



Title

Subtitle

SubSubtitle

SS 2018 - WS 2018/2019

Authors

Contents

1	Geschichte	1
1.1	Industrieroboter	1
1.2	Serviceroboter	1
1.2.1	Definition	1
1.2.2	Klassen	2
2	Software-Architekturen für mobile Robotersysteme	3
2.1	Probleme und Anforderungen	3
2.1.1	Definition mobile Roboter	3
2.1.2	Umgebung mobiler Roboter	3
2.1.3	Roboterkontroll-Architekturen	4
2.1.4	Anforderungen an das Kontrollsystem eines autonomen Roboters . . .	4
2.2	Mögliche Modelle	5
2.2.1	Klassisches Modell - der funktionale Ansatz	5
	List of Figures	7
	Listings	8

1 Geschichte

1.1 Industrieroboter

Nach Definition der VDI-Richtlinie 2860 sind Industrieroboter universell einsetzbare Bewegungsautomaten mit mehreren Achsen, deren Bewegungen hinsichtlich Bewegungsfolge und Wegen bzw. Winkel frei programmierbar und sensorgeführt sind.

- Zeichnen sich durch **Schnelligkeit**, **Genauigkeit**, **Robustheit** und eine hohe **Traglast** aus.
- Einsatzgebiete: Schweißen, Kleben, Schneide, Lackieren

Zunehmend **kollaborative** Roboter, Cobots:

- Industrieroboter, die mit Menschen gemeinsam arbeiten
- Nicht mehr durch Schutzeinrichtungen im Produktionsprozess von Menschen getrennt
- Nimmt Menschen wahr, verursacht keine Verletzungen

1.2 Serviceroboter

1.2.1 Definition

- Ein **Serviceroboter** ist eine **frei programmierbare Bewegungseinrichtung**, die **teil- oder vollautomatisch** Dienstleistungen verrichtet.
- **Dienstleistungen** sind dabei Tätigkeiten, die nicht der direkten industriellen ERzeugung von Sachgütern, sondern der Verrichtung von **Leistungen für Menschen und Einrichtungen** dienen.
- Einteilung in zwei Klassen

1.2.2 Klassen

- Roboter, die für professionellen Einsatzbereich: **Rettung, Landwirtschaft, Medizin**
- Roboter für den Privaten gebrauch: **Staubsauger, Rasenmäher, Pfleger**

2 Software-Architekturen für mobile Robotersysteme

2.1 Probleme und Anforderungen

2.1.1 Definition mobile Roboter

'Unter einem Roboter verstehen wir eine frei programmierbare Maschine, die auf Basis von Umgebungssensordaten in geschlossener Regelung in Umgebungen agiert, die zur Zeit der Programmierung nicht genau bekannt und/oder dynamisch und oder nicht vollständig erfassbar sind.' ⇒ **Joachim Herzberg, Mobile Roboter**

2.1.2 Umgebung mobiler Roboter

Bei **mobilen Robotern** ist die Umgebung im Detail **nicht bekannt und generell nicht kontrollierbar**

- Alle Aktionen sind von der aktuellen Umgebung abhängig
- Details sind erst zum Zeitpunkt der Ausführung der Aktionen bekannt
- Mobile Roboter müssen in einer geschlossenen Regelung
 - die Umgebung mit Sensoren erfassen
 - die Daten auswerten
 - Aktionen daraus planen
 - Aktionen mittels Koordination der Aktuatoren umsetzen

2.1.3 Roboterkontroll-Architekturen

Herausforderungen

- Robotersystem besteht aus den Gebieten **Wahrnehmung**, **Planung** und **Handlung**
- Herausforderungen an eine Roboterkontroll-Architektur, sie muss:
 - Sensorwerte erfassen und auswerten
 - Pfade planen
 - Hindernisse vermeiden
 - Komplexe Algorithmen in langen Zeitzyklen ausführen

Probleme bei der Software-Erstellung zur Roboterkontrolle

- Roboter sind eingebettete Systeme, die in geschlossener Regelung laufen und die Sensorströme in **Echtzeit verarbeiten** müssen
- Unterschiedliche Aufgaben → Unterschiedliche Zeitzyklen
- Unterschiedlicher Zeitskalen → kein standardisierter Kontroll- oder Datenfluss den die Architektur abbilden könnten
- Für etliche algorithmische Teilprobleme sind **keine effizienten Verfahren** bekannt
- **Prozessorkapazität ist begrenzt**

2.1.4 Anforderungen an das Kontrollsystem eines autonomen Roboters

Robustheit

- Die Umgebung des Systems kann sich ständig ändern
- Auf eine Umgebungsänderung sollte der Roboter sinnvoll reagieren und nicht verwirrt stehen bleiben.
- Verwendete Modelle der Umgebung sind ungenau.

2.2 Mögliche Modelle

Unterschiedliche Ziele

- Der Roboter verfolgt zu einem Zeitpunkt eventuell Ziele, die im Konflikt zueinander stehen.
- **Beispiel:** der Roboter soll ein bestimmtes Ziel ansteuern, dabei aber Hindernissen ausweichen.

Sensorwerte von mehreren Sensoren

- Sensordaten können verrauscht sein
- Sensoren können fehlerhafte oder inkonsistente Messwerte liefern, weil der Sensor z.B. außerhalb seines Bereichs misst für den er zuständig ist und dies nicht überprüfen kann.

Erweiterbarkeit

- Wenn der Roboter neue Sensoren erhält, sollte dies leicht in das Programm integriert werden können.

2.2 Mögliche Modelle

2.2.1 Klassisches Modell - der funktionale Ansatz

Das **klassische Modell** wird auch als hierarchisches Model oder funktionales Model bezeichnet. Ist ein Top-Down Ansatz, besteht aus drei Abstraktionsebenen

- Die unterste Ebene: **Pilot**
- Mittlere Eben: **Navigator**
- Oberste Ebene: **Planer**

Sense-Think-Act-Cycle oder **SMPA** (Sense - Model - Plan - Act).

- Sensordaten, die vom Fahrzeug geliefert werden, werden in den zwei unteren Ebenen vorverarbeitet.
- Konstruktion oder Aktualisierung eines Weltmodells
- **Planer** ist die Basis aller Entscheidungen basieren auf dem zugrundeliegenden Weltmodell

2.2 Mögliche Modelle

- Tatsächliche Fahrbefehle werden durch unterste Ebene ausgeführt

Zyklus wird ständig wiederholt \Rightarrow wenn alle Ebenen richtig funktionieren resultiert daraus ein intelligentes Verhalten und die Erfüllung der Aufgabe.

List of Figures

Listings