14. Fachgespräch Autonome Mobile Systeme (AMS '98), Karlsruhe, Germany,
30. Nov. - 1. Dez. 1998, © Springer-Verlag, Reihe "Informatik Aktuell", pp 180-187.

Ein verhaltensorientierter Ansatz zum flächendeckenden Fahren in a priori unbekannter Umgebung

Michael Kasper, Dirk Müller, Ewald von Puttkamer

Universität Kaiserslautern • Fachbereich Informatik
AG Robotik und Prozeßrechentechnik • Postfach 3049 • D-67653 Kaiserslautern
kasper@informatik.uni-kl.de • http://ag-vp-www.informatik.uni-kl.de

Kurzfassung

In diesem Aufsatz wird ein Verfahren zum flächendeckenden Fahren in zunächst unbekannter Umgebung beschrieben, wie es z.B. für Reinigungsanwendungen im Heimbereich benötigt wird. Parallel zur Durchführung der Reinigungsaufgabe wird dabei die Umgebung exploriert und kartiert.

Der verhaltensorientierte Ansatz ermöglicht eine robuste, zielgerichtete und dennoch ressourcenschonende Implementierung und gestattet es, einzelne Verhaltensweisen leicht durch verbesserte oder auch speziell erlernte Versionen auszutauschen. Das vorgestellte Verfahren wurde simulativ getestet und wird in Kürze auf einem realen Roboter erprobt.

1. Einleitung

Servicerobotern wird als "Leitprodukten der Zukunft" in den kommenden Jahren ein großes Wachstumspotential vorausgesagt [Schraft und Schmierer 98]. Einen relativ hohen Anteil an diesen auf den Dienstleistungssektor spezialisierten Robotern werden dabei verschiedenste Reinigungsmaschinen einnehmen. Neben spezialisierten Robotern zur Fassaden- oder Fahrzeugreinigung kommt hier der Bodenreinigung eine Schlüsselrolle zu. Während es bereits seit einigen Jahren (teil-)autonome Bodenreinigungsroboter für den kommerziellen Einsatz auf Flughäfen, Bahnhöfen oder in Supermärkten gibt (z.B. [Hako 94], [Kent 98], [Lawitzky et al. 98]), existieren für den Heimbereich bisher nur Prototypen, wie die Roboter der Firmen Electrolux oder Minolta.

Der vorliegende Aufsatz konzentriert sich auf die Steuerungskomponente eines solchen autonomen Bodenreinigungsroboters für den Privatbereich. Nach der Diskussion der Anforderungen an diese Systeme wird ein verhaltensorientierter Ansatz vorgestellt, der diesen entgegenkommt. Die ersten experimentellen Ergebnisse werden präsentiert und eine Zusammenfassung, verbunden mit einem Ausblick auf das in der Entwicklung befindliche reale System, beschließt den Beitrag.

2. Anforderungen und Voraussetzungen

Autonomous Home Cleaning Robots müssen leicht und klein genug sein, um zwischen Stuhl- und Tischbeinen reinigen zu können. Sie benötigen eine robuste und feh-

lertolerante Steuerung, die auch mit dynamischen Objekten umgehen kann und den Roboter zielgerichtet und effizient steuert. - Der Prototyp der Firma Electrolux [Electrolux 97] arbeitet z.B. rein indeterministisch und somit nicht zielgerichtet.

Da für die sich häufig ändernden Einsatzumgebungen (Veränderung innerhalb einer Umgebung, Einsatz in verschiedenen Umgebungen) der Aufwand einer a priori Kartenerstellung nicht gerechtfertigt erscheint, sollte der Roboter parallel zur Reinigungsaufgabe seine Umgebung explorieren und kartieren. Es sind nur wenige Ansätze bekannt, die dies berücksichtigen (z.B. [Burhanpukar 94] oder [Furuhashi et al. 92]), wobei aber entweder die Algorithmen nicht veröffentlicht sind, oder der Nachweis der praktischen Einsetzbarkeit fehlt.

Schließlich sollten autonome Reinigungsroboter sowohl einfach in der Handhabung als auch preiswert sein. Letzteres stellt aus heutiger Sicht neben der begrenzten Batterietechnologie die größte Einschränkung dar. Die technologischen Anforderungen lassen sich mit entsprechender Ausstattung prinzipiell lösen. Durch die finanziellen Restriktionen, die noch stärker als im industriellen Bereich greifen, und den daraus resultierenden Beschränkungen in der Sensorik und der Rechenleistung scheiden jedoch viele klassische Lösungsansätze von vornherein aus.

Auf der anderen Seite kann die Realisierung durch einige Annahmen vereinfacht werden, die sich auf ein einzelnes reales System im Heimbereich übertragen lassen: Um die Navigation zu erleichtern, kann ein runder Roboter mit der Fähigkeit zu Punktdrehungen vorausgesetzt werden. Ferner werden in der Einsatzumgebung keine "aggressiven" dynamischen Hindernisse auftreten, bei denen ein einfaches Ausweichen oder Stehenbleiben des Roboters zur Kollisionsvermeidung nicht ausreichen würde.

Eine weitergehende Vereinfachung ergibt sich aus der Annahme, daß ein Localizer-Modul existiert, welches aus den Sensordaten (ggf. eines speziellen Sensors) die aktuelle Position des Roboters relativ zur Startposition bestimmt. Arbeiten wie [Weiß et al. 94], [Lu und Milios 97] oder [Thrun et al. 98] zeigen Möglichkeiten auf, wie z.B. durch Korrelation von Sensordaten die Roboterposition ausreichend genau bestimmt werden kann.

Eine Alternative stellen spezielle Sensorsysteme zur Positionsbestimmung dar, die sich auf künstliche Landmarken stützen. Sofern diese Systeme den finanziellen Voraussetzungen genügen und mit nur geringen Umweltveränderungen auskommen, erscheint ihr Einsatz durchaus gerechtfertigt. Ein entsprechendes System wird derzeit in der Arbeitsgruppe der Autoren entwickelt, soll hier jedoch nicht näher diskutiert werden.

3. Systemstruktur

Im folgenden wird eine verallgemeinerte verhaltensorientierte Architektur vorgestellt, welche in ihren Wurzeln auf Arbeiten von Brooks zurückgeht [Brooks 86]. Durch ihren modularen Aufbau, die kurzen Reaktionszeiten und die hohe Fehlertoleranz sowie durch den relativ geringen Ressourcenbedarf wird die verhaltensorientierte Kontrollstruktur den genannten Anforderungen in idealer Weise gerecht. Sie bietet zudem den Vorteil, daß die einzelnen Module weitgehend unabhängig voneinander implementiert und getestet sowie bei Bedarf zu einem späteren Zeitpunkt durch verbesserte Versionen ersetzt werden können.

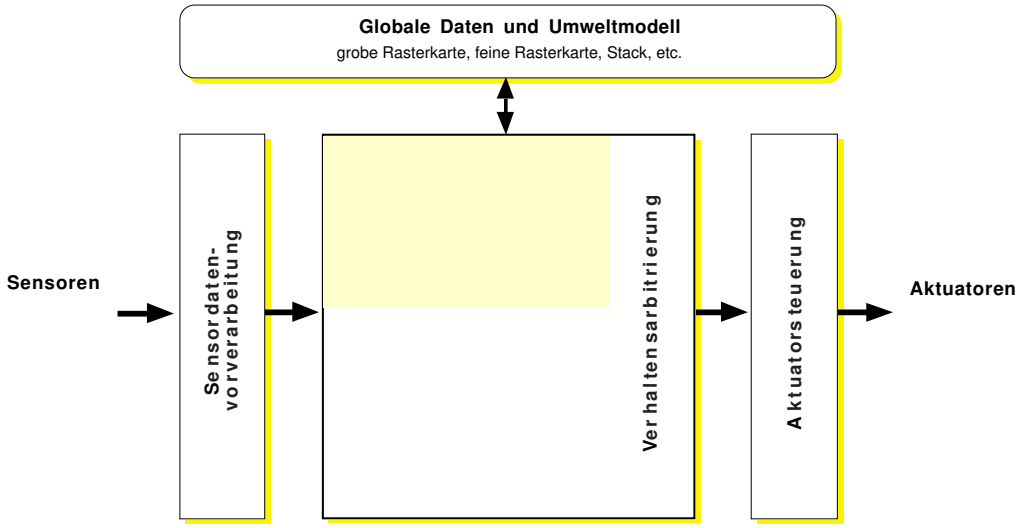

Abbildung 1: verallgemeinerte verhaltensorientierte Kontrollstruktur

Im Gegensatz zu der von Brooks vorgeschlagenen *subsumption architecture* kommt bei dieser Arbeit jedoch eine allgemeinere Arbitrierungseinheit (vgl. [Rosenblatt 97], [Arkin 98]) zur Verhaltensauswahl zum Einsatz. Darüber hinaus werden globale Daten und Zustände zugelassen, die Voraussetzung für die Umweltmodellierung sind.

Neben konkurrierenden, auf verschiedene Aufgaben spezialisierten *aktuatorischen Verhaltensschichten* (ähnlich den Bewegungsmakros in [Hofner 97]) wurden verdeckte, im Hintergrund laufende *background behaviors* implementiert, welche lediglich globale Daten verändern bzw. die Arbitrierung beeinflussen. Das Gesamtverhalten des Roboters wird durch das emergente Zusammenspiel der einzelnen Verhaltensweisen bestimmt. Abbildung 1 zeigt die generelle Systemsstruktur.

Die Verhaltensmodule greifen im wesentlichen auf drei Datenstrukturen zurück: eine hochauflösende Rasterkarte zur Umweltmodellierung (Repräsentation von freien, bereits abgearbeiteten bzw. von Hindernissen besetzten Flächenstücken), eine grobe Rasterkarte zur Navigation (s.u.) sowie auf einen Stack, auf dem noch abzufahrende Anfangsstrecken abgelegt werden. Die flächendeckende Fahrweise ergibt sich durch sukzessives Abarbeiten der auf dem Stack befindlichen Streckensegmente, was letztlich einer backtracking-Strategie entspricht.

Der zugrundeliegende Algorithmus läßt sich folgendermaßen skizzieren:

1. Fahre geradeaus bzw. bündig entlang der eigenen Spur bis zum Auftreffen auf ein Hindernis. (Unter einer *Spur* wird eine vom Roboter bereits abgefahrene Bahn verstanden.) Beobachte dabei die benachbarten Bahnen und speichere freie, noch nicht befahrene Streckenabschnitte auf einem Stack ab.
2. Solange der Stack noch Einträge enthält, fahre zu der entsprechenden Position und beginne von vorne.
3. Ende

Dieser Algorithmus wird durch die einzelnen, verdeckten und aktuatorischen Verhaltensweisen realisiert.