

# Zusammenfassung

Die vorliegende Arbeit befasst sich mit Lernverfahren für autonome mobile Roboter. Die Dissertation entstand im Laufe meiner Mitarbeit am RoboCup Projekt der Freien Universität Berlin.

Das Hauptziel war es, das Verhalten von Robotern zu verbessern sowie den Entwicklungsaufwand zu minimieren. Dies wurde durch Lernen auf unterschiedlichen Komplexitäts- und Abstraktions-Ebenen des Roboters erreicht. Von der schnellsten Steuerungsebene auf der Roboterelektronik bis zu einer automatisch gelernten Simulation des Roboters reicht das Spektrum, in dem die Lernmethoden angewendet wurden. Ich fasse die von mir bearbeiteten Probleme und deren Lösungen kurz zusammen:

1. Eine der wichtigsten Aufgaben war das Lernen der zukünftigen Positionen und Orientierungen des Roboters. Diese Vorhersage der Roboterdynamik ist essentiell, um die Totzeit zwischen der Bildaufnahme des Spielfeldes und der tatsächlichen Ausführung der Befehle zu kompensieren. Die Verzögerung beträgt etwa 130 ms und entspricht bei maximaler Geschwindigkeit einer Bewegung von 20 cm. Die effektivste Möglichkeit, das Delay zu überbrücken, ist die vorhergesagten Zustände des Roboters statt den aktuellen Werten der Bildverarbeitung für die Verhaltenssteuerung zu nutzen.

Ich habe unterschiedliche Methoden zum Lernen der Vorhersage implementiert und verglichen. Als besonders geeignet für die Schätzung der Zukunft haben sich ARMA und ein neuronales Netz, das mit dem Backpropagation-Algorithmus trainiert wurde, erwiesen. Außerdem habe ich das Delay des Gesamtsystems mit verschiedenen Methoden und bei unterschiedlichen Voraussetzungen gemessen und die Anteile, die bestimmte Komponenten dazu beitragen, abgeschätzt.

2. Ein Simulationssystem ist notwendig, um neue Verhalten schnell und ohne Hardwareaufwand testen zu können. Auf diese Weise können mehrere Personen gleichzeitig am System arbeiten und das Verhalten verbessern.

Basierend auf der gelernten Roboterdynamik habe ich das vorhandene Simulationssystem der Roboter erheblich verbessert und verallgemeinert.

Konventionelle Simulationsmethoden erfordern ein genaues physikalisches Modell für die simulierten Roboter. Diese Modelle bei ständig weiterentwickelten Robotern anzupassen ist zeitaufwändig und fehleranfällig. Mit dem lernbaren Simulationssystem ist eine manuelle Anpassung der Fahrphysik nicht mehr notwendig.

3. Besonders beim Beschleunigen des Roboters ist der Fehler zwischen gewünschter und tatsächlicher Fahrtrichtung beträchtlich. Der Grund ist einerseits ein sehr einfaches Modell für die Umsetzung der Richtungsgeschwindigkeiten des Roboters in Steuerwerte für die Motoren und andererseits das Regelungssystem auf dem Roboter selbst.

Um diesen Fehler zu kompensieren, habe ich ein Verfahren ausgearbeitet, das durch den Vergleich des gewünschten und des beobachteten Roboterhaltens eine Korrektur der Fahrbefehle an den Roboter lernt.

Die Berichtigung findet zwischen der Berechnung der Fahrbefehle und dem Senden derselben zu dem Roboter statt. Dies hat den Vorteil, dass die Korrektur für das restliche System transparent geschieht und für jeden Robotertyp unabhängig erfolgen kann. Zur Optimierung habe ich einerseits eine Tabelle benutzt, die zu jeder gewünschten Fahrtrichtung einen Fahrbefehl enthält und andererseits eine Methode entwickelt, die den bestmöglichen Fahrbefehl anhand der Auswirkungen auf den momentanen Zustand des Roboters aller möglichen Fahrbefehle ermittelt.

4. Ein schwerwiegendes Problem war das Ausfallen von Motoren während wichtiger Spiele. Ich habe deshalb im Rahmen der Dissertation Mechanismen zur automatischen Erkennung und Kompensation eines Motordefekts entwickelt. Unter Verwendung verschiedener Robotertypen wurde gelernt, wie ein Roboter mit ausgefallenen Motoren reagiert und welche Befehle erforderlich sind, um adäquat zu fahren.
5. Der Regelkreis der verwendeten Roboter besteht aus drei PID-Controllern. Die manuelle Anpassung der insgesamt 18 Parameter ist eine langwierige Aufgabe. Die Konfiguration ist außerdem sehr subjektiv und damit nicht optimal.

Ich habe ein Verfahren konzipiert, das die optimalen Parameter der PID-Controller lernt. Die Fahrweise der Roboter wurde damit verbessert, das Durchdrehen der Räder vermindert. Das Lernprozess ist auch auf andere Parameterprobleme leicht übertragbar. Die verwendete Methode betrachtet die Parameter als völlig unabhängig voneinander und variiert ganze Sätze von Parametern gleichzeitig, um auf diese Weise schnell eine optimalere Kombination zu finden.

6. Das letzte in der vorliegenden Arbeit vorgestellte Lernverfahren beschäftigt sich mit dem Fahren des Roboters zu einer Zielposition. Das genaue Fahren des Roboters hängt unter anderem davon ab, wie schnell und exakt er bremsen kann. Eine nicht optimal eingestellte Bremskurve

führt zu einer unangemessen langsamen Fahrt des Roboters, zu einer ungenauen Positionierung oder zur Oszillation. Die Stützstellen der Bremskurve habe ich mit unterschiedlichen Lernverfahren so optimiert, dass das Ergebnis eine schnelle und exakte Fahrweise ist. Ich habe Monte-Carlo-Methoden und generische Algorithmen zur Modifikation der Bremskurve verwendet.

Die Intention dieser Arbeit war, Lernen in allen Verhaltensebenen eines autonomen, mobilen Roboters anzuwenden, um unterschiedliche Probleme zu lösen, die mit konventionellen Verfahren nur mit erheblich mehr Aufwand oder überhaupt nicht zu bewältigen sind. Ich habe exemplarisch wichtige Komponenten des FU-Fighters-Frameworks erweitert und gezeigt, dass Lernen in allen Gebieten der Robotik anwendbar ist und wesentlich zu einer schnellen, automatisierten Weiterentwicklung beiträgt. Lernen mit realen Robotern ist ein relativ junges Gebiet, das über großes Forschungs- und Anwendungs-Potenzial verfügt.

