sensorfrage dabei an Bedeutung.

#### 2.3.1 Planungsablauf

Die Vorschläge verschiedener Autoren zur systematischen Planung von Montagesystemen weisen große Ähnlichkeiten untereinander auf [31-33]. Die Planungsschemata basieren auf einer gestuften Vorgehensweise und nutzen Methoden aus der Konstruktionssystematik. Ihre Abweichungen untereinander, sieht man einmal von der Zahl der verwendeten Arbeitsschritte oder ihrer Benennung ab, betreffen letztlich Detailfragen. Der in Bild 7 skizzierte Planungsablauf zeichnet sich gegenüber anderen Vorschlägen durch eine detaillierte Beschreibung der Montageaufgabe in den Planungsschritten drei und vier aus. Die Vorteile einer Aufgabenstrukturierung (Schritt 3) für eine rechnerunterstützte Arbeitsweise werden

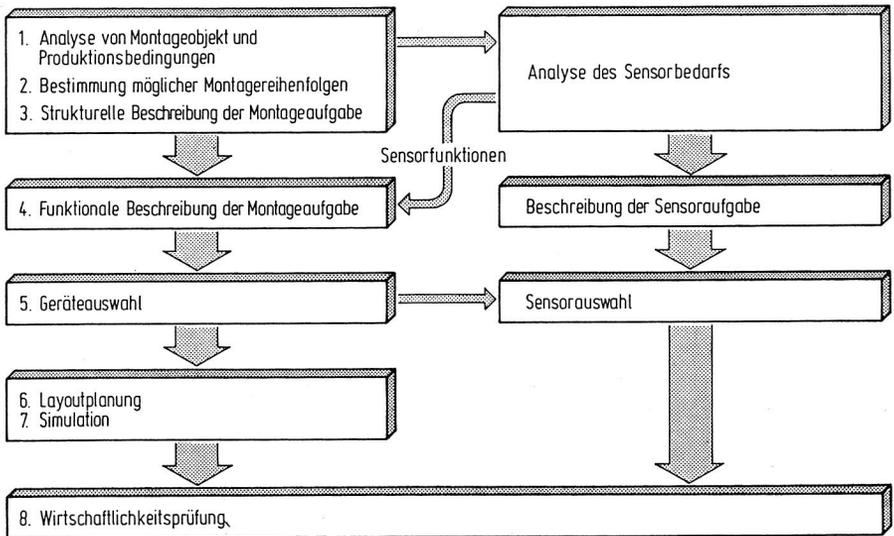
in [33] für die Montageplanung eines elektrischen Schalters behandelt, es wird dort ein formales Modell als Strukturierungsinstrument eingesetzt.

Zu Beginn der Planungsaufgabe steht die Analyse des Montageobjekts und der Produktionsbedingungen. Sie gibt Aufschluß über den Aufbau des Produktes, das montiert werden soll, und liefert die Anforderungen, die an die zu planende Montageaufgabe gestellt werden (s. Bild 7). Im nächsten Schritt wird geprüft, in welcher Reihenfolge die einzelnen Teile des Produktes zusammengesetzt werden können. Aus den verschiedenen Möglichkeiten wird eine günstige ausgewählt und als Montageablaufstruktur graphisch dargestellt (Schritt 3) [33]. Die funktionale Beschreibung der Montageaufgabe entsteht durch Einfügen der erforderlichen Handhabungs- und Montagefunktionen in diese Struktur und ist die Vorlage, anhand derer Geräte als Funktionsträger ausgewählt werden können. Ihre räumliche Anordnung hält der Layoutplan fest. Aus der Simulation der Anlage nach dem bisherigen Entwurf können Hinweise auf die Stationsaufteilung sowie auf günstige Pufferanordnung und -größe abgeleitet werden. Hält die Lösung einer Überprüfung der Wirtschaftlichkeit stand, kann die Detailkonstruktion der Elemente des Montagesystems beginnen.

### 2.3.2 Einbeziehung des Sensoraspekts in die Montageplanung

Während der ersten drei Planungsschritte (Bild 7), in denen die strukturelle Beschreibung der Montageaufgabe entsteht, wird der Bedarf an Sensoren festgelegt, soweit er bereits in der Konzeptionsphase des Montagesystems bedeutsam ist. In dieser frühen Planungsphase werden ausschließlich diejenigen Sensoraufgaben behandelt, die spürbare Auswirkungen auf das Montagekonzept haben; alle übrigen Sensoraufgaben einfachen Charakters, die später zur Verfeinerung einer Lösung beitragen, bleiben zunächst außer acht. Ergebnis der Bedarfsana-

lyse ist zum einen die Beschreibung einer Zahl von 'Sensoraufgaben', für die anschließend geeignete Geräte ausgewählt werden müssen, zum anderen fließen 'Sensorfunktionen' in die funktionale Beschreibung der Montageaufgabe ein (s. Bild 7).

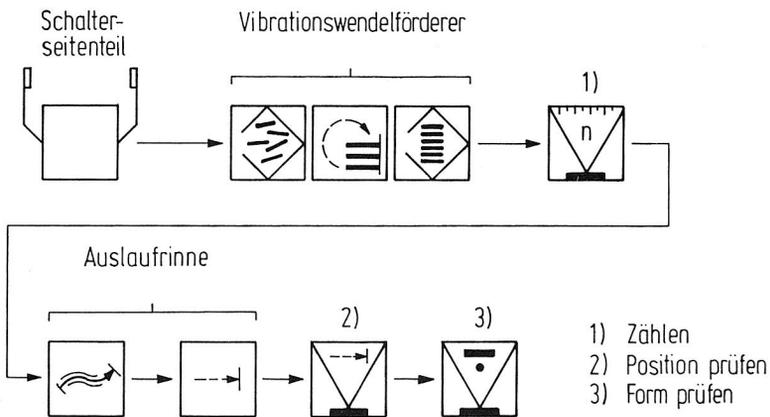


**Bild 7:** Planungsschritte für die Erstellung eines automatisierten Montagesystems

Die 'Sensoraufgabe' gibt an, was der Sensor innerhalb des Montagesystems tun soll und enthält alle Angaben, die für die Sensorauswahl unter den zu erwartenden Bedingungen des Einsatzes relevant sind. Das sind neben Angaben über das Montageobjekt und das geplante Montagesystem auch solche über die Umgebungsbedingungen. Die Sensoraufgabe ist damit die Eingangsgröße zur Sensorauswahl (s. Bild 7), welche in Abschnitt 2.4.2 verfeinert wird, und stellt die Schnittstelle zwischen der Planung der mechanischen Teilsysteme und der Planung der MSR-Einrichtungen dar. Die Abkürzung MSR steht für die Aufgabenbereiche Messen, Steuern und Regeln.

Kern der Sensoraufgabe ist die 'Sensorfunktion', sie ist von

allen konkreten Bedingungen und Realisierungshinweisen befreit und charakterisiert die vorliegende Problemstellung auf möglichst abstrakte Weise (vgl. Bild 12), damit der Auswahlvorgang nicht unnötig durch unbeabsichtigte Lösungshinweise eingeschränkt wird. Bild 7 zeigt als zweiten Ausgang der sensorischen Bedarfsanalyse die besondere Gruppe von Sensorfunktionen, die in die Montageaufgabenbeschreibung einfließt. Sie sind, wie diese Beschreibung selbst, am Materialfluß orientiert. Meist handelt es sich um Prüf- oder Kontrollfunktionen. Sie erscheinen als Funktionen der Eingangsprüfung im Bereich der Bereitstellungs-pfade oder, Fügefunktionen nachgelagert, als Funktionen der produktbegleitenden Qualitätsprüfung. Bild 8 zeigt eine solche Funktionskette aus der Teilebereitstellung. Im Bild sind mögliche Funktionsträger für die Handhabungsfunktionen angegeben. Die Symbole entstammen der VDI-Richtlinie 2860 [34]. Das Seitenteil eines elektrischen Schalters, in einer Bauform für die Schienenmontage in Schaltschränken und Steuereinrichtungen, wird als Schüttgut angeliefert, in einem Vibrationswendelförderer vereinzelt und geordnet und schließlich in einer Auslaufrinne vor einem Anschlag positioniert. Als Kontrollfunktionen sind innerhalb dieses



**Bild 8:** Funktionskette aus der Teilebereitstellung

Ablaufs das Zählen der ausgebrachten Teile (1), das Prüfen der Position am Anschlag (2) und am Schluß die Formprüfung (3) vorgesehen. Für die drei Kontrollfunktionen werden in der Regel sensorische Funktionsträger ausgewählt, insbesondere dann, wenn die Ergebnisinformation der Anlagensteuerung zugänglich sein soll.

Die übrigen Sensorfunktionen - zu ihnen gehören diejenigen, die bei der Anlagenüberwachung den engen Materialflußbezug überschreiten und solche, die spezielle Aspekte von Fügeoperationen erfassen - finden im Montagekonzept zunächst keinen Niederschlag. Trotzdem ist die Auswahl ihrer sensorischen Funktionsträger nicht unabhängig vom Fortgang der Montageplanung. Die gerätetechnischen Festlegungen des Montagekonzeptes wirken sich nämlich in Form von Randbedingungen (Bewegungsgeschwindigkeit, Kräfte, Materialien, Anordnung etc.) auf die Sensorauswahl aus.

#### 2.4 Auswahl von Sensoren

Zur Verfeinerung des Planungsvorganges bezüglich des Sensoraspekts bei automatisierten Montagesystemen (vgl. Abschnitt 2.3.2) wird im folgenden die 'Sensorauswahl' behandelt (s. Bild 7). Dazu geht Abschnitt 2.4.1 zunächst auf die Arbeitsweise von Sensoren ein. Abschnitt 2.4.2 stellt eine systematische Auswahlmethode für Sensoren vor, für die Abschnitt 2.4.3 Möglichkeiten zur unterstützenden Verwendung von Rechnern aufzeigt.

##### 2.4.1 Arbeitsweise von Sensoren

Nach der allgemeinen Funktionsdefinition aus Abschnitt 2.1.3 erfüllt ein Sensor innerhalb eines Montagesystems die Funktion der Informationserfassung aus dem Prozeß. Diese Sensorfunktion charakterisiert die vorliegende Problem-

stellung aus der Sicht des Sensoranwenders. Für den Sensorhersteller dagegen umfaßt der Sensor selbst eine Reihe von Funktionen. Neben der Hauptfunktion, der Meßwertgewinnung, können verschiedene Nebenfunktionen existieren, die die Betriebssicherheit oder die Meßgenauigkeit erhöhen. Beispielsweise steigt die Meßgenauigkeit bei der Entfernungsmessung mit Ultraschall spürbar durch eine Meßwertkorrektur nach der Lufttemperatur. Die Temperaturkompensation ist damit eine Nebenfunktion des Ultraschallsensors. Hilfsfunktionen dagegen tragen nicht zur eigentlichen Messung bei, sie können Betriebsvoraussetzungen oder Hilfen für die praktische Inbetriebnahme darstellen. Bild 9 zeigt den Funktionsumfang von Sensoren und enthält Beispiele für Hilfs- und Nebenfunktionen. Die Schwierigkeit des Auswahlvorgangs bei Sensoren liegt darin, daß die unterschiedlichen Sichtweisen von Sensorhersteller und Sensoranwender möglichst gut zur Deckung gebracht werden müssen.

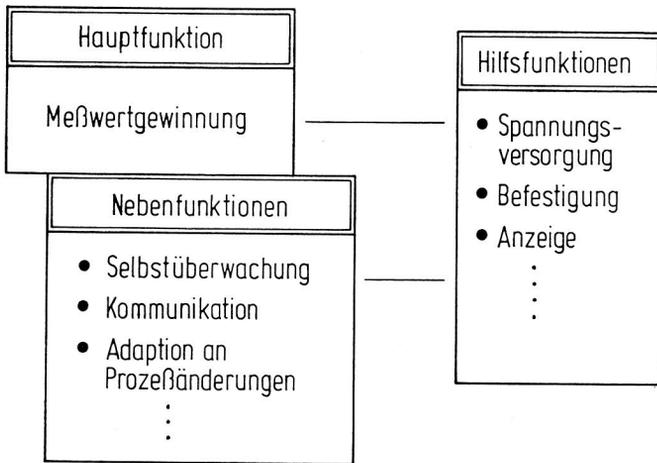


Bild 9: Funktionsumfang von Sensoren

Für die weitere Behandlung soll im folgenden Exkurs ein Prozeßbegriff eingeführt werden: Ganz allgemein wird ein 'Prozeß' als eine dynamische Aufeinanderfolge von Zuständen

eines Objekts betrachtet. Prozesse bestehen aus einer Folge von Vorgängen, die die Änderungen des Objektzustandes bewirken. Folglich besteht der Montageprozeß aus einer solchen Abfolge von Vorgängen bzw. Operationen, die zum Zusammensetzen von Teilen zu größeren Einheiten notwendig sind. Zur Kontrolle und Lenkung von Prozessen aller Art müssen zunächst die Objektzustände erfaßt werden, um in Verbindung mit der Kenntnis über den gewünschten Änderungsverlauf die Grundlage zur Prozeßbeeinflussung zu bilden. In technischen Prozessen, deren Objekte Stoffe und Energien beinhalten, dienen Meßeinrichtungen bzw. Sensoren dazu, die Zustände in Form charakteristischer Größen zu erfassen. Anschließend wandeln sie diese Größen und liefern die gesuchte Zustandsinformation in einer zur Weiterverarbeitung geeigneten Art.

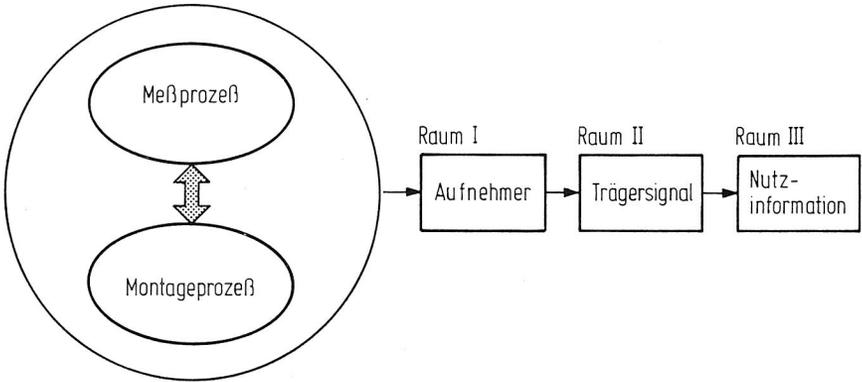
In systemtechnischer Betrachtung findet der eigentliche Produktionsprozeß in einem System 'Fabrik' statt. Wie ein solches System schrittweise in Teilsysteme zergliedert werden kann, läßt sich auch ein Gesamtprozeß in entsprechender Weise in Teilprozesse zerlegen. Benachbarte Teilprozesse stehen im Regelfall in Wechselwirkung miteinander. Das gilt auch für einen Meßprozeß, der die Gewinnung charakteristischer Kenngrößen aus einem Montageprozeß zur Aufgabe hat. Gegenständlich betrachtet existiert zwischen dem Meßobjekt, also dem Objekt, auf das sich die Messung bezieht, und anderen Elementen aus dem Montagesystem sowie der Meßeinrichtung eine Vielzahl gegenseitiger Einflüsse. Wirken sie unerwünscht auf die Meßeinrichtung ein, werden sie als Störungen, wirken sie dagegen in umgekehrter Richtung von der Meßeinrichtung auf das Meßobjekt ein, so werden sie als Rückwirkungen bezeichnet. Aus dem Montageprozeß lassen sich einzelne Prozeßgrößen aufnehmen, um Erkenntnisse über einen gefragten Zusammenhang abzuleiten, wobei verschiedene Prozeßgrößen den gesuchten Sachverhalt verschieden gut widerspiegeln können. Bietet sich dennoch keine hinreichend geeignete Prozeßgröße an, oder soll der Sensor auch in abge-

wandelter Umgebung vergleichbare Ergebnisse liefern, kann es notwendig werden, im Meßprozeß gewisse Hilfsgrößen zu erzeugen und die Veränderung dieser Größen in der Wechselwirkung mit dem Montageprozeß für die Messung auszuwerten. Die Hilfsgrößen zeichnen sich dadurch aus, daß ihre Veränderung die gesuchte Information so gut repräsentiert, daß die Auswertung im Meßprozeß möglich ist. Zur Entfernungsmessung nach dem Triangulationsverfahren wird beispielsweise eine Hilfsgröße in Gestalt eines gebündelten Lichtstrahls, ausgesendet zum Meßobjekt, verwendet, während bei Kraftmessungen mit Dehnungsmeßstreifen-aufnehmern die Dehnung des Werkstücks direkt in Form der Meßstreifendehnung erfaßt wird. Im letzten Fall entfällt eine Hilfsgröße. Nach dieser Überlegung kann man die Aufnehmer der Sensoren als passiv bezeichnen, die die Energie zur Signalerzeugung primär dem Montageprozeß entnehmen und diejenigen als aktiv, die die Energie primär aus dem Meßprozeß beziehen.

HOFMANN [35] unterscheidet zwar ebenfalls zwischen aktiv und passiv arbeitenden Aufnehmern, benutzt als Unterscheidungsmerkmal aber die benötigte Hilfsenergie. Zur Beurteilung der durch die Messung bewirkten Beeinflussung des Montageprozesses ist die obige Einteilung allerdings ergiebiger; beide Klassifikationen haben große Ähnlichkeit, sind aber nicht deckungsgleich.

Die Informationsgewinnung durch selektives Erfassen von Prozeßgrößen mit Sensoren kann als mehrstufiger Abbildungsvorgang betrachtet werden. Jeder Abbildungsvorgang hat die Aufgabe, gezielt Informationen aus dem logisch vorangehenden Raum herauszufiltern und in einer für den jeweiligen Raum typischen Form darzustellen. Bild 10 zeigt diesen Vorgang, aufgeteilt in drei Stufen. Der erste Abbildungsraum beinhaltet den Aufnehmer, dessen geeignete Auswahl einen besonders maßgeblichen Einfluß auf die Qualität der Informationsselektion des gesamten Abbildungsvorgangs hat. Die geforderten Aufnehmermerkmale und -eigenschaften werden festgelegt durch

die Eigenarten des Montageprozesses unter dem Einfluß störender Umgebungsbedingungen und durch Art und Ausprägung der gesuchten Prozeßinformation. Der zweite Abbildungsraum trägt dem Entwicklungsstand der Mikroelektronik insofern Rechnung, als die meisten Sensoren in irgendeiner Form zumindest Teile der Signalbearbeitung elektrisch durchführen. Das Träger-signal ist damit innerhalb des Gesamtvorgangs diejenige charakteristische Zwischengröße, die die gesuchte Information elektrisch repräsentiert. Diesem Signal können verschiedene nichtelektrische Umwandlungsvorgänge vorgela-



Sensortyp	Abbildungsraum		
	I	II	III
DMS - Kraftsensor	DMS - Streifen	frequ.-mod. Signal	Kraftwert
Bilderkennungssystem	CCD - Array	Videosignal	Mustermerkmale
Triangulationssensor	PSD	Stromsignal	Entfernungswert
induktiver Näherungsschalter	Spule	Frequenzsignal	Anwesenheit

→ Abbildung  
 ↔ Wechselwirkung

**Bild 10:** Abbildungsvorgänge bei der Informationsgewinnung aus dem Prozeß

gert sein. Der dritte Abbildungsraum enthält die gesuchte Nutzinformation in der gewünschten Darstellung, beispielsweise als Strom- oder Spannungseinheitssignal, als Datei festen Formats oder als Druckerscrieb. Im unteren Teil von Bild 10 sind beispielhaft Sensoren und ihre technischen Realisierungsformen in den drei Abbildungsräumen aufgeführt.

Jeder der genannten Abbildungsvorgänge bewirkt einen Informationsverlust. Soweit selektiv die gesuchte Information ausgefiltert werden kann, ist dieser Verlust erwünscht, andernfalls unerwünscht. Der Verlust kann rein quantitativ sein bei technisch bedingter Begrenzung eines merkmalttragenden Signals oder der Verschlechterung des Nutz-/Störsignal-Verhältnisses durch Rauscheinwirkung, er kann aber auch qualitativer Art sein. Qualitative Informationsverluste treten beispielsweise bei der Abbildung einer räumlichen Szene auf einem zweidimensionalen CCD-Array einer Halbleiterkamera oder beim Wechsel der Darstellungsart in Signalen auf: analog - diskret, kontinuierlich - diskontinuierlich.

Die bei der Beobachtung von Montageprozessen erfassbaren Merkmale lassen sich entweder dem Montageobjekt, also Einzelteilen oder Baugruppen, dem Montageprozeß oder der Montageumwelt zuordnen (Bild 11). Ausgehend von einer Ansammlung von Einzelteilen oder vorgefertigten Baugruppen entsteht in den einzelnen Stationen des Montagesystems Schritt für Schritt das fertige Produkt. Beim Durchlaufen der Montageschritte wechselt die vorliegende, schrittweise wachsende Teilegruppe mit ihrer Gestalt auch ihre Beurteilungsmerkmale. Zur Prüfung der Qualität einer Teilegruppe reicht die Messung von Grundmerkmalen (z.B. Maße und Gewichte) allein nicht mehr aus, zusätzlich müssen Merkmale wie Funktionsfähigkeit, Spiel etc. erfaßt werden, die charakteristisch für das entstehende Produkt sind. Die Merkmale des Montageprozesses selbst variieren je nach Art der ausgeführten Operation. Prozeß und Objekt sind gleichermaßen dem Einfluß der Umwelt ausgesetzt. Umwelterschei-

nungen, die als kritisch bekannt sind, werden anhand festgelegter Merkmale überwacht, um negative Auswirkungen auf das Produkt möglichst klein zu halten.

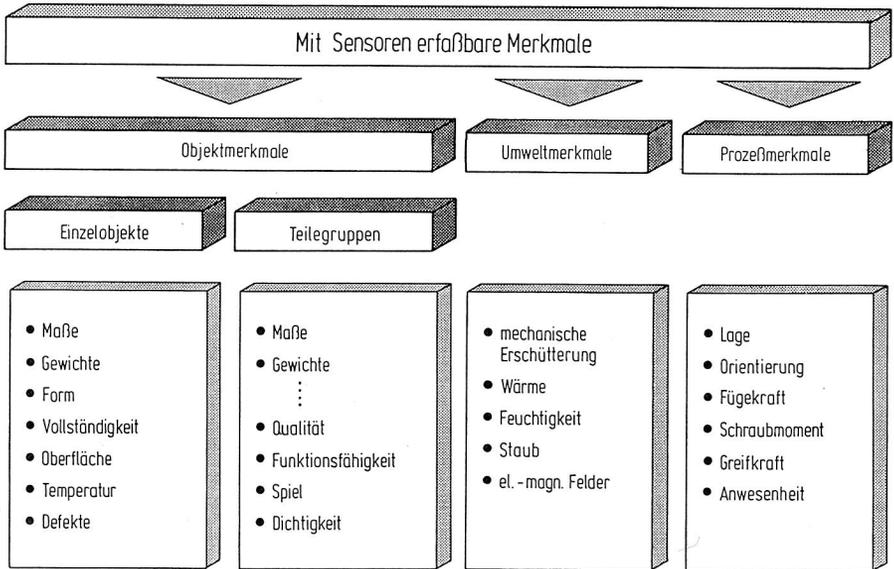


Bild 11: Mit Sensoren erfassbare Merkmale bei der Montage

#### 2.4.2 Systematische Sensorauswahl

Eine systematische Methode zur Sensorauswahl stellen EVERSHEIM und HAUSMANN in [13,14] vor. Ausgehend von der Montageaufgabe entwickeln die Autoren in mehreren Schritten eine 'Überwachungsstrategie', aus der sie ein Anforderungsprofil für die einzusetzende Sensorik ableiten. Ein Vergleich des Anforderungsprofils mit dem verfügbaren Sensorangebot führt auf geeignete Lösungen, die später einer wirtschaftlichen Bewertung unterzogen werden. Der eigentliche Auswahlvorgang zerfällt dabei in zwei Schritte. Zunächst wird das Sensorangebot in einer 'Grobauswahl' herstellernerneu-

tral untersucht, wobei die Sensoren hinsichtlich ihres durchschnittlichen Leistungsvermögens gekennzeichnet werden. Ist auf diese Weise die 'Sensorart' festgelegt, werden in einer 'Feinauswahl' herstellerspezifisch geeignete Sensoren bestimmt. Einen gänzlich anderen Weg für die gleiche Aufgabe deutet DOLL in [36] an, er schlägt einen Expertensystemansatz vor. Als Datenbasis bei einer Prototypimplementierung verwendet er eine herstellerneutrale Beschreibung der Sensoren und läßt den Benutzer über eine allgemeine Beschreibung der Meßaufgabe auf die äußere Schale des Expertensystems zugreifen. Durch die Beantwortung einiger Übersichtsfragen sowie weiterer Fragen zu Meß- und Umgebungsbedingungen soll der Benutzer den Lösungsraum systematisch eingeschränken können.

Die recht kurzgefaßten Ausführungen des Autors [36] erlauben keine Einschätzung für die praktische Eignung des vorgeschlagenen Expertensystemansatzes. Da aber bislang kaum allgemeingültige Zusammenhänge zwischen Sensoraufgaben und Sensoreigenschaften bekannt sind, und das Spektrum von Sensorkonzepten sehr groß ist, dürften erhebliche Schwierigkeiten bei der Erstellung eines ausreichenden Regelwerkes zu erwarten sein. Im weiteren wird deshalb eine neue Auswahlmethode vorgestellt, die sich gut in den gewöhnlichen Planungsablauf für Montagesysteme integrieren läßt und den Erfahrungsschatz des Montageplaners nutzt. Die Methode ist für die rechnerunterstützte Arbeitsweise geeignet und ausgelegt für Einfachsensoren, die Messungen im Sinne von DIN 1319 [9] ausführen. Der Hauptunterschied zu der oben vorgestellten Methode von EVERSHEIM und HAUSMANN [13,14] liegt in der durchgängigen Herstellerneutralität des Auswahlvorgangs. Besonders verfeinert sind deshalb die einzelnen Schritte bis zur Festlegung des Meßkonzepts (s. Bild 12), nach dem der gesuchte Sensor arbeiten soll. Das in Bild 12 gezeigte schrittweise Vorgehen führt systematisch zu einer abgewogenen Lösung und verleiht große Sicherheit, die wichtigsten Aspekte berücksichtigt zu haben.

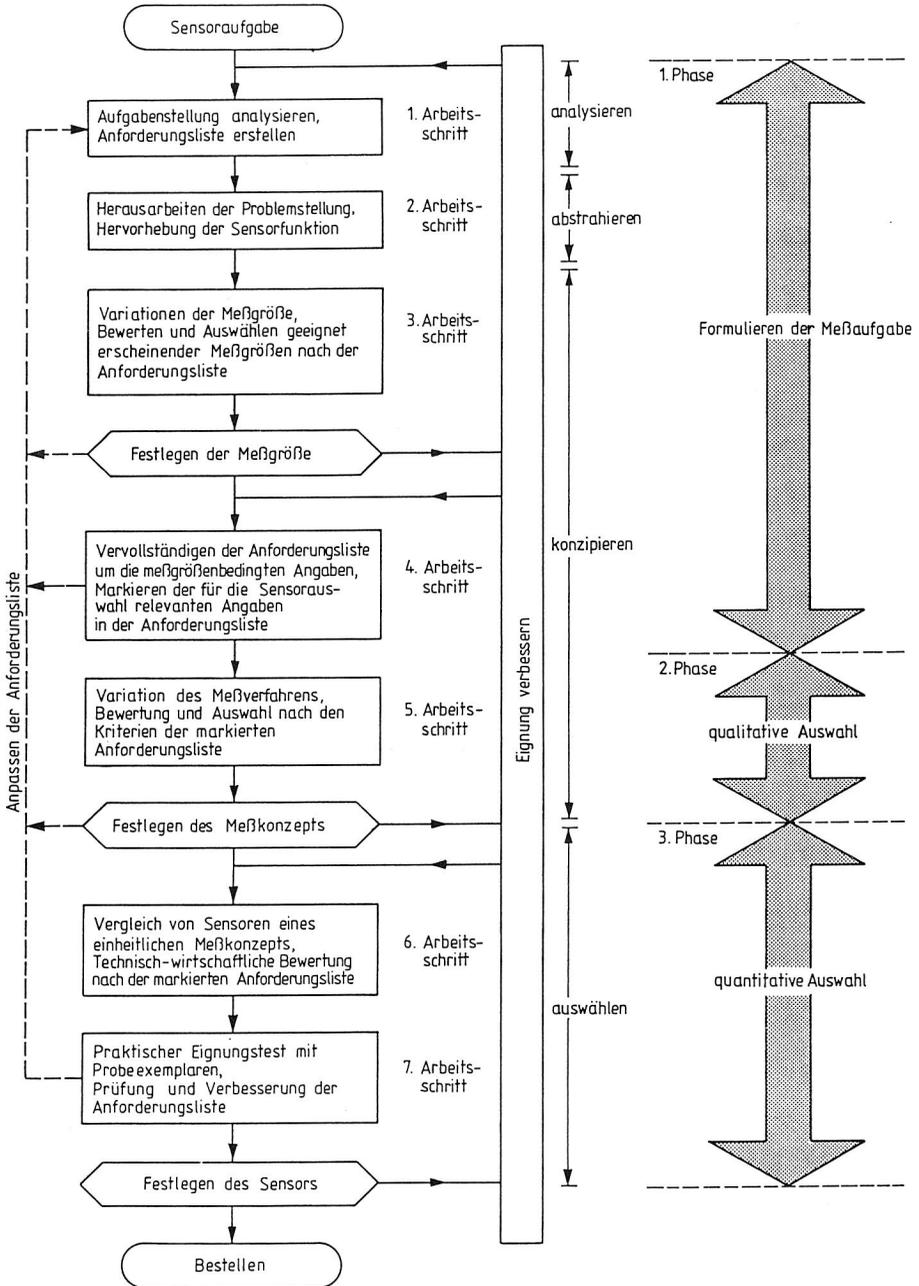


Bild 12: Vorgehensweise bei der Sensorauswahl

Der Auswahlalgorithmus (Bild 12) benötigt die 'Sensoraufgabe' (s. Abschnitt 2.3.2) als Eingangsinformation und liefert als Ergebnis die Sensorbestellung; der Auswahlvorgang ist in die drei Phasen <Formulieren der Meßaufgabe>, <qualitative Auswahl> und <quantitative Auswahl> gegliedert. Besondere Bedeutung kommt dem Formulieren der Meßaufgabe zu. Ihre wesentlichen Merkmale sind die Meßgröße und die Eigenschaften, die der Sensor aufzuweisen hat. Methodisch umfaßt die erste Phase die Festlegung der Meßgröße und die Quantifizierung meßgrößenbezogener Kenngrößen, die die Basis für die nachfolgenden Auswahlvorgänge bilden. Diese Phase kann interpretiert werden als Übersetzung der Sensoraufgabe, die sich begrifflich an der Montageaufgabe und dem zugehörigen Montagesystem orientiert, in die Meßaufgabe, die im Begriffssystem der Meßtechnik formuliert ist. Die Notwendigkeit dieser Übersetzung stellt eine Grundproblematik der Betriebsmeßtechnik dar [28], da sie zwischen den anwendenden Fachdisziplinen und der Meßgeräte herstellenden Fachdisziplin angeordnet ist, die jeweils eigene Fachsprachen entwickelt haben. In der zweiten Phase, der <qualitativen Auswahl>, fällt die Entscheidung über die Art des Sensors bezüglich seines Meßkonzepts, während die dritte Phase, die <quantitative Auswahl>, der wertenden Gegenüberstellung gleichartiger Sensoren dient. Der Vergleich wird anhand relevanter technischer Daten vollzogen.

Das Schema in Bild 12 ist unterteilt in eine Folge von Arbeits- und Entscheidungsschritten. Jeder Entscheidungsschritt bestimmt den Fortgang oder die Wiederholung des letzten oder weiter davor liegender Arbeitsschritte. Zweck der Wiederholung von Arbeitsschritten ist zum einen die Verbesserung der Eignung, zum anderen die Sicherstellung, daß tatsächlich ein Sensor ausgewählt werden kann. Da das Unternehmen, das das Montagesystem einsetzen will, in der Regel weder Kapazität zur Entwicklung noch zur Herstellung spezieller Sensoren hat, muß das Ergebnis der Auswahl ein käufliches Produkt sein. Dabei kann es vorkommen, daß zu

einem gewählten günstigen Meßkonzept das Marktangebot an Sensoren einfach zu klein ist, als daß auch nur ein Exemplar existiert, das der Anforderungsliste genügt. In diesem Fall ist entweder die Anforderungsliste anzupassen oder ein anderes Meßkonzept zu wählen.

Da die Sensorfrage aus Sicht der Montageplanung bei der Diskussion alternativer Lösungskonzepte eine Nebenrolle einnimmt, wird die Sensoraufgabe als Eingangsgröße des Auswahlvorgangs meist unvollständig vorliegen. Im ersten Arbeitsschritt muß also zunächst die Aufgabenstellung geklärt und die Information zusammengetragen werden, die noch fehlt, um den Sensor festlegen zu können. Ein gutes Hilfsmittel zur Zusammenstellung der Angaben in einer Ordnung, die die Durchführung der nachfolgenden Arbeits- und Entscheidungsschritte erleichtert, stellt eine Anforderungsliste dar. Zur Unterstützung bei der Erstellung können Listen von Kenngrößen verwendet werden. Die einzelnen Angaben sollten klar

<u>Merkmale</u>	<u>Beispiele</u>
Geometrie	Abmaße, Anordnung (Montagesystem, Montageobjekt)
Kinematik	Bewegungsart, Geschwindigkeit
Kräfte	Steifigkeit, Federeigenschaften, Verformung, Gewicht
Umgebung	Temperatur, Staub, Störfelder
Signal	Ausgangsgröße, Anzeige
Sensor	größte zulässige Abmessungen, Schutzart, geforderte Prüfzeichen, Montagevorschriften, Anschlußart der Energieversorgung
Meßobjekt	Oberfläche, Material, Zugänglichkeit, Empfindlichkeit
Betrieb	Wartungsfreiheit oder -Intervalle, besondere Sicherheitsvorschriften (Laser, Radioaktivität)
Kosten	max. Bezugskosten
Termin	Lieferzeit
Meßgröße	Meßbereich, zul. Toleranz, Meßunsicherheit, dyn. Verhalten, Einfluß- und Störgrößen

Bild 13: Kenngrößen zum Erstellen einer Anforderungsliste

unterschieden werden in Forderungen, deren Erfüllung zwingende Voraussetzung der Zulässigkeit einer Lösung ist, und Wünsche, die nach Möglichkeit berücksichtigt werden sollten und bei der Bewertung der späteren Lösungen ihren Niederschlag finden. Bild 13 enthält solche Kenngrößen und erläutert durch einige Angaben, welchen Anforderungstyp sie umfassen sollen. Sie stellen eine Art Gedächtnisstütze dar, um bei der Informationssammlung möglichst keine wichtigen Aspekte unberücksichtigt zu lassen. Anforderungslisten sind übliche Hilfsmittel aus der Konstruktionspraxis; Pahl gibt für verschiedene Branchen in [37] einige Beispiele aus der industriellen Praxis an.

Nach der Aufgabenanalyse sind drei weitere Arbeitsschritte nötig, um die endgültige Meßaufgabe zu erhalten. Häufig enthält eine Aufgabenstellung unbeabsichtigte Lösungshinweise, die mögliche günstige Lösungsvarianten ungeprüft bereits im Vorgriff ausschließen können, sofern es nicht gelingt, echte von scheinbaren Einschränkungen zu unterscheiden. Obwohl Meßgrößen in der Montagetechnik nur selten explizit gesucht werden, tauchen sie doch nicht selten in der Aufgabenstellung auf. Sie verdanken ihre Nennung meist mehr der Erinnerung an frühere Lösungswege als der detaillierten Problemanalyse. Der zweite Arbeitsschritt soll deshalb den Kern der Aufgabe herausarbeiten, die Sensorfunktion. Dazu muß von allen Bedingungen und konkreten Realisierungsmöglichkeiten abstrahiert und die Aufgabenstellung soweit wie möglich vereinfacht werden. Damit ist die höchste Stufe der Abstraktion erreicht, alle nachfolgenden Arbeitsschritte sind gekennzeichnet durch den Übergang vom 'Abstrakten zum Konkreten'.

Sollte noch keine Meßgröße als festgelegte Eingangsforderung vorliegen, ist deren Auswahl unter Berücksichtigung vorliegender Einschränkungen Inhalt des dritten Arbeitsschritts. Bei komplizierten Montageprozessen kann es sehr schwierig sein, eine Meßgröße zu finden, die eine gesuchte Prozeßer-

scheinung genügend gut wiedergibt. Erst recht bedarf es großer meßtechnischer Erfahrung und genauer Prozeßkenntnis, um aus alternativen Meßgrößen diejenige herauszufinden, die der Anforderungsliste auf bestmögliche Weise gerecht wird. Ein Beispiel aus der Montagetechnik soll die Problematik verdeutlichen: Die Montageaufgabe besteht darin, zwei Metallplatten kraftschlüssig miteinander zu verbinden. Zur Erzeugung der Verbindung soll das Fügeverfahren 'Schrauben' dienen. Nach ROTH [38] werden die haftschlüssigen Verbindungen in die Gruppe der kraftschlüssigen und die Untergruppe der reibschlüssigen Verbindungen eingeordnet, d.h. die Kräfte, die den Zusammenhalt der Bauelemente bewirken, werden durch Reibung erzeugt. Der Schraubprozeß soll automatisch beendet werden, wenn der Zusammenhalt der Verbindung ein gefordertes Maß erreicht hat.

Die Belastung, der die Verbindung standhalten soll, tritt erst beim späteren Gebrauch des Produktes auf, wenn Zug- oder Druckkräfte an den Metallplatten angreifen (Bild 14). Da diese Kräfte die reibungslose Montage behindern würden, wenn sie bereits während des Fügevorganges wirksam wären, fixieren Vorrichtungen die beiden Platten in einer geeigneten Relativposition. In dieser Fixierung werden aber die Reibkräfte selbst oder andere eng verwandte Größen nicht wirksam und kommen deshalb als Meßgrößen während des Schraubvorganges nicht in Betracht. Es liegt nahe, ersatz-

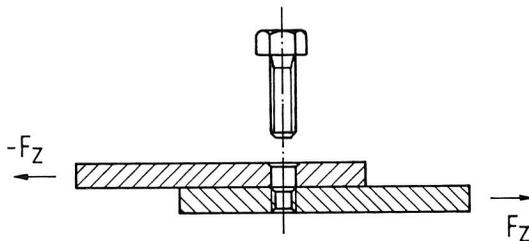


Bild 14: Belastung von Schraubverbindungen

weise die Vorspannkraft zu messen, mit der die Schraube die Platten aufeinanderpreßt, und den Zusammenhang zur Reibkraft über empirisch bestimmte, relativ gut reproduzierbare Beziehungen herzustellen. Eine direkte Messung dieser Ersatzgröße scheidet allerdings wegen mangelnder Zugänglichkeit aus, da der Meßaufnehmer in der Schraube selbst oder zwischen den Platten untergebracht werden müßte.

Erfahrungsgemäß bereitet die Automatisierung des Schraubens große Probleme. Neben anderem sind diese schlechten Erfahrungen in Empfehlungen an die Produktkonstruktion eingeflossen, Schraubverbindungen, wenn möglich, zu vermeiden. Diese Verbindungsart weist aber bezüglich der Vielfalt ihrer Belastbarkeit so erhebliche Vorzüge gegenüber anderen Verbindungsarten auf, daß Schraubverbindungen auch nach neueren Untersuchungen [39] noch immer 44% aller Verbindungen ausmachen. Die Breite des Spektrums unterschiedlicher Schraubfälle mit sehr verschiedenen Anforderungen an automatische Überwachungs- und Steuerungseinrichtungen, sowie ausgeprägte Schwierigkeiten bei der praktischen Meßgrößenerfassung haben in der Vergangenheit zur Entwicklung mehrerer verschiedener Anzugsverfahren geführt. Automatische Schraubgeräte sind heute in der Regel für zweistufige Anzugsverfahren ausgelegt. Zunächst wird mit hoher Drehzahl bis zur Kopflage eingeschraubt und anschließend mit hohem Moment angezogen. Zur Zeit sind drei Anzugsverfahren in der Praxis zu finden, die mit verschiedenen Meßgrößen arbeiten [40]:

- Das drehmomentgesteuerte Anziehen ist das einfachste und gleichzeitig ungenaueste Verfahren. Es verwendet das Drehmoment als Meßgröße.
- Das streckgrenzengesteuerte Anziehen gilt als genaueste Methode zur Montage von Schraubverbindungen und benutzt die Elastizitätsänderung der Schraube als Meßgröße. Sie wird indirekt über Drehmoment und Drehwinkel erfaßt.

- Das drehwinkelgesteuerte Anziehen greift auf die Längung der Schraube als Meßgröße zu. Sie wird indirekt mit Hilfe des Drehwinkels aufgenommen.

Im Auswahlschema (Bild 12) schließt sich nach dem Abschluß der Überlegungen zur Meßgrößenauswahl, dem dritten Arbeitsschritt, der erste Entscheidungsschritt an. Von den in Frage kommenden Meßgrößen müssen eine oder mehrere festgelegt werden, die im weiteren als verbindlich gelten. Der vierte Arbeitsschritt greift die Erkenntnisse aus der Meßgrößenauswahl auf und leitet Angaben zur Ergänzung der Anforderungsliste ab, die im Zusammenhang mit der Meßgröße stehen (s. Bild 13). Solche Angaben wie Meßbereich, Meßunsicherheit etc., die für die Sensorauswahl das Fundament darstellen, sind praktisches Ergebnis des Übersetzungsvorgangs von der Sensoraufgabe zur Meßaufgabe.

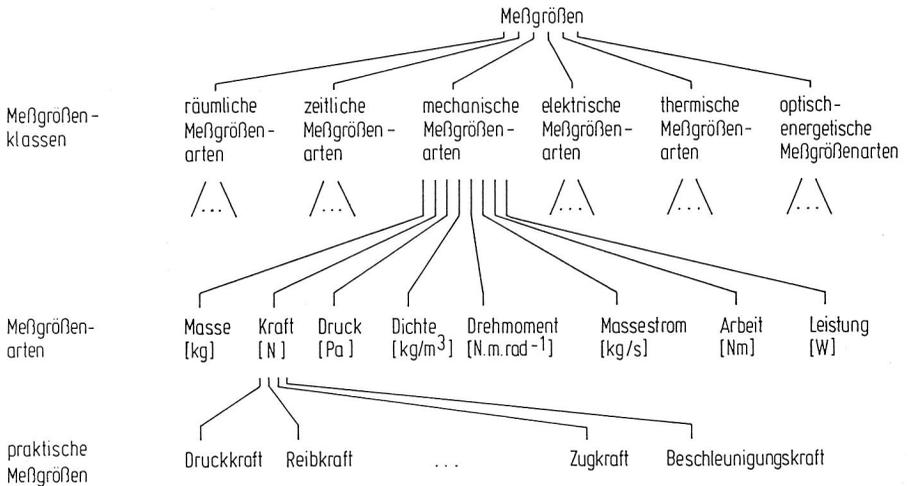
Nicht alle Angaben, die sich im Zuge der vorangegangenen Arbeitsschritte in der Anforderungsliste gesammelt haben, werden in den beiden folgenden Auswahlphasen tatsächlich gebraucht. Um die Übersichtlichkeit zu erhöhen, sollten die relevanten Angaben herausgesucht und markiert werden.

Der fünfte Arbeitsschritt führt im Ergebnis auf den zweiten Entscheidungsschritt, der das Meßkonzept festlegt, nach dem der Sensor arbeiten soll. Zum Meßkonzept gehören im Kern die Meßgröße, das Meßprinzip und das Meßverfahren. Als Meßprinzip bezeichnet man diejenige charakteristische physikalische Erscheinung, die der Messung zugrunde liegt. Beispielsweise kann für die Meßgröße 'Länge' die Interferenz des Lichtes und für die 'Kraft' die elastische Verformung als Meßprinzip dienen. Das Meßprinzip repräsentiert die physikalische Beziehung zwischen Meßobjekt und Sensor und ist deshalb besonders gut als Unterscheidungsmerkmal hinsichtlich der Verwendbarkeit für spezielle Meßaufgaben geeignet. Die praktische Anwendung eines Meßprinzips führt auf ein Meßverfahren,

wobei gewöhnlich zwischen 'direkten' und 'indirekten' Meßverfahren unterschieden wird. Die direkten Meßverfahren gewinnen den gesuchten Meßwert einer Meßgröße durch direkten Vergleich mit einem Bezugswert derselben Meßgröße, während die indirekten Meßverfahren den Meßwert auf andersartige physikalische Größen zurückführen [9]. Kann für die gesuchte Meßgröße kein geeignetes Meßprinzip gefunden werden oder ist die Meßgröße nicht zugänglich, wie im zuvor behandelten Schraubbeispiel, muß auf ein indirektes Meßverfahren ausgewichen werden. In diesem Fall sind eine oder mehrere Ersatzgrößen auszuwählen, für die Meßprinzipien zur Verfügung stehen. In der Praxis kann das Meßproblem damit die Auswahl mehrerer Sensoren erforderlich machen. Die Entscheidung für ein indirektes Meßverfahren führt im Planungsschema zurück auf den dritten Arbeitsschritt, auszuwählen sind jetzt geeignete Ersatzgrößen.

Zur Verbesserung der Übersicht empfiehlt es sich, mit einem Klassifikationsschema der bekannten Meßprinzipien zu arbeiten und den Suchbereich systematisch durch Übergänge zu Stufen geringeren Klassifikationsumfangs einzuengen. Da durch den dritten Arbeitsschritt bereits die Meßgröße festgelegt ist, liegt es nahe, bei der Klassifikation der Meßprinzipien auf der höchsten Ebene den Bezug zu den Meßgrößen herzustellen. Um überschaubare Zuordnungen zu erhalten, wurde auch für die Meßgrößen eine Klassifikation durchgeführt. Orientiert am internationalen Einheitensystem, das sechs Bezugsgrößen festlegt, sind sechs Klassen für die Meßgrößen vorgesehen (vgl. Bild 15). Jede Meßgrößenklasse enthält auf der nächsttieferen Klassifikationsebene eine Reihe von Meßgrößenarten, zu denen jeweils eine Basisgrößenart gehört. Die Meßgrößenarten sind Oberbegriffe für die praktischen Meßgrößen der untersten Klassifikationsebene, die begrifflich an der praktischen Aufgabenstellung orientiert sind. So gehören die praktischen Meßgrößen 'Druckkraft', 'Reibkraft', 'Zugkraft' und 'Beschleunigungskraft' zur gemeinsamen Meßgrößenart 'Kraft', die ihrerseits der

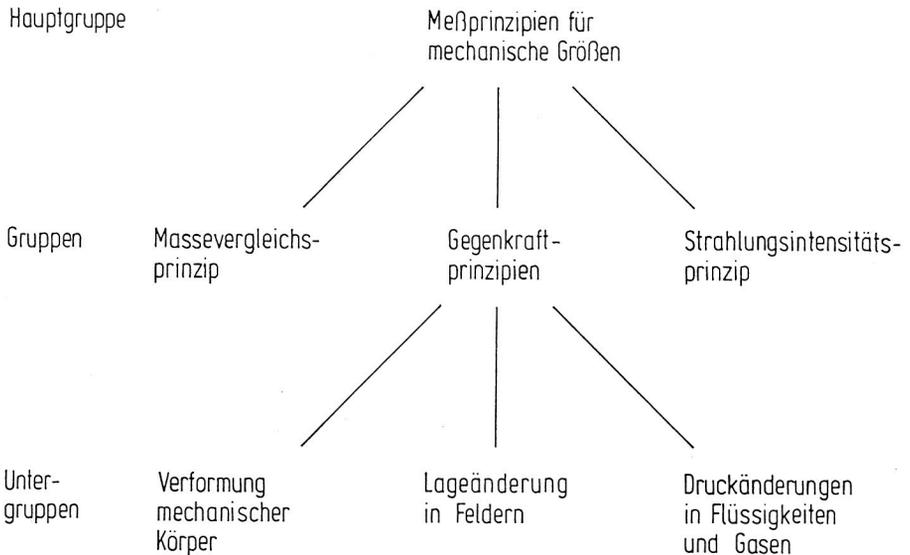
Klasse der mechanischen Meßgrößenarten angehört. Bild 15 zeigt die Klassifikation der Meßgrößen am Beispiel der mechanischen Meßgrößenarten.



**Bild 15:** Klassifikation von Meßgrößen

Zu jeder Klasse von Meßgrößen läßt sich nun eine Klasse von Meßprinzipien (Hauptgruppe) angeben, die über mehrere untergeordnete Klassifikationsebenen die geeigneten Meßprinzipien umfaßt. Für die mechanischen Meßgrößenarten entsteht damit die Klassifikation von Bild 16, die sich an einen Vorschlag von HOFMANN [35] anlehnt. Für die übrigen Meßgrößenklassen lassen sich ähnliche Klassifikationen mit einer wechselnden Zahl von Klassifikationsebenen angeben.

Da Sensoren fast ausschließlich elektrische Ausgangssignale liefern, muß an irgendeiner Stelle der internen Meßkette ein elektrisches Signal gewonnen werden. Um die Herstellungskosten niedrig zu halten und kleine Baugrößen zu erreichen, werden deshalb bevorzugt elektrische Meßprinzipien eingesetzt, und zwar auch dann, wenn mechanische oder optische genauer sind.



**Bild 16:** Klassifikation von Meßprinzipien für mechanische Meßgrößen

Im sechsten Arbeitsschritt (s. Bild 12) findet die eigentliche Auswahl der Sensoren verschiedener Hersteller statt. Verglichen werden Geräte der Meßkonzepte, die in den vorangegangenen Schritten ausgewählt worden sind. Bewertung und Auswahl erfolgt nach technisch-wirtschaftlichen Kriterien anhand der markierten Anforderungsliste. Als Methode kommt beispielsweise die Nutzwertanalyse [41] oder die technisch-wirtschaftliche Bewertung nach VDI-Richtlinie 2225 [42] in Betracht.

Der siebte und letzte Arbeitsschritt sieht einen praktischen Eignungstest vor. Bis zu diesem Zeitpunkt wurde der Auswahlvorgang in der Hauptsache vom Erfahrungsschatz des Planers getragen. Da Messungen oft durch unvorhergesehene Dinge beeinträchtigt werden und Meßbedingungen selten vergleichbar sind, dient dieser Schritt zur Bestätigung der Wahl. Gleichzeitig erweist sich damit die Anforderungsliste nachträglich als ausreichend und treffend. Sie kann archiviert in späte-

ren ähnlich gelagerten Fällen eine wertvolle Hilfe darstellen. Abschließend kann der Sensor in seiner spezifischen Bauform nun festgelegt und bestellt werden. Zeigt sich jedoch, daß die Messung mit dem ausgewählten Sensor Schwierigkeiten bereitet, muß die Anforderungsliste ergänzt und in einen voranstehenden Arbeitsschritt zurückgegangen werden.

### 2.4.3 Rechnerunterstützung bei der Sensorauswahl

Die Systematisierung des Auswahlvorganges nach dem Schema des vorangegangenen Abschnitts ist Voraussetzung für die unterstützende Verwendung von Rechnern. Durch programmtechnische Umsetzung des Auswahlalgorithmus und Einbindung einer Sensordatenbank als herstellerunabhängige Datenbasis entsteht ein Planungsinstrument für den Sensoraspekt in Montagesystemen. Die grundsätzliche Arbeitsweise eines solchen Auswahlsystems skizziert Bild 18.

Bei der Beschränkung auf Sensoren des Massenbedarfs reicht für die Darstellung einer Meßkette eine Unterteilung in drei Elemente aus. Ein Sensor setzt sich demnach zusammen (Bild 17) aus den Elementen 'Informationsaufnahme', 'Informationsübertragung' und 'Informationsauswertung'. Das Übertragungselement ist oft nur ein Spezialkabel, das die Verbindung zwischen dem prozeßnah installierten Aufnahmeelement und dem Auswerteelement herstellt. In jedem Fall legt der Sensorhersteller die Zusammengehörigkeit der Einzelelemente durch



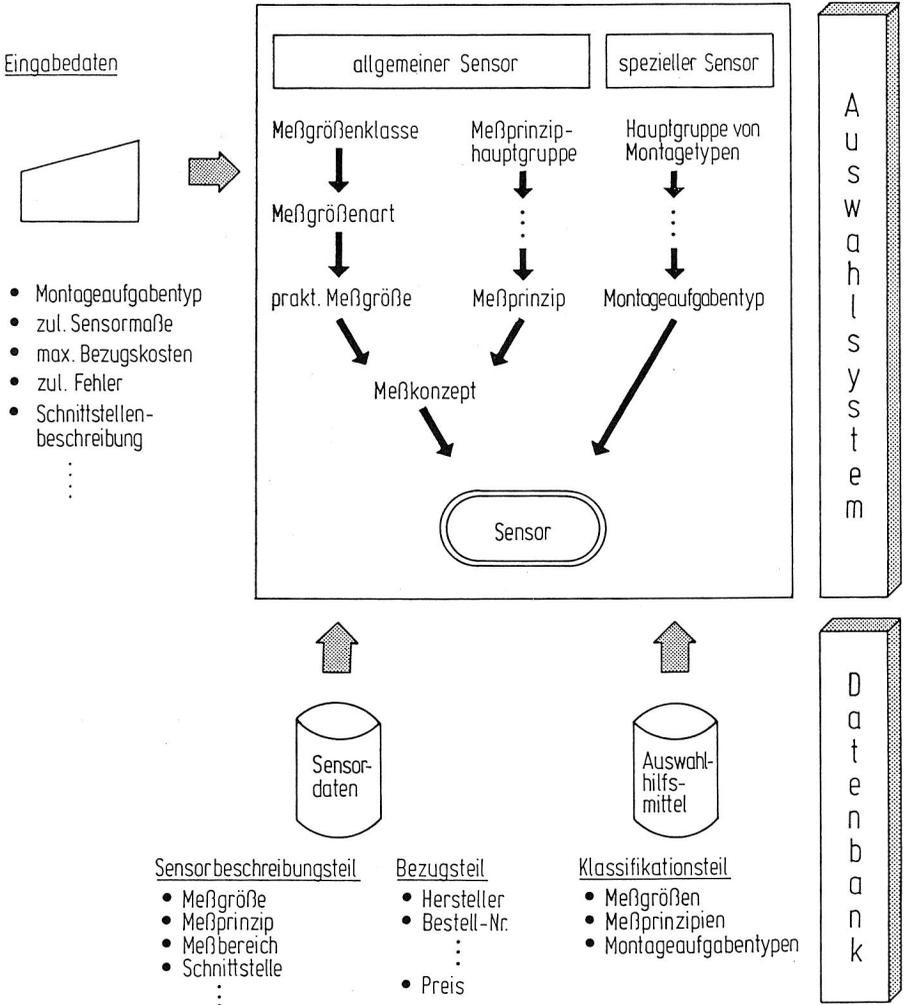
Bild 17: Sensor als dreielementiges Gebilde

Auslegung spezifischer Schnittstellen fest, deren besondere Eigenschaften dem Sensoranwender im Regelfall gleichgültig sind. Innerhalb der Datenbank braucht die Kombinierbarkeit der Einzelelemente deshalb nicht durch technische Daten beschrieben zu werden, es reichen einfache Vorgänger- oder Nachfolgerbeziehungen aus. Einfacher liegen die Verhältnisse bei Sensoren, die vollständig in ein Gehäuse eingebaut sind, sie können als einzelne Elemente betrachtet werden, die im wesentlichen durch Eingangs- und Ausgangsmerkmale beschreibbar sind.

Neben dem sensorbezogenen Teil, der sich aus dem Bezugs- und dem Sensorbeschreibungsteil zusammensetzt, enthält die Datenbank in Bild 18 eine dritte Komponente, den Klassifikationsteil, die den Zusammenhang zum eigentlichen Auswahlssystem herstellt. Entsprechend der Untergliederung des Systems in einen Teil zur Behandlung allgemeiner und einen zur Behandlung spezieller Sensoren, sind in diesem Teil der Datenbank die korrespondierenden Klassifikationen für Meßgrößen und Meßprinzipien (vgl. Bilder 15 und 16) sowie eine Klassifikation für Montageaufgabentypen abgebildet. Als spezielle Sensoren, in deren Gruppe z.B. die im letzten Abschnitt erwähnten Schraubensensoren einzuordnen sind, sollen Spezialkonstruktionen verstanden werden, die unter Berücksichtigung der spezifischen Randbedingungen auf einen ganz bestimmten Typ von Montageaufgabe hin ausgelegt sind. Zu dieser Gruppe gehören auch diejenigen Sensoren, die im Sinne von DIN 1319 [9] keine echten Messungen durchführen, sondern eine symbolische Zuordnung zwischen dem beobachteten Zusammenhang und der gelieferten Nutzinformation vornehmen. Allgemeine Sensoren sind dagegen für möglichst vielfältige Verwendungsmöglichkeiten gedacht und zumeist einfacher aufgebaut. Innerhalb der Klassifikation der Montageaufgabentypen (s. Bild 19), die als Ordnungsschema für die speziellen Sensoren dient, bietet es sich an, die Hauptgruppe 'Fügen' nach DIN 8593 [43] und die Hauptgruppe 'Handhaben' nach der VDI-Richtlinie 2860 [34] genauer zu beschreiben.

Die sinnvolle Gliederungstiefe sollte sich aber in jedem Fall am Sensorangebot orientieren.

Die Notwendigkeit zur getrennten Behandlung spezieller und allgemeiner Sensoren entsteht durch die in Bild 18 vorgese-



**Bild 18:** Arbeitsweise eines rechnergestützten Auswahlsystems für Sensoren

hene, vereinfachte Beschreibung des Meßkonzepts. Das Meßkonzept enthält nämlich keine Detailbeschreibung des Meßverfahrens, sondern eben nur das Meßprinzip, nach dem der Sensor arbeiten soll. Für die meisten einfach gelagerten Fälle reicht diese Angabe auch aus, da über das Meßprinzip den meisten praktischen Randbedingungen einer Sensoraufgabe Rechnung getragen werden kann. Diese Vereinfachung hat aber zur Folge, daß der Auswahlmechanismus automatisch auf mehrere Einzelsensoren führt, wenn die Meßgröße durch eine indirekte Messung, bestehend aus zwei oder mehr Einzelmessungen, erfaßt wird. Ein Gerät, das beispielsweise als Schraubensensor

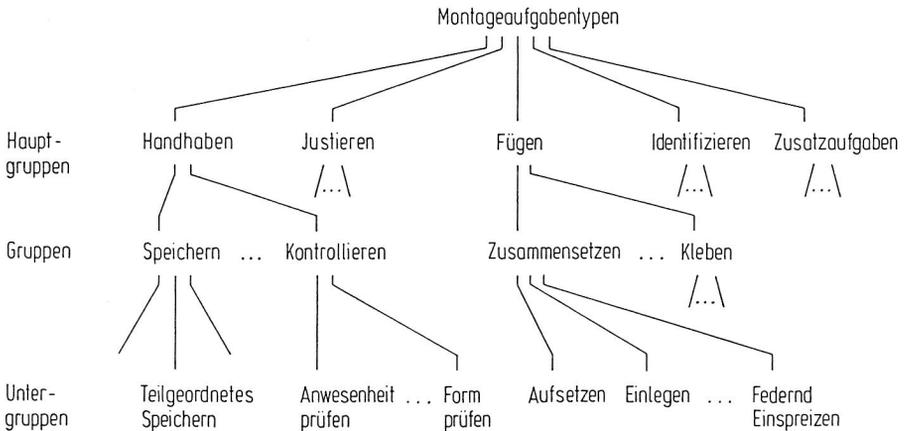


Bild 19: Typen von Montageaufgaben

für das 'Streckgrenzengesteuerte Anziehen' (s. Abschnitt 2.4.2) angeboten wird, enthält aber intern Meßeinrichtungen für den Drehwinkel und das Drehmoment und ist deshalb nur über eine genaue Beschreibung des Meßverfahrens identifizierbar. Da den Sensoranwender aber zumeist das sensorinterne Meßverfahren nicht interessiert, es häufig auch gar nicht bekannt ist, erfahren die speziellen Sensoren eine Sonderbehandlung, und die datenbankinterne Beschreibung des Meßverfahrens für allgemeine Sensoren kann sich, soweit es

den Auswahlvorgang betrifft, auf die Angabe des primären Meßprinzips beschränken. Selbstverständlich kann die Datenbank zusätzliche Einträge enthalten, die das Meßverfahren näher spezifizieren, für den Auswahlmechanismus aber nicht direkt relevant sind.

Das vorangehend skizzierte Auswahlssystem übernimmt die Suche nach Sensoren nicht selbständig; der Bediener fällt die Entscheidungen eines jeden Schritts nach eigenem Gutdünken. Andernfalls wären Methoden der 'Künstlichen Intelligenz' erforderlich, für deren Entscheidungsregeln bezüglich der Sensorfrage aber derzeit kaum allgemeingültiges Wissen vorliegt. Im Grunde genommen stellt das System eine Bedienoberfläche für eine Sensordatenbank dar, die auf die Arbeitsweise eines Meßtechnikers abgestimmt ist.

Weder das Auswahlschema nach Abschnitt 2.4.2 noch ein darauf aufbauendes System ist für einen meßtechnisch unerfahrenen Bearbeiter geeignet. Bei schwierigen Aufgabenstellungen bedarf es in den verschiedenen Bewertungs- und Entscheidungsschritten durchaus eines gewissen fachlichen Erfahrungsschatzes, um mit hoher Sicherheit einen geeigneten Sensor zu finden. Das gilt im übrigen auch für die von EVERSHEIM und HAUSMANN [13,14] vorgeschlagene Auswahlmethode. Der Sinn des schematisierten Vorgehens liegt darin, der hohen Dynamik des Sensormarktes Rechnung zu tragen und über die bekannten Lösungsmöglichkeiten hinaus neue greifbar und durch Rechnerunterstützung miteinander vergleichbar zu machen.

### 3. Sensorgestützte Roboterprogrammierung

#### 3.1 Methoden der Bewegungsprogrammerstellung

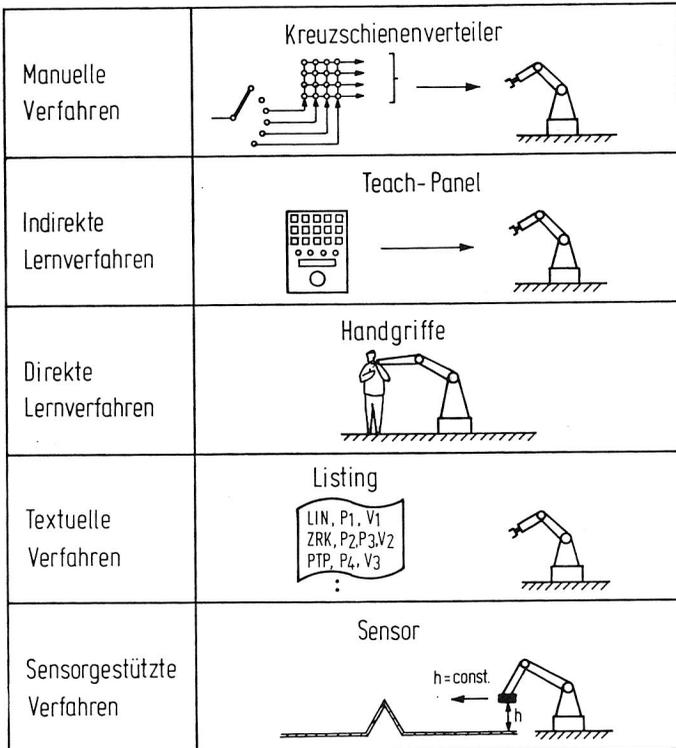
Häufige Änderungen am Produkt bzw. an den Produktionsmitteln ziehen einen spürbaren Aufwand zur Bewegungsprogrammerstellung für die Montageroboter nach sich. Da die Effizienz, mit der die Programme erstellt werden können, sich maßgeblich auf die Flexibilität der Geräte auswirkt, gilt, seit Roboter überhaupt in der Produktion Verwendung finden, die Entwicklung verbesserter Programmierverfahren als wichtiges Ziel. Die heute bekannten Verfahren lassen sich grob in vier Gruppen einteilen (vgl. Bild 20):

- Manuelle Verfahren,
- Lernverfahren
  - \* Eingabe des Bewegungsablaufs über Teach-Panel (indirektes Teach-In),
  - \* Direktes Verfahren der Achsen über Handgriffe (direktes Teach-In),
- Textuelle Verfahren,
- Sensorgestützte Verfahren.

Einzig die textuellen Verfahren erlauben, die Programmierung wahlweise online am Roboter oder offline an einem speziellen Programmierplatz durchzuführen. Die anderen Verfahren sind immer an das Gerät gebunden.

Manuelle Programmierverfahren kommen mit relativ einfachen technischen Mitteln aus, sie arbeiten beispielsweise mit Kreuzschienenverteiltern oder einfachen Logiksteuerungen und verstellbaren Anschlägen, finden aber nur noch bei sehr einfachen Handhabungsgeräten Verwendung. Als Standardverfahren im industriellen Bereich haben sich Lernverfahren durchgesetzt, die meist gemischt mit einfachen Roboterprogrammiersprachen angeboten werden. Sie erlauben dank ihrer Bedienerfreundlichkeit und Anschaulichkeit einen problem-

losen Ersteinsatz bei Anwendern, die noch keine Robotererfahrung gesammelt haben. Als nachteilig erweisen sich allerdings die meist unzureichenden Programmdokumentationsmöglichkeiten sowie die langen Stillstandszeiten der Geräte während der Programmierphasen. Rein textuelle Programmier-



**Bild 20:** Programmierverfahren für Industrieroboter

verfahren haben sich bisher nicht durchsetzen können, da die Bewegungsbeschreibungen besonders bei räumlichen Aufgabenstellungen sich als zu wenig anschaulich erwiesen haben, und graphische Hilfsmittel noch nicht in ausreichendem Maß zur Verfügung stehen. Bei den sensorgestützten Verfahren liegen bisher nur wenige Erfahrungen unter industriellen Bedin-

gungen vor, sie stellen derzeit noch sehr hohe Anforderungen an die fachliche Qualifikation der Gerätebediener.

### 3.1.1 Textuelle Programmierung

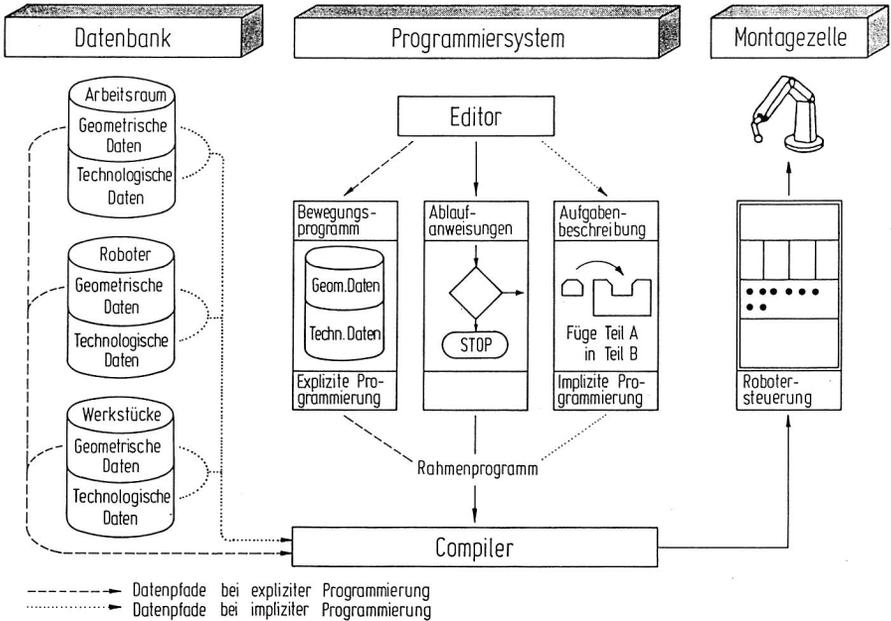
Verschiedene Programmiersprachen unterschiedlichen Sprachumfangs sind in der Vergangenheit für Roboter entwickelt und meist in Forschungslaboratorien, zum kleineren Teil auch in der Praxis getestet worden [44-49]. Aktuell ist aber die Diskussion um Sprachkonzepte, geeignete Datentypen und die notwendige Mächtigkeit der Sprachen etwas zurückgetreten gegenüber der Diskussion um Vor- und Nachteile der Programm-erstellung in der Arbeitsvorbereitung (offline). Als Kompromiß bieten die meisten Steuerungshersteller heute die Möglichkeit an, Roboterprogramme auf einem Personal Computer zu erstellen und nach Fertigstellung in die Robotersteuerung zu laden [50,6]. Verwendung finden die einfachen Sprachen, die derzeit zur Ergänzung der Lernverfahren gebräuchlich sind. Der Anwender kann in der Regel ein so erstelltes Rahmenprogramm am Roboter vervollständigen, indem er mit dem Teach-Panel die Positionen einlernt, die zuvor nicht exakt bekannt waren. Der Zeitanteil, den der Roboter mit Programmieraufgaben belegt ist, sinkt auf diese Weise beträchtlich. Als Zielvorstellung der weiteren Entwicklung wird aber derzeit der völlige Verzicht auf Lernverfahren angesehen. Die Steuerungen würden einfacher werden, wenn die Teach-Mechanismen und die zugehörigen Sicherheitsfunktionen entfallen könnten. Außerdem könnten, was als bedeutsamer angesehen wird, die Montagesysteme kompakter aufgebaut werden, wenn die Roboter bei Programmänderungen nicht mehr für Bediener zugänglich sein müßten. Als Ersatz für die Anschaulichkeit des realen Systems benötigt der Benutzer bei der Offline-Programmierung einen komfortablen Editor, ein Paket zur graphischen Simulation und einen geometrischen Modellierer für die Geräte und Vorrichtungen der Montagezelle [51-53].

Der relativ hohe Zeitaufwand, der zur Erstellung einer Modellwelt erforderlich ist, erscheint aus heutiger Sicht dadurch akzeptabel, daß die Ergebnisse der Arbeiten in der Modellwelt auf zweifache Weise nutzbar sind. Zum einen besteht bereits in der Planungsphase, also bevor die Montagezelle physikalisch existiert, die Möglichkeit, Kollisions- und Erreichbarkeitsstudien mit verschiedenen Roboterkinematiken durchzuführen sowie Taktzeitabschätzungen vorzunehmen. Die Ergebnisse können die Basis zur Beurteilung der Zellenkonzepte verbessern. Zum anderen trägt die Verlagerung von Erstellung und Test der Bewegungsprogramme in die Arbeitsvorbereitung zur Verbesserung des Nutzungsgrades der Montagezelle bei, wenn sie später in den Produktionsablauf integriert ist.

Bei den textuellen Programmierverfahren lassen sich nach der Art der genutzten Daten 'explizite', d.h. bewegungsorientierte Verfahren, und 'implizite', d.h. aufgabenorientierte Verfahren unterscheiden (Bild 21). Unter Verwendung der in der Modellbeschreibung enthaltenen Information über Arbeitsraum, Roboter und Werkstücke generiert der Compiler bei der impliziten Programmierung im Idealfall das Roboterprogramm selbsttätig und allein aus der Aufgabenbeschreibung, also ohne zusätzliche Angabe geometrischer oder technologischer Daten durch den Programmierer. Bei der expliziten Programmierung muß der Programmierer das zur Aufgabendurchführung benötigte Bewegungsprogramm mit den zugehörigen Ablaufanweisungen angeben. Er kann sich jedoch die Eingabe einer großen Zahl von Geometriedaten ersparen, wenn er die benötigten geometrischen Daten des Arbeitsraums oder der Werkstücke aus den vorhandenen Datenbeständen abrufen.

Implizite Programmiersysteme werden derzeit nicht industriell angeboten, der Kenntnisstand über die gängigen Montageverfahren ist dafür noch zu gering. Es gibt allerdings Systeme im Laborstadium [54]. Explizite Programmiersysteme

beginnen Einzug in die industrielle Praxis zu halten, sie bekommen nach heutigem Stand die Geometrieinformation aus CAD-Systemen in Form von Frame-Listen zur Verfügung gestellt. LAUER beschreibt in [50] ein solches Programmiersystem mit Zugriff auf CAD-Dateien relativ einfachen Zuschnitts, das für industrielle Anwender gedacht ist.



**Bild 21:** Die Verwendung existierender Daten zur Roboterprogrammierung

Unabhängig davon, ob es sich nun um implizite oder explizite Programmierverfahren handelt, sind rein offline erzeugte Programme heute nur in Ausnahmefällen ohne Nachkorrektur mit Lernverfahren verwendbar. Die größte Schwäche der Offline-Programmierung liegt nämlich begründet in der notwendig hohen Übereinstimmung der Modellwelt, in der programmiert wird, und der realen Welt, in der das Programm ausgeführt

werden soll. In der Praxis stimmen aber weder der Montagezellenaufbau, noch die Kinematik des Roboters mit den Modellen so gut überein, daß Genauigkeiten erreicht werden können, wie sie die Kleinteilmontage erfordert. Die Verwendung von Aluminiumprofilen zum Zellenaufbau, der Trend zur Modularisierung der Systemelemente und nicht zuletzt die aus Taktzeitgründen bevorzugte leichte Bauart von Robotern laufen der Forderung nach absoluter und hoher Genauigkeit aller Positionen innerhalb der Gesamtzelle diametral entgegen. Obwohl damit zu rechnen ist, daß in Kürze Montage-roboter mit ausreichend kleinen Positionsabweichungen verfügbar sein werden [55], ist mit einem breiten Durchbruch der Offline-Programmierung erst dann zu rechnen, wenn Mechanismen zur Verfügung stehen, die unter Verwendung von Sensorinformation entweder das Zellenmodell oder das Bewegungsprogramm der realen Umwelt anpassen können.

### 3.1.2 Sensorgestützte Programmierung

Unter Verwendung geeigneter Sensoren kann der Roboter selbst als Meßmaschine arbeiten und Geometrieinformationen aufnehmen, die zur Verbesserung von Parametern der Modellwelt oder direkt zur Erstellung von Bewegungsprogrammen für den Roboter genutzt werden können. Da die Methoden der Offline-Programmierung noch keineswegs ausgereift sind, und bisher kaum Erfahrungen mit der Parametervariation bei Modellen durch Sensoreinfluß vorliegen, herrscht die direkte Nutzung der Geometrieinformation zur Generierung von Bewegungsprogrammen vor. Diese Methode wird als 'sensorgestützte Programmierung' bezeichnet und eignet sich besonders gut für Aufgaben, bei denen mathematisch schwer beschreibbare Werkstückgeometrien die Roboterbahnen festlegen. Der sensorbestückte Roboter nimmt die gesuchte Geometrieinformation von der Werkstückoberfläche ab und überführt sie auf diese Weise vom mechanischen Geometriespeicher der Werkstückgestalt in die rechnerinterne Darstellung der Roboter-

steuerung.

In der Literatur findet sich eine Reihe von Arbeiten, die die Verwendung von Sensorinformation bei der Bewegungsprogrammerstellung behandelt. Die Autoren berichten, je nach Art des verwendeten Verfahrens, von beträchtlicher Verkürzung der Programmierzeiten gegenüber konventioneller Programmierweise [11,57]. Gegenüber der Offline-Programmierung ist es zwar nachteilig, daß der Roboter während der Programmierphasen keine Fertigungsaufgaben übernehmen kann, der Vorteil der Methode liegt aber in der erreichbaren Genauigkeit, denn der Sensor erfaßt neben den Nutzdaten auch sämtliche Ungenauigkeiten des mechanischen Aufbaus und des Roboters selbst. Die hochgenaue räumliche Vermessung der gesamten Zelle und ihrer Geräte entfällt.

Grundsätzlich hat der Roboter bei der sensorgestützten Programmierung in der Vorbereitungsphase den Sensor im Eingriff und fährt mit Hilfe von Steuerungsmechanismen einen Weg auf der Oberfläche des Werkstücks ab. Die Steuerung speichert dabei zunächst die gewonnene Geometrieinformation und wandelt sie später in ein ablauffähiges Bewegungsprogramm um. In der Ausführungsphase ist anstelle des Sensors das Werkzeug im Eingriff, und die Steuerung korrigiert die zuvor mit dem Sensor erfaßte Information entsprechend der Werkzeugmaße. Nach dem Arbeitsaufwand, den die Vorbereitungsphase verursacht, lassen sich zwei Typen von Verfahren unterscheiden:

- a) Der Bediener erzeugt ein Bewegungsprogramm, das den Sensor auf der Werkstückoberfläche entlangführt. Diese Bahn ist der Werkstückgeometrie bereits so gut angepaßt, daß auf die Rückführung der Sensorsignale im Sinne einer Regelung verzichtet werden kann. Der Sensor erfaßt Geometrieabweichungen, die als Korrekturwerte in die Feinabstimmung des Bewegungsablaufs einfließen [11,56,57].

b) Das vom Bediener zu erstellende Bewegungsprogramm besteht nur aus einer geringen Stützpunktzahl; im Extremfall reichen Start- und Zielpunkt aus. Zwischen den Stützpunkten wird die Bahn des Sensors online an den Verlauf der Werkstückoberfläche angepaßt. Aufgezeichnet wird die Roboterposition mit Hilfe der internen Inkrementalgeber. Da hier eine Sensorregelschleife notwendig ist, können Kraft- oder Positionssensoren verwendet werden [10,58-62].

Beide Verfahrenstypen beruhen darauf, daß Sensoren eine vorgegebene Bahn an die Werkstückgegebenheiten anpassen. Je nach Art der verwendeten Sensorinformation und der vorliegenden Werkstückgestalt muß der Programmierer unter genauer Kenntnis der Arbeitsweise des Verfahrens die Stützpunkte des Bewegungsprogramms an geeignete Stellen setzen und die erforderlichen Korrekturbewegungen auf die vorliegende Aufgabe abstimmen.

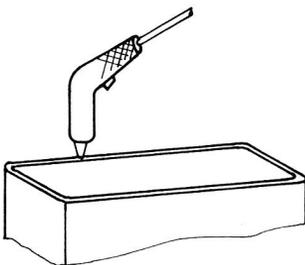
Im Gegensatz dazu wird im folgendem ein neues Verfahren vorgestellt, bei dem sich die Festlegung von Stützpunkten erübrigt, wodurch die Bedienung des gesamten Einlernvorgangs inklusive Vorbereitung stark vereinfacht werden kann.

### 3.2 Entwicklung eines neuen sensorgestützten Programmierverfahrens

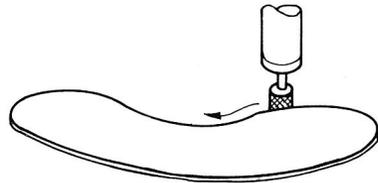
Häufig sind neben dem eigentlichen Montagevorgang, der zum Fortgang des Zusammenbauens beiträgt, weitere Vorgänge erforderlich, die zusätzliche Arbeiten an Einzelwerkstücken, Baugruppen oder den fertig montierten Produkten umfassen [63]. Einige dieser Arbeiten, wie z.B. 'Entgraten', 'Polieren', 'Kleber auftragen' u.a.m., enthalten Werkzeugbewegungen, deren Geometrie festgelegt ist durch die Geometrie der fraglichen Produkte. Um bei kleinen Losen verschiedener Produkte, die im Wechsel aufeinander folgen, nicht eine

Vielzahl individueller Programme erstellen und warten zu müssen, kann es in manchen Fällen sinnvoll sein, mit Sensorunterstützung jeweils für ein Los identischer Produkte ein neues Bewegungsprogramm zu erstellen. Bei dieser Organisationsform sind kleinere produktseitige Änderungen tolerierbar, ohne daß neue Einrichtarbeiten erforderlich werden.

Konstruktive Festlegungen bei den Werkstücken bewirken relativ häufig, daß alle notwendigen Werkzeugbewegungen in eine Ebene fallen, wodurch ein Bewegungsfreiheitsgrad wegfällt. Bild 22 zeigt zwei Beispiele dieser Aufgabenklasse. Im linken Teil bringt eine Klebepistole Kleber in eine Rinne ein, die später z.B. einen Gehäuseboden oder eine Zierblende aufnimmt. Rechts fährt ein Entgratwerkzeug an einer Blechkante entlang, deren Stanzgrat den optischen Eindruck des Endproduktes beeinträchtigen würde. Bei dieser Klasse von Aufgaben, die sich durch ebene Werkzeugbewegungen auszeichnet, hat der Anwender die Möglichkeit, sein Vorwissen über den Verlauf der späteren Roboterbahnen zur Vereinfachung des Programmiervorgangs zu verwenden. Anstelle einer Anzahl ausgewählter Stützpunkte, wie es ein Teil der



a) Kleber auftragen



b) Entgraten von Blechen

Bild 22: Typische Arbeiten mit ebenen Werkzeugbewegungen zur Vor- und Nachbereitung von Fügevorgängen in der Montage

oben genannten Autoren vorschlägt, legt er nun lediglich die Ebene fest, in der die Bewegung erfolgen soll und sorgt im übrigen dafür, daß der Sensor an das Werkstück herangeführt wird. Die gesuchte Roboterbahn entsteht dann eindeutig als Schnitt der Werkstückoberfläche mit dieser Ebene. Die Schnittlinie stellt eine ebene Kontur im Raum dar, die der Sensor in einer Folge von Abtastschritten abfahren soll. Dazu muß die Steuerung in jedem Abtastpunkt aus den aufgenommenen Meßgrößen ein neues Bahnstück errechnen, dem der sensorbestückte Roboter bis zum nächsten Abtastpunkt folgt.

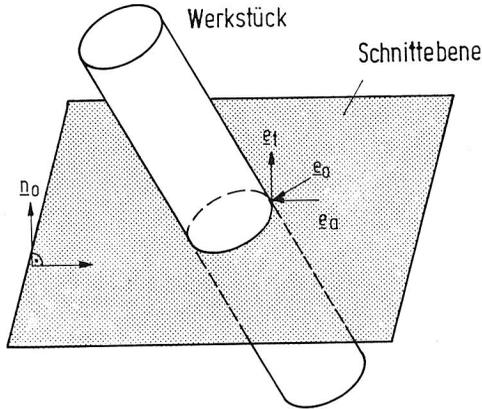


Bild 23: Lage der Richtungsvektoren

Die Bewegung, die den Roboter unter Sensorführung das Werkstück umfahren läßt, und aus der später die Kontur ermittelt werden soll, läßt sich erzeugen aus zwei überlagerten, vektoriellen Führungsfunktionen, die zu jedem Zeitpunkt senkrecht aufeinander stehen. Die eine sorgt für eine Bewegung, die den Abstand des Roboters zur Werkstückoberfläche konstant hält, während die andere für das Vorschreiten entlang der Kontur sorgt. Mit Hilfe des Einheitsnormalenvektors der Werkstückoberfläche  $e_0(j)$ , der in jedem Abtastpunkt zu bestimmen ist, lassen sich die Richtungs-

vektoren beider Führungsfunktionen festlegen. Das normierte Kreuzprodukt des Einheitsvektors  $\underline{e}_0(j)$  mit dem Einheitsnormalenvektor  $\underline{n}_0$  der Schnittebene, die der Anwender festgelegt hat, liefert nach

$$\underline{e}_t(j) = \frac{\underline{e}_0(j) \times \underline{n}_0}{\|\underline{e}_0(j) \times \underline{n}_0\|} \quad (3.1)$$

den Einheitsvektor  $\underline{e}_t(j)$ , der die Vorschubrichtung angibt. Er steht senkrecht zum Einheitsvektor  $\underline{e}_0(j)$  und bildet im Meßpunkt eine Tangente an die Werkstückoberfläche (Bild 23). Um den gewünschten Abstand zum Werkstück zu halten und gleichzeitig die Schnittebene nicht zu verlassen, wird ein Einheitsvektor  $\underline{e}_a(j)$  innerhalb der Schnittebene bestimmt, der die Korrekturrichtung zum Ausgleich der durch die Vorschubbewegung verursachten Entfernungsabweichung angibt:

$$\underline{e}_a(j) = \underline{n}_0 \times \underline{e}_t(j) . \quad (3.2)$$

Die beiden vektoriellen Führungsfunktionen setzen sich zusammen aus ihren Einheitsvektoren der Richtung (Gleichungen (3.1) und (3.2)), die unter Berücksichtigung der im Abtastpunkt aufgenommenen Meßinformation berechnet werden, und zugehörigen, konstanten Beträgen, die der Anwender als Vorgabe festlegt (vgl. Bild 29).

### 3.2.1 Auswahl eines Meßkonzepts

Da die Roboterbewegung durch Werkstück und Schnittebene lediglich geometrisch bestimmt ist und der Anwender unter Berücksichtigung der technologischen Randbedingungen der Werkzeugoperation das gewünschte Geschwindigkeitsprofil selbst wählen soll, läßt sich als Sensorfunktion (vgl. Kap. 2.1.4) das Einlernen von Roboterbahnen angeben. Neben dieser Sensorfunktion enthält die Sensoraufgabe, die den Zugang zum Auswahlschema für Sensoren nach Abschnitt 2.4.2 herstellt,

eine Reihe zusätzlicher Angaben, die bedeutsam sind für den gesuchten Sensor. Dazu gehören u.a. die geometrischen Abmaße der in Frage kommenden Werkstücke, ihre Werkstoffe und Oberflächenbeschaffenheit, der Bereich zulässiger Orientierungen für den Einheitsnormalenvektor  $\underline{n}_0$  der Schnittebene, die maximal zulässigen Abtastkräfte für den Fall, daß taktile Meßverfahren ausgewählt werden, sowie die Begrenzungen des Arbeitsraums. Diese Angaben fließen in geeigneter Form in die Anforderungsliste ein. Zur Sicherung der universellen Verwendbarkeit des Sensors muß sich, als weitere Anforderung, das festzulegende Meßkonzept in jedem Fall eignen für die Vermessung kleinerer Werkstücke, frei zugänglicher Hohlräume sowie beliebiger Werkstückgeometrien, einschließlich solcher mit Kanten und Ecken.

Da durch die Merkmale des Programmierverfahrens die erforderlichen Meßgrößen, nämlich der Abstand des Roboters vom Werkstück und die drei weiteren geometrischen Größen, die zusammen den Einheitsnormalenvektor  $\underline{e}_0(j)$  ergeben, bereits festgelegt sind, ist nach der Vervollständigung der Anforderungsliste um die meßgrößenbezogenen Angaben (4. Arbeitsschritt) die erste Phase der Sensorauswahl abgeschlossen (vgl. Bild 12).

Prinzipiell kann die Orientierung der Werkstückoberfläche (Vektor  $\underline{e}_0(j)$ ) auf indirektem Wege, nämlich durch Messung der Entfernung von einem Referenzpunkt auf mindestens drei verschiedene Oberflächenpunkte, die nicht auf einer Linie liegen, erfaßt werden (vgl. 5. Arbeitsschritt in Bild 12). Die Mittelung der Einzelmessungen liefert gleichzeitig die ebenfalls gesuchte Entfernungsgröße. GRUHLER [60] hat bei seinen Arbeiten ein solches Verfahren zur Programmierung von Bearbeitungsrobotern verwendet und mit mechanoelektrischen Wegaufnehmern gearbeitet. Dieser Typ von Meßverfahren schränkt allerdings die zulässigen Werkstückgeometrien spürbar ein, da einwandfreie Meßergebnisse nur zu erhalten sind, wenn die Krümmung der Oberfläche im meßtechnisch

relevanten Bereich annähernd konstant und möglichst gering ist. Für die Beispielaufgaben aus Bild 22 eignet sich dieser Typ von Meßverfahren damit nicht.

Die Eingangsforderung sieht aber ausdrücklich ein möglichst breites Spektrum von Oberflächenformen vor, weshalb statt der geometrischen im weiteren ersatzweise mechanische Meßverfahren geprüft werden. Ausgangspunkt ist dabei das Gegenkraftprinzip, das im stationären Fall für einen Ausgleich von Kräften sorgt. Eine einzelne statische Kraft ist in dieser Vorstellung nicht ohne andere entgegengerichtete, die erste kompensierende Kräfte denkbar. Drückt ein taktiller, unbewegter Sensor auf eine Werkstückoberfläche, bildet sich stationär eine Kraft aus, die der Sensor auf das Werkstück ausübt, und gleichzeitig eine gleichgroße Gegenkraft, die vom Werkstück auf den Sensor wirkt. Beide Kräfte stehen senkrecht auf der Werkstückoberfläche und zeigen in die gleiche bzw. die entgegengesetzte Richtung wie der Normalenvektor der Werkstückoberfläche. Der Sensor berührt die Werkstückoberfläche nur in einem Punkt und erfaßt dort die drei Kraftkomponenten, aus denen die gesuchten Meßgrößen abgeleitet werden können. Dabei ist die Entfernungsgröße in der Länge des gemessenen Kraftvektors und die Oberflächenorientierung in seiner Richtung enthalten. Die Erfassung aller Größen in einem Meßpunkt macht es möglich, auch stark gekrümmte Oberflächen und schmale Blechkanten abzutasten und sichert dadurch breite Verwendungsmöglichkeiten für den Sensor.

Nebenbei ist die kombinierte Erfassung von Orientierung und Entfernung im gleichen Meßpunkt eine Voraussetzung dafür, daß auch Roboter mit eingeschränkten Freiheitsgraden Werkstückoberflächen in beliebigen Schnittebenen abtasten können. Denkt man sich die Schnittebene nämlich um 45 Grad zur x,y-Ebene verdreht, sind die gängigen Montageroboter in SCARA- oder Linearbauweise (s. Bild 2) von sich aus nicht mehr in der Lage, die Orientierung eines Sensors zur Werk-

stückoberfläche nachzuführen, was bei den oben skizzierten geometrischen Meßverfahren unumgänglich wäre. Eigens für den Programmiervorgang müßte in diesem Fall eine zusätzliche Drehachse am Roboter oder in der Werkstückaufspannung installiert werden. Unangenehm an den mechanischen Meßverfahren ist allerdings, daß durch die Vorschubbewegung Reibkräfte auftreten, die als Störgröße die Orientierungsmessung beeinträchtigen. Bei erhöhten Genauigkeitsforderungen ist es deshalb notwendig, die Reibkräfte in einem Störmodell zu berücksichtigen und die gemessenen Größen entsprechend zu korrigieren.

Mögliche Meßprinzipien zur Erfassung der Kraftgrößen, aus denen die gesuchten Meßgrößen bestimmt werden sollen, gehören zur Klasse der 'Gegenkraftprinzipien' (vgl. Bild 16), von denen aus Gründen der Baugröße und der Kosten die 'Verformung mechanischer Körper' als das geeignetste erscheint [35]. Praktische Verwendung findet dieses Meßprinzip besonders häufig bei Sensoren, die die Verformung von Meßkörpern mit Hilfe von Dehnungsmeßstreifen aufnehmen und solchen, die den Piezoeffekt nutzen. Piezoelektrische Aufnehmer gibt es in sehr kleinen Bauformen auch für mehrere Koordinaten, bei der vorliegenden Aufgabe erweist es sich aber als sehr nachteilig für den praktischen Betrieb, daß die Aufnehmer nicht in der Lage sind, statische Kräfte zu messen. Diese Eigenschaft macht die Sensoren zwar nicht grundsätzlich ungeeignet, würde aber relativ aufwendige steuerungstechnische Zusatzmaßnahmen erforderlich machen, weshalb dieser Aufnehmertyp nicht in die weiteren Betrachtungen einbezogen wird.

Zusammenfassend nochmals die markanten Eigenschaften des nach den obigen Überlegungen geeigneten Meßkonzepts: Ein indirektes Meßverfahren soll verwendet werden, das die gesuchten Meßgrößen aus drei Ersatzgrößen, nämlich Kräften, ableitet und mit dem Meßprinzip der 'Verformung mechanischer Körper' arbeitet. Dehnungsmeßstreifen sollen dabei die

Verformungen geeigneter Meßkörper aufnehmen und auf ein elektrisches Signal abbilden, das sich zur Weiterverarbeitung in der Steuerung eignet.

Prinzipiell bietet das Verfahren der Bahnkorrektur, das die Autoren der Gruppe b (vgl. Abschnitt 3.1.2) behandeln, freizügigere Möglichkeiten bei der Wahl eines Meßkonzepts. Da eine Roboterbahn in Form einer groben Stützpunktfolge immer als existent vorausgesetzt werden kann, ist es nämlich möglich, die erwünschte Korrekturrichtung, sofern die vorliegende Aufgabe entsprechend beschaffen ist, auf eine Komponente zu beschränken, wodurch statt der vier wie im vorliegenden Fall nur eine Größe gemessen werden muß. In [62] stellen die Autoren eine solche Anwendung vor, sie verwenden einen industriell hergestellten Entfernungsmesser, der nach dem Triangulationsprinzip arbeitet.

### 3.2.2 Prüfung marktüblicher Sensoren

Seit einigen Jahren sind Kraft-Momenten-Sensoren für Roboteranwendungen aus industrieller Herstellung erhältlich. Sie arbeiten zum überwiegenden Teil mit Dehnungsmeßstreifen und sind meist zur Überwachung von Bearbeitungsvorgängen wie Schleifen und Fräsen gedacht; der Bereich ihrer größten Genauigkeit liegt bei größeren Kräften. Bei praktischen Untersuchungen mit Probeexemplaren (Arbeitsschritt 7 des Auswahlschemas) zeigte sich, daß die mechanische Verformung der erhältlichen Sensoren minimal ist, und daß diese Erscheinung ihre Eignung als Programmierhilfsmittel mindert. Soll das in Abschnitt 3.2 skizzierte Verfahren zum Einlernen von Roboterbahnen auch an Oberflächen mit relativ abrupten Richtungsänderungen und bei größerer Fahrgeschwindigkeit noch einwandfrei funktionieren, ist ein gewisser Verformungsweg des Sensors unter Krafteinfluß unverzichtbar. Der Sensor muß in der Lage sein, die bei höheren Fahrgeschwindigkeiten unvermeidlichen Bahnfehler des Roboters

auszugleichen, ohne den Kontakt zum Werkstück zu verlieren. Da aus Gründen der Oberflächenschonung die Kräfte zur Abtastung der Oberfläche klein sein sollen, muß der Sensor zwangsläufig relativ elastisch aufgebaut sein. Grundsätzlich läßt sich natürlich die erforderliche Elastizität auch in die Sensoraufhängung [59] verlagern, es ist aber relativ schwierig, eine solche mit kleinen Abmessungen zu konstruieren, die in den drei Koordinatenrichtungen wenigstens annähernd gleiche Elastizität aufweist. Von Nachteil sind elastische Aufhängungen auch deshalb, weil die auf den Sensor wirkenden Kräfte nur noch einen kleinen Anteil der Abtastkräfte darstellen und sich das Nutzsignal/Störsignal-Verhältnis dadurch bedeutend verschlechtert.

HIRZINGER [64] geht bei seinen Arbeiten von einer endlichen Steifigkeit des Roboters selbst aus und verwendet einen relativ unelastischen Sensor. Bei näherer Prüfung ist diese Methode aber nur haltbar bei Geräten mit Harmonik-Drive-Getrieben oder solchen, die mit langen Torsionsstangen zur Übertragung der Antriebsenergie vom entfernt angeordneten Motor zum Gelenk arbeiten. Roboter mit Kugelumlaufspindeln oder Zahnriemen in der Kraftübertragung sind meist jedoch so steif, daß eine zusätzliche Elastizität in der Meßrichtung untergebracht werden muß.

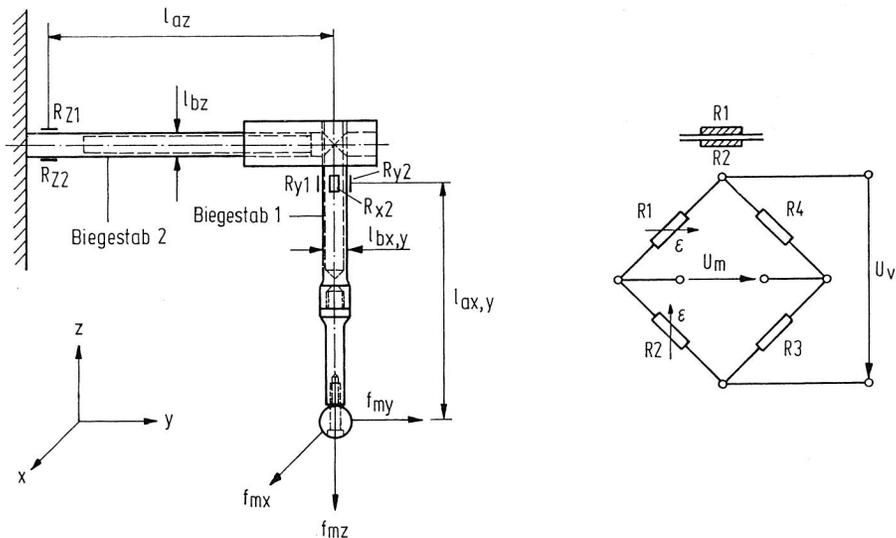
### 3.2.3 Entwicklung eines neuen Sensors

Die im vorangegangenen skizzierten Überlegungen und die Ergebnisse der praktischen Untersuchungen flossen in die Anforderungsliste ein und führten zu dem Ergebnis, daß kein auf dem Markt erhältliches Produkt ausreichend gut geeignet war. Anhand der Anforderungsliste wurde deshalb ein neuer Sensor speziell auf die Forderungen der Programmieraufgabe hin konzipiert und aufgebaut (s. Bild 24). Er besteht aus zwei Biegestäben, die senkrecht zueinander angeordnet und zur Erhöhung der Elastizität als dünnwandige Rohre aus-

geführt sind. Die Verformung dieser Meßkörper unter dem Einfluß der Kraft nehmen Dehnungsmeßstreifen auf. Die Kraftkomponenten  $f_{mx}$  und  $f_{my}$  verursachen an der Stelle, an der die Dehnungsmeßstreifen auf Stab 1 geklebt sind, ein Biegemoment, das mit Hilfe einer Brückenschaltung gemessen wird. Für die dabei wirksame Dehnung gilt folgende Beziehung:

$$\epsilon_{x,y} = \frac{f_{mx,my} l_{ax,y}}{W_b E_{st}} \quad \text{mit } W_b = \text{Widerstandsmoment,} \\ E_{st} = \text{E-Modul.} \quad (3.3)$$

Diese Anordnung der Dehnungsmeßstreifen eliminiert grundsätzlich Widerstandsänderungen, die auf Temperatureinflüsse zurückgehen, und mißt prinzipiell nur das Biegemoment in der gewünschten Richtung; die Brückenspannung der zweiten Komponente ist davon jeweils unabhängig. Die Anordnung des zweiten Biegestabes senkrecht zum ersten (Bild 24) erfüllt einen doppelten Zweck, sie sorgt in allen drei Koordinaten dafür, daß sowohl die Kräfte mit gleicher Empfindlichkeit



**Bild 24:** Kraftsensor mit Meßbrücke

gemessen als auch annähernd gleich große Elastizitäten im Sensoraufbau erzielt werden können. Es sind deshalb zwei weitere Dehnungsmeßstreifen vorgesehen, die das durch Kräfte in z-Richtung verursachte Biegemoment in Stab 2 aufnehmen.

Beim realen Aufbau ist die oben angenommene Symmetrie gestört durch Ungenauigkeiten bei der Ausrichtung der Dehnungsmeßstreifen und Empfindlichkeitstoleranzen der Streifen untereinander. Zusätzliche Störeinflüsse resultieren aus Querschnittsverformungen des Meßkörpers unter Belastung und der Verwendung des zweiten Biegestabes, dessen Anordnung auch bei Kräften in y-Richtung Änderungen der Brückenspannung in der z-Komponente bewirkt. Verkoppelungen der gemessenen Einzelkomponenten miteinander sind die Folge, sie müssen bei der Berechnung der Kraft  $\underline{f}_m$  durch Multiplikation des Meßspannungsvektors  $\underline{u}_m$  mit einer Entkoppelmatrix  $\underline{K}$  eliminiert werden:

$$\underline{f}_m = \underline{K} \underline{u}_m . \quad (3.4)$$

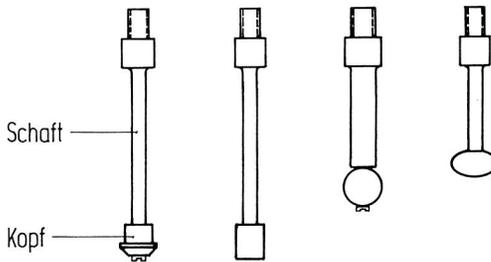
Da die Einflußfaktoren auf den Zusammenhang zwischen Kraft- und Meßspannungsvektor vielfältig sind, und eine Reihe von Störgrößen nur schwer quantifizierbar ist, werden die Elemente der Entkoppelmatrix experimentell bestimmt. Dazu läßt man drei bekannte Kräfte  $\underline{f}_1 \dots \underline{f}_3$  auf den Sensor wirken, mißt die zugehörigen Brückenspannungen  $\underline{u}_{m1} \dots \underline{u}_{m3}$  und setzt Kräfte und Spannungen in die folgende Gleichung ein:

$$\underline{K} = (\underline{f}_1, \underline{f}_2, \underline{f}_3) (\underline{u}_{m1}, \underline{u}_{m2}, \underline{u}_{m3})^{-1} . \quad (3.5)$$

Die Kräfte müssen linear unabhängig, können im übrigen aber beliebig sein.

In Biegestab 1 ist von unten eine auswechselbare Sensorspitze eingeschraubt, die es erlaubt, den Sensor auf einfache Weise an eine neue Aufgabenstellung anzupassen. Beispielsweise kann es bei Bewegungen in Hohlräumen nötig

sein, eine Spitze mit langem Schaft, oder zur Abtastung von schmalen Kanten eine Spitze mit zylinderförmigem Kopf zu verwenden. Bild 25 zeigt eine Auswahl von Sensorspitzen für verschiedene Verwendungszwecke. Bietet die Kinematik des Roboters die nötigen Freiheitsgrade, empfiehlt es sich, eine Spitze mit drehbar gelagerter Kugel zu verwenden und die Drehachse der Kugel über die Bewegung auf der Werkstückoberfläche entlang parallel zum Normalenvektor  $\underline{n}_0$  der Schnittebene zu halten. Durch eine gute Gleitlagerung kann damit die Reibung, die beim Umfahren des Werkstücks als Störgröße die Orientierungsmessung beeinträchtigt, stark



**Bild 25:** Beispiele verschiedener Sensorspitzen

vermindert und vor allem ihre Abhängigkeit von der Oberflächenbeschaffenheit des Werkstücks weitgehend eliminiert werden. Die Oberflächenrauigkeit des Werkstücks ebenso wie die des Tastkopflagers erzeugt einen eigenen Rauschanteil im Meßsignal, der störend auf die Konturverfolgungsbewegung einwirkt. Ihren Einfluß kann die Verwendung eines möglichst großen Tastkopfradius mildern. Andererseits muß der Kopfradius natürlich klein genug sein, um das kleinste interessierende Geometrieelement gerade noch abbilden zu können.

Da die Qualität der Konturerfassung ebenso wie die Stabilität des Abtastvorgangs maßgeblich von der Genauigkeit der Kraftmessung abhängen, muß die Entkoppelmatrix hin und

wieder neu bestimmt werden. Das ist z.B. immer dann erforderlich, wenn die Sensorspitze durch eine andere ersetzt wird oder der Sensor durch einmalige Überlastung eine bleibende Verformung erfahren hat. Der Vorgang zur Bestimmung der Matrix ist allerdings relativ aufwendig und fehleranfällig, sofern er manuell vorgenommen wird, was nahelegt, ihn mit Hilfe des Roboters automatisch auszuführen. Sensor und Roboter gemeinsam zu vermessen, hat zusätzlich den Vorteil, daß der Einfluß einer kleinen Verdrehung der Koordinatensysteme von Roboter und Sensor gegeneinander, verursacht durch eine unpräzise Sensorbefestigung, ebenfalls über die Entkoppelmatrix eliminiert werden kann. Wählt man nun die Kräfte  $\underline{f}_1, \underline{f}_2, \underline{f}_3$  in Gleichung (3.5) so, daß sie jeweils exakt in eine Koordinatenrichtung deuten, und ihr Betrag auf  $f_0$  normiert ist, nimmt die Matrix der Kräfte die einfache Form

$$(\underline{f}_1, \underline{f}_2, \underline{f}_3) = \begin{pmatrix} f_0 & 0 & 0 \\ 0 & f_0 & 0 \\ 0 & 0 & f_0 \end{pmatrix} = f_0 \underline{I} \quad (3.6)$$

an. Zur automatischen Bestimmung der Elemente der Entkoppelmatrix fährt der Roboter den Sensor in einer speziellen Meßbecke nacheinander an drei dort installierte, zusätzliche Kraftsensoren heran, die exakt nach den Roboterkoordinaten ausgerichtet sind, und stellt die Beträge der wirkenden Kräfte jeweils auf den normierten Wert  $f_0$  ein. Die drei Meßspannungsvektoren  $\underline{u}_{m1}, \underline{u}_{m2}$  und  $\underline{u}_{m3}$ , die der Sensor für die einzelnen Koordinatenrichtungen an die Steuerung übergibt, liefern dann in Verbindung mit dem normierten Kraftbetrag  $f_0$  nach

$$\underline{K} = f_0 (\underline{u}_{m1}, \underline{u}_{m2}, \underline{u}_{m3})^{-1} \quad (3.7)$$

die gesuchte Entkoppelmatrix  $\underline{K}$ .

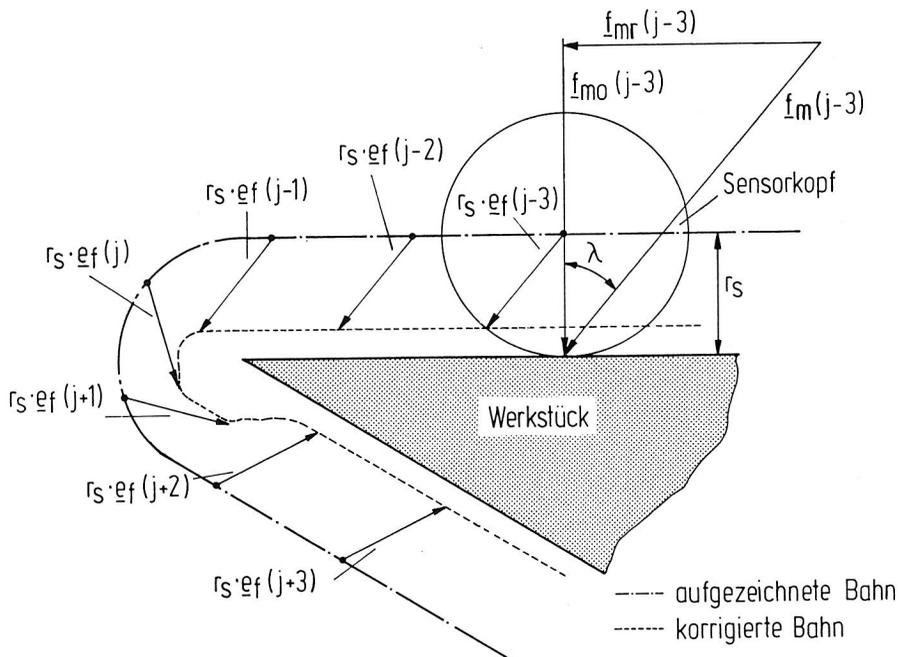
### 3.2.4 Berechnung der Kontur

Tastet der Fühler mit einer Kraft  $\underline{f}_m$  eine Werkstückoberfläche ab, verbiegen sich entsprechend ihrer jeweiligen Elastizität Roboterarm und Fühler. Unter der Annahme elastischer Verhältnisse sind die wirkende Kraft und die daraus resultierende Verformung einander proportional, der Verformungsweg konzentriert sich allerdings fast ausschließlich auf den Sensor. Die Wahl der Sensorelastizität erfolgte ja gerade nach der Maßgabe, eine sichere Konturverfolgung mit kleinen Kräften zu ermöglichen. Bei manueller Elastizitätsmessung kann deshalb ohne großen Fehler die Verbiegung des Roboters vernachlässigt werden. Größere Genauigkeit läßt sich allerdings erzielen, wenn man die im zurückliegenden Abschnitt eingeführte Meßbecke, mit deren Hilfe die Entkoppelmatrix automatisch bestimmt werden kann, auch zur Vermessung der wirksamen Elastizitäten in den drei Koordinatenrichtungen verwendet. Die Messungen liefern in diesem Fall die Gesamtelastizitäten von Sensor und Roboter, so wie sie zur Berechnung der Kontur gebraucht werden.

Stationär würde sich die Kraft im Berührungspunkt eines kugelförmigen Sensorkopfes mit der Werkstückoberfläche ausbilden. Während der Programmierphase bewegt sich die Sensorkugel aber auf der Oberfläche des Werkstücks entlang, wobei entweder im Lager des Sensorkopfes oder, bei feststehendem Kopf, auf der Werkstückoberfläche Reibung entsteht. Die dabei auftretenden Reibkräfte  $\underline{f}_{mr}$  (s. Bild 26) verfälschen die gesuchten Meßgrößen, weil unter der Reibwirkung der Meßkraftvektor  $\underline{f}_m$  nicht mehr senkrecht auf der Werkstückoberfläche steht. Diese Erscheinung wirkt sich zum einen ungünstig auf die Konturverfolgungsbewegung aus, indem die in Gleichung (3.2) berechnete Richtung  $\underline{e}_t$  der Vorschubbewegung, die immer senkrecht zum gemessenen Einheitsnormalenvektor der Werkstückoberfläche errechnet wird, durch die Wirkung des Meßfehlers vom Werkstück wegzeigt statt an seiner Oberfläche entlang. Dadurch wirkt die fehlerbehaftete

Vorschubbewegung störend auf die Kraftregelung ein. Zum anderen reduziert unangenehmerweise die Reibwirkung die erzielbare Genauigkeit bei der Konturbestimmung.

Bild 26 erläutert diesen Zusammenhang. Eine Sensorkugel umfährt die Spitze eines Werkstücks, wobei die in der Steuerung aufgezeichnete Bahn um den Abstand  $r_s$ , das ist der Radius der Sensorkugel, zum Werkstück verschoben erscheint. Nimmt man die Richtung  $\underline{e}_f$  der sich ausbildenden Meßkraft  $\underline{f}_m$  als (fehlerbehaftete) Normalenrichtung der unbekanntes Werkstückoberfläche an und korrigiert die Bahn mit den Maßen der Sensorspitze in dieser Richtung (s. Bild 26), so erhält man die gepunktet dargestellte Bahn, die offensichtlich gerade an Werkstückecken besonders große Fehler aufweist.



**Bild 26:** Einfluß der Reibung auf die Genauigkeit der Konturbestimmung

Zur Verbesserung des Programmiererergebnisses und gleichzeitigen Verringerung des Störeinflusses auf die Kraftregelung empfiehlt es sich, den reibungsbedingten Meßfehler rechnerisch zu korrigieren. Unter der Annahme, daß der Reibwinkel  $\lambda$  (s. Bild 26) über der Bewegung nur geringen Änderungen unterliegt, kann man den Einheitsvektor  $\underline{e}_f(j)$  der Meßkraft  $\underline{f}_m(j)$  um diesen Reibwinkel verdrehen, wobei der Normalenvektor der Schnittebene  $\underline{n}_0$  als Drehachse dient, und erhält dadurch einen gegenüber  $\underline{e}_f(j)$  verbesserten Einheitsvektor  $\underline{e}_0(j)$  für die Normalenrichtung der Werkstückoberfläche nach

$$\underline{e}_0(j) = \underline{R}(\underline{n}_0, \lambda) \underline{e}_f(j) . \quad (3.8)$$

Abschnitt 4.3.3 behandelt die Berechnung einer Drehmatrix vom Typ  $\underline{R}(\underline{n}_0, \lambda)$  für die allgemeine Lage der Drehachse im Raum (vgl. Gleichung (4.13)).

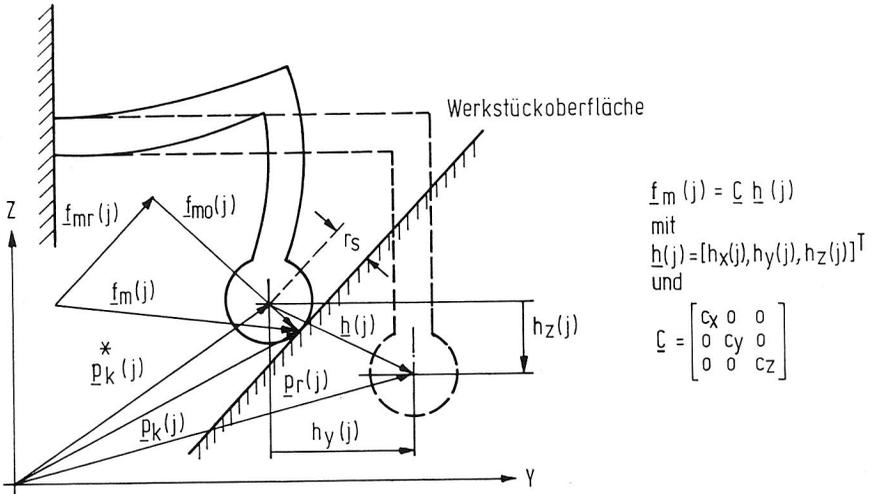
Aus Bild 26 geht weiterhin hervor, daß es zur Verringerung des Bahnfehlers grundsätzlich von Vorteil ist, den Tastkopfradius möglichst klein zu wählen. Der Gewinn an Genauigkeit bei der Konturbestimmung muß bei dieser Festlegung allerdings mit einer Verschlechterung der Abtasteigenschaften an rauhen Werkstückoberflächen erkauft werden (vgl. Abschnitt 3.2.3).

Nimmt man bei unverbogenem Sensor den Mittelpunkt der Kugel als Roboterposition  $\underline{p}_r(j)$  an, dann erscheint die Konturposition  $\underline{p}_k^*(j)$  in Bild 27 um den konstanten Abstand  $r_s$  zur tatsächlichen Kontur verschoben. Von der scheinbaren Konturposition  $\underline{p}_k^*(j)$  aus gelangt man zum wirklichen Konturberührungspunkt  $\underline{p}_k(j)$ , indem man die Strecke  $r_s$  in Richtung von  $\underline{e}_0(j)$  zurücklegt:

$$\underline{p}_k(j) = \underline{p}_k^*(j) + r_s \underline{e}_0(j) . \quad (3.9)$$

Aus Bild 27 ist ersichtlich, daß der Einfluß unterschied-

licher Elastizitäten in den Koordinatenrichtungen des Sensors zu unterschiedlichen Richtungen von Kraft- und



**Bild 27:** Sensorverformung unter Krafteinwirkung

Verbiegungsvektor führt. Die Verbiegung rührt von der wirksamen Verbiegungskraft  $\underline{f}_m$  her, zu deren Erfassung der Sensor dient. Es gilt:

$$\underline{f}_m(j) = \underline{C} \underline{h}(j) , \tag{3.10}$$

wobei sich der Verbiegungsvektor  $\underline{h}(j)$  als Differenz von Roboterposition  $p_r(j)$  und scheinbarer Konturposition  $p_k^*(j)$  mit

$$\underline{h}(j) = p_r(j) - p_k^*(j) \tag{3.11}$$

ausdrücken läßt. Formt man die Gleichungen (3.10) und (3.11) um und setzt sie in Gleichung (3.9) ein, erhält man eine Berechnungsvorschrift für die Konturposition

$$p_k(j) = p_r(j) - \underline{c}^{-1} f_m(j) + r_s e_0(j) , \quad (3.12)$$

die neben den konstanten Größen wie Kugeldurchmesser  $r_s$  und Elastizität  $\underline{c}$  auch die meßpunktabhängigen Größen  $p_r(j)$ ,  $f_m(j)$  und  $e_0(j)$  enthält.

### 3.2.5 Regelungskonzept

Bedingt durch die Art der Ersatzgrößen, die nach dem Meßkonzept (s. Abschnitt 3.2.1) die gesuchten Meßgrößen rekonstruieren sollen, hat die Regelung des Roboters zwei Aufgaben gleichzeitig zu erfüllen. Sie muß dafür sorgen, daß der Roboter eine durch das Werkstück definierte Bahn in der Schnittebene abfahren kann, und muß zusätzlich gewisse Vorgaben für die dabei am Sensor wirkenden Kräfte einhalten. Die Kräfte und Positionen, die sich bei der Bewegung einstellen, stehen über die Elastizität von Sensor und Roboter miteinander in Verbindung und sind deshalb natürlich nicht unabhängig voneinander einstellbar.

Zur Berücksichtigung beider Forderungen ist die Einführung einer hybriden Kraft-Positions-Regelung notwendig, wobei prinzipiell unterschiedliche Strukturvarianten möglich sind. PAUL [65] schlägt vor, die Aufgaben der Kraft- und der Positionsregelung auf die Achsen zu verteilen. Das bedeutet, daß, stellungsbezogen in Abhängigkeit von Werkstückgeometrie und Roboterkinematik, bestimmte Achsen die Kraft und die übrigen die Position regeln. Nach einem alternativen Regelungskonzept wird die aufgabenbezogene Achsumschaltung umgangen, und alle Achsen sind gleichermaßen an der Kraft- und der Positionsregelung beteiligt. Dazu haben WHITNEY [66], KUNTZE [59] und HIRZINGER [67] Kaskadenstrukturen verwendet, bei denen die Kraftregelung der Lageregelung überlagert ist. Kaskadenförmige Regelungen sind im Zusammenhang mit elektrischen Antrieben sehr ver-

breitet; üblicherweise sind der Lageregelschleife noch eine Drehzahl- und eine Stromregelschleife untergeordnet. Die Stellgrößen der überlagerten Regler werden dabei zu Sollwerten für die unterlagerten.

Grundsätzlich eignet sich das Strukturkonzept der Kaskadenregelung dazu, die Merkmale des in Abschnitt 3.2 neu eingeführten Programmierverfahrens abzubilden [68,69]. Obwohl diese Struktur den günstigsten Zugang zur hybriden Kraft-Positions-Regelung verspricht, weil seine Realisierung, aufbauend auf marktüblichen Positionsregelungen in industriellen Robotersteuerungen, relativ einfach möglich ist, wird hier eine andere Struktur vorgestellt. Sie vermeidet einige Nachteile der Kaskadenstruktur, indem sie die Auflösung der Kontaktkraft unabhängig macht von der Auflösung der Positionsgeber und dadurch auch für geringe Elastizitäten von Roboter und Sensor noch einen relativ glatten Kraftverlauf ermöglicht. Außerdem verhindert sie einen Dynamikverlust bei der Regelung, der bei Kaskadenstrukturen, wenn man einmal spezielle Zustandskonzepte [70] außer acht läßt, meist nicht zu vermeiden ist.

In einer gewissen Ähnlichkeit zu dem von PAUL [65] vorgeschlagenen Konzept sieht das neue Strukturkonzept eine Aufgabenteilung für die Regelungen der einzelnen Koordinaten vor. Begünstigt durch die Definition einer Schnittebene, dem Kernstück des Programmierverfahrens, ist es möglich, in der Richtung, die der Normalenvektor der Schnittebene festlegt, gänzlich auf eine Kraftregelung zu verzichten. Sinnvollerweise legt man ein aufgabenspezifisches Koordinatensystem  $E_s = (x_s, y_s, z_s)$  an, dessen z-Koordinate  $e_{z_s}$  mit dem Einheitsnormalenvektor  $\underline{n}_0$  der Schnittebene zusammenfällt. Für die z-Koordinate des aufgabenspezifischen Koordinatensystems reicht eine einfache Positionsregelung aus, die dafür sorgt, daß der Roboter nicht aus der Schnittebene herausfährt. Die Suchbewegung des Roboters um das Werkstück herum beschränkt sich auf diese Weise ausschließlich auf die

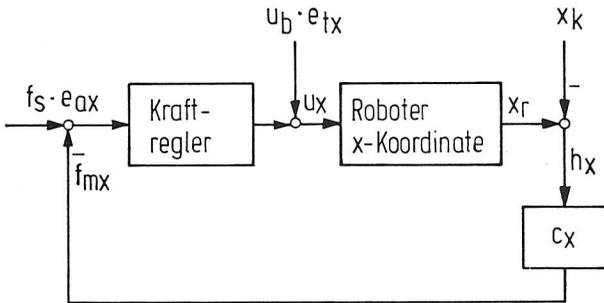
x- und y-Koordinaten des aufgabenspezifischen Koordinatensystems, auf das sich im übrigen, da keine Verwechslungsgefahr besteht, die weiteren Betrachtungen beziehen, ohne daß eine besondere Kennzeichnung vorgenommen wird.

Für den Fall linearer Elastizität stehen die physikalischen Größen Kontaktkraft und Position in einem konstanten Verhältnis zueinander, so daß die Regelungen beider Größen prinzipiell äquivalent sind. Demnach kann man gänzlich auf Lageregler für die beiden Koordinaten verzichten, die die Schnittebene aufspannen, und statt dessen Kraftregler direkt der Strom- und Geschwindigkeitsregelung überlagern. Faßt man die Mechanik und die beiden unteren Regelschleifen zu einer Einheit zusammen, so läßt sich der Roboter in jeder Koordinate als integrierendes System darstellen, das am Eingang mit einer geschwindigkeitsproportionalen Spannung  $u$  (Geschwindigkeitssollwert) angesteuert wird und am Ausgang die Position liefert.

Soll die Verarbeitung der beiden orthogonalen Führungsfunktionen für den Kraftausgleich und den Vorschub in analoger Weise erfolgen, müßte die Vorschubfunktion als zusätzlicher Kraftsollwert tangential zur Kontur ausgegeben werden. Da die Tangentialkraft aber ausschließlich auf den Reibeinfluß zurückgeht und nicht direkt mit der Geschwindigkeit, die man eigentlich einstellen will, korreliert ist, erhält man auf diese Weise keine definierte Regelgröße. Der Regelkreis ist dadurch praktisch offen, und der Vorschub erfolgt gesteuert. Das ist zwar grundsätzlich unproblematisch, da in der Regel konstante Geschwindigkeiten angestrebt werden; unangenehmerweise ist in diesem Fall aber die sich einstellende Vorschubdynamik von der Charakteristik des Kraftreglers abhängig.

Da der Einhaltung der Kraft im Hinblick auf einen ungestörten Programmierbetrieb größere Bedeutung beigemessen wird als einem exakten Geschwindigkeitsverlauf - das Abtasten des

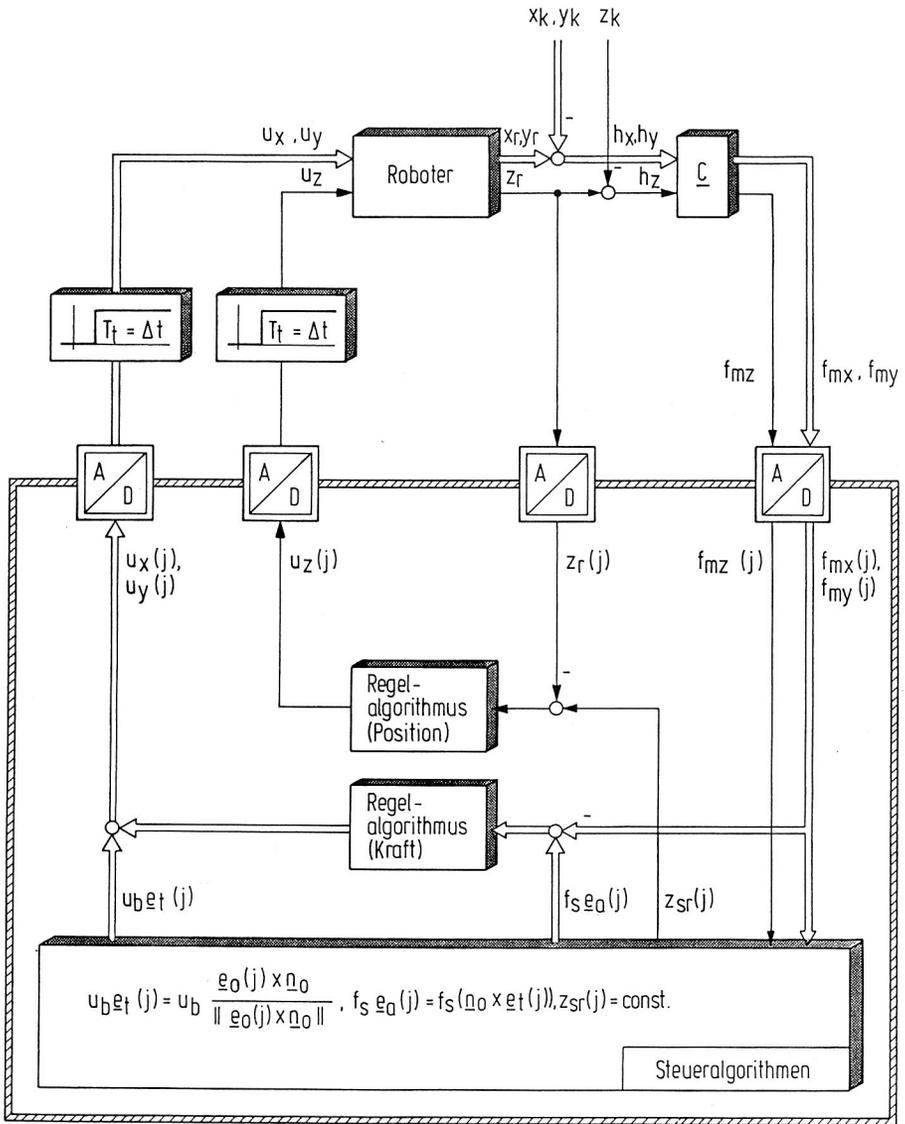
Werkstücks ist nur möglich mit anliegendem Sensor - wird eine Struktur gewählt, die aus einer Kraftregelung und einer Vorschubsteuerung zusammengesetzt ist, wobei die Vorschubaufschaltung nach dem Kraftregler erfolgt (s. Bild 28). Der Vorschub hat für den Kraftregelkreis damit den Charakter einer Störgröße. Aus Bild 28 ist erkennbar, daß die x-Komponente des Einheitsvektors  $\underline{e}_a$  (Gleichung (3.2)) den Anteil des vom Anwender gewählten Kraftsollwertes  $f_s$  für die x-Koordinate festlegt. Bei anliegendem Sensor führt die Positionsdifferenz zwischen Roboter ( $x_r$ ) und Kontur ( $x_k$ ) zur Sensorverbiegung  $h_x$ , die über die Elastizität  $c_x$  die Kraft



**Bild 28:** Kraftregelung mit Vorschubsteuerung für die x-Koordinate

$f_{mx}$  entstehen läßt. Der Richtungsvektor  $\underline{e}_t$  des Vorschubs (Gleichung (3.1)) gewichtet mit seiner x- und y-Komponente den Anteil des vom Anwender gewählten Geschwindigkeitswertes  $u_b$  für die jeweilige Koordinate.

Das Konzept der Verteilung von Kraft- und Positionsregelung auf die einzelnen Koordinaten ist in Bild 29 für eine Realisierung mit Mikrorechner dargestellt. Dabei wurde mit  $\underline{n}_0 = \underline{e}_z$  der Spezialfall gewählt, für den das aufgabenspezifische und das roboterspezifische Koordinatensystem unverdreht zueinander stehen. Aus diesem Grund entfällt eine Koordinatentransformation. Der Rechner bestimmt zu jedem

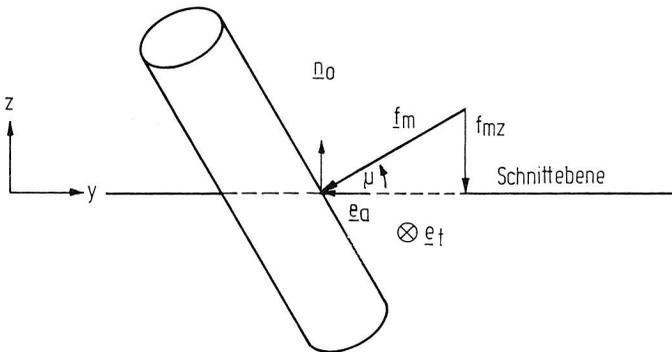


----- Systemgrenze Mikrorechner  $t = j \cdot \Delta t (j = 1, 2, \dots)$

**Bild 29:** Konzept der koordinatenverteilten Kraft- und Positionsregelung mit Mikrorechner für den Spezialfall  $n_0 = e_z$

Abtastzeitpunkt  $j$  aus den Meßgrößen  $f_{mx}(j)$ ,  $f_{my}(j)$  und  $f_{mz}(j)$  die Führungsfunktionen  $u_{b\hat{e}_t}(j)$  für den Vorschub und  $f_{s\hat{e}_a}(j)$  für den Kraftausgleich, die beide in die  $x,y$ -Ebene fallen. Für die  $z$ -Koordinate ist eine reine Positionsregelung vorgesehen, der ein konstanter Sollwert  $z_{sr}(j)$  aufgeschaltet wird. Dieser Wert gibt die Verschiebung der Schnittebene zum Nullpunkt des Roboterkoordinatensystems an. Um eine exakte Beschreibung der Abtastregelkreise zu erhalten, sind die verschiedenen Rechenzeiten der Steuer- und Regelalgorithmen in Totzeitgliedern zusammengefaßt, deren Zeitverzug  $T_t$  künstlich auf das Zeitintervall  $\Delta t$  zwischen zwei Abtastungen verlängert ist, damit A/D und D/A Wandler synchron arbeiten können. Für den Reglerentwurf lassen sich diese Totzeiten in die Streckenbeschreibungen für die drei Roboterkoordinaten einbeziehen [71].

Wie schon Bild 23 andeutet, können Werkstücke natürlich auch schräg zur Schnittebene liegen, sodaß die sich ausbildenden Meßkräfte auch  $z$ -Komponenten enthalten können. Für einen solchen Fall ist in Bild 30 der Blick in die Schnittebene hinein dargestellt. Erkennbar ist, daß trotz



**Bild 30:** Lage der Richtungsvektoren beim Blick in die Schnittebene

schräg stehenden Werkstücks die Aufgabe darin besteht, den Fühler in der festgelegten Ebene um das Werkstück zu führen. Kraftkorrekturrichtung  $\underline{e}_a$  und Vorschubrichtung  $\underline{e}_t$  liegen deshalb auch weiterhin in dieser Ebene, nicht so aber der Vektor der Meßkraft  $\underline{f}_m$ . Zerlegt man den Kraftvektor in seine drei Komponenten, ist erkennbar, daß ein Ausregeln der z-Komponente den Sensor unweigerlich aus der vorgegebenen Ebene herausführen würde. Sind die Richtungsvektoren der beiden Führungsfunktionen erst einmal berechnet, werden zur Ausregelung der Kraft lediglich die x- und y-Komponente benötigt. Daraus wird deutlich, daß für das oben eingeführte Programmierverfahren im Kern tatsächlich nur zwei Koordinaten notwendig sind, und die dritte ausschließlich zur Einhaltung von Randbedingungen dient.

Obwohl der z-Komponente der Kraft für die Kraftausregelung keine Bedeutung zukommt, kann sich die Kraftregelaufgabe nicht ausschließlich auf die x- und y-Komponente beschränken. Liegt das Werkstück nämlich sehr flach zur Schnittebene, wird die z-Komponente sehr viel größer als die beiden anderen Komponenten und der Sensor droht in z-Richtung überlastet zu werden, wenn der Sollwertbetrag  $f_s$ , der die Bezugslänge für die beiden verbleibenden Kraftkomponenten darstellt, nicht reduziert wird. Ist mit starken Schwankungen der z-Komponente der Kraft  $\underline{f}_m$  zu rechnen und der verwendete Fühler sehr empfindlich gegen Überlastung, kann es günstiger sein, anstelle des konstanten Kraftsollwertbetrages  $f_s$  (vgl. Bild 29) einem variablen Wert  $f_s(j)$  zu verwenden, der aus einem Kraftorientierungswert  $f_s^*$ , den der Anwender vorgibt, nach

$$f_s(j) = f_s^* \cos \mu(j) \quad (3.13)$$

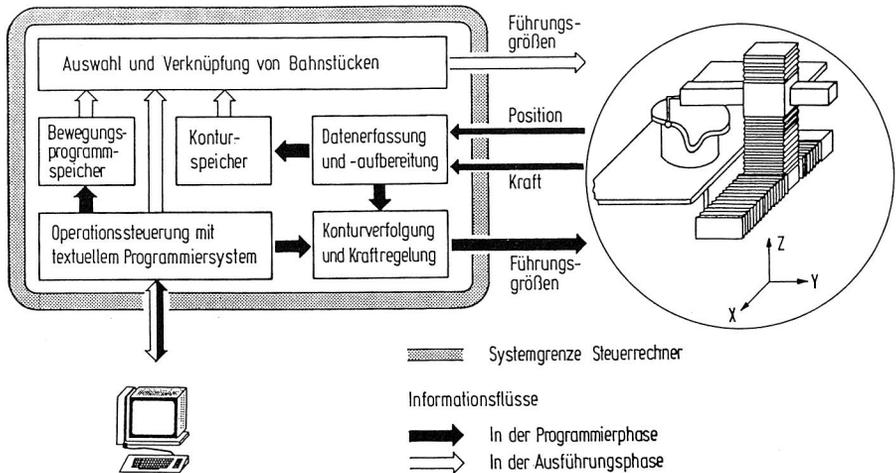
berechnet werden kann.

Prinzipiell eignet sich für das in Abschnitt 3.2 eingeführte sensorgestützte Programmierverfahren, das die Programmier-

bewegung aus zwei orthogonalen Führungsfunktionen erzeugt, sowohl das Strukturkonzept der kaskadierten Kraft-Positions-Regelung [59,66-69] als auch das der koordinatenverteilten Kraft-Positions-Regelung nach Bild 29. Für die erste Variante steckt erheblicher Rechenaufwand in der Berechnung der Richtungsvektoren für die Führungsfunktionen nach Gleichung (3.1) und (3.2) sowie in der Meßfehlerkorrektur nach Gleichung (3.8), deren Ergebnisse bei der zweiten Variante relativ einfach gefunden werden können. In diesem Fall steckt lediglich ein gewisser Rechenaufwand in den erforderlichen Koordinatentransformationen. In jedem Meßpunkt müssen nämlich der Meßkraftvektor  $\underline{f}_m(j)$  und der Positionsvektor  $\underline{p}_r(j)$  in das aufgabenspezifische Koordinatensystem transformiert werden, in dem die Berechnung aller aufgabenspezifischer Algorithmen erfolgt. Eine Rücktransformation bildet zum Schluß den resultierenden Stellvektor  $\underline{u}(j)$  wieder im Roboterkoordinatensystem ab.

### 3.2.6 Intergration in ein Programmiersystem

Zum Test des neuen sensorgestützten Programmierverfahrens wurde ein Experimentiersystem, bestehend aus einem dreiachsigen kartesischen Roboter und einem industriellen 16-Bit-Mikrorechnersystem aufgebaut. Bild 31 zeigt die Hauptfunktionen des Systems und die Informationsflüsse in den beiden Betriebsarten Programmierung und Programmausführung. Das System kann neben der Sensormethode auch mit einer interpretativen Programmiersprache industriellen Standards programmiert werden [72]. Die Steuerung speichert die beiden möglichen Typen von Bewegungsbahnen, entweder die eingelesenen Konturstücke oder textuell beschriebene Bahnen, in unterschiedlichen Modulen. Sie verknüpft die verschiedenen Bahnstücke erst bei der Ausführung miteinander.

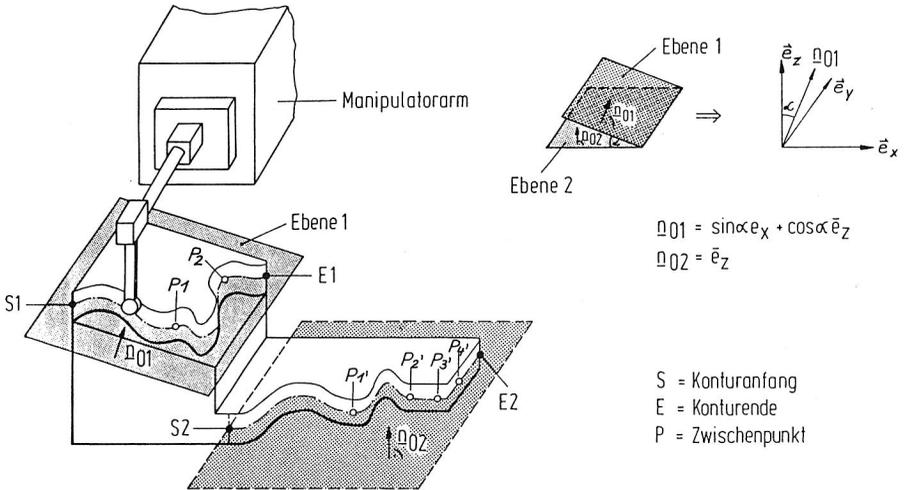


**Bild 31:** Informationsflüsse im Laborsystem in der Programmierungs- und Ausführungsphase

Auf der Oberfläche des Werkstücks in Bild 32 lassen zwei Schnittebenen  $n_{01}$  und  $n_{02}$  zwei verschiedene Konturen entstehen. Die im Bild als  $S_1$ ,  $E_1$  und  $S_2$ ,  $E_2$  bezeichneten Start- und Zielpunkte legt der Programmierer fest und bestimmt dadurch den Definitionsbereich der Kontur, die der Roboter gerade einlernt. Ebenfalls in der Einlernphase kann der Bediener zusätzlich Zwischenpunkte  $P_1$ ,  $P_2$  bzw.  $P'_1 \dots P'_4$  (s. Bild 32) markieren, an die bestimmte Steuerungsreaktionen gebunden werden können, die später in der Ausführungsphase zur Wirkung kommen. Beispielsweise lassen sich bestimmte Konturabschnitte auswählen, die Bahngeschwindigkeit an schwierigen Konturstücken verändern oder Werkzeuge ansteuern.

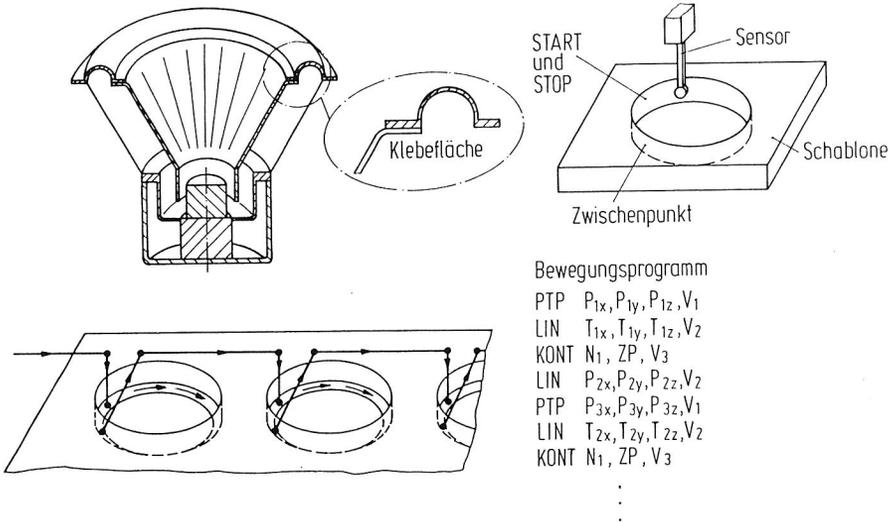
Innerhalb eines Bewegungsprogramms ruft der Befehl 'KONT' eine zuvor mit dem Kraftsensor eingelernte Kontur aus dem Speicher auf, wobei die Angabe von Zwischenpunkten  $P_i$  optional ist. Für den Fall, daß der Befehl zwei Punkte

enthält, ist nur der Teil der Kontur gültig, der zwischen diesen Markierungen liegt. Enthält der Befehl nur einen Punkt, fährt der Roboter die abgespeicherte Kontur vom Anfang bis zu der Stelle, an der der Zwischenpunkt gesetzt wurde.



**Bild 32:** Konturabtastung in verschiedenen Ebenen und mit Kennzeichnung von Zwischenpunkten

Bild 33 zeigt an einem Einsatzbeispiel, beim Kleben von Lautsprechermembranen, einen solchen Fall. Zwei Klebungen verbinden die Einspannsicke mit dem Lautsprecherkorb auf der einen und mit der Konusmembran auf der anderen Seite. Im Bild ist der Arbeitsschritt dargestellt und mit der Grobstruktur eines Bewegungsprogramms beschrieben, in dem eine Klebepistole Kleber auf die Membranen der palettierten Lautsprecher aufträgt. Dabei darf die Klebebahn nicht den vollen Membranumfang beinhalten, sie muß aus technologischen Gründen am Zielpunkt ZP enden. Andernfalls würde beim Aufsetzen der Sicke, in einem nachfolgenden Arbeitsgang, überschüssiger Kleber hervorquellen.



**Bild 33:** Sensorprogrammierung bei der Lautsprecherherstellung

### 3.3 Simulation des Programmiervorgangs

Die Programmierbewegung, die den Sensor auf der Kontur des Werkstückes umfahren läßt, ändert in Abhängigkeit von der Werkstückgeometrie beständig ihre Richtung. Bedingt durch die Berechnungsart der beiden Führungsfunktionen, die diese Bewegung erzeugen, treten dabei nichtlineare Verkopplungen zwischen den ansonsten als unabhängig und linear angenommenen Komponenten der Systemelemente 'Sensor', 'Regler' und 'Roboter' auf. Die Art der Verkopplung hängt maßgeblich vom Verlauf der Kontur ab. Im weiteren wird sich zeigen, daß bestimmte Sensoreigenschaften zu zusätzlichen Verknüpfungen der Koordinaten untereinander führen. Die Gesamtheit dieser nichtlinearen Erscheinungen macht die mathematische Behandlung des Gesamtsystems schwierig, insbesondere lassen sich Aussagen über die Auswirkungen von Störeinflüssen nicht

mit den handlichen Methoden der linearen Systemtheorie ableiten.

Gerade die Frage nach der Auswirkung von Störungen auf den Abtastvorgang ist aber bei dem neuen Programmierverfahren insofern von besonderer Bedeutung, als die Empfindlichkeit gegen Störungen insgesamt größer ist als bei anderen Verfahren der Sensorprogrammierung [10,11,56-62]. Im Gegensatz zu diesen existiert nämlich vor dem Start des Sensors kein Bewegungsprogramm. Die Steuer- und Regelalgorithmen berechnen unter Zuhilfenahme der Meßgrößen in jedem Abtastpunkt neu ein Bahnstück, das bis zum nächsten Abtastpunkt verfolgt wird. Da sich am praktischen Aufbau eine Vielzahl von Einflüssen in unbekannter Weise überlagert, sollen im weiteren Simulationsstudien den Einfluß einzelner Störgrößen auf die Programmierbewegung zeigen helfen. Die Erkenntnisse aus den Studien können dann in Verfeinerungen des Verfahrens ihren Niederschlag finden.

Der Sensor zusammen mit den beiden Führungsfunktionen bewirkt eine enge Verbindung zwischen Werkstück und Roboter. Da das Werkstück die Geometrie der unbekannt Roboterbahn vorgibt, kann der Roboter nicht als unabhängiges Objekt betrachtet werden. Ein mathematisches Gesamtmodell, das neben einem Modell für den Roboter ein eigenes Umweltmodell enthält, stellt deshalb die Basis der Simulation dar. Das Robotermodell umfaßt eine kontinuierliche Komponente, die die Dynamik des mechanischen Systems und der Antriebe beschreibt, und eine diskrete Komponente, die diejenigen Algorithmen beinhaltet, die im realen System der Mikrorechner ausführt. Das sind im Kern die Mechanismen der Kraftregelung und Vorschubsteuerung nach Bild 29. Da die Programmierbewegung ausschließlich in der Schnittebene erfolgt, reicht es aus, zwei Koordinaten für das Modell anzusetzen, ohne daß dadurch die Allgemeingültigkeit der Simulationsergebnisse beeinträchtigt wird. Gewählt werden wie in Bild 29 die x- und y-Koordinate; die Positionsrege-

lung in z-Richtung findet im Modell also keine Darstellung. Gegenstand des Robotermodells ist der gleiche dreiachsige kartesische Roboter (vgl. Bild 31), der auch für die praktischen Tests des Programmierverfahrens benutzt wurde.

### 3.3.1 Modellansatz für die kontinuierliche Komponente des Roboters mit Sensor

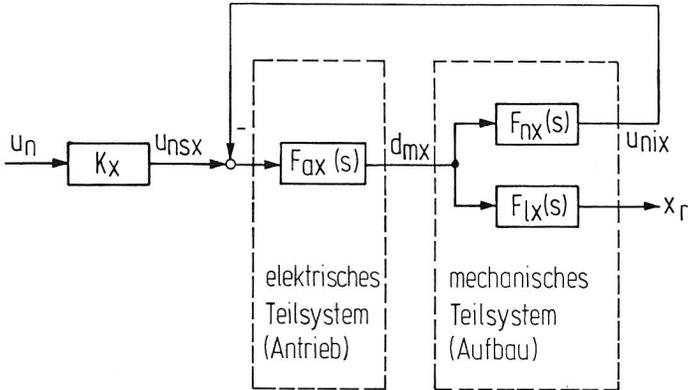
Im Sinne des Systembegriffs aus Kapitel 2.1.3 stellt ein Roboter ein künstliches, konkretes System dar, das sich durch seine Gestalt klar von seiner Umgebung abgrenzen läßt. Das System Roboter gehört zu den dynamischen Systemen, die sich dadurch auszeichnen, daß jeweils eine gewisse Verzögerungszeit vergeht, bis ein neuer Systemzustand erreicht ist. Um sein Systemverhalten mathematisch beschreibbar zu machen, wird ein Modell erstellt, das bezüglich des Bewegungsverhaltens den realen Roboter in seinen wesentlichen Eigenschaften abbildet. Prinzipiell ist nach der Arbeitsmethode der Systemtechnik zur Modellierung die Kenntnis der inneren Systemstruktur nicht erforderlich, da das Verhalten von Systemen ausschließlich durch die Verbindungsgrößen zwischen Systemelementen und der Umgebung, also durch Ein- und Ausgangsgrößen, beschrieben werden kann [25]. Trotzdem kann es vorteilhaft sein, Modelle zu verwenden, die die Struktur des mechanischen und elektrischen Systems annähern, u.a. deshalb, weil die Diskussion der Vorgänge unter verschiedenen Randbedingungen anschaulicher wird. Aus diesem Grunde wird im weiteren ein mathematisches Robotermodell erstellt, das die verzögernden Eigenschaften des elektrischen Antriebs und die trägen und elastischen Eigenschaften des mechanischen Aufbaus berücksichtigt.

Nähere Untersuchungen des elektromechanischen Systems Roboter führen auf eine Anzahl miteinander in Verbindung stehender physikalischer Prozesse, für die durch Auswertung von Energiebilanzen ein sogenanntes Strukturmodell erstellt

werden kann. Darunter versteht man die mathematische Beschreibung des Systemverhaltens bei Erfassung derjenigen Teilsysteme, die zur Erzielung der geforderten Modellgenauigkeit berücksichtigt werden müssen. Die Problematik bei dieser Vorgehensweise liegt darin, daß nicht immer von vornherein ausgesagt werden kann, wie genau das Modell für den vorgesehenen Verwendungszweck sein muß und welche Konsequenzen bestimmte Vernachlässigungen zur Folge haben. Diese Unsicherheit zwingt dazu, das Modell zu erproben und dabei zu prüfen, ob Verbesserungen erforderlich - oder umgekehrt - Vereinfachungen möglich sind. Es muß bezüglich der Komplexität der Modellstruktur abgewogen werden zwischen dem Wunsch nach möglichst exakter Nachbildung der Realität und dem dazu erforderlichen Aufwand zur Erfassung der zugehörigen Modellparameter. Methodisch wurde deshalb zunächst ein relativ detaillierter Strukturansatz gewählt und dieser durch Überprüfung der Relevanz der Teilprozesse für den geplanten Verwendungszweck stufenweise vereinfacht.

Bei komplexen Modellstrukturen ist es nicht mehr ohne weiteres möglich, Ursachen für Abweichungen zwischen Modell- und Systemverhalten zu lokalisieren. Insbesondere ist es schwierig, zwischen Mängeln in der Modellstruktur und unzureichender Genauigkeit einzelner Modellparameter zu unterscheiden. Es mußten deshalb möglichst überschaubare Teilsysteme gefunden werden, die getrennt ausgemessen werden konnten. Da am Roboter aber nicht beliebig Meßstellen zugänglich waren, blieb als Ausweg übrig, im elektromechanischen Gesamtsystem jeder Achse die Teilsysteme Antrieb und Mechanik zu unterscheiden und getrennt zu untersuchen. Dadurch konnten die Auswirkungen einzelner Parameter auf das Systemverhalten genauer studiert und Meßungenauigkeiten ausgeglichen werden. Bild 34 zeigt die Aufteilung des Gesamtsystems, dessen zwei Blöcke das Motordrehmoment  $d_{mx}$  miteinander verbindet. Eine Umrechnung des Strukturmodells überführt die Zustandsgleichungen, die das Systemverhalten der Einzelblöcke wie auch des Gesamtsystems beschreiben, in

Übertragungsfunktionen. Diese Darstellungsart hat den Vorteil, daß die Qualität der Beschreibung durch Messung des Amplituden- und Phasengangs am Gerät relativ einfach geprüft werden kann.



**Bild 34:** Bestimmung des Übertragungsverhaltens einer Roboterachse aus mehreren Übertragungsfunktionen am Beispiel der x-Achse

Zwei Übertragungsfunktionen  $F_{nx}(s)$  und  $F_{lx}(s)$  beschreiben das Bewegungsverhalten des mechanischen Aufbaus, wobei die Ausgangsgröße  $x_r$  die Position des Achskopfes und die Ausgangsgröße  $u_{nix}$  die geschwindigkeitsproportionale Spannung, die der Tachogenerator liefert, darstellt. Die letztere führt an den Eingang des Antriebsverstärkers zur Berechnung der Regeldifferenz des Geschwindigkeitsregelkreises zurück. Da das Motordrehmoment selbst nur durch den Einbau einer Meßwelle direkt hätte bestimmt werden können, wurde der Ankerstrom des Motors erfaßt, und daraus durch Multiplikation mit der Drehmomentkonstanten das Drehmoment errechnet. Umgekehrt wurden die Drehmomentverläufe für die Messungen am mechanischen Teilsystem durch Einspeisung in den Eingang des Stromreglers erzeugt. Damit errechnet sich die Gesamtübertragungsfunktion der x-Achse aus

$$F_{gx}(s) = \frac{x_r(s)}{u_x(s)} = K_x \frac{F_{ax}(s) F_{lx}(s)}{1 + F_{ax}(s) F_{nx}(s)} . \quad (3.14)$$

Vorraussetzung für die Arbeit mit Übertragungsfunktionen ist die ausschließliche Verwendung linearer Modellelemente. Von den markanten nichtlinearen Eigenschaften des mechanischen Aufbaus, die Getriebe, Lagerung und Führungen verursachen, konnte das Umkehrspiel vernachlässigt werden, es fällt mit nur ca. 1/100 mm sehr gering aus; die Reibung mußte im interessierenden Geschwindigkeitsbereich linearisiert werden.

Im Antriebsteil sind Sollwertführungen für die Stromregelschleife sowie verschiedene Begrenzungen zum Schutz von Motor und Leistungsstufe enthalten, deren Wirkung bei größeren Stellamplituden eintritt. Ausgehend von der Annahme, daß an glatten Konturen, mit denen die Simulation im wesentlichen arbeiten soll, kleine Stellamplituden auftreten, bleiben sämtliche nichtlinearen Elemente des Antriebsverstärkers unberücksichtigt.

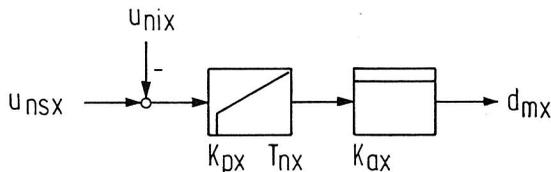


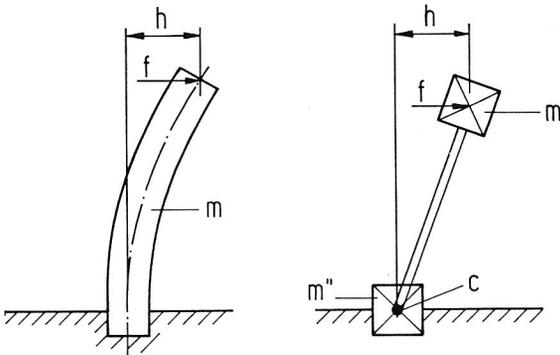
Bild 35: Modell des Antriebs für die x-Achse

Ebenfalls unberücksichtigt bleiben kann die Rückwirkung der EMK auf die Ankerspannung, die im Ankerstromkreis den Charakter einer Störgröße hat, da das Störverhalten der Stromregelung wesentlich schneller ist als das Zeitverhalten der EMK-Schleife. Nach Zusammenfassung der kleinen Zeitkonstanten läßt sich das Zeitverhalten der Stromregelschleife recht gut durch eine Ersatzzeitkonstante erster Ordnung

annähern [73], für die Werte von 1 - 1,5 ms gemessen wurden. Das entspricht Eckfrequenzen von ca. 100 bis 160 Hz . Im Vergleich zur Dynamik des mechanischen Aufbaus, dessen zu berücksichtigende Eigenfrequenzen zwischen 8 und 35 Hz liegen, kann die Stromregelschleife deshalb als P- Glied betrachtet werden. Bild 35 zeigt das reduzierte Strukturbild des Antriebs nach diesen Vereinfachungen. Die Übertragungsfunktion des Antriebsblocks für das Beispiel der x-Achse ergibt sich dann zu

$$F_{ax}(s) = \frac{d_{mx}}{u_{nsx} - u_{nix}} = \frac{K_{ax} K_{px} (1 + s T_{nx})}{s T_{nx}}. \quad (3.15)$$

Den prinzipiellen mechanischen Aufbau des Roboters zeigt Bild 37 mit Blick auf die x,z-Ebene. Auch dieser Systemteil ist mit konzentrierten Parametern wesentlich einfacher beschreibbar. Die Linearachsen, aus denen der Roboter aufgebaut ist, können als einseitig eingespannte Biegebalken betrachtet werden, deren Verbiegungsverhalten sich durch eine einfache Ersatzanordnung [74] annähern läßt. Zwei Punktmassen, die durch einen masselosen Stab miteinander verbunden sind (Bild 36), ersetzen dabei den Balken, wobei



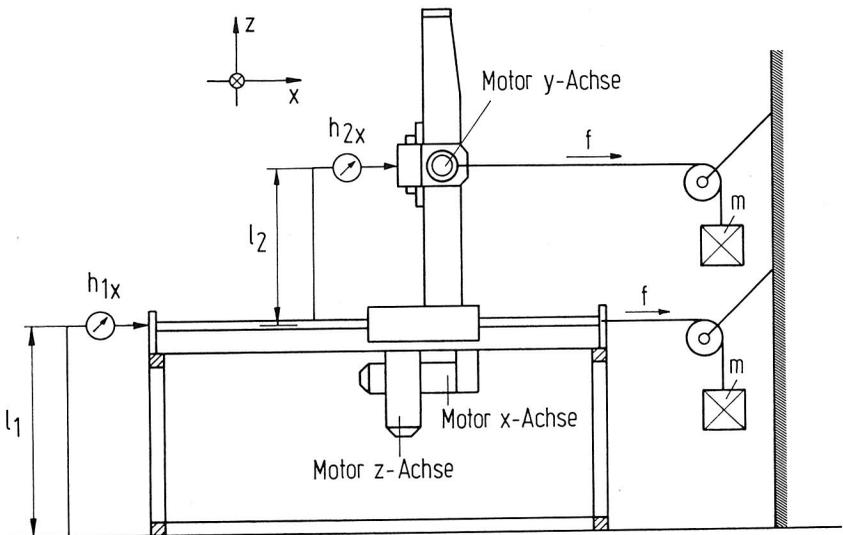
**Bild 36:** Einseitig eingespannter Biegebalken und Modell mit konzentrierten Parametern

die Einspannseite die Elastizität konzentriert enthält. Sollen bei Annahme von gleicher kinetischer Energie beide Systeme synchrone Schwingungen ausführen, ist die folgende Masseverteilung erforderlich [74]:

$$m' = \frac{33}{140} m, \quad m'' = m - m' . \quad (3.16)$$

Aus dem Verhältnis von Kraft  $f$  und Verbiegung  $h$  erhält man die Elastizität  $c$ , die, aus Gründen der Übersichtlichkeit, ein ausgefüllter Vollkreis im Einspannpunkt repräsentiert (vgl. Bild 36), obwohl das gesamte mechanische Modell keine Drehelastizitäten enthält. Die Elastizitäten haben die Einheit  $N/m$ .

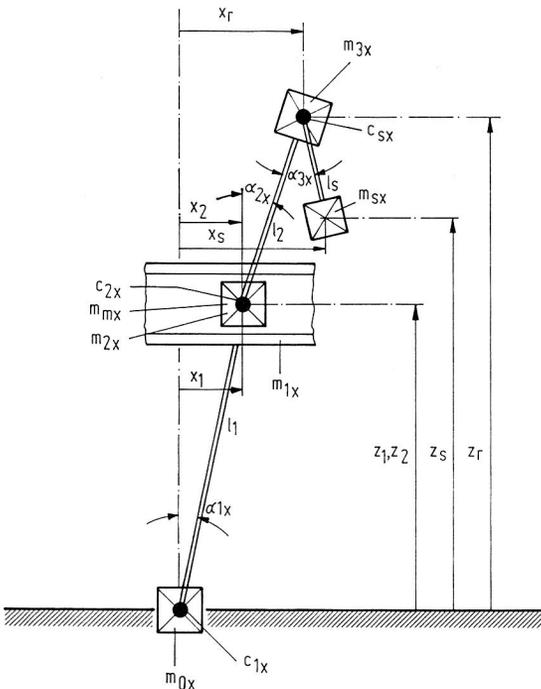
Bild 37 zeigt die Meßanordnung zur Bestimmung der Elastizitäten der  $x$ -Achse. Nacheinander wirkt eine statische Kraft  $f$  auf Sockel und Vertikalarm, wobei eine Meßuhr die jewei-



**Bild 37:** Mechanischer Aufbau des Roboters mit Meßanordnung zur Bestimmung der Elastizitäten [77]

ligen Verbiegungen der Konstruktionselemente anzeigt. Die Betrachtung der Masseverteilung des mechanischen Aufbaus läßt vermuten, daß sich im wesentlichen zwei Schwingungen bei Anregung der x- oder y-Achse ausbilden werden (vgl. Bild 37). Die drei Achsen sind in Tischhöhe auf einer Stahlrohrkonstruktion aufgebaut, die sich unter Einwirkung der resultierenden Kräfte meßbar verwindet, weiterhin ist auf den Schlitten der x-Achse der Turm aus y- und z-Achse aufgebaut, der in seiner Gesamtheit zu Schwingungen neigt. Eine dritte Schwingung ist vom Sensor selbst zu erwarten, soweit ihn das Werkstück nicht an der Ausbildung freier Schwingungen hindert.

Im Modellansatz der x-Achse (Bild 38) ist für jede der drei Schwingungen je ein Massepunkt und eine Elastizität vor-



Koordinaten der Massepunkte

$$x_1 = l_1 \sin \alpha_{1x}$$

$$x_2 = x_2$$

$$x_r = x_2 + l_2 \sin \alpha_{2x}$$

$$x_s = x_r + l_s \sin (\alpha_{3x} - \alpha_{2x})$$

$$z_1 = l_1 \cos \alpha_{1x}$$

$$z_2 = z_1$$

$$z_r = z_1 + l_2 \cos \alpha_{2x}$$

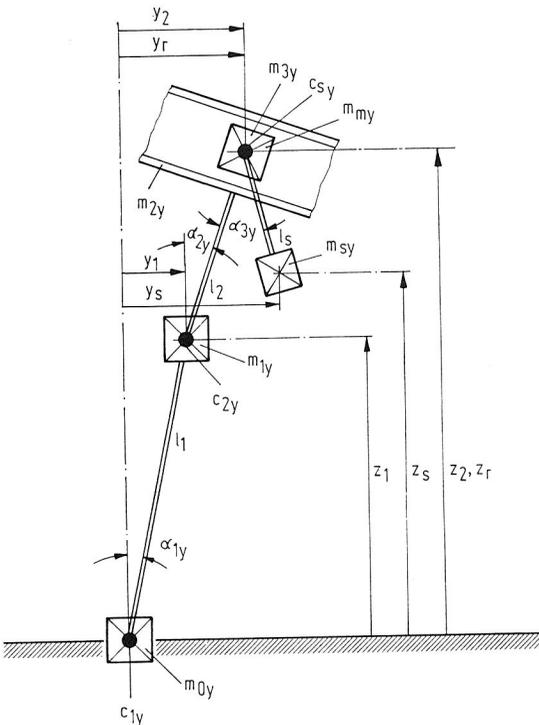
$$z_s = z_r - l_s \cos (\alpha_{3x} - \alpha_{2x})$$

(3.17)

**Bild 38:** Modellansatz für die Mechanik der x-Achse

gesehen. Entsprechend der Gleichungen (3.16) teilt man die Masse auf in einen schwingenden und einen fixierten Anteil.

Die Massen  $m_{0x}$  und  $m_{1x}$  sind der Sockelschwingung zugeordnet, wobei der Masse  $m_{1x}$  die Schlittenführung zugeschlagen ist. Die Masse  $m_{2x}$  enthält den Schlitten und den Anteil des Vertikalarms, der die Turmschwingung nicht mit ausführt,  $m_{3x}$  den schwingenden Vertikalarmanteil zuzüglich Führungskopf und Horizontalarm. Da die Sensormasse sehr klein gegenüber den Roboter Massen ist, wird der nichtschwingende Masseanteil vernachlässigt, übrig bleibt die schwingende Masse  $m_{sx}$ . Am gleichen Ort wie  $m_{2x}$  wird das auf Linearbewegung umgerechnete Trägheitsmoment von Antriebsmotor und Kraftübertragung als  $m_{mx}$  angesetzt. Die Positionen der



Koordinaten der Massepunkte

$$\begin{aligned}
 y_1 &= l_1 \sin \alpha_{1y} \\
 y_2 &= y_1 + l_2 \sin \alpha_{2y} \\
 y_r &= y_r \\
 y_s &= y_r + l_s \sin (\alpha_{3y} - \alpha_{2y}) \\
 z_1 &= l_1 \cos \alpha_{1y} \\
 z_2 &= z_1 + l_2 \cos \alpha_{2y} \\
 z_r &= z_2 \\
 z_s &= z_r - l_s \cos (\alpha_{3y} - \alpha_{2y})
 \end{aligned}$$

(3.18)

Bild 39: Modellansatz für die Mechanik der y-Achse

einzelnen Massepunkte lassen sich unter Beachtung der einschränkenden Bedingungen durch Führungen und Verbindungsstäbe mit den Gleichungen (3.17) angeben.

Die y-Achse bildet bei Anregung ganz ähnliche Schwingungen aus, im Unterschied zur x-Achse sitzt der Antrieb aber jetzt auf dem Schlitten der y-Achse. Bild 39 zeigt den entsprechenden Modellansatz. Für die Sockelschwingung ergibt sich eine vergleichbare Masseaufteilung wie bei der x-Achse, die Masse  $m_{1y}$  enthält allerdings zusätzlich den Anteil des Vertikalarms, der nicht an der Turmschwingung beteiligt ist. Die Masse  $m_{2y}$  enthält den schwingenden Vertikalarmanteil und den Führungskopf und  $m_{3y}$  den Horizontalarm. Die Gleichungen (3.18) geben die Positionen der Massepunkte an.

Verkoppelungserscheinungen der Schwingungen in der x- und y-Koordinate untereinander wurden im Hinblick auf die Handlichkeit des Modells vernachlässigt.

Das Gerät ist relativ steif aufgebaut und es können sehr kleine Torsionswinkel erwartet werden, für die die Näherungen  $\sin \alpha \approx \alpha$  und  $\cos \alpha \approx 1$  zulässig sind. Mit diesen Näherungen lassen sich die Gleichungen (3.17) und (3.18) linearisieren. Das ist Voraussetzung dafür, daß die LAGRAN-GEschen Gleichungen 2. Art, die im weiteren benutzt werden, auf ein lineares Differentialgleichungssystem führen. Außerdem vereinfacht sich die Modellstruktur nochmals, da mit den obigen Näherungen auch die Verkoppelungen von x- und y-Achse mit der z-Achse vernachlässigt werden können.

Um den Grad des Modells sowie die Zahl seiner Parameter klein zu halten, kann eine Abschätzung der Eigenfrequenzen der Feder-Masse-Schwinger nach

$$\omega = \sqrt{\frac{c}{m}} \quad (3.19)$$

wertvolle Hinweise geben. Nach dieser Abschätzung ist die

Sockelschwingung gegenüber der Turmschwingung vernachlässigbar, da ihre Frequenz etwa den doppelten Wert erreicht. Damit werden  $\alpha_{1x}$  und  $\alpha_{1y}$  zu Null. Nach den vorangegangenen Vereinfachungen lassen sich die verbleibenden freien Parameter aus den Gleichungen (3.17) und (3.18) in den Koordinatenvektoren  $\underline{q}_x$  für die x-Achse, und  $\underline{q}_y$  für die y-Achse anordnen:

$$\underline{q}_x = (x_2, x_r, x_s)^T, \quad (3.20)$$

$$\underline{q}_y = (y_2, y_r, y_s)^T. \quad (3.21)$$

### 3.3.2 Ableitung der Bewegungsgleichungen

Die zur Beschreibung des Roboters dienenden Ersatzmodelle stellen Systeme von Punktmassen dar. Bei der Ableitung der zugehörigen Bewegungsgleichungen treten zu den  $3n$  Zeitfunktionen, die die Bewegungen der  $n$  Massenpunkte in den drei kartesischen Koordinaten beschreiben, durch den Einfluß von Führungen, Lagern, Verbindungsstäben etc. zusätzlich  $m$  Zwangsbedingungen, die die Lösung des Systems aus  $3n+m$  Gleichungen erschweren. Interessiert man sich von vornherein nur für die Bewegung, empfiehlt es sich, auf die Beschränkungen der Bewegungsfreiheit Rücksicht zu nehmen und nur so viele 'unabhängige Koordinaten' zu verwenden, wie zur Beschreibung des Zustandes des Systems nötig sind. Besitzt das System  $s$  Freiheitsgrade, bleiben  $3n-m=s$  Koordinaten willkürlich wählbar.

Die LAGRANGESchen Gleichungen zweiter Art für nicht-konservative holonome Systeme sind in [75] in folgender Formulierung angegeben:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i^* \quad (i = 1, 2, \dots, f). \quad (3.22)$$

Die linke Seite enthält die verallgemeinerten Kräfte des konservativen Systemteils, während die  $Q_i^*$  der rechten Seite kein Potential besitzen. Schreibt man Gleichung (3.22) vektoriell und setzt für die potentialfreien Kräfte  $Q^*$  die von den Motoren herrührenden Antriebskräfte  $\underline{f}_a$  sowie die im wesentlichen durch Reibung verursachten Dämpfungskräfte  $\underline{f}_d$  ein, so erhält man

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\underline{q}}} - \frac{\partial L}{\partial \underline{q}} = \underline{f}_a - \underline{f}_d . \quad (3.23)$$

Darin ist  $L=T-V$  die LAGRANGE-Funktion, die für das betrachtete mechanische System des Roboters mit  $T=T(\dot{\underline{q}})$  die kinetische und mit  $V=V(\underline{q})$  die potentielle Energie der eingepprägten Kräfte und Momente enthält. In den unabhängigen Koordinaten  $\underline{q}$  ausgedrückt, läßt sich die skalare LAGRANGE-Funktion damit anschreiben als:

$$L = \frac{1}{2} ( \dot{\underline{q}}^T \underline{M} \dot{\underline{q}} - \underline{q}^T \underline{C} \underline{q} ) . \quad (3.24)$$

Da  $\underline{M}$  und  $\underline{C}$  konstante Matrizen sind und die Dämpfungskräfte  $\underline{f}_d$  als geschwindigkeitsproportional angenommen werden, gilt

$$\begin{aligned} \frac{d}{dt} \frac{\partial L}{\partial \dot{\underline{q}}} &= \frac{d}{dt} \frac{\partial T(\dot{\underline{q}})}{\partial \dot{\underline{q}}} = \underline{M} \ddot{\underline{q}} , \\ \frac{\partial L}{\partial \underline{q}} &= - \frac{\partial V(\underline{q})}{\partial \underline{q}} = - \underline{C} \underline{q} \end{aligned} \quad (3.25)$$

und  $\underline{f}_d = \underline{D} \dot{\underline{q}}$  .

Eingesetzt in die Gleichung (3.23) erhält man damit die Bewegungsgleichung in folgender Formulierung:

$$\underline{M} \ddot{\underline{q}} + \underline{C} \underline{q} = \underline{f}_a - \underline{D} \dot{\underline{q}} . \quad (3.26)$$

Nach den Beschleunigungen aufgelöst lautet sie

$$\ddot{\underline{q}} = - \underline{M}^{-1} \underline{C} \underline{q} - \underline{M}^{-1} \underline{D} \dot{\underline{q}} + \underline{M}^{-1} \underline{f}_a . \quad (3.27)$$

Ersetzt man  $m_{mx}$  durch das auf Linearbewegung umgesetzte Trägheitsmoment von Motor und Getriebe, so erhält man nach den obigen Vernachlässigungen mit dem Vektor der unabhängigen Koordinaten  $\underline{q}_x$  die kinetische Energie der x-Achse nach:

$$\begin{aligned} T_x &= \frac{1}{2} (m_{2x} \dot{x}_2^2 + m_{3x} \dot{x}_r^2 + m_{mx} \dot{x}_2^2 + m_{sx} \dot{x}_s^2) \\ T_x &= \frac{1}{2} \dot{\underline{q}}_x^T \underline{M}_x \dot{\underline{q}}_x \end{aligned} \quad (3.28)$$

mit  $\underline{M}_x = \text{Diag}(m_{2x} + m_{mx}, m_{3x}, m_{sx})$  .

Für die potentielle Energie der Federelemente ergibt sich

$$\begin{aligned} V_x &= \frac{1}{2} (c_{2x} (x_r - x_2)^2 + c_{sx} (x_s - x_r)^2) \\ V_x &= \frac{1}{2} \underline{q}_x^T \underline{C}_x \underline{q}_x \end{aligned} \quad (3.29)$$

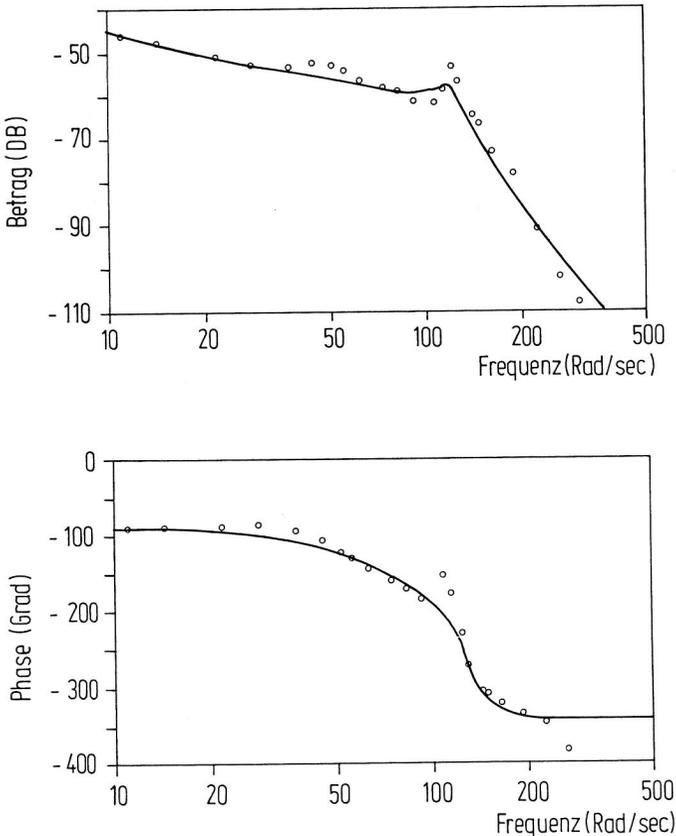
$$\text{mit } \underline{C}_x = \begin{pmatrix} c_{2x} & -c_{2x} & 0 \\ -c_{2x} & c_{sx} + c_{2x} & -c_{sx} \\ 0 & -c_{sx} & c_{sx} \end{pmatrix} .$$

Die zur Beschleunigung des Aufbaus verfügbare Kraft, also diejenige Kraft  $f'_{ax}$ , die nach Abzug der zur Überwindung von Verlusten aufzubringenden Kraft von der Motorkraft verbleibt, greift in Richtung von  $x_2$  an  $m_{2x}$  an, der Vektor  $\underline{f}_{ax}^T$  lautet deshalb

$$\underline{f}_{ax}^T = (f'_{ax}, 0, 0) . \quad (3.30)$$

Die dämpfenden Eigenschaften des Roboters sind in Gleichung (3.25) als geschwindigkeitsproportional angesetzt. Jeder Elastizität wird formal eine Dämpfung zugeordnet, deren Festlegung mit meßtechnischen Mitteln aber keine gut reproduzierbaren Ergebnisse liefert. Der Dämpfungsparameter  $d_{2x}$ ,

der der Elastizität  $c_{2x}$  zugeordnet ist, dient deshalb als 'Abstimmungsinstrument' für das Übertragungsverhalten des Achsmodells. Bild 40 zeigt die erzielte Qualität anhand des Frequenzgangvergleichs mit der realen Roboterachse. Die Kreise im Amplituden- und Frequenzgang sind die durch Frequenzanregung gefundenen Meßpunkte, die durchgezogene Linie steht für das Modellverhalten, das die Übertragungsfunktion  $F_{gx}(s)$  nach Verknüpfung (s. Bild 34) der Einzelübertragungsfunktionen  $F_{ax}(s)$ ,  $F_{1x}(s)$  und  $F_{nx}(s)$  nach Gleichung (3.14) liefert.



**Bild 40:** Frequenzgang  $F_{gx}(s)$  der x-Achse  
(o Meßpunkte, - Modellverhalten) [77]

Da der Sensor zu den variablen Elementen der Versuchsanordnung gehört, ist es nicht sinnvoll, ihn in die Übertragungsfunktion  $F_{gx}(s)$  einzubinden. Demzufolge steht die Sensordämpfung  $d_{sx}$  auch nicht als Abstimmungsparameter zur Verfügung und muß ersatzweise experimentell durch Ausschwingversuche festgelegt werden. Die Auswertemethode geht aus Bild 41 hervor, wobei sich die Abklingkonstante nach [76] aus

$$\delta = \frac{d_s}{2 m_s} \quad (3.31)$$

berechnen läßt. Darin ist  $m_s$  der für die jeweilige Koordinatenrichtung gültige schwingungsfähige Masseanteil des Sensors. Aus dem Verhältnis zweier aufeinanderfolgender Ampli-

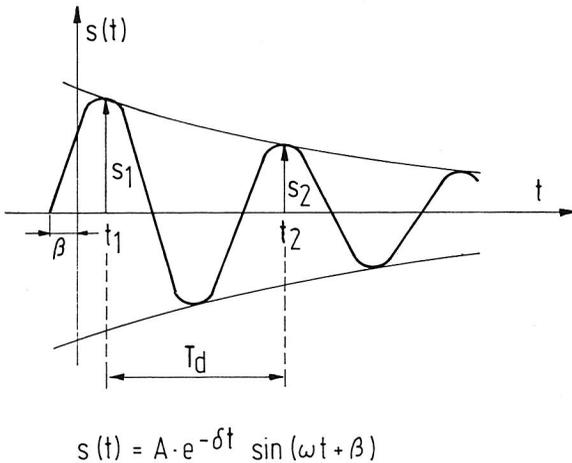


Bild 41: Dämpfungsbestimmung beim Ausschwingversuch

tuden läßt sich unter Verwendung von Gleichung (3.31) und der Periodendauer  $T_d$  der Schwingung die Dämpfung  $d_s$  des Sensors für jede Koordinatenrichtung getrennt berechnen:

$$d_s = 2 \frac{m_s}{T_d} \ln \frac{s_1}{s_2} . \quad (3.32)$$

Mit den oben getroffenen Vernachlässigungen verbleiben auch für die y-Achse drei unabhängige Koordinaten, die der Koordinatenvektor  $\underline{q}_y$  (vgl. Gleichung 3.21) enthält. Damit erhält man die kinetische Energie der y-Achse

$$\begin{aligned} T_y &= \frac{1}{2} (m_{2y} \dot{Y}_2^2 + m_{3y} \dot{Y}_r^2 + m_{my} (\dot{Y}_r - \dot{Y}_2)^2 + m_{sy} \dot{Y}_s^2) \\ &= \frac{1}{2} \underline{\dot{q}}_y^T \underline{M}_y \underline{\dot{q}}_y \end{aligned} \quad (3.33)$$

$$\text{mit } \underline{M}_y = \begin{pmatrix} m_{2y} + m_{my} & -m_{my} & 0 \\ -m_{my} & m_{3y} + m_{my} & 0 \\ 0 & 0 & m_{sy} \end{pmatrix}$$

und die potentielle Energie der Federelemente

$$\begin{aligned} V_y &= \frac{1}{2} (c_{2y} Y_2^2 + c_{sy} (Y_s - Y_r)^2) \\ V_y &= \frac{1}{2} \underline{q}_y^T \underline{C}_y \underline{q}_y \end{aligned} \quad (3.34)$$

$$\text{mit } \underline{C}_y = \begin{pmatrix} c_{2y} & 0 & 0 \\ 0 & c_{sy} & -c_{sy} \\ 0 & -c_{sy} & c_{sy} \end{pmatrix} .$$

Die nach Abzug der Verluste verbleibende Antriebskraft  $\underline{f}_{ay}^T$  greift in Richtung von  $y_r$  am Schlitten und in negativer  $y_2$ -Richtung an der Führung an. Der Kraftvektor ergibt sich dann zu

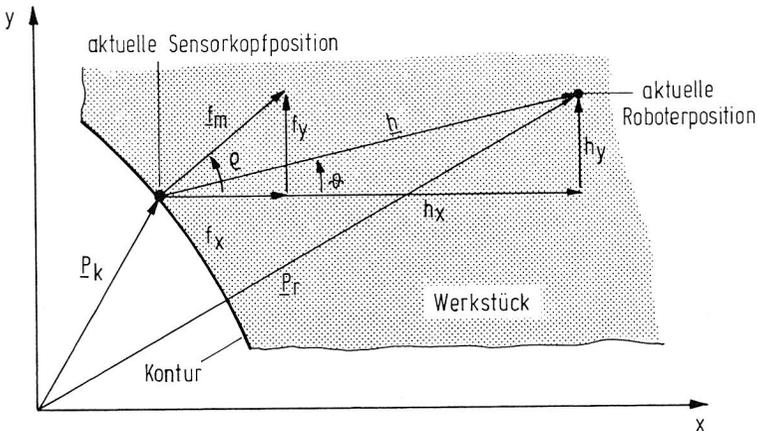
$$\underline{f}_{ay}^T = (-f'_{ay}, f'_{ay}, 0) . \quad (3.35)$$

### 3.3.3 Umweltmodell

Die sensorgestützten Programmierverfahren beruhen auf der Idee, die Gestalt von Werkstücken als mechanische Geometriespeicher nutzbar zu machen. Aufgrund des in Abschnitt 3.2.1 festgelegten Meßkonzepts ist der Werkstoff relativ beliebig, sofern nur die Verformung des Werkstücks unter dem Einfluß der Meßkraft ausreichend klein und seine Oberfläche grundsätzlich für taktile Meßverfahren geeignet ist. Das Umweltmodell kann sich demnach auf das abzutastende Werkstück beschränken, wobei allein seine Geometriebeschreibung von Interesse ist. Die besondere Eigenschaft der ebenen Werkzeugbewegungen, auf deren Voraussetzung das hier betrachtete Programmierverfahren beruht, hat im vorangegangenen bereits zu einem zweiachsigen Ansatz für das Robotermodell geführt, sodaß auch das Umweltmodell in zwei Koordinaten ausreichend repräsentiert erscheint. Die Geometriebeschreibung des Werkstücks schrumpft damit auf diejenige ebene Kontur zusammen, die die Schnittebene auf der Werkstückoberfläche abbildet.

Verbindendes Element zwischen den Roboterachsen auf der einen und dem Werkstück auf der anderen Seite ist der Sensor. Seine schwingungsfähige Masse liegt während der Programmierbewegung am Werkstück an und wird auf eine Bahn gezwungen, die durch die Werkstückgestalt und die Führungsfunktionen bestimmt wird. Geht man davon aus, daß die dämpfenden Reiberscheinungen bei der Abtastbewegung ausreichendes Gewicht haben, um Schwingungen der Sensorspitze auf der Werkstückoberfläche weitgehend zu verhindern, und schlägt die Abmessungen des Sensorkopfes der Kontur zu (vgl. Bild 26), kann man sich den Sensor vereinfachend als masselosen, elastischen Biegestab vorstellen, der am Roboter eingespannt ist und auf die Kontur drückt. Bei Kenntnis der aktuellen Roboterposition bleibt die Frage zu beantworten, an welchem Punkt der Sensor die Kontur berührt.

Die Vernachlässigung der Reibung führt dazu, daß sich im zweiachsigen Fall die Meßkraft  $\underline{f}_m$  immer als Senkrechte auf der Kontur ausbildet. Fällt man von der Roboterposition  $\underline{p}_r$  das Lot auf die Kontur, erhält man die Konturposition  $\underline{p}_k$  genau für den Fall, daß Verbiegungsvektor  $\underline{h}$  und Meßkraftvektor  $\underline{f}_m$  die gleiche Richtung haben. Aus den Bildern 27 und 42 geht aber hervor, daß diese Bedingung in der Regel nicht erfüllt ist. Das würde nämlich bedeuten, daß die Elastizitäten beider Sensorkoordinaten exakt identisch wären, was allein aus fertigungstechnischen Gründen praktisch nicht erreichbar ist und im übrigen bereits durch die Sensor konstruktion mit zwei senkrecht zueinander angeordneten Biegestäben verhindert wird (vgl. Bild 24). Bild 42 zeigt an einer beliebigen Kontur die sich für unterschiedliche Elastizitäten einstellenden Richtungen von Meßkraft und Sensorverbiegung. Angenommen ist hier, daß der Roboter vom



**Bild 42:** Bestimmung des Konturberührungspunktes in der Simulation

Nullpunkt aus an die Kontur herangefahren ist, und die Roboterposition, bedingt durch die Sensorverbiegung, im Werkstück liegt. Es wird erkennbar, daß die Steigungswinkel  $\rho$  und  $\theta$  der Kraft- und Verbiegungsgeraden über die Elastizi-

täten miteinander verknüpft sind. Es gilt nämlich:

$$\tan \vartheta = \frac{c_{sx}}{c_{sy}} \tan \rho . \quad (3.36)$$

Die Verbiegungsgerade - das ist diejenige Gerade, die durch die Roboterposition geht und die Kontur im gesuchten Konturpunkt schneidet - kann als Funktion in impliziter Form angegeben werden:

$$F_v(x,y) = y - (\tan \vartheta)x - b_v = 0 . \quad (3.37)$$

Da die Meßkraft im Berührungspunkt senkrecht auf der Tangente an die Kontur steht, lassen sich die Steigungen der Kraftgeraden und der Tangente durch

$$\tan \rho = - \frac{1}{m_t} \quad (3.38)$$

in Verbindung bringen, wobei  $m_t$  die Steigung der Tangente ist. Liegt auch die Konturgleichung  $F_k(x,y)=0$  in impliziter Form vor, so liefert ihre Ableitung die Tangentensteigung:

$$m_t = \frac{dy}{dx} = - \frac{\partial F_k / \partial x}{\partial F_k / \partial y} = - \frac{f_{kx}}{f_{ky}} . \quad (3.39)$$

Dieser Zusammenhang führt unter Berücksichtigung der Gleichung (3.38) und (3.36) auf eine Formulierung für die Verbiegungsgerade (Gl. 3.37), die die Ableitung der Konturgleichung enthält:

$$F_v(x,y) = y - \frac{c_{sx}}{c_{sy}} \frac{f_{ky}}{f_{kx}} x - b_v = 0 . \quad (3.37a)$$

Bestimmt man aus der Tatsache, daß die Verbiegungsgerade durch die Roboterposition verläuft, noch den y-Achsenabschnitt  $b_v$ , dann erhält die Verbiegungsgerade schließlich die Form:

$$F_v(x, y) = y - y_r - \frac{c_{sx}}{c_{sy}} \frac{f_{ky}}{f_{kx}} (x - x_r) = 0 . \quad (3.37b)$$

Mit dem Schnittpunkt dieser Geraden und der Kontur hat man den gesuchten Konturberührungspunkt gefunden. Im Regelfall führt die Schnittpunktsuche auf eine Gleichung höherer Ordnung, aus deren Lösungsmenge die richtige Lösung auszuwählen ist. Für einen Kreis als Konturverlauf, der sich als sehr geeignetes Untersuchungsobjekt erwiesen hat, erhält man eine Gleichung vierter Ordnung in einer Koordinate des gesuchten Berührungspunktes. In Frage kommen nur die reellen Lösungen, die gesuchte ist dann diejenige, die der Roboterposition am nächsten liegt.

Die zur Konturbeschreibung benutzten Funktionen müssen stetig und überall differenzierbar sein. Bei Gleichungen höherer Ordnung werden die Simulationszeiten allerdings ziemlich lang, weil die Schnittpunktberechnung in der engen Schrittweite der quasikontinuierlichen Simulation durchgeführt werden muß. Dieser aufwendige Rechenvorgang ist erforderlich, um die Verhältnisse der Tastbewegung nachzubilden, er entfällt beim realen System.

Eine eindeutige Lösung erhält man bei der Verwendung einer Gerade als Kontur. Die Konturgleichung lautet in diesem Fall

$$F_k(x, y) = y + \frac{1}{\tan \rho} x - b_k = 0 . \quad (3.40)$$

Der Schnittpunkt mit der Verbiegungsgerade (Gleichung 3.37b) ist der Konturpunkt

$$x_k = \frac{\tan \rho (b_k c_{sy} - y_r c_{sy} + x_r c_{sx} \tan \rho)}{c_{sy} + c_{sx} \tan^2 \rho}$$

$$y_k = - \frac{1}{\tan \rho} x_k + b_k . \quad (3.41)$$

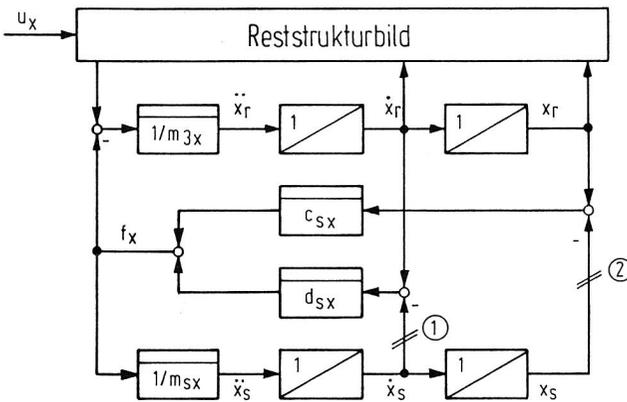
Gleichung (3.41) macht deutlich, daß zusätzlich zu den Verkoppelungen, die durch die Berechnung der zwei Führungsfunktionen (vgl. Abschnitt 3.2) zwischen den verschiedenen Koordinaten entstehen, weitere Verkoppelungen wirksam werden, die auf den Einfluß der Kontur zurückgehen. Offenbar kann die Kontur in ihrem Wirkungseingriff auf den Roboter nicht als fest eingeprägte Störung von außen betrachtet werden (vgl. Bilder 28 und 29), denn die Roboterposition selbst wirkt auf den Berührungspunkt der Sensorspitze an der Kontur zurück (vgl. Gleichung (3.41)). Diese wechselseitigen Verkoppelungen, deren Art vom Verlauf der Kontur bestimmt wird, lassen die Kontur in die Dynamik des Gesamtsystems eingreifen, was den Entwurf gut angepaßter Regler für den praktischen Betrieb erschwert, da grundsätzlich die abzutastenden Werkstückgeometrien beliebig sein sollten.

Der obere Teil von Bild 43 enthält einen Ausschnitt aus dem Strukturbild des Robotermodells für die x-Achse, das aus den Bewegungsgleichungen des vorangegangenen Abschnitts hervorgeht. Gut erkennbar ist der Feder-Masse-Schwinger, der die Sensorspitzenschwingung nachbildet. Dieser Modellausschnitt beschreibt den Fall, daß der Sensor in seiner Schwingfähigkeit nicht durch das Werkstück behindert wird, der Sensor also den Kontakt zum Werkstück verloren hat und demzufolge keine Information über die gesuchten Meßgrößen liefern kann. Die vorangegangenen Überlegungen gehen aber davon aus, daß der Sensor im Kontaktfall als masseloser Biegestab betrachtet werden kann, für dessen Verformung der Roboter die Kraft  $\underline{f}$  aufbringen muß. Sie berechnet sich mit den Elastizitäten und Dämpfungen des Sensors aus Verformungsweg und -geschwindigkeit nach

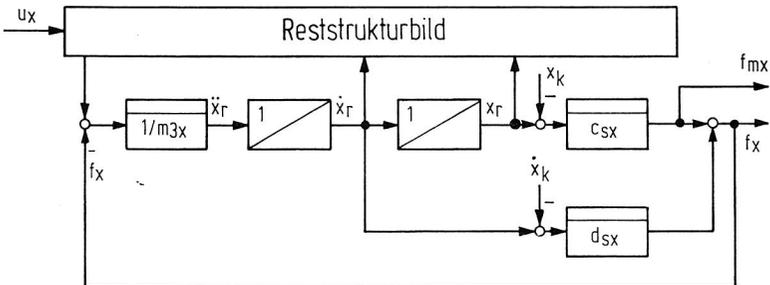
$$\underline{f} = \underline{C}_s (\underline{p}_r - \underline{p}_k) + \underline{D}_s (\dot{\underline{p}}_r - \dot{\underline{p}}_k) . \quad (3.42)$$

Gleichung (3.41) und (3.42) stellen die Verbindung zwischen Roboter- und Umweltmodell her, wobei letztere die erforderlichen Zusammenhänge liefert, um den Strukturbildaus-

schnitt (vgl. Bild 43a) ohne Konturkontakt derart zu modifizieren, daß er den am Werkstück entlangtastenden Sensor beschreibt. Dazu trennt man die mit (1) und (2) gekennzeichneten Verbindungen auf und speist statt der freien Sensorposition und -geschwindigkeit die entsprechenden Konturgrößen ein. Der Sensor selbst kann über die Dehnungsmeßstreifen allerdings nur den Kraftanteil nach dem ersten Term aus Gleichung (3.42) erfassen, der von seinem Verformungsweg bestimmt wird. Der Kraftanteil, der auf den Dämpfungseinfluß



a) ohne Konturkontakt



b) mit Konturkontakt

**Bild 43:** Ausschnitte aus den Strukturbildern der x-Achse mit Sensor, ohne (a) und mit Konturkontakt (b)

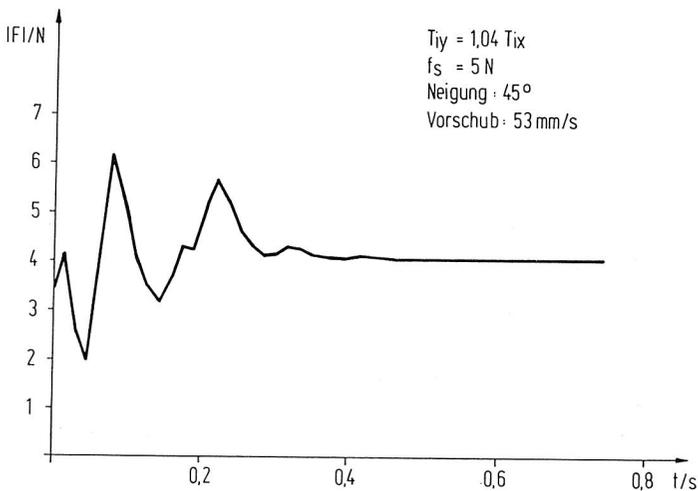
zurückgeht, muß zwar ebenfalls vom Roboter aufgebracht werden, schlägt sich aber nicht im Meßsignal nieder, sondern wird in den Biegestäben in Wärme umgesetzt. Er ist bei praktischen Fahrgeschwindigkeiten um Größenordnungen kleiner als der Meßkraftanteil  $\underline{f}_m$ .

### 3.3.4 Ergebnisse der Simulation

Um die wesentlichen Merkmale des realen Programmiervorgangs in der Simulation nachzuempfinden, wird die Anfahrbewegung an das Werkstück heran in die Betrachtung mit einbezogen. Der eigentliche Programmiervorgang kann nämlich erst beginnen, nachdem Sensor und Werkstück Kontakt miteinander gefunden haben. Gleichzeitig löst diese Vorgehensweise die Problematik der Bestimmung sinnvoller Anfangswerte für die Einzelelemente des Gesamtmodells. In der Realität muß die Anfahrbewegung bahngesteuert mit Positionsregelung erfolgen, damit das Werkstück in keinem Fall verfehlt wird, während in der Simulation eine einfache, ungerichtete Bewegung ausreicht, weil sich der Startpunkt der Anfahrt immer an eine geeignete Position in der Umgebung des Werkstücks verlegen läßt, von der aus der Roboter den Sensor sicher in Kontakt zur Kontur bringen kann. Registriert die Steuerungskomponente des Modells die Konturberührung in Form einer Sensorverbiegung, startet sie die Kraftregler, die den gewünschten Sollwert einstellen. Gleichzeitig schaltet die Steuerung den Vorschub auf, um die Fahrt entlang der Kontur beginnen zu lassen. Um dabei die Kraftschwankungen im Rahmen zu halten und ein Ablösen von der Kontur während der Beschleunigungsphase zu verhindern, ist ein 'Sanftanlauf' vorgesehen. Die Steuerung erhöht dabei den Vorschubwert über 10 Abtastperioden rampenförmig von null bis auf den gewünschten Endwert  $u_b$ .

Nach Bild 44, das den Verlauf des Kraftbetrages bei einer simulierten Fahrt entlang einer 45 Grad Gerade darstellt,

klingen die heftigen Schwingvorgänge, die sich beginnend mit der Konturberührung einstellen, auf etwa 25 Abtastperioden nahezu vollständig ab. Die Kontur ist vollständig glatt modelliert, und der Kraftbetrag stellt sich nach der Einschwingphase erwartungsgemäß ebenso glatt ein. Bemerkenswert ist allerdings, daß bei gleichem Fahrweg, den x- und y-Achse über die Bewegung zurücklegen (Gerade unter 45 Grad), der vorgegebene Sollwert von 5N nicht erreicht wird, sondern im Kraftbetrag eine Regeldifferenz erhalten bleibt, die bei einer um 90 Grad gedrehten Gerade ihr Vorzeichen ändert.

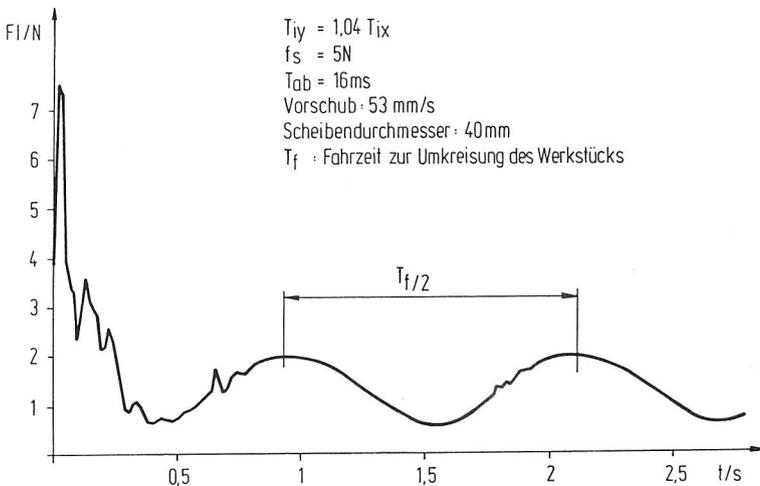


**Bild 44:** Simulierte Konturverfolgung an einer Geraden [78]

Diese Regeldifferenz geht auf die Art der Vorschuberzeugung zurück, die als Störgröße auf die Kraftregelung einwirkt. In diesem Strukturkonzept (vgl. Bild 28) erfolgt der Vorschub nicht geregelt, sondern gesteuert. Nimmt man die beiden Roboterachsen vereinfachend als einfache Integratoren an, so unterscheiden sich ihre Integrationszeitkonstanten ( $T_{ix}$  und  $T_{iy}$ ) um ca. 4%. Trotz dieses geringen Wertes macht die Abweichung voneinander sich doch in der Form störend bemerkbar, daß die Vorschubbewegung abweichend von ihrer Führungsfunktion nicht mehr parallel zur Kontur

gerichtet ist, sondern, je nach Lage der Geraden, von ihr weg oder auf sie zu. Diese fehlerhafte Fahrtrichtung wirkt sich als Störung aus und führt zu einer Regelabweichung in der Normalkraft, weil für alle Messungen am Laborsystem wie am Modell einfache Proportionalregler verwendet wurden.

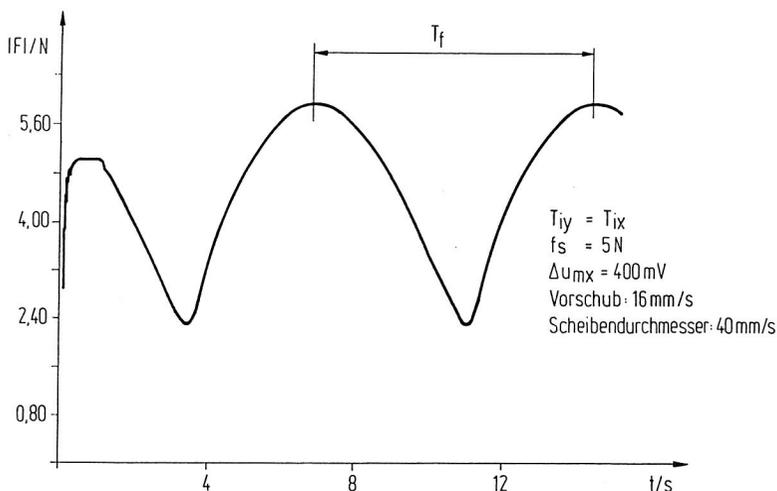
Bei der Fahrtsimulation um eine Scheibe zeigen die aufgezeichneten Führungsfunktionen des Vorschubs einen streng sin/cos-förmigen Verlauf, während die Kraftgrößen unter dem Einfluß der unterschiedlichen Integrationskonstanten einen verzerrten Verlauf der Winkelfunktionen annehmen. Die Periodendauer der Schwingungen entspricht der Zeit  $T_f/2$  zur halben Umkreisung des Werkstücks. Die Verzerrungen in den gemessenen Kraftverläufen wirken sich als Schwankungen im Kraftbetrag aus, deren Frequenz durch die Betragsbildung den doppelten Wert der Fahrkreisfrequenz annimmt (siehe Bild 45). Die kurzen Störspitzen bei 0,7 und 1,8 sec haben keine Bedeutung für das Programmierverfahren. In der x,y-Ebene, die die Fahrebene aufspannt, befindet sich der Roboter in den Störaugenblicken entweder gerade unter oder gerade über



**Bild 45:** Simulierte Konturverfolgung  
an einer Kreisscheibe [78]

dem Mittelpunkt der Kreisscheibe, und die Berechnung des Konturberührungspunktes wirft hier numerische Schwierigkeiten auf, da der Nenner sehr kleine Werte annimmt. In einem schmalen Bereich um diese Problemstellen wird auf eine Näherungsgleichung umgeschaltet, die die Störung im Kraftbetragsverlauf, wie Bild 45 zeigt, zwar nicht völlig beseitigen, aber doch stark reduzieren kann.

Bedingt durch den störenden Einfluß diskreter Steuer- und Regelmechanismen wird bei der Kreisfahrt der vorgegebene Sollwert von 5N nicht einmal näherungsweise erreicht, der Kraftbetrag schwankt stattdessen um einen Mittelwert von ca. 1,5N. Zu jedem Abtastzeitpunkt berechnet die Steuerung neue Werte der Führungsfunktion für den Vorschub, die sie aber durch die künstliche Rechenzeitverlängerung (siehe Bild 29) erst zum nachfolgenden Abtastzeitpunkt wieder ausgibt. Die auf den Steuereingang des Roboters aufgeschaltete Führungsfunktion ist demnach immer veraltet, wobei die dadurch bedingte Regelabweichung umso größer ausfällt, je schneller der Roboter fährt und je stärker die Krümmung der abzu-



**Bild 46:** Simulierte Konturverfolgung an einer Kreisscheibe mit Offsetfehler im Meßkanal [78]

tastenden Kontur ist. Bei der Fahrt um eine Kreisscheibe zeigt die Führungsfunktion des Vorschubs von der Kontur weg und verringert den Kraftbetrag bei zunehmender Geschwindigkeit bis zur Ablösung, während die Kreisfahrt in einem Loch die Wirkung der fehlerhaften Führungsfunktionsberechnung umkehrt, und der Kraftbetrag unter sonst gleichen Bedingungen bis auf ca. 10N in den Amplitudenspitzen ansteigt.

Bild 46 dokumentiert die Wirkung einer Offsetstörung im Meßkanal der Kraft. Für die Simulation wurden, um die Einflüsse anderer Erscheinungen zu unterdrücken, zwei identische Achsen (Modell der x-Achse) verwendet, wobei dem Meßkanal der x-Richtung ein Offset von 400mV aufgeschaltet wurde. Die sich einstellende Schwingung des Kraftbetrages hat im Gegensatz zu den vorher behandelten Störungen (vgl. Bild 45) die Periodendauer  $T_f$  der Fahrkreisfrequenz.

Die Einflüsse auf das Verhalten der in der Simulation behandelten Störungen konnten am Laborsystem qualitativ bestätigt werden. Die Fülle der verschiedenen Störquellen, die einem realen Aufbau anhaften, hätten aber, ohne unterstützende Verwendung der Simulationsergebnisse, eine eindeutige Zuordnung von Störquelle und Störwirkung unmöglich gemacht.

Zusammenfassend bestätigen die Messungen am Modell die Eingangsvermutung, daß die besonders große Bedienerfreundlichkeit, durch die sich das neue Programmierverfahren auszeichnet, sehr wohl ihren Preis hat. Das Verfahren ist beim Abtastvorgang relativ empfindlich gegen eine Reihe von Störungen. Zwar reduziert ihre Wirkung die maximal erreichbare Fahrgeschwindigkeit, der Abtastvorgang bleibt aber grundsätzlich auch unter Fehlereinwirkung beherrschbar. Die auf die Rechentotzeit zurückgehenden Fehler (vgl. Bild 45) sind nur durch eine höhere Abtastfrequenz zu verkleinern, was einen leistungsfähigeren Rechner erforderlich macht, die übrigen markanten Störeinflüsse lassen sich aber durch

relativ einfache Steuerungsmaßnahmen in ihrer Wirkung reduzieren (vgl. z.B. Abschnitte 3.2.3 und 3.2.4). Die Simulation ist dabei ein geeignetes Instrument, um Anhaltswerte für die Signifikanz der einzelnen Störungen in Bezug auf das Abtastverhalten zu erhalten.

### 3.4 Datenreduktion bei sensorgestützter Programmierung

Die dichte Folge von Meßpunkten, die bei den meisten sensorgestützten Programmierverfahren entsteht, ist bei größerer Länge der Bahnen für Archivierungszwecke relativ ungeeignet. Nimmt man an, daß in jedem Abtastpunkt ein Meßwert gespeichert wird, ist mit einem Meßpunktabstand von etwa 0,1mm zu rechnen. Sieht man für die drei Koordinaten eines jeden Meßpunktes einen Datensatz von 12 Byte vor, belegt die eingelernte Bahnkurve pro Meter Bahnlänge einen Speicherplatz von 120 KByte. Roboter müssen meist eine Reihe von Programmen resident verfügbar haben, sodaß eine Voraussetzung für die praktische Verwendbarkeit der Programmiermethode darin besteht, daß eine Beschreibungsform für die Meßpunktfolge gefunden wird, die einerseits geringen Rechenaufwand bei der Gewinnung erfordert und aus der andererseits eine dichte Punktfolge als Führungsgrößenverlauf für den Roboter rekonstruiert werden kann, die genügend gut mit der aufgenommenen Geometrie übereinstimmt.

Die im weiteren behandelten Algorithmen sind zwar grundsätzlich für beliebige Ebenen im Raum geeignet, der Eigenart des bisher betrachteten Programmierverfahrens wird aber insofern Rechnung getragen, als sich die Betrachtung grundsätzlich auf die zwei Koordinaten der  $x,y$ -Ebene beschränkt. Eine geeignete Transformation kann jede Meßkurve, die durch eine anders gelagerte Schnittebene entsteht, in die  $x,y$ -Ebene abbilden.

### 3.4.1 Bestimmungsmethoden für Ersatzkurven

Für einfache Fälle reicht es aus, Meßpunkte in festem Abstand voneinander als Stützpunkte auszuwählen und nur diese im Speicher abzulegen. Bei starken Schwankungen der Konturkrümmung kann das Ergebnis verbessert werden, wenn der Selektionsvorgang von Hand beeinflussbar ist. In keinem Fall dürfen die Stützpunktabstände aber so groß werden, daß beim Wiederabspielen der Bahn die Stufung im Führungsgrößenverlauf sich signifikant in der ausgeführten Roboterbahn abbildet. Es sei denn, man erzeugt, bevor die Stützpunkte den Lageregelkreisen der einzelnen Achsen als Führungsgrößen aufgeschaltet werden, wieder eine dichte Folge dazwischenliegender Punkte, die die Stufung im Führungsverlauf ausreichend glättet. In modernen Bahnsteuerungen übernimmt diese Aufgabe der Interpolator, der dafür als Eingangsgröße eine Bahnbeschreibung braucht, die gewöhnlich aus der Kennzeichnung des Bahntyps und einer Zahl von Stützpunkten besteht (vgl. Abschnitt 4.4.3). Ziel der Datenreduktion muß es also sein, eine Beschreibung für die Meßpunktfolge zu finden, die ein Interpolator verarbeiten kann.

Die Meßpunkte der Kontur mögen über einem festen Abszissen-gitter, den Meßzeitpunkten  $t_1 < t_2 < \dots < t_n$ , als Folge von Funktionswerten in zwei Koordinaten vorliegen. Zunächst wird nun eine Vorschrift gesucht, nach der daraus eine Funktion  $p_e(t)$  rekonstruiert werden kann. Im allgemeinen ist es natürlich nicht möglich, aus endlich vielen vorgegebenen Funktionswerten die Funktion selbst eindeutig zu bestimmen, man kann aber auf verschiedene Arten eine Näherung an die gesuchte Funktion berechnen. Beispielsweise eignen sich die Methoden der Approximation und der Interpolation für diese Aufgabe. Bei der Approximation versucht man, eine Funktion zu finden, die glatt durch die Meßwertfolge verläuft und dabei eine zuvor aufgestellte Ausgleichsbedingung bezüglich der Abstände der Meßwerte von der Approximationsfunktion einhält. Durch die Verwendung einer größeren Zahl von Meßpunkten als

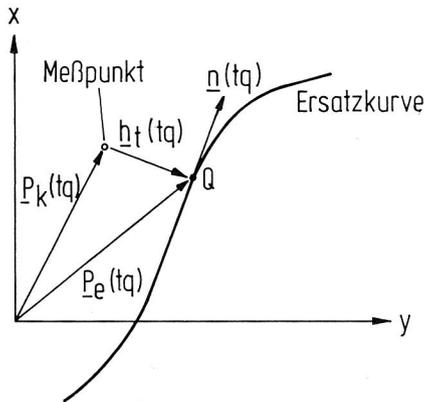
zur Bestimmung der Funktionsparameter erforderlich ist, läßt sich der Einfluß von Meßfehlern auf die Genauigkeit der Näherung verringern.

Im Gegensatz dazu können bei der Interpolation nur soviele Stützpunkte aus den Meßwerten in die Bestimmung der interpolierenden Funktion einfließen, wie die Funktion an Parametern zur Verfügung hat. Die Näherungsfunktion hat dabei die Eigenschaft, an den Stützpunkten exakt mit den ausgewählten Meßpunkten übereinzustimmen, während im dazwischenliegenden Bereich die Abweichung mehr oder weniger groß sein kann. Gibt man den Funktionstyp und die Parameterzahl vor, muß zuerst die zulässige Zahl von Stützpunkten möglichst repräsentativ aus der Gesamtmenge der Meßpunkte ausgewählt werden, wobei die Schwierigkeit darin liegt, daß sich in der Regel nicht vorhersagen läßt, wie geeignet die Auswahl bezüglich des Zusammenspiels von Konturverlauf und Näherungsfunktionstyp ist.

Hat man mit dem abzutastenden Werkstück ein technisches Produkt vorliegen, bestimmen neben ästhetischen auch technische Gesichtspunkte seine Gestalt. Ein Schnitt durch das Werkstück legt auf seiner Oberfläche eine Kontur fest, die in vielen Fällen nicht nur aus einem einzelnen Formelement besteht, sondern sich vielmehr aus einer Reihe von Kurvenzügen wie Geraden, Kreisausschnitten, Kegelschnitten und mathematisch schwer beschreibbaren Linien zusammensetzt. Im allgemeinen wird man nicht erwarten können, daß sich diese Folge von Linienelementen über die gesamte Kontur durch eine einzelne Funktion brauchbar annähern läßt. Besteht die Kontur selbst aus einer Reihe von Linienstücken, so liegt es nahe, mehrere Funktionen zu ihrer Annäherung zu verwenden und sie stückweise aneinanderzusetzen. Damit für die Ersatzkurve ein geschlossener Linienzug entsteht, müssen die einzelnen Funktionen natürlich stetig ineinander übergehen. Gerade diese Forderung können Approximationsmethoden aber in der Regel nicht erfüllen. Die Funktionsparameter werden näm-

lich entsprechend der Ausgleichsbedingung für jedes Funktionsstück getrennt berechnet, ohne Anschlußbedingungen mit einzubeziehen. Bei der Interpolation ist es dagegen immer möglich, die Stetigkeitsbedingung zu erfüllen, indem man einen Stützpunkt als Endpunkt eines Bahnabschnittes und denselben Stützpunkt noch einmal als Startpunkt des nachfolgenden Abschnitts verwendet.

Soll die Interpolation stückweise erfolgen, dann kommt zum Problem der Stützpunktauswahl das der Festlegung der Bahnabschnitte hinzu. Orientiert an den fertigungstechnischen Randbedingungen des Bewegungsprogramms, läßt sich eine sichere Obergrenze für die Länge dieser Bahnabschnitte angeben, die sich gut mit dem Rechner prüfen läßt. Es muß



**Bild 47:** Berechnung des Meßpunktabstandes  $\underline{h}_t$  von der Ersatzkurve

nämlich sichergestellt sein, daß die Abweichungen zwischen der Näherungskurve und den Meßpunkten eine zulässige Toleranzschwelle nicht übersteigen, damit die technologischen Forderungen der Operation noch erfüllbar sind. Da es bei den aufgenommenen Bewegungsbahnen nicht darum geht, die Konturfunktion in ihrem zeitlichen Verlauf möglichst gut nach-

zubilden, sondern ausschließlich die geometrische Übereinstimmung interessiert, ist der kürzeste Abstand  $h_t$  zwischen den Meßpunkten und den als Ersatz berechneten Kurvenstücken ein geeignetes Fehlermaß. Diesen Abstand erhält man, wenn man von einem Meßpunkt  $p_k(t_q)$  (vgl. Gleichung 3.12) aus das Lot auf die Ersatzbahnkurve  $p_e(t_q)$  errichtet und die Strecke  $h_t$  zwischen dem Meßpunkt und dem Lotfußpunkt Q berechnet (Bild 47). Der Abstandsvektor  $\underline{h}_t(t_q)$  muß also senkrecht stehen auf dem Tangentenvektor  $\underline{n}(t_q)$  an die Ersatzkurve im Punkt Q. Anders ausgedrückt muß das Skalarprodukt zwischen beiden Vektoren zu Null werden:

$$\underline{n}(t_q) \cdot \underline{h}_t(t_q) = 0 . \quad (3.43)$$

Als Ersatzfunktionen werden im weiteren ausschließlich Polynome dritten und fünften Grades sowie Geraden und Kreisbögen betrachtet. Die beiden letzten Funktionen sind direkt als mögliche Konturelemente zu betrachten, während Polynome mathematisch gut zu behandeln sind und allgemein günstige Interpolationseigenschaften aufweisen.

Sinnvollerweise wird die Parameterform

$$p(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} = \begin{pmatrix} f_1(t) \\ f_2(t) \end{pmatrix} \quad (3.44)$$

zur mathematischen Darstellung der Kurvenzüge verwendet; sie hat den Vorteil, auch geschlossene Kurven beschreiben zu können. Verwendet man Polynome als Ersatzfunktionen  $f_1(t)$ ,  $f_2(t)$  dann führt die Auswertung der Bedingungsgleichung (3.43) auf ein Polynom in  $t_q$  vom Grade  $2n-1$ , wenn  $n$  der Grad der Ersatzpolynome ist. Das Newtonsche Näherungsverfahren liefert die Lösungen für  $t_q$  recht schnell, wobei nach Plausibilität die richtige auszuwählen ist. Den gesuchten Abstand für den gefundenen Wert von  $t_q$  erhält man aus

$$h_t(t_q) = \left| p_e(t_q) - p_k(t_q) \right| . \quad (3.45)$$

Für einen Kreis läßt sich der Abstandswert  $h_t$  eines Meßpunktes günstiger als mit Gleichung (3.45) durch folgenden Ausdruck angeben:

$$h_t(t_q) = \left| r - \sqrt{(p_{mk} - p_k(t_q))^2} \right| . \quad (3.46)$$

Darin ist  $p_{mk}$  der Mittelpunktvektor des Kreises und  $r$  sein Radius.

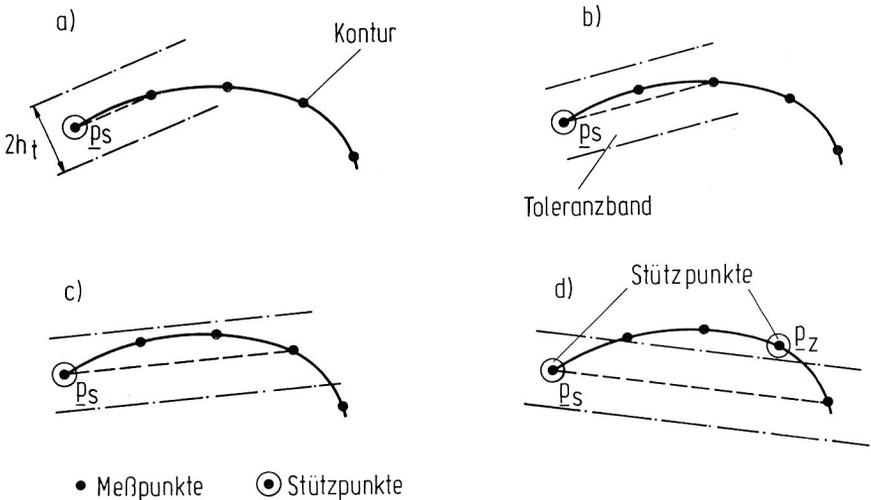
### 3.4.2 Meßpunktselektion durch Überprüfen der Bahntoleranz

GRUHLER stellt in [79] und [80] ein Verfahren zur automatischen Datenreduktion vor, das Bahnabschnitte durch Geradenstücke oder Kreisbögen ersetzt und die Auswahl des Typs der interpolierenden Funktion nach der Maßgabe, möglichst wenig Meßpunkte in den Speicher aufzunehmen, selbsttätig vornimmt. Die Grundidee des Verfahrens liegt darin, die Länge eines Bahnabschnitts so groß zu wählen, daß die Ersatzkurve an der ungünstigsten Stelle einen Abstand  $h_t$  zur Meßkurve hat, der gerade noch unterhalb einer vorgewählten Toleranzschwelle liegt. Zu diesem Zweck nimmt die Steuerung, beginnend mit drei Meßpunkten, solange jeweils einen weiteren Meßpunkt hinzu, bis das Toleranzkriterium verletzt wird. Der letzte davor liegende Meßpunkt wird als Bahnstützpunkt in den Speicher aufgenommen.

Zur Parameterfestlegung der interpolierenden Funktion sind bei Verwendung von Geraden nur zwei Stützpunkte nötig, die allein aus Stetigkeitsgründen an den Übergangsstellen zu den benachbarten Bahnabschnitten liegen müssen. Bei Kreisbögen kommt noch ein dritter Punkt hinzu, der irgendwo zwischen den beiden anderen liegen kann. Die Steuerung berechnet Geraden- und Kreisinterpolation nach dem beschriebenen Muster parallel, und wählt schließlich diejenige Interpolationsart aus, die unter Einhaltung der Toleranzbedingungen den längeren Bahnabschnitt liefert. Fällt die Länge bei

beiden Interpolationsarten gleich aus, wird die Linearinterpolation bevorzugt, weil ein zugehöriger Datensatz weniger Speicherplatz benötigt.

GRUHLER [80] spricht bei freier Meßpunktverteilung im Raum von 'Toleranzschläuchen' mit einem zulässigen Radius  $h_t$  (vgl. Gl. 3.45), die um die Ersatzkurven gelegt werden. Die Interpolation ist dann solange zulässig, wie sämtliche Meßpunkte eines Bahnabschnitts innerhalb des Toleranzschlauchs liegen. Überträgt man diese Vorstellung auf den vorliegenden zweidimensionalen Konturverlauf, kann man von Toleranzbändern sprechen, die die Ersatzkurven umgeben.



**Bild 48:** Überprüfen des maximalen Näherungsfehlers bei Verwendung von Toleranzbändern

Bild 48 zeigt die beschriebene Methode am Beispiel einer linearen Ersatzkurve. Bild 48d liefert den zweiten Stützpunkt  $p_z$ , der in Verbindung mit dem ersten die Ersatzkurve eindeutig bestimmt. Ausgehend vom Zielpunkt  $p_z$  des gerade behandelten Bahnabschnitts beginnt der Vorgang mit Bild 48a

anschließend für den nachfolgenden Bahnabschnitt aufs neue.

### 3.4.3 Kubische Spline-Interpolation

Bei der im zurückliegenden Abschnitt behandelten Methode der stückweisen Interpolation mit Geraden und Kreisbögen kann zwar die Stetigkeit der Ersatzkurve an den Verbindungsstellen, nicht aber die Stetigkeit ihrer Ableitungen gewährleistet werden. Da von 'glatten' Kurven eine höhere Genauigkeit der Näherung erwartet werden darf, bieten Interpolationsverfahren Vorteile, die für die einzelnen Koordinaten mit Polynomen niedrigen Grades arbeiten und an den Übergangsstellen von Bahnabschnitten neben der Stetigkeit der Funktionen auch diejenige in mehreren Ableitungen gewährleisten. Dieser Interpolationstyp wird als 'Spline-Interpolation' bezeichnet.

Eine kubische Spline-Funktion  $S(t)$  über einem gegebenen Abszissengitter  $t_1 < t_2 < \dots < t_n$  besteht aus  $n-1$  jeweils in  $t_i, t_{i+1}$  definierten Polynomen dritten Grades, die in  $t_i$  ( $i=2, \dots, n-1$ ) zweimal stetig differenzierbar aneinandergesetzt sind. Es gilt  $S(t_i) = x_i$ , d.h. an den Stützstellen  $t_i$  stimmen die Funktionswerte der Stützpunkte mit denen der Spline-Funktion  $S(t)$  überein. Setzt man die kubischen Polynome in folgender Form an:

$$f_i(t) = A_i + B_i(t-t_i) + C_i(t-t_i)^2 + D_i(t-t_i)^3, \quad (3.47)$$

beachtet die Koppelbedingungen zwischen den Bahnabschnitten und gibt die ersten Ableitungen  $\dot{x}_1$  und  $\dot{x}_n$  im Anfangs- und Endpunkt vor, dann läßt sich daraus ein lineares Gleichungssystem zur Bestimmung der  $4(n-1)$  Koeffizienten  $A_i, B_i, C_i$  und  $D_i$  aufstellen. Das Gleichungssystem mit einer detaillierten Herleitung findet sich beispielsweise in [81]. Man kann die Koeffizienten der Gleichung (3.47) auch unter Vorgabe der zweiten Ableitungen  $\ddot{x}_1$  und  $\ddot{x}_n$  in den Anfangs- und Endpunkten bestimmen. Werden diese zweiten Ableitungen zu

Null gesetzt, erhält man 'natürliche kubische Splines', die in der Technik große Verwendung gefunden haben und deren Herleitung u.a. in [82] oder [83] zu finden ist. Es wurde allerdings für den Vergleich der Verfahren die Formulierung für die Vorgabe der ersten Ableitung verwendet, da man für Bewegungsbahnen von Robotern gewöhnlich ein Geschwindigkeitsprofil vorgibt. Die Auswirkungen durch die verschiedenartige Festlegung der Randbedingungen betreffen im wesentlichen den ersten und letzten Abschnitt des Spline, der Frage, welche der Formulierungen die günstigere ist, kommt deshalb praktisch nur für den Fall Bedeutung zu, daß eine Kontur durch Aneinandersetzen mehrerer Splines angenähert wird.

Versuche mit Polynomen fünften Grades haben keine Verbesserungen gezeigt, gelegentlich waren die Näherungseigenschaften durch den unruhigen Funktionsverlauf sogar schlechter als bei solchen vom Grade drei.

Die Ersatzkurve soll nicht nur eine gute Übereinstimmung mit der Kontur zeigen, sondern darüber hinaus wenig Speicherplatz benötigen. Deshalb kommt einer sinnvollen Auswahl derjenigen Meßpunkte, die als Stützpunkte dienen sollen, große Bedeutung zu. Da keine allgemeingültigen mathematischen Verfahren bekannt sind, nach denen die Stützpunkte allein aufgrund von Genauigkeitsforderungen errechnet werden können, wurde an verschiedenen Konturen eine große Anzahl von Versuchen durchgeführt, die gezeigt haben, daß es in den meisten Fällen ausreicht, einen konstanten Stützpunktabstand zu wählen, den man nach der gewünschten Genauigkeit variieren kann. Bei einem einheitlichen Meßpunktabstand und einer Festlegung von Stützpunkten auf jedem fünfzigsten Meßpunkt waren die Ergebnisse fast immer recht zufriedenstellend. Erwartungsgemäß haben die Versuche darüber hinaus gezeigt, daß kubische Splines schlechte Anpassungseigenschaften an Ecken bzw. kurzen Abschnitten mit starker Krümmung haben. Zusätzlich eingefügte Stützpunkte in

eine ansonsten gleichmäßige Stützpunktverteilung in Bereichen starker Krümmung und kurz vor bzw. nach Ecken konnten die Ergebnisse spürbar verbessern. Als brauchbare Methode, um solche Bereiche mit dem Rechner ausfindig zu machen, hat sich die Beobachtung der Steigungsänderung der Meßkurve erwiesen [85]. Dabei wurde eine Änderung des Steigungswinkels um mehr als 30 Grad als Ecke interpretiert.

Auch durch Einfügen zusätzlicher Stützpunkte können kubische Splines Ecken nicht exakt abbilden. Ist diese Forderung zwingend, so ist es unumgänglich, den Abschnitt zu unterbrechen und zwei Splines aneinanderzusetzen. In diesem Fall kann man von vornherein mit einem größeren Meßpunktabstand arbeiten; bei Tests war jeder 200. Meßpunkt in der Regel ausreichend.

Eine weitere Verbesserung der Näherungseigenschaften der Ersatzkurve ist erzielbar, wenn das Interpolationsergebnis selbst analysiert wird, und in Bereichen großer Abweichungen bei einer nachfolgenden zweiten Splineberechnung weitere Stützpunkte eingefügt werden. Unter Verwendung der vorliegenden Splinefunktion berechnet man dazu die Bahnfehler nach Gleichung (3.45) über alle Meßpunkte. Da sich die Koeffizienten von Gleichung (3.47) sehr effizient unter Verwendung des Gaußschen Algorithmus bestimmen lassen, kann die Splineberechnung problemlos solange unter Einbeziehung der Verbesserungen wiederholt werden, bis die gewünschte Qualität erreicht ist. Dabei wurden die besten Ergebnisse erzielt, wenn die eingefügten Stützpunkte in der Mitte der fraglichen Bahnabschnitte plazierte wurden. Um Stützpunkte nicht unnötig dicht zu setzen, sollte der Startwert für den Stützpunktabstand nicht zu gering gewählt werden.

### 3.4.4 HERMITESche Spline-Interpolation

Sind neben den Funktionswerten  $x_i$  über einem Abszissengitter  $t_1 < t_2 < \dots < t_n$  zusätzlich Ableitungen verschiedener Grade vorgegeben, spricht man von HERMITEScher Spline-Interpolation. Für den Fall, daß die ersten Ableitungen vollständig vorgegeben sind, und die stetige Differenzierbarkeit der zusammengesetzten Funktion gefordert wird, läßt sich das Problem durch jeweils in  $t_i, t_{i+1}$  definierte Polynome  $f_i$  dritten Grades lösen, die dann nicht mehr gekoppelt sind, wie bei den in Abschnitt 3.4.3 behandelten kubischen Splines. Aus diesem Grunde ist auch kein lineares Gleichungssystem zu lösen. Die Polynome  $f_i$  sind gegeben durch

$$f_i(t) = A_i + B_i t + C_i t^2 + D_i t^3 \quad (t_i \leq t \leq t_{i+1}), \quad (3.48)$$

wobei wieder ein Polynom eine Koordinate als eine Funktion der Zeit repräsentiert. Greift man die Vektorschreibweise von Gleichung (3.44) wieder auf, läßt sich der Funktionsverlauf der Ersatzkurve im  $i$ -ten Bahnabschnitt schreiben als:

$$p_e(t) = \underline{H}_i \underline{t}(t)$$

mit  $\underline{t}(t) = (1, t, t^2, t^3)^T$ . (3.49)

Die Koeffizientenmatrix  $\underline{H}_i$  enthält die Polynomkoeffizienten  $A_i, \dots, D_i$  für die verschiedenen Koordinaten; sie wird bestimmt aus den vier vorgegebenen Randbedingungen für die Position und Geschwindigkeit an beiden Enden des  $i$ -ten Bahnabschnittes. Hierbei nutzt man die Eigenschaft der Interpolation aus, daß die interpolierende Funktion mit der anzunähernden Funktion an endlich vielen Stellen übereinstimmt, und stellt die durch die Meßkurve zu den Zeitpunkten  $t_i$  und  $t_{i+1}$  festgelegten Positionen  $p_e(t)$  und Geschwindigkeiten  $\dot{p}_e(t)$  in einer Matrix zusammen. Man erhält damit:

$$[p_e(t_i), \dot{p}_e(t_i), p_e(t_{i+1}), \dot{p}_e(t_{i+1})] = H_i [\underline{t}(t_i), \dot{\underline{t}}(t_i), \underline{t}(t_{i+1}), \dot{\underline{t}}(t_{i+1})] . \quad (3.50)$$

Auflösen nach der Koeffizientenmatrix  $H_i$  liefert folgende Formulierung:

$$H_i = [p_e(t_i), \dot{p}_e(t_i), p_e(t_{i+1}), \dot{p}_e(t_{i+1})] [\underline{t}(t_i), \dot{\underline{t}}(t_i), \underline{t}(t_{i+1}), \dot{\underline{t}}(t_{i+1})]^{-1} . \quad (3.51)$$

Die Invertierung der 4x4-Matrix in diesem Ausdruck muß für jeden Bahnabschnitt berechnet werden und verursacht dadurch einen nicht unerheblichen Rechenaufwand. BOLLINGER/DUFFIE schlagen in [84] deshalb vor, anstelle der tatsächlichen Abszisse  $t_i \leq t \leq t_{i+1}$  im i-ten Bahnabschnitt die 'normierte' Variable  $s$  mit  $0 \leq s \leq 1$  einzuführen. Die Inverse der Matrix braucht in diesem Fall nur einmal berechnet zu werden, da durch die Normierung die Matrixelemente für alle Bahnabschnitte konstant sind.

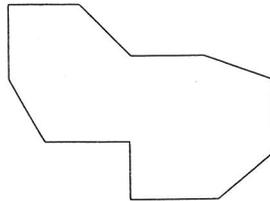
Gegenüber der kubischen Spline-Interpolation hat die HERMITESche die günstige Eigenschaft, daß die einzelnen Polynome der Bahnabschnitte nicht miteinander gekoppelt sind. Durch diese Eigenschaft ist es möglich, in mehreren aufeinanderfolgenden Berechnungsläufen zusätzliche Stützpunkte in einen Bahnabschnitt hineinzusetzen oder andere Stützpunkte aus der Ersatzkurve herauszunehmen, ohne daß diese Maßnahmen Auswirkungen auf die Polynome der übrigen Bahnabschnitte haben. Der Rechenaufwand bleibt gering, weil nur die jeweils betroffenen Polynome neu berechnet werden müssen. Beim Einfügen von Stützpunkten wird die gleiche Methodik verwendet wie in Absatz 3.4.3 beschrieben; es werden allerdings bessere Ergebnisse erzielt, wenn sie anstatt in der Mitte des Bahnabschnitts an die Stelle des größten Fehlers gesetzt werden.

Abgeleitet von einer Untersuchung am Kreis wird in [84] eine Abschätzung für den Fehlerzuwachs in der Mitte des Bahnabschnittes bei Verdoppelung der Bahnabschnittslänge angegeben. Demnach nimmt der Fehler etwa um den Faktor 15 zu, wenn man die Länge des Bahnabschnittes verdoppelt. Eigene Untersuchungen konnten diesen Wert bestätigen. Diese Abschätzung läßt sich mit gutem Erfolg als Prüfkriterium dafür verwenden, ob aus der Ersatzkurve bestimmte Stützpunkte entfernt werden dürfen, ohne daß das vorgegebene Toleranzband verlassen wird. Im übrigen hat sich für die Stützpunktauswahl die gleiche Strategie bewährt, wie für kubische Splines. Das gilt insbesondere für die Behandlung von Ecken.

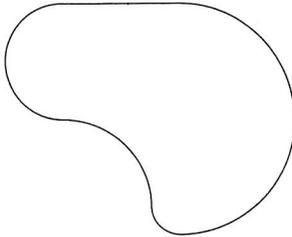
#### 3.4.5 Vergleich der Verfahren

Die drei in den vorangegangenen Abschnitten behandelten Verfahren zur Datenreduktion weisen methodisch und mathematisch so erhebliche Unterschiede auf, daß zu ihrer Beurteilung die nackte Anzahl an Stützpunkten, die bei Einhaltung der Toleranzbedingungen zur Beschreibung der Bahnen erforderlich ist, nicht ausreichend erscheint. An drei charakteristischen Testkonturen wurden deshalb Vergleiche durchgeführt und dabei weitere Kriterien hinzugenommen, die Hinweise für die praktische Verwendbarkeit der Verfahren liefern können. Der 'größte Fehler', das ist die maximale Abweichung zwischen Meßkurve und Ersatzkurve über den gesamten Bahnverlauf (siehe Gleichung (3.45)), und die 'Summe der Fehlerquadrate' dienen gemeinsam als Maß für die erzielte Näherungsqualität. Die 'Rechenzeit' schließlich läßt eine Beurteilung der zu erwartenden Rechnerbelastung zu. Diese Größe ist insofern bedeutsam, als die Reduktionsrechnung im Regelfall auf dem Steuerrechner des Roboters erfolgt, der während dieser Zeit nicht oder nur eingeschränkt für andere Aufgaben zur Verfügung steht. Die ermittelten und in Bild 50 angegebenen Zeiten stellen dabei

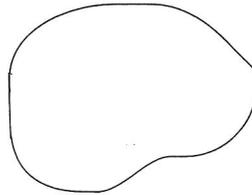
selbstverständlich nur Orientierungswerte dar, sie wurden bei Programmierung der Algorithmen in FORTRAN auf einer VAX 750 ermittelt. Leistungsfähige 32-Bit-Mikroprozessoren mit Arithmetikeinheit dürften aber ähnliche Werte erreichen.



a) Kontur 1



b) Kontur 2



c) Kontur 3

**Bild 49:** Verschiedene Testkonturen bestehend aus unterschiedlichen Geometrieelementen

Die drei Testkonturen (Bild 49) sind mit einem Modellierer in Anlehnung an den realen 'Abtastvorgang' so erzeugt worden, daß die simulierten 'Meßpunkte' in konstanten Abständen aufeinander folgen. Sie beinhalten typische Konturelemente aus der Fertigungstechnik und sind in ihrer Zusammenstellung auf die Qualitäten und Schwächen der Interpolationsverfahren ausgelegt. Kontur 1 (Bild 49a) besteht nur aus Geraden, die unter verschiedenen Winkeln aufeinandertreffen und umfaßt 5603 Punkte im Abstand von 0,1 mm. Für Kontur 2 (Bild 49b), die 5826 Punkte enthält, wurden nur Kreisbögen mit unterschiedlichen Radien verwendet, während Kontur 3 (Bild 49c) aus mehreren Grundformen

wie Kreis, Ellipse, Parabel usw. mit insgesamt 4817 Punkten zusammengesetzt ist.

Folgende Interpolationsverfahren wurden in die Untersuchung einbezogen:

- 1) Linearinterpolation (vgl. Abschnitt 3.4.2)
- 2) Kombinierte Linear- und Zirkularinterpolation (vgl. Abschnitt 3.4.2)
- 3) Interpolation mit kubischen Splines (vgl. Absch. 3.4.3)
  - a) Verwendet wurde ein durchgehender kubischer Spline für die gesamte Bahn. Die Stützpunkte wurden im Abstand von 5 Punkten vor und nach einer Ecke sowie bei Überschreitung eines Wertes von 30 Grad für die Summe der Steigungswinkeländerungen, nie aber weiter als 200 Meßpunkte auseinander, gesetzt.
  - b) Bei gleichmäßigem Abstand der Stützpunkte (jeder 200. Meßpunkt) wurde der Kurvenzug an Ecken aufgetrennt. Für jede dadurch neu entstandene Bahn wurde eine weitere Splinefunktion angesetzt.
- 4) Interpolation mit HERMITESchen Splines (nach Absch.3.4.4)
  - a) Verwendet wurde ein durchgehender HERMITEScher Spline dritten Grades. Die Stützpunkte liegen unmittelbar vor und nach einer Ecke sowie an Stellen, an denen die Summe der Steigungswinkeländerungen der Meßkurve einen Wert von 30 Grad überschreitet. Der maximale Stützpunktabstand beträgt 200 Meßpunkte.
  - b) Es gelten die gleichen Bedingungen wie bei den kubischen Splines (vgl. 3b).

Der maximal zulässige Bahnfehler, also die halbe Breite des Toleranzbandes, beträgt einheitlich 1 mm. Bei Überschreitung der Toleranzschwelle wurden zusätzliche Stützpunkte in die

betreffenden Bahnabschnitte eingefügt, und zwar bei kubischen Splines in der Mitte und bei HERMITESchen Splines an der Stelle des größten Fehlers. Bei HERMITESchen Splines konnte die Stützpunktzahl weiter verringert werden durch Entfernen von Stützpunkten aus Bereichen, in denen der Bahnfehler deutlich unter der Toleranzschwelle lag. Die Ergebnisse des Vergleichs sind in Bild 50 tabellarisch und in Bild 51 übersichtsweise zusammengestellt.

Die geringste Anzahl von Stützpunkten benötigt danach für alle drei Konturen das Verfahren der kombinierten Linear-Zirkularinterpolation. Die reine Linearinterpolation fällt dagegen, besonders an gekrümmten Konturen, spürbar ab, sie liegt dann etwas ungünstiger als die HERMITESche Spline-Interpolation, die mit rund der Hälfte der Stützpunkte auskommt, die die kubische Spline-Interpolation benötigt. Diese Erscheinung gleicht den Umstand aus, daß ein Datensatz

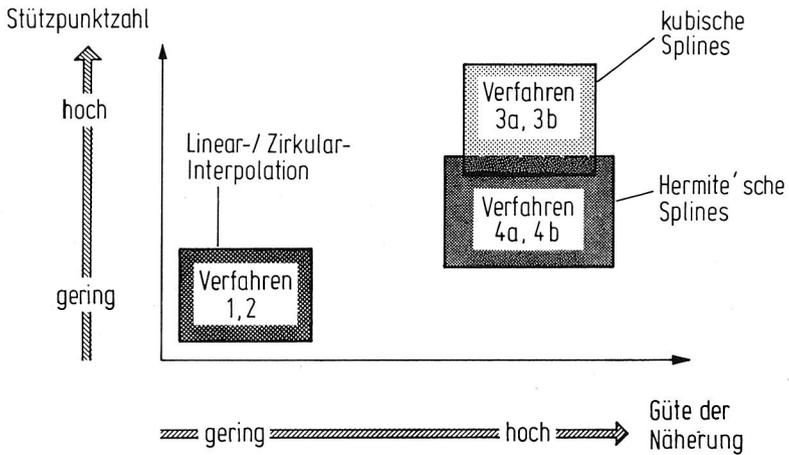
	Verfahren 1	Verfahren 2	Verfahren 3a	Verfahren 3b	Verfahren 4a	Verfahren 4b
	Linear- interpolation	Linear-/ Zirkular- interpolation	kubischer Spline, ohne Unter- brechung	kubischer Spline, mit Unter- brechung	Hermite'sche Splines, ohne Unter- brechung	Hermite'sche Splines, mit Unter- brechung
<b>Kontur 1</b>						
Zahl der Stützpunkte	12	12	54	34	23	15
größter Fehler in mm	1,000	1,000	0,731	0,443	0,868	0,868
Fehlerquadratsumme	1 593,023	1 593,028	360,752	54,196	41,923	41,936
Rechenzeit in s	150,0	150,0	10,0	5,4	11,5	10,2
<b>Kontur 2</b>						
Zahl der Stützpunkte	26	7	32	32	15	15
größter Fehler in mm	0,998	0,998	0,699	0,699	0,487	0,487
Fehlerquadratsumme	2 763,275	1 131,233	98,245	98,245	121,194	121,194
Rechenzeit in s	105,0	410,0	5,1	5,1	12,5	12,5
<b>Kontur 3</b>						
Zahl der Stützpunkte	20	10	28	26	14	14
größter Fehler in mm	0,999	1,000	0,740	0,762	0,574	0,546
Fehlerquadratsumme	2 230,048	1 538,067	114,157	89,119	145,064	120,162
Rechenzeit in s	90,0	160,0	8,1	4,1	11,2	8,5

**Bild 50:** Vergleich von Interpolationsverfahren zur Erzeugung stetiger Ersatzkurven  $p_e(t)$  [85]

für die HERMITESche Spline-Interpolation neben den Funktionswerten  $p_e(t_i)$  auch die ersten Ableitungen  $\dot{p}_e(t_i)$  enthalten muß, während für die kubische Spline-Interpolation die Speicherung der Funktionswerte ausreicht. Der spürbar geringere Speicherplatzbedarf der Verfahren 1 und 2 muß allerdings mit einer Einbuße an Qualität bezahlt werden. Beide Verfahren nutzen in jedem Bahnabschnitt die Toleranzbereiche voll aus, und die hohe Fehlerquadratsumme läßt auf allgemein schlechte Übereinstimmung schließen. Bei den beiden Spline-Verfahren hängt die erzielte Qualität sehr stark von den Eigenschaften der Testkontur und der Anzahl der aneinandergesetzten Splines ab, sie ist in jedem Fall bedeutend größer als bei den beiden ersten Verfahren. Bei der Näherung der gemessenen Kontur mit einer durchgehenden Bahnkurve, die zwangsläufig auch die ursprünglich vorhandenen Ecken abrundet, sind HERMITESche den kubischen Splines deutlich überlegen, insbesondere im Bereich von Ecken ist die Näherungsqualität der kubischen Splines wenig überzeugend.

Zusammenfassend läßt sich feststellen, daß die beiden HERMITESchen Spline-Verfahren die universellsten sind, sie bieten bei vertretbarem Speicherplatzbedarf sehr gute Näherungseigenschaften für alle Typen von Meßkurven (vgl. Bild 51). Kubische Splines dagegen machen vergleichsweise große Schwierigkeiten an Knicken und Ecken. Für ausschließlich glatte Konturen sind sie aber ebensogut geeignet wie die HERMITESchen. Die Verfahren 1 und 2 sind on-line-fähig, d.h. die Datenausriebung kann parallel zur Programmierbewegung erfolgen. Im Gegensatz zu den Spline-Methoden, zu deren Behandlung sämtliche Meßpunkte einer Bahn zunächst in einem Zwischenspeicher abgelegt werden müssen, reichen bei ihnen in diesem Fall bereits relativ kleine Zwischenspeicher aus, da nur die Meßpunkte jeweils eines Bahnabschnittes dem Auswahlmechanismus unterzogen werden. Diese Arbeitsweise treibt allerdings die Rechenzeit soweit in die Höhe, daß für den On-Line-Betrieb die Länge des Bahnab-

schnitts begrenzt werden muß. Damit geht der Vorteil der grundsätzlich geringen Speicherintensität, durch die sich diese Verfahren auszeichnen, zum Teil wieder verloren. Die Frage schließlich, ob es günstiger ist, mit einem durchgehenden Spline oder mit mehreren aneinandergesetzten zu arbeiten, hängt zum einen von der Problemstellung, zum anderen aber auch von der internen Organisation des zur Wiedergabe verwendeten Interpolators ab. Kann der Splineinterpolator nämlich nur jeweils einen Datensatz im voraus



**Bild 51:** Grundeigenschaften von Interpolationsverfahren zur Annäherung gemessener Kurvenverläufe

aufbereiten, ist also nur ein Wechsellpuffer und kein Umlaufpuffer vorgesehen (siehe Abschnitt 4.4.3), läßt sich aufgrund des Zeitbedarfs der Vorbereitungsrechnung für den nachfolgenden Bahnabschnitt eine minimale Satzwechselzeit angeben, die nicht unterschritten werden kann. In diesem Fall ist es günstiger, mit mehreren aneinandergesetzten Splines zu arbeiten und dichte Stützpunktfolgen im Bereich von Ecken zu vermeiden.

#### 4. Montage mit Duplexroboter

##### 4.1 Notwendigkeit und Nutzen der Duplexmontage

In der Montage werden mehrere vorgefertigte Einzelteile zu einem Produkt zusammengesetzt; gewöhnlich durchläuft das Produkt dabei sukzessive eine Folge von Montagestationen. Theoretisch könnte auch ein einzelner Roboter sämtliche Montageoperationen ausführen, die Forderungen nach wirtschaftlicher Gerätenutzung und hoher technischer Zuverlässigkeit zwingen aber meist dazu, mit mehreren Montagegeräten zunächst Vormontagebaugruppen zu bilden und diese später zum Endprodukt zusammensetzen. Bei Verwendung von Robotern als flexible Montagegeräte weist diese Organisation jedem Roboter eine einzelne oder eine kleine Zahl zusammengehöriger Aufgaben zu, wobei der Roboter mit seinen zur Durchführung der Aufgabe erforderlichen Peripheriegeräten eine funktional abgeschlossene Montagezelle bildet.

Diese Grundausstattung von Montagezellen eignet sich allerdings nicht für alle Montageaufgaben; in bestimmten Fällen erscheint es heute sinnvoll, miteinander koordinierte Roboter zu verwenden.

Trotz der hohen Flexibilität, die Roboter grundsätzlich als Montagegeräte aufweisen, ist die Gesamtflexibilität von Montagezellen häufig eher gering. Verantwortlich dafür sind u. a. die heute gebräuchlichen Bereitstellungseinrichtungen, die recht genau den jeweiligen Einzelteilen angepaßt sind und deshalb bei Produktänderungen meist ausgetauscht werden müssen; selbst Produktvarianten benötigen häufig eigene Bereitstellungseinrichtungen. Bei einem vorgegebenen Spektrum von Produktvarianten versucht man sich heute dadurch zu helfen, daß man die alternativen Bereitstellungseinrichtungen nebeneinander vorsieht und zwischen ihnen automatisch umschaltet. Das führt durch die begrenzten Arbeitsräume der Roboter zu Anordnungsproblemen und grenzt dadurch die

mögliche Zellenflexibilität ein.

HUSSAINI und JAKOPAC [86] zeigen als möglichen Ausweg eine Vision auf, die den größeren Teil der Bereitstellungsgeräte durch weitere Roboter ersetzt, die derart miteinander koordiniert werden, daß sie die Fähigkeiten der jeweils anderen nutzen und einander bei der Teilepositionierung und -orientierung, den Hauptaufgaben der Teilebereitstellung, unterstützen. Um die Einschränkungen durch die relativ begrenzte Armreichweite der Roboter aufzuheben, stellen sie sich auf mobilen Plattformen befestigte Roboter vor, deren Zusammenarbeit sich frei an der Arbeitsaufgabe orientiert. Von der neuen Mobilität der Roboter erwarten sie außerdem, daß die verbleibenden peripheren Geräte von mehreren Robotern und damit wirtschaftlicher genutzt werden können.

Die Bewegungsführung in einem solchen Multirobotersystem mit frei veränderlichen und sich teilweise gegenseitig überlappenden Arbeitsräumen hängt von einer Vielzahl von Einflußgrößen ab, was dazu führt, daß die Bewegungsprogramm-erstellung mit den heute üblichen Methoden (vgl. Abschnitt 3.1) kaum noch beherrschbar ist. Einen Lösungsweg für das Problem zeigen Expertensystemansätze auf [87,88], bei denen der Programmierer sein Problem in einer aufgabenorientierten Sprache beschreibt, und die Zellensteuerung selbständig die schwierigen Koordinationsfragen behandelt.

Mit Blick auf den Stand der Technik gehen einige Autoren [89-93] von der Vorstellung aus, daß bereits zwei kooperierende Roboter eine Reihe von Aufgaben bearbeiten können, für die bislang nur manuelle Lösungen oder Spezialgeräte in Frage kommen. Sie beobachten, daß Menschen mit ihren zwei Armen eine Vielzahl sehr diffiziler Handhabungs- und Montagearbeiten beherrschen und dabei gleichzeitig eine sehr hohe Flexibilität erreichen. Sie sehen darin einen gewichtigen Hinweis darauf, daß auch zwei kooperierende Roboterarme zu maßgeblichen Verbesserungen der Leistungsfähigkeit

automatisierter Produktionssysteme beitragen können.

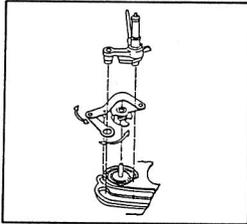
KOHNO et al. [89] berichten von der Automatisierung eines Montagevorgangs an Videorecordern, der bis dahin manueller Ausführung vorbehalten war. Die Aufgabe der beiden Roboter besteht darin, denjenigen Mechanismus zu montieren, der später zu Beginn des Aufnahme- oder Abspielvorgangs das Band aus der Kassette herauszieht und um den mit Magnetköpfen bestückten Führungszylinder herumführt. In der Endmontage setzen die Roboter in einem Arbeitsgang zwei vormontierte Baugruppen in das Chassis ein und fügen dabei den Mechanismus zusammen. Eine dieser beiden Baugruppen besteht aus drei Komponenten, die jeweils frei um ihre Verbindungsachsen drehbar sind (s. Bild 52); sie ist forminstabil und muß durchgängig bis zum Abschluß des Montagevorgangs, zunächst durch die Bereitstellungspalette und später durch den Greifer, in einer vorbestimmten Fixierung gehalten werden. Die Formstabilität der gesamten Baugruppe stellt sich erst im fertig eingebauten Zustand ein.

Eine weitere Aufgabenstellung, für die bisher kaum Lösungen bekannt sind, betrifft die automatische Montage biegeschlaffer Werkstücke, zu denen z.B. Schläuche und Kabel sowie Dichtungen und Federn großer Abmessungen gehören (s. Bild 52). Lediglich in einigen Fällen ist es durch besondere Greiferkonstruktionen gelungen, spezielle Vertreter dieser Werkstückgruppe automatisch zu montieren [94,95]. Werkstückvarianten lassen sich auf diesem Wege aber kaum beherrschen.

Biegeschlaffe Werkstücke haben die Eigenart, im wesentlichen nur Zug-, aber keine Druckkräfte aufnehmen zu können und sind im übrigen sehr forminstabil. Da diese Eigenschaften zu großen Handhabungsproblemen führen, versucht man, auf die Produktkonstruktion derart Einfluß zu nehmen, daß diese problembehaftete Gruppe von Werkstücken entweder entfallen oder soweit wie möglich durch biegesteife Konstruktionselemente ersetzt werden kann. Da diese konstruktiven Ein-

griffe aber häufig anderen Zielsetzungen, z.B. der Baukastenbauweise, zuwiderlaufen und darüber hinaus anders geartete, zum Teil aber nicht weniger große Probleme bei der Gerätemontage verursachen, ist es wünschenswert, diese

forminstabile Baugruppen



forminstabile Werkstücke

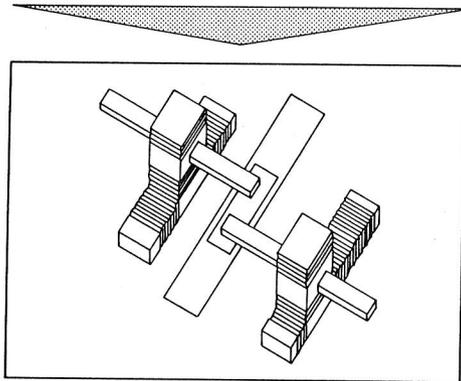
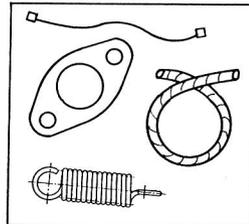


Bild 52: Aufgabenstellungen für die Duplexmontage

Werkstückgruppe für die unvermeidlichen Fälle automatisch und soweit wie möglich mit flexiblen Einrichtungen beherrschen zu können. Ein neuer Lösungsansatz dazu liegt darin, in Anlehnung an die manuelle Montage zwei Roboterarme zu verwenden, deren Greifer die meist langgestreckten Werkstücke an beiden Enden fassen (Bild 52).

#### 4.2 Die Bewegungskordinierung zweier Roboter als Lösungsansatz für spezielle Montageprobleme

Ziel dieses Kapitels ist es, Wege zu erweiterten Robotersteuerungen aufzuzeigen, die durch die Bereitstellung verschiedener Koordinationsmechanismen die Duplexmontage für forminstabile Werkstücke und Baugruppen möglich machen (s. Bild 52). Zu diesem Zweck beleuchtet der nachfolgende Abschnitt zunächst die Problemstellung für diesen Typ von Montageaufgaben genauer, während die nachfolgenden darauf aufbauend mögliche Konzepte für eine aufgabenangepaßte Bewegungskordinierung zweier Roboter aufzeigen.

##### 4.2.1 Problemstellung für die Montage forminstabiler Werkstücke und Baugruppen

Zur Behandlung der oben skizzierten Spezialaufgabe bei der Videorecordermontage haben KOHNO et al. [89] eine Montagezelle mit zwei Robotern aufgebaut, deren Bewegungsverläufe stark abhängen von der Lage der aktuellen Palettenplätze, von denen die vormontierten Baugruppen aufgenommen werden. Unabhängig von diesem vorgelagerten Handhabungsvorgang müssen beide Roboter jedoch die Fügebewegung gemeinsam ausführen. Bedingt durch die wechselnden Fahrtstrecken der beiden Geräte variieren die Ausführungszeiten des Handhabungsvorgangs beträchtlich, was das Zusammenführen der Bewegungen für die Fügephase erschwert. Im einfachsten Fall würde man die längstmöglichen Fahrtzeiten berücksichtigen und ausreichend lange Wartezeiten vorsehen, um den Start für die Fügephase gemeinsam auszuführen. Voneinander abweichende Zeitbasen der beiden Robotersteuerungen würden aber langfristig ein Auseinanderdriften des Startzeitpunktes für beide Geräte zur Folge haben, sofern dies nicht durch die Synchronisierung der beiden Bewegungen verhindert wird. KOHNO et al. [89] haben bei ihrer Realisierung durch eine Erweiterung um kommerzielle Kommunikationsprozessoren einen

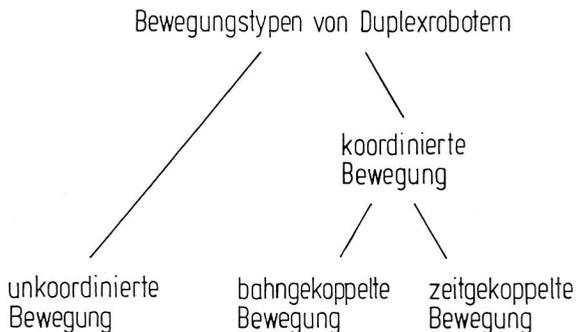
Synchronisationsweg zwischen den beiden getrennten Robotersteuerungen hergestellt und den Beginn der gemeinsamen Fügebewegung als Synchronisationszeitpunkt gewählt. Dadurch gelingt es ihnen, den mittleren Zeitbedarf der gesamten Montageoperation zu senken, da bei jedem Montagegang das Gerät, das den Synchronisationspunkt zuletzt erreicht, den Startzeitpunkt der gemeinsamen Bewegungsphase individuell festlegt.

Soll über die Montage komplizierter Baugruppen hinaus auch das Spektrum der biegeschlaffen Teile beherrschbar sein, müssen zwei Roboter gemeinsame Bewegungen ausführen können, die zusätzlich zur Synchronität ihres Beginns eine geometrische Bedingung erfüllen. Die Greifer beider Roboter haben die langgestreckten, flexiblen Werkstücke nämlich so zu führen, daß sie weder durchhängen noch unzulässig gedehnt werden. In jedem Fall müssen die Bewegungen beider Geräte derart aufeinander abgestimmt sein, daß die Strecke zwischen beiden Greifern konstant bleibt.

#### 4.2.2 Typen von Bewegungen

In der Regel werden koordinierte Bewegungen innerhalb einer Montagezelle nicht durchgängig erforderlich sein; sie werden meist, ähnlich wie beim Videorecorderbeispiel, nur über ausgewählte Bewegungsabschnitte erfolgen. Eine Analyse des Aufgabenspektrums führt auf die Unterscheidung zwischen Phasen unabhängiger Bewegungen, in denen jeder Roboter eigene Teilaufgaben bearbeitet, und Phasen abhängiger Bewegungen, die zur Durchführung einer gemeinsamen Teilaufgabe erforderlich sind. Zugeordnet zu diesen Bewegungsphasen wird ein Bewegungstyp im weiteren 'unkoordinierte Bewegung' genannt, während ein zweiter Typ die Bezeichnung 'koordinierte Bewegung' erhält (siehe Bild 53). Eine Montagezelle mit Duplexrobotern muß insbesondere dann über unkoordinierte Bewegungen verfügen, wenn sie für ein gewis-

ses Spektrum an Montageaufgaben ausgelegt ist, das auch Aufgaben umfaßt, die sinnvoll nur Einzelgeräte bearbeiten können.



**Bild 53:** Für forminstabile Montageobjekte erforderliche Bewegungstypen von Duplexrobotern

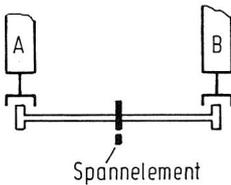
Beim Typ der koordinierten Bewegungen läßt sich sinnvoll noch einmal ein Typ mit kinematischem Bezug der Einzelbewegungen untereinander, der 'bahngekoppelte Bewegung' heißen soll, vom Typ der 'zeitgekoppelten Bewegung' unterscheiden (vgl. Bild 53), für den dieser Bezug nicht existiert. Gemeinsam gilt für die koordinierten Bewegungen, daß beide Einzelbewegungen gleichzeitig beginnen, und ihre Ausführungsdauer gleich ist.

Bild 54 zeigt am Beispiel der Kabelmontage bei einem Videorecorder eine typische Problemstellung für koordinierte Handhabungs- und Montagebewegungen. Das Kabel (siehe Bild 54) verbindet die Übertragungselektronik am Videokopfrad mit den Elektronikbaugruppen im Recordergehäuse und schließt an beiden Enden mit mehrpoligen Steckern ab. Dargestellt sind die vier Schritte des idealisierten Bewegungsablaufs, wobei zum Ausdruck kommt, daß auch für diese Art von Montageaufgabe ein Wechsel zwischen zwei Bewegungstypen erforderlich ist. Bahngekoppelte Bewegungen bestimmen die

Handhabungsphase, während der nachfolgende mehrstufige Montagevorgang (Stecken und Verlegen) mit zeitgekoppelten Bewegungen arbeitet. Dem ersten Schritt vorgelagert ist das Bereitstellen, Indexieren und Spannen des Kabels sowie das Heranfahren beider Roboter an die Aufnahmeposition. Nun greifen zunächst (erster Schritt) beide Roboter je einen Stecker und führen das Kabel gemeinsam an den eigentlichen Montageort. Zu Beginn des zweiten Schrittes schwenkt Gerät A den Stecker um 90 Grad, steckt ihn in die Buchse am Kopfrad (zweiter Schritt) und dient im dritten Schritt als Hilfs-

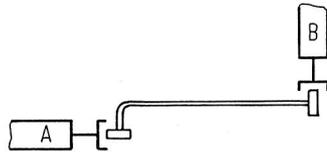
1. Schritt

Greifen der indexierten Stecker und hinbringen zum Montageort



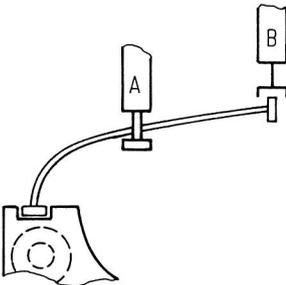
2. Schritt

Schwenken von Greifer A und Stecker



3. Schritt

Spannen und Einlegen mit beiden Armen



4. Schritt

Positionieren und Stecken mit Greifer B

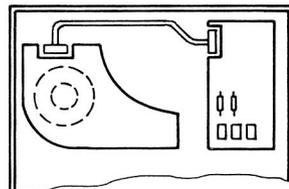


Bild 54: Kabelmontage bei einem Videorecorder

gerät zum Verspannen und Verlegen des Kabels. Der vierte Schritt schließt den Montagevorgang mit dem Stecken des zweiten Steckers durch das Gerät B ab.

Bahngekoppelte Bewegungen müssen, damit sie zur Handhabung biegeschlaffer Werkstücke geeignet sind, derart aus zwei koordinierten Bahnbewegungen zusammengesetzt sein, daß zumindest die Entfernung zwischen den Greifern konstant bleibt. Zur Beschreibung der Zuordnung der Greifer untereinander hat sich der Vektor ihres Abstands als geeignet erwiesen. Während TARN et al. [90] im Zusammenhang mit der Handhabung von Blechen und Stangen einen konstanten Abstandsvektor verwendet, kann für bestimmte Handhabungsvorgänge in der Montage eine kontinuierliche Verdrehung dieses Vektors durchaus sinnvoll sein. Die beiden Roboter müssen nicht einmal in jedem Fall über die nötigen Freiheitsgrade der zusätzlichen Drehbewegung verfügen, da, zumindest in gewissen Grenzen, viele biegeschlaffe Teile eine Verdrehung unbeschadet überstehen und gelegentlich durch die Abstimmung des Greifertyps auf das Werkstück zusätzliche Freiheitsgrade entstehen (vgl. Bild 69, Federbeispiel).

Bestimmend für das im folgenden vorgestellte Konzept der Bewegungskordinierung zweier Roboter ist die Unterscheidung der zusammengehörigen Individualbewegungen in eine unabhängige Bewegung, die vom Anwender frei vorgebar sein soll, und eine abhängige, die aus der ersten abgeleitet werden soll. Während beim bahngekoppelten Bewegungstyp die unabhängige Bewegung in Verbindung mit dem gewünschten Verhalten des Abstandsvektors die abhängige Bewegung vollständig bestimmt, gibt die unabhängige Bewegung beim zeitgekoppelten Typ lediglich ein gemeinsames Zeitraster vor.

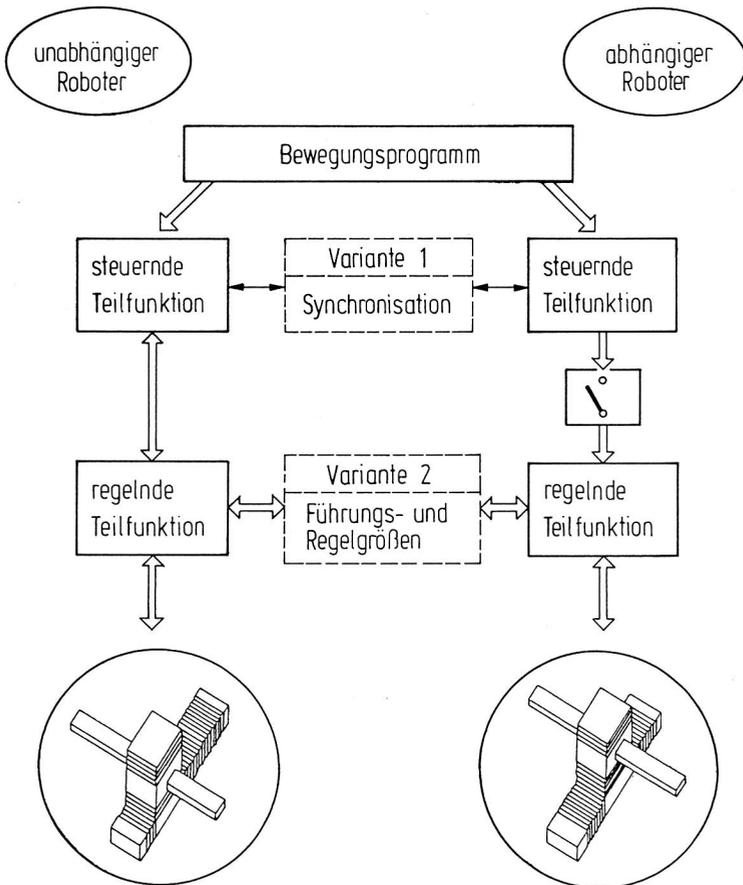
#### 4.2.3 Realisierungskonzepte für koordinierte Bewegungen

Die Problematik eines geeigneten Steuerungskonzeptes für die Montage forminstabiler Objekte mit zwei koordinierten Roboterarmen liegt vor allem in der Unterschiedlichkeit der dazu erforderlichen, in Abschnitt 4.2.2 abgeleiteten, Bewegungstypen. Die Fachliteratur kann zur Lösung nur bedingt Hilfestellung geben, da die meisten Arbeiten auf die Behandlung einer ausgewählten Aufgabe beschränkt sind [89,90,93]. Die deutlichsten Hinweise auf mögliche Steuerungsstrukturen enthalten einige Arbeiten, die sich mit der Problematik bahngekoppelter Bewegungen beschäftigen [90-92]. Forschungsgegenstand der Arbeiten ist der Entwurf eines Regelungssystems, das in der Lage ist, die Forderungen einer vorgegebenen Aufgabenstellung ausreichend gut zu erfüllen. Interessant wird die Themenstellung dadurch, daß die Autoren sich am Handhabungsproblem unhandlicher, großer Werkstücke wie Bleche oder Stangen orientieren, für die aus Gründen der Traglast oder der wirksamen Trägheitsmomente ein einzelner Roboter überfordert ist. In diesem Fall sind die beiden Greifer in ihrer Lage zueinander durch den Werkstückkörper nahezu fixiert, was wegen der geringen Elastizitäten der Roboter und des Objektes während der Handhabungsbewegung zu großen Kräften führen kann, die zwischen beiden Robotern über das Werkstück zur Wirkung kommen.

Nahezu einheitlich hat sich für Roboter industrieller Bauart ebenso wie für Werkzeugmaschinen eine Funktionsgliederung des bewegungserzeugenden Steuerungsteils durchgesetzt, die eine steuernde und eine regelnde Teilfunktion vorsieht. Die steuernde enthält als Kernstück einen Interpolator (vgl. Abschnitt 4.4.3) und erzeugt aus den vorliegenden Bewegungsprogrammen einen sehr fein abgestuften Führungsgrößenverlauf, der die gewünschte Bewegung nach Geometrie und Geschwindigkeitsverlauf repräsentiert. Dieser Führungsgrößenverlauf ist Eingangsgröße für die regelnde Teilfunktion; sie umfaßt im wesentlichen die Lageregelung des

Roboters und zumeist kaskadenartig unterlagerte Geschwindigkeits- und Stromregelungen.

Bild 55 zeigt beide Teilfunktionen angeordnet für zwei Roboter. Im Bild enthalten sind zusätzlich zwei mögliche Steuerungsstrukturen für den Duplexroboterbetrieb. Variante 2 ist der bevorzugte Studiengenegenstand der Autoren, die sich mit Regelungsstrukturen für Duplexroboter beschäftigen [90-92]. Da sich bei bahngekoppelten Bewegungen die abhängige



**Bild 55:** Verschiedene Steuerungsstrukturen für Duplexroboter

Bewegung eindeutig aus der unabhängigen ableiten läßt, ist der Schalter im Informationsfluß des abhängigen Roboters geöffnet und die entsprechende steuernde Teilfunktion überflüssig. Die beiden regelnden Teilfunktionen tauschen dabei je nach Lösungsansatz in verschiedener Form Positions- und Kraftdaten miteinander aus [90-92]. Diese Strukturvariante ist offenbar gut abgestimmt auf die Forderungen bahngekoppelter Bewegungen, macht aber Schwierigkeiten beim Wechsel von diesem Bewegungstyp auf einen anderen oder umgekehrt. In beiden Fällen muß für einen glatten Führungsgrößenübergang gesorgt werden, der auf keinen Fall die Durchführung der beabsichtigten Montageoperationen beeinträchtigen darf.

Die beabsichtigte Flexibilitätssteigerung bei der Montage forminstabiler Objekte macht aber gerade verschiedene Bewegungstypen (vgl. Bild 53) sowie den problemlosen Übergang zwischen ihnen erforderlich, was sich mit der Struktur des oberen Querpfades (Variante 1) in Bild 55 grundsätzlich leichter realisieren läßt. Die Aufgabe der Bewegungskoordination kann dabei auf zwei relativ unabhängige Teilaufgaben zurückgeführt werden. Zum einen muß zur vorliegenden unabhängigen Bewegung eine abgestimmte abhängige Bewegung generiert und zum anderen für die Synchronisierung der daran beteiligten steuernden Rechenprozesse gesorgt werden. Für sämtliche Bewegungstypen sind unabhängige Steuerungen für beide Geräte vorgesehen, wobei ihre Abstimmung untereinander über den Austausch von Synchronisierungsvariablen erfolgt.

Die rein gesteuerte Koordination zwischen beiden Robotern würde allerdings durch die unvermeidlichen Bahnfehler, deren Auswirkungen auf das Werkstück bei dieser Struktur unkorrigiert bleiben, die Handhabung starrer Körper recht problematisch machen. Die biegeschlaffen Werkstücke dagegen, auf deren Handhabung die bahngekoppelten Bewegungen im vorliegenden Fall auszulegen sind, stellen durch ihre

Beschaffenheit aber gerade keine starre Verbindung zwischen beiden Greifern her, sondern eine elastische. Dadurch fällt die Forderung an die Konstanz der Greiferentfernungen voneinander wesentlich geringer aus als bei starren Werkstücken, und Variante 1 erweist sich auch für bahngekoppelte Bewegungen als tauglich.

Nebenbei eignet sich Strukturvariante 1 besser dazu, industrielle Steuerungen der heutigen Steuerungsgeneration zu nutzen, da unter Echtzeitbedingungen lediglich Zustandsvariable zur Bewegungssynchronisation miteinander auszutauschen sind. Strukturvariante 2 benötigt dagegen den Zugriff auf interne, digital repräsentierte Steuerungsgrößen, die heute in der Regel nicht zugänglich sind, und verursacht gegenüber Strukturvariante 1 höheren Rechenaufwand. Bei räumlich getrennten Steuerungen ist außerdem, nach heutigem Stand der Technik, die Übertragung kompletter Führungs- und Regelgrößenverläufe mit erheblichen Kosten verbunden.

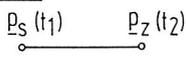
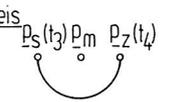
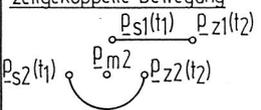
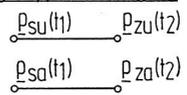
#### 4.3 Ermittlung abgestimmter Bahnkurven

Die weiteren Ableitungen beziehen sich ausschließlich auf die Strukturvariante 1 aus Bild 55. Danach berechnen zwei unabhängige Steuerungen Abschnitte von Bewegungen für die beiden Roboter, die entsprechend des Bewegungstyps aufeinander abgestimmt sind. Synchronisationsmechanismen sorgen dafür, daß bei koordinierten Bewegungen die Ausführung der Abschnitte bei beiden Geräten zeitgleich beginnt, wobei sich die geforderte Art der Koordination einstellt. Da die Ausführungsdauer eines Bewegungsabschnitts bei allen koordinierten Bewegungstypen für beide Geräte stets gleich sein soll, ergibt es sich von selbst, daß nach einmaliger Synchronisation zu Beginn der abhängigen Bewegungsphase beide Individualbewegungen mit jedem Abschnitt erneut synchron beginnen. Die im weiteren behandelten Bahnberechnungen

beschränken sich auf die geometrischen Grundelemente Kreis und Gerade, die heute zur Grundausstattung industrieller Bahnsteuerungen gehören. Da für eine Vielzahl von Montageaufgaben Montagegeräte mit drei Freiheitsgraden ausreichen (vgl. Kapitel 1), bleiben zusätzliche Orientierungsfreiheitsgrade unberücksichtigt. Die Berechnungen lassen sich jedoch ohne Schwierigkeit auf sechs Freiheitsgrade erweitern.

Bild 56 stellt die für die Duplexmontage erforderlichen Bewegungstypen zusammen und enthält neben Hinweisen zu Besonderheiten einen möglichen Parametersatz zur Beschreibung jeweils eines Bewegungsabschnitts. Orientiert am Vorstellungsvermögen eines Programmierers ist es bei industriellen Realisierungen fast ausnahmslos üblich, Abschnitte von Bahnen auf der Basis von Stützpunkten zu definieren. Diese Art der Beschreibung läßt sich speicherplatzgünstig verwalten, meist sehr anschaulich aus der Problemstellung ableiten und bereitet keine grundsätzlichen Schwierigkeiten bei der Überführung in bahnbeschreibende, mathematische Funktionen, die steuerungsintern benötigt werden (vgl. Abschnitt 4.4). Zur Beschreibung einer Geraden reichen zwei Stützpunkte aus, die wegen des notwendigen stetigen Übergangs aufeinanderfolgender Bahnabschnitte jeweils an Anfang und Ende eines Abschnitts gesetzt werden müssen. Ein Kreis benötigt einen weiteren Stützpunkt, der beim Einlernen von Positionen mit der Teach-In Methode (vgl. Abschnitt 3.1) gewöhnlich an beliebiger Stelle auf dem Kreisbogen zwischen Anfangs- und Endpunkt plaziert wird. Für die steuerungsinterne Berechnung ist allerdings als dritter Stützpunkt der Kreismittelpunkt  $p_m$  besser geeignet; er läßt sich aus den drei eingelernten Punkten errechnen. Zur vollständigen Bestimmung der Bewegungen ist es nach industriellem Standard üblich, eine konstante Fahrtgeschwindigkeit  $v_b$  für jeweils einen Abschnitt anzugeben. Davon abweichend sieht Bild 56 mit Angabe von Start- und Zielzeitpunkt ( $t_1, t_2$  bzw.  $t_3, t_4$ ) eines Abschnitts eine äquivalente Darstellung vor, die die

Charakteristika der Bewegungstypen besser zum Ausdruck bringt.

Bewegungstyp		Merkmale	Beschreibungsparameter
unkoordinierte Bewegung	Gerade 	<ul style="list-style-type: none"> <li>• unabhängige Ausführungsdauer</li> </ul>	$\underline{p}_s, \underline{p}_z$ $t_1, t_2$
	Kreis 	<ul style="list-style-type: none"> <li>• freier kinematischer Bezug</li> </ul>	$\underline{p}_s, \underline{p}_m, \underline{p}_z$ $t_3, t_4$
koordinierte Bewegung	zeitgekoppelte Bewegung 	<ul style="list-style-type: none"> <li>• gleiche Ausführungsdauer</li> <li>• freier kinematischer Bezug</li> </ul>	$\underline{p}_{s1}, \underline{p}_{z1}$ $\underline{p}_{s2}, \underline{p}_{m2}, \underline{p}_{z2}$ $t_1, t_2$
	bahngekoppelte Bewegung 	<ul style="list-style-type: none"> <li>• gleiche Ausführungsdauer</li> <li>• fester kinematischer Bezug</li> </ul>	$\underline{p}_{su}, \underline{p}_{zu}$ <sup>1)</sup> $\underline{p}_{sa}, \underline{p}_{za} = f(\underline{p}_{su}, \underline{p}_{sa}, \underline{p}_{zu})$ $t_1, t_2$

Indizes: a abhängig  
u unabhängig  
s Start-  
m Mittelpunkt-  
z Ziel-  
1..4 unspezifisch unterscheidend

1) Parametersatz für eine bahngekoppelte Geradenbewegung

**Bild 56:** Eigenschaften von Bewegungstypen für Duplexroboter

Zeitgekoppelte Bewegungen benötigen für beide Bahnen grundsätzlich die gleichen Parametersätze wie die unkoordinierten Bewegungen, die Start- und Zielzeitpunkte  $t_1$  und  $t_2$  fallen aber für beide Einzelbewegungen zusammen. Bei den bahngekoppelten Bewegungen reduziert sich der erforderliche Parametersatz nochmals, da sich unter Berücksichtigung der gewünschten Veränderung des Abstandsvektors die Stützpunkte der abhängigen Bewegung aus der unabhängigen errechnen lassen. Bild 56 enthält dazu als Beispiel einen Parametersatz für eine bahngekoppelte Geradenbewegung.

#### 4.3.1 Einführung eines globalen Koordinatensystems

Für den Aufbau einer Testumgebung sind zwei Roboter in kartesischer Bauweise ausgewählt worden, die derart gegeneinander verdreht und verschoben aufgestellt worden sind, daß ihre Arbeitsräume sich zu einem Teil überschneiden, beiden Geräten darüber hinaus aber ein unabhängiger Arbeitsbereich erhalten bleibt (Bild 57). Diese Aufstellungsart sichert der Montagezelle die Eignung für ein breites Spektrum von Montageaufgaben, sieht Bereiche vor, in denen

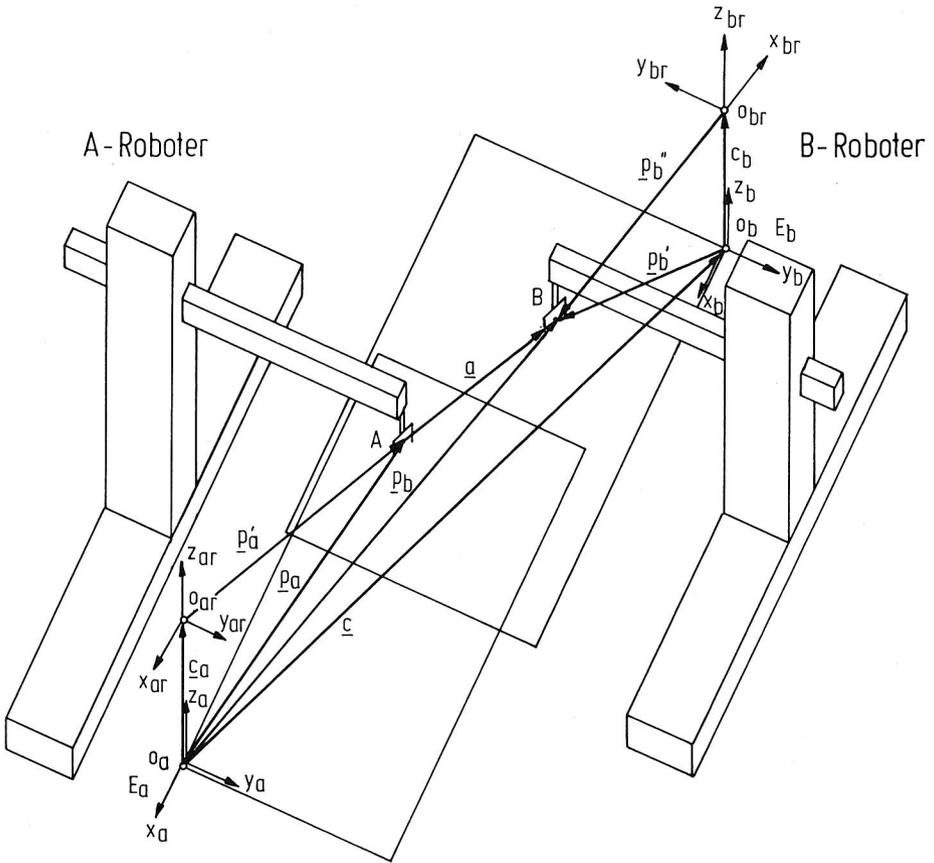


Bild 57: Globales Koordinatensystem für zwei Roboter

unkoordinierte Bewegungen ohne Kollisionsgefahr möglich sind und erhöht gleichzeitig den für Bereitstellungsgeräte verfügbaren Raum. Die Bereitstellungspositionen für biegeschlaffe Werkstücke ordnet man sinnvollerweise im oder nahe am überlappenden Teil des Arbeitsraumes an, diejenigen für normale Werkstücke und Baugruppen können auch im nicht überlappenden Teil liegen.

Um zu Betriebsbeginn, beim Anfahren der Referenzpunkte, die Gefahr einer Kollision zu verringern, sind die beiden Geräte gegeneinander verdreht worden; ihre x- und y-Achsen sind einander entgegengesetzt gerichtet. Die Festlegung eines neuen Koordinatensystems, in dem die Bewegungsbeschreibung grundsätzlich erfolgen soll, befreit dabei von unübersichtlichen Vorstellungen, die mit dem Umgang der referenzpunktbezogenen Koordinatensysteme  $E_{ar}$  und  $E_{br}$  verbunden sind. Die Inkrementalgeber der Achsen liefern die Positionen beider Robotergreifer bezogen auf die gerätespezifischen Koordinatensysteme  $E_{ar}$  und  $E_{br}$  des A- und B-Roboters, die in Bild 57 als  $p'_a$  und  $p''_b$  eingezeichnet sind. Einfache Vektoroperationen bilden diese Positionen im globalen Koordinatensystem  $E_a = (x_a, y_a, z_a)$  ab, das anschaulicherweise im Fußpunkt des A-Roboters angeordnet ist:

$$\underline{p}_a^{<a>} = \overrightarrow{O_a A}^{<a>} = {}^a R_{ar} \underline{p}'_a^{<ar>} + \underline{c}_a^{<a>} \quad (4.1)$$

$$\underline{p}_b^{<a>} = \overrightarrow{O_a B}^{<a>} = {}^a R_{br} \underline{p}''_b^{<br>} + \underline{c}_a^{<a>} \quad (4.2)$$

$$\text{mit } \underline{p}_b^{<b>} = \overrightarrow{O_b B}^{<b>} = {}^b R_{br} \underline{p}''_b^{<br>} + \underline{c}_b^{<b>} \quad (4.3)$$

Darin bezeichnet die hochgestellte Spitze Klammer  $< >$  an einem Vektor die Repräsentation des Vektors innerhalb des darin benannten Koordinatensystems. Der Vektor  $\underline{c}_a^{<a>}$ , der die Verschiebung der beiden Roboter gegeneinander zum Ausdruck bringt, repräsentiert demnach Vektor  $\underline{c}_a = \overrightarrow{O_a O_b}$  im System  $E_a$ . Die Matrix  ${}^b R_{br}$  beschreibt die Drehung des Koordinatensystems  $E_b$  in die Orientierung des Koordinatensystems  $E_{br}$ . Da es

sich dabei um eine 180 Grad Drehung handelt, nimmt sie die einfache Form

$${}^b r_{\mathbb{R}} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ an.} \quad (4.4)$$

Da grundsätzlich die Auswahl des Roboters, der die unabhängige Bewegung ausführt, frei sein soll, werden im weiteren nicht die den Robotern zugeordneten Bewegungen mit den Indizes a und b benannt, sondern ausschließlich die unabhängige und die abhängige Bewegung mit  $\underline{p}_u(t)$  und  $\underline{p}_a(t)$  bezeichnet.

#### 4.3.2 Bahngekoppelte Bewegungen mit konstantem Abstandsvektor

Bahngekoppelte Bewegungen sind zur Handhabung biegeschlaffer Werkstücke gedacht. Für diese Aufgabe muß eine Folge von Individualbewegungen die Greifer beider Roboter zunächst derart an die Bereitstellungseinrichtungen heranbringen, daß das Werkstück an seinen Enden aufgenommen werden kann. Nach dem Greifvorgang müssen beide Geräte alle nachfolgenden Handhabungsbewegungen gemeinsam ausführen, was zuvor eine Synchronisation zwischen ihnen voraussetzt. Mit dem Abschluß des Synchronisationsvorgangs beginnt nun die abhängige Bewegungsphase, die dadurch gekennzeichnet ist, daß zwei kinematisch abgestimmte Bewegungen die Handhabung des Werkstücks bewirken. Dabei dürfen bestimmte Werkstücke, z.B. Flachbandkabel oder langgestreckte Dichtungen, keine Verwindungen erfahren. Legt man zunächst für diejenige Folge von Bewegungsabschnitten, die die unabhängige Bewegung bildet, geeignete Bestimmungsparameter fest (vgl. Bild 56), so lassen sich daraus unter Beachtung der Verdrehempfindlichkeit der Werkstücke Bestimmungsparameter ableiten, die die Abschnittsfolge der abhängigen Bewegung beschreiben. Über

den gesamten Handhabungsvorgang muß derjenige Zustand der geometrischen Zuordnung beider Greifer zueinander erhalten bleiben, den das Werkstück in der Bereitstellungseinrichtung vorgegeben hat. Der Abstandsvektor  $\underline{a}$  zwischen beiden Greifern, der sich berechnen läßt aus den Startvektoren beider Bewegungen

$$\underline{a} = \underline{p}_{sa} - \underline{p}_{su} \quad (4.5)$$

für den ersten Abschnitt der abhängigen Bewegungsphase, repräsentiert diesen Zustand.

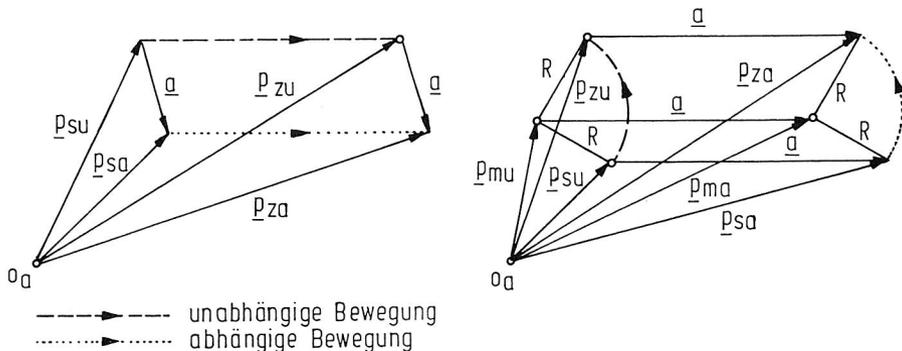
Sieht man für die einander zugeordneten Bewegungen eines jeden Abschnitts stets Bahnen gleichen geometrischen Typs vor (Kreis bzw. Gerade) und berücksichtigt dabei, daß der Startvektor  $\underline{p}_{s,i}$  des i-ten Abschnitts grundsätzlich identisch sein muß mit dem Zielvektor  $\underline{p}_{z,i-1}$  des (i-1)-ten Abschnitts, damit ununterbrochene Roboterbahnen entstehen, so läßt sich für diejenigen Abschnitte mit Geradenbewegungen die abhängige Bahn eindeutig aus dem Zielvektor  $\underline{p}_{zu}$  der unabhängigen Bahn bestimmen. Im i-ten Abschnitt errechnet sich der Zielvektor  $\underline{p}_{za,i}$  dann nach

$$\underline{p}_{za,i} = \underline{p}_{zu,i} + \underline{a} . \quad (4.6)$$

Liegt im i-ten Abschnitt eine Zirkularbewegung vor, muß zusätzlich zum Zielvektor  $\underline{p}_{za,i}$  auf gleiche Weise noch der Mittelpunktvektor

$$\underline{p}_{ma,i} = \underline{p}_{mu,i} + \underline{a} \quad (4.7)$$

bestimmt werden. Bild 58 zeigt je einen Abschnitt der dabei entstehenden Bewegungsbahnen für beide Geräte. Von selbst ergibt sich, daß bei gleicher Ausführungsdauer und gleicher Bahnlänge auch die Bahngeschwindigkeiten  $v_{bu}$  und  $v_{ba}$  eines Abschnitts für beide Bewegungen gleich sind.



**Bild 58:** Bahngekoppelte Geraden- und Kreisbewegung mit konstantem Abstandsvektor (ein Bahnabschnitt)

#### 4.3.3 Bahngekoppelte Drehbewegungen

Existieren durch die Beschaffenheit eines Werkstücks, z.B. einer Feder oder eines einadrigen Drahtes, zusätzliche Bewegungsfreiheitsgrade der Roboter zueinander, oder verfügen die Roboter über zusätzliche Achsen, die die entsprechenden Freiheitsgrade bereitstellen, dann läßt sich eine spezielle bahngekoppelte Kreisbewegung festlegen, die zur Unterscheidung als Drehbewegung bezeichnet wird. Sie setzt sich aus zwei zirkularen Einzelbewegungen um eine gemeinsame Drehachse  $\underline{g}_i$  zusammen. Die abhängige Zirkularbewegung erfolgt um den gleichen Drehwinkel  $\alpha_i$  wie die unabhängige, wobei die Radien sich unterscheiden und die beiden Drehebeneen gegeneinander um einen Abstand  $h$  versetzt sein können. Innerhalb eines solchen Bewegungsabschnitts bleibt der Abstandsvektor  $\underline{a}$  zwischen beiden Greifern nicht konstant, er erfährt vielmehr eine Drehung um die Drehachse  $\underline{g}_i$  und zwar um den gleichen Winkel  $\alpha_i$ , den die beiden Individualbewegungen zurücklegen. Die Länge des Vektors verändert sich jedoch dabei nicht, so daß die Werkstücke nicht gedehnt werden (siehe Bild 59).



während das Vektorprodukt derselben Vektoren die Drehachse

$$\underline{g}_i^{<1>} = \underline{p}'_{su,i}^{<1>} \times \underline{p}'_{zu,i}^{<1>} \quad (4.9)$$

liefert. Die unabhängige Zirkularbewegung überführt den Startvektor  $\underline{p}'_{su,i}^{<1>}$  durch Drehung um die Drehachse in den Zielvektor  $\underline{p}'_{zu,i}^{<1>}$ . Diese Drehung läßt sich beschreiben mit einer zunächst noch unbekanntem Drehmatrix  $\underline{R}(\underline{g}_i, \alpha_i)$ , wobei  $\underline{g}_i$  die Drehachse und  $\alpha_i$  den Drehwinkel im  $i$ -ten Bahnabschnitt darstellen. Diese Drehmatrix läßt sich berechnen aus einer Folge von günstig ausgewählten Koordinatentransformationen. Innerhalb eines neuen Koordinatensystems  $E_{2,i}$ , dessen  $z$ -Achse auf der Drehachse und dessen  $x$ -Achse auf dem Vektor  $\underline{p}'_{su,i}^{<1>}$  liegt, läßt sich die Drehung recht einfach beschreiben durch

$$\underline{p}'_{zu,i}^{<2>} = \underline{R}(\underline{z}_{2,i}, \alpha_i) \underline{p}'_{su,i}^{<2>} \quad (4.10)$$

$$\text{mit } \underline{R}(\underline{z}_{2,i}, \alpha_i) = \begin{pmatrix} \cos \alpha_i & -\sin \alpha_i & 0 \\ \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 1 \end{pmatrix} . \quad (4.11)$$

Beschreibt die Matrix  ${}^1_2\underline{R}_i$  eine Koordinatensystemdrehung, die einen Vektor von einem System  $E_{1,i}$  in ein anderes System  $E_{2,i}$  überführt, läßt sich die gesuchte Drehmatrix durch folgenden Ausdruck angeben:

$$\underline{R}(\underline{g}_i, \alpha_i) = {}^1_2\underline{R}_i^T \underline{R}(\underline{z}_{2,i}, \alpha_i) {}^1_2\underline{R}_i . \quad (4.12)$$

Bei Verwendung der Abkürzungen  $s = \sin \alpha_i$  und  $c = \cos \alpha_i$  nimmt sie nach einigen Umformungen schließlich die folgende Form an:

$$\underline{R}(\underline{q}_i, \alpha_i) = \begin{pmatrix} c+m_x^2(1-c) & -m_z s+m_x m_y(1-c) & m_y s+m_x m_z(1-c) \\ m_z s+m_x m_y(1-c) & c+m_y^2(1-c) & -m_x s+m_y m_z(1-c) \\ -m_y s+m_x m_z(1-c) & m_x s+m_y m_z(1-c) & c+m_z^2(1-c) \end{pmatrix}. \quad (4.13)$$

Darin sind  $m_x = m_{x,i}$ ,  $m_y = m_{y,i}$ ,  $m_z = m_{z,i}$  die Richtungskosini der Drehachse mit den  $x, y, z$ -Koordinatenachsen von System  $E_{1,i}$ . Da auch die abhängige Bewegung eine ebene Zirkularbewegung ist, und die Drehung um die gleiche Drehachse  $\underline{q}_i$  erfolgt, ist die Drehmatrix aus Gleichung (4.13) ebenso für die abhängige Bewegung gültig:

$$\underline{p}'_{za,i}^{<1>} = \underline{R}(\underline{q}_i, \alpha_i) \underline{p}'_{sa,i}^{<1>}. \quad (4.14)$$

Der Vektor des Drehpunktes

$$\underline{p}'_{ma,i}^{<1>} = (\underline{e}_{g,i}^{<1>} \cdot \underline{p}'_{sa,i}^{<1>}) \underline{e}_{g,i}^{<1>}, \quad (4.15)$$

liegt ebenfalls auf der Drehachse  $\underline{q}_i^{<1>}$ , deren Einheitsvektor in Gleichung (4.15) mit  $\underline{e}_{g,i}^{<1>}$  angegeben ist. Transformiert man die Vektoren  $\underline{p}'_{za,i}^{<1>}$  und  $\underline{p}'_{ma,i}^{<1>}$  aus Gleichung (4.14) und (4.15) in das globale Koordinatensystem  $E_a$  zurück, erhält man die gesuchten Bestimmungsparameter der abhängigen Bewegung  $\underline{p}_{za,i}^{<a>}$  und  $\underline{p}_{ma,i}^{<a>}$  im  $i$ -ten Abschnitt. Nimmt man konstante Bahngeschwindigkeiten für beide Individualbewegungen an, dann muß ihr Verhältnis im  $i$ -ten Bahnabschnitt dem Verhältnis ihrer Radien entsprechen. Bei Verwendung der zuvor eingeführten Beschreibungsparameter läßt sich dieser Zusammenhang folgendermaßen ausdrücken:

$$\frac{v_{ba,i}}{v_{bu,i}} = \frac{\|\underline{p}_{sa,i}^{<a>} - \underline{p}_{ma,i}^{<a>}\|}{\|\underline{p}_{su,i}^{<a>} - \underline{p}_{mu,i}^{<a>}\|}. \quad (4.16)$$

#### 4.3.4 Zeitgekoppelte Bewegungen

Die Montage forminstabiler Baugruppen mit einem Duplexsystem erfordert häufig keinen fixierten kinematischen Bezug der Individualbewegungen untereinander, vielmehr bestimmt die Aufgabe die besondere Art der Bewegungsführung. Unabhängig von der Art der Individualbewegungen muß in jedem Fall sichergestellt sein, daß bestimmte Bahnabschnitte zeitgleich begonnen werden, wozu eine Synchronisation zwischen beiden Bewegungen erforderlich ist. Sorgt man dafür, daß beide Bewegungen für jeden Abschnitt die gleiche Ausführungsdauer benötigen, können, nach einmaliger Synchronisation zu Beginn der koordinierten Bewegung, Wartezeiten beim Übergang von einem auf den nächsten Abschnitt vermieden werden. Der zeitgekoppelte Bewegungstyp sorgt auf diese Weise für durchgängig fließende Bewegungen und stellt minimale Taktzeiten sicher. Die Ausführungsdauer beider Bewegungen eines Abschnitts berechnet sich bei konstanter Bahngeschwindigkeit aus dem Quotienten von Bahnlänge und Bahngeschwindigkeit der unabhängigen Bewegung.

#### 4.4 Steuerungskonzepte

Die Zielsetzungen bei der Entwicklung von Robotersteuerungen sind an der Marktentwicklung orientiert, die derzeit noch von der Werkzeughandhabung, vor allem für Punktschweiß-, Bahnschweiß- und Spritzaufgaben, dominiert wird (vgl. Bild 1). Das Marktsegment für Montageroboter ist zur Zeit für aufwendige Sonderentwicklungen noch zu klein, es muß aber für die nahe Zukunft mit großen Wachstumsraten gerechnet werden. Im weiteren wird nun eine Steuerungsstruktur abgeleitet, die den Forderungen der Duplexmontage Rechnung trägt und besonders für geglättete Bewegungsverläufe ausgelegt ist. Zum Schutz der Werkstücke sind gerade für den koordinierten Roboterbetrieb glatte, ruckfreie Bewegungen besonders wichtig.

#### 4.4.1 Glatter Übergang zwischen zwei Bewegungsabschnitten

Bedingt durch die Beschränkung auf zwei Grundtypen von Einzelbewegungen (Gerade und Kreis) ist es in der Regel unumgänglich, einen Bewegungsablauf, der zur Bearbeitung einer Handhabungs- oder Montageaufgabe erforderlich ist, in eine Folge von Bewegungsabschnitten zu zerlegen. Die Verwendung von Stützpunkten als Beschreibungsparameter stellt zwar sicher, daß die einzelnen Bahnabschnitte stetig ineinander übergehen, macht aber Sprünge im Geschwindigkeits- und Beschleunigungsverhalten der gewünschten Bewegungen unvermeidlich. Berücksichtigt man den einheitlichen Grundaufbau von Robotersteuerungen mit ihrer Trennung in eine steuernde und eine regelnde Teilfunktion aus Bild 55, wird sichtbar, daß die gewünschten Bewegungsverläufe repräsentiert sind in den Führungsgrößenverläufen, die der regelnden Teilfunktion aufgeschaltet werden. Der reale Roboter mit seinen immanenten Trägheiten kann diese gewünschten Bewegungen, insbesondere die Geschwindigkeits- und Beschleunigungssprünge, aber nicht ausführen und verläßt in der Folge die vorgegebene Bahn. Für zusätzliche Bahnfehler sorgt die regelnde Teilfunktion, wenn starke Führungsgrößenänderungen Begrenzungserscheinungen in den unterlagerten Regelkreisen hervorrufen oder die Lageregelkreise der Achsen mit P-Reglern ausgestattet sind. Dieser Reglertyp wird üblicherweise in industriellen Steuerungen verwendet und kann Abweichungen zwischen der Bewegungsvorgabe durch die steuernde Teilfunktion und dem realen Bewegungsverhalten der Achse nicht vermeiden. Diese Abweichung, bekannt unter dem Begriff 'Schleppfehler', wirkt sich bei räumlichen Bewegungen, die durch koordiniertes Verfahren verschiedener Geräteachsen erzeugt werden, in Form von Bahnfehlern aus.

Da Bahnfehler somit ohnehin unvermeidlich sind, kann es in bestimmten Fällen sinnvoll sein, definierte Bahnabweichungen bereits in die Vorgaben für die regelnde Teilfunktion hineinzubringen und die dadurch gewonnenen Freiheitsgrade zu

nutzen, um Begrenzungserscheinungen in den unterlagerten Regelkreisen zu vermeiden und insgesamt das Fahrverhalten zu glätten. Gerade für kurze Taktzeiten, die für die Montage kleinvolumiger Produkte angestrebt werden, trägt diese Maßnahme zur Geräte- und Werkstückschonung bei.

In einer begrenzten Übergangszone um die Nahtstelle zweier Bahnabschnitte sollen Polynomfunktionen die exakten Führungsgrößenverläufe für die Lageregler der einzelnen Achsen ersetzen. Läßt man zur Vereinfachung der Rechnung die Zeitzählung jeweils dort mit Null beginnen, wo zwei Bewegungsabschnitte ohne die Glättungspolynome aneinandergesetzt wären, dann erfolgt der Übergang auf die Polynome zum Zeitpunkt  $t = -T_a$  und derjenige auf den nächsten Abschnitt bei  $t = +T_a$  (siehe Bild 60). Damit in jedem Fall unbeabsichtigte

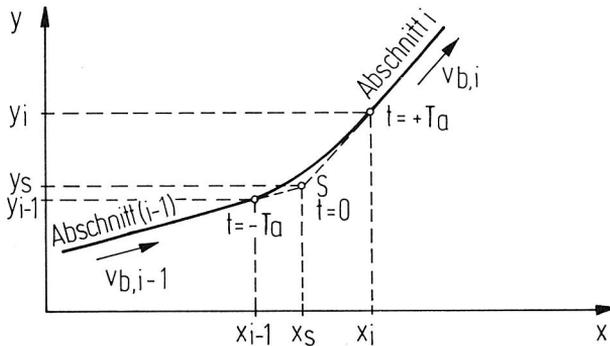


Bild 60: Überschleifen zwischen zwei Bewegungsabschnitten

Bahnverzerrungen vermieden werden können, muß  $T_a$  mindestens so groß gewählt werden, daß die langsamste Achse in dieser Zeit aus dem Stillstand bis auf Maximalgeschwindigkeit beschleunigen kann. Wählt man  $T_a$  größer, wird der Bewegungsverlauf ruhiger, die Übereinstimmung mit der vorgegebenen Bahn aber schlechter. Nennt man  $o$  die verallgemeinerte Position einer Achse und möchte sowohl Geschwindigkeit als auch Beschleunigung der gewünschten Achsbewegung glatt

vorgeben, dann läßt der Überschleifvorgang sich durch ein Polynom fünften Grades als Funktion der Zeit beschreiben:

$$o(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4 + b_5 t^5 . \quad (4.17)$$

Die Koeffizienten der Polynome in den einzelnen Achsen werden aus Stetigkeitsforderungen für Position, Geschwindigkeit und Beschleunigung bestimmt. Dazu wird angesetzt:

$$\begin{aligned} o(-T_a) &= o_{i-1} & o(+T_a) &= o_i \\ \dot{o}(-T_a) &= \dot{o}_{i-1} & \dot{o}(+T_a) &= \dot{o}_i \\ \ddot{o}(-T_a) &= \ddot{o}_{i-1} & \ddot{o}(+T_a) &= \ddot{o}_i . \end{aligned} \quad (4.18)$$

Schreibt man Gleichung (4.17) für die sechs allgemeinen Randbedingungen (4.18) an, erhält man ein System von sechs Gleichungen zur Bestimmung der Koeffizienten  $b_0 \dots b_5$ . In Vektorschreibweise erhält man als Lösung den Ausdruck

$$\underline{b} = \underline{M}(T_a) \underline{k} \quad (4.19)$$

$$\text{mit } \underline{k} = (o_{i-1}, o_i, \dot{o}_{i-1}, \dot{o}_i, \ddot{o}_{i-1}, \ddot{o}_i)^T \quad (4.20)$$

$$\text{und } \underline{b} = (b_0, b_1, b_2, b_3, b_4, b_5)^T , \quad (4.21)$$

wobei die Matrix  $\underline{M}$  als einzige Variable die Überschleifzeit  $T_a$  enthält. Damit liefert Gleichung (4.19) die zur Beschreibung des Übergangsvorganges einer Achse von einem auf den nachfolgenden Bewegungsabschnitt nötigen Polynomkoeffizienten  $b_i$ , wenn der spezielle Randbedingungsvektor  $\underline{k}$  (Gleichung (4.20)) für diesen Fall eingesetzt wird.

Liegen die Polynomkoeffizienten  $b_i$  für alle drei Achsen vor, läßt sich daraus im globalen Koordinatensystem  $E_a$  der

zeitvariable Führungsgrößenvektor  $\underline{p}_f(t)$  bilden, der den Bewegungsverlauf eines Robotergreifers im Intervall  $-T_a < t < +T_a$  angibt. Schreibt man Gleichung (4.17) in Vektorschreibweise um

$$\underline{o}(t) = \underline{b}^T \underline{t} \quad (4.17a)$$

$$\text{mit } \underline{t} = (1, t, t^2, t^3, t^4, t^5)^T \quad (4.22)$$

und berücksichtigt Gleichung (4.19), so läßt sich der Führungsgrößenvektor  $\underline{p}_f(t)$  anschreiben als:

$$\underline{p}_f(t) = \begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} = \begin{pmatrix} \underline{k}_x^T \\ \underline{k}_y^T \\ \underline{k}_z^T \end{pmatrix} \underline{M}^T \underline{t} . \quad (4.23)$$

#### 4.4.2 Koordiniertes Überschleifen bei bahngekoppelten Bewegungen

Da bahngekoppelte Bewegungen mit konstantem Abstandsvektor (vgl. Abschnitt 4.3.2) beide Greifer stets identisch, aber konstant gegeneinander versetzt, führen, sind die Randbedingungen der Übergangsvorgänge qualitativ identisch, wodurch der Abstandsvektor  $\underline{a}$  auch in der Übergangsphase zwischen zwei Bewegungsabschnitten stets konstant bleibt. Dies gilt im allgemeinen aber nicht für Drehbewegungen (vgl. Abschnitt 4.3.3), da die Radien beider Kreisbewegungen und in der Folge davon auch die Bahngeschwindigkeiten sehr unterschiedlich sein können. Unangenehmerweise verändert sich in der Übergangsphase aber neben der Orientierung auch die Länge des Abstandsvektors, sodaß der Führungsgrößenvektor der abhängigen Bewegung  $\underline{p}_{fa}(t)$  nicht allein aus den sechs Stetigkeitsbedingungen pro Achse nach Gleichung (4.18) bestimmt werden kann; es ist außerdem eine Zusatzbedingung

zu erfüllen, die sicherstellt, daß die Länge des Abstandsvektors auch in der Übergangsphase konstant bleibt. Diese Bedingung läßt sich mathematisch formulieren:

$$a = \sqrt{a_x^2 + a_y^2 + a_z^2} = \sqrt{(p_{fa}(t) - p_{fu}(t))^2} = \text{konst.}$$

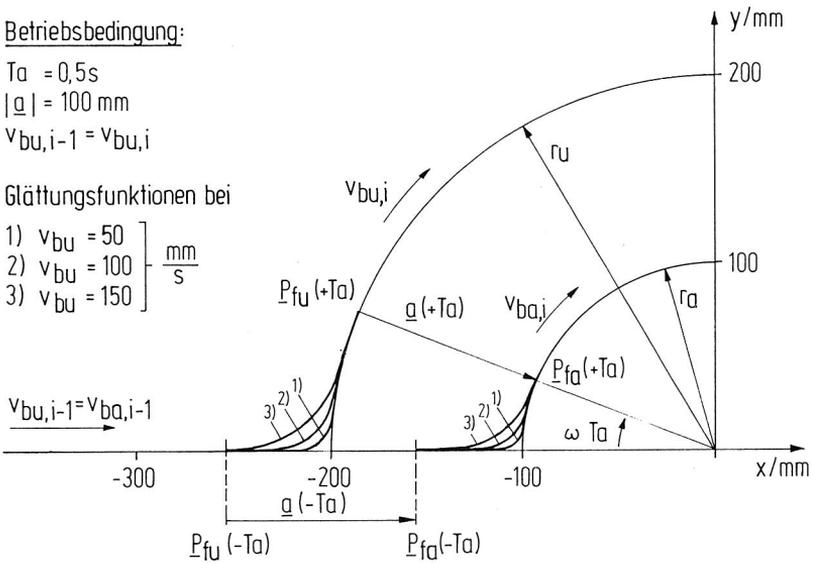
$$\text{für } -T_a \leq t \leq +T_a . \quad (4.24)$$

Unter Einbeziehung dieser Zusatzbedingung können nun allerdings die Polynomkoeffizienten nicht mehr in geschlossener Form bestimmt werden, weshalb ein anderer Ansatz in diesem Fall sich als günstiger erweist. Er geht davon aus, daß die Überschleiffunktion für die abhängige Bewegung aus derjenigen der unabhängigen Bewegung berechnet wird:

$$p_{fa}(t) = p_{fu}(t) + \underline{a}(t) , \quad (4.25)$$

wobei der Abstandsvektor  $\underline{a}(t)$  jetzt als zeitvariante Größe angesetzt wird. Für den Typ der bahngekoppelten Drehbewegung ist  $\underline{a}(t)$  in Gleichung (4.25) lediglich in seiner Länge festgelegt, die Freiheitsgrade seiner Orientierung stehen demnach zur Berücksichtigung von Randbedingungen zur Verfügung. Zu ihrer sinnvollen Festlegung wird zunächst der Spezialfall betrachtet, daß einer Drehbewegung im  $i$ -ten Abschnitt eine bahngekoppelte Geradenbewegung vorangeht. Bild 61 zeigt beide Führungsgrößenverläufe für diesen Fall; beide Übergangspolynome wurden getrennt nach Gleichung (4.23) berechnet. Die Geradenbewegung führt beide Greifer im Abstand  $a$  voneinander auf der gleichen Gerade entlang, wodurch die zwei Kreisbahnen der Drehbewegung in dieselbe Drehebene fallen und denselben Kreismittelpunkt haben. Die Verläufe der Übergangspolynome sind für die Geschwindigkeit der unabhängigen Bewegung  $v_{bu,i-1} = v_{bu,i} = 50\text{mm/s}$ ,  $100\text{mm/s}$  und  $150\text{mm/s}$  im Bild eingezeichnet. Soll nun Gleichung (4.25) die Länge des Abstandsvektors zwischen beiden Greifern auch während des Übergangs konstant halten, kann der Vektor  $\underline{a}$  selbst nicht konstant sein, da andernfalls in Bild 61 die

Position, in der zum Zeitpunkt  $t=+T_a$  der Übergang auf den Kreis der abhängigen Bewegung erfolgt, rechts und oberhalb von der gewünschten Auftreffstelle liegen würde. Die unabhängige Bewegung würde sich bei dieser Art der Sollwertführung kurzfristig nach rückwärts richten und dabei sicher die geforderten Kontinuitätsbedingungen verletzen. Der Grund für den Sprung in der Positionsführung ist darin zu suchen, daß der Vektor  $\underline{a}$  auf dem Übergangspolynom der unabhängigen Bewegung nicht an der Kreisdrehung um den Winkel  $\beta$  teil-



**Bild 61:** Glättung der Führungsgrößenverläufe beim Übergang von einer bahngekoppelten Geradenbewegung auf eine Drehbewegung

nimmt. Folglich läßt sich eine sprungfreie Positionsführung dadurch erzielen, daß man die Drehung geeignet auf den Abstandsvektor wirken läßt. Zur Erfüllung der anderen Randbedingungen (4.18) reicht es allerdings nicht aus, die Winkelgeschwindigkeit  $\dot{\beta}(t)$  konstant zu wählen, es ist vielmehr notwendig, auch für den Winkel  $\beta(t)$  ein Polynom

fünfter Ordnung anzusetzen:

$$\beta(t) = \underline{b}^T \underline{t} . \quad (4.26)$$

Um die Randbedingungen nach (4.18) zu erfüllen, werden äquivalente Randbedingungen für die Drehbewegung aufgestellt:

$$\begin{aligned} \beta(-T_a) &= \beta_{i-1} & \beta(+T_a) &= \beta_i \\ \dot{\beta}(-T_a) &= \dot{\beta}_{i-1} & \dot{\beta}(+T_a) &= \dot{\beta}_i \\ \ddot{\beta}(-T_a) &= \ddot{\beta}_{i-1} & \ddot{\beta}(+T_a) &= \ddot{\beta}_i . \end{aligned} \quad (4.27)$$

Setzt man den Randbedingungsvektor

$$\underline{k} = (\beta_{i-1}, \beta_i, \dot{\beta}_{i-1}, \dot{\beta}_i, \ddot{\beta}_{i-1}, \ddot{\beta}_i)^T \quad (4.28)$$

in Gleichung (4.19) ein, erhält man den Koeffizientenvektor  $\underline{b}$ , der das Polynom  $\beta(t)$  (siehe Gleichung (4.26)) bestimmt. Für den allgemeinen Fall fällt die Drehachse nicht auf die z-Achse wie in Bild 61, und der Abstand zwischen den Dreh Ebenen hat die Länge  $h$ . Die Drehoperation  $\underline{R}(\underline{g}_i, \beta(t))$  (siehe Gleichung (4.13)) beschreibt deshalb eine Drehung um die Drehachse  $\underline{g}_i$  mit dem zeitabhängigen Winkel  $\beta(t)$ . Sie wird angewendet auf den Abstandsvektor  $\underline{a}_0$ , der sich zum Zeitpunkt  $t=0$  ohne Übergangspolynome ergeben würde. Für den Fall, daß der Drehbewegung eine bahngekoppelte Bewegung mit konstantem Abstandsvektor  $\underline{a}$  vorangeht, gilt deshalb:

$$\underline{a}_0(t=0) = \underline{p}_{sa,i} - \underline{p}_{su,i} . \quad (4.29)$$

Damit ergibt sich der Vektor  $\underline{a}(t)$ , mit dessen Hilfe Gleichung (4.25) den Führungsgrößenverlauf für die abhängige Bewegung berechnet, aus:

$$\underline{a}(t) = \underline{R}(\underline{g}_i, \beta(t)) \underline{a}_0(0) . \quad (4.30)$$

Die Drehmatrix  $R(\underline{g}_i, \beta(t))$  erhält man aus Gleichung (4.13) durch Ersetzen von Winkel  $\alpha_i$  durch den zeitvariablen Winkel  $\beta(t)$ ; die Drehung beginnt zum Zeitpunkt  $t=-T_a$ , endet zum Zeitpunkt  $t=+T_a$  und durchläuft den Gesamtwinkel  $\omega T_a$ .

Die Gleichungen (4.25) und (4.30) eignen sich grundsätzlich nicht nur für den behandelten Spezialfall (siehe Bild 61), sondern auch zur Berechnung aller Kombinationen einer Drehbewegung mit bahngekoppelten Bewegungen konstanten Abstandsvektors. Vertauscht man allerdings die Reihenfolge der Bewegungsabschnitte, d.h. geht die Drehbewegung einer Bewegung mit konstantem Abstandsvektor voran, ist Gleichung (4.29) zu ersetzen durch die allgemeinere Formulierung

$$\underline{a}(-T_a) = \underline{p}_{fa}(-T_a) - \underline{p}_{fu}(-T_a) , \quad (4.31)$$

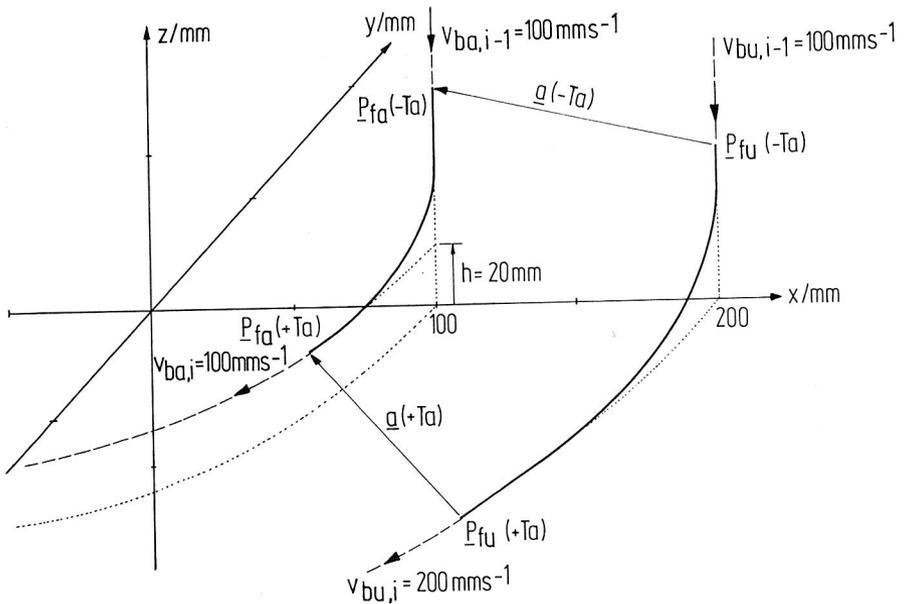
worin  $\underline{p}_{fa}(-T_a)$  und  $\underline{p}_{fu}(-T_a)$  die Führungsgrößen der abhängigen und der unabhängigen Bewegung im (i-1)-ten Bewegungsabschnitt zum Zeitpunkt des Übergangs auf die Glättungsfunktionen sind. Der Vektor  $\underline{a}(-T_a)$  ersetzt dann den Vektor  $\underline{a}_0(0)$  in Gleichung (4.30). Bild 62 zeigt die Ergebnisse der alternativen Methode zur Berechnung der Glättungsfunktionen. Bedingt durch die gegenseitige Greiferzuordnung bei der vorangehenden bahngekoppelten Geradenbewegung, fallen die beiden Drehebene nicht wie in Bild 61 zusammen.

Etwas komplizierter wird dagegen der Übergang zweier Drehbewegungen aufeinander. Dazu müssen nämlich zwei Drehungen mit Drehwinkeln  $\beta(t)$  und  $\gamma(t)$  um die beiden Drehachsen berechnet und stetig zum Zeitpunkt  $t=0$  ineinander überführt werden. Dazu sind sinnvolle Randbedingungen für  $t=0$  anzusetzen, beispielsweise

$$\beta(0) = \omega T_a, \quad \dot{\beta}(0) = \ddot{\beta}(0) = \gamma(0) = \dot{\gamma}(0) = \ddot{\gamma}(0) = 0. \quad (4.32)$$

Bei Berücksichtigung der oben behandelten Sonderfälle liefert die Berechnungsmethode nach Gleichung (4.25) in

jedem Fall einen fehlerfreien Führungsgrößenverlauf für die abhängige Bewegung. Die Berechnung der zeitabhängigen Drehmatrix  $R(g_i, \beta(t))$  sowie ihre Anwendung auf den Abstandsvektor (siehe Gleichung (4.30)) ist allerdings ziemlich rechenintensiv und muß im Zeitraster des Interpolators durchgeführt werden. Dieser Aufwand erscheint nur dann gerechtfertigt, wenn der durch getrennte Berechnung der Glättungsfunktionen verursachte Fehler zu Schäden am Werkstück führen würde. Untersuchungen an Beispielkurven haben ergeben, daß, bei einer Auslegung der Überschleifzeit  $T_a$  nach der erforderlichen Beschleunigungszeit für die langsamste Achse, die Längenfehler des Abstandsvektors für gängige biegeschlaaffe Werkstücke durchweg tolerierbar sind. Aus Sicht der Handhabungstechnik bringt die fehlerfreie Führungsgrößenberechnung für die relativ unempfindlichen,



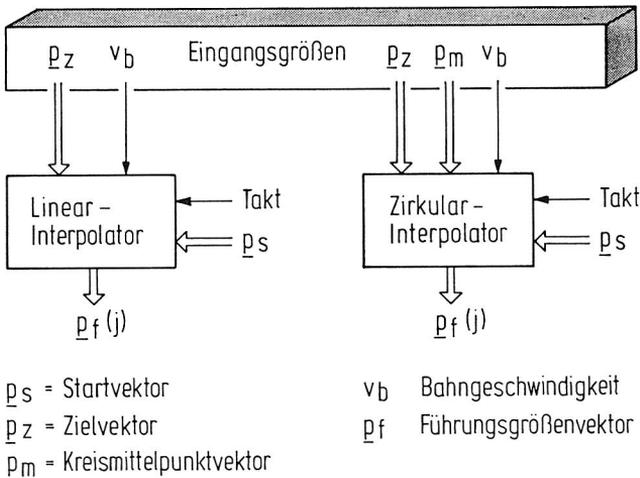
**Bild 62:** Exakte Berechnung der Glättungsfunktionen beim Übergang auf eine Drehbewegung mit verschobenen Drehebene ( $T_a = 0,5 \text{ s}$ )

elastischen Werkstücke in diesem Fall keinen spürbaren Vorteil.

Anders stellt sich die Situation allerdings dar, wenn der Programmierer die Überschleifzeit  $T_a$  in kollisionsunkritischen Phasen als freien Glättungsparameter zur Verfügung haben soll. Für  $T_a \geq 1s$  und höhere Bahngeschwindigkeiten wachsen die Führungsgrößenfehler schnell über 10mm hinaus, so daß die exakte Führungsgrößenberechnung mit ihrem erhöhten Rechenaufwand dann nicht mehr verzichtbar ist.

#### 4.4.3 Grundstruktur eines Interpolators

Um ein mehrachsiges Gerät auf beliebigen Raumkurven entlangzuführen, müssen seine Achsen entsprechend der Vorgaben der Bewegungsbahngeometrie und der Bahngeschwindigkeit koordiniert werden. Innerhalb einer Robotersteuerung übernimmt der Interpolator diese Aufgabe. Aus den Informationen des



**Bild 63:** Eingangs- und Ausgangsgrößen von Linear- und Zirkularinterpolatoren (ein Bahnabschnitt)

Bewegungsprogramms berechnet er eine Stützpunktfolge, die als Führungsgrößenfolge  $p_f(j)$  den Lageregelkreisen aufgeschaltet wird. Die geforderte Bahngüte bestimmt die notwendige Schrittweite zwischen den Stützpunkten auf der Bahn. Bild 63 zeigt die grundsätzliche Arbeitsweise von Linear- und Zirkularinterpolatoren. Ein Linearinterpolator berechnet im Interpolationstakt aus den beiden Bahnvektoren  $p_s$ ,  $p_z$  (vgl. Bild 56) und der Bahngeschwindigkeit  $v_b$  eine Folge von Führungsgrößen  $p_f(j)$ , die zum einen auf der gewünschten Gerade im Raum liegen und zum anderen durch ihren Abstand voneinander die gewünschte Bahngeschwindigkeit einstellen. Die Kreisinterpolation arbeitet zusätzlich mit einem dritten Vektor, gewöhnlich mit dem Kreismittelpunktvektor  $p_m$ . Außer den Bahnstartvektoren  $p_{s,i}$ , die mit den Bahnzielvektoren  $p_{z,i-1}$  des vorangegangenen Bewegungsabschnitts identisch sind, enthält das Bewegungsprogramm alle bewegungsbeschreibenden Angaben inclusive einer Kennung des Bahntyps (linear, zirkular) und führt sie satzweise dem Interpolator als Eingangsgrößen zu (vgl. Bild 63).

Als industrieller Standard können derzeit zweistufige Interpolationskonzepte gelten. Dabei errechnet ein Grobinterpolator in relativ großer Schrittweite Stützstellen auf der gesuchten Bahn, die ein Feininterpolator in kleinerer Schrittweite nochmals unterteilt. Zweistufige Konzepte sind rechenzeitgünstiger als einstufige und bringen zusätzliche Vorteile bei Gelenkgeräten, deren Bewegungsprogramme kartesisch erstellt werden. In diesem Fall ist der Feininterpolator nämlich einer Koordinatentransformation nachgeordnet, die die kartesischen Positionen ins Gelenksystem des Roboters abbildet. Der Verfahrbereich, den die einzelnen Achsen zu einem Grobinterpolationsschritt im kartesischen Raum beitragen, kann je nach Kinematik des Roboters und Lage des Punktes im Raum stark differieren, was große Führungsgrößen sprünge zur Folge haben kann. Der nachgelagerte Feininterpolator unterteilt die durch die Koordinatentransformation verzerrten, achsspezifischen Grobinter-

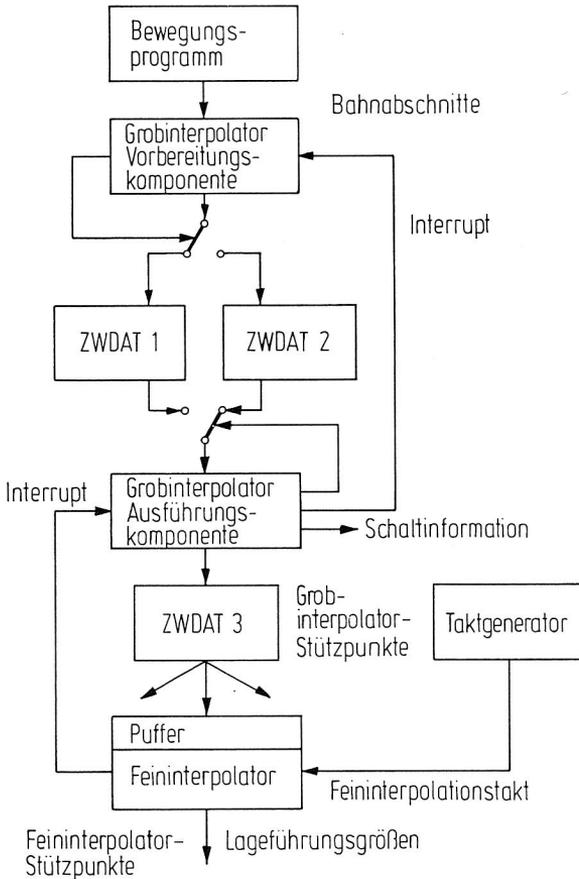
polationsschrittanteile, verfeinert dadurch die Stufung der Führungsgrößen und trägt zur Glättung des Bewegungsverhaltens bei.

Bedingt durch die getaktete Arbeitsweise von Mikrorechnern ist es günstig, bei einer Softwarelösung den Grobinterpolator mit Ausgabe im Zeitraster zu realisieren [96]. Der Interpolator ermittelt dabei in gleichen Zeitabständen Zuwachsstücke für die Stützpunktfolge. Die zeitliche Belastung des Rechners ist stark abhängig vom Bahntyp und dem verwendeten Interpolationsverfahren, kann aber in jedem Fall reduziert werden durch Aufspalten des Rechenprogramms in einen vorbereitenden Teil, der für einen Bahnabschnitt nur einmal durchlaufen werden muß, und einen zweiten einfacheren Teil, der für jeden Grobinterpolationsschritt berechnet werden muß [96].

Es ist weniger günstig, den vorbereitenden, satzbezogenen Rechenprozeß im Zeitraster des schrittbezogenen Rechenprozesses unterzubringen, da auf diese Weise zwar Rechenzeit eingespart werden kann, das Zeitraster sich aber zwangsläufig an der Summe der Rechenzeiten beider Prozesse orientieren muß. Da der vorbereitende Rechenprozeß grundsätzlich einen gesamten Bahnabschnitt betrifft, ist es vorteilhafter, ihn aus dem schrittweisen Grobinterpolationstakt herauszunehmen und ihn satzbezogen vorausseilend den jeweils nächsten Bahnabschnitt berechnen zu lassen. Das gibt größere Freiheiten bei der Wahl des Interpolationszeitrasters, das unter Berücksichtigung des zulässigen Sehnenfehlers bei der Kreisbewegung [97] und der für andere Aufgaben benötigten Rechenleistung festzulegen ist. Die Freiheiten, die der Programmierer bei der Wahl der Bewegungsabschnitte zur Verfügung hat und die Anforderungen verschiedener Montageaufgaben lassen allerdings die zeitliche Dauer eines Satzes derart variieren, daß die Steuerung das Unterschreiten einer minimalen Satzdauer verhindern muß, um sicherzustellen, daß der Rechenprozeß tatsächlich rechtzeitig vor Beginn des

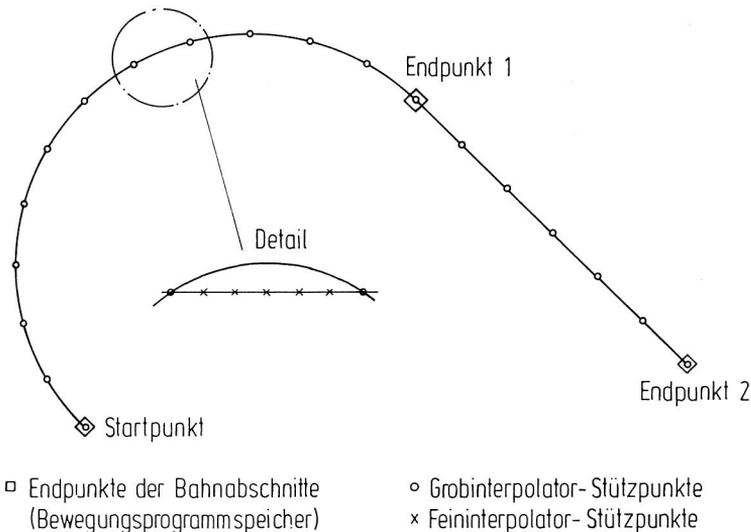
neuen Satzes abgeschlossen ist.

Die satzbezogen voreilende Organisation des vorbereitenden Rechenprozesses begünstigt zudem die Einbeziehung der Glättungsfunktionen, da in diesem Fall im Interpolator jeweils gleichzeitig die Daten der beiden aufeinanderfolgenden Bewegungsabschnitte vorliegen, die zur Bestimmung der Polynomkoeffizienten (vgl. Gleichung (4.19)) erforderlich sind.



**Bild 64:** Struktur eines zweistufigen Interpolators

Obige Überlegungen führen zu einem Interpolatorkonzept, das über die Aufteilung in Grob- und Feininterpolator hinaus den Grobinterpolator selbst in einen satzbezogenen und einen schrittbezogenen Teil zerlegt. Bild 64 demonstriert das Konzept am Beispiel eines Testaufbaus. Die Vorbereitungs-komponente des Grobinterpolators führt alle Berechnungen aus, die für einen Bahnabschnitt nur einmal gemacht werden müssen; dazu zählen z.B. die Berechnungen der Fahrzeit, der Zahl der Grobinterpolationsschritte und der Koeffizienten der Glättungsfunktionen. Zwei Zwischendatensätze ZWDAT1 und ZWDAT2 arbeiten im Wechselpufferbetrieb und übernehmen abwechselnd die Ergebnisse der Vorbereitungsrechnung. Die Ausführungskomponente des Grobinterpolators übernimmt die Daten des aktuellen Bewegungsabschnitts, errechnet daraus die Grobinterpolator-Stützpunkte und legt sie in einem dritten Zwischendatensatz (ZWDAT3) ab. Zusätzlich koordiniert sie die Ausgaben der Schaltinformation mit der Bewegung.



**Bild 65:** Arbeitsweise eines zweistufigen Interpolators bei Linear- und Zirkularinterpolation

Bild 65 zeigt einen prinzipiellen Bahnverlauf mit Unterteilung in zwei Bahnabschnitte. Eingezeichnet sind Anfangs- und Endpunkte der Bahnabschnitte, die Grobinterpolator-Stützpunkte sowie, in einem Detailausschnitt, die Feininterpolator-Stützpunkte, die als Lageführungsgrößen den Lageregelkreisen aufgeschaltet werden.

Auch für den Feininterpolator sieht Bild 64 eine Realisierung mit Zeitrasterausgabe vor. Das Zeitraster des schrittweisen Grobinterpolators ist davon ein ganzzahliges Vielfaches. Die drei Rechenprozesse weisen ausgehend vom Feininterpolator fallende Priorität auf.

Im weiteren werden ausschließlich Interpolationsverfahren behandelt, die mit rekursiver Funktionsberechnung arbeiten. Dabei werden nicht die Gleichungen der Bahn direkt berechnet, sondern bereits vorliegende Stützpunkte auf der Bahn genutzt, um den jeweils nachfolgenden Stützpunkt zu ermitteln. Verfahren der direkten Funktionsberechnung sind gleichermaßen geeignet, benötigen aber demgegenüber einen höheren Rechenaufwand. Ein detaillierter Vergleich verschiedener Interpolationsverfahren findet sich bei BINDER [96].

#### 4.4.4 Linearinterpolation

Diese Interpolationsart führt den Greifer eines Roboters entlang einer Geraden vom Start- zum Zielpunkt. Läßt man die Glättungsfunktionen außer acht, berechnet die Vorbereitungs-komponente (vgl. Bild 64) zunächst den Verfahrensweg, die Fahrzeit und die Anzahl  $N$  der Grobinterpolationsschritte, die entsprechend dem gewählten Zeitraster des Grobinterpolators benötigt werden. Daraus läßt sich schließlich die Größe der pro Grobinterpolationsschritt zu verfahrenenden Inkremente für jede Achse berechnen. Für eine allgemeine Koordinate  $o$  erhält man aus Startwert  $o_s$  und Zielwert  $o_z$

$$o_{ink} = \frac{o_z - o_s}{N} . \quad (4.33)$$

Diese Inkremente addiert die Ausführungskomponente zu den aktuellen Achsführungsgrößen  $o_k$  hinzu

$$o_{k+1} = o_k + o_{ink} . \quad (4.34)$$

Da für die Berechnung der neuen Funktionswerte (Index: k+1) nur die vorausgehenden Werte (Index: k) benötigt werden, liegt ein Verfahren zur Rekursion erster Ordnung vor; die Bahngeschwindigkeit  $v_b$  stellt sich über den gesamten Bahnabschnitt konstant ein.

Soll sich nun die Bahngeschwindigkeit vom Start- zum Zielpunkt konstant verändern, liegt eine konstant beschleunigte Bewegung vor, die allgemein mit

$$p(t) = p_0 + v_s t + \frac{1}{2} \left( \frac{v_z - v_s}{T} \right) t^2 \quad (4.35)$$

beschrieben werden kann. Darin ist  $v_s$  die Geschwindigkeit am Start,  $v_z$  diejenige am Ziel und T die Fahrzeit. Stellt man sich die gesamte Fahrtstrecke für eine Achse wieder als Summe von Weginkrementen vor, dann muß die Länge der Inkremente über die Fahrtstrecke stetig zunehmen. Der gesamte Fahrweg  $o_{ges}$  einer Achse nach N Grobinterpolationsschritten läßt sich danach angeben durch

$$\begin{aligned} o_{ges} &= o_z - o_s \\ &= o_{ink} + (o_{ink} + \Delta o_{ink}) + (o_{ink} + 2\Delta o_{ink}) + \dots \\ &\quad \dots + (o_{ink} + (N-1)\Delta o_{ink}) \\ &= N o_{ink} + \frac{1}{2} N(N-1) \Delta o_{ink} . \end{aligned} \quad (4.36)$$

Löst man die letzte Gleichung nach den Inkrementaländerungen  $\Delta o_{ink}$  auf, so erhält man

$$\Delta o_{ink} = \frac{2(o_z - o_s - N o_{ink})}{N(N-1)} . \quad (4.37)$$

Die Ausführungskomponente des Grobinterpolators addiert nun jeweils zu den bestehenden Achssollwerten  $o_k$  die aktuellen Inkremente  $o_{ink,k}$  hinzu, die selbst nach einer weiteren Rekursion modifiziert werden:

$$o_{k+1} = o_k + o_{ink,k} \quad (4.38)$$

$$o_{ink,k+1} = o_{ink,k} + \Delta o_{ink} . \quad (4.39)$$

Für die danach berechnete beschleunigte Bewegung benötigt der Interpolator abweichend von Bild 63 anstelle einer allgemeinen Bahngeschwindigkeit  $v_b$  als Eingangsgröße die Geschwindigkeiten  $v_s$  am Start und  $v_z$  am Ziel.

#### 4.4.5 Zirkularinterpolation

Die Zirkularinterpolation nach Bild 66 geht von der Parameterdarstellung

$$x(t) = r \cos(\varphi_s + \omega t)$$

$$y(t) = r \sin(\varphi_s + \omega t) \quad (4.40)$$

für den Kreis in der Hauptebene aus.  $N$  Interpolationsschritte der Winkelschrittweite  $\delta$  teilen den Kreisbogen auf. Nach dieser Darstellung müßte jeder Interpolationsschritt die Berechnung der beiden Winkelfunktionen durchführen. Zur Rechnerentlastung hat BINDER in [96] ein Rekursionsverfahren zweiter Ordnung vorgeschlagen, das sich dazu eignet, die zeitintensiven Berechnungen in die Vorberei-

tungskomponente des Grobinterpolators zu verlagern, während für die Ausführungskomponente nur Additionen und Multiplikationen auszuführen bleiben. Dieses Rekursionsverfahren führt auf die Gleichungen

$$x_k = x_{k-2} - 2 \sin\delta Y_{k-1}$$

$$Y_k = Y_{k-2} + 2 \sin\delta x_{k-1} . \quad (4.41)$$

Die Sinusfunktion des Winkels  $\delta$  wird nur einmal in der Vorausberechnung für den gesamten Bahnabschnitt bestimmt. Der Mittelpunktswinkel  $\delta$ , der pro Grobinterpolationsschritt verfahren werden muß, errechnet sich aus

$$\delta = \frac{\varphi_z - \varphi_s}{N} , \quad (4.42)$$

wobei  $\varphi_s$  der Startwinkel und  $\varphi_z$  der Zielwinkel in der Hauptebene ist (siehe Bild 66). Soll eine Kreisbahn im Raum gefahren werden, kann das gleiche Interpolationsverfahren

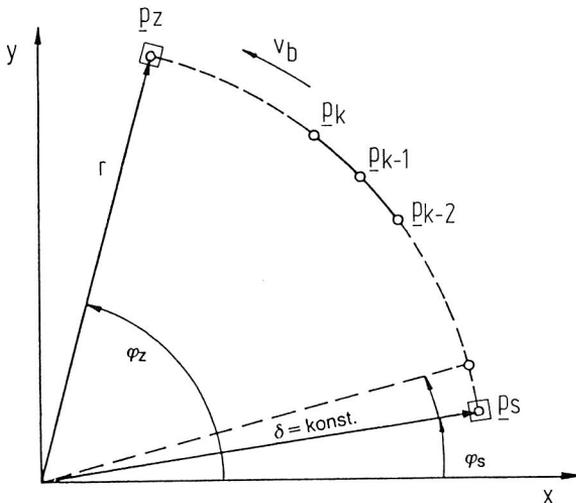


Bild 66: Zirkularinterpolation in einer Hauptebene

wie in der Hauptebene benutzt und der entstehende Kreisabschnitt mit Hilfe einer geeigneten Transformation in die gewünschte Ebene, in frei wählbarer Mittelpunktslage, abgebildet werden. Eine solche Transformation läßt sich sehr kompakt in der Denavit-Hartenberg (DH) Notation  ${}^1_2\mathbb{T}$  ausdrücken. Die  $(4,4)$  Matrix  ${}^2_1\mathbb{T}$  gibt die zugehörige inverse Transformation an. Sie beschreibt die Lage eines orthogonalen Koordinatensystems  $E_2$  relativ zu einem anderen orthogonalen Koordinatensystems  $E_1$ . Die Matrix enthält in den ersten drei Spalten Einheitsvektoren  $\underline{e}_{x2}$ ,  $\underline{e}_{y2}$ ,  $\underline{e}_{z2}$ , die die Richtung der Koordinatenachsen des gedrehten Systems  $E_2$  im Ausgangssystem  $E_1$  beschreiben. In der vierten Spalte ist die Lage des Ursprungs des neuen Koordinatensystems  $E_2$  im Ausgangssystem  $E_1$  als Vektor  $\underline{u}^{<1>}$  angegeben. Die letzte Zeile ist bei allen DH-Matrizen mit den Elementen 0, 0, 0, und 1 besetzt:

$${}^2_1\mathbb{T} = \begin{bmatrix} \underline{e}_{x2}^{<1>} & \underline{e}_{y2}^{<1>} & \underline{e}_{z2}^{<1>} & \underline{u}^{<1>} \\ 0 & 0 & 0 & 1 \end{bmatrix} . \quad (4.43)$$

Zur Abbildung eines in der Hauptebene interpolierten Kreissegments in eine andere Ebene mit verschobener Kreismittelpunktslage wird auf jeden Vektor  $\underline{p}^{<1>}$ , der einen Interpolationspunkt in der Hauptebene bezeichnet, die Transformation  ${}^1_2\mathbb{T}$  angewendet:

$$\underline{p}^{<2>} = {}^1_2\mathbb{T} \underline{p}^{<1>} . \quad (4.44)$$

Die Vorbereitungskomponente des Grobinterpolators muß dazu die Transformation  ${}^2_1\mathbb{T}$  sowie die zugehörige inverse Transformation  ${}^1_2\mathbb{T}$  bestimmen, während Gleichung (4.44) von der Ausführungskomponente zu berechnen bleibt.

#### 4.4.6 Satzweise Synchronisation

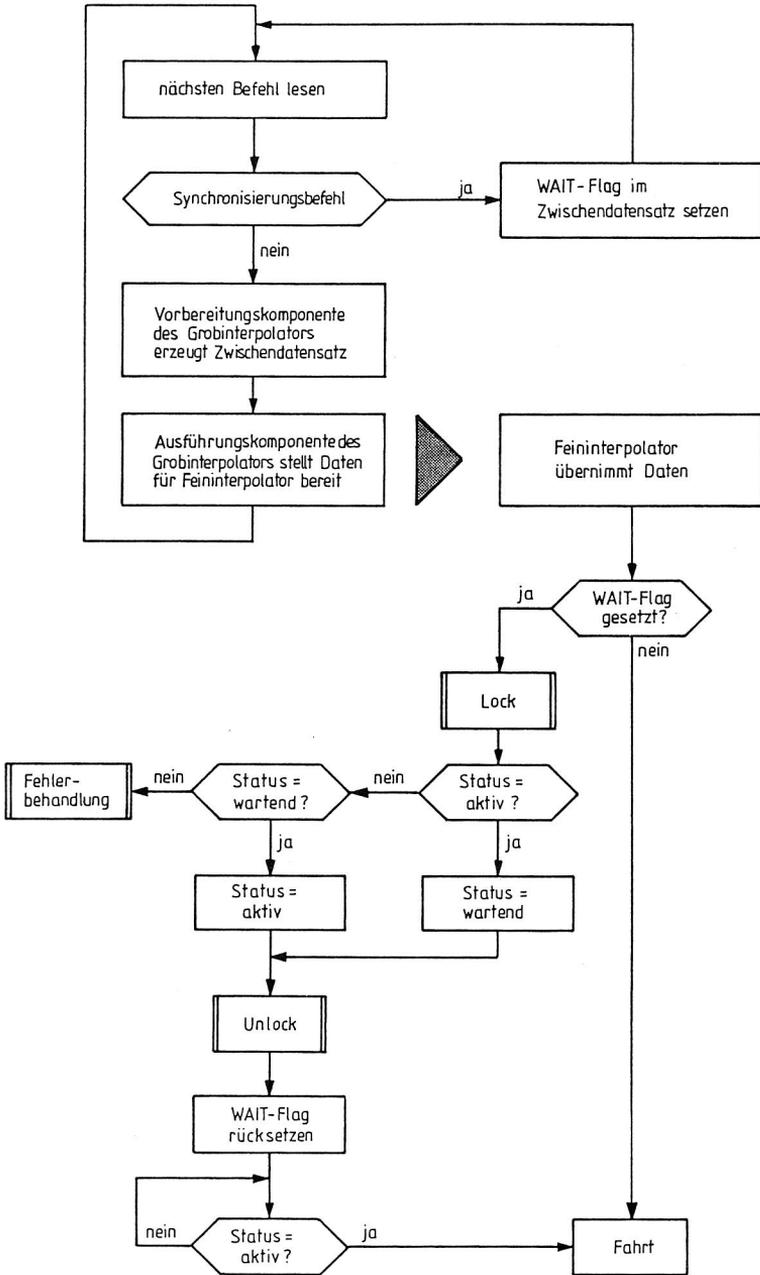
Den Überlegungen von Abschnitt 4.2.3 zufolge läßt sich die Aufgabe der Bewegungskordinierung zweier Roboter in zwei relativ unabhängige Teilaufgaben zerlegen. Die eine Teilaufgabe, die Berechnung einer abhängigen Bewegung, bahn- oder zeitgekoppelt zur unabhängigen Bewegung, behandelt Abschnitt 4.3, während für die zweite Teilaufgabe, die Synchronisation der beteiligten Rechenprozesse im weiteren eine Lösung vorgestellt wird, die den Besonderheiten zweistufiger Interpolatoren Rechnung trägt.

Beim weithin verbreiteten Bahnsteuerungskonzept entstehen Roboterbewegungen durch Aufschalten von Führungsgrößenfolgen auf die Lageregelkreise der einzelnen Achsen (vgl. Bild 63). Zur Bewegungskordinierung zweier Roboter, die mit getrennten Steuerungen arbeiten, ist es erforderlich, die Ausgabe ihrer kinematisch aufeinander abgestimmten Führungsgrößenfolgen miteinander zu synchronisieren. Stellt man sich einstufige Interpolatoren ohne satzbezogene Vorbereitungsrechnung als jeweils in sich abgeschlossene Rechenprozesse vor, muß offenbar derjenige Prozeß, der die Synchronisierungsstelle zuerst erreicht, solange in einen Wartezustand übergehen, bis der andere Prozeß diese Stelle ebenfalls erreicht hat. Für die Dauer des Wartezustandes unterbricht der fragliche Prozeß seine Führungsgrößenabgabe und bringt damit den zugehörigen Roboter zum Stillstand. Im Bewegungsprogramm repräsentiert ein spezieller Synchronisierbefehl diejenige Stelle, an der die abhängige Bewegungsphase beginnt.

Etwas komplizierter wird der Vorgang beim zweistufigen Interpolator mit satzbezogener Vorbereitungskomponente nach Bild 64, weil jeder Interpolator in diesem Fall mit drei Rechenprozessen arbeitet. Bei schrittweiser Abarbeitung des Bewegungsprogramms finden die vorbereitenden Rechenprozesse beider Grobinterpolatoren den Synchronisierbefehl zu irgend-

einem Zeitpunkt, sein Informationsgehalt muß aber die Feininterpolator-Rechenprozesse erreichen, die die eigentliche Synchronisierung vollziehen und zu diesem Zweck in Wartezustände übergehen können. Die Synchronisierungsinformation kann grundsätzlich auf gleichem Weg den Interpolator durchlaufen wie die übrige interne Interpolatorinformation. Innerhalb der Datensätze, die die Verbindung zwischen den Interpolatorkomponenten herstellen, wird sie in Form eines WAIT-Flags repräsentiert (siehe Bild 67). Im Feininterpolator aktiviert diese lokale Variable denjenigen Programmteil, der den gegenseitigen Austausch von Zustandsinformation zwischen beiden Prozessen vornimmt. Derjenige Rechenprozeß, der die Synchronisierungsinformation in Form des WAIT-Flags als erster erhält, setzt die globale Variable Status auf 'wartend' und teilt auf diese Weise dem zweiten Rechenprozeß seinen eigenen Wartezustand mit. Erreicht den zweiten Prozeß die Synchronisierungsinformation ebenfalls, setzt dieser die globale Variable Status wieder auf 'aktiv' und beendet dadurch den Wartezustand des ersten Prozesses. Um undefinierte Zustände zu verhindern, die durch nahezu gleichzeitige Zugriffe auf die globale Variable auftreten können, wird der Zugriff jeweils eines Prozesses durch die Anweisung 'Lock' gesperrt und erst nach dem Ende der Operation durch 'Unlock' wieder freigegeben (siehe Bild 67).

Wenngleich die zwei unabhängigen Feininterpolator-Rechenprozesse durch gegenseitigen Austausch der Zustandsvariable 'Status' für ihre Synchronität zu Satzbeginn des ersten koordinierten Bewegungsschrittes sorgen, laufen doch die beiden Prozesse durch unterschiedliche Zeitbasen ihrer internen Taktgeber in ihrem zeitlichen Bezug unweigerlich wieder auseinander. Wiederholte Synchronisierung über den Austausch von Zustandsvariablen, beispielsweise zu Beginn eines jeden Satzes der abhängigen Bewegungsphase, kann diese Drift zwar begrenzen, läßt aber den Bewegungsablauf durch häufige Zwischenstops sehr unruhig werden. Soll deshalb die Synchronisierung nur einmal, und zwar zu Beginn der ab-



**Bild 67:** Programmablauf für die WAIT-Synchronisation bei zweistufiger Interpolation

hängigen Bewegungsphase, erfolgen, müssen entweder Nachführmaßnahmen vorgesehen werden, die letztlich auf eine Mischung aus Variante 1 und 2 von Bild 55 hinauslaufen - beispielsweise kommt die Regelung der Bahngeschwindigkeit oder des Abstandsvektorbetrags in Betracht - oder die Drift kann von vorneherein durch die Verwendung eines gemeinsamen Zeitrasters ausgeschlossen werden. Für die letzte Variante stellt ein gemeinsamer Interpolationstakt einen ausreichend engen Bezug her, da die Gerätesteuern räumlich nahe beieinander angeordnet sind, und die Signallaufzeiten im Verhältnis zu typischen Interpolationstaktzeiten vernachlässigbar sind. Vorausgesetzt, der Takt kann von extern eingespeist werden, und die Interpolatoren verfügen über interne Mechanismen zum Ausgleich der Rundungsfehler, die aus der begrenzten Wortlänge resultieren, stellt der globale Interpolationstakt die einfachste Lösung des Problems der driftenden Zeitbasen dar.

#### 4.5 Sprachkonzept für die Duplexmontage

Um die Leistungsfähigkeit der Steuerungsmechanismen effizient testen zu können, wurde eine einfache interpretative Sprache entworfen, die den Anforderungen der Duplexmontage Rechnung trägt. Sie basiert auf zwei Typen von Befehlen. Steuerbefehle nehmen Einfluß auf den Ablauf des Programms, während Bewegungsbefehle explizite Roboterbewegungen bewirken. Alle Aktivitäten beider Roboter beschreibt der Programmierer in einem globalen Bewegungsprogramm, aus dem die Steuerung selbsttätig für jeden Roboter ein individuelles Programm generiert.

Bewegungsbefehle bestehen im Kern aus einer abgekürzten Beschreibung des Bahntyps zwischen Start- und Zielpunkt eines Bahnabschnitts. Zwei Klassen von Befehlen sind vorgesehen. Individualbefehle beziehen sich auf jeweils ein Gerät, während koordinierende Befehle beide Geräte im

Zusammenhang ansprechen. Diese Einteilung entspricht dem Anforderungsprofil der Montage forminstabiler Werkstücke und Baugruppen, das beiden Robotern ebenso die Behandlung voneinander unabhängiger Einzelaufgaben abfordert wie die Durchführung gemeinsamer Aufgaben. Geometrischer Bezugspunkt ist in jedem Fall das globale Koordinatensystem  $E_a = (x_a, y_a, z_a)$ .

Für Individualbewegungen ist ein Satz von Bewegungsbefehlen vorhanden, wie er heute gewöhnlich für komfortable industrielle Robotersteuerungen verfügbar ist (siehe Bild 68). Befehle für achsweise unkoordinierte Punkt-zu-Punkt Bewegungen (PPO) sowie für Kreis- (ZRK) und Geradenbewegungen (LIN) im Raum sind implementiert. Den Grundumfang ergänzen die Befehle für Punkt-zu-Punkt Bewegungen mit geglättetem Übergang auf den folgenden Bewegungsabschnitt (PPM) und für beschleunigte Linearbewegungen (ACC), bei denen die Geschwindigkeit kontinuierlich vom angegebenen Anfangswert bis zur Endgeschwindigkeit zunimmt. Als Beschleunigungsstrecke dient die gesamte Länge des Bahnabschnitts. Mit dieser Variation der gewöhnlichen Linearinterpolation läßt sich sehr sanftes Fahrverhalten auch unter Ausnutzung hoher Verfahrgeschwindigkeiten erzielen. Alle Einzelbefehle sind mit einem Präfix versehen, das eine Kennzeichnung des ausgewählten Roboters darstellt.

Aus der Klasse der koordinierenden Befehle ist eine Gruppe von Befehlen zur Beschreibung bahngekoppelter Bewegungen vorgesehen (siehe Bild 68), aus der jeder einheitlich mit einem P als erstem Präfix gekennzeichnet ist. Das zweite Präfix legt den Roboter fest, der für die Dauer des Bahnabschnitts die unabhängige Bewegung ausführen soll, womit für den anderen automatisch die abhängige Bewegung vereinbart ist. Die abgekürzte Beschreibung des Bahntyps bezieht sich ebenso wie der angehängte Parametersatz in jedem Fall auf die unabhängige Bewegung, woraus die Steuerung selbsttätig die abhängige Bewegung generiert. Mit Ausnahme des

P#DRE-Befehls, der den Drehbewegungstyp aus Abschnitt 4.3.3 beschreibt, sorgen alle bahnkoppelnden Befehle dafür, daß der Abstandsvektor über den gesamten Bewegungsabschnitt konstant gehalten wird.

Die dritte Gruppe der Bewegungsbefehle in Bild 68 dient dazu, beide Roboter Bewegungsabschnitte in der abhängigen Bewegungsphase zeitgleich durchfahren zu lassen. Befehle dieses Typs sind durch ein T als erstes Präfix gekennzeichnet. Zeitkoppelnde Befehle sind immer zweizeilig, die erste Zeile enthält die Beschreibung der unabhängigen Bewegung,

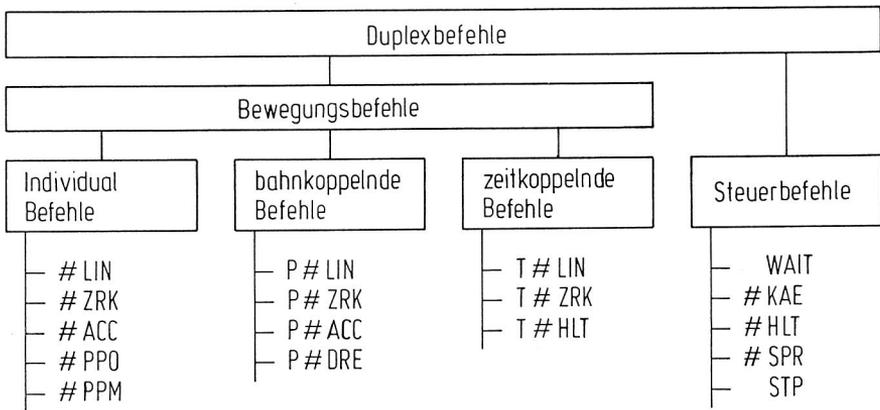


Bild 68: Bewegungs- und Steuerbefehle für die Duplexmontage

während die zweite Zeile lediglich die Bahn der abhängigen Bewegung angibt. Unter der Forderung nach gleicher Ausführungsdauer wird ihre Bahngeschwindigkeit aus der unabhängigen Bewegung abgeleitet. Zeitkoppelnde Befehle erzeugen fließende Bewegungen für beide Geräte, da, bedingt durch die gleiche Ausführungsdauer der Bewegungsabschnitte, beide Bewegungen immer gleichzeitig das Ende eines Bahnabschnitts erreichen und dadurch die Synchronisierung beider Bewegungsabläufe mit der unvermeidlichen Konsequenz von Zwischenstops entfallen kann.

Ein zeitkoppelnder Befehl, der aus einem Zirkular- und einem Haltbefehl zusammengesetzt ist, eignet sich vorzüglich dazu, längliche Werkstücke gegenüber ihrer Ausrichtung bei der Aufnahme gezielt zu verdrehen. Auf diese Weise können Werkstücke sehr elegant durch Engstellen hindurch und um Hindernisse herumgeführt werden. Bedingt durch die häufig beengten Handhabungs- und Fügerräumverhältnisse sind diese Gegebenheiten nicht gerade selten. Bild 69 stellt eine Sequenz von Handhabungsschritten dar, die zum Einhängen einer Feder erforderlich ist. Skizziert sind jeweils die Zustände am Ende eines Bewegungsabschnitts, die zugehörigen Befehle stehen jeweils darunter. Dem ersten Schritt gehen die Bereitstellung der Feder sowie der Greifvorgang voraus. Die Schritte 2, 3 und 4 führen die Feder durch die Engstelle, Schritt 5 dient zum Ausrichten, und Schritt 6 schließt den Montagevorgang ab. Die feinen Bewegungen zum Einfädeln der Federösen in die Haken der Baugruppe sind weggelassen worden.

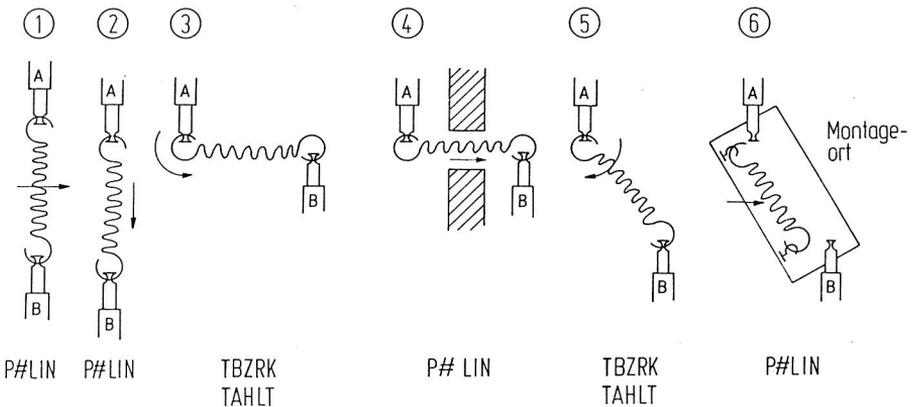


Bild 69: Sechs Schritte einer Federmontage

Die Befehlssyntax sieht eine Befehlskennung vor, die den Bahntyp sowie ein bzw. zwei Präfixe enthält. Sind zwei Präfixe vorhanden, kennzeichnet das erste in jedem Fall koordinierte Bewegungen vom zeit- oder bahngekoppelten Typ,

während das zweite denjenigen Roboter festlegt, der die unabhängige Bewegung ausführt. Kommt nur ein Präfix vor, ist nur der damit angesprochene Roboter gemeint. Lediglich die beiden Steuerbefehle WAIT und STP haben kein Präfix, sie sind von allgemeiner Gültigkeit für beide Roboter. Als Beispiel zum Befehlsaufbau möge die Beschreibung einer beschleunigten Linearbewegung für Roboter B dienen:

BACC X., Y., Z., VS., VZ., SI\$.

Sie enthält als Parameter die drei Koordinaten des Zielvektors, die Geschwindigkeiten zu Anfang und Ende des Abschnitts sowie eine Schaltinformation (SI) zur Bedienung peripherer Einrichtungen.

Zusätzlich zu den Bewegungsbefehlen sind einige hilfreiche Befehle zur Ablaufsteuerung vorhanden (siehe Bild 68). Der Synchronisierbefehl WAIT löst den im Abschnitt 4.4.6 beschriebenen Synchronisiervorgang aus und führt dadurch den Übergang von der unabhängigen in die abhängige Bewegungsphase herbei. Über den Befehl #KAE lassen sich die Konstanten eines Datenfeldes verändern, das u.a. die Überschleifzeit  $T_a$  enthält. Auf diese Weise kann das Übergangverhalten zwischen bestimmten Bahnabschnitten gezielt verändert werden. Der Befehl #HLT läßt den ausgewählten Roboter für eine angegebene Zeit T am Endpunkt der letzten Bewegung verweilen, #SPR ruft einen Sprung auf, dessen Sprungziel der Bewegungsbefehl einer anderen Programmzeile ist, und mit dem Befehl STP wird das Bewegungsprogramm abgeschlossen.

#### 4.6 Struktur eines Testsystemaufbaus

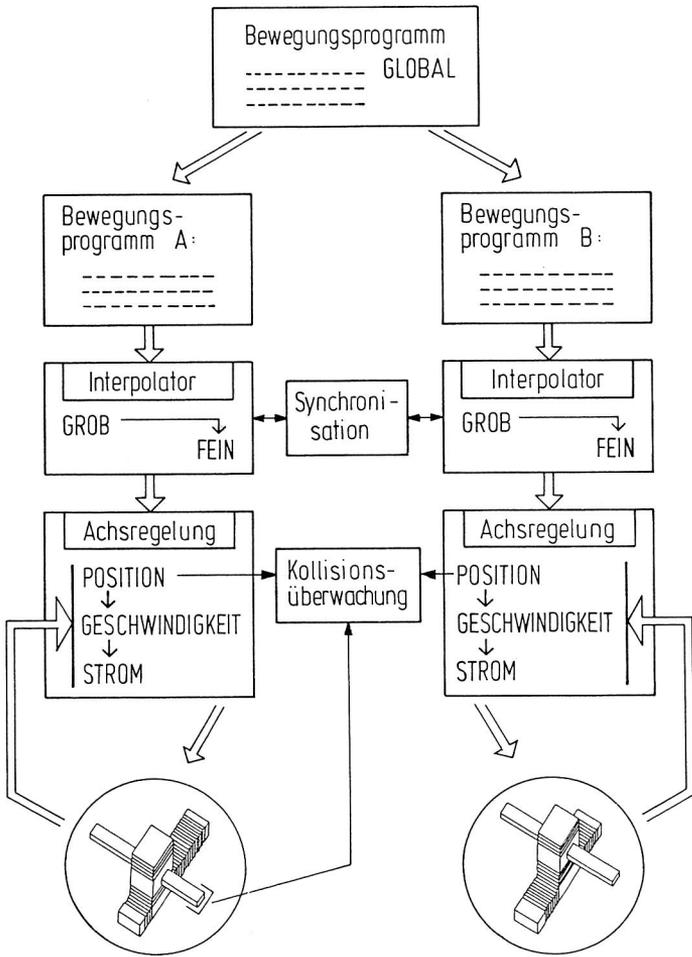
Die praktischen Tests der Steuerungsmechanismen wurden unter Verwendung der im zurückliegenden Abschnitt skizzierten 'Duplexsprache' auf einem eigens für diesen Zweck erstellten Steuerungssystem durchgeführt. Es wurde als reines Experi-

mentiersystem konzipiert, sodaß der Umfang des heute bei Robotersteuerungen üblichen Bedienkomforts nicht vollständig erstellt zu werden brauchte. Ebenso konnten Sicherheitsmaßnahmen auf einen unumgänglichen Teil beschränkt bleiben. Die zwei Steuerungen für die beiden Roboter arbeiten mit je einem Rechner eines industriellen 16-Bit-Mikrorechnersystems und nutzen einen globalen Speicherbereich zum gegenseitigen Austausch von Information. Von den Komponenten der Bahnsteuerung ist der Grobinterpolator in Software realisiert, während der Feininterpolator und die Lageregelung mit der unterlagerten Geschwindigkeits- und Stromregelung aus Hardwarekomponenten besteht (siehe Bild 70).

Das vom Anwender erstellte globale Bewegungsprogramm beschreibt das Bewegungsverhalten beider Roboter der Montagezelle. Davon ausgehend generiert die Steuerung, auf der Basis der im Abschnitt 4.3 abgeleiteten Algorithmen, selbsttätig zwei individuelle Bewegungsprogramme für je einen Roboter. Ein Interpreter, der der Vorbereitungskomponente des Grobinterpolators zugeschlagen ist, liest während der Programmausführung zeilenweise den Programmcode eines solchen Individualprogramms und überführt ihn in eine für den Interpolator verarbeitbare Darstellung. Prinzipiell kann die Generierung der Individualprogramme online oder offline durchgeführt werden. Es wurde die Offline-Variante gewählt, da sie eine geringere Rechnerbelastung zur Folge hat und bessere Testmöglichkeiten als die Online-Variante bietet. Der damit verbundene größere Speicherplatzbedarf - zusätzlich zum globalen Programm müssen ständig zwei vollständige Individualprogramme verwaltet werden - wurde für den Testbetrieb in Kauf genommen.

Zur Sicherung des Betriebs wurde eine Kollisionskontrolle installiert, die während der Programmausführung aktiv ist und Kollisionsgefahren mit Hilfe von Sensoren erfaßt. Da die Kollisionsmöglichkeit als einer der kritischen Punkte bei der Montage mit Duplexsystemen zu werten ist, wurde zur

Erhöhung der Sicherheit eine zweistufige Anordnung der Kollisionskontrolle gewählt, die zur indirekten Überwachung die Winkelkodierer der einzelnen Achsen auswertet und zur direkten Überwachung ergänzend mit einer verformungsempfindlichen Schutzhaube arbeitet (siehe Bild 70). Die indirekte Überwachung basiert auf der Definition variabler Schutzräume [98], deren gegenseitige Überschneidung abgeprüft wird. Eine Voraussetzung für diese Methode ist die Existenz ausreichend



**Bild 70:** Steuerungsstruktur zur Duplexmontage

genauer kinematischer Modelle der Roboter, was bei kartesischer Bauform der Geräte leicht erfüllbar ist. Die Steuerung nutzt beide Überwachungsstrategien ausschließlich schaltend, sie setzt im Fehlerfall sofort die Antriebe still.

## 5. Zusammenfassung

Die aktuelle Problemstellung in der Montage ist vom Spannungsverhältnis zwischen weitreichenden Automatisierungsbestrebungen und wachsender Variantenvielfalt gekennzeichnet. Verstärkt durch den Einfluß abnehmender Produktlebensdauer werden steigende Flexibilitätsforderungen an automatisierte Montagesysteme gestellt, die zu erfüllen es in jedem Fall einer Vielzahl aufeinander abgestimmter Maßnahmen bedarf. In der vorliegenden Arbeit wurde daraus der Aspekt der steuerungstechnischen Maßnahmen aufgegriffen.

Es wurde eine Methode zur Sensorauswahl entworfen, die den Planer von Montagesystemen unterstützen soll. Sie geht von den Aufgaben und Funktionen der gesuchten Sensoren im Montagesystem aus und schränkt auf systematische Art und Weise das Spektrum fraglicher Sensoren ein. Die Frage der erforderlichen Sensorausstattung ist bereits in einem relativ frühen Stadium der Planung relevant, weil unter Umständen der mechanische Aufbau durch Einbeziehung von Sensoren vereinfacht und die Systemverfügbarkeit erhöht werden kann. Zur Berücksichtigung eines möglichst breiten Marktangebotes wurde besonderer Wert auf durchgängige Herstellerneutralität gelegt. Für den Planer stellt ein darauf basierendes, rechnergestütztes Auswahlssystem mit Zugriff auf eine Sensordatenbank ein wertvolles Hilfsmittel dar, um den jeweils neuesten Entwicklungsstand im Sensorbereich berücksichtigen zu können.

Über den automatischen Anlagenbetrieb hinaus lassen sich Sensoren sehr wirkungsvoll zur Erleichterung von Anpassungsarbeiten an Montagesystemen verwenden, die in der Folge von Produktänderungen nötig werden. Neben Vermessungs- und Kontrollarbeiten eignen sich Sensoren unter Einbeziehung geeigneter Steuerungs- und Regelungsmechanismen auch für die Bewegungsprogrammerstellung der Roboter.

Ein neues sensorgestütztes Programmierverfahren wurde entwickelt, das mit minimalem Vorbereitungsaufwand auskommt und prinzipiell ohne manuelle Eingriffe betrieben werden kann. Praktische Untersuchungen zeigen in Übereinstimmung mit den Ergebnissen einer Simulationsstudie, daß der Programmiervorgang auch unter dem Einfluß von Störungen beherrschbar bleibt. Steuerungsmaßnahmen zur Kompensation bekannter Störungen können allerdings sowohl das Abtastverhalten als auch insgesamt die Qualität des Programmierergebnisses spürbar verbessern.

Im allgemeinen wird die Programmiermethode erst durch die Reduktion der aufgenommenen Meßdaten praktisch verwendbar. Der geringe Umfang der verbleibenden Datensätze begünstigt die Verwaltung der Bewegungsprogramme in den Steuerungen und ist Voraussetzung für ihre effiziente Archivierung. Die Datenreduktion muß in der Art erfolgen, daß später mit Hilfe des steuerungsinternen Interpolators die Meßkurve ausreichend gut wiederhergestellt werden kann. Bei einem Vergleich verschiedener Interpolationsverfahren hat sich dabei die HERMITESche Spline-Interpolation als die universellste erwiesen.

Die übliche Montagezellengrundausrüstung mit einem Roboter und mehreren Peripheriegeräten eignet sich nicht für alle Montageaufgaben. In bestimmten Fällen erscheint ein neuer Lösungsansatz vielversprechend, der in Anlehnung an die manuelle Arbeitsweise zwei Roboterarme vorsieht, deren Bewegungen nach den Forderungen der Montageaufgabe miteinander koordiniert werden. Die Wirksamkeit dieses Ansatzes konnte bestätigt werden anhand einer Problemgruppe der automatischen Montage, den forminstabilen Werkstücken und Baugruppen, für die bisher kaum Automatisierungslösungen bekannt geworden sind.

Für dieses Aufgabenspektrum wurden einige Typen koordinierter Bewegungen zusammengestellt und geeignete Algorith-

men zur steuerungstechnischen Umsetzung abgeleitet. Da die Beschreibung derart aufeinander abgestimmter Armbewegungen mit konventionellen Programmiersprachen sehr umständlich ist und zu langen Programmerstellungszeiten führt, wurde ein einfaches Sprachkonzept für die Duplexmontage entworfen, das sich bei Versuchen an Beispielaufgaben bewährt hat. Als möglicher Realisierungsweg wurde die Struktur des Steuerungssystems vorgestellt, das eigens für den praktischen Test der Algorithmen erstellt wurde.

Mit der vorliegenden Arbeit konnte verdeutlicht werden, daß Sensoren und spezifische Steuerungsfunktionen maßgeblich die Flexibilität automatisierter Montagesysteme erhöhen können. Die Weiterentwicklung beider Elemente erscheint als notwendige Voraussetzung dafür, daß auch bei hoher Produkt- und Variantenvielfalt künftige Montagesysteme wirtschaftlich betreibbar sein können.

## 6. Literaturverzeichnis

1. Abele, E.;  
Bäßler, R.;  
Wolf, E.M.: Stand und Entwicklung in der  
Serienmontage.  
wt.-Z. ind. Fertig. 75 (1985),  
S.15-19
2. Frankenhauser, B.;  
Bäßler, R.: Flexibel automatisierte Montage.  
Industrieanzeiger 23/1987, S.40-44
3. Stober, R.: Montageroboter auf dem Vormarsch.  
ZwF 81 (1986) Nr.1, S.15-18
4. Schweizer, M.: Robotereinsatz wird zur Normal-  
investition.  
Roboter 2/88, S.24-28
5. Autorenkollektiv: Rationelle Montage von Produkt-  
varianten.  
Aachener Werkzeugmaschinen  
Kolloquium 1984, S.117-130
6. Classe, D.: Automatisierung der Produktion.  
atp Bd. 28 (1986) Nr.2, S.65-74
7. Feldmann, K;  
Kudorfer, A.: Automatisierte Montagesysteme am  
Beispiel elektromechanischer und  
elektronischer Produkte.  
VDI-Berichte Nr. 479, S.37-44,  
1983, Praxis der Montage-  
automatisierung
8. Abele, E.;  
Wolf, E.M.: Neue Anwendungen von Montage-  
robotern für elektrotechnische  
Produkte.  
wt.-Z. ind. Fertig. 75 (1985) Nr.4,  
S.214-218
9. DIN 1319 Blatt 1:  
Grundbegriffe der Meßtechnik,  
Nov. 1971

10. Balling, G.;  
Fuehrer, D.: Einfache Programmerstellung für Roboter durch sensorgesteuerte Raumpunktgenerierung. Energie & Automation 9 (1987) Heft 4, S.12-14
11. Gzik, H.;  
Utner, W.: Programmierverfahren für Industrieroboter zum Lichtbogenschweißen. VDI-Z 128 (1986) Nr. 17 - Sept.(I), S.673-676
12. Müller, U.: Planung einer wirtschaftlichen Qualitäts- und Funktionsüberwachung in der Einzel- und Serienfertigung. Dissertation RWTH Aachen, Veröffentlicht als Fortschritts-Bericht VDI-Z Reihe 2 Nr. 73, Düsseldorf: VDI-Verlag 1984
13. Eversheim, W.;  
Hausmann, A.: Die richtigen Sensoren auswählen. VDI-Z Band 127 (1985) Nr. 10 - Mai(II), S.373-376
14. Eversheim, W.;  
Hausmann, A.: Planung des Sensoreinsatzes für flexibel automatisierte Montagesysteme mit Industrierobotern. VDI-Z Band 127 (1985) Nr.1/2 - Januar (I/II), S.37-40
15. Wörn, H.: Sensoren für Roboter in Montage und Handhabung. Zwf 81 (1986) 10, S.531-536
16. Dillmann, R.: A Sensor Controlled Gripper with Tactile and Non-Tactile Sensor Environment. Proc. of the 2nd Int. Conf. on Robot Vision and Sensory Controls, Stuttgart 1982, S.159-170
17. Dillmann, R.;  
Hugel, Th.;  
Meier, W.: Ein sensorintegrierter Greifer als modulares Teilsystem für Montage-roboter. Robotersysteme 2 (1986), S.247-252

18. Ahrens, U.: Möglichkeiten und Probleme der Anwendung von Luft- Ultraschallsensoren in der Montage- und Handhabungstechnik. Robotersysteme 1, 1985, S.19-28
19. Rembold, U.; Blume, Ch.; Dillmann, R.; Mörtel, Ch.: Technische Anforderungen an zukünftige Montageroboter. Teil 1. VDI-Z Band 123 (1981) Nr.18, S.763-772
20. Geißelmann, H.: Sensor-Roboter-System zum Vereinzeln und Ordnen von Werkstücken. FhG-Berichte 2-80, S.26-29
21. Nehr, G.; Martini, P.: Die Kopplung eines Werkstückerkennungssystems mit einem Industrieroboter. VDI-Z Band 124 (1982) Nr. 10 - Mai (II), S.375-381
22. Schünemann, T.M.; Lehnen, H.: Berücksichtigung unterschiedlicher Flexibilitätsgrade bei der Investitionsplanung von Industrierobotern. Zwf 78 (1983) 11, S.501-506
23. Identträger im VW-Motorenwerk. Roboter 3/87, S.60-66
24. Kögler, J.: Entwurf und Aufbau eines Lochcodelesers. Studienarbeit am Lehrstuhl für Technische Elektronik der Universität Erlangen-Nürnberg, 1986
25. Beitz, W.: Übersicht über Konstruktionsmethoden. Konstruktion 24 (1972), S.68-72
26. Brockhaus Enzyklopädie, 17. Auflage des großen Brockhaus, 1968

27. Wiendahl, H.P.: Funktionsbetrachtungen technischer Gebilde - Ein Hilfsmittel zur Auftragsabwicklung in der Maschinenbauindustrie. Dissertation TU Aachen, 1970
28. Richter, W.: Grundsätzliche Probleme der Meßtechnik in automatisierten Produktionsprozessen mit Beispielen aus der elektronischen und der Kernstrahlungsmeßtechnik. Dissertation TU Dresden, Sektion Physik, 1971
29. Naumann, G.: Über Möglichkeiten der Meßwertverarbeitung mittels digitaler Rechner. Nachrichtentechnik 20 (1970) H.6, S.201-208
30. Hengsberg, J.; Schmitt, K.H.; Sturm, B.; Winkler, O.: Messen, Steuern und Regeln in der Chemischen Technik. Band 4, Springer Verlag, 3. Auflage, 1983
31. Witte, K.W.: Rationalisierungsreserven nutzen: Flexible Automatisierung in der Montage. Teil 2: Konzipieren flexibel automatisierter Montagesysteme, VDI-Z 125 (1983), Nr. 11 - Juni(1), S.461-467
32. Walter, J.: Systematische Planung flexibel automatisierter Montageanlagen. VDI-Z 127 (1985), Nr. 9 - Mai(1), S.313-318
33. Scholz, W.; Classe, D.: Formales Modell zur Beschreibung von Montageaufgaben. Bei der AV zur Veröffentlichung eingereicht, Mai 1987
34. VDI-Richtlinie 2860: Handhabungsfunktionen, Handhabungseinrichtungen, Begriffe, Definitionen, Symbole. (Entwurf) Oktober 1982

35. Hofmann, D.: Handbuch Meßtechnik und Qualitätssicherung. Friedrich Vieweg & Sohn, Braunschweig Wiesbaden, 1983
36. Doll, T.J.: Nichttaktile Sensoren für Roboter und Sensoreinsatzplanung. Robotersysteme 2 (1986), S.55-62
37. Pahl, G.: Klären der Aufgabenstellung und Erarbeiten der Anforderungsliste. Konstruktion 24 (1972) S.195-199
38. Roth, K.: Einheitliche Systematik der Verbindungen. VDI-Berichte Nr. 493, S.1-12, 1983, Spektrum der Verbindungstechnik - Auswählen der besten Verbindungen mit neuen Konstruktionskatalogen.
39. Warnecke, H.J.: Handbuch Handhabungs- Montage- und Industrierobotertechnik. mi-Verlag moderne Industrie, Nov.1985
40. Weule, H.: Schrauben in der automatischen Montage. VDI-Berichte Nr. 479, S.61-70, 1983, Praxis der Montage-automatisierung
41. Zangemeister, C.: Nutzwertanalyse in der Systemtechnik. München, WITtemannsche Buchhandlung 1970
42. VDI-Richtlinie 2225: Technisch-wirtschaftliches Konstruieren. Berlin Köln, Beuth-Vertrieb GmbH
43. DIN 8593 Teil 0: Fertigungsverfahren Fügen - Einordnen, Unterteilung, Begriffe, (Entwurf) Juni 1984

44. Gruver, W.A.  
et al.: Evaluation of commercially available  
Robot Programming Languages.  
Proc. of the 13th Int. Symp. on  
Industrial Robots, Chicago 1983,  
S.58-68
45. Liebermann, L.J.;  
Wesley, M.A.: Autopass: An Automatic Programming  
System for Computer Controlled  
Mechanical Assembly.  
IBM J. Res. Development 21 : 321
46. Arai, T.;  
Matsumoto, A.: Intermediate Language for Robot  
Controllers.  
Proc. of the 4th Int. Conf. on  
Assembly Automation, 11.-13.  
Oktober 1983, Tokio, S.55-66
47. Popplestone, R.J.;  
Ambler, A.P.;  
Bellos, J.: RAPT: A language for describing  
assemblies.  
The Industrial Robot, Sept. 1978,  
S.131-137
48. Die Robotersprache AML.  
Elektronik 5, 1984, S.132-133
49. Jakob, W.: Praxisgerecht und zukunfts-  
orientiert: SRL.  
Elektronik 26, 1983, S.117-118
50. Lauer, P.: Neues Offline-Programmiersystem für  
Robotersteuerung.  
ZwF 81 (1986) Nr.5, S.253-255
51. Weck, M.;  
Niehaus, Th.;  
Osterwinter, M.: Graphisch interaktives Programmier-  
und Testsystem für Industrieroboter.  
Robotersysteme 2 (1986), S.193-201
52. Milberg, J.;  
Wrba, P.: Roboter-Einsatzplanung und Offline-  
Programmierung mit USIS.  
ZwF 81 (1986) Nr. 9, S.484-488
53. Feldmann, K.;  
Eisele, R.;  
Kleineidam, G.: Verfahrenskette zur Planung und  
Programmierung von Montagesystemen.  
ZwF 82 (1987) Nr. 9, S.521-527

54. Zülke, D.: Offline-Programmierung numerisch gesteuertter Industrieroboter. Dissertation Aachen 1983
55. Peters, K.: Ein Beitrag zur Berechnung und Kompensation von Positionierfehlern an Industrierobotern. Dr. Ing. Diss. Universität Karlsruhe 1985
56. Stute, G.;  
Erne, H.: Neue Entwicklungen in der Steuerungstechnik für Handhabungssysteme. wt-Z. ind. Fertig. 70 (1980), S.505-509
57. Schulz, M.: Sensorunterstützte Programmierung von Raumkurven für das Entgraten mit Industrierobotern. ZWF 81 (1986) Nr. 7, S.378-379
58. Hirzinger, G.: Adaptiv sensorgeführter Roboter mit besonderer Berücksichtigung der Kraft-Momenten-Rückkopplung. Robotersysteme 1 (1985), S.161-171
59. Kuntze, H.;  
Brodbeck, E.;  
Grünbacher, S.;  
Jacubasch, A.: Kraft- und Positionsregelung eines Industrieroboters zur Bearbeitung komplexer Werkstücke. VDI Berichte Nr. 598, 1986, S.99-117
60. Gruhler, G.: Programmierung von Bearbeitungsrobotern durch sensorgesteuertes Nachführen. wt-Z. ind. Fertig. 73 (1983), S.165-168
61. Bollinger, J.G.: Automatisches Programmieren von Schweißrobotern. wt-Z. ind. Fertig. 71 (1981), S.481-484

62. Kemkens, K.;  
Becker, N.;  
Gerbig, V.;  
Lechner, J.;  
Poleschinski, R.:  
Regelung von Roboterbewegungen  
aufgrund optischer Abstandsmessung.  
VDI Berichte Nr. 598, 1986,  
S.133-143
63. Classe, D.;  
Scholz, W.:  
Verfahrensbeschreibende Parameter-  
Schlüssel zur systematischen  
Montageautomatisierung.  
wt Werkstattstechnik 77 (1987),  
S.684-689
64. Hirzinger, G.:  
Force Feedback Problems in  
Robotics.  
Second IASTED Symposium on  
Modelling, Identification and  
Control, Davos, March 1982
65. Paul, R.;  
Shimano, B.:  
Compliance and Control.  
Proc. Joint Autom. Control Conf.  
San Francisco, USA, 1976
66. Whitney, D.E.:  
Historical perspective and state of  
the art in robot force control.  
Proc. 1985 IEEE Int. Conf. on  
Robotics and Automation, St. Louis,  
March 25-28, 1985
67. Hirzinger, G.;  
Landzettel, K.:  
Sensory feedback structures for  
robots with supervised learning.  
Proc. 1985 IEEE Int. Conf. on  
Robotics and Automation, St. Louis,  
March 25-28, 1985
68. Feldmann, K.;  
Classe, D.:  
Sensor aided robot-programming.  
Proc. of the 5th Int. Conf. on  
Robot Vision and Sensory Controls,  
Amsterdam, Oct. 29-31, 1985,  
S.369-382
69. Schmidt, H.:  
Kraftregelung eines Industrie-  
roboters mit Mikrorechner.  
Diplomarbeit am Lehrstuhl für  
Elektrische Antriebe und  
Steuerungen der Universität  
Erlangen-Nürnberg, 1985

70. Wurmthaler, Ch.: Ein Beitrag zum Entwurf von Zustandsreglern und deren Einsatz unter praxisnahen Randbedingungen. Dr.-Ing. Dissertation, Universität Erlangen-Nürnberg, 1979
71. Hippe, P.; Wurmthaler, Ch.: Zustandsregelung. Springer Verlag Berlin, Heidelberg, New York, Tokyo, 1985
72. Classe, D.: Steuerungssystem für Montage-roboter. Automatisierungstechnik at, 34.Jahrgang, Heft 10 (1986), S.381-387
73. Pfaff, G.; Meier, Ch.: Regelung elektrischer Antreibe II. Oldenbourg Verlag, 1982
74. Hopfengärtner, H.: Modellbildung und Regelung elektrischer Servoantriebe am Beispiel eines Industrieroboters. Dr.-Ing. Dissertation, Universität Erlangen, 1980
75. Budo, A.: Theoretische Mechanik. VEB Verlag der Wissenschaften, Berlin 1974
76. Beitz, W.; Küttner, K.H.: Dubbel: Taschenbuch für den Maschinenbau. 15. Auflage, Springer Verlag, 1983
77. Spingler, M.: Mathematische Modellierung des dynamischen Verhaltens eines dreiachsigen Industrieroboters. Studienarbeit am Institut für Regelungstechnik, Universität Erlangen-Nürnberg, 1986
78. Welker, J.: Erstellung eines Simulationssystems zur Untersuchung von Konturverfolgungsalgorithmen für einen Roboter. Diplomarbeit am Institut für Regelungstechnik, Universität Erlangen-Nürnberg, 1987

79. Gruhler, G.: Geometrieverarbeitung für selbsttätig programmierte Bearbeitungsroboter.  
wt-Z. Ind. Fertig. 74 (1984), S.325-328
80. Gruhler, G.; Keppeler, M.: Automatische Datenreduktion und Auswahl der Interpolationsart bei Roboterbewegungsprogrammen.  
HGF Kurzberichte 84/1 (Loseblattsammlung). Essen: Girardet 1984
81. Späth, H.: Spline-Algorithmen zur Konstruktion glatter Kurven und Flächen.  
R. Oldenbourg Verlag, 1973
82. Keppeler, M.: Führungsgrößenerzeugung für numerisch bahngesteuerte Industrieroboter.  
ISW-Berichte 53, Universität Stuttgart, 1984
83. Törnig, W.: Numerische Mathematik für Ingenieure und Physiker.  
Band 2, Springer Verlag, 1979
84. Bollinger, J.; Duffie, N.: Computer Algorithms for High Speed Continuous-Path Robot Manipulation.  
Annals of the C.I.R.P. 28 (1979) Nr.1, S.391-395
85. Friedrich, K.: Datenreduktion bei sensorunterstützter Roboterprogrammierung.  
Studienarbeit am Lehrstuhl für Technische Elektronik, Universität Erlangen-Nürnberg, 1987
86. Hussaini, S.S.; Jakopac, D.E.: Multiple Manipulators and Robotic Workcell Coordination.  
Proc. of the Int. Conf. on robotics and automation, 8.-10. April 1986, San Francisco, S.1236-1241

87. Blanchfield, P.; Benzeltout, B.; Rhodes, D.J.: Multi-Robot Cell Controller Using an Expert System Approach
88. Soroka, B.I.; Movich, R.C.: Programming and Simulating a Three-Armed Cartesian Robot. Proc. of the Int. Conf. on robotics and automation, 8.-10. April 1986, San Francisco, S.1766-1771
89. Kohno, M.; Miyakawa, A.; Masahiro, H.: Real Time Synchronization of Two Robots for Coordinated Assembly. Proc. of the 16th International Symposium on Industrial Robots, 30.Sept.-2.Oct. 1986, Brüssel, S.219-228
90. Tarn, T.I.; Bejczy, A.K.; Yun, X.: Coordinated Control of Two Robot Arms. Proc. of the Int. Conf. on robots and automation, 8.-10. April 1986, San Francisco, S.1192-1203
91. Ishida, T.: Force control in coordination of two arms. Proc. of the Int. Joint Conf. Artificial Intelligence, Aug. 1977, S.717-722
92. Alford, C.O.; Belyeu, S.M.: Coordinated Control of Two Robot Arms. Int. Conf. on Robotics, Atlanta, Georgia, 13.-15. März, 1984, S.468-473
93. Zheng, Y.F.; Sias, F.R.: Two Robot Arms in Assembly. Proc. of the Int. Conf. on robotics and automation, 8.-10. April, San Francisco, S.1230-1235
94. Warnecke, H.-J.; Frankenhauser, B.: Montage biegeschlaffer Teile mit Industrierobotern. wt-Z. ind. Fert. 76 (1986) Nr.1, S.8-11

95. Nolting, F.W.: Flexible Automatisierung von Fügeprozessen in der Feinwerktechnik-Einhaken von Zugfedern. Feinwerktechnik und Meßtechnik 94 (1986) 7, S.455-458
96. Binder, D.: Interpolation in numerischen Bahnsteuerungen. Springer Verlag, Berlin, Heidelberg, New York, 1979
97. Schütz, H.F.: Entwurf und Realisierung leistungsfähiger Interpolatoren für Industrieroboter-Bahnsteuerungen auf einem Mikrorechner. Diplomarbeit, Lehrstuhl für Elektrische Antriebe und Steuerungen, Universität Erlangen-Nürnberg, 1985
98. Freund, E.; Hoyer, H.: Ein Verfahren zur automatischen Kollisionsvermeidung für Roboter. Robotersysteme 1 (1985), S.67-73

## Lebenslauf

- 1954 Detlef Classe, geb. am 1. Juli 1954 in Halle/  
Saale  
Eltern: Hans Joachim Classe  
Marianne Classe, geb. Formen
- 1961 - 1965 Besuch der Grundschule in Wesel
- 1965 - 1974 Besuch des Staatl. Gymnasiums und des Städt.  
Naturwissenschaftlichen Gymnasiums in Wesel
- 1974 - 1975 Studium der Physik an der Philips-Universität  
Marburg
- 1975 - 1981 Studium der Elektrotechnik an der Universität  
GH Siegen, Studienschwerpunkt: Automatisie-  
rungstechnik, Abschluß: Diplom-Ingenieur
- 1981 - 1982 Studienreise nach Zentralamerika
- 1982 Projektierungsingenieur bei der Schering AG,  
Berlin
- Ab 1983 Wissenschaftlicher Mitarbeiter am Lehrstuhl  
für Fertigungsautomatisierung und Produktions-  
systematik der Universität Erlangen-Nürnberg
- 1983 - 1988 Akademischer Rat am selben Lehrstuhl
- Ab 1986 Oberingenieur für Steuerungs- und Sensortech-  
nik am selben Lehrstuhl
- 1988 Promotionsprüfung am 26. September 1988 an der  
Technischen Fakultät der Universität Erlangen-  
Nürnberg



# Reihe

# Fertigungstechnik

# Erlangen

**Innovationspotentiale in der rechnerintegrierten Produktion durch  
wissensbasierte Systeme**

Andreas Hemberger. 208 Seiten, 107 Bilder. 1988. Kartoniert.

**Beitrag zur Steigerung der Flexibilität automatisierter Montagesy-  
steme durch Sensorintegration und erweiterte Steuerungskonzepte**

Detlef Classe. 194 Seiten, 70 Bilder. 1988. Kartoniert.

**Projektierung von Montagesystemen**

Friedrich-Wilhelm Nolting. 201 Seiten, 107 Bilder, 1 Tabelle. 1989.  
Kartoniert.

**Nutzungsgradsteigerung von Montagesystemen durch den Einsatz  
der Simulationstechnik**

Karsten Schlüter. 177 Seiten, 97 Bilder. 1989. Kartoniert.

**Aufbau von Modellen zur Lageregelung von Industrierobotern**

Shir-Kuan Lin. 168 Seiten, 46 Bilder. 1989. Kartoniert.