Technische Hochschule
Brandenburg
University of
Applied Sciences
Fachbereich
Informatik und Medien

Real-time reinforcement learning von Handlungsstrategien für humanoide Roboter

Colin Christ

Masterarbeit • Studiengang Informatik • Fachbereich Informatik und Medien • 02.02.2018

Aufgabenstellung

Ziel ist es, ein Programm zur Demonstration von Reinforcement Learning (RL) mit humanoiden Robotern zu entwickeln. Das Lernen soll mit verschiedenen RL-Algorithmen ermöglicht werden, die zur Programmlaufzeit einem Lerner zugeordnet und auch miteinander kombiniert werden können. Über eine grafische Benutzeroberfläche soll das Lernen visualisiert werden, siehe Abb. 2 und Abb. 3. In einer vorangegangenen Arbeit wurde das Szenario durch Q-Learning mit Q-Tabelle und Ɛ-greedy-Exploration gelöst. In dieser Arbeit soll durch Anwendung anderer RL-Algorithmen ein Ansatz gefunden werden, das Lernen zu beschleunigen, bzw. schneller eine gute Handlungsstrategie zu finden.

Szenario

Der Versuch wird mit Naos durchgeführt. Naos sind ca. 40 cm große, humanoide Roboter. Der Versuchsaufbau ist in Abb. 1 zu sehen. Der rote Nao nimmt die Rolle des Lerners, bzw. Agenten, ein. Er soll lernen, die roten Bälle von der mittleren Schiene auf die rechte Schiene und die blauen Bälle auf die linke Schiene zu sortieren. Er kann immer nur den ersten Ball auf der mittleren Schiene nutzen. Die Aufgabe des blauen Naos ist es, die Bälle von den äußeren Schienen wieder in auf die mittlere Schiene zurück zu sortieren und zu erkennen, ob der Agent die Bälle richtig einsortiert.
Der Agent befindet sich zu jedem Zeitpunkt in einem Zustand, der sich aus der Stellung der Hände, den Positionen der Arme und der Farbe des ersten Balls auf der mittleren Schiene zusammensetzt. In jedem Iterationsschritt hat der Agent mehrere zustandsabhängige vordefinierte Aktionen, von denen er eine, entsprechend seiner aktuellen Handlungsstrategie, auch Policy genannt, ausführt. Die Aktionen erlauben dem Agenten, verschiedene festgelegte Koordinaten im Raum mit den Armen anzufahren, die Hände zu öffnen oder zu schließen oder die Ballfarbe eines Balls zu erkennen.

Verwendete Algorithmen

In der Arbeit wurde RL in verschiedene Bereiche unterteilt:

- Temporal Difference Learning: Q-Learning und Q(λ)
- Exploration: Ɛ-greedy und Softmax
- Estimator: Q-Tabelle und Tile Coding
- Planer: Prioritized Sweeping

Der Agent kann jeweils ein Modul aus jedem Bereich zum Lernen benutzen.

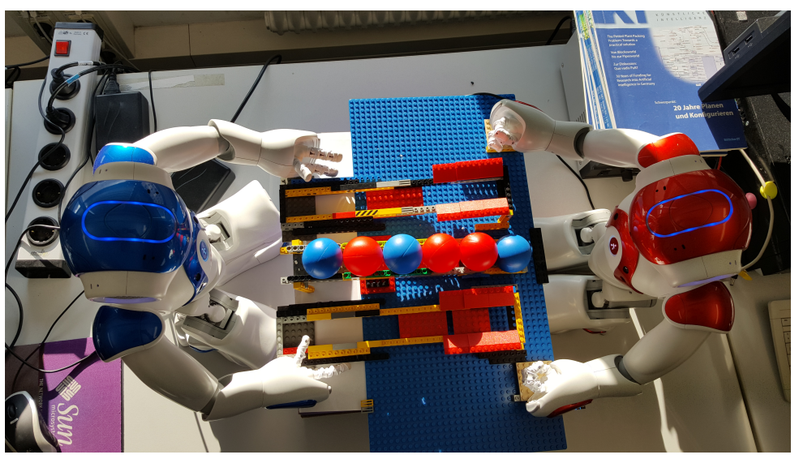

Abb. 1: Versuchsaufbau aus der Vogelperspektive

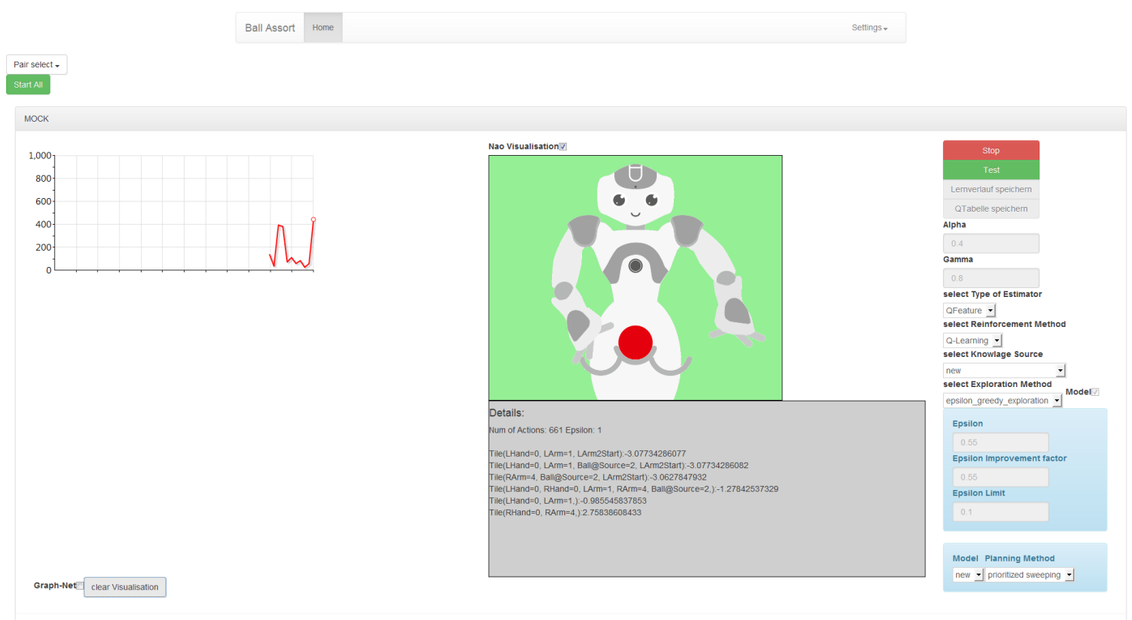

Abb. 2: Grafische Benutzeroberfläche, Visualisierungen des Lernens.

Versuche

Fünf Versuchsreihen mit jeweils drei bis fünf unterschiedlichen Lernkonfigurationen und Parametern wurden in Echtzeit durchgeführt um die beste Konfiguration zu ermitteln. Da die Motoren der Naos nach ca. fünf Minuten überhitzen, wurden die Versuche in einer Simulation durchgeführt, in der eine Aktionsausführung mindestens eine Sekunde dauerte.

Ergebnisse

Q-Learning mit einer Q-Tabelle, Ɛ-greedy-Exploration und Prioritized Sweeping mit α und ɣ mit 0,9 erwies sich in den Versuchen als die beste Konfiguration. Eine gute Policy wurde in ca. 40 Minuten gefunden. Abb. 3 zeigt die ermittelte Handlungsstrategie zu Einsortierung eines blauen Balls.

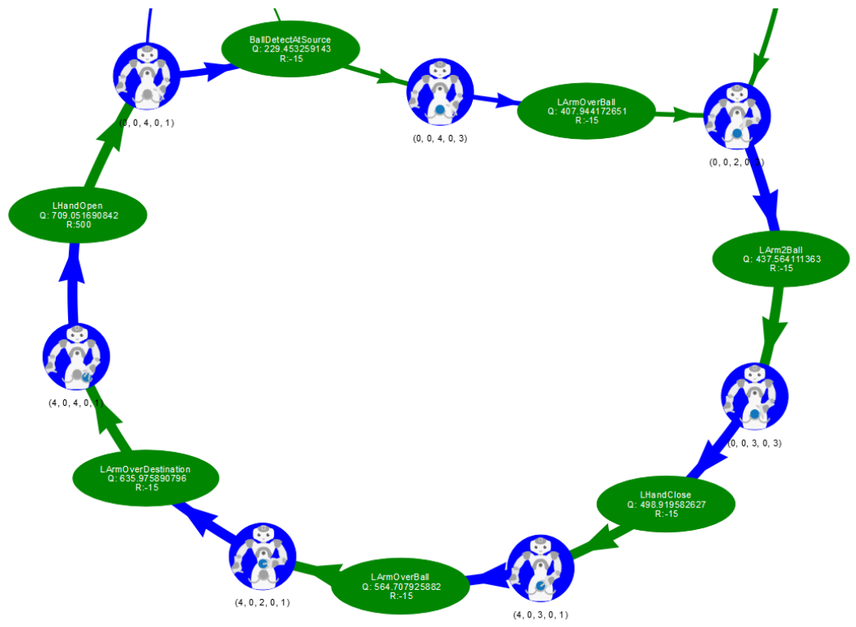

Abb. 3: Verhaltensgraph des Agenten zur Einsortierung eines blauen Balls.

Fazit

In dieser Arbeit wurde ein Programm mit grafischer Benutzeroberfläche implementiert, das es Anwendern ermöglicht, RL-Algorithmen in einem Szenario mit humanoiden Robotern durchzuführen. Versuche, in denen das Programm verwendet wurde, ergaben, dass eine gute Handlungsstrategie in ca. 40 Minuten ermittelt werden kann. Das Projekt soll in Folgearbeiten weitergeführt werden. Das Überhitzungsproblem muss zukünftig gelöst, sowie weitere RL-Algorithmen hinzugefügt werden.

Betreuung Dipl.-Inform. Ingo Boersch und Prof. Dr.-Ing. Jochen Heinsohn • Technische Hochschule Brandenburg