FACHHOCHSCHULE BRANDENBURG
FACHBEREICH INFORMATIK UND MEDIEN

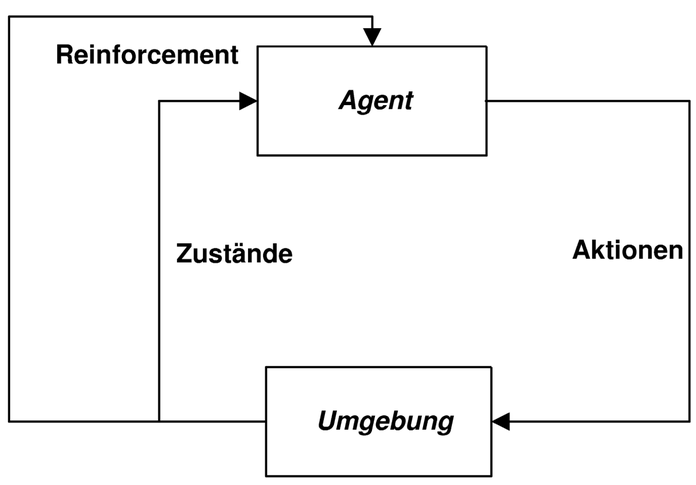
Reinforcement Lernen in Mobilen Systemen

Diplomarbeit, vorgelegt von René Eggert

Aufgabenstellung:

Zielstellung des Themas ist die Evaluierung des Reinforcement-Lernens (speziell Q-Lernen) zur Anwendung in mobilen Systemen.

Hierbei sollen die verschiedenen Arten der Methode theoretisch und experimentell verglichen und beispielhaft im Simulator sowie im realen Roboter umgesetzt werden. Die Anwendungsdomäne sei dabei das Erlernen eines Verhaltens zur Hindernisvermeidung mittels Sonarsensoren. Die Implementierung soll besonderen Wert auf die Wiedernutzbarkeit legen.

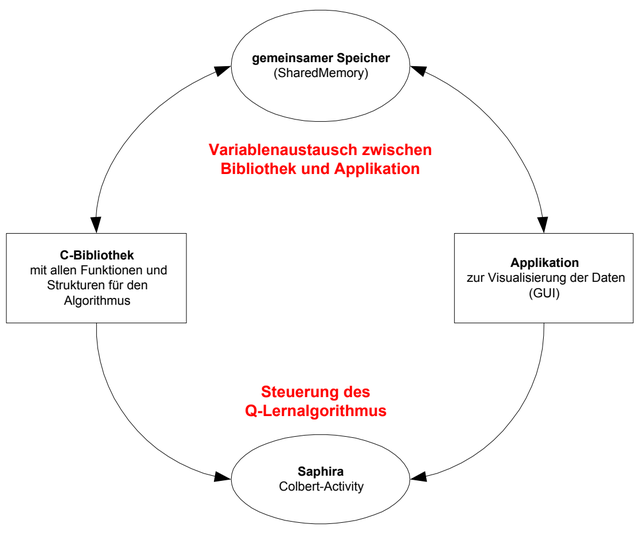


Reinforcement Lernen:

Lernen durch Reinforcement (Verstärkendes Lernen) ist eine mögliche Art, Autonome Roboter so zu programmieren, dass sie mit Hilfe von Sensoren auf wechselnde Umwelteinflüsse reagieren.

Die Steuerung des Agenten erfolgt durch Aktionen. Die Anwendung von solchen Aktionen führt den Agenten in einen Nachfolgezustand. Zustände dienen zur Beschreibung der Umgebung. Das System erhält eine Bewertung des Zustands in Form des Reinforcementsignals. Der Agent erhält entweder eine Belohnung für einen guten Zustand oder eine Strafe für einen entsprechend schlechten Zustand.

Systemarchitektur:



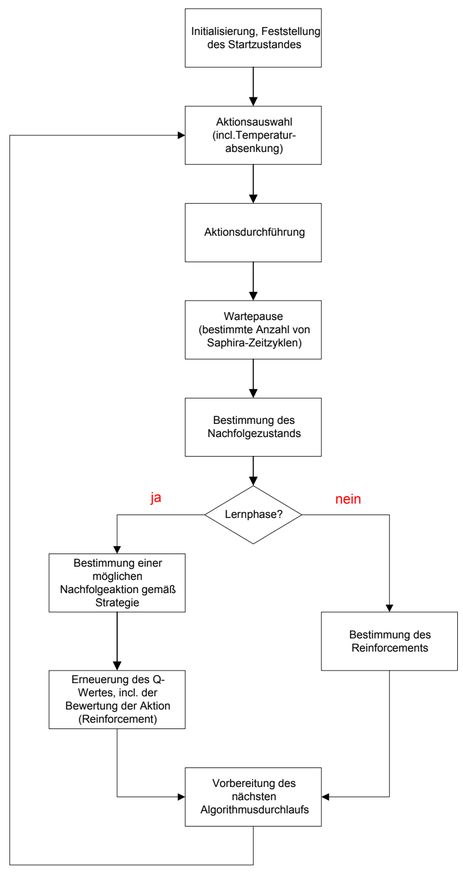
Das in der Saphira-Oberfläche geladene Colbert-Activity steuert die einzelnen Schritte des Q-Algorithmus durch Aufruf der Funktionen.

Das Colbert-Activity lädt zu Beginn die C-Bibliothek, um alle Funktionen und Strukturen dem Algorithmus bekannt zu machen. Die Applikation zeigt nach dem Start die jeweils aktuellen Werte des Algorithmus. Durch die Benutzung von Schaltflächen besitzt der Anwender Einfluss auf den Algorithmus.

Der Variablenaustausch zwischen den Strukturen und Variablen der C-Bibliothek und den Anzeigedialogen der Applikation erfolgt über den gemeinsamen Speicher.

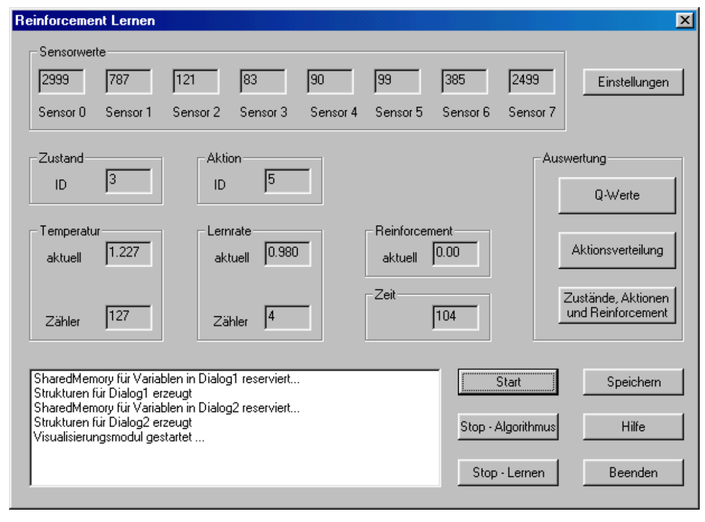
Q-Lernalgorithmus:

Nach der einmaligen Feststellung des Initialzustandes laufen alle Schritte des Q-Algorithmus in Form einer Schleife ab. Im GUI kann die Lernphase abgeschaltet werden. Durch das Beenden der Lernphase werden die Q-Werte nicht mehr aktualisiert. Die Nachfolgeaktionen in den einzelnen Zuständen stehen fest.



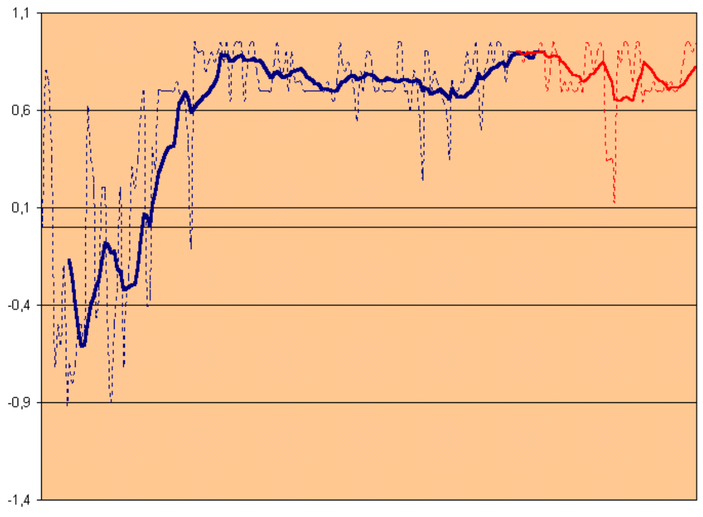
schematisierter Ablauf des Q-Lernalgorithmus

Visualisierung der Daten:



Zur Visualisierung der Daten des Algorithmus dient ein Graphic User Interface (GUI), welches die verschiedenen Parameter anzeigt und verändern kann.

Neben dem Hauptdialog zur Anzeige der wichtigsten Lernwerte und zur Steuerung des Algorithmus gibt es einen Dialog für die Einstellungen der Parameter, sowie Dialoge zur Auswertung des Algorithmus. Der Einstellungsdialog lässt die Änderung von Parametern zu, um den Lernalgorithmus erneut ablaufen zu lassen. Die Dialoge Aktionswahrscheinlichkeit, Q-Werte und Zustände, Aktionen und Reinforcement zur Anzeige von auswertungsrelevanten Daten vervollständigen die Benutzeroberfläche.

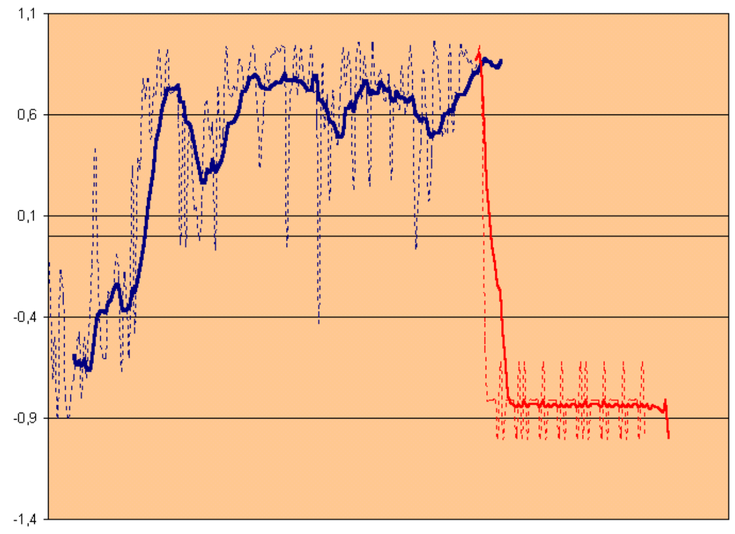


Beispiel für eine positive Entwicklung des Reinforcements über die Zeit (Standardparameter)
(blau-Lernphase,rot-Bewertung nach der Lernphase)

Auswertung:

Der Roboter erfüllt die Aufgabe der Hindernisvermeidung durch Reinforcement Lernen mittels dem Lernverfahren Q-Lernen. Nach einiger Zeit wird der Kontakt mit Hindernissen vermieden. Die Kurve der Lernergebnisse zeigt bei der Auswahl der passenden Parameter stetig nach oben. Das gelernte Verhalten kann anschließend in der Umgebung eingesetzt werden.

Fazit:
Das Q-Lernen ist für die Hindernisvermeidung gut geeignet. Die Wahl der Parameter stellt ein wichtiges Kriterium dar, um für die Umgebung auch die passende Navigation bereitzustellen. Durch Einsatz anderer Lernverfahren können die erzielten Lernergebnisse u.U. noch verbessert und die Ansätze der Diplomarbeit weiterverfolgt werden.



Beispiel für eine negative Entwicklung des Reinforcements über die Zeit nach Ende der Lernphase (Parameter mit hohen Geschwindigkeiten)

Betreuer: Prof. Dr. J. Heinsohn
Dipl.-Inform. I. Boersch

Abgabe: 10.01.2002