Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Ausarbeitung

Manuel Trittel

Reinforcement Learning in der Modellfahrzeugnavigation

Fakultät Technik und Informatik
Department Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Inhaltsverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

1 Einführung

„Die Entwicklung intelligenter Systeme, die selbständig komplexe Aufgaben lösen, ist ein zentrales Forschungsgebiet in der Informatik. Im Vordergrund steht hierbei stets die Frage, wie das System lernen kann, sich korrekt zu verhalten, so dass die vorgegeben Ziele verwirklicht werden“ (Wolter, 2008, S.1).

Das in Kapitel 2 beschriebene Reinforcement Learning (RL) ist eine typische Lernsituation aus dem Maschinellen Lernen. Ziele werden rein durch Erfahrungen und ohne vorgegebene Lösungswege erreicht. Diese Ausarbeitung beschäftigt sich mit ersten Ansätzen autonome Lernalgorithmen nach den Konzepten des RL im Rahmen der FAUST-Projekte (Fahrerassistenz- und Autonome Systeme) einzusetzen. Für Details zu den FAUST-Projekten sei auf (FAUST, 2008) verwiesen.

Es wird auf autonomen Modellfahrzeugen (siehe Abb. 1.1) gearbeitet, welche unter Anderem am Carolo-Cup Wettbewerb[1] der TU Braunschweig teilnehmen. Die Zielsetzung für die nächsten zwei bis drei Monate ist die Konzeption und Implementierung einer Geschwindigkeitsregelung mit Hilfe des RL. Mit wenigen Probedurchfahrten eines festen Parcours soll das Fahrzeug erlernen, mit welchen Geschwindigkeiten es maximal fahren darf ohne von der Fahrbahn abzukommen.

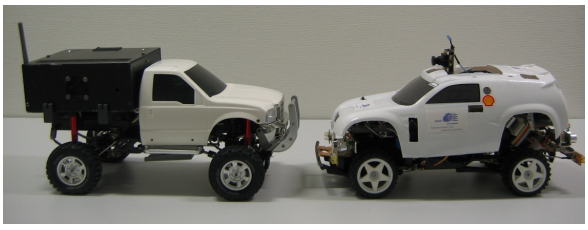

Abbildung 1.1: Autonome Modellfahrzeuge im FAUST-Projekt

Im Hauptteil dieser Arbeit werden einführend theoretische Grundlagen und beispielhafte Algorithmen des Reinforcement Learnings vorgestellt (Kapitel 2). Darauf aufbauend wird das methodische Vorgehen für den konkreten Anwendungsfall beschrieben (Kapitel 3) und damit verbundene Risiken abgeschätzt (Kapitel 4). Abschließend und zusammenfassend wird ein Ausblick auf Optimierungspotentiale und weitere Anwendungsfälle gegeben (Kapitel 5).

[1] http://www.carolo-cup.de/

2 Reinforcement Learning

In diesem Kapitel wird das Reinforcement Learning in das Maschinelle Lernen eingeordnet. Es werden grundlegende Konzepte erklärt und Parameter zusammen gefasst. Anschließend wird eine Klasse von RL Lernmethoden mit zwei Beispielen vorgestellt.

2.1 Grundlagen des RL

Beim Maschinellen Lernen in Bezug auf die Fahrzeugnavigation befindet sich ein Fahrzeug bzw. Agent in einer irgendwie gearteten Umwelt, siehe Abb. 2.1. Diese kann zum Teil bekannt (z.B. durch Kartografie) oder unbekannt sein. Sie kann durch verschiedenste Sensoriken wahrgenommen werden. Daten von Sensoren, aber auch Kamerabilder oder Laserwerte können Eingabeparameter für das Fahrzeug darstellen, auf dessen Grundlage Entscheidungen getroffen und Aktionen ausgeführt werden müssen, um definierte Ziele zu erreichen.

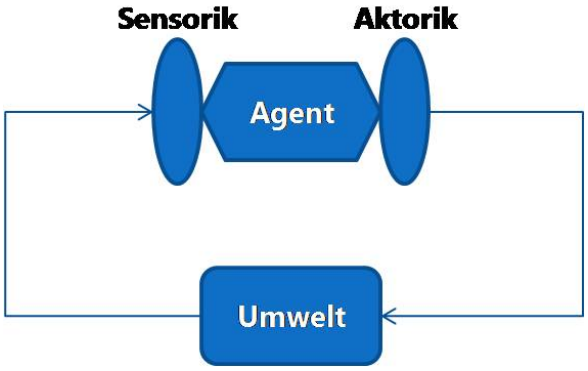

Abbildung 2.1: Agenteninteraktion mit seiner Umwelt

Das Reinforcement Learning ist eine der drei Hauptlernsituationen des Maschinellen Lernens in der sich der Agent befinden kann, siehe Abbildung 2.2.

Beim **Supervised Learning** erhält der Agent Eingabedatensätze, auf deren Basis er eine Entscheidung trifft. Hier gibt es nun sozusagen einen Lehrer, der dem Agenten anschließend die beste Entscheidung mitteilt. Dieser widerum passt intern seine Logik entsprechend

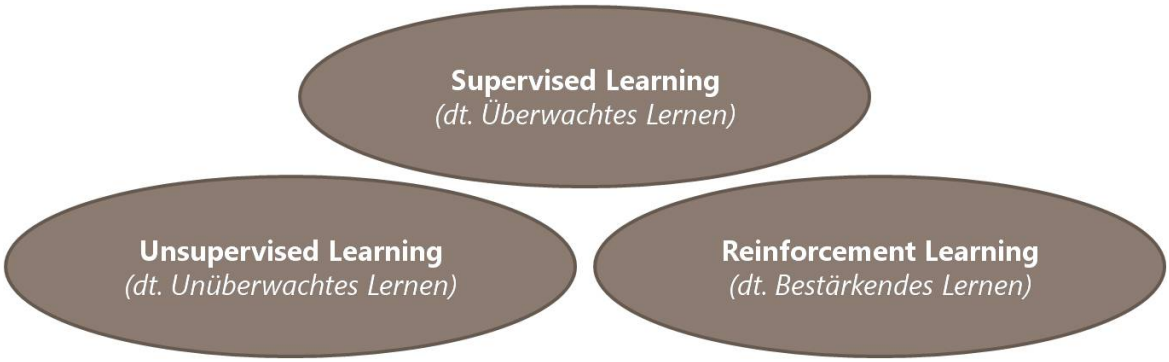

Abbildung 2.2: Lernsituationen des maschinellen Lernens

dem neuen Wissen an.
Das **Unsupervised Learning** verfügt über keinen solchen Lehrer und keine vorher bekannten Zielentscheidungen. Der Agent muss anhand der Vielfalt der Eingabedatensätze Schemata erkennen und diese klassifizieren. Die Klassifikation ist auch eine der häufigsten Anwendungsbereiche.
Beim **Reinforcement Learning** ist die Lernsituation eine Mischform aus Supervised und Unsupervised Learning. Der Agent erhält nach Auswahl seiner Entscheidung zwar keine beste Lösung präsentiert, er erhält jedoch ein Feedback: Ein sogenanntes „Reward“. Eine Belohnung oder auch Bestrafung, bei der es sich meistens um einen numerischen Wert handelt. Im Gegensatz zum Supervised Learning erfährt der Agent also nicht wie er reagieren soll, sondern wie gut er in der Vergangenheit reagiert hat. Diese Rewards können anfänglich auch verzögert Auftreten (z.B. Sieg eines Schachspiels) und müssen im Laufe vieler Trial-and-Error Testläufe gelernt werden.

Abkürzung	*Parameter (engl.)*	*Parameter (dt.)*
s	State	Zustand
a	Action	Aktion
r	Reward	Belohnung
π	Policy	Strategie

Tabelle 2.1: Parameter des Reinforcement Learning

Tabelle 2.1 fasst die grundlegenden Parameter im Reinforcement Learning zusammen. Alle Eingabeparameter ergeben einen Zustand s. Aus diesem Zustand heraus wird einen Aktion a bestimmt, für die der Agent je nach Güte eine Belohnung oder Bestrafung r erhält. Es ist stets das Ziel mit Hilfe einer Bewertungsfunktion die Gesamtbelohnung zu maximieren. Dies entspricht einer optimalen Strategie π. Um diese zu erreichen gibt es verschiedene Ansätze und Algorithmen. Als theoretische Grundlage dienen in vielen Fällen Markov-

Entscheidungsprozesse. Für tiefgreifendere Informationen zur Markov-Theorie sei auf Kapitel 13 in (Alpaydin, 2008) verwiesen.

2.2 Temporal Difference Learning

Im Folgenden wird der Ansatz des Lernens mit temporaler Differenz (TD-Learning) beschrieben. Das TD-Learning ist eine Klasse von Lernverfahren, die auf Ansätzen der Monte Carlo Methode und der Dynamischen Programmierung basiert (vgl. Wolter, 2008, S.15). Von der Monte Carlo Methode stammt das unabhängige Lernen rein aus Erfahrungen, während die Idee eine bereits vorhandene Bewertungsfunktionen kontinuierlich mit den neuen Erfahrungen anzupassen aus der Dynamischen Programmierung kommt.

Zwei verbreitete TD-Verfahren sind das **Q-Learning** und der **SARSA** Algorithmus. Ausgehend von einer initialen Bewertungsfunktion und einer beliebigen Anfangsstrategie lernt der Agent nun Stück für Stück seine Umwelt kennen und passt seine Bewertungsfunktion anhand der gemachten Erfahrungen an, um sich der optimalen Strategie anzunähern. Im Wesentlichen unterscheiden sich die genannten Algorithmen dadurch, dass SARSA ein sogenanntes On-Policy-Verfahren ist, das die zu optimierende Strategie gleichzeitig auch zur Aktionsauswahl nutzt, während beim Q-Learning (Off-Policy-Verfahren) hierfür eine separate Strategie Anwendung findet. Für detailliertere Informationen zu TD-Verfahren und speziell zu SARSA und Q-Learning sei auf Kapitel 6 in (Sutton und Barto, 1998)

Ein typisches Problem bei den TD-Verfahren ist das Finden eines brauchbaren Mittelwegs zwischen Exploration und Exploitation. Exploration steht für die Erkundung der Umwelt und dem Ausprobieren neuer, unbekannter Aktionsfolgen. Bei der Exploitation wird das bisher Gelernte ausgenutzt und schnellstmöglich das Ziel zu erreichen. Eine frühe Exploitation liefert die bisher optimale Aktionsfolge ohne zu wissen, ob eventuell bessere (noch unbekannte) Möglichkeiten existieren.
Ein Lösungsansatz für dieses Problem ist das ϵ-Greedy Verfahren. Bei diesem Verfahren wird mit einer Wahrscheinlichkeit von ϵ eine zufällige Aktion gewählt und mit 1-ϵ die bisher beste, gelernte Aktion. Beginnt man nun mit einem ϵ von 1 und verringert es kontinuierlich auf 0, so erkundet der Agent die Umgebung zu Anfang völlig willkürlich und nutzt im Laufe der Zeit die gemachten Erfahrungen zunehmend aus, um optimale Aktionsfolgen auszuführen. Je nachdem wie schnell man das ϵ verringert, lässt sich das Verhältnis zwischen Exploration und Exploitation steuern.

3 Methodisches Vorgehen am konkreten Anwendungsfall

In diesem Kapitel wird der konkrete Anwendungsfall mit dessen Zielsetzungen vorgestellt und konkrete Ansätze zum Erreichen der Ziele entwickelt.

Das Fahrzeug soll einen festen Kurs in möglichst geringer Zeit abfahren. Hierzu muss es mit wenigen Testläufen erlernen, wie schnell es abhängig von der jeweiligen Position fahren darf. Auf geraden Streckenabschnitten soll mit maximaler Geschwindigkeit gefahren werden. In Kurven so schnell wie möglich. Die Fahrbahn darf zu keinem Zeitpunkt verlassen werden. Hierzu werden die seitlichen Beschleunigungskräfte mit einem Beschleunigungssensor gemessen und aufgezeichnet. Je nach Beschaffenheit von Fahrbahn und Reifen gibt es eine maximale Kraft die nicht überschritten werden darf, ehe das Fahrzeug abdriftet. Vereinfachend wird von gleichbleibenden Rahmenbedingungen ausgegangen und die maximale Kraft empirisch ermittelt. Bei den folgenden Ausführungen betrachten wir beispielhaft den Rundkurs aus Abbildung 3.1.

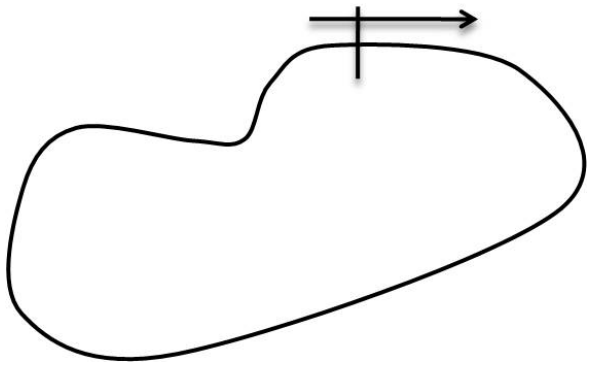

Abbildung 3.1: Beispielstrecke für das Modellfahrzeug

Abbildung 3.2 illustriert die gemessenen seitlichen Beschleunigungskräfte für je ein einmaliges Umfahren (ab der Startmarkierung) des Beispielkurses bei geringer und hoher Geschwindigkeit.

Je kleiner der Kurvenradius und schneller das Fahrzeug desto größer werden die Beträge der gemessenen Extremwerte (Peaks). Die empirisch ermittelten Maximalbeträge stellen

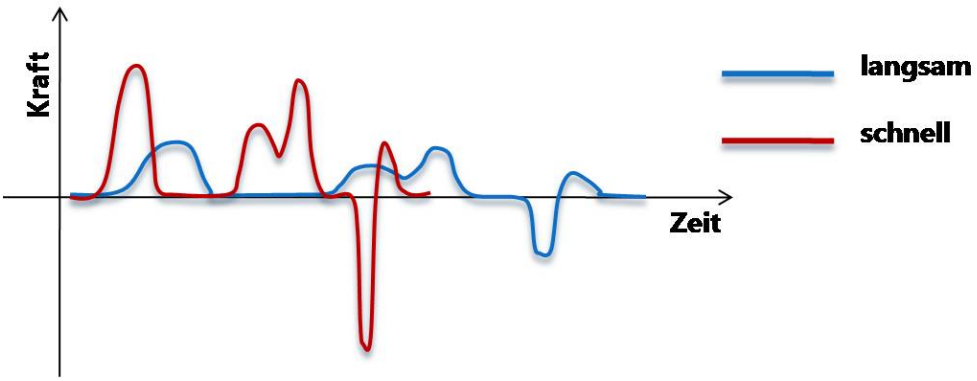

Abbildung 3.2: Beispielhafte Sensordaten bei unterschiedlichen Fahrgeschwindigkeiten

einen Grenzbereich dar, der nicht über- bzw. unterschritten werden darf, siehe Abb. 3.3. Andernfalls kann es passieren, dass das Fahrzeug aus der Kurve getrieben wird und die Fahrbahn verlässt.

Zu erreichen ist, dass die Peaks sich an diesen Grenzbereich annähern, ohne ihn zu überschreiten. Das Fahrzeug muss aus der gefahrenen Geschwindigkeit und den gemessenen Beschleunigungskräften lernen, wie stark es die zuvor gefahrene Geschwindigkeit vergrößern (oder auch verkleinern) muss, um sich möglichst optimal anzunähern. Abbildung 3.4 verdeutlich diesen Vorgang am Beispiel des ersten Peaks der Messdaten.

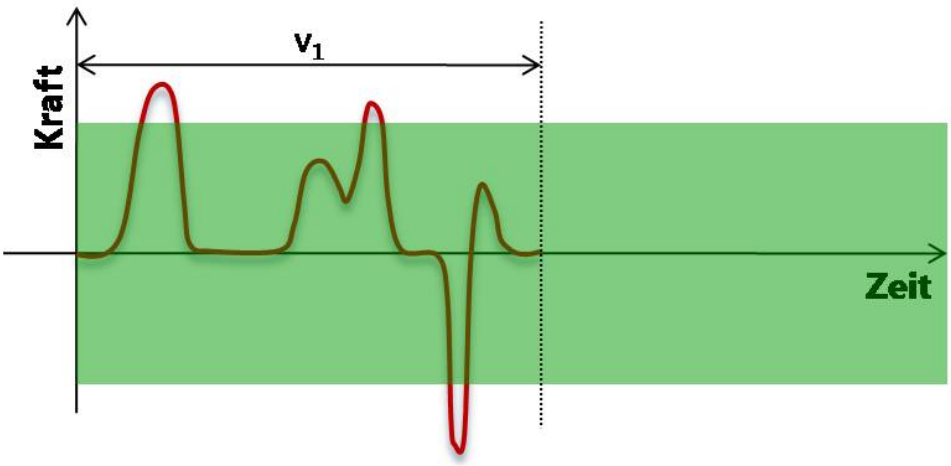

Abbildung 3.3: Überschreitung der maximalen seitlichen Beschleunigungskraft

Der erste Peak überschreitet die erlaubte, maximale seitliche Beschleunigungskraft. Das Fahrzeug wird in der betroffenen Kurve nicht mehr mit der zuvor konstanten Geschwindigkeit v_1 fahren, sondern diese für die Durchfahrt neu berechnen. Folglich ändern sich auch

die Beschleunigungskräfte. Dieses Vorgehen erfordert wenige Ressourcen für die Datenhaltung. Neben der Pflege der aktuellen Messreihe, müssen lediglich korrespondierende Geschwindigkeiten gespeichert werden. Als Abschätzung dient eine Geschwindigkeit für jedes Extremum in der Messreihe.

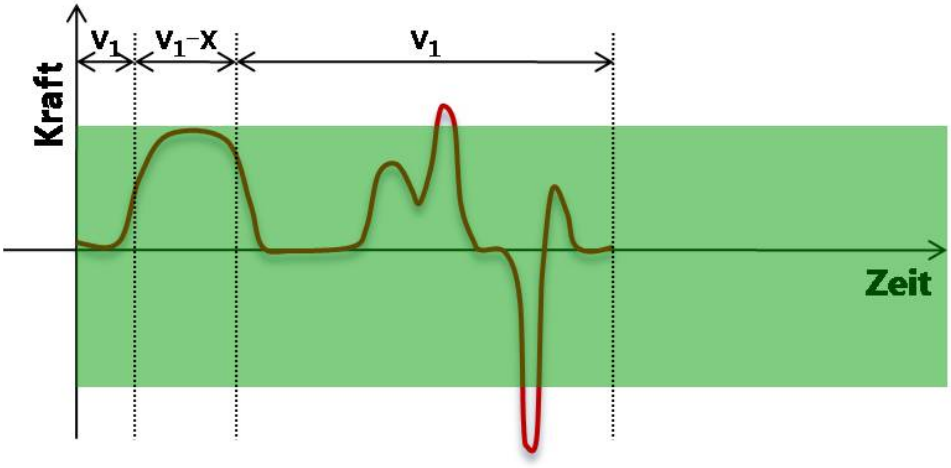

Abbildung 3.4: Regulierung der Beschleunigungskraft durch Geschwindigkeitsänderungen

Eine möglichst schnelle Annäherung an die erlaubte Beschleunigungskraft hängt in erster Linie von der neuen Geschwindigkeit ab. Um diese zu bestimmen kann man unterschiedlich ansetzen. Im einfachsten Fall wäre eine Änderung mit festen Schrittweiten denkbar, was unter Umständen viele Testdurchläufe notwendig machen würde. Alternativ wäre auch eine Geschwindigkeitsänderung linear zur Abweichung der Beschleunigungskraft möglich. Eine weitere Überlegung ist ausgehend von den aktuellen Messwerten alle Peaks so zu manipulieren, dass diese eine Soll-Kurve ergeben, die es zu erreichen gilt. Aus den Änderungen zu dieser Soll-Kurve, sowie der vorher gefahrenen Geschwindigkeit wird die neue Geschwindigkeit abgeleitet. Die Peaks können gestreckt bzw. gestaucht oder aber auch durch neue Funktionen ersetzt werden. Hier kämen z.B. Kubische Splines (KS) oder Radiale Basisfunktionen (RBF) in Frage.

4 Abschätzung der Risiken und Unsicherheitsfaktoren

In diesem Kapitel werden bisher angetroffene und erkannte Problempunkte spezifiziert und Lösungsvorschläge gemacht.

Bei der Umsetzung der Konzepte können Probleme an verschiedenen Stellen auftreten. Der **Beschleunigungssensor** muss noch auf dem Testfahrzeug installiert und getestet werden. Die Genauigkeit der Messwerte ist bisher nicht bekannt, aber Grundlage für die Ermittlung der maximalen seitlichen Beschleunigungskraft und Neuberechnungen der Fahrgeschwindigkeit.

Ein grundlegendes Problem beim Erstellen einer im vorherigen Kapitel vorgeschlagenen Soll-Kurve ist die Zeitvariation. Die zeitliche Zuordnung der Messwerte variiert, wenn man die Peaks an die Grenzwerte annähert. Abbildung 4.1 verdeutlicht das Problem.

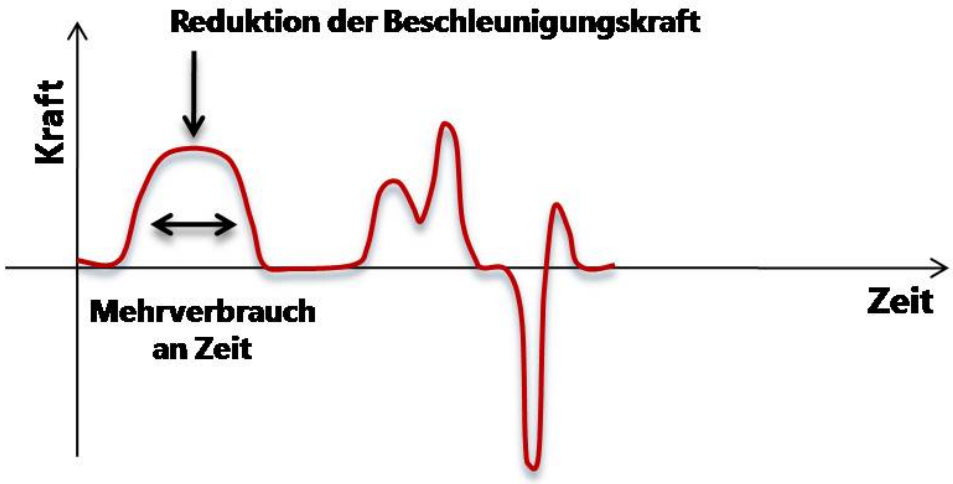

Abbildung 4.1: Auswirkungen einer Geschwindigkeitsreduktion auf die Beschleunigungskraft

Bei jeder Geschwindigkeitsänderung, wie z.B. der Geschwindigkeitsreduktion in Abb. 4.1, ändern sich auch die Beschleunigungskräfte und Maxima der Messreihe. Da das Fahrzeug

langsamer bzw. schneller fährt benötigt es für die gleiche Strecke nun auch mehr bzw. weniger Zeit. Für ein entsprechendes **Zeit-Mapping** muss gesorgt werden. Raum-Zeit Berechnungen auf physikalischen Grundgesetzen könnten bereits ausreichende Abhilfe für dieses Problem schaffen.

Ein weiteres Problem tritt auf da das Fahrzeug die Strecke unabhängig von einer festen Linie abfährt und diese nicht immer in gleicher Zeit bewältigt. So kann es passieren, dass die Position in der Messreihe und die aktuelle Fahrzeugposition zeitlich divergieren. Besonders bei längeren Strecken kann es notwendig werden Synchronisationspunkte beim Abfahren der Strecke zu bestimmen, damit das Fahrzeug die eigene Position innerhalb der aktuellen Messreihe wieder erkennt. Eine globale Umgebungskartografie, wie sie in (Rull, 2008) beschrieben ist, kann zur Lösung dieses Problems beitragen. Auffällige Streckenverläufe oder feste Punkte in der Umgebung können in einer solchen globalen Karte gekennzeichnet werden. Beim späteren Erkennen dieser Merkmale kann sich das Fahrzeug mit der aktuellen Messreihe der Beschleunigungswerte synchronisieren.

5 Ausblick

Um die beschriebenen Ansätze weiter zu entwickeln und möglicherweise auftretende Probleme zu erkennen wird eine Simulation geschrieben, mit deren Hilfe die Algorithmen und Abläufe getestet werden. In diesem Rahmen werden die verschiedenen Ansätze neue Fahrgeschwindigkeiten zu ermitteln untersucht. Insbesondere die Eignung von Radialen Basisfunktionen und Kubischen Splines zur Funktionsapproximation bei den Messreihen. Hierbei sollen von Anfang an (Sensor-)Ungenauigkeiten berücksichtigt werden. Parallel zur Simulationsentwicklung wird ein Beschleunigungssensor auf dem Fahrzeug installiert, in Betrieb genommen und auf empirischer Basis die maximale seitliche Beschleunigungskraft unter verschiedenen Umwelteinflüssen bestimmt. Ist schließlich ein praktisch geeignetes Gesamtkonzept erstellt, wird dieses von der Simulation auf das Fahrzeug portiert und unter realen Einflüssen in Betrieb genommen.

Existieren erste funktionsfähige Prototypen werden **Optimierungsmöglichkeiten** betrachtet. Neben einer Verbesserung der Effizienz der Algorithmen, kann z.B. auch ein schnelleres Approximieren an die optimalen Geschwindigkeiten angestrebt werden.

Im **Hinblick auf ein Masterarbeitsthema** könnten im Laufe der Entwicklung **komplexere Problempunkte** auftreten, aber auch die **Erweiterung auf andere Anwendungsfälle** in den Vordergrund rücken. Zum Beispiel wird die maximale seitliche Beschleunigungskraft, die auf das Fahrzeug wirken kann, ohne dass es von der Fahrbahn abkommt, noch vereinfachend als konstant angenommen und empirisch ermittelt. Unter echten Bedingungen ist diese jedoch variabel. Nässe, Eis, der Zustand der Reifen oder unbefestigte Untergründe wirken entscheidend darauf ein. Ein weiterer, denkbarer Anwendungsfall könnte die Ermittlung dieser Werte mit Hilfe von RL Methoden sein. Oder aber auch das Finden von zeitlichen oder räumlichen Ideallinien in Kurven.

Literaturverzeichnis

[Alpaydin 2008] ALPAYDIN, Ethem: *Maschinelles Lernen*. München : Oldenbourg, 2008. – ISBN 978-3-486-58114-0

[FAUST 2008] HAMBURG, HAW: *FAUST Fahrerassistenz- und Autonome Systeme*. 2008. – URL `http://www.informatik.haw-hamburg.de/faust.html`. – Abruf: 2008-12-08

[Rull 2008] RULL, Andrej: *Sensorbasierte Umgebungskartierung mit lokaler Positionskorrektur für autonome Fahrzeuge*, Hochschule für Angewandte Wissenschaften Hamburg, Bachelorarbeit, 2008

[Sutton und Barto 1998] SUTTON, Richard S. ; BARTO, Andrew G.: *Reinforcement Learning - An Introduction*. Cambridge : MIT Press, 1998. – ISBN 978-0262193986

[Wolter 2008] WOLTER, Anne: *Reinforcement Learning in der Roboter-Navigation*. Saarbrücken : Verlag Dr. Müller, 2008. – ISBN 978-3-639-04702-8