

# Kollisionserkennung und -vermeidung für einen mobilen Manipulator

Thomas Wösch und Werner Neubauer

Corporate Technology, Information & Communications  
Siemens AG  
D-81739 München  
{thomas.woesch,werner.neubauer}@mchp.siemens.de

**Zusammenfassung** Im Bereich der mobilen Servicerobotik sind Interaktionsmöglichkeiten zwischen Mensch und Maschine erwünscht, die deutlich über derzeit verfügbare hinausgehen. In diesem Beitrag werden ein *Kollisionsvermeidungssystem* und *Kontaktreflexsystem* vorgestellt, die zusammen mit einer „*künstlichen Haut*“ die unmittelbare Interaktion im gleichen Arbeitsbereich ermöglichen. Die berührungslose und taktile Sensorik des Systems dient der Umgebungswahrnehmung und trägt darüber hinaus zur Sicherheit des Benutzers bei. Beispielhafte Ergebnisse werden mit Hilfe einer Simulation und anhand von Experimenten mit einem realen Roboter gezeigt.

## 1 Einleitung

Mobile Serviceroboter werden bereits erfolgreich bei Transport- und Reinigungsaufgaben eingesetzt. Mehr und mehr rückt der Arbeitsbereich der Roboter dabei in die Nähe des Menschen, wenn zum Beispiel ein Reinigungsroboter während der Öffnungszeiten im Supermarkt reinigt [1]. Noch näher kommt der Mensch dem Roboter, sobald der Roboter unmittelbare Aufgaben für den Menschen ausführt, wie zum Beispiel Hol- und Bringdienste, oder wenn Aufgaben sogar interaktiv durchgeführt werden. Dazu soll der Roboter entsprechend nachgiebig auf Berührungen reagieren oder auch manuell „geführt“ werden können (Softrobotik). Solche Fähigkeiten benötigt zum Beispiel ein Haushaltsassistent. Für derartige Szenarien braucht der Roboter eine Steuerung, die die Sicherheit von Personen und Umgebung gewährleistet. Gleichzeitig muß die Roboterbewegung für den Menschen vorhersehbar und plausibel sein, um die Akzeptanz solcher Systeme zu gewährleisten. Dieser Beitrag beschäftigt sich mit der Umgebungswahrnehmung und der Bewegungssteuerung von Robotern in Alltagsumgebungen von durchschnittlicher Komplexität, wie beispielsweise einer Büroumgebung.

Für die Bewegungssteuerung von Robotern wurden in den letzten 10 Jahren bevorzugt probabilistische Planer entwickelt [2,3]. Planer dieser Klasse zeichnen sich besonders durch die Eigenschaft aus, verhältnismäßig einfach auf höherdimensionale Problemstellungen erweiterbar zu sein. Die Beliebtheit dieser Planer ist in der steigenden Effizienz der Algorithmen im Bereich der Computergrafik zur Berechnung von Abstandsinformationen [4,5] aus einem Umgebungs-



**Abbildung 1.** Experimentalsystem

dell und in der steigenden Rechnerleistung begründet. Trotz der ständig schneller werdenden Planer benötigt man für eine erfolgreiche Bewegungsplanung ein System, welches auf unvorhergesehene Hindernisse reagiert. Um eine Bewegungsplanung zu ermöglichen, ist ein vollständig bekanntes Umgebungsmodell erforderlich. Im Bereich der Servicerobotik kann jedoch im allgemeinen mit der eingeschränkten Sensorik die Umgebung nur unvollständig wahrgenommen werden. Die Umgebung ist zudem in den meisten Fällen dynamisch, so daß wiederholtes Neuplanen erforderlich wäre. Dabei können Kollisionen unter Umständen nicht in Echtzeit vermieden werden. Eine Möglichkeit, dieses Problem zu lösen, wird in [6] vorgeschlagen. Dabei wird ein Bewegungsplan generiert, der bei der Ausführung einen lokalen reaktiven Handlungsspielraum zuläßt.

Im vorliegenden Beitrag wird ein rein reaktiver Ansatz zur Bewegungssteuerung innerhalb dieses Handlungsspielraumes vorgeschlagen. Ähnlich der Potentialfeldmethode werden dabei zielführende und kollisionsvermeidende Bewegungsvorgaben überlagert. Für sehr viele Situationen erzeugt dieses Vorgehen bereits das gewünschte Roboterverhalten. Da die Potentialfeldmethode jedoch zu lokalen Minima neigt, wird dem System zu einem späteren Zeitpunkt noch eine Planungskomponente hinzugefügt.

Zur Erzeugung der kollisionsvermeidenden Bewegungsvorgaben verwendet die vorgeschlagene Steuerung ein 3D-Modell der Umgebung, welches aus den Sensordaten eines Laserscanners berechnet wird. Sobald sich der Roboter einem



Abbildung 2. Laser-Scanner

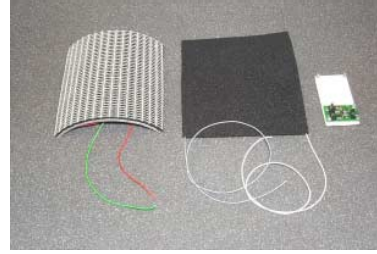


Abbildung 3. Taktile Sensor

Objekt im Umgebungsmodell unzulässig nähert, ermittelt ein *Kollisionsvermeidungssystem*, basierend auf den berechneten Abstandsinformationen, eine Bewegung, die den Arm in einen sicheren Abstand relativ zu den Hindernissen führt. Aufgrund der hohen Dynamik der Roboterumgebung und der zum Teil großen Abschattung des Sensorsichtbereiches durch den eigenen Roboterarm ist das Umgebungsmodell unvollständig. Deshalb ist das gesamte Robotersystem zusätzlich mit taktilen Sensoren umgeben, siehe Abbildung 1 und 3. Diese haben die Funktion einer „künstlichen Haut“<sup>1</sup>. Sollte es zu einem oder mehreren Kontakten zwischen Hindernissen und dem Roboter kommen, leitet ein *Kontaktreflexsystem* abhängig von Ort und Anpreßdruck der Kontakte eine entsprechende Ausweichbewegung ein.

Die Ergebnisse werden in einer Simulationsumgebung und auf dem in Abbildung 1 dargestellten Experimentalsystem demonstriert. Das Robotersystem besteht aus einem redundanten Roboterarm mit 8 Freiheitsgraden, der auf einer nicht-holonomen mobilen Plattform mit Differentialantrieb montiert ist. Am Kopf des Experimentalsystems befinden sich ein Laserscanner und eine Stereokamera. Beide Sensoren liefern die benötigten 3D-Tiefeninformationen. Für eine detailliertere Beschreibung des Experimentalaufbaus und des gesamten Steuerungsansatzes siehe [7].

## 2 Umgebungserfassung

Für die Erstellung und Aktualisierung des Umgebungsmodells stehen drei Informationsquellen zur Verfügung. Zunächst hat der Roboter Kenntnis über die eigene Position in der Umgebung und über die Position des Roboterarmes. Mit Hilfe der bekannten Robotergeometrie kann das Kollisionsvermeidungssystem Selbstkollisionen ausschließen. Das Navigationssystem des Roboters [8] besitzt eine zweidimensionale Karte der Umgebung. Diese Karte wird entweder vorgegeben, oder der Roboter erstellt und aktualisiert diese selbst. Wände und andere Objekte dieser Karte können zusammen mit der Robotergeometrie ebenfalls in

<sup>1</sup> Siemens-Patent, amtl. Aktz. 19959703.0

das dreidimensionale Umgebungsmodell übernommen werden. Die dritte Informationsquelle zur Aktualisierung des Umgebungsmodells sind alle Sensoren, also der Laserscanner, die Stereokamera und die taktile Haut.

Der verwendete Laserscanner kann die Umgebung nur in einer Ebene wahrnehmen. Für eine dreidimensionale Szenenansicht wird der Scanner mit einem Servoantrieb über die Szene geschwenkt (Abbildung 2). Aus den Sensorrohdaten wird ein 3D-Oberflächenmodell der Szenenansicht generiert.

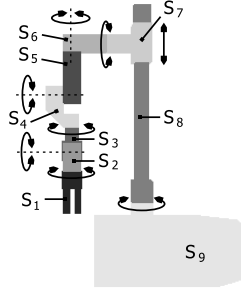
Die Stereokamera liefert wesentlich schneller ein dreidimensionales Bild der Szene, da sie nicht geschwenkt werden muß. Die Tiefendaten der Stereokamera sind ohne den Einsatz von strukturiertem Licht jedoch dünn besetzt, falls die Objekte nicht genügend Struktur aufweisen.

Die taktile Haut dient im wesentlichen dazu, bei unvollständigem Umgebungswissen die Sicherheit für Mensch und Maschine zu gewährleisten. Jedes Flächenstück der Haut liefert dazu Position und Anpreßdruck einer Kontaktstelle. Abbildung 3 zeigt ein Flächenstück, bestehend aus zwei leitfähigen Schaumstoffschichten, einer netzförmigen Isolationsschicht und einer Auswerteelektronik. Im Ruhezustand werden die aufeinanderliegenden Schaumstoffschichten durch die Isolationsschicht getrennt gehalten. Bei einem Kontakt werden die Schaumstoffflächen aufeinandergepreßt. Über Spannungsteiler und Übergangswiderstand beider Schichten werden der Ort des Kontakts und die Größe des Drucks ermittelt. Alle Elektroniken der Hautstücke sind durch ein Bussystem verbunden, so daß auch mehrere Kontakte gleichzeitig detektiert werden können.

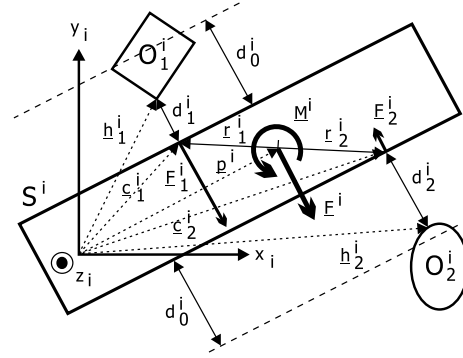
### 3 Bewegungssteuerung

Die mit dem Experimentalsystem in Abbildung 1 zu realisierenden Manipulationsaufgaben, lassen sich in verschiedene Bewegungsklassen einteilen. Die nachfolgend aufgelisteten Bewegungsklassen leiten sich aus den unterschiedlichen Anforderungen für die Einhaltung von Position und Orientierung des Endeffektors sowie die Nutzung der Redundanz des Roboterarms während des Bewegungsablaufes ab.

- **Aufgabenorientierte Bewegung:** Die Endeffektorposition *und* -orientierung folgt einer vorgegebenen Trajektorie. Das Kollisionsvermeidungssystem kann bei Hindernissen in Armnähe die Redundanz des Roboterarmes für Ausweichbewegungen nutzen. Ist die Ausweichbewegung nicht ohne Abweichen des Endeffektors von seiner vorgeschriebenen Trajektorie möglich, muß die Trajektorie neu geplant werden.
- **Semi-Aufgabenorientierte Bewegung:** Hier muß gegenüber der aufgabenorientierten Bewegung für die Endeffektorlage nur Position *oder* Orientierung eingehalten werden. Ein Beispiel hierfür ist der Transport eines offenen Flüssigkeitsbehälters, bei dem die Position eventuell abweichen darf, jedoch die Orientierung eingehalten werden muß.
- **Umgebungs-Aufgabenorientierte Bewegung:** Bei dieser Bewegung wird die Restriktion für die Endeffektorlage weiter gelockert. Abweichungen von der vorgegebenen Trajektorie für den Endeffektor sind bis zu einem gewissen



**Abbildung 4.** Segmente  $S^i$  des Roboters dargestellt in unterschiedlichen Grautönen.



**Abbildung 5.** Geometrische Definitionen und virtuelle Kräfte und Momente für ein vereinfachtes Robotersegment  $S^i$  mit 2 Hindernissen in dessen Sicherheitsbereich.

Grad zulässig, sofern ein Kompromiß mit der Hindernisvermeidung gefunden werden kann.

- **Umgebungsorientierte Bewegung:** Hier ist keine Bewegung für den Endeffektor vorgegeben. Bewegungen werden bei dieser Klasse ausgelöst, sobald sich ein Hindernis dem Roboter unzulässig nähert.

### 3.1 Roboter-Segmentierung

Der Roboter wird, entsprechend seiner Kinematik, in Segmente  $S_i \in \mathbb{S}$  aufgeteilt. Abbildung 4 zeigt  $I = 9$  Segmente  $S^i$  ( $1 \leq i \leq I$ ) des Roboters in unterschiedlichen Grautönen. Zu bemerken ist an dieser Stelle, daß die nicht-holonome Plattform und der Roboterarm nicht getrennt, sondern als eine kinematische Kette betrachtet werden. Um jedes Segment ist ein Sicherheitsbereich  $\mathbb{Z}^i \subseteq \mathbb{R}^3$  angeordnet. Sobald der Abstand eines Robotersegment  $S_i \in \mathbb{S}$  zu einem Hindernis den minimalen Sicherheitsabstand  $d_0^i > 0$  unterschreitet, wird diese Sicherheitszone verletzt. Die Berechnung der daraufhin einzuleitenden Ausweichbewegung wird im folgenden Abschnitt beschrieben.

### 3.2 Kollisionsvermeidungsstrategie

Abbildung 5 zeigt ein vereinfachtes Segment  $S^i$  mit  $L = 2$  Hindernissen innerhalb dessen Sicherheitsbereichs  $\mathbb{Z}^i$ . Für jedes Hindernis berechnet das Kollisionsvermeidungssystem eine virtuelle Kraft  $\underline{F}_l^i \in \mathbb{R}^3$  ( $1 \leq l \leq L$ ). Diese liegt parallel zum minimalen Abstandsvektor zwischen Hindernis und Segment

$$\underline{d}_l^i = \underline{h}_l^i - \underline{c}_l^i \quad . \quad (1)$$

Die Stärke der Kraft ist proportional zur Eindringtiefe

$$e_l^i = d_0^i - d_l^i \quad . \quad (2)$$

mit

$$d_l^i = \|\underline{h}_l^i - \underline{c}_l^i\| \quad . \quad (3)$$

Daraus ergibt sich eine virtuelle Kraft

$$\underline{F}_l^i = \begin{cases} -K_f^i e_l^i \nabla(d_l^i) & d_l^i \leq d_0^i \\ 0 & d_l^i > d_0^i \end{cases} \quad (4)$$

mit

$$\nabla(d_l^i) = \frac{\underline{h}_l^i - \underline{c}_l^i}{d_l^i} \quad . \quad (5)$$

Der konstante Faktor  $K_f^i$  in (4) berücksichtigt den Dimensionsunterschied zwischen Kraft und Länge. Da die Eindringweite (2) immer positiv ist, erhält man für  $K_f^i > 0$  eine Kraft, die das Armsegment vom Hindernis abstößt. Im Falle mehrerer generierter Kräfte, an demselben Segment, werden diese zu einer resultierenden Kraft

$$\underline{F}^i = \sum_{l=1}^L \underline{F}_l^i \quad (6)$$

zusammengefaßt. Diese Kraft setzt dabei an einem Kontrollpunkt  $\underline{p}^i$  (7) des Segments  $S^i$  an. Die Position des Kontrollpunktes errechnet sich unter Verwendung der Segmentaufpunkte  $\underline{c}_n^i$  zu

$$\underline{p}^i = \frac{1}{N} \sum_{n=1}^N \underline{c}_n^i \quad . \quad (7)$$

Betrachtet man den Kontrollpunkt  $\underline{p}^i$  und die einzelnen Kräfte  $\underline{F}_l^i$ , so ergibt sich ein Moment  $\underline{M}^i \in \mathbb{R}^3$  zu

$$\underline{M}^i = K_m^i \sum_{l=1}^L \underline{r}_l^i \times \underline{F}_l^i \quad (8)$$

mit

$$\underline{r}_l^i = \underline{c}_l^i - \underline{p}^i \quad .$$

$K_m^i$  dient dabei als Verstärkungsfaktor. Wenn sich nur ein Hindernis im Sicherheitsbereich eines Segments befindet, bildet sich kein Drehmoment um den Kontrollpunkt. Die Kraft  $\underline{F}^i$  (6) und das Moment  $\underline{M}^i$  (8) werden zu einem verallgemeinerten Kraft-Momenten-Vektor

$$\underline{\delta}^i = \begin{bmatrix} \underline{F}^i \\ \underline{M}^i \end{bmatrix} \quad (9)$$

zusammengefaßt, der nun alle auf Segment  $S^i$  wirkenden virtuellen Kräfte und Momente vereint.

Um aus dem Kraft-Momenten-Vektor (9) eine Ausweichbewegung zu bestimmen, wird dieser mit Hilfe der transponierten Jakobimatrix  $\left(\underline{J}_p^i\right)^T$  [9] in den Gelenkraum abgebildet. Damit erhält man den resultierenden Gelenk-Momenten-Vektor

$$\underline{\tau}^i = \left(\underline{J}_p^i\right)^T \underline{\delta}^i \quad . \quad (10)$$

Betrachtet man alle Segmente, erhält man für das Gesamtsystem den Gelenk-Momenten-Vektor

$$\underline{\tau} = \sum_{i=1}^I \underline{\tau}^i \quad . \quad (11)$$

Durch die Überlagerung der Gelenk-Momenten-Vektoren (10) der Einzelsysteme zu einem gemeinsamen Vektor (11) wird das gesamte System vor möglichen Kollisionen bewahrt. Durch Multiplikation von (11) mit der Verstärkungsmatrix  $\underline{K}_q(\underline{q})$ , die unter anderem systembedingte Restriktionen, wie beispielsweise Gelenkwinkelbeschränkungen, maximale Geschwindigkeiten oder Beschleunigungsgrenzwerte, berücksichtigt, erhält man schließlich den Gelenkwinkelgeschwindigkeitsvektor

$$\underline{\dot{q}} = \underline{K}_q(\underline{q}) \underline{\tau} \quad . \quad (12)$$

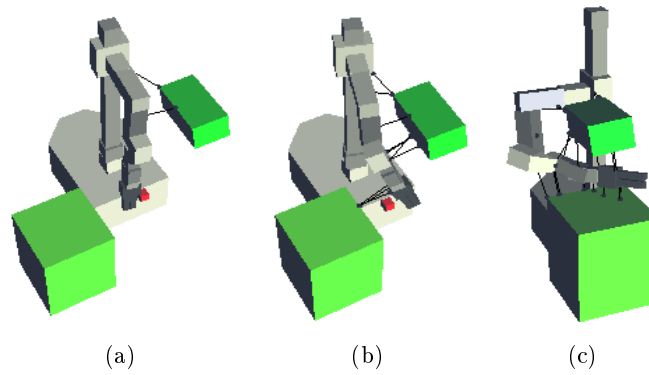
Mit obigen Geschwindigkeitsvektor (12) werden die Gelenke des Roboters angesteuert, und es kommt zu einer Ausweichbewegung.

Die taktile Sensorik des Roboters liefert bei Kontakten unmittelbar Abstandsvektoren der Länge  $d_i^i = 0$ . Diese Abstandsvektoren werden ohne den Umweg über das Umgebungsmodell in obigem Verfahren, zusammen mit den modellbasiert berechneten Abstandsvektoren, verwendet. Der Betrag des Kontaktdrucks wird dabei als Maß für den Abstand verwendet, der Normalenvektor zur Roboter Oberfläche als Richtungsvektor und der Ort des Kontakts als Aufpunkt am entsprechenden Segment.

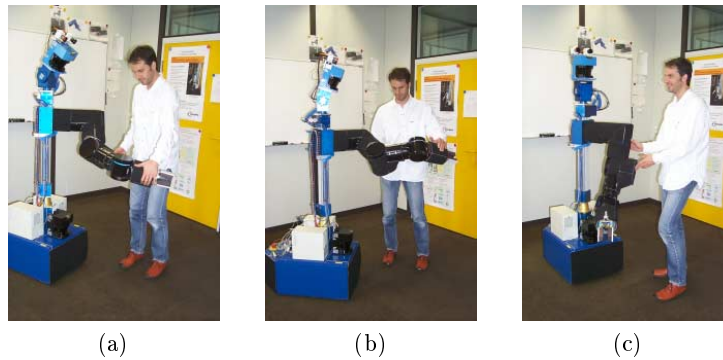
## 4 Ergebnisse

Abbildung 6 zeigt Simulationsergebnisse, aus denen eine qualitative Aussage über die Auswirkung des Kollisionsvermeidungssystem auf das Bewegungsverhalten des Roboters abgeleitet werden kann. Der Roboter hat in diesem Szenario keine Aufgabe zu lösen, er soll lediglich Hindernissen ausweichen. Tatsächlich soll er hier nur mit seinem Arm ausweichen, Position und Orientierung der Plattform bleiben unverändert. Die Positionen der Hindernisse sind dem System zu jedem Zeitpunkt bekannt. Zwei Hindernisse bewegen sich in Abbildung 6a in Richtung Roboter. Sobald ein Hindernis in einen Sicherheitsbereich des Roboters gelangt, siehe Abbildung 6a, berechnet das Kollisionsvermeidungssystem eine geeignete Ausweichbewegung, siehe Abbildungen 6b-c.

Abbildung 7 zeigt die Funktionsweise des Kontaktreflexsystems am realen Roboter. Die Steuerung des Roboters besitzt lediglich Informationen über Lage und Geometrie des Arms und der Plattform. Die Umgebung wird in diesem



**Abbildung 6.** Simulationsbeispiel einer empfohlenen Armbewegung des Kollisionsvermeidungssystems, wenn zwei bekannte zum Roboter sich hin bewegende Objekte in dessen Sicherheitsbereiche gelangen.



**Abbildung 7.** Beispiel einer Ausweichbewegung infolge eines physikalischen Kontakts am realen System.

Anwendungsbeispiel nur über die taktile Sensorik wahrgenommen. Das Kollisionsvermeidungssystem ist deaktiviert, um unmittelbare Interaktionen mit dem Roboter zu ermöglichen. Das Kontaktreflexsystem kann somit von einem Anwender benutzt werden, um den Roboter interaktiv in eine gewünschte Position zu „führen“ (siehe Abbildungen 7a-c).

## 5 Zusammenfassung und Ausblick

Der vorliegende Beitrag beschreibt eine Methode zur Kollisionsvermeidung eines auf einer nicht-holonomen Plattform montierten redundanten Manipulators. Das Kollisionsvermeidungssystem generiert Ausweichbewegungen, die die Kollisionsfreiheit zu dynamischen Objekten sicherstellen. Darüber hinaus wurde ein Kontaktreflexsystem vorgestellt, das über einen taktilen Sensor auf physika-

liche Kontakte reagiert, indem geeignete Ausweichbewegungen eingeleitet werden. Beide Ansätze beruhen auf der gleichen Strategie und können deshalb einfach zu einem Gesamtsystem integriert werden. Die resultierenden Bewegungen des Roboters erscheinen „plausibel“ und fügen sich deshalb harmonisch in das Interaktionsszenario Mensch-Maschine ein. Für die Überlagerung der Ausweichbewegungen mit den Bewegungsvorgaben für den Endeffektor gibt es verschiedene Möglichkeiten, die sich in die hier vorgeschlagenen Bewegungsklassen einteilen lassen.

Weiterführende Arbeiten beschäftigen sich mit der Erweiterung der Funktion des Sicherheitsbereichs. Neben der Eindringtiefe eines Hindernisses, können auch die Eindringgeschwindigkeit in den Sicherheitsbereich und die Verweildauer im Sicherheitsbereich berücksichtigt werden. Schließlich werden die hier vorgestellten Systeme in die bereits am Roboter implementierten planenden Komponenten integriert, um deren lokalen reaktiven Handlungsspielraum zur Kollisionsvermeidung und Interaktion auszunutzen.

Die diesem Bericht zugrundeliegenden Arbeiten wurden teilweise mit Mitteln des Bundesministeriums für Bildung und Forschung unter den Förderkennzeichen *01IN601A2* und *01IL902DO* gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

## Literatur

1. G. Lawitzky, “A Navigation System for Service Robots - From Research to Products,” in *FSR 2001* (A. Halme, R. Chatila, und E. Prassler, Hrsg.), S. 15–20, Yleisjäljennös–Painopörssi, 2001.
2. L. E. Kavraki, P. Svetka, J. C. Latombe, und M. Overmars, “Probabilistic roadmaps for path planning in high dimensional configuration spaces,” in *IEEE Transactions on Robotics and Automation* *12(4)*, S. 566–580, 1996.
3. N. M. Amato, O. B. Bayazit, L. K. Dale, C. Jones, und D. Vallejo, “Choosing good distance metrics and local planners for probabilistic roadmap methods,” in *International Conference on Robotics and Automation*, (Leuven Belgium), 1998.
4. M. C. Lin und J. F. Canny, “A Fast Algorithm for Incremental Distance Calculation,” in *International Conference on Robotics and Automation*, S. 1008–1014, April 1991.
5. E. Larsen, S. Gottschalk, M. C. Lin, und D. Manocha, “Fast Distance Queries with Rectangular Swept Shere Volumes,” in *International Conference on Robotics and Automation*, (San Francisco), April 2000.
6. O. Brock und O. Khatib, “Elastic Strips: A Framework for Integrated Planning and Execution,” in *Preprints, 6th International Symposium on Experimental Robotics (ISER'99)*, (Sydney), 1999.
7. G. v. Wichert, T. Wösch, S. Gutmann, und G. Lawitzky, “MobMan – Ein mobiler Manipulator für Alltagsumgebungen,” in *Autonome Mobile Systeme (AMS'00)* (R. Dillmann, H. Wörn, und M. v. Ehr, Hrsg.), Informatik aktuell, S. 55–62, Springer Verlag, Heidelberg, 2000.
8. G. Lawitzky, “Das Navigationssystem SINAS,” in *Proceedings Robotik 2000, VDI-Berichte 1582*, (Düsseldorf), S. 77–82, VDI-Verlag, June 2000.
9. M. W. Spong und M. Vidyasagar, *Robot Dynamics and Control*. John Wiley & Sons Inc., 1989.