

Semantische Situationsrepräsentation und Aktionsbewertung zur Planung von Fahrmanöversequenzen

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der KIT-Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

Ralf Kohlhaas

aus Neuwied

Tag der mündlichen Prüfung: 24. Juni 2020

Erster Gutachter: Prof. Dr. J. Marius Zöllner

Zweiter Gutachter: Prof. Dr. Ralf H. Reussner

Danksagung

Auf dem Weg zu meiner Promotion haben mich viele Menschen begleitet, denen ich an dieser Stelle „Danke“ sagen möchte.

Ich danke ganz besonders meinem Doktorvater Prof. J. Marius Zöllner für sein Vertrauen und seine Bereitschaft, meine Promotion zu begleiten. Ich konnte sehr von seiner wissenschaftlichen Unterstützung und Betreuung in vielen fachlichen Gesprächen im Rahmen der Promotionsseminare aber auch im Büro, auf dem Flur und im Versuchsfahrzeug profitieren. Meinem Gutachter Prof. Ralf Reussner möchte ich für seine Unterstützung und die offenen und motivierenden Gespräche danken. Auch möchte ich meinen Prüfern Prof. Jörg Henkel und Prof. Jörn Müller-Quade danken, dass sie meine Prüfung ermöglicht und diese mit interessanten Fragen bereichert haben. Ich danke Herrn Giesbaum für die technische Unterstützung bei meiner Prüfung, die trotz Videokonferenz und Ausfall einer Kamera reibungslos abgelaufen ist.

Diese Arbeit entstand zu großen Teilen während meiner Zeit als wissenschaftlicher Mitarbeiter am FZI Forschungszentrum Informatik in der Abteilung TKS. Ich habe am FZI eine besondere wissenschaftliche Umgebung vorgefunden, in der ich meine Arbeit einbringen konnte. Einen großen Anteil daran hatten meine Kollegen bei TKS und der Nachbarabteilung IDS sowie die Kollegen vom KIT. Ich habe eine kollegiale, freundschaftliche und konzentrierte Atmosphäre vorgefunden, in der ich meine Arbeit gestalten konnte. Vielen Dank dafür.

Besonders hilfreich war dabei der wissenschaftliche Austausch in Kaffeepausen, beim Mittagessen, an Whiteboards, in den Laboren und vor allem im Rahmen von wissenschaftlichen Seminaren und den jährlichen Klausurtagungen. Danke für eure Gedanken, Ideen, Vorschläge, Rückmeldungen, eure Offenheit und Motivation. Und danke für die freundschaftliche Zusammenarbeit, aus der auch viele gemeinsame Veröffentlichungen entstanden sind. Gerne denke ich an unsere gemeinsamen Erlebnisse auf Konferenzreisen nach Amerika, China und Australien zurück.

Ich möchte mich besonders bei Dennis und Thomas, mit denen ich mir zu Beginn ein Büro geteilt haben, sowie Thomas und Tobias bedanken. Die vier haben mich, als ich ans FZI gekommen bin, herzlich in ihr Team aufgenommen und mich bei der Findung und Umsetzung meines Promotionsthemas unterstützt. Auf ihren Arbeiten und Ergebnissen konnte ich meine eigene Promotion aufbauen. Mein Dank gilt auch den beiden Abteilungsleitern Marcus und später Thomas, die u. a. durch den Aufbau des TAA-Labors und des Living Lab Automotive sehr gute Rahmenbedingungen für wissenschaftliches Arbeiten geschaffen haben. Ein herzliches

Dankeschön an Sonja und Diana im Sekretariat, die mich bei organisatorischen Problemen und im Kontakt mit Professoren und Universität unterstützt haben.

Mit den Jahren kamen weitere Kollegen hinzu, die mein Arbeitsleben und damit auch meine Promotion bereichert haben. Gerne denke ich an die vielen Stunden zurück, die ich zusammen mit Thomas, Thomas, Marc, Marc, Sebastian und Jens in CoCar verbracht habe, um neue Ideen und Algorithmen auszuprobieren, eine der zahllosen Fahrdemonstrationen oder Filmdrehs vorzubereiten, aber auch um zu erfahren, was autonomes Fahren wirklich bedeutet. Danken möchte ich meinem späteren Zimmerkollegen Florian für viele inspirierende fachliche und private Gespräche. Danke für die angenehme Zeit, in der wir unser gemeinsames Interesse am Filmemachen und Perfektionieren von Powerpoint-Präsentationen in dem einen oder anderen kleinen Film und Vortrag verewigen konnten. Ich danke auch den Kollegen aus dem RobustSENSE-Projekt, die durch ihren Einsatz eine bemerkenswerte Fahrdemonstration zusammengestellt haben, in deren Rahmen ich meine Konzepte im realen Fahrversuch ausprobieren konnte.

Danken möchte ich den vielen Studentinnen und Studenten und wissenschaftlichen Hilfskräften, die mich in meiner wissenschaftlichen Arbeit im Rahmen meiner Promotion unterstützt haben. Besonders danke ich Daniel, Dominik, Ecatarina, Ghazi, Isabel, Matheus, Max, Pierre, Silviu und Thomas für ihre frischen Ideen, wissenschaftliche Arbeiten und tatkräftige Unterstützung in und um CoCar herum. Thanks Michael for proof-reading the abstract.

Ich möchte ganz herzlich meiner Familie danken. Meine Eltern haben mir durch ihre Unterstützung ein Studium und die anschließende Promotion ermöglicht. Ich möchte ihnen dafür danken, dass sie mich immer wieder ermutigt haben, meine Arbeit zu einem erfolgreichen Ende zu bringen. Ein besonderer Dank gilt meiner Schwester Annika und ihrer Familie. Annika war mir mit ihrer Promotion ein anspornendes Beispiel und hat in unzähligen Stunden durch das Korrekturlesen meiner Arbeit zu deren Qualität beigetragen. Ganz besonders möchte ich meiner Frau Annamaria danken, die die ganze Zeit der Promotion an meiner Seite war und mir die nötigen Freiheiten gegeben hat, neben der Familie die notwendige Zeit in meine Arbeit zu stecken. Danken möchte ich auch meinen vier Kindern Tabea, Joshua, Oliver und Christopher, die das eine oder andere Mal auf ihren Papa verzichten mussten, weil der noch seine Doktorarbeit schreiben musste.

Abstract

Highly automated driving and autonomous vehicles have been a strong focus of research not only by automobile manufacturers and Tier 1 suppliers, but also IT companies and research institutes. Self-driving vehicles perceive their environment using sensors and carry out driving actions in order to reach their destination in a safe and efficient manner. In doing so, a model of the current traffic situation is often created. This model includes and integrates both current sensor data and background knowledge such as map data. Building on this, driving maneuvers can be planned to be carried out by the actuators of the vehicle.

In recent years, significant advances have been made in the field of highly automated driving. A primary challenge that has emerged is the automated understanding of complex traffic situations and, based on this, the generation of optimal driving strategies. Especially in inner-city areas, automated vehicles have to cope with a multitude of different road topologies. In these cases, specialized situation-dependent solutions reach their limits. Additionally, the existence of dynamic road users induces a higher combinatorial complexity to the planning process.

The goal of this work is both the active and passive support of the driver and the provision of abstract driving maneuvers for automated driving. The scientific contributions are

- an abstraction of the traffic situation through a relational, semantic scene representation and, based on this,
- methods for efficient planning and dynamic validation of driving strategies.

Using a relational model, the search space for planning driving maneuvers can be discretized at semantic boundaries. This enables a generalized planning process that is efficient due to reduction of the search space. Backprojection of the semantic plans into a geometric representation enables the dynamic validation and optimization of the resulting plans with regard to energy efficiency and comfort.

Hints on optimizing the driving style are communicated to the driver as part of an assistance system or are made available for trajectory planning of an automated vehicle. The methods are evaluated both on synthetic data and in real driving tests.

Kurzfassung

Hochautomatisiertes Fahren und autonome Fahrzeuge stehen seit einigen Jahren im Fokus der Forschung von Automobilherstellern, Zulieferern aber auch IT-Unternehmen sowie Forschungseinrichtungen. Selbstfahrende Fahrzeuge nehmen ihre Umgebung mittels Sensoren wahr und führen Fahraktionen aus, um ihr Ziel sicher und effizient zu erreichen. Dabei wird häufig ein Modell der Verkehrssituation erstellt, das sowohl aktuelle Sensordaten als auch Hintergrundwissen wie Kartendaten zusammenführt. Darauf aufbauend können dann Fahrmanöver geplant werden, die von der Aktorik des Fahrzeugs ausgeführt werden.

In den letzten Jahren wurden große Fortschritte auf dem Gebiet des hochautomatisierten Fahrens erzielt. Herausforderungen, die sich dabei gezeigt haben, sind das automatisierte Verstehen von komplexen Verkehrssituationen und darauf aufbauend die Generierung von optimalen Fahrstrategien. Automatisierte Fahrzeuge müssen gerade im innerstädtischen Bereich mit einer Vielzahl von unterschiedlichen Straßentopologien zurechtkommen. Situationsabhängige Speziallösungen stoßen dabei an ihre Grenzen. Hinzu kommt die durch dynamische Verkehrsteilnehmer induzierte kombinatorische Komplexität im Planungsprozess.

Ziel dieser Arbeit ist die aktive und passive Unterstützung des Fahrers sowie die Bereitstellung von abstrakten Fahrmanöversequenzen für automatisiertes Fahren. Wissenschaftliche Beiträge sind dabei

- eine Abstraktion der Verkehrssituation durch eine relationale, semantische Szenenrepräsentation sowie darauf aufbauend
- Methoden zur effizienten Planung und dynamischen Validierung von Fahrmanöversequenzen.

Durch eine relationale Modellierung kann der Planungsraum für Fahrmanöver an semantischen Grenzen diskretisiert werden. Dies erlaubt eine einheitliche und durch die Reduktion des Raums effiziente Planung. Die Rückführung der semantischen Pläne in eine geometrische Darstellung ermöglicht die dynamische Validierung und Optimierung resultierender Pläne hinsichtlich Energieeffizienz und Komfort.

Hinweise zur optimalen Fahrweise werden als Teil einer Assistenzfunktion dem Fahrer kommuniziert oder für die Trajektorienplanung automatisierter Fahrzeuge zur Verfügung gestellt. Die Verfahren werden sowohl auf synthetischen Daten als auch im realen Fahrversuch evaluiert.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Wissenschaftlicher Beitrag	4
1.3	Struktur	5
2	Grundlagen und Stand der Technik	7
2.1	Automatisierungsgrade	8
2.2	Wissenschaftliche und technische Ausgangsbasis	10
2.2.1	Relevante Projekte	11
2.2.2	Forschungsfahrzeug „CoCar“	13
2.3	Verkehrsszenenrepräsentationen	14
2.3.1	Fahrzeugmodelle	15
2.3.2	Umgebungsmodelle	20
2.4	Straßengeometrie und -topologie	24
2.4.1	Straßenkarten	26
2.4.2	Fahrspurkarten	26
2.4.3	Topologische Netze	27
2.4.4	Spurgenaue Karten – Lanelets	27
2.5	Perzeption von Verkehrsteilnehmern	29
2.5.1	Objekterkennung auf Videobildern	29
2.5.2	Objekterkennung auf Video-Tiefenbildern	32
2.5.3	Fusion	34
2.5.4	Objekterkennung auf Laserscanner-Punktwolken	35
2.6	Prädiktion	35
2.6.1	Manöverbasierte Prädiktion von Routen und Trajektorien	36
2.7	Kartierung und Lokalisierung	38
2.8	Ansätze zur Aktionsentscheidung	39
2.9	Aktionsplanung	42
2.9.1	Graphbasierte Planer - Graphensuche	43
2.9.2	Geometrische Planer	46
2.9.3	Homotopiebasierte Planung	48
2.10	Zusammenfassung und Fazit	50
3	Semantische Planung zur Manöverentscheidung	51
3.1	Einbettung in die Systemarchitektur	52
3.2	Definition Manöver	53
3.3	Konzeptionelle Anforderungen an eine Manöverentscheidung	54
3.4	Struktur des Gesamtsystems	55
3.4.1	Geometrische Umweltmodellierung	57

3.4.2	Semantisches Umweltmodell	61
3.4.3	Semantische Planung	63
3.4.4	Dynamikvalidierung und -evaluierung	64
3.5	Zusammenfassung und Fazit	65
4	Semantische Szenenmodellierung	67
4.1	Relationale Szenenmodellierung – Verkehrsszenenontologie . .	67
4.1.1	Relationen zwischen statischen Szenenobjekten	69
4.1.2	Relationen zum Pfad	70
4.1.3	Relationen zu Fahrzeugen	71
4.2	Semantischer Planungsraum	76
4.3	Aufbau des Zustandsraums	78
4.3.1	Betrachtung der statischen Szenenobjekte	79
4.3.2	Betrachtung der dynamischen Szenenobjekte	80
4.3.3	Temporale Fortsetzung	81
4.3.4	Anwendung von Regeln – Modellierung von Verkehrsregeln	83
4.3.5	Zusammenfassung des Ablaufs	85
4.4	Analyse	85
4.4.1	Beispielszenarien	87
4.4.2	Relationale Szenenmodellierung	89
4.4.3	Semantischer Zustandsraum	93
4.4.4	Größe des Zustandsraums	97
4.4.5	Komplexitätsklasse der Zustandsraumgröße	98
4.5	Probabilistische Betrachtung der semantischen Szenenbeschreibung	101
4.5.1	Varianten der Unsicherheit	101
4.5.2	Ansätze zur probabilistischen Modellierung des Zustandsraums	103
4.6	Zusammenfassung und Fazit	108
5	Planung auf semantischen Zuständen	111
5.1	Graphensuche	113
5.2	Homotopieklassen	114
5.2.1	Homotopiebausteine	116
5.2.2	Anwendung der Homotopiebausteine	119
5.3	Evaluation	120
5.3.1	Planungsverfahren	121
5.3.2	Anwendungen	129
5.4	Zusammenfassung und Fazit	133
6	Dynamikvalidierung und -evaluierung	135
6.1	Anforderungen und Eigenschaften der Geschwindigkeitsprofiloptimierung	135
6.1.1	Repräsentation des Lösungsraums	136
6.1.2	Diskretisierung des Zustands- und Aktionsraums	136

6.1.3	Mögliche Verkehrsszenenklassen	137
6.1.4	Art des Optimums	137
6.1.5	Zielfunktion	137
6.1.6	Laufzeit	138
6.2	Optimierungsverfahren	139
6.2.1	Dynamisches Programmieren	139
6.2.2	Modellbasierte lokale Optimierung	144
6.2.3	Vorberechnete optimale Geschwindigkeitsprofile	146
6.2.4	Modellprädiktion	153
6.2.5	Sichtgraphenverfahren	156
6.2.6	Zusammenfassung	157
6.3	Dynamikvalidierung von semantischen Plänen	158
6.3.1	Semantischer Plan als Ort-Zeit-Schlauch	159
6.3.2	Einbindung in Heuristik-Suche	161
6.4	Evaluation	162
6.4.1	Dynamikvalidierung von semantischen Plänen	163
6.4.2	Integrierte Dynamikvalidierung - EA*S	165
6.4.3	Laufzeitverhalten	169
6.4.4	Anwendung – Überholassistentz	170
6.5	Zusammenfassung und Fazit	171
7	Evaluation im Gesamtsystem	173
7.1	Szenario	173
7.2	Systemarchitektur und Schnittstellen	173
7.3	Visualisierung der Verarbeitungsschritte	175
7.4	Demonstration und Ergebnisse	178
8	Zusammenfassung und Ausblick	181
8.1	Zusammenfassung der Ergebnisse	181
8.2	Ausblick	182

1 Einleitung

Hochautomatisiertes Fahren und autonome Fahrzeuge stehen im Fokus der Forschung von Automobilherstellern, Zulieferern aber auch IT-Unternehmen sowie Forschungseinrichtungen. In den letzten Jahren wurden große Fortschritte auf diesem Gebiet erzielt. Selbstfahrende Fahrzeuge nehmen ihre Umgebung mittels Sensoren wahr und führen Fahraktionen aus, um ihr Ziel sicher und effizient zu erreichen. Dabei wird häufig ein Modell der Verkehrssituation erstellt, das sowohl aktuelle Sensordaten als auch Hintergrundwissen wie z.B. Kartendaten zusammenführt. Darauf aufbauend können dann Fahrmanöver in unterschiedlichen Abstraktionen geplant werden, die von der Aktorik des Fahrzeugs ausgeführt werden.

In dieser Arbeit werden Methoden für die Situationsanalyse zur Bewertung und Auswahl von optimierten Fahrmanöversequenzen hinsichtlich Energieeffizienz und Komfort vorgestellt. Dazu wird basierend auf Umfelddaten eine semantische Straßenszenenrepräsentation zur Ableitung von Fahrstrategien aufgebaut. Ziel ist dabei die aktive und passive Unterstützung des Fahrers sowie die Bereitstellung von abstrakten Fahrmanöversequenzen für automatisiertes Fahren. Ausgehend von einer geometrischen Umfeldmodellierung wird ein semantischer Zustandsraum abgeleitet. Hierauf können Manöversequenzen geplant werden. Für die Manöversequenzen werden optimale Geschwindigkeitsprofile abgeleitet, die eine Validierung und Bewertung der Sequenzen erlauben. Fahrhinweise werden als Assistenzfunktion dem Fahrer kommuniziert oder für die Trajektorienplanung automatisierter Fahrzeuge zur Verfügung gestellt.

1.1 Motivation

Komplexe Fahrsituationen stellen selbst für erfahrene Autofahrer eine Herausforderung dar. Häufig kann die optimale Fahrentscheidung nur auf Grundlage von großer Erfahrung und unter Kenntnis der genauen örtlichen Gegebenheiten getroffen werden. Die optimale Fahrentscheidung hängt dabei von vielen unterschiedlichen Parametern der Verkehrsszene ab. Abbildung 1.1 veranschaulicht einige der relevanten Faktoren und Einflussgrößen anhand einer relativ einfachen Verkehrsszene.

Die Abbildung stellt eine Kreuzungssituation im innerstädtischen Bereich dar. Ein Fahrzeug (grau) nähert sich auf einer zweispurigen Straße einer beampelten Kreuzung und möchte diese geradeaus passieren. Vor ihm steht ein weiteres Fahrzeug (rot) auf der rechten Spur an der Ampel. Es hat den Blinker rechts

1 Einleitung

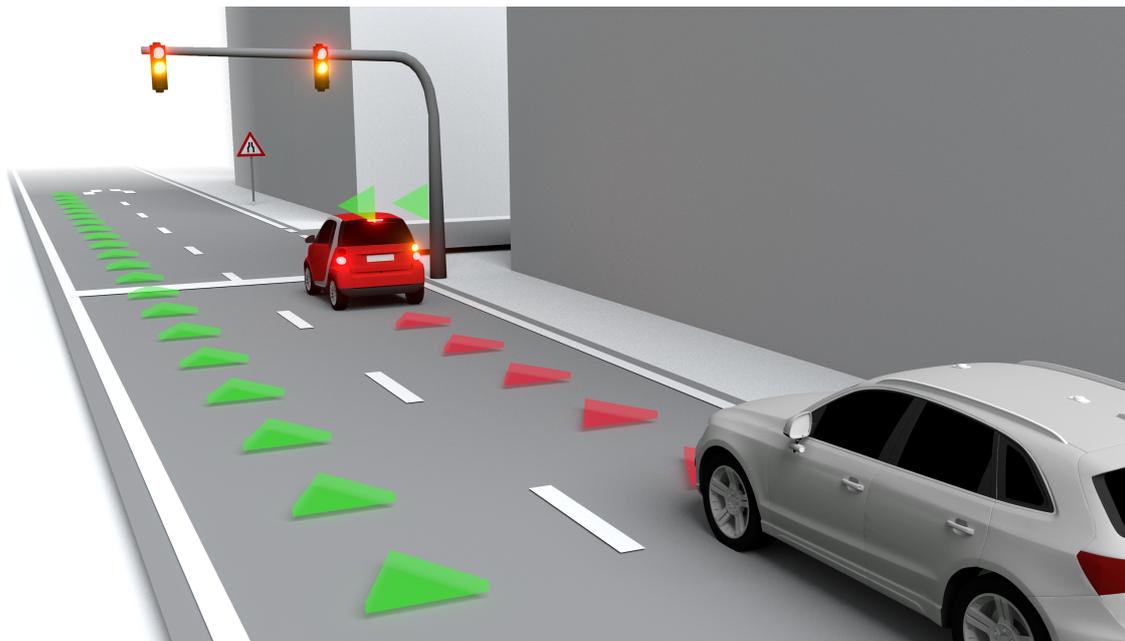


Abbildung 1.1: Schematische Darstellung einer Verkehrssituation in der Innenstadt. Das hintere Fahrzeug hat verschiedene Optionen bei der Wahl seiner Fahrmanöver (rote und grüne Pfeile). Diese sind abhängig von Faktoren und Einflussgrößen der Verkehrsszene wie z. B. : gültige Verkehrsregeln, Straßenverlauf und -topologie, Verkehrsinfrastruktur, andere Verkehrsteilnehmer und ihr zukünftiges Verhalten, allgemeiner zeitlicher Verlauf der Szene, aktueller Egozustand, Grenzen der Fahrzeugdynamik sowie gewünschte Route.

gesetzt, so dass davon ausgegangen werden kann, dass es in die Straße nach rechts einbiegen wird. Die Ampel zeigt rot, so dass ein Passieren der Kreuzung für beide Fahrzeuge noch nicht erlaubt ist. Das zusätzlich eingeschaltete gelbe Licht signalisiert jedoch einen baldigen Wechsel auf grün und somit freie Fahrt.

Das graue Fahrzeug hat nun prinzipiell zwei Möglichkeiten. Es kann auf der Spur bleiben (rote Pfeile), muss jedoch dazu stark verzögern. Auch wenn das vorausfahrende Fahrzeug in naher Zukunft über die Ampel fahren wird, so wird seine Beschleunigung und Geschwindigkeit aufgrund des geplanten Abbiegevorgangs gering ausfallen.

Alternativ kann das graue Fahrzeug die Spur wechseln und das rote Fahrzeug überholen. Auch wenn durch den Spurwechsel ein erhöhter Aufwand entsteht, scheint diese Alternative die bessere zu sein. Das graue Fahrzeug wird dann nicht mehr durch das rote in seiner Geschwindigkeit beschränkt. Unter Berücksichtigung des bevorstehenden Grünschaltes der Ampel kann diese mit wenig verringerter oder mit gleichbleibender Geschwindigkeit passiert werden.

Im Allgemeinen ist im innerstädtischen Bereich die Spurwahl auch stark durch die

geplante Route des Fahrzeuges bestimmt, um z. B. bei Abbiegungen oder Spurgabelungen auf die gewünschte Straße fahren zu können. In diesem Beispiel hat die Spurwahl jedoch keinen Einfluss auf die zukünftige Ausführung der geplanten Route, da beide Spuren im weiteren Straßenverlauf zu einer zusammengeführt werden.

An diesem Beispiel wird klar, dass selbst eine einfache Fahrentscheidung wie ein Spurwechsel in dieser Kreuzungssituation von vielen Faktoren und Einflussgrößen abhängt, die bei der Entscheidungsfindung berücksichtigt werden müssen. Im Folgenden werden diese in systematischer Form allgemein besprochen.

Die Straßentopologie also die Gestalt der Straße wie z. B. das Vorhandensein von Kreuzungen oder die Anzahl der Fahrspuren definiert zusammen mit der jeweils gewünschten Route die möglichen Bewegungen der Verkehrsteilnehmer und somit auch des Ego-Fahrzeugs. Häufig spielen Straßenabschnitte eine Rolle, die außerhalb des direkten Wahrnehmungsbereiches eines Fahrzeugs liegen. Hier kann Kartenmaterial hinzugezogen werden, um diese Information im Fahrzeug bereitzustellen.

Der aktuelle Egozustand, d. h. die aktuelle Fahrzeugposition und -geschwindigkeit, spielt als Ausgangszustand eine wichtige Rolle bei der Fahrentscheidung. Auch Grenzen der Fahrzeugdynamik wie maximale Brems- oder Kurvenbeschleunigung sowie Komfortparameter des Fahrers bzw. der Insassen schränken die Fahrentscheidung zusätzlich ein.

Andere Verkehrsteilnehmer beschränken zum einen die Möglichkeiten der Fahrzeugbewegung, zum anderen sind sie auch Voraussetzung für einige Fahrmanöver, wie z. B. Überholmanöver. Bei anderen Verkehrsteilnehmern ist nicht nur ihre Position und Geschwindigkeit sondern auch zukünftiges Verhalten bzw. ihre geplante Route relevant.

Verkehrsregeln wie die zulässige Maximalgeschwindigkeit werden zum einen explizit durch Verkehrsschilder, jedoch auch durch die Umgebung bzw. den Kontext der Szene gegeben. Elemente der Verkehrsinfrastruktur beeinflussen ebenfalls die Fahrentscheidung. Bei zeitlich veränderlichen Signalen, wie Ampeln ist nicht nur der aktuelle Zustand, sondern auch der zukünftige zeitliche Verlauf des Signals entscheidend.

Die Szene wird beschrieben durch gültige Verkehrsregeln, den Straßenverlauf und -topologie, die Position und aktuelle Bewegung der Verkehrsteilnehmer sowie weitere Verkehrsinfrastruktur wie z. B. Ampeln und deren Zustand. Für eine Fahrentscheidung wird jedoch noch weiteres Wissen z.B. über den weiteren Streckenverlauf, die zukünftigen Bewegungen der anderen Verkehrsteilnehmer und zukünftige Zustandsänderungen der Infrastruktur benötigt.

Es ist somit nicht nur die aktuelle Situation, sondern auch deren zukünftige zeitliche Entwicklung und der weitere Verlauf der Straßentopologie entlang der Route entscheidend. Eine Entscheidung rein auf Basis der aktuellen Situation ist nur mit sehr viel „Erfahrung“ möglich. Denn es muss die Situation in ihrer ganzen

1 Einleitung

Komplexität aus Zusammenhängen, möglichen Entwicklungen und Auswirkungen modelliert werden und eine Abbildung auf eine Entscheidung vorhanden sein. Soll diese Entscheidung automatisiert getroffen werden, sind entscheidungsbasierte Verfahren, wie verhaltensbasierte Ansätze, strategiebasierte Verfahren wie case-based reasoning oder Lernverfahren wie künstliche neuronale Netze auf eine umfassende Datenbasis und eine aufwändige Vorverarbeitung angewiesen, die alle auftretenden Situationen weitestgehend abdeckt. Planungsverfahren verlagern diese Komplexität auf die Ausführungszeit und sind somit auch für bisher unbekannte Szenarien einsetzbar. Je nach Komplexität und Modellierung des Planungsraums stoßen diese Verfahren jedoch an Rechenzeitgrenzen, so dass sinnvolle Abstraktionen verwendet werden müssen. Auch müssen gerade im Straßenverkehr semantisch beschriebene Regeln, wie z. B. Verkehrsregel in die Planung mit aufgenommen werden können. Daher wird in dieser Arbeit zur Verhaltensentscheidung eine Planung auf einer semantischen Zustandsraumbeschreibung vorgestellt.

1.2 Wissenschaftlicher Beitrag

Aktuelle Herausforderungen im Bereich des automatisierten Fahrens sind das automatisierte Verstehen von komplexen Verkehrssituationen und darauf aufbauend die Generierung von optimalen Fahrstrategien. Automatisierte Fahrzeuge müssen gerade im innerstädtischen Bereich mit einer Vielzahl von unterschiedlichen Straßentopologien zurechtkommen. Situationsabhängige Speziallösungen stoßen dabei an ihre Grenzen. Hinzu kommt die durch dynamische Verkehrsteilnehmer induzierte kombinatorische Komplexität im Planungsprozess.

Diese Arbeit beschreibt neuartige Ansätze zur aktive und passive Unterstützung des Fahrers sowie zur Bereitstellung von abstrakten Fahrmanöversequenzen für automatisiertes Fahren. Wissenschaftliche Beiträge stellen dabei eine Abstraktion der Verkehrssituation durch eine relationale, semantische Szenenrepräsentation sowie darauf aufbauend Methoden zur effizienten Planung und dynamischen Validierung von Fahrmanöversequenzen dar.

Durch eine relationale Modellierung kann der Planungsraum für Fahrmanöver an semantischen Grenzen diskretisiert werden. Dies erlaubt eine einheitliche und durch die Reduktion des Raumes effiziente Planung. Die Rückführung der semantischen Pläne in eine geometrische Darstellung ermöglicht die dynamische Validierung und Optimierung resultierender Pläne hinsichtlich Energieeffizienz und Komfort.

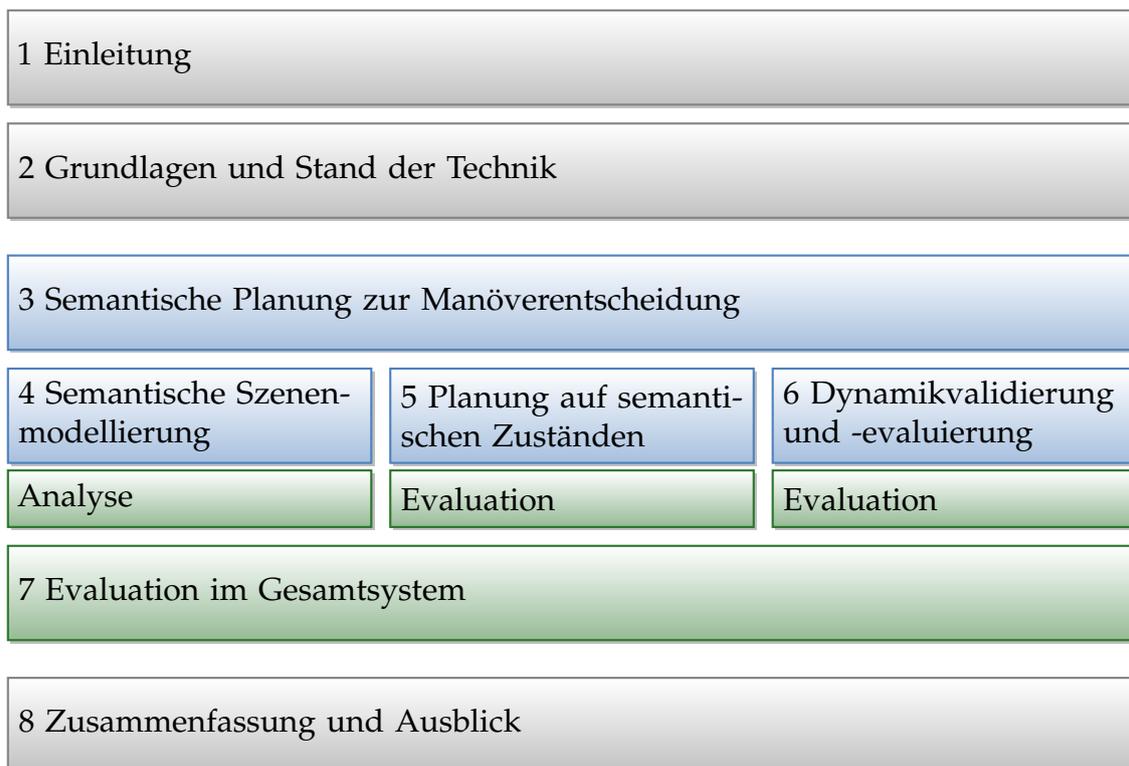


Abbildung 1.2: Schematische Darstellung des Aufbaus der Arbeit.

1.3 Struktur

Die vorliegende Arbeit gliedert sich wie in Abbildung 1.2 dargestellt. Nach der Einleitung in Kapitel 1 werden in Kapitel 2 die Grundlagen der in dieser Arbeit vorgeschlagenen Konzepte dargestellt. Einleitend wird die aktuelle Entwicklung hin zum automatisierten Fahren sowie ausgewählte Vorhaben im Bereich des automatisierten Fahrens, die den wissenschaftlichen Rahmen dieser Arbeit bilden, vorgestellt. Dabei werden auch verwandte Konzepte und Arbeiten zur Verhaltensentscheidung beleuchtet.

Das Konzept zur Ableitung optimierter Fahrmanöversequenzen auf Basis einer semantischen Szenenbeschreibung bildet der Kern dieser Arbeit und wird in Kapitel 3 vorgestellt. Detaillierte Beschreibungen und Analysen der drei wesentlichen Teilaspekte des Konzeptes finden sich in den darauffolgenden Kapiteln: Semantische Szenenmodellierung (4), Planung auf semantischen Zuständen (5) und Dynamikvalidierung und -evaluierung (6). Die jeweiligen Teilaspekte werden am Ende eines jeden Kapitels einzeln analysiert bzw. evaluiert.

Abschließend wird in Kapitel 7 die Anwendbarkeit des hier vorgestellten Konzepts in einem Gesamtsystem für das automatisierte Fahren im Rahmen einer realen Fahrdemonstration evaluiert. Kapitel 8 schließt den Rahmen mit einer Zusammenfassung und gibt einen Ausblick auf weitere mögliche Forschungsthemen, die auf dieser Arbeit aufbauen können.

2 Grundlagen und Stand der Technik

Automatisierte Fahrzeuge interagieren mit ihrer Umgebung über Sensoren und Aktoren. Mit Hilfe von Sensoren nehmen sie ihre Umgebung wahr (Perzeption). Auf Basis der wahrgenommenen Informationen leiten sie anschließend Fahrentscheidungen ab (kognitive Entscheidung). Aktuatoren werden verwendet, um den eigenen Zustand in der Umgebung zu ändern oder Signale an andere Verkehrsteilnehmer zu senden (Ausführung). Automatisierte Fahrzeuge können somit als mobile technisch kognitive Systeme bezeichnet werden [TKZ+16]. Der Signalfluss der Interaktion eines solchen Systems mit seiner Umgebung ist in Abbildung 2.1 schematisch dargestellt.

Im Rahmen der kognitiven Entscheidung wird häufig aus den wahrgenommenen Informationen ein konsistentes Umgebungsmodell aufgebaut. Zusammen mit einer Abbildung des eigenen Zustands ergibt sich eine Repräsentation der Gesamtsituation. Im Bereich des automatisierten Fahrens beinhaltet diese Verkehrsszenenrepräsentation z. B. Kartendaten sowie Informationen über andere Verkehrsteilnehmer, ihre Interaktionen untereinander und mit dem eigenen Fahrzeug sowie das eigene Fahrzeugmodell. Aufbauend auf dieser Situationsrepräsentation werden Fahrentscheidungen getroffen. Diese können entweder direkt auf Basis der vorhandenen Daten abgeleitet werden oder unter Berücksichtigung der prädizierten Szenenentwicklung geplant werden.

Auf einzelne Komponenten des automatisierten Autos als kognitiven Systems wird im Folgenden genauer eingegangen. Zuvor wird die historische und aktuelle Entwicklung hin zum automatisierten Auto entlang verschiedener Automatisierungsgrade und anhand verschiedener Beispiele veranschaulicht. Anschließend

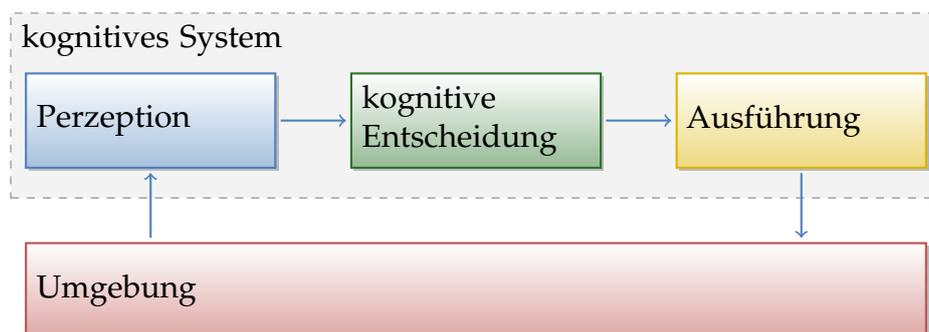


Abbildung 2.1: Ein kognitives System interagiert mit seiner Umgebung. Der Agent besteht hauptsächlich aus drei Komponenten: Perzeption, kognitive Entscheidung und Aktion.

In Anlehnung an [TKZ+16]

werden Ansätze und Verfahren zur Verkehrsszenenrepräsentation vorgestellt. Beim automatisierten Fahren kommt dabei der Darstellung der statischen Umgebung z. B. durch Straßenkarten eine besondere Rolle zu. Es wird gezeigt, wie die benötigten Informationen durch Perzeptionskomponenten bereitgestellt werden können. Abschließend werden Verfahren zur Aktionsgenerierung mit besonderem Fokus auf der Aktionsplanung vorgestellt.

2.1 Automatisierungsgrade

Seit der Erfindung des Autos vor 125 Jahren hat sich dieses von einer Kutsche mit Motor und ohne Pferd zu einem intelligenten, vernetzten System mit vielerlei Assistenz- und Automatisierungsfunktionen entwickelt. Besonders in den letzten Jahren mit der Miniaturisierung von Elektronik halten immer mehr Steuergeräte (engl.: electronic control units, ECUs) Einzug in aktuelle Serienprodukte. Zusammen mit entsprechenden Sensoren für die Überwachung des Fahrzeugzustandes und des -umfeldes ermöglicht dies dem Fahrzeug immer mehr Bereiche der Fahraufgabe zu übernehmen. Von einfachen Geschwindigkeitsregelanlagen über adaptive Abstandsregeltempomaten bis hin zu Staupiloten und Einparkassistenten reicht die Liste der verfügbaren Assistenzfunktionen.

Entsprechend ihrer Fähigkeiten und der Aufgabenverteilung zwischen Fahrer und Fahrzeug unterscheidet man mehrere Automatisierungsgrade. Sie legen fest wie die Kooperation von Fahrer und Fahrzeug gestaltet ist und definiert die Verantwortungsverteilung. Es gibt verschiedene Ansätze die Automatisierungsgrade zu differenzieren.

Im Projekt HAVEit¹ (2008 - 2011) werden fünf Automatisierungsgrade unterschieden. Hierbei liegt der Fokus auf der Zusammenarbeit zwischen Fahrer und Fahrzeug. So wird z. B. auch ein rein passiv assistiertes Fahren beschrieben, bei dem der Fahrer die Fahraufgabe vollständig übernimmt und lediglich Hinweise vom Fahrzeug bekommt.

Driver only

Der Fahrer fährt ganz ohne Assistenzfunktionen.

Assisted

Der Fahrer übernimmt die Fahraufgabe. Assistenzfunktionen können maximal Hinweise z. B. über Anzeigen oder akustische Signale geben.

Semi-automated

Das Fahrzeug übernimmt den Teil der Fahraufgabe, der die Längsregelung betrifft. Die Querführung wird weiterhin vom Fahrer übernommen.

¹ HAVE IT Website: haveit-eu.org (6.8.2017)

Highly-automated

Das Fahrzeug übernimmt Längs- und Querverführung. Der Fahrer überwacht das Fahrgeschehen und übernimmt in Notsituationen und bei Fehlfunktionen.

Fully-automated

Das Fahrzeug ersetzt den Fahrer vollständig. Es wird kein Fahrer mehr im Fahrzeug benötigt.

Die National Highway Traffic Safety Administration (NHTSA) der USA definierte 2013 Automatisierungsgrade mit einer etwas anderen Ausrichtung [Nat13]. Hier liegt der Fokus auf den Fähigkeiten des Fahrzeugs. Auch das Bundesamt für Straßenwesen (BASt) hat eine sehr ähnliche Nomenklatur mit folgenden Stufenbezeichnungen veröffentlicht: Driver only, Assistent, Teilautomatisiert, Hochautomatisiert und Vollautomatisiert. [GAA+12]

Aufbauend auf der Nomenklatur der NHTSA hat die Society of Automotive Engineers (SAE) einen Standard veröffentlicht [On-16], der seither weithin gebräuchlich ist. Dieser greift die unterschiedlichen Definitionen des höchsten Automatisierungsgrades der Definitionen von NHTSA und BASt auf.

Die SAE-Stufen sind wie folgt definiert:

No-Automation (Level 0)

Der Fahrer hat die volle Fahraufgabe.

Driver Assistance (Level 1)

In dieser Automatisierungsstufe sind eine oder mehrere Automatisierungsfunktionen aktiv. Beispiele sind elektronisches Stabilitätsprogramm (ESP) oder Notbremsassistent.

Partial Automation (Level 2)

Mindestens zwei Kontrollfunktionen arbeiten zusammen, um einen Teil der Fahraufgabe zu übernehmen. Ein Beispiel ist das Zusammenwirken von Abstandsregeltempomat (engl.: adaptive cruise control, ACC) und aktivem Spurhalteassistent.

Conditional Automation (Level 3)

Das Fahrzeug übernimmt die Fahrfunktion in einigen ausgewählten Situationen. Es erkennt seine Systemgrenzen und übergibt, falls es die Fahrfunktion nicht mehr ausführen kann mit einer ausreichenden Vorwarnzeit an den Fahrer.

High Automation / hoch automatisiert (Level 4)

Das Fahrzeug übernimmt die Fahrfunktion in allen Situationen. Vor dem Erreichen einer Systemgrenze kann das Fahrzeug eine Übergabe an den Fahrer anfordern. Es in diesem Fall auch ohne Eingreifen des Fahrers ein sicherer Zustand erreicht.

Full Automation / vollständig automatisiert (Level 5)

Das Fahrzeug übernimmt die Fahrfunktion in allen Situationen. Ein automatisierter Betrieb ist in allen Fahrsituationen möglich, in denen auch ein Mensch fahren kann.

Die Bedeutung gleicher Stufen in verschiedenen Definitionen weicht trotz ähnlicher Bezeichnung teilweise ab. Eine Übersicht der Level und eine Zuordnung der verschiedenen Bezeichnungen wird in [Smi13] gegeben. Um Widersprüche zu vermeiden wird im Folgenden bei der Angabe von Automatisierungsstufen die Definition der SAE verwendet. Zur Unterscheidung von Assistenzfunktionen wird folgende Bezeichnung verwendet:

Passive Assistenz

Rein hinweisgebendes System entsprechend der Stufe „Assisted“ in der Definition von HAVEit.

Aktive Assistenz

Assistenzfunktionen die in die Steuerung des Fahrzeugs eingreifen entsprechend der Stufe „Driver Assistance“ oder höher der SAE.

2.2 Wissenschaftliche und technische Ausgangsbasis

Auf dem Weg zum vollständig automatisierten Fahren arbeiten viele Forschungseinrichtungen, Fahrzeughersteller und Zulieferer jedoch auch Unternehmen aus dem IT-Bereich an selbst fahrenden Fahrzeugen. So forschen deutschlandweit z. B. die Universität Ulm², Braunschweig³, Freie Universität Berlin⁴, das Karlsruher Institut für Technologie (KIT)⁵ sowie das FZI Forschungszentrum Informatik (FZI)⁶ an Lösungen für das autonome Fahren.

Auch Autohersteller und Zulieferer haben in den letzten Jahren Demonstrationsfahrten und Projekte zum automatisierten Fahren durchgeführt. So hat z. B. Daimler im Rahmen des Projekts „S 500 INTELLIGENT DRIVE⁷“ im September 2013 eine autonome Langstreckenfahrt mit einer 2014er Mercedes S-Klasse

² Autonomes Fahren und intelligente Fahrzeugsysteme - Universität Ulm: www.uni-ulm.de/forschung/forschungsschwerpunkte/autonomes-fahren-und-intelligente-fahrzeugsysteme/ (26.03.2020)

³ Stadtpilot: www.tu-braunschweig.de/stadtpilot (26.3.2020)

⁴ AutoNOMOS Labs | Autonomous Cars from Berlin: autonomos-labs.com (26.03.2020)

⁵ KIT - MRT - Research / Forschung - Autonomous Driving: www.mrt.kit.edu/3203.php (27.03.2020)

⁶ Sichere und intelligente Fahrzeuge : FZI Forschungszentrum Informatik: www.fzi.de/de/forschung/forschungsfelder/detail/ffeld/sichere-und-intelligente-fahrzeuge/ (26.03.2020)

⁷ Pionierleistung: Autonome Langstreckenfahrt im Überland- und Stadtverkehr: Mercedes-Benz S-Klasse INTELLIGENT DRIVE fährt autonom auf den Spuren von Bertha Benz - Daimler Global Media Site: media.daimler.com/marsMediaSite/de/instance/ko/Pionierleistung-Autonome-Langstreckenfahrt-im-Ueberland--und-Stadtverkehr-Mercedes-Benz-S-Klasse-INTELLIGENT-DRIVE-faehrt-autonom-auf-den-Spuren-von-Bertha-Benz.xhtml?oid=9904223 (26.03.2020)

durchgeführt. 2014 folgte ein autonomer Lastwagen („Future Truck“⁸), 2017 ein autonom fahrender Bus („Future Bus“⁹) sowie zusammen mit Bosch ein System für automatisiertes Parken¹⁰. Aktuell forschen die beiden Firmen an Lösungen für das automatisierte Fahren¹¹. Auch der Autohersteller Audi zeigte z. B. 2015 pilotiertes Fahren auf der Autobahn mit dem Forschungsfahrzeug „Jack“ oder eine Teilnahme an einem Bergrennen auf den Pikes Peak im Jahr 2010 mit dem fahrerlosen Fahrzeug „Shelley“¹². Volvo hat mit dem Projekt „Drive Me“ autonomes Fahren für ausgewählte Testkunden in Schweden angekündigt¹³ und Tesla plant automatisierte Fahrfunktionen nicht nur auf der Autobahn sondern auch in Innenstädten anzubieten¹⁴.

In den letzten Jahren sind auch immer mehr Technologiekonzerne in den Bereich des automatisierten Fahrens eingestiegen. So forscht z. B. Google seit 2009 im Rahmen des Google Self-Driving Car Project¹⁵ (seit Ende 2016 Waymo¹⁶) an autonomen Fahrzeugen. Auch der Fahrdienstanbieter Uber¹⁷ und Apple sind wie viele andere Firmen auch in den Markt eingestiegen¹⁸.

2.2.1 Relevante Projekte

Aufgrund der Vielzahl an Projekten und Unternehmungen im Bereich autonomes Fahren kann hier nur eine kleine Auswahl beleuchtet werden. Im Folgenden liegt der Fokus daher auf Projekten, die stark zur wissenschaftlichen und technischen Ausgangsbasis dieser Arbeit beigetragen haben.

Schon Ende des letzten Jahrhunderts konnten erste automatisierten Fahrfunktionen wie das Halten der eigenen Fahrspur und Geschwindigkeit sowie automatisierte Spurwechsel im Rahmen des Projekts PROMETHEUS gezeigt werden [DGN87]. Größere Sichtbarkeit erreichte das Thema jedoch mit den so-

⁸ Mercedes-Benz Future Truck 2025: www.daimler.com/innovation/autonomes-fahren/mercedes-benz-future-truck.html (26.03.2020)

⁹ Der Mercedes-Benz Future Bus: www.daimler.com/innovation/autonomes-fahren/future-bus.html (26.03.2020)

¹⁰ Automated Valet Parking – schnell, sicher, fahrerlos | Bosch Global: www.bosch.com/de/stories/automated-valet-parking (26.03.2020)

¹¹ Daimler und Bosch: Start des San José Pilotprojektes für automatisierten Mitfahrerservice: www.daimler.com/innovation/case/autonomous/pilotstadt-san-jose.html (26.03.2020)

¹² Das pilotierte Fahren | Audi MediaCenter: www.audi-mediacenter.com/de/audi-auf-der-ces-2016-5294/das-pilotierte-fahren-5303 (26.03.2020)

¹³ Schwedische Familien unterstützen Volvo: Autonomes Fahren im Rahmen des Drive Me Projekts - Volvo Car Austria Pressezentrum: www.media.volvocars.com/at/de-at/media/pressreleases/217555/schwedische-familien-unterstutzen-volvo-autonomes-fahren-im-rahmen-des-drive-me-projekts (26.03.2020)

¹⁴ Model 3 | Tesla: www.tesla.com/de_de/model3 (26.03.2020)

¹⁵ Google Self-Driving Car Project: www.google.com/selfdrivingcar/how (3.6.2016)

¹⁶ Waymo: waymo.com (26.03.2020)

¹⁷ Advanced Technologies Group | Uber ATG: www.uber.com/de/de/atg/ (26.03.2020)

¹⁸ Autonomous Vehicle Disengagement Reports 2018: www.dmv.ca.gov/portal/dmv/detail/vr/autonomous/disengagement_report_2018 (26.03.2020)

genannten Challenges der Defense Advanced Research Projects Agency des US-amerikanischen Verteidigungsministeriums (DARPA). In der „Grand Challenge“ im Jahr 2004 und 2005 sollten autonome Fahrzeuge auf einer Strecke von 150 bzw. 132 Meilen durch die Mojave-Wüste fahren. Während 2004 noch kein Team die Strecke bewältigen konnte, erreichten 2005 fünf Teams das Ziel. Im Jahr 2007 lag der Fokus der „Urban Challenge“ auf der Fahrt in einem innerstädtischen Szenario. Die Fahrzeuge mussten bei ihrer Fahrt über einen 60 Meilen langen Kurs z. B. an Kreuzungen abbiegen, Vorfahrtsregeln beachten und auf der Straße wenden können. In der „Urban Challenge“ konnte sich das Team „AnnieWAY“ das aus dem Sonderforschungsbereich/Transregio 28 Kognitive Automobile (SFB/TR 28)¹⁹ hervorgegangen war, mit dem gleichnamigen Fahrzeug als eines von zwei nicht amerikanischen Teams für das Finale qualifizieren.²⁰

Das Fahrzeug „AnnieWAY“ kam auch in der Vorbereitung des Projekts „S 500 INTELLIGENT DRIVE“ zum Einsatz.²¹ Im Rahmen dieses Projekts wurde im September 2013 eine automatisierte Langstreckenfahrt mit einer 2014er Mercedes S-Klasse durchgeführt. Das Fahrzeug war ausgestattet mit seriennaher Sensorik und ausgerüstet für automatisiertes Fahren auf öffentlichen Straßen. Ziel war es, die historische Route, die Berta Benz 1888 mit dem Modell Nummer 3 des Patent-Motorwagens ihres Mannes Carl Benz zurücklegte, vollständig automatisiert zu fahren²². Die über 100 km lange Route führt von Mannheim über Heidelberg, Bruchsal und Durlach nach Pforzheim durch mehr als 20 Städte und Ortschaften. Die in diesem Projekte eingeführte Kartenrepräsentation wird auch in dieser Arbeit als Basis der geometrischen Umfeldrepräsentation verwendet (siehe Kapitel 2.4.4).

Aufbauend auf den Erkenntnissen und Algorithmen aus dem Projekt „S 500 INTELLIGENT DRIVE“ nahm auch bei der Grand Cooperative Driving Challenge (GCDC)²³ 2016 das team „AnnieWAY“ erfolgreich teil [TSP+18]. Die GCDC ist ein Wettbewerb, an dem europäische Teams mit vernetzten und automatisiert fahrenden Fahrzeugen miteinander und gegeneinander kooperative Fahrmanöver durchführen. Im Jahr 2016 sollte ein Einfädelszenario auf einer Autobahn sowie ein effizientes Passieren einer T-Kreuzung gezeigt werden. Alle Fahrzeuge kommunizierten über spezielle C2C-Protokolle (engl. car to car) miteinander um z. B. ihre genaue Position und Geschwindigkeit anderen Fahrzeugen zur Verfügung zu stellen. Die hierzu verwendete Technik kommt auch in der Evaluation dieser Arbeit zum Einsatz.

¹⁹ KIT - IES - Forschung - Verteilte Kooperation kognitiver Automobile: ies.anthropomatik.kit.edu/forschung_verteilte_kooperation.php (26.03.2020)

²⁰ MRT Institut für Mess- und Regelungstechnik: www.mrt.kit.edu/annieway/uc07.html (26.03.2020)

²¹ KIT - Das KIT - Medien - Presseinformationen - PI Archiv - PI 2013: www.kit.edu/kit/pi_2013_13901.php (26.03.2020)

²² Bertha Benz Memorial Route: www.bertha-benz.de (26.03.2020)

²³ Grand Cooperative Driving Challenge 2016: gcdc.net (27.05.2018)

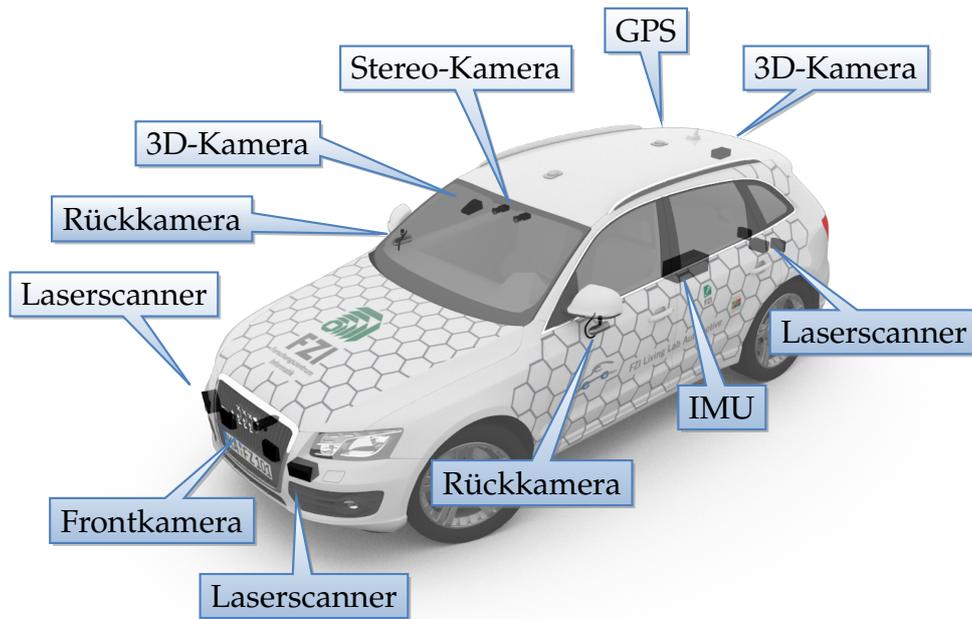


Abbildung 2.2: CoCar – Sensorsetup des Forschungsfahrzeugs des FZI. Das Fahrzeug wurde im Rahmen dieser Arbeit ausgebaut und als Evaluationsplattform eingesetzt.

2.2.2 Forschungsfahrzeug „CoCar“

Im Jahr 2010 wurde am FZI das Forschungsfahrzeug CoCar – Cognitive Car²⁴ aufgebaut. Dabei konnte auf die Erfahrungen aus dem SFB/TR 28 und mit dem Forschungsfahrzeug „AnnieWAY“ zurückgegriffen werden. So konnte z. B. das Steuerungs- und Regelungskonzept mittels einer dedizierten echtzeitfähigen Hardware übernommen werden.

Das Forschungsfahrzeug CoCar basiert auf einem Audi Q5, der mit Aktuatorik zum automatisierten Fahren ausgerüstet wurde. Für die Wahrnehmung der Umgebung und des eigenen Zustands verfügt es über eine Vielzahl teils redundant ausgelegter Sensoren (siehe Abbildung 2.2):

Videokameras

Am Fahrzeug z. B. an den Außenspiegeln und im Kühlergrill sowie im Fahrzeuginnenraum z. B. hinter der Windschutzscheibe sind mehrere Videokameras mit unterschiedlichen Brennweiten und Sensorgrößen angebracht. Bei Bedarf können weitere Kameras flexibel am Fahrzeug angebracht werden.

Stereokamera

Ein Stereokamerapaar ist zentral hinter der Windschutzscheibe angebracht. Die notwendige Datenverarbeitung zur Entfernungsberechnung ist auf einem FPGA-Board im Kofferraum realisiert.

²⁴ FZI Forschungszentrum Informatik: CoCar: www.fzi.de/projekt/cocar (26.03.2020)

3D-Kamera

Hinter der Frontscheibe und hinter der Rückscheibe ist jeweils eine 3D-Kamera angebracht. Diese basieren auf dem Licht-Laufzeit-Verfahren (engl.: time of flight, TOF). Infrarotlicht-Einheiten z. B. in der Stoßstange senden dazu moduliertes Licht in die Szene. Die hier eingesetzten Kameras sind sogenannte photonic mixing devices (PMDs) und nehmen das Fahrzeugumfeld mit einer Auflösung von 64x16 Pixeln auf.

Laserscanner

Das Fahrzeug ist mit mehreren Laserscannern ausgestattet. Rund um das Fahrzeug sind fünf IBEO Lux²⁵ Sensoren angebracht. Diese tasten auf jeweils vier Ebenen das Fahrzeugumfeld ab. Somit ergibt sich eine 360° Rundumsicht um das Fahrzeug. Eine Fusionseinheit im Kofferraum des Fahrzeugs berechnet aus den 3D-Punktwolken Objekthypothesen mit Geschwindigkeit und Objektklasse. Zusätzlich können auf dem Dach des Fahrzeugs verschiedene Sensoren wie ein Velodyne HDL-32E²⁶ angebracht werden, der das Fahrzeugumfeld (360°) mit 32 Zeilen abtastet.

GPS und IMU

Um eine hochgenaue globale Positions-, Orientierungs- und Geschwindigkeitsbestimmung zu ermöglichen werden ein Globales Positionsbestimmungssystem (engl.: global positioning system, GPS) und eine inertielle Messeinheit (engl.: inertial measurement unit, IMU) in Kombination genutzt. Bei guten Bedingungen wird eine Positionsgenauigkeit von unter 2 cm erreicht.

CoCar wird in verschiedenen Projekten im Bereich des assistierten und automatisierten Fahrens eingesetzt: z. B. zur Steigerung der Energieeffizienz [KSN+11], zur Fahrerbeobachtung und -modellierung [BNK+11] oder auch zur Entwicklung von Augmented-Reality-Methoden [ZKB+14] zur Verschmelzung von Simulation und realer Testfahrt. Somit stehen verschiedene Basisfunktionalitäten zur Datenaufzeichnung, Objektdetektion und Trajektorienplanung zur Verfügung, z. B. Algorithmen zur Ampelerkennung [WWZ16], Verkehrsszenenprädiktion [KKS+15] und zur Trajektorienplanung [KOH+15]. Auch viele der im Rahmen dieser Arbeit entwickelten Methoden und Verfahren wurde auf dem Forschungsfahrzeug CoCar integriert und evaluiert.

2.3 Verkehrsszenenrepräsentationen

In der Verkehrsszenenrepräsentation wird häufig zwischen der Repräsentation des Zustands des Ego-Fahrzeugs **ego** und der Darstellung der Fahrzeugumgebung unterschieden. Für das Ego-Fahrzeug werden dabei häufig komplexere Modelle verwendet. Dies liegt zum einen daran, dass mehr Informationen zum eigenen

²⁵ Ibeo.Reference: www.ibeo-as.com/ibeoreference/reference-sensor-system (26.03.2020)

²⁶ HDL-32E | Velodyne Lidar: velodynelidar.com/hdl-32e.html (26.03.2020)

Fahrzeug zur Verfügung stehen und zum anderen genauere Modelle für die Aktionsentscheidung benötigt werden, damit z. B. Zielgrößen wie Energieverbrauch und Fahrkomfort berücksichtigt werden können. Bei der Repräsentation der Umgebung kommen je nach Anwendung unterschiedliche Modelle mit verschiedenen Abstraktionsgraden von sensordatennahen, subsymbolischen Modellen mit wenig bis keiner semantischen Interpretation über geometrische Modelle bis hin zu rein semantischen Modellen zum Einsatz.

2.3.1 Fahrzeugmodelle

Abhängig von der Anwendung können Fahrzeugmodelle verschiedene Aspekte eines realen Fahrzeug auf unterschiedlichen Abstraktionsebenen abbilden. Für die kinematische Betrachtung kann z. B. ein einfaches Modell aus 2D-Pose (x, y, ϕ) , Geschwindigkeit in Fahrtrichtung v sowie Gierrate $\dot{\phi}$ verwendet werden. Gegenüber einem realen Fahrzeug sind hier jedoch Drehungen auf der Stelle oder mit sehr kleinem Wendekreis möglich.

Soll dies ausgeschlossen werden, kann das Einspurmodell (bzw. Ackermannmodell) [RS40] verwendet werden. Hierbei wird neben der Fahrzeugpose und der Geschwindigkeit der Lenkwinkel modelliert. Das Fahrzeug wird dabei als Zweirad mit einer starren Hinter- und einer gelenkten Vorderachse modelliert.

Zur Modellierung der Fahrdynamik gibt es ebenfalls verschiedene Abstraktionsmöglichkeiten von der Modellierung der reinen Längsdynamik bis hin zu komplexeren Modellen bei denen Längs- und Querdynamik zusammen berücksichtigt werden (z. B. Kammscher Reibkreis). Im Rahmen dieser Arbeit stehen Modelle zur Längsdynamik im Fokus. Ein häufig verwendetes Modell ist hierbei das Punktmassenmodell, bei dem das Fahrzeug durch einen Punkt mit der Masse m repräsentiert wird, auf den eine resultierende Kraft F wirkt. Entsprechend der Newtonschen Bewegungsgleichung wirken sich diese dann auf die Beschleunigung a des Punkts aus.

$$\ddot{x} = \dot{v} = a = m \cdot F \quad (2.1)$$

Die Kraft F setzt sich dabei aus verschiedenen Kräften zusammen. Einfluss auf die Bewegung eines Fahrzeugs haben die Fahrwiderstände sowie die von Motor und Bremse auf die Räder wirkenden Drehmomente [BKZ+11]: Auf ein ungebremstes Fahrzeug mit ausgekuppeltem Motor wirken lediglich die Fahrwiderstände und verzögern dieses auf einer ebenen Fahrbahn bis zum Stillstand. Zu den Fahrwiderständen gehören [GU10]:

Luftwiderstand F_{Luft}

Der Luftwiderstand ist abhängig von der Fahrzeuggeschwindigkeit, der Windgeschwindigkeit, der Stirnfläche des Fahrzeugs sowie seiner Geometrie

2 Grundlagen und Stand der Technik

(c_w -Wert). Mit zunehmender Geschwindigkeit nimmt der Luftwiderstand überproportional zu.

Radwiderstände F_{Rad}

Der Radwiderstand fasst Widerstände, die am Rad entstehen zusammen. Er beinhaltet Rollwiderstände wie Walkwiderstand und Reibwiderstand sowie Effekte der Fahrbahn z. B. verursacht durch Schräglauf des Rads.

Steigungswiderstand F_{Steig}

Der Steigungswiderstand beschreibt die Kraft, die notwendig ist, um das Fahrzeug eine Steigung entgegen der Erdbeschleunigung g hinauf zu bewegen. Er ist abhängig vom Steigungswinkel α und der Masse m des Fahrzeugs: $F_{\text{Steig}} = \sin \alpha \cdot m \cdot g$. Bei negativer Steigung kann der Steigungswiderstand entgegen seinem Namen auch zu einer Beschleunigung des Fahrzeugs führen.

Außer den Fahrwiderständen wirken auch Kräfte aus dem Motor- und Bremssystem des Fahrzeugs auf dieses.

Antriebskraft F_{Antrieb}

Die Antriebskraft ist die vom Motor über Getriebe und Räder aufgebrachte Kraft auf das Fahrzeug. Sie ist maßgeblich für das Beschleunigen und Fahren eines Fahrzeugs verantwortlich.

Bremskraft F_{Brems}

Die Bremskraft wirkt entgegen der Bewegungsrichtung und somit in der Regel der Antriebskraft entgegen. Sie wird von den Bremsen über die Räder übertragen und führt zu einer Verzögerung des Fahrzeugs. Die dabei abgebaute kinetische Energie des Fahrzeugs wird in den Bremsen in Wärme umgewandelt und geht somit für den Antrieb des Fahrzeugs verloren.

Motor Schleppmoment F_{Schlepp}

Ist der Motor eingekuppelt, wird jedoch kein Gas gegeben, so wird der Motor vom fahrenden Fahrzeug in Bewegung gehalten (Schubbetrieb). Durch Reibung entstehende Kräfte wirken verzögernd auf das Fahrzeug.

Rekuperationsmoment F_{Rekup}

Bei elektrischen Antrieben kann das Schleppmoment des Motors künstlich erhöht werden. Dabei wird die zusätzlich abgebaute, kinetische Energie in elektrische Energie umgewandelt und kann in der Batterie gespeichert werden (Rekuperation).

Während sich der Steigungswiderstand gut aus der Fahrzeugmasse und der Streckentopologie berechnen lässt, sind die anderen Fahrwiderstände komplexer zu modellieren. Sie können allerdings unter gewissen Randbedingungen in Summe mit einem sogenannten Coast-Down-Test ermittelt werden. Dabei wird das Fahrzeug auf einer ebenen Fläche ausrollen gelassen und die Beschleunigung in Abhängigkeit der Geschwindigkeit bestimmt.

Für ein konventionell betriebenes Fahrzeug werden in [RSK+11] verschiedene Betriebsmodi vorgestellt. Zusammen mit einem Betriebsmodus zur Rekuperation werden diese für die Wahl einer optimalen Fahrstrategie herangezogen.

Konstante Geschwindigkeit

Fahren mit konstanter Geschwindigkeit. Dabei wird der Motor, z. B. ein Verbrennungsmotor (engl.: internal combustion engine, ICE) oder ein Elektromotor verwendet.

Ausrollen (coasting)

Ausrollen mit ausgekuppeltem Motor. Der Motor läuft im Leerlauf. Mit Start-Stopp-Automatik wäre auch eine Abschaltung des Motors denkbar.

Schubbetrieb (shear)

Ausrollen mit eingekuppeltem Motor. Der Motor läuft im Schubbetrieb mit abgeschalteter Kraftstoffzufuhr.

Rekuperation

Ausrollen mit eingekuppeltem Elektromotor. Der Motor wird als Generator betrieben und wandelt kinetische Energie in elektrische um.

Bremsen

Verzögerung durch die Reibbremse. Der Motor läuft mit abgeschalteter Kraftstoffzufuhr im Schubbetrieb.

Stillstand

Das Fahrzeug steht still. Der Motor wird über die Start-Stopp-Automatik abgeschaltet.

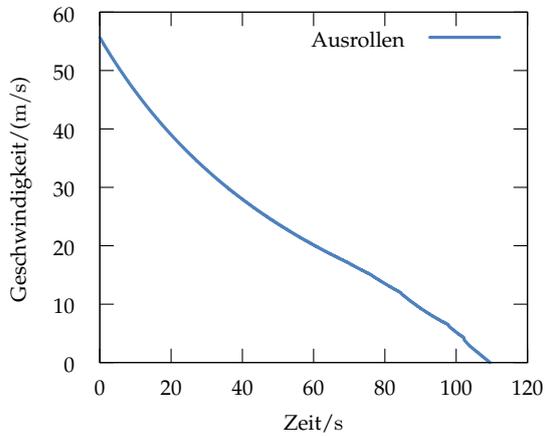
Beschleunigung

Das Fahrzeug beschleunigt mit Hilfe des Motors. Es wird eine durchschnittliche ökonomische Beschleunigung angenommen.

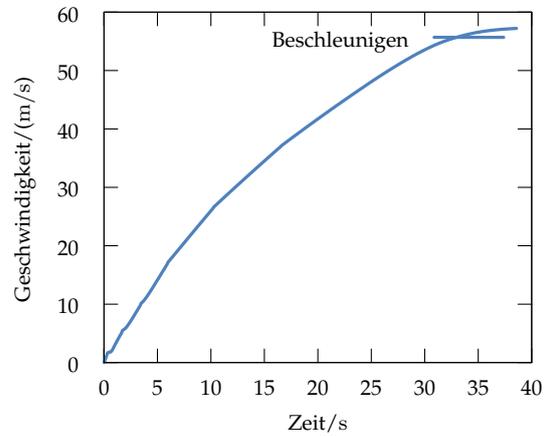
Für ein Fahrzeug kann auf Basis der verschiedenen Betriebsmodi ein Dynamik- und Verbrauchsmodell aufgestellt werden [KSN+11]. Zur Bildung des Dynamikmodells wird pro verzögerndem Betriebsmodus ein Coast-Down-Test durchgeführt. Dabei wird das Fahrzeug im entsprechenden Betriebsmodus (Ausrollen, Schubbetrieb, Rekuperation) von der maximal betrachteten Geschwindigkeit ausrollen gelassen. Die Geschwindigkeit $v(t)$ und zurückgelegte Strecke $s(t)$ wird über der Zeit aufgezeichnet (siehe Abbildung 2.3a). Daraus lässt sich die jeweilige Beschleunigung in Abhängigkeit der Geschwindigkeit herleiten. Zur Modellierung der Beschleunigung kann ähnlich verfahren werden, jedoch wird hierbei das Fahrzeug von 0 km/h auf die gewünschte Höchstgeschwindigkeit beschleunigt (Abbildung 2.3b). Für die Bremsung wird eine konstante Beschleunigung angenommen, da diese im Regelfall von einem Fahrzeug unabhängig von anderen Fahrwiderständen eingeregelt werden kann.

Ähnlich wie der Geschwindigkeitsverlauf kann auch der Verbrauch eines Fahrzeugs für die verschiedenen Betriebsmodi gemessen und für die Modellierung bereitgestellt werden. Abbildung 2.3c zeigt z. B. den Verbrauch eines Fahrzeugs in

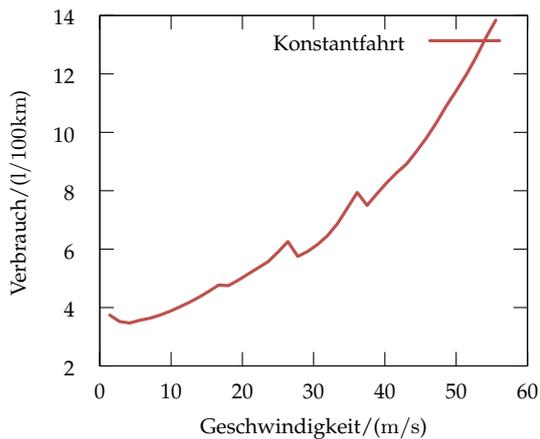
2 Grundlagen und Stand der Technik



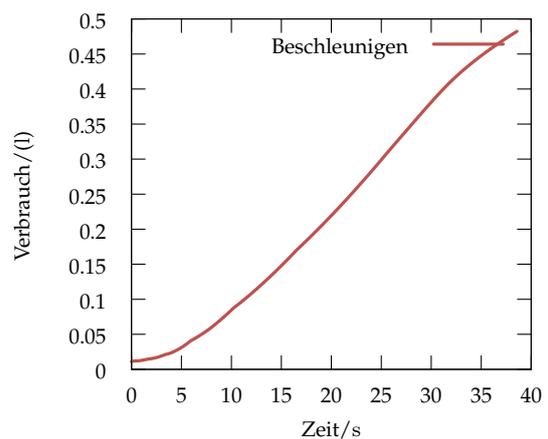
(a) Coast-Down-Test: 200 km/h auf 0 km/h



(b) Beschleunigung: 0 km/h auf 200 km/h



(c) Treibstoffverbrauch bei Konstantfahrt mit unterschiedlichen Geschwindigkeiten



(d) Treibstoffverbrauch bei Beschleunigung von 0 km/h auf 200 km/h

Abbildung 2.3: Messungen eines Fahrzeugs zur Bildung eines Dynamik- und Verbrauchsmodells

Ähnlich veröffentlicht in [KSN+11] (Kohlhaas et al.)

Litern pro 100 km mit Verbrennungsmotor bei Konstantfahrt mit unterschiedlichen Geschwindigkeiten.

Die gemessenen Werte können in einem Fahrzeugmodell für die folgende Optimierung zusammengefasst werden. Als Grundlage der Dynamikberechnungen in dieser Arbeit werden zwei verschiedene Modelle aus den Daten abgeleitet und verwendet: Ein einfaches, schnell berechenbares für ebene Strecken, das Berechnung von Geschwindigkeitsprofilen und zugehörigen Verbräuchen ermöglicht sowie ein komplexeres, iteratives Verfahren, das die Bewegung des Fahrzeugs anhand der ermittelten Beschleunigungen auch unter Berücksichtigung der Streckentopologie simuliert.

Das erste Berechnungsmodell verwendet die aufgezeichneten Funktionen $v(t)$ und $s(t)$, die für jeden Betriebsmodus aufgezeichnet wurden. Soll nun das Verhalten des Fahrzeugs mit einer gegebenen Geschwindigkeit v_0 und einem gegebenen Betriebsmodus für eine gewisse Zeit Δt ermittelt werden, wird wie folgt vorgegangen: Sei t_i der Zeitpunkt der Aufzeichnung mit $v(t_i) = v_0$ und $t_j = t_i + \Delta t$. Dann ist $\Delta s = s(t_j) - s(t_i)$ die in Δt zurückgelegte Strecke. Da der Geschwindigkeitsverlauf und Positionsverlauf nur diskret aufgezeichnet werden kann, muss ggf. zwischen benachbarten Werten interpoliert werden. Analog kann auch der Verbrauch für diese Aktion berechnet werden.

Soll auch das Höhenprofil der Strecke berücksichtigt werden, so kann die Bewegung des Fahrzeugs iterativ berechnet werden als:

$$s(t) = s(t - \Delta t) + v(t - \Delta t) \cdot \Delta t + \frac{1}{2}a(t - 1) \cdot \Delta t^2 \quad (2.2)$$

Wobei sich a aus den verschiedenen Fahrwiderständen und Momenten zusammensetzt:

$$a = m \cdot (F_{\text{Luft}} + F_{\text{Rad}} + F_{\text{Steig}} + F_{\text{Motor}} + F_{\text{Brems}}) \quad (2.3)$$

$$= a_{\text{Luft}}(v) + a_{\text{Rad}}(v) + a_{\text{Steig}} + a_{\text{Motor}}(v) + a_{\text{Brems}} \quad (2.4)$$

Wobei F_{Motor} je nach Betriebsmodus F_{Antrieb} , F_{Schub} oder F_{Rekup} entspricht und für jede der Kräfte $a_{\dots} = m \cdot F_{\dots}$ gilt. Dabei lässt sich $a_{\text{Betrieb}}(v) = a_{\text{Luft}}(v) + a_{\text{Rad}}(v) + a_{\text{Motor}}(v)$ direkt aus einem Coast-Down-Test herleiten und a_{Steig} analytisch aus dem Steigungsprofil der Strecke berechnen. Dies ermöglicht eine einfache Implementierung dieses Modells durch eine tabellarische Darstellung gemessener $a_{\text{Betrieb}}(v)$ und einer streckenabhängigen Berechnung von a_{Steig} .

$$a = a_{\text{Betrieb}}(v) + a_{\text{Steig}} \quad (2.5)$$

Dynamikmodell

Entscheidend für die Übertragung des Raddrehmoments auf eine Antriebskraft des Fahrzeugs ist der Reibungskoeffizient oder auch maximaler Kraftschlussbeiwert μ_H . Er gibt das Verhältnis aus der über die Reifen übertragbaren Kraft und

der Anpresskraft (oder auch Gewichtskraft unter Vernachlässigung aerodynamischer Effekte) des Fahrzeugs an. Der Reibungskoeffizient für Gummireifen auf trockenem Asphalt liegt in der Regel zwischen 0,5 und 1. Somit ist die Beschleunigung des Fahrzeugs (ohne Ausnutzung von aerodynamischen Effekten) auf ebener Strecke auf der Erde auf einen Wert kleiner $g \approx 9,81 \text{ m/s}^2$ (Erdbeschleunigung) beschränkt. Hieraus ergeben sich zum einen eine Beschränkung der Längsbeschleunigung wie auch der Querschleunigung und somit eine radiusabhängige maximale Durchfahrtgeschwindigkeit in Kurven.

Um komfortabel zu fahren, muss die maximale Beschleunigung jedoch deutlich niedriger liegen. Beschränkt wird die Längsbeschleunigung auch durch die Motorleistung bzw. das Motordrehmoment. Aus der Straßenverkehrsordnung, den lokalen Vorgaben (z. B. Geschwindigkeitsbeschränkungen) sowie den Fahrerwünschen ergeben sich weitere Beschränkungen der Geschwindigkeit.

2.3.2 Umgebungsmodelle

Zur Darstellung des Fahrzeugumfelds werden verschiedene Modelle unterschieden. Im Folgenden wird eine Klassifikation anhand der verschiedenen Abstraktionsgrade in drei Kategorien vorgeschlagen: subsymbolische Modelle, geometrische Modelle und semantische Modelle.

Subsymbolische Modelle

Subsymbolische Modelle stellen eine Repräsentation der Umgebung dar, die wenig bis gar nicht von den verwendeten Sensordaten abstrahiert. Es werden keine expliziten Objektinformationen gespeichert. Beispiele hierfür sind gitterbasierte Karten (grid maps) oder sogenannte Belegtheitsgitter (occupancy grids) [BGV+09], [GBS+09]. Sie bestehen aus meist äquidistanten Zellen, die über einen oder mehrere Belegtheitswerte die Existenz eines Hindernisses oder anderer Strukturen an dieser Stelle des Umgebungsmodells kodieren. Dabei kann ein Objekt mehrere Zellen belegen oder es können auch mehrere Objekte in einer Zelle zu liegen kommen. Die Zellen können auch entlang geometrischer Grundstrukturen wie Fahrspuren angeordnet sein [STT+13].

Die Belegtheit kann binär oder in Form einer Wahrscheinlichkeit angegeben sein. Auch Erweiterungen, die einzelne Zellen z. B. mit Geschwindigkeiten oder Klassen attribuieren, sind möglich. Belegtheitsgitter sind eine sensornahe Repräsentationsform, die in der Regel wenig Semantik enthält. Vorteile sind der durch die äquidistante Diskretisierung konstante Speicherbedarf sowie die konstante Laufzeit- und Speicherkomplexität vieler darauf angewendeter Algorithmen.

Gitter werde auch zur Repräsentation von statischen Karten eingesetzt. In ihnen können Hindernisse wie Wände oder auch andere Informationen wie die Position

von Fahrspurmarkierungen dargestellt werden. Diese können dann zur Planung oder auch zur Lokalisierung mit Hilfe aktueller Sensordaten verwendet werden.

Eine effiziente Darstellung wird über sogenannte Quadrees im zweidimensionalen bzw. Octrees im dreidimensionalen Raum erreicht. Für Lokalisierungsaufgaben und zur schnelleren Abstandsberechnung, z. B. zur Kollisionserkennung, kann zusätzlich zur Belegtheit auch die Entfernung und die Richtung zum nächstgelegenen Objekt gespeichert werden [KOH+15]. Dies erlaubt z. B. bei der Lokalisierung eine schnelle Karteneinpassung von Sensordaten und die Unterscheidung verschiedener Seiten eines Objektes, was z. B. bei der Kartierung von Räumlichkeiten mit dünnen Wänden die Kartenqualität verbessert.

Geometrische Modelle

Geometrische Modelle bilden die Umgebung durch geometrische Objekte unterschiedlicher Dimensionalität z. B. Punkte, Linien, Flächen oder Körper ab. Es kommen zweidimensionale Kartendarstellungen oder auch dreidimensionale Repräsentationen zum Einsatz. Häufig wird zwischen statischer Infrastruktur, die ggf. schon a priori bekannt ist, und dynamischen Objekten, die z. B. von der Sensorik des Fahrzeugs erkannt werden, unterschieden. Wird die Umgebung in Fahrzeugkoordinaten repräsentiert, können auch statische Objekte eine dynamische Repräsentation aufweisen.

Bei der Repräsentation von Straßenszenarien kommen Darstellungen auf verschiedenen Abstraktionsniveaus zum Einsatz. So können z. B. einzelne Objekte und Umgebungseigenschaften wie Fahrspurmarkierungen, Leitpfosten und Fahrbahnbereiche als geometrische Strukturen abgebildet werden. Dies erlaubt eine detaillierte Darstellung der Umgebung, wie sie z. B. bei der Lokalisierung auf Basis von Sensordaten eingesetzt werden kann.

Da Straßenmarkierungen häufig Linien und Flächen beschreiben, ist es je nach Anwendung sinnvoll, von der genauen geometrischen Ausdehnung zu abstrahieren und diese nur als geometrische Linien und Flächen zu repräsentieren. Um die Geometrie von Straßenmarkierungen abzubilden, können z. B. einfache Polygonzüge, mathematische Kurven wie Clothoide oder sogenannte arc splines [SMJ12] verwendet werden. Arc splines sind Polygonzüge (splines) die stückweise aus Kreisbogensegmenten und geraden Streckensegmenten zusammengesetzt sind. Diese geometrische Darstellung hat den Vorteil, dass sich viele in Lokalisierungs- und Planungsaufgaben benötigte Operationen wie Abstände und Segmentlängen effizient berechnen lassen. So ist beispielsweise die Offsetkurve eines arc splines, also eine Kurve mit konstantem Abstand zu diesem, wieder ein arc spline.

Eine stärkere Abstraktion stellt Fahrspuren oder auch ganze Straßen als geometrische Objekte in Form von Linien dar. Im so genannten Roadgraph [HW11], [KH10] werden Fahrspuren als Kanten des Graphs dargestellt. Der Graph kann ein gesamtes Straßennetz repräsentieren und bildet somit eine Form einer Straßenkarte. Auf

2 Grundlagen und Stand der Technik

weitere Darstellungsmöglichkeiten von Straßenkarten wird in Kapitel 2.4 genauer eingegangen.

Bewegliche Objekte wie Fahrzeuge oder Fußgänger werden häufig in Objektlisten dargestellt. Sie werden somit je nach betrachteter Dimensionalität als Punkt, Pose, Fläche oder Körper repräsentiert. Jedes Objekt kann dabei gewisse Eigenschaften wie Position, Ausrichtung, Geschwindigkeit, Form und Ausdehnung aufweisen. Auch zusätzliche semantische Eigenschaften wie eine Objektklasse können den Objekten zugeordnet werden.

Je nach verwendetem Bezugssystem und abhängig von der Anwendung können Objektpositionen, -ausrichtungen und -geschwindigkeiten relativ zum Fahrzeug oder zu einem ortsfesten Koordinatensystem (kartesisch oder auch in Geokoordinaten) sowie in Relation zu Infrastrukturelementen wie dem Fahrspurverlauf dargestellt werden. In Kombination mit dem oben erwähnten Roadgraph werden Objektpositionen in Relation zu einer Kante als Offset zu dieser angegeben. So kann auch die Bewegung eines Objekts einfach entlang einer Kante des Roadgraphs also entlang der gefahrenen Fahrspur repräsentiert werden, was bei Fahrzeugen z. B. die Güte einer Bewegungsprädiktion im Allgemeinen verbessert.

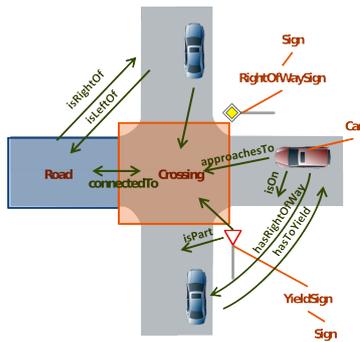
Semantische Modelle

Semantisches Wissen über Verkehrssituationen wird häufig mittels Ontologien modelliert. Eine Ontologie ist eine formale, explizite Spezifikation einer gemeinsam genutzten Konzeptualisierung. Konzeptualisierung ist dabei ein abstraktes Modell eines Phänomens in der Welt, das formal also maschinenlesbar und explizit definiert ist. Die Gültigkeit einer Ontologie sollte nicht nur individuell sondern zumindest innerhalb einer größeren Gruppe z. B. einer Fachdomäne akzeptiert sein. [SBF98]

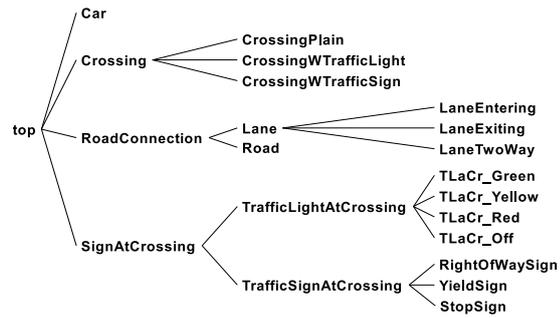
Entsprechend ihrer Anwendungen können Ontologien in Domänen- bzw. Applikationsontologien eingeteilt werden. Domänenontologien beschreiben dabei die Klassen und Relationen einer bestimmten Domäne. So beinhaltet eine Ontologie zur Beschreibung von Straßenverkehrsszenen Elemente wie Fahrzeuge, Fahrstreifen, Verkehrsschilder aber auch Verkehrsregeln. Applikationsontologien sind hingegen stärker auf die eigentliche Anwendung zugeschnitten. Eine Ontologie zur Verhaltensentscheidung im Straßenverkehr könnte auch Elemente wie Stoppen an einer Kreuzung oder Vorfahrt Gewähren beinhalten.

Abbildung 2.4 zeigt beispielhaft eine Applikationsontologie für ein Fahrerassistenzsystem an Kreuzungen [HZW11]. Entsprechend der Anwendung liegt der Fokus der Modellierung stark auf den Verkehrsregeln an der Kreuzung (z.B. *hasRightOfWay*) und den dafür erforderlichen Entitäten (z. B. **TrafficLightAtCrossing**, **YieldSign**) und geometrischen Relationen (*isRightOf*).

Beim Aufbau von Ontologien unterscheidet man zwei Konstrukte [AFIG14]: Terminological box (TBox) und assertional box (ABox). Die TBox beschreibt dabei



(a) Beispiel einer Kreuzungsszene mit Objektklassen und Relationen



(b) Zugrundeliegende Ontologie (terminological box (TBox)). Dargestellt sind lediglich die taxonomischen Relationen.

Abbildung 2.4: Applikationsontologie einer Kreuzung für ein Fahrerassistenzsystem

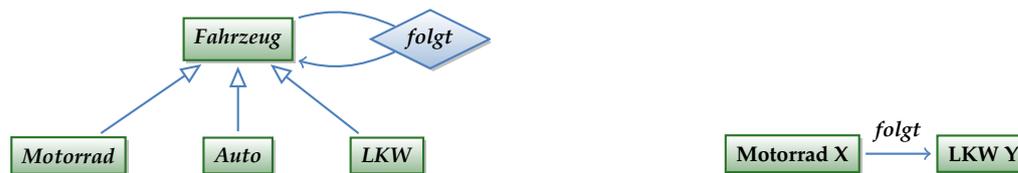
Bildquelle: [HZW11] © 2011 IEEE

die Konzepte innerhalb der Ontologie als Klassen. Klassen können bestimmte Attribute haben. Des Weiteren werden Relationen zwischen den Klassen definiert. Es gibt zwei Formen von Relationen: Eine ist die taxonomische Relation *ist ein*, die die Klassenhierarchie beschreibt. Attribute werden entsprechend der Hierarchie vererbt. Eine zweite Form sind die Attribut-Relationen zwischen zwei Klassen. Diese Relationen werden wie Attribute auf Kindklassen vererbt. In der ABox werden Entitäten als Instanzen der Klassen dargestellt. Hierbei können z. B. Daten von realen Experimenten oder Sensoren als Attribute und Relationen einfließen.

In Abbildung 2.5b ist eine Ontologie zur Beschreibung von einfachen Verkehrssituationen dargestellt. Links ist die TBox visualisiert. Sie beschränkt sich auf die Klassen *Fahrzeug*, *Auto*, *LKW* sowie *Motorrad*. Dabei ist *Fahrzeug* die Elternklasse. Auf dieser ist eine weitere Relation *folgt* definiert, die den Sachverhalt ausdrückt, dass ein Fahrzeug einem anderen folgt. Auf der rechten Seite ist eine zugehörige Instanziierung (ABox) dargestellt. Hier sind zwei Fahrzeuge, ein Motorrad (**Motorrad X**) sowie ein LKW (**LKW Y**) dargestellt. Die Relation *folgt* kann nun auf Instanzen der Kindklassen angewendet werden und somit den Sachverhalt ausdrücken, dass das Motorrad dem LKW folgt.

Bisher hat sich keine einheitliche graphische Darstellung von Ontologien durchgesetzt. Daher werden in dieser Arbeit taxonomische Relationen in Anlehnung an die Notation der unified modeling language (UML) durch eine durchgezogene Linie mit geschlossener nicht ausgefüllter Pfeilspitze dargestellt (\rightarrow). In der Literatur werden in vielen Abbildungen von Ontologien Attribut-Relationen gar nicht visualisiert. Um diese Relationen darzustellen wird hier in Anlehnung an die Chen-Notation aus dem entity-relationship-model (ERM) eine Raute (\diamond) verwendet.

Zusätzlich können Regeln auf einer Ontologie definiert werden. Mit Hilfe dieser Regeln können unbekannte Informationen aus partiellem Wissen inferiert werden.



(a) TBox bestehend aus den Klassen: *Fahrzeug*, *Auto*, *Motorrad* und *LKW* sowie den Relationen: *ist ein* und *folgt*

(b) ABox als Instantiierung der TBox für eine Szene in der das **Motorrad X** dem **LKW Y** folgt

Abbildung 2.5: Graphische Darstellung einer einfachen Ontologie zur Beschreibung einer Verkehrsszene. Klassen bzw. Entitäten sind in grün dargestellt, Relationen in blau.

Regeln werden zumeist in der Form von prädikatenlogischen Formalen wie in der semantic web rule language (SWRL) [HPSB+04] dargestellt. Aufgrund der prädikatenlogischen Formulierung der Regeln können keine widersprüchlichen Informationen verarbeitet werden. Bei sich widersprechenden Fakten sind andere Konzepte wie Marcow logic networks (MLNs) [DKP+06] geeigneter.

Neben der hier schon erwähnten Darstellung von Kreuzungsszenarien [HZW11] werden ähnliche Ontologien auch in anderen Arbeiten angewendet. Einige Klassen wie Fahrspursegmente oder Fahrzeuge und auch Relationen zwischen diesen werden häufig in ähnlicher Weise verwendet. Jedoch werden je nach Applikation auch spezielle Klassen und Relationen verwendet. Die Straßenszenendarstellung des Team „AnnieWay“ [KPV+07] benutzt eine speziell auf die Szenarien der DARPA Urban Challenge zugeschnittene Ontologie. Aufgrund der Verwendung amerikanischer Verkehrsregeln wie four-way stop sind Relationen über die zeitliche Abfolge des Erreichens einer Kreuzung modelliert (z. B. *arrives_before*). Auch für die Beschreibung von Verkehrsszenen basierend auf Kamerabildern wird eine ähnliche Ontologie verwendet [HTL07]. Hierbei liegt jedoch ein starker Fokus auf der Abbildung verschiedener Straßengeometrien. Somit sind in der verwendeten Ontologie spezielle Klassen für verschiedene Geometrie-Typen, wie Kurven oder Spurverengungen vorgesehen.

2.4 Straßengeometrie und -topologie

Zur Darstellung von Straßennetzen kommen geometrisch topologische Karten zum Einsatz. Hierbei wird der befahrbare Bereich sowie andere interessante Bereiche und Objekte mittels geometrischer Objekte wie Punkte, Polygonzüge und Flächen dargestellt. Häufig werden Straßennetze auch durch geometrische Graphenstrukturen dargestellt, bei denen die Kanten befahrbare Bereiche wie Straßen oder Fahrspuren darstellen. Knoten werden zur Verzweigung von Wegen z. B. bei Kreuzungen oder zur Detaillierung der Geometrie verwendet.

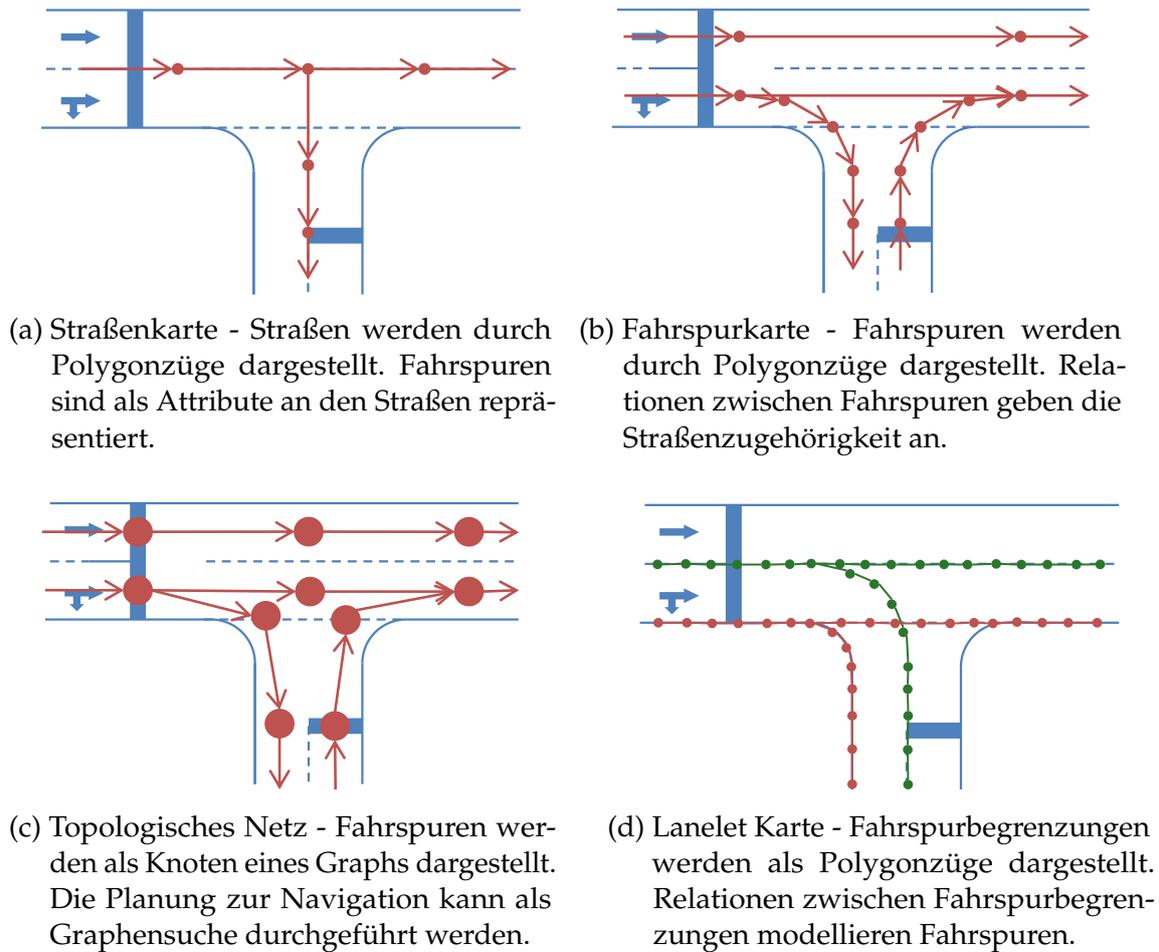


Abbildung 2.6: Vergleich verschiedener geometrisch topologischer Repräsentationsformen von Straßennetzen

Je nach Granularität können Punkte, Polygone oder Flächen verschiedene Semantiken aufweisen. Durch Attribute, sogenannte Tags, und Relationen können geometrischen Objekte weitere semantische Eigenschaften zugewiesen werden. Zusätzlich wird die Topologie der Strukturen, also geometrische Relationen einzelner Objekte zueinander wie die Übergangsmöglichkeit von einer zur einer anderen Straße an einer Kreuzung, repräsentiert. Aufgrund des höheren Abstraktionsniveaus und der stärkeren semantischen Aussagekraft werden in dieser Arbeit geometrisch topologische Karten verwendet. Im Folgenden werden verschiedene Granularitätsstufen dieser Karten vorgestellt. Abbildung 2.6 zeigt eine Übersicht der verschiedenen Repräsentationsformen am Beispiel einer Kreuzung.

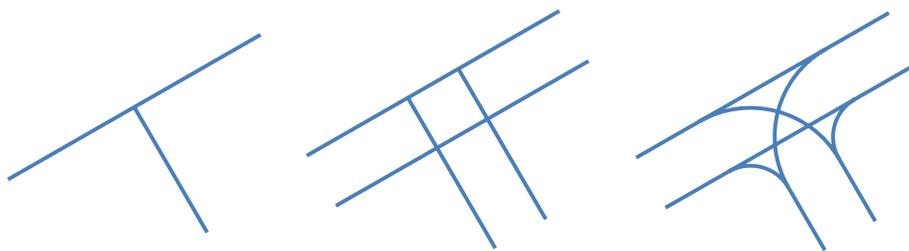


Abbildung 2.7: Verschiedene Repräsentationsmöglichkeiten einer T-Kreuzung in OSM. Modellierung von Straßen (links), einzelner Spuren (Mitte) sowie aller Fahr- und Abbiegespuren (rechts).

Erstveröffentlichung in: [KKJ+14a] (Kuhnt, Kohlhaas et al.)

2.4.1 Straßenkarten

Karten geben die Position und den Verlauf von Straßen auf der Erde an. Im offenen Kartendienst OpenStreetMap (OSM)²⁷ werden Straßen als Polygone aus georeferenzierten Punkten dargestellt. Dabei wird zumeist nur ein Polygonzug pro Straße verwendet. Soll die Straßengeometrie genauer abgebildet werden, so müssen Informationen wie die Anzahl der Fahrspuren oder die Breite dieser zusätzlich an die Kanten oder Knoten der Kartendarstellung annotiert werden. Die genaue Lage und der Verlauf der einzelnen Fahrspuren ist somit nicht bekannt. Zusatzinformationen wie die Art und Höhe der Straße oder das aktuelle Geschwindigkeitslimit werden mittels Attributen, so genannten Tags, annotiert. Eine beispielhafte Darstellung wird in Abbildung 2.6a gezeigt.

Kreuzungen können auf sehr unterschiedliche Weise dargestellt werden. Die einfachste Form ist die Verbindung zweier Wege durch einen gemeinsamen Punkt (Abbildung 2.7 links). In diesem Fall ist die Geometrie der Kreuzung weitestgehend unbekannt. Eine etwas detailliertere Modellierung der Kreuzung stellt die Aufteilung der Straße in zwei Wege dar, jeweils einen für jede Fahrtrichtung (Abbildung 2.7 mitte). Diese Darstellung wird häufig bei größeren Kreuzungen und bei baulicher Trennung der Fahrtrichtungen verwendet. Hier kann zumindest die Größe der Kreuzung aus der Geometrie abgeschätzt werden. Eine dritte Möglichkeit ist die Modellierung aller möglicher Fahrwege (Abbildung 2.7 rechts). Hierbei wird die Geometrie von Abbiegespuren detaillierter dargestellt. Ein Weg kann jedoch weiterhin für mehrere Fahrspuren stehen, z. B. bei Kreuzungen mit zweispurigen Abbiegespuren.

2.4.2 Fahrspurkarten

Eine etwas detailliertere Darstellung der Straßengeometrie bieten Karten, die jede Fahrspur einzeln modellieren (vergleiche Abbildung 2.6b). Dies ist z. B. bei dem in der DARPA Urban Challenge verwendeten Kartenformat route network definition

²⁷ OpenStreetMap: www.openstreetmap.org (23.11.2017)

file (RNDF)²⁸ der Fall. Hier werden einzelne Fahrspuren durch Polygonzüge repräsentiert. Fahrspuren sind einer Straße zugeordnet und können an Kreuzungen mittels sogenannter entry und exit way points verbunden werden. Fahrspuren tragen Informationen über ihre Breite und Markierungsarten für ihre linke und rechte Begrenzung. Eine ähnliche Darstellungsform ist der RoadGraph wie er in [KH10] vorgestellt wird.

2.4.3 Topologische Netze

Für die Planung von Fahrmanövern und Routen wird zusätzlich zur geometrischen auch topologische Information benötigt. Graphbasierte Ansätze wie Roadgraph oder OSM beinhalten diese Information implizit. Sie haben allerdings den Nachteil, dass topologische und geometrische Information teilweise inkonsistent verwendet werden. So stellen in diesen Darstellungen die Knoten feste geometrische Orte dar. Objekte werden jedoch nicht diesen Orten, sondern den Kanten also den Relationen zwischen ihnen zugeordnet (vergleiche Kapitel 2.3.2).

Eine klare Trennung bieten hier topologische Netze. In topologischen Netzen werden Bereiche der befahrbaren Fläche durch Knoten in einer Graphstruktur dargestellt (siehe [Thr98] und [Fre12]). Übergangsmöglichkeiten zwischen diesen sind durch Kanten repräsentiert. Somit unterscheidet sich diese Darstellung grundlegend von den oben beschriebenen Straßen- und Fahrspurkarten, bei denen befahrbare Flächen durch Kanten definiert werden. Zur Darstellung einer Straßenkarte werden also einzelne Fahrspursegmente durch Knoten repräsentiert. Je nach Anwendung kann es sinnvoll sein, die maximale Länge der verwendeten Straßensegmente zu beschränken [Fre12].

Topologische Netze eignen sich gut zur Wegplanung. Daher werden sie auch in anderen Kartenmodellen angewendet, um Navigationsaufgaben zu lösen. Für RNDF wurde die topologische Information in der Erweiterung RNDF-Graph aufgenommen. Auch in der später vorgestellten Kartendarstellung durch Lanelets wird diese Information als topologischer Graph bereitgestellt. Dabei wird die geometrische Struktur durch eine duale topologische Struktur ergänzt. Eine mögliche Modellierung der T-Kreuzung als topologisches Netz ist in Abbildung 2.6c dargestellt.

2.4.4 Spurgenaue Karten – Lanelets

Wird eine geometrisch präzisere Darstellung der befahrbaren Fläche benötigt, reicht die Darstellung einer Fahrspur als einfacher Polygonzug nicht aus. Eine spurgenaue Straßenrepräsentation bietet die Darstellung als Lanelets [BZS14].

²⁸ DARPA Grand Challenge 2005: archive.darpa.mil/grandchallenge/docs/RNDF_MDF_Formats_031407.pdf (23.11.2017)

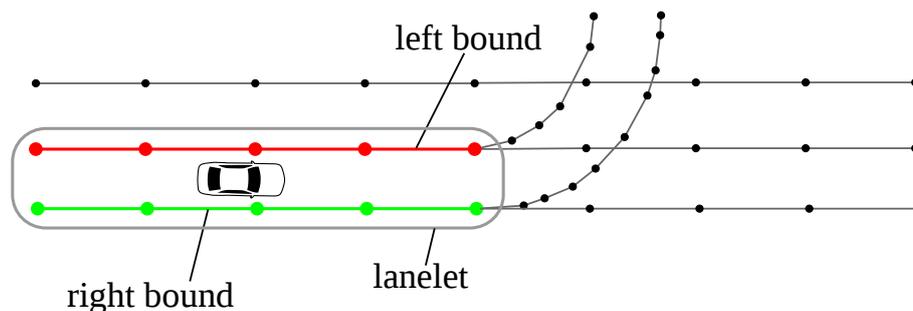


Abbildung 2.8: Kartenausschnitt dargestellt mit Lanelets. Bei einem Lanelet sind die Begrenzungspolygone (rot: linke Begrenzung, grün: rechte Begrenzung) farblich markiert.

Bildquelle: [BZS14] © 2014 IEEE

Diese wurde im Rahmen der Entwicklung des Mercedes-Benz S 500 INTELLIGENT DRIVE eingeführt (siehe Kapitel 2.2) [ZBS+14]. Das Kartenformat wurde vom FZI weiterentwickelt und als „Lanelet2“ veröffentlicht [PPJ+18]. Ein Lanelet beschreibt einen elementaren Fahrspurabschnitt. Er wird durch seine linke und rechte Begrenzung in Form von Polygonen dargestellt. Abbildung 2.8 visualisiert ein allgemeines Anwendungsbeispiel einer Laneletkarte. Die Polygone bestehen aus georeferenzierten Punkten. Dies erlaubt eine beliebige Verortung der Lanelets auf der Erde. Abbildung 2.6d zeigt einen Teil der T-Kreuzung modelliert als spurgenaue Karte.

Mehrere Lanelets können zu einer Karte zusammengefasst werden. Die Nachbarschafts- und Folgebeziehungen der einzelnen Segmente werden durch gemeinsame Punkte bzw. gemeinsame Begrenzungspolygone dargestellt. Bei Lanelets, die Nachbarspuren darstellen, sind die Punkte des gemeinsamen Begrenzungspolygons gleich. Aufeinanderfolgende Lanelets teilen sich jeweils Anfangs- bzw. Endpunkt der Begrenzungspolygone.

Wie oben erwähnt wird die topologische Struktur der Lanelets durch eine Graphenstruktur abgebildet, die aus der geometrischen Information extrahiert werden kann. Bei dieser werden die Lanelets als Knoten und Verbindungen zwischen diesen als Kanten dargestellt. Somit ergibt sich ein topologisches Netz. Dies ermöglicht eine effiziente Routenplanung von einem Start- zu einem Ziel-Lanelet vergleichbar mit der eines Navigationssystems.

Zusätzlich werden bestimmte Relationen zwischen einzelnen Lanelets, die sich nicht effizient aus der Geometrie herleiten lassen, in der Karte annotiert. Wenn sich z. B. zwei Fahrspuren kreuzen, wird zwischen ihnen eine Relation „cross“ annotiert. Diese Relation kann maschinell aus der Geometrie abgeleitet und in die Karten eingetragen werden.

Für die Durchführung der Berta-Benz-Fahrt wurde die Karte so erstellt, dass für die Fahrt keine Spurwechsel auf der digitalen Karte nötig waren. Abbildung 2.9 zeigt einen Ausschnitt der verwendeten Karte. Die Fahrspur 2 modelliert die Linksabbiegerspur vor einer Kreuzung. Sie ist so modelliert, dass ein direkter

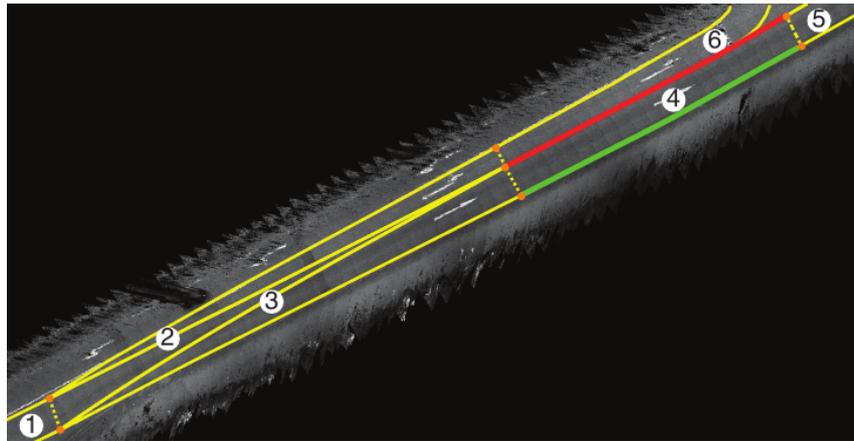


Abbildung 2.9: Kartendarstellung mit Lanelets. Modellierung einer Abbiegespur vor einer Kreuzung.

Bildquelle: [ZBS+14] © 2014 IEEE

Übergang ohne Spurwechsel von Fahrspur 1 möglich ist. Der Wechsel auf die Abbiegespur wird aufgrund der Modellierung frühestmöglich durchgeführt, so dass lediglich vorausfahrende Fahrzeuge berücksichtigt werden müssen. Würden die vorhandenen Straßenmarkierungen als Modellierungsgrundlage verwendet, wäre zunächst ein Übergang zu Fahrspur 3 und dann ein Spurwechsel zu Fahrspur 2 nötig.

2.5 Perzeption von Verkehrsteilnehmern

Ziel der Perzeption ist es, aus von Sensoren gewonnenen Daten die relevanten Objekte und Strukturen im Fahrzeugumfeld zu identifizieren, zu klassifizieren und zu verorten. In automatisierten Fahrzeugen werden unterschiedliche Sensoren für die Umfeldwahrnehmung eingesetzt. Das Versuchsfahrzeug CoCar (siehe Kapitel 2.2.2) verfügt über mehrere, teils komplementäre Sensoren. Es stehen viele verschiedene Verfahren zur Objektdetektion und -klassifikation auf dem Fahrzeug zur Verfügung. Im Folgenden werden Verfahren zur Objektdetektion vorgestellt, die im Rahmen dieser Arbeit z. B. für die Evaluation der Verfahren auf realen Sensordaten für den Aufbau eines geometrischen Umfeldmodells verwendet werden.

2.5.1 Objekterkennung auf Videobildern

Die Objekterkennung auf Videobildern, wie sie z. B. zur Autodetektion [ZKS+14] oder zur Ampel- und Verkehrszeichendetektion [NGZ11] verwendet wird, erfolgt häufig über die in Abbildung 2.10 dargestellte Verarbeitungskette. Auf von Kameras aufgenommenen Videodaten werden in der Segmentierung interessante

2 Grundlagen und Stand der Technik



Abbildung 2.10: Typische Verarbeitungskette eines videobildbasierten Objektdektors

Bildbereiche ermittelt, in denen sich eines der gesuchten Objekte befindet. Eine einfache Form der Segmentierung ist das Sliding-Window-Verfahren, bei dem ein Suchfenster über das Bild geschoben wird und somit mögliche Objektkandidaten erzeugt werden können. Das Suchfenster kann in der Größe variiert werden, um Objekte in unterschiedlichen Entfernungen detektieren zu können. Dieses Verfahren liefert sehr viele Objekthypothesen und ist daher im Zusammenspiel mit einem aufwändigen Klassifikationsverfahren in der Praxis nicht direkt einsetzbar.

Durch die Einschränkung der Fensterpositionen z. B. unter Annahme bestimmter möglicher Objektpositionen kann die Anzahl der Suchregionen reduziert werden. Beispielsweise kann mit dem Wissen, dass Fahrzeuge in der Regel auf der Bodenebene fahren, der Suchraum stark eingeschränkt werden. Eine weitere Vorselektion der Suchregionen kann durch die Verwendung eines schnellen Klassifikators erreicht werden. Hier bieten sich kaskadierte Klassifikatoren an, wie eine Kombination aus Haar-Wavelets und Ada-Boosting [VJ03]. Diese haben die Eigenschaft, dass auf einem Großteil der Suchregionen nur wenige Operationen durchgeführt werden müssen, um festzustellen, dass sich das gesuchte Objekt nicht in der Region befindet. Der Klassifikator muss jedoch so parametrisiert sein, dass bei der Segmentierung alle Suchregionen an die Klassifikation weitergegeben werden, in denen sich das gesuchte Objekt befindet.

Die von der Segmentierung bereitgestellten Bildregionen werden von der Klassifikation validiert. Soll nur eine Klasse von Objekten (z. B. Autos) erkannt werden, genügt ein Klassifikator, der zwischen einer Klasse und Hintergrund unterscheiden kann. Da von der Segmentierung im Idealfall nur wenige Kandidaten-Regionen weitergegeben werden, können rechenaufwändigere Klassifikatoren zum Einsatz kommen.

Ein Klassifikator besteht aus einer Merkmalsextraktion, die das Eingabebild in einen Merkmalsvektor überführt sowie einem Klassifikationsverfahren auf einem Merkmalsvektor. Als Merkmalsvektor können im einfachsten Fall die Farbwerte der einzelnen Pixel des Bilds verwendet werden. Es können jedoch auch lokale Kantenmerkmale extrahiert werden, ein so genanntes histogram of oriented gradients (HOG). Als Klassifikationsverfahren haben sich in den 2000er Jahren support vector machines (SVMs) etabliert [Vap98]. Abbildung 2.11 zeigt das Ergebnis einer Objektdetektion auf einer semivirtuellen Autobahnscene.

In den letzten Jahren sind vermehrt Verfahren basierend auf tiefen neuronalen Netzen erforscht und erfolgreich eingesetzt worden. Vor allem der Einsatz sogenannter convolutional neural networks (CNNs) [WWZ16] zeigt sehr vielversprechende Ergebnisse. Hierbei können sowohl verschiedene Objektklassen als auch die Positionen von den zugehörigen Boundingboxen durch ein Netz auf



Abbildung 2.11: Videobasierte Detektion von Autos auf einer semivirtuellen Autobahnscene

Erstveröffentlichung in [ZKS+14] (Zofka, Kohlhaas et al.) © 2014 IEEE

hoch paralleler Hardware wie Grafikkarten effizient und mit sehr hoher Präzision ermittelt werden. Diese Netze vereinen somit Segmentierung und Klassifikation sowie die darin verwendete Merkmalsextraktion.

Detektierte und klassifizierte Objekte liegen zunächst lediglich als Bildregionen im 2-dimensionalen Bildraum vor. Zur weiteren Verarbeitung müssen diese in den 3-dimensionalen Raum um das Fahrzeug hineinprojiziert werden. Durch die Abbildungseigenschaften einer 2D-Kamera kann die Größe und Entfernung eines im Bild gefundenen Objektes nicht eindeutig rekonstruiert werden. Auf Basis von Annahmen zur Größe oder Position ist jedoch eine Rekonstruktion der Position möglich. Bei Autos z. B. können Annahmen über die zu erwartende Breite oder ihre Position (auf der Bodenebene) getroffen werden.

Um konsistente Aussagen über beobachtete Objekte über einen längeren Zeitraum hinweg treffen zu können, müssen die Detektionsergebnisse aufeinanderfolgender Bilder zeitlich fusioniert werden (Tracking). Hierbei sind zwei Probleme zu lösen: Die Bewegung eines Objektes muss anhand aufeinanderfolgender Messungen geschätzt werden. Bei mehreren Messungen pro Bild müssen diese korrekt mit den jeweiligen Objekten assoziiert werden.

Für die Bewegungsschätzung eines Objektes aus diskreten Messungen hat sich das Kalman-Filter etabliert [Kal60]. Dieses probabilistische Filter ist eine Spezialisierung des Bayesschen Filters. Beim Kalman-Filter werden die Annahmen getroffen, dass die Unsicherheit des Objektzustands sowie das Messrauschen und das Rauschen im Bewegungsmodell jeweils durch die ersten beiden Momente eindeutig beschrieben ist. Des Weiteren wird ein lineares Messmodell und ein lineares Bewegungsmodell angenommen. Liegen nichtlineare Modelle vor, müssen diese mit Hilfe der Jacobimatrix lokal linearisiert werden.

Für die Verfolgung von Fahrzeugen reichen in der Regel einfache, lineare Bewegungsmodelle aus. Sollen Fahrzeuge in der Umgebung eines Ego-Fahrzeugs detektiert und in einem Ego-Fahrzeug-festen Koordinatensystem verfolgt werden, führen jedoch Rotationsbewegungen des Ego-Fahrzeugs zu Nichtlinearitäten im Bewegungsmodell der Objekte. Durch geschickte Wahl der Zustandsräume und Koordinatensysteme können Nichtlinearitäten jedoch vermieden werden. Wird zwar ein Ego-Fahrzeug-zentriertes Koordinatensystem verwendet, dessen Ausrichtung jedoch immer parallel zu einem ortsfesten Koordinatensystem ist, bleibt die Linearität im Bewegungsmodell der Objekte erhalten. Die Rotation des Ego-Fahrzeugs und somit die der eingesetzten Sensoren kann im Messmodell berücksichtigt werden ohne die Linearität dieses Modells zu beeinträchtigen.

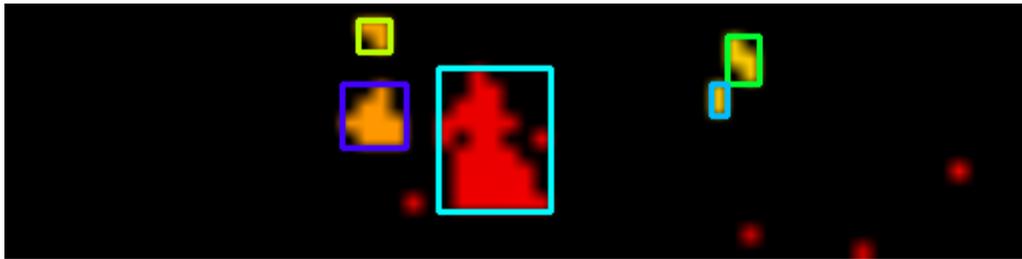
2.5.2 Objekterkennung auf Video-Tiefenbildern

Tiefenbildkameras liefern ein 3-dimensionales Abbild der wahrgenommenen Szene. Dabei repräsentiert jedes Pixel eine gemessene Entfernung von der Kamera zu einem Szenenobjekt an dieser Stelle. Je nach Messprinzip wird in weiteren Kanälen die Reflektanz oder auch die Farbe der Szene an der jeweiligen Stelle repräsentiert. In Fahrzeugen kommen verschiedene Messprinzipien zum Einsatz.

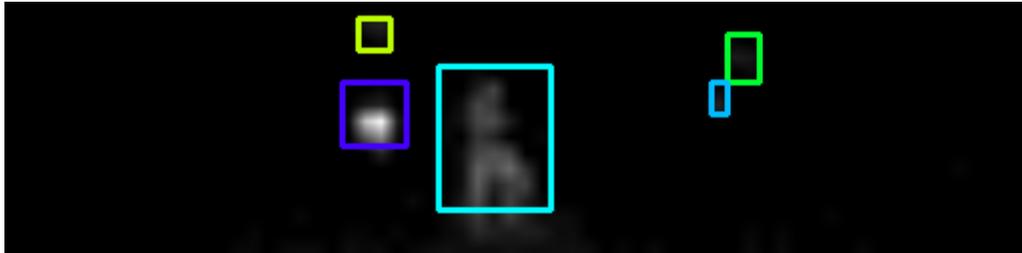
Stereokameras funktionieren ähnlich wie ein menschliches Augenpaar. Aufgrund der leicht unterschiedlichen Blickwinkel der beiden verwendeten Kameras kann die Entfernung von Objekten geschätzt werden. Das Lichtschnittverfahren hingegen verwendet ein bekanntes, in die Szene projiziertes Muster. Durch Veränderungen des durch eine Kamera wahrgenommenen Musters durch die 3D-Struktur der Szene kann die Geometrie ähnlich wie bei der Stereobildverarbeitung rekonstruiert werden.

PMD-Kameras arbeiten nach dem so genannten Licht-Laufzeit-Verfahren (TOF). Es wird also die Laufzeit des Lichts von einer aktiven Lichtquelle zu einem Szenenobjekt und wieder zurück zur Kamera gemessen. Dabei wird moduliertes Infrarotlicht von speziellen Lichtquellen ausgestrahlt. Die Kamera misst die Laufzeit des von der Szene zurückgeworfenen Lichtes und kann somit die Entfernung der reflektierenden Objekte ermitteln. Die Bildpunkte (Pixel) eines von einer PMD-Kamera bereitgestellten Bilds codieren somit die Entfernung der erfassten 3D-Struktur zur Kamera. Des Weiteren wird häufig auch ein Reflektanzwert als eigener Kanal im Bild kodiert. Die im automotive Bereich eingesetzten TOF-Kameras sind in der Lage Umgebungslicht z. B. von der Sonne zu kompensieren und können somit unter fast allen Beleuchtungsverhältnissen eingesetzt werden. Im Gegensatz zu normalen Videokameras ist die Auflösung dieser Kameras jedoch stark beschränkt. Das Kamerasystem, das auf dem Versuchsträger CoCar eingesetzt wird, hat eine Auflösung von 64×16 Pixeln.

Auf Basis dieser Daten können verschiedene Verfahren zur Objektsegmentierung angewendet werden. Eine Möglichkeit ist die Verwendung von unseeded region



(a) Tiefenbild einer TOF-Kamera. Farbige Rechtecke repräsentieren segmentierte Objekte.



(b) Intensitätsbild einer TOF-Kamera. Der segmentierte Fußgänger (hellblau) ist auch für einen menschlichen Betrachter kaum als solches zu erkennen.

Abbildung 2.12: Objekterkennung auf einem Tiefenbild einer TOF-Kamera

growing [SZV+08]. Hierbei werden Bildregionen mit annähernd gleicher Entfernung zur Kamera zu Objekthypothesen zusammengefasst. Bei diesem Ansatz wird ein Objekt nur dann richtig erkannt, wenn es an allen Stellen annähernd gleichen Abstand zur Kamera hat, also annähernd parallel zur Bildebene der Kamera steht. Dies führt vor allem bei schrägen Flächen wie Hauswänden oder der Bodenebene zu falschen Segmentierungen.

Eine weitere Möglichkeit ist die systematische Suche nach geneigten Ebenen auf Basis von u - bzw. v -Histogrammen [SRZ10]. Die Bilddaten werden hierbei zunächst in ein u - und ein v -Histogramm transformiert. Dabei werden Flächen, die gegenüber der Bildebene lediglich entlang der u - oder v -Achse gedreht sind auf Linien abgebildet. Diese können mit Hilfe einer Hough-Transformation detektiert werden. Somit können die Bodenebene, Wände, aber auch andere Objekte wie Menschen oder Autos robust segmentiert werden. Im Gegensatz zum unseeded region growing werden für eine erfolgreiche Detektion jedoch mehr Bildpunkte pro Objekt benötigt. Somit eignet sich das Verfahren gut für hochauflösende 3D-Kameras.

Die Projektion erkannter Objekte von der Bildebene in das Fahrzeugumfeld ist aufgrund der vorhandenen Abstandsinformation ohne weitere Modellannahmen möglich. Abbildung 2.12 zeigt die Objektsegmentierung auf einem Tiefenbild [KSL+13]. Eine Klassifikation der Segmentierungsergebnisse allein auf den Sensordaten der 3D-Kamera ist aufgrund der geringen Auflösung nur schwer möglich. Hier kann eine Fusion von 3D und 2D Videodaten eingesetzt werden.

2 Grundlagen und Stand der Technik

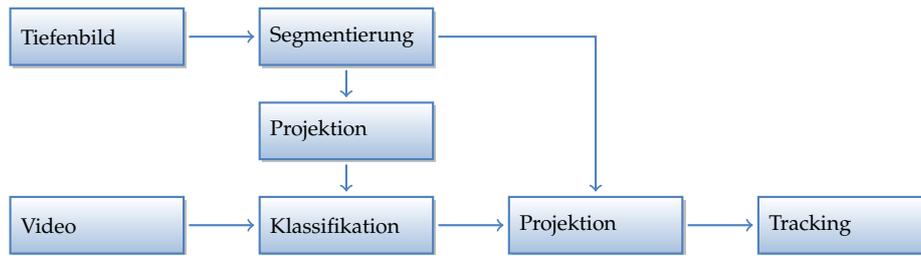


Abbildung 2.13: Verarbeitungskette eines kombinierten Objektdetektors für 2D- und 3D-Bilder

2.5.3 Fusion

Die Fusion von komplementären Sensordaten kann die Umgebungswahrnehmung verbessern. Abbildung 2.13 zeigt eine mögliche Verarbeitungskette, wie sie zur Detektion von Fußgängern eingesetzt wird [KSS+10]. Hierbei wird zunächst eine Objektsegmentierung auf dem Tiefenbild durchgeführt. Mittels der 3D-Information und dem Wissen über die relative Positionierung der 2D- und 3D-Kamera können die gefundenen Objekte in das Kamerabild projiziert und dort als Segmentierungsergebnis verwendet werden [SSG+09]. Auf dem hochauflösten, farbigen Bild der 2D-Kamera kann dann eine Klassifikation des Objekts durchgeführt werden. Abbildung 2.14 zeigt eine Beispielszene in der Objekte anhand der Daten einer Tiefenbildkamera segmentiert werden (unten). Nach der Projektion in das Videobild können die Bildbereiche in Fußgänger (rot) und andere Objekte (grün) klassifiziert werden (oben). Mittels Tracking werden die Objekte über mehrere Bilder hinweg verfolgt. Die Historie wird als Linie dargestellt.



Abbildung 2.14: Objektdetektion und Klassifikation auf Basis eines Tiefenbilds einer TOF-Kamera und eines Videobilds. Die in das Videobild projizierten Objekthypothesen werden klassifiziert (rot: Fußgänger).

2.5.4 Objekterkennung auf Laserscanner-Punktwolken

Mit mehrzeiligen Laserscannern wie sie z. B. auf dem Forschungsfahrzeug CoCar (siehe Kapitel 2.2.2) zum Einsatz kommen, kann die Umgebung dreidimensional erfasst werden. Das Messprinzip, bei dem mittels rotierender Laserstrahlen die Umgebung abgetastet wird, erlaubt über die Laufzeitanalyse der Strahlen eine direkte Bestimmung von Entfernungen zu bestimmten Punkten der Szene. Die Sensordaten repräsentieren die Umgebung als dreidimensionale Punktwolke. Die lokale Auflösung hängt dabei von der Anzahl der verwendeten Lasern, dem Öffnungswinkel sowie der Abtastrate und Rotationsgeschwindigkeit der Laser ab. Durch die konstante Winkelauflösung nimmt die horizontale und vertikale Auflösung mit der Entfernung linear ab.

Durch Verfahren wie Bodenebenenextraktion und Clustering können einzelne Objekte in der Szene segmentiert werden. Durch die beschränkte Sicht auf die Szene kann die dreidimensionale Struktur von Objekten nicht komplett erfasst werden. Fahrzeuge und andere annähernd quaderförmige Objekte werden nur als L-förmige oder U-förmige Strukturen wahrgenommen. Durch Modellannahmen und durch die Verfolgung eines Objekts über mehrere Zeitschritte hinweg können die realen Abmaße des Objekts geschätzt werden. Dies wird im Falle der Lux-Sensoren von IBEO bereits in einer dedizierten Hardware durchgeführt, die Signale von bis zu sechs Sensoren fusioniert und zeitlich konsistente Objektlisten bereitstellt. Anhand von Größe und Bewegungsprofil der Objekte wird auch eine grobe Klassifikation durchgeführt, so dass Fußgänger, Autos, LKWs und statische Objekte unterschieden werden können.

Auch auf Laserscanner-Punktwolken können zur Segmentierung und Klassifikation Lernverfahren wie CNNs angewendet werden [PPS+18]. So ist es z. B. möglich, jedem Punkt eine Objektklasse zuzuordnen. Die Güte der Klassifikation hängt stark mit der Größe der Trainingsmenge zusammen. Daher wurde eine Methode entwickelt, um ähnlich zu der in Kapitel 2.5.3 vorgestellten Projektion Segmentierungsergebnisse aus Videodaten auf die Punktwolken zu übertragen.

2.6 Prädiktion

Die Prädiktion ist Teil der Situationsinterpretation und trifft auf Basis der aktuellen Situation Vorhersagen über den Verlauf der Situation. Da viele Szenen nicht deterministisch sind, ist eine exakte Vorhersage im Allgemeinen nicht möglich. Daher werden Verfahren der Wahrscheinlichkeitstheorie wie Dichteverteilungen für die Beschreibung von Zuständen verwendet.

Häufig werden Modellannahmen getroffen, um die Wahrscheinlichkeitsdichten beherrschbar zu machen. Gängige Annahmen sind, dass Objekte den Gesetzen der Physik unterliegen und sich ihre Position, Geschwindigkeit oder Beschleunigung

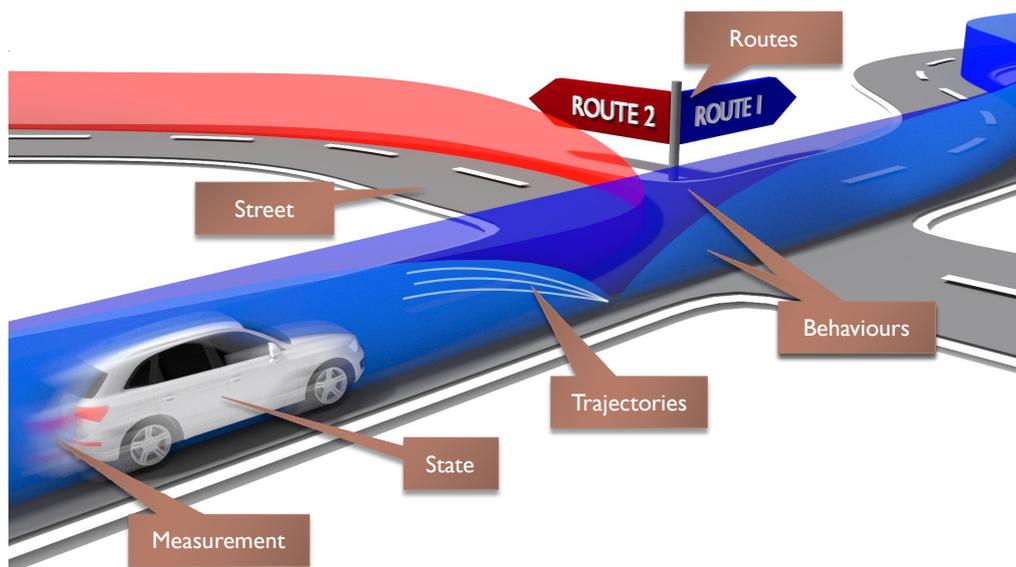


Abbildung 2.15: Schematische Darstellung der wichtigsten Begriffe des Szenenmodells. Abhängig von der Straße (Street) und der gewählten Route (Route) können einem Fahrzeug verschiedene Verhaltensmuster (Behaviours) zugeordnet werden. Diese führen zu unterschiedlichen Trajektorien (Trajectories). Mittels einer kontinuierlichen Messung (Measurement) des aktuellen Zustands (State) können Rückschlüsse auf Verhalten und Routenwahl inferiert werden.

Erstveröffentlichung in [KKS+15] (Kuhnt, Kohlhaas et al.) © 2015 IEEE

nur in bestimmten Grenzen verändert. Jedoch finden bei genauerer Kenntnis der Szene auch komplexere und aussagekräftigere Modelle Anwendung.

Zur Prädiktion der Fahrzeugdynamik können Modelle aus Kapitel 2.3.1 zum Einsatz kommen. Beim so genannten Einspur-Modell wird die Kinematik von radgetriebenen Fahrzeugen mit einer gelenkten Achse berücksichtigt. Eine einfache Form der Prädiktion ist eine direkte Anwendung des Kalman-Filters (vergleiche Kapitel 2.5.1). Hier kann durch mehrfache Ausführung des Prädiktionsschritts und unter Auslassung des Messschritts eine Prädiktion für diskrete Zeitpunkte in der Zukunft erreicht werden.

2.6.1 Manöverbasierte Prädiktion von Routen und Trajektorien

Für eine Prädiktion über einen längeren Zeitraum hinweg reichen einfache Modelle im Allgemeinen nicht aus. Bei Verkehrssituationen ist es z. B. sinnvoll von einem sinnvollen Fahrverhalten auszugehen (z. B. dass ein Fahrzeug im Regelfall der Straße folgt). Durch eine Prädiktion auf Basis von Verhaltensmustern, die von der Straßentopologie und -geometrie abhängig sind, können die Prädiktionsergebnisse verbessert werden (vergleiche Kapitel 2.4.2). Abbildung 2.15 zeigt

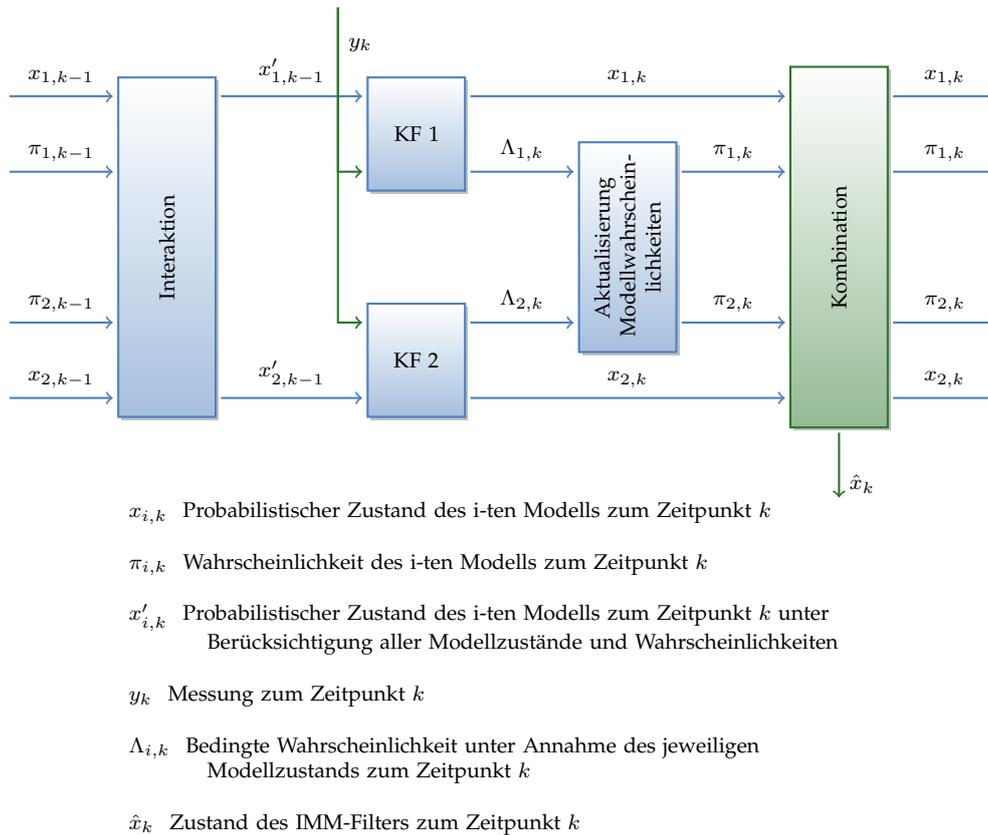


Abbildung 2.16: Blockdiagramm eines IMM-Filters mit zwei Modellen

eine schematische Darstellung der wichtigsten Begriffe des hier verwendeten Szenenmodells.

Auf Basis eines allgemeinen Szenenmodells kann ein interacting multiple model filter (IMM-Filter) [BSLK01] abgeleitet werden, das verschiedene an die Straßengeometrie angepasste Verhaltensmuster schätzt [KKS+15]. Ein IMM-Filter ist ein Bayessches Filter und eine Erweiterung des Kalman-Filters um verschiedene Modelle. Es liegt die Annahme zugrunde, dass sich das System in jedem Zeitpunkt entsprechend eines Modells aus einer diskreten Menge von Modellen verhält. Ein Wechsel zwischen den Modellen ist mit einer gewissen Wahrscheinlichkeit möglich und wird über einen Markow-Prozess modelliert.

Der Modellzustand ist nicht direkt beobachtbar, lässt sich jedoch mit Hilfe eines hidden Markow model (HMM) ermitteln. Dabei dient die aktuelle Performanz der von den Modellen induzierten Kalman-Filtern jeweils als Messung für den aktuellen Modellzustand. Umgekehrt wird die Wahrscheinlichkeitsverteilung über den Modellzustand für die Fusion der einzelnen Kalman-Filter-Zustände verwendet.

Abbildung 2.16 zeigt eine schematische Darstellung eines IMM-Filters mit zwei Modellen. Im Zentrum stehen Kalman-Filter (KFs) für jedes Modell, die einen Zu-

stand ($x'_{i,k-1}$) und eine aktuelle Messung (y_k) fusionieren. Zusätzlich zum geschätzten Zustand ($x_{i,k}$) wird auch eine bedingte Wahrscheinlichkeit für die Messung für das jeweils verwendete Modell (Λ_i) ermittelt. Diese Wahrscheinlichkeiten dienen als Eingabe für das HMM, über das die Modellwahrscheinlichkeiten aktualisiert werden. Auf Basis dieser werden dann die Einzelzustände der Kalman-Filter zu einem Gesamtzustand (\hat{x}_k) kombiniert. Für den nächsten Filterschritt werden die Zustandsübergangswahrscheinlichkeiten im HMM verwendet, um die Zustände der einzelnen Filter anzupassen und so eine Interaktion der Modelle zu ermöglichen.

An einer Kreuzung können so z. B. mehrere Verhaltensmuster für Geradeausfahrt, Anhalten und Abbiegen bei der Prädiktion berücksichtigt werden. Durch kontinuierliche Messungen kann das aktuell wahrscheinlichste Verhaltensmuster ermittelt und dieses dann zur Prädiktion verwendet werden. Hierdurch ist nicht nur eine genauere Prädiktion der Fahrzeugzustands möglich, auch eine Aussage über die von dem Fahrzeug gewählte Route, kann gleichzeitig getroffen werden. Um die Routenprognose weiter zu verbessern, können auch Interaktionen mit anderen Verkehrsteilnehmern berücksichtigt werden [KSS+16].

2.7 Kartierung und Lokalisierung

Die Position und Form der Fahrspuren vor dem Fahrzeug und um das Fahrzeug kann mit Hilfe von Sensoren im Fahrzeug, z. B. Kameras, erfasst werden. Hierzu können nicht nur Fahrspurmarkierungen sondern ggf. auch andere Objekte wie Leitbarken in Baustellen [GNL+09] erkannt und interpretiert werden. Einfache Fahrspurmodelle können dann durch Filterverfahren geschätzt werden. Komplexere Fahrspurtopologien können durch diese Modelle nicht erfasst werden. Hier haben sich hierarchische Ansätze etabliert [KPZ+16], die den Fahrspurverlauf von einfachen Spurmerkmalen über kurze Fahrspursegmente und zusammengesetzte Fahrspuren bis hin zu ganzen Straßen modellieren. Um mit den den erfassten Daten zugrunde liegenden Unsicherheiten, Unschärfen und Widersprüchen umzugehen, kommen verschiedene Inferenzverfahren, wie Partikelinferenz [GOZ12] oder das so genannte Depth First Message Passing [Töp14] zum Einsatz.

Für eine vorausschauende Manöverplanung reicht im Regelfall der Sensorhorizont des Fahrzeugs nicht aus, so dass auf weitere Informationen wie Kartenmaterial zurückgegriffen werden muss. Frei verfügbares Kartenmaterial (wie OSM) weist jedoch vielerorts nicht den benötigten Detailgrad auf.

Um die Geometrie einzelner Fahrspuren aus weniger detailliert modellierten Kreuzungen ableiten zu können, wird in [KKJ+14b] und [KKJ+14a] ein parametrisches Kreuzungsmodell vorgestellt. Dabei wird die Kreuzung aus mehreren Korridoren zusammengesetzt, die jeweils durch einen Spline modelliert werden. Ein Korridor beschreibt dabei eine Menge von Fahrspuren, die von einer zu einer anderen Straße führen. Durch Anwendung von Randbedingungen wie maximalen

Kurvenradien innerhalb der Kreuzung werden die Anfangs und Endpunkte der Korridore sowie die Stützpunkte der Splines bestimmt. Mit Hilfe dieses Verfahrens ist es möglich, aus grob modellierten Kreuzungen spurdetaillierte Modelle abzuleiten. Eine genaue Übereinstimmung der erzeugten Fahrspuren zu realen Fahrspuren kann nicht gewährleistet werden. Eine notwendige Korrektur der Fahrspuren kann manuell auf Basis von Luftbildern oder aufgezeichneten Objekt- oder Ego-Trajektorien erfolgen.

Wird die Fahrspurinformation aus einer Karte extrahiert und nicht mit Hilfe der Fahrzeugsensorik erfasst, so wird die genaue Position des Fahrzeugs in der Karte benötigt. Ist die Karte georeferenziert, also ihre genaue Lage auf der Erde bekannt, so kann mit Hilfe einer genauen Verortung des Fahrzeugs z. B. mittels einer Kombination aus GPS und IMU (vergleiche Kapitel 2.2.2) die Karte zum Fahrzeug registriert werden. Andere Verfahren verwenden Sensormessungen und erkannte Fahrspurmerkmale [KKJ+14b], [KKJ+14a] oder auch umgebende Fahrzeuge [KOK+16], um diese mit der Karte abzugleichen und eine Verortung auf der Karte zu schätzen. Dabei ist es auch möglich, auf Basis der erkannten Merkmale die Karteninformation zusammen mit der Positionsschätzung zu verbessern [HSS15].

2.8 Ansätze zur Aktionsentscheidung

Auf Basis der gesammelten Informationen über die Umgebung und den eigenen Zustand des Fahrzeugs muss ein automatisiertes Fahrzeug Fahrentscheidungen treffen. Die verschiedenen Ansätze zur Entscheidungsfindung lassen sich entsprechend ihrer Grundprinzipien in verschiedene Kategorien unterteilen:

Verhaltensbasierte Ansätze kommen häufig im Bereich der Robotik vor, werden dort auch für die Steuerung von Fahrzeugen verwendet [ALB+02] und sind somit prinzipiell für den Einsatz in automatisierten Fahrzeugen geeignet. Bei diesen Verfahren werden verschiedene Verhalten vorgegeben, wie ein Ausweichverhalten oder ein Pfadfolgeverhalten. Die Verhalten generieren Steuerungsbefehle, die über alle Verhalten hinweg aufaggregiert werden. Entsprechend der aktuellen Situation werden Verhalten aktiviert oder deaktiviert und somit ihre Steuerungsbefehle stärker oder schwächer gewichtet. Während dieser Ansatz ein reaktives Verhalten zeigt, ist es damit nur schwer möglich vorausschauendes Verhalten abzubilden.

Regelungsbasierte Ansätze versuchen auf Basis des aktuellen Systemzustands und gegebener Sollwerte eine gewisse Fehlerfunktion zu minimieren. Zum Einsatz kommen diese Ansätze z. B. bei den oben genannten Verhalten. Im Automobilbereich kommen diese Ansätze z. B. im intelligent driver model (IDM) [THH00] auch bei Fahrerassistenzfunktionen wie ACC zum Einsatz. Regelungsbasierte Ansätze sind eine einfache, schnelle und gut erforschte Möglichkeit, lokal optimale Steuerungsbefehle zu erzeugen. Wie

auch bei den verhaltensbasierten Ansätzen sind sie weniger gut geeignet, um vorausschauende Entscheidungen zu treffen. Eine besondere Form der regelungsbasierten Ansätze ist die modellprädiktive Regelung (engl.: model predictive control, MPC)[DP04]. Hierbei wird ein Systemmodell mit verschiedenen Steuerungsparametern in die Zukunft prädiziert und das sich ergebende Verhalten bewertet. Zur Steuerung wird dann der Parametersatz mit der besten Bewertung verwendet. Auch wenn der Ansatz schon allein vom Namen her zu den regelungsbasierten Ansätzen zählt, kann er auch als Form einer Planung mit relativ kurzem Zeithorizont verstanden werden. So erlaubt er z. B. auch eine begrenzt vorausschauende Entscheidung zu treffen.

Planungsverfahren explorieren den möglichen Zustandsraum des Systems. Häufig wird dabei ein Zielzustand oder eine Menge von Zielzuständen festgelegt. Von den dabei erzeugten Plänen, Sequenzen von Aktionen bzw. Steuerungsparametern, wird dann einer für die Ausführung anhand von bestimmten Kriterien ausgewählt. Während der Planung kann die Entwicklung der Umgebung um das eigene Fahrzeug herum prädiziert werden und sogar Reaktionen der Umgebung auf den eigenen Plan berücksichtigt werden [KZZ18]. Planungsverfahren werden z. B. für die dynamische Pfadplanung [LF09] oder auch zur Planung von Einparktrajektorien [KOH+15] [Zie15] eingesetzt. Je nach Planungshorizont können so vorausschauende Verhaltensentscheidungen getroffen werden.

Häufig werden Heuristiken verwendet, um den Planungsprozess zu beschleunigen. Ein Beispiel hierfür ist die rekursive Konfliktauflösung [SP14] zur globalen Ableitung von Fahrmanövern mehrerer Fahrzeuge auf Autobahnen. Hier wird ausgenutzt, dass das Verhalten eines Fahrzeugs primär von dem der vorausfahrenden Fahrzeuge abhängt.

Strategiebasierte Verfahren wie Case-Based Reasoning [Vac09], partially observable Markov decision process (POMDP) [BGD14], Dynamische Programmierung (DP) und neuronale Netze [HBW+17] basieren auf einer vorberechneten Verhaltensrichtlinie (engl.: policy). Ausgehend von der aktuellen Situation wird ein Verhalten entsprechend dieser Strategie ausgewählt. Ein vorausschauendes Verhalten wird dadurch erreicht, dass beim Erzeugen der Verhaltensrichtlinie schon eine Prädiktion der Situation in die Zukunft berücksichtigt wird. Strategiebasierte Verfahren haben das Potential zur Ausführungszeit sehr schnell zu sein, da keine Prädiktion der Situation oder umfangreiche Berechnung von Folgezuständen notwendig ist. Limitiert werden diese Verfahren jedoch durch die Generierung und Darstellung der Strategie. Hierzu sind große Datenmengen und lange Rechenzeiten notwendig. Auch funktioniert das System nur in solchen Situationen gut, die bei der Erstellung so oder so ähnlich berücksichtigt wurden.

POMDPs sind eine häufig verwendete Form der strategiebasierten Verfahren, die es erlaubt, eine Strategie automatisiert aus Lerndaten abzuleiten [BGD14]. Im Rahmen des Projekts Stadtpilot [UM13], [UM14] wurden sie eingesetzt,

um Spurwechselmanöverentscheidungen automatisiert zu treffen und auszuführen. Dazu werden die Umfelddaten vor der Verarbeitung durch den POMDP stark vorverarbeitet. Basierend auf der Belegung verschiedener Bereiche im Fahrzeugumfeld und spezifischen Messgrößen zu Fahrzeugen in diesen Bereichen werden in zwei Vorverarbeitungseinheiten eine Spurwechselfähigkeit sowie ein Spurwechsellnutzen abgeleitet. Aus diesen Größen entscheidet dann ein POMDP, ob ein Spurwechsel durchgeführt werden soll oder nicht. Durch die Verwendung von aufwändigen, szenariospezifischen Vorverarbeitungseinheiten weist das System eine Vielzahl von Parametern auf, die experimentell eingestellt werden müssen und nicht durch die verwendeten POMDPs gelernt werden können. Eine Erweiterung auf andere Verkehrssituationen ist somit mit viel manuellem Aufwand verbunden.

Ende-zu-Ende-Lernen mit tiefen neuronalen Netzen ist eine weitere Form der strategiebasierten Verfahren. Hierbei wird ein neuronales Netz mit Sensoreingaben und zugehörigen Steuerbefehlen trainiert. Zur Laufzeit können dann neue Steuerbefehle aus den aktuellen Sensordaten generiert werden. In [BTD+16] wird z. B. ein neuronales Netz auf Videobildern von drei Kameras und zugehörigen Lenkwinkeln des Fahrzeugs trainiert. Um bei der späteren Ausführung robust gegen kleine Abweichungen zu sein, müssen auch Trainingsdaten abseits der gefahrenen Ideallinie mit zugehörigen Korrekturingriffen in den Trainingsdaten vorhanden sein. Hierzu können mittels projektiver Transformationen augmentierte Videodaten und zugehörige Lenkwinkelkorrekturen in die Trainingsdaten aufgenommen werden. Somit kann das neuronale Netz menschliches Fahrverhalten auf verschiedenen Straßengeometrien lernen und später imitieren. Um auch bei Kreuzungen oder bei mehrspurigen Straßen Abbiegemanöver oder Spurwechsel zu ermöglichen, können Informationen über die gewünschte Fahrtrichtung integriert werden [HBW+17]. Die Ausgabe des neuronalen Netzes ist nicht nur auf direkte Fahrzeugsteuersignale wie den Lenkwinkel beschränkt. Es kann auch eine vollständige Trajektorie generiert werden, die dann z. B. als Trajektorienvorschlag in einer anschließenden Optimierung oder Planung validiert und optimiert werden kann [HBD+17].

Optimierungsverfahren können ebenfalls für die Aktionsentscheidung angewendet werden. Hierzu wird der Steuerungsbefehl oder der zeitliche Verlauf der Steuerungsbefehle parametrisch dargestellt und diese Darstellung hinsichtlich einer Zielfunktion optimiert. Häufig werden Mehrzieloptimierungen eingesetzt um z. B. sowohl die Fahrzeit als auch den Energieverbrauch auf einer Strecke zu optimieren. Dabei kann im Allgemeinen nur ein sogenanntes Pareto-Optimum erreicht werden, bei dem sich die Erreichung keines Teilziels mehr verbessern lässt, ohne die eines anderen Teilziels zu verschlechtern. Viele Optimierungsverfahren erreichen lediglich lokale Minima. Somit ist es für die Anwendung dieser Verfahren wichtig, eine initiale Lösung in der Nähe des angestrebten Optimums angeben zu können. Für wenig komplexe Situationen und Fahrentscheidungen wie bei der optimalen Anfahrt

auf eine Ampel mit bekannten Schaltzeiten sind Optimierungsverfahren gut einsetzbar [RSK+11].

Zustandsautomaten sind eine graphische Methode der Verhaltensentscheidung. Sie werden durch Zustände, Zustandsübergänge und Aktionen beschrieben. Zustandsautomaten haben die Form eines gerichteten Graphens in dem Zustände als Knoten dargestellt werden. Ein Wechsel des Zustands ist entlang der die Zustände verbindenden Kanten möglich. Die Zustandswechsel sind in der Regel an bestimmte Bedingungen geknüpft. Die Auswahl der Aktion kann auf Basis des aktuellen Zustands, einer Zustandsänderung oder auch entsprechend bestimmter Eingaben z. B. aus den Wahrnehmungskomponenten in dem jeweiligen Zustand erfolgen.

Das Team „AnnieWAY“ z. B. verwendet auf verschiedenen Fahrzeugen zur Verhaltensgenerierung hierarchische, nebenläufige Zustandsautomaten [ZBS+14] [KZP+09]. In diesen sind verschiedene Situationen wie das Gewähren von Vorfahrt oder die Berücksichtigung von Ampeln abgebildet. Durch die Ausnutzung von Hierarchien ist eine leichtere Modellierung der notwendigen Zustände möglich. Die Nebenläufigkeit verschiedener Zustände und Situationen ermöglicht eine getrennte Modellierung von verschiedenen Situationsaspekten, die zur Ausführungszeit jedoch gleichzeitig eintreten und behandelt werden können. Dennoch ist ein sehr großer manueller Aufwand notwendig, alle auftretenden Situationen im Zustandsautomat zu modellieren.

In eingeschränkten Situationen und Szenarien sind Zustandsautomaten jedoch gut geeignet. In der GCDC wurden für die erfolgreiche Durchführung beider Szenarien Zustandsautomaten mit lediglich 7 bzw. 2 Zuständen benötigt [TSP+18]. Im Umgang mit vielfältigeren Szenarien stoßen Zustandsautomaten jedoch aufgrund des hohen manuellen Modellierungsaufwands an ihre Grenzen.

2.9 Aktionsplanung

Die Aktionsplanung kann als klassisches Planungsproblem aufgefasst werden, bei dem in einem Planungsraum ausgehend von einem Startzustand ein Plan zu einem oder mehreren Endzuständen gesucht wird. Dabei stehen verschiedene Aktionen zur Verfügung, um von einem Zustand zu einem anderen zu gelangen.

Häufig kann der Zustands- und Aktionsraum als Graph dargestellt werden, bei dem Zustände als Knoten des Graphen und die jeweiligen Aktionen als Kanten dargestellt werden. Die Planung auf einem solchen Raum entspricht einer Graphensuche (siehe Kapitel 2.9.1). Ist der Zustandsraum kontinuierlich, kommen entweder sampling-basierte Planer zum Einsatz oder Verfahren, die dennoch eine Graphensuche als Planungsverfahren ermöglichen, indem der Raum auf eine diskrete Anzahl von Zuständen reduziert wird (vergleiche Kapitel 2.9.2).

2.9.1 Graphbasierte Planer - Graphensuche

Bei einer graphbasierten Planung wird der Zustands- und Aktionsraum als gerichteter Graph dargestellt und somit das Planungsproblem auf eine Graphensuche reduziert. Ein Graph $G = (V, E)$ ist ein Tupel aus einer Menge von Knoten V und Kanten E . Gerichtete Kanten können dabei als Tupel zweier Knoten $(v_i \rightarrow v_j) \in E$ aufgefasst werden. Die Aufgabe einer Graphensuche ist es, in einer Graphstruktur einen Pfad \mathbf{p} aus Knoten $\mathbf{p} = (v_0, v_1, \dots, v_n)$ zu finden, der von einem Startknoten $v_s = v_0 \in V$ zu einem Zielknoten $v_z = v_n \in V_{\text{Ziel}} \subseteq V$ aus einer Menge von Zielzuständen (V_{Ziel}) führt. Dabei müssen alle Knoten des Pfads mit Kanten verbunden sein:

$$\forall i < n : (v_i \rightarrow v_{i+1}) \in E$$

Im Allgemeinen sind mehrere Lösungen möglich, vor allem dann, wenn die Menge der Zielzustände mehrere Elemente enthält. Häufig werden an den Pfad bestimmte Optimalitätsanforderungen gestellt. So kann z. B. der kürzeste oder der günstigste Pfad von Interesse sein. Hierzu kann der Graph mit Kosten für Knoten ($c : V \mapsto \mathbb{R}_0^+$) oder Kanten ($c : E \mapsto \mathbb{R}_0^+$) angereichert werden.

Die Gesamtkosten eines Pfads sind dann die Summe über die Kosten der besuchten Knoten:

$$c(\mathbf{p}) = \sum_{i=0}^n c(v_i) \quad (2.6)$$

bzw. über die Kosten der benutzten Kanten:

$$c(\mathbf{p}) = \sum_{i=0}^{n-1} c(v_i \rightarrow v_{i+1}) \quad (2.7)$$

Ohne Kenntnis der Struktur oder Kosten können für die Graphensuche nur einfache Suchalgorithmen eingesetzt werden. Diese Algorithmen sind generisch einsetzbar und stellen keine besonderen Anforderungen an den Graph. Dies führt jedoch auch zu einem relativ hohen Rechenaufwand bei der Suche eines Pfads. Steht Hintergrundwissen über die Struktur des Graphs oder eine Heuristik der attribuierten Kosten zur Verfügung, kann die Suche beschleunigt werden.

Im Folgenden werden grundlegende Algorithmen zur Graphensuche vorgestellt und ihre Eigenschaften hinsichtlich der Laufzeit dargestellt.

Einfache Suchalgorithmen

Ist kein weiteres Wissen über die Graphstruktur vorhanden, so kann ein Pfad lediglich durch systematisches Traversieren des Graphen gefunden werden. Zwei bekannte Verfahren sind hierbei Breitensuche (engl.: breadth-first search, BFS)

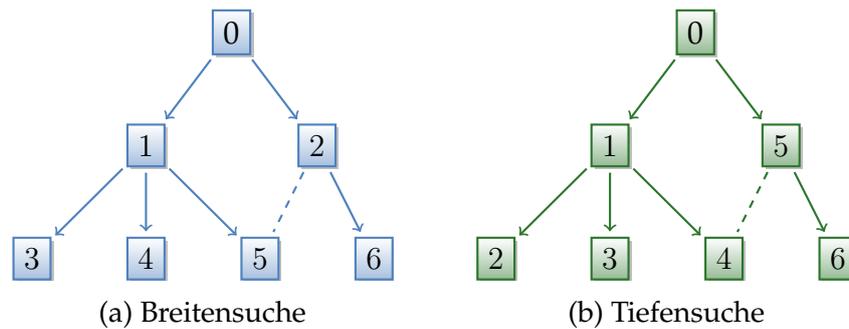


Abbildung 2.17: Unterschied in der Suchreihenfolge von Breiten- und Tiefensuche

sowie Tiefensuche (engl.: depth-first search, DFS) [CLR+09]. Beide Verfahren traversieren den Graphen entlang der Kanten. Knoten werden jeweils nur einmal besucht. Somit spannen beide Algorithmen ausgehend vom Startzustand eine Baumstruktur auf. Die Algorithmen unterscheiden sich in der Reihenfolge in der die Knoten eines Graphs in die Suche einbezogen werden. Abbildung 2.17 zeigt exemplarisch den Unterschied in der Abarbeitungsreihenfolge. Entlang der gestrichelt dargestellten Kanten wird die Suche nicht fortgesetzt, da jeder Knoten nur einmal besucht wird.

Durch die systematische Abarbeitung des gesamten Graphen ergibt sich eine maximale Laufzeit von:

$$\mathcal{O}(|V| + |E|) \quad (2.8)$$

Heuristikbasierte Suchalgorithmen

Eine Verbesserung des Laufzeitverhaltens kann erreicht werden, wenn Informationen über die gewählte Kostenfunktion bekannt sind. Die A*-Suche (A*S) [HNR68] erfolgt gezielt in Richtung des Zielknotens. Dabei arbeitet sie ähnlich wie BFS oder DFS, jedoch wird immer der Knoten als nächstes expandiert, der auf einem Pfad mit den aktuell kleinsten zu erwartenden Kosten liegt. Diese Kosten $f(v)$ setzt sich zusammen aus den bisherigen minimalen Kosten $g(v)$ von v_s zum Knoten v und einer Schätzung $h(v)$ der Kosten von diesem Knoten bis zu einem Zielknoten v_z .

$$f(v) = g(v) + h(v) \quad (2.9)$$

Abbildung 2.18 verdeutlicht die Zusammenhänge der verschiedenen Kostenfunktionen.

Jedem Knoten werden die minimalen Kosten $g(v_i)$ zur Erreichung dieses Knotens zugeordnet. Diese sind zu Beginn der Suche für jeden Knoten unendlich groß und werden beim Besuchen des Knotens durch den Suchalgorithmus neu bestimmt. Da Knoten über unterschiedliche Pfade erreicht werden können, kann $g(v_i)$ im Laufe der Suche mehrfach nach unten korrigiert werden. Wird z. B. in dem in

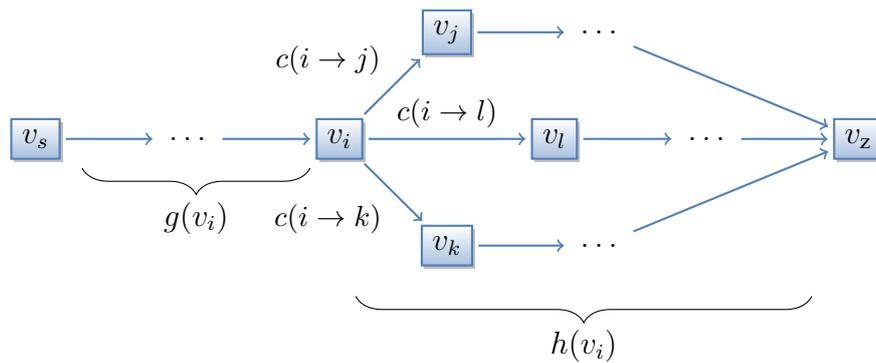


Abbildung 2.18: Anwendung einer Heuristik in der A*-Suche. Die kleinsten zu erwartenden Kosten $f(v_i)$ für den Knoten v_i setzen sich zusammen aus den bisherigen Kosten $g(v_i)$ und den geschätzten Kosten bis zum Zielknoten v_z , welche durch $h(v_i)$ bestimmt werden.

In Anlehnung an [Ham15]

Abbildung 2.18 dargestellten Teilgraph mit Kosten auf den Kanten der Knoten v_i expandiert, so werden die Kosten von v_j berechnet als:

$$g(v_j) := \min(g(v_j), g(v_i) + c(v_i \rightarrow v_j)) \quad (2.10)$$

Für die Auswahl des als nächstes zu expandierenden Knotens wird eine Heuristik h verwendet, die die Kosten $h^*(v)$ eines Pfads ausgehend von einem Knoten v hin zum Zielknoten abschätzt. Die A*-Suche ist optimal, wenn die Heuristik gewissen Eigenschaften genügt:

sicher

$$\forall v \in V : h(v) = \infty \Rightarrow h^*(v) = \infty$$

zielführend

$$\forall v \in V_{\text{Ziel}} : h(v) = 0$$

zulässig

$$\forall v \in V : h(v) \leq h^*(v)$$

monoton

$$\forall (v, v') \in E : h(v) \leq h(v') + c(v \rightarrow v')$$

Durch diese geschickte Wahl des nächsten zu expandierenden Knotens wird die Suche im Regelfall beschleunigt, ohne die Optimalität der Lösung zu verlieren. Es existiert kein Algorithmus, der unter Verwendung der gleichen Heuristik ein besseres Laufzeitverhalten bei gleichzeitiger Optimalität der Lösung aufweisen würde. [HNR68]

2.9.2 Geometrische Planer

Bei geometrischen Planern liegt ein kontinuierlicher Zustandsraum vor. Der Raum ist in der Regel unterteilt in Bereiche, die zu einem Plan gehören können (Freiraum) und solche, die nicht zu einem gültigen Plan gehören (Hindernisse). Aufgrund der Berechenbarkeit kann jedoch nur eine diskrete Menge an Zuständen in der Planung berücksichtigt werden. Planungsverfahren lassen sich anhand der verwendeten Diskretisierung des Freiraums unterscheiden. Im Folgenden werden verschiedene Planungsansätze vorgestellt.

Bei sampling-basierten Planern wird der diskrete Planungsraum zur Planungszeit ausgehend von einer Menge möglicher Ausgangszustände und Aktionen bzw. Aktionsketten bestimmt. Bekannte Vertreter der sampling-basierten Planer sind die rapidly-exploring random trees (RRTs). Hierbei wird beginnend mit einem Startzustand ein Suchbaum aufgebaut, der den Suchraum exploriert. Ausgehend von schon betrachteten Zuständen werden mittels einer Menge von Aktionen neue Zustände bestimmt und in den Baum aufgenommen. Liegt ein Zustand nahe genug am Zielzustand kann die Suche beendet werden oder durch weitere Iterationen nach einem besseren Ergebnis gesucht werden. Das Verfahren ist somit asymptotisch optimal.

Auch bei diesem Verfahren kann die Suche durch Ausnutzung einer Heuristik beschleunigt werden (RRT*). Eine weitere Option zur Beschleunigung ist der simultane Aufbau von zwei Suchbäumen, einer vom Startzustand aus und einer vom Zielzustand aus (RRT-connect). Wird in beiden Bäumen der gleiche oder ähnliche Zustand erreicht, können sie verbunden werden und ein Plan abgeleitet werden. Eine Verknüpfung beider Ansätze, die die jeweiligen Vorteile vereint, stellt das Verfahren RRT*-connect dar [KOH+15].

Eine andere Möglichkeit ist es, den Suchraum vor der Planung zu diskretisieren und in eine Graphenstruktur umzuwandeln, um dann Algorithmen zur Graphensuche anwenden zu können. Der Raum kann z. B. durch ein äquidistantes Gitter angenähert werden und jeder freien Gitterzelle kann ein Knoten in der Graphenstruktur zugeordnet werden. Knoten benachbarter Zellen werden durch Kanten verbunden. Dabei ergeben sich die typischen Probleme einer äquidistanten Diskretisierung. Wird zu grob diskretisiert, können existierende Lösungen ggf. nicht gefunden werden. Je feiner jedoch die Diskretisierung gewählt wird, desto größer ist der Aufwand zur Berechnung einer Lösung.

In zweidimensionalen Planungsräumen können Voronoi-Diagramme verwendet werden, um den Suchraum einzuschränken [TS89]. Voronoi-Diagramme beschreiben Orte im Raum, die zu zwei oder mehr Hindernissen den gleichen Abstand haben. Sie stellen somit eine topologieerhaltende Reduktion des Suchraums dar. Im 2-dimensionalen bilden die Diagramme eine Gitterstruktur. Diese kann in einen Graphen umgewandelt werden und so zur Pfadsuche verwendet werden.

Das Sichtgraph-Verfahren ist eine weitere Methode den Suchraum zu reduzieren. Hierbei wird davon ausgegangen, dass Hindernisse durch Polygone dargestellt

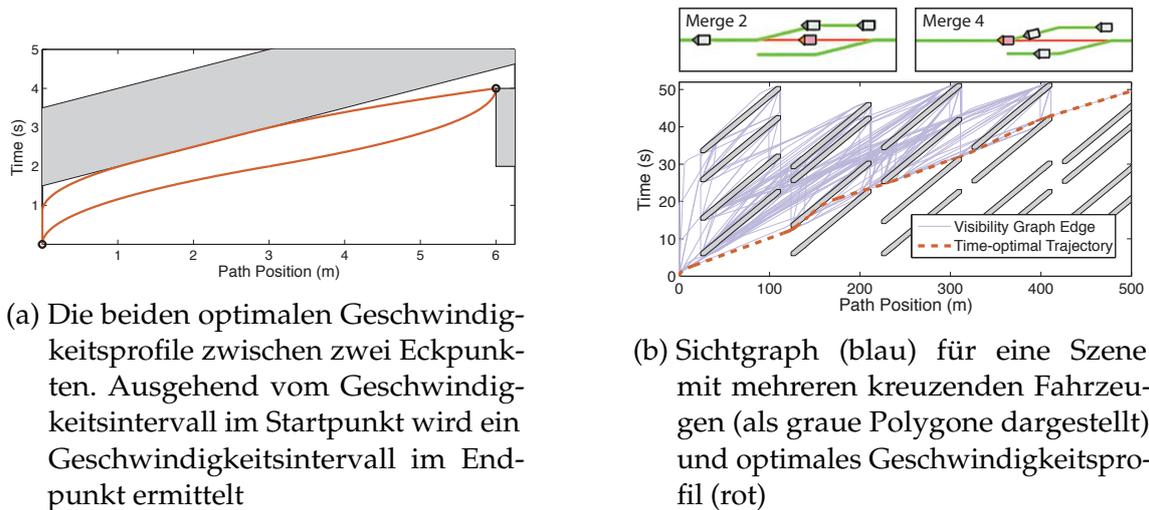
werden, deren Rand zum Freiraum gehört. Zum Aufbau eines Sichtgraphen werden alle Polygonecken sowie Start- und Zielpunkt der Planung in Knoten umgewandelt. Zwei Knoten werden dann durch eine Kante verbunden, wenn es zwischen den zugehörigen Polygonecken eine Sichtverbindung gibt, diese also durch eine Strecke verbunden werden können, die kein Hindernis-Polygon schneidet. Im Gegensatz zu der äquidistanten Diskretisierung sind sowohl die Methode der Voronoi-Diagramme als auch das Sichtgraphverfahren vollständig. Wenn ein Plan vom Start- zum Zielpunkt existiert, dann gibt es auch einen Pfad durch den entstehenden Graph.

Das bisher vorgestellte Sichtgraph-Verfahren geht von der Möglichkeit der holonomen Bewegung im Zustandsraum aus, wie es z. B. bei der Bewegungsplanung für holonome Roboter der Fall ist. Mit leichten Modifikationen lässt sich das Verfahren jedoch auch auf die Planung eines zeitoptimalen Geschwindigkeitsprofils auf einem gegebenen Pfad anwenden [JH13]. Als Zustandsraum wird hier die Ort-Zeit-Ebene gewählt. Dynamische Hindernisse, die den Pfad des Ego-Fahrzeugs kreuzen oder zeitweise überlappen, können als Polygone in der Ort-Zeit-Ebene dargestellt werden. Somit lässt sich die Planung des Geschwindigkeitsprofils auf eine Planung in einem zweidimensionalen Planungsraum abbilden, die mit Hilfe des Sichtgraph-Verfahrens gelöst werden kann.

Stückweise lineare Bewegungen in der Ort-Zeit-Ebene, wie sie beim Sichtgraph-Verfahren entstehen, können jedoch nicht von einem massebehafteten Objekt ausgeführt werden. Die Geschwindigkeit des Objekts, die durch die Steigung in der Ort-Zeit-Ebene repräsentiert wird, wäre stückweise konstant und an den Übergängen zweier Streckensegmente käme es zu unendlich großen Beschleunigungen. Daher wird für dieses Verfahren davon ausgegangen, dass die Geschwindigkeit v stetig und beschränkt und die Beschleunigung a beschränkt ist. Bewegungen können in der Ort-Zeit-Ebene in Richtung beider Achsen nur in positiver Richtung stattfinden.

Zu jedem Eckpunkt von Hindernissen wird nun die Erreichbarkeit von anderen Eckpunkten aus bestimmt. Dabei werden für jeden Vorgängerpunkt zwei optimale Geschwindigkeitsprofile erstellt. Eines, das den Zielpunkt so langsam wie möglich (mit geringst möglicher Steigung) erreicht und eines, das diesen so schnell wie möglich erreicht (mit größtmöglicher Steigung). Aus diesen beiden kann dann für den Punkt ein mögliches Geschwindigkeitsintervall bestimmt werden. Abbildung 2.19a zeigt exemplarisch die zwei beschränkenden Geschwindigkeitsprofile zwischen zwei Knoten des Sichtgraphen. Anschließend kann entgegen der Pfadrichtung das genaue Geschwindigkeitsprofil innerhalb der Intervalle abgeleitet werden. Der resultierende Sichtgraph sowie das ermittelte Geschwindigkeitsprofil sind für ein Beispiel in Abbildung 2.19b in blau bzw. rot dargestellt. Aufgrund der gleichzeitigen Betrachtung des Pfads, der Geschwindigkeit und der Zeit wird das Planungsverfahren von seinem Autor als Pfad-Geschwindigkeit-Zeit-Planer (engl.: path-velocity-time-planner, PVT-Planer) bezeichnet.

Das Verfahren ist sehr schnell und es kann gezeigt werden, dass auf diese Weise ein zeitoptimales Geschwindigkeitsprofil entsteht. Im Rahmen dieser Arbeit wird



(a) Die beiden optimalen Geschwindigkeitsprofile zwischen zwei Eckpunkten. Ausgehend vom Geschwindigkeitsintervall im Startpunkt wird ein Geschwindigkeitsintervall im Endpunkt ermittelt

(b) Sichtgraph (blau) für eine Szene mit mehreren kreuzenden Fahrzeugen (als graue Polygone dargestellt) und optimales Geschwindigkeitsprofil (rot)

Abbildung 2.19: Bestimmung eines zeitoptimalen Geschwindigkeitsprofils mittels eines Sichtgraph-Verfahrens

Bildquelle: [JH13] © 2013 IEEE

dieses Verfahren für die Dynamikvalidierung von Manöverketten eingesetzt.

2.9.3 Homotopiebasierte Planung

Innerhalb einer Verkehrsszene lassen sich Fahrzeugtrajektorien in verschiedene Homotopieklassen einordnen. Zwei Trajektorien zwischen gleichen Start- und Endpunkten sind Elemente der selben Homotopieklasse, wenn sie sich jeweils durch eine kontinuierliche Deformation auf die andere abbilden lassen ohne ein Hindernis zu schneiden [BLK12]. Durch die Anwesenheit von Hindernissen wie andere Verkehrsteilnehmer existieren im Allgemeinen mehrere abzählbar viele Homotopieklassen.

In [BTZ+15; Ben15] werden diese Homotopieklassen als Manövervarianten bezeichnet. In strukturierten Verkehrsszenen können Manövervarianten durch die Begegnungsreihenfolge mit anderen Verkehrsteilnehmern unterschieden werden. In Abbildung 2.20 ist eine Beispielszene mit einem Ego-Fahrzeug (weiß) und drei weiteren Fahrzeugen dargestellt. Entsprechend der kombinatorischen Begegnungsreihenfolgen ergeben sich hier drei Manövervarianten. Das Fahrzeug kann die zwei vorausfahrenden Fahrzeuge vor dem Gegenverkehr überholen, es kann zwischen ihnen einscheren, um den Gegenverkehr passieren zu lassen oder es kann warten, bis der Gegenverkehr vorbei ist, und dann überholen. Eine Unterscheidung zwischen letzterer Variante und einem Nichtüberholen ist hierbei nicht vorgesehen und wird lediglich über die Wahl des Geschwindigkeitsprofils abgebildet.

In unstrukturierten Verkehrsszenen können mittels einer Diskretisierung des Raum-Zeit-Volumens und einer darauf aufbauenden Mittelachsentransformati-

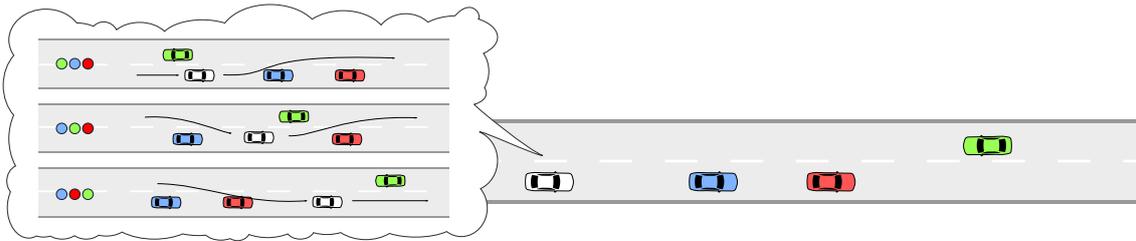


Abbildung 2.20: Manövervarianten für ein Ego-Fahrzeug in einer Verkehrsszene mit drei weiteren Fahrzeugen können anhand der Begegnungsreihenfolge unterschieden werden

Bildquelle: [Ben15]

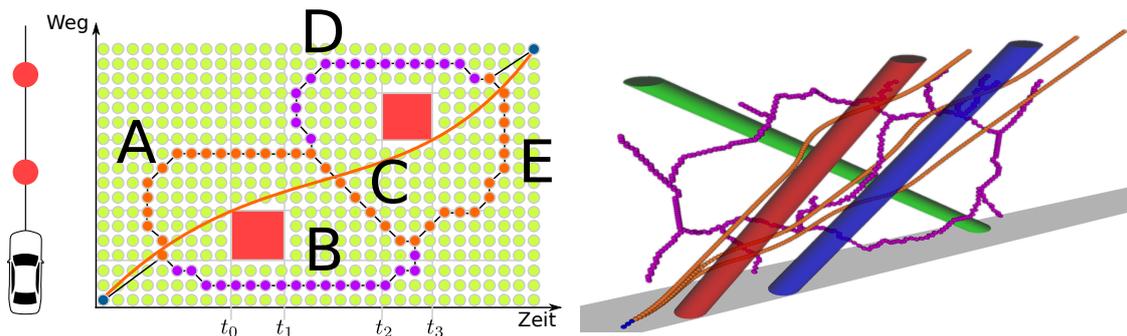


Abbildung 2.21: Beispielhafte Skelettierung eines Planungsraums. Links: Skelettierung eines zweidimensionalen Planungsraums (orange, magenta) sowie eine resultierende Trajektorie nach der lokalen Optimierung des orangenen Pfads. Rechts: Skelettierung des diskretisierten Raum-Zeit-Volumens aus dem Beispiel aus Abbildung 2.20.

Bildquelle: [Ben15]

on Repräsentanten verschiedener Homotopieklassen gefunden werden [Ben15]. Diese Trajektorien können dann als Initialisierung für eine lokale Optimierung verwendet werden. Abbildung 2.21 zeigt eine beispielhafte Skelettierung eines zweidimensionalen Planungsraums sowie die Skelettierung der Szene aus Abbildung 2.20.

Zusätzlich wird noch die Seite des Passiervorgangs benötigt. Für einfache Verkehrsszenarien wie eine Straße mit zwei Spuren und drei Fahrzeuge lassen sich somit die Manövervarianten herleiten. Mithilfe der Manövervarianten können Randbedingungen für einen Trajektorienplaner generiert werden, so dass jeweils eine lokal optimale Trajektorie für die gefundenen Homotopieklassen erzeugt werden kann.

2.10 Zusammenfassung und Fazit

Ein autonomes Fahrzeug kann als kognitives System mit Komponenten zur Perzeption, kognitiven Entscheidung und Ausführung der eigentlichen Fahraufgabe aufgefasst werden. Forschung in diesem Bereich gibt es schon seit mehreren Jahrzehnten. So kann die hier vorgestellte Arbeit auf umfangreiche Grundlagen und Vorarbeiten aus dem wissenschaftlichen und technischen Bereich zugreifen. Zur Evaluierung der Konzepte im realen Fahrversuch kann z. B. das Forschungsfahrzeug CoCar des FZI mit seiner umfangreichen Sensorik und vollständiger automatisierter Aktuierung eingesetzt werden. Auf diesem stehen Algorithmen zur Perzeption und Prädiktion der Verkehrsszene zur Verfügung, die die Sensordaten in Form von Objektlisten aufbereiten. Dabei können Unschärfen und Unsicherheiten in der Perzeption und Prädiktion durchgängig probabilistisch modelliert werden.

Im Bereich der Verkehrsszenenrepräsentation existiert ein breites Spektrum von sensordatennahen subsymbolischen Repräsentationen über symbolische bis hin zu semantischen Darstellungen. Für die Manöverentscheidung bieten semantische Darstellungen wie Ontologien eine passende Abstraktionsebene. Ein wichtiger Bestandteil der Verkehrsszenenrepräsentation ist die Modellierung der Straßengeometrie und -topologie. Hier kommen zwei Darstellungsformen, eine auf Straßenebene (OSM) und eine auf Fahrspurebene (Lanelets), zum Einsatz.

Für die Aktionsentscheidung in einem technisch kognitiven System existieren unterschiedliche Ansätze, die jeweils Stärken und Schwächen aufweisen. Strategiebasierte Verfahren können in der Regel Fahrentscheidungen sehr schnell treffen. Sie erfordern jedoch einen großen Aufwand bei der Erstellung, da z. B. bei Zustandsautomaten alle auftretenden Zustände und möglichen Aktionen modelliert werden müssen. Lernverfahren wie künstliche neuronale Netze ermöglichen einen automatisierten Aufbau der Strategie, benötigen jedoch eine große Menge an Trainingsdaten. Entscheidungen von Lernverfahren können nur schwer bis gar nicht nachvollzogen werden, was das Analysieren und Beheben von Fehlern erschwert.

Planungsverfahren explorieren den möglichen Aktionsraum und bieten das Potential einer längeren Vorausschau. Je nach Wahl des Planungsraums sind diese Verfahren jedoch Laufzeitintensiv. Für die Berechnung eines optimalen Geschwindigkeitsprofils unter Berücksichtigung dynamischer Randbedingungen stellt eine spezielle Form eines Sichtgraphverfahrens ein global optimales, vollständiges und schnelles Verfahren dar. Es wird im Rahmen dieser Arbeit zur Dynamikvalidierung von Manöverketten eingesetzt. Arbeiten zur Reduktion der kombinatorischen Komplexität während der Planung zeigen, dass rein geometrische Planer bei der Aktionsplanung im Straßenverkehr an die Grenzen der Berechenbarkeit stoßen. Abhilfe kann hier eine vorgelagerte Aktionsplanung auf einem höheren Abstraktionsniveau bieten, wie sie in dieser Arbeit vorgeschlagen wird.

3 Semantische Planung zur Manöverentscheidung

Um eine bestimmte Verkehrssituation zu bewältigen, müssen gewisse diskrete Entscheidungen über den weiteren Fahrverlauf getroffen werden. Soll z. B. einem Fahrzeug gefolgt oder soll es überholt werden, kann in eine Kreuzung eingefahren oder muss an der Haltelinie zunächst angehalten werden? Je nach Entscheidung werden verschiedene Fahrmanöver ausgeführt.

Im Allgemeinen kann die Fahraufgabe als Planungsproblem eines mobilen Agenten angesehen werden. Einige der hierzu existierenden Ansätze wurden in Kapitel 2.9 vorgestellt. Viele der geometrischen Planer (siehe Kapitel 2.9.2) können zwar im Straßenverkehr eingesetzt werden, haben ihre Stärken jedoch in wenig strukturierter Umgebung wie Parkplätzen. Vor allem die Arbeiten im Bereich der Homotopieklassen bzw. Manövervarianten (siehe Kapitel 2.9.3) zeigen, dass rein geometrische Planer in Situationen mit komplexerer Topologie wie Verkehrsszenen an ihre Komplexitätsgrenze stoßen und zusätzliches Wissen über den Aufbau der Szene eingebracht werden muss. In dieser Arbeit wird daher eine Planung auf einem semantischen Zustandsraum als Ansatz zur Manöverentscheidung vorgeschlagen, die die Ableitung optimaler Manöver bzw. Manöversequenzen für die Fahrerassistenz und das automatisierte Fahren ermöglicht.

Das Hauptziel der Einführung einer Manöverentscheidung ist die Reduktion der geometrischen Planungskomplexität zur Erfüllung der Fahraufgabe. Dabei muss ausgehend von einer Darstellung der aktuellen Situation eine Einschränkung des Planungsraums oder der darauf sinnvollen Aktionen erfolgen. Dieses Ziel ist nicht nur im Kontext hoch automatisierter Fahrzeuge und dabei zum Einsatz kommender geometrischer Planungsverfahren relevant, sondern auch im Rahmen verschiedener Fahrerassistenzsysteme, die den Fahrer aktiv oder passiv bei seiner Fahrentscheidung unterstützen. Die verschiedenen Anwendungsfelder machen eine Einsatzmöglichkeit in verschiedenen Systemarchitekturen und die Einbettung in bereits bestehende Architekturen notwendig.

In diesem Kapitel wird das Konzept der semantischen Manöverentscheidung detailliert vorgestellt. Die Manöverentscheidung soll im Kontext anderer Komponenten für Fahrerassistenz und automatisiertes Fahren eingesetzt werden. Die sich hierdurch ergebenden Randbedingungen und strukturellen Vorgaben werden in Kapitel 3.1 dargestellt. Aufbauend auf einer Definition des Manöverbegriffs (Kapitel 3.2) werden weitere Anforderungen an eine Manöverentscheidung (Kapitel 3.3) abgeleitet. Anschließend werden notwendige Komponenten im Kontext der zugrunde liegenden Systemarchitektur in Kapitel 3.4 vorgestellt und detailliert.

3 Semantische Planung zur Manöverentscheidung

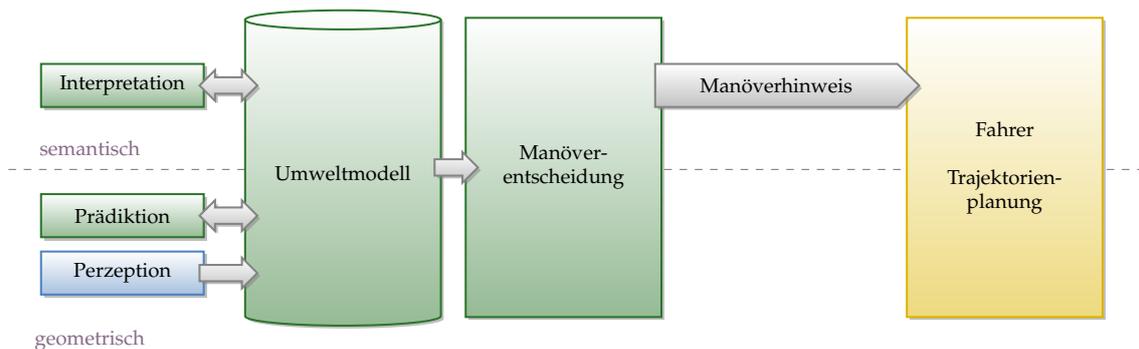


Abbildung 3.1: Horizontale Sicht auf die konkretisierte Systemarchitektur. Sie basiert auf der allgemeinen Systemarchitektur für kognitive Agenten (siehe Abbildung 2.1). Die Komponenten lassen sich in einen geometrischen Bereich (unten) und einen semantischen Bereich (oben) aufgliedern.

Abschließend wird die Eignung einer semantischen Planung für die Manöverentscheidung dargestellt (Kapitel 3.5).

3.1 Einbettung in die Systemarchitektur

Die Manöverentscheidung muss sich in eine Gesamt-Systemarchitektur einfügen. Aufbauend auf der klassischen Verarbeitungskette kognitiver Agenten (siehe Abbildung 2.1) wird hier eine Konkretisierung vorgestellt. Sie ist in Abbildung 3.1 dargestellt. Die Systemarchitektur besteht aus Komponenten zu Perzeption (blau), kognitiven Entscheidung (grün) und Aktionsausführung (gelb). Die Kommunikation zwischen den einzelnen Komponenten ist mit Pfeilen in grau visualisiert.

In dieser Architektur wird ein zentrales Umweltmodell verwendet, in dem alle Informationen über die Umgebung (Umfeld) und das Ego-Fahrzeug gehalten werden. Die Perzeption, also die Wahrnehmung des Umfelds und des eigenen Zustands, liefert eine vorwiegend geometrische Sicht, die im Umweltmodell zusammengeführt wird. Dynamische Informationen über andere Verkehrsteilnehmer werden über die Fahrzeugsensorik wie Kameras oder Laserscanner erfasst. Hierbei kommen Verfahren zur Objektsegmentierung, Klassifizierung und zur temporalen Fusion zum Einsatz, die geometrische Größen wie Objektpositionen und -geschwindigkeiten aus Sensordaten ableiten. Auf Basis der gesammelten Daten über die Dynamik der Szene kann mittels Verfahren zur Prädiktion das Umweltmodell mit einer Schätzung der zukünftigen Veränderung der Umwelt angereichert werden. Eine Interpretation leitet aus den geometrischen Daten des Umweltmodells semantische Zusammenhänge ab.

Auf den Daten des Umweltmodells basiert die Manöverentscheidung. Diese liefert Hinweise zu den zu fahrenden Manövern (z. B. Manöversequenzen) an einen Fahrer oder im Falle eines automatisierten Fahrzeugs z. B. an eine nachgelagerte

Trajektorienplanung. Eine explizite Manöverentscheidung z. B. als dedizierte Komponente einer Systemarchitektur hat gegenüber einer impliziten Entscheidung während einer Bewegungsplanung den Vorteil, dass diskrete Entscheidungen, wie sie im Straßenverkehr getroffen und teilweise auch z. B. durch Lichtsignale kommuniziert werden müssen, direkt verfügbar sind und nicht erst aus einer geplanten Trajektorie extrahiert werden müssen.

Entsprechend dem Abstraktionsniveau der verarbeiteten bzw. bereitgestellten Daten können die Module und Kommunikationswege in zwei Bereiche unterteilt werden: einen geometrischen und einen semantischen. So sind die von Perzeption und Prädiktion bereitgestellten Daten vorwiegend geometrischer Art. Erst die Interpretation reichert die Daten mit semantischer Information an. Dies spiegelt sich auch in der übergreifenden Darstellung des Umweltmodells wieder. Die Kommunikation zwischen Manöverentscheidung und Fahrer bzw. Trajektorienplanung verläuft überwiegend auf semantischer Ebene. Es sind jedoch auch je nach Anwendung geometrische Informationskanäle denkbar (siehe hierzu Kapitel 3.4).

3.2 Definition Manöver

Bisher wurde der Begriff des Fahrmanövers auf intuitive und abstrakte Weise zur Darstellung der Konzepte verwendet. Für eine weitere, vertiefende Betrachtung ist eine genaue Definition notwendig. Im allgemeinen Sprachgebrauch ist ein Fahrmanöver nicht klar definiert. So wird z. B. von einem Überholmanöver gesprochen, ein Spurwechsel kann jedoch sowohl als eigenständiges Fahrmanöver als auch als Teil eines Überholmanövers aufgefasst werden. Eine mögliche Definition ist:

„Ein Fahrmanöver ist eine Aktion bzw. Aktionsfolge [...] eines Verkehrsteilnehmers, welche die aktuelle Fahrsituation in eine neue Fahrsituation überführt.“ [Sch09]

Diese Definition zeigt eine wesentliche Eigenschaft eines Fahrmanövers. Durch ein Fahrmanöver ändert sich die Fahr- bzw. Verkehrssituation. Betrachtet man diese Verkehrssituation als eine Konstellation der beteiligten Verkehrsobjekte, also ihrer Relationen zueinander, so lässt sich ein Fahrmanöver im Rahmen dieser Arbeit wie folgt definieren:

Definition 3.2.1: Fahrmanöver

Ein Fahrmanöver eines Verkehrsteilnehmers ist eine abstrakte Aktion, die eine oder mehrere Relationen des Verkehrsteilnehmers zu anderen Szenenobjekten ändert.

Legt man diese Definition zugrunde, so kann ein Überholmanöver als ein Fahrmanöver betrachtet werden, bei dessen Ausführung sich ein Verkehrsteilnehmer nicht

mehr hinter, sondern vor einem anderen befindet. Es lässt sich jedoch auch als eine Sequenz aus Manövern betrachten: Spurwechsel, Vorbeifahren, Spurwechsel.

Ein Manöver ist nicht zwingend mit einer Aktion wie Beschleunigen, Bremsen oder Lenken des Verkehrsteilnehmers verbunden. Zum Beispiel kann auf einer mehrspurigen Straße das Vorbeifahren an einem langsameren Fahrzeug ein Manöver darstellen, auch wenn hierzu keine Anpassung der Geschwindigkeit notwendig ist.

3.3 Konzeptionelle Anforderungen an eine Manöverentscheidung

Eine Manöverentscheidung muss in einer gegebenen Verkehrssituation ein mögliches Fahrmanöver ableiten können. Je nach Verkehrssituation kann es notwendig sein, nicht nur das aktuelle Manöver zu betrachten, sondern eine ganze Sequenz von Fahrmanövern (z. B. beim Überholen). Eine Manöverplanung, also die strukturierte Suche einer oder mehrerer Manöversequenzen, ist in diesem Fall und somit auch im Allgemeinen notwendig.

Die Planung solcher Manöversequenzen hat gegenüber anderen Entscheidungsverfahren weitere Vorteile. Sie ist sehr gut für unbekannte Situationen geeignet, da sie den sich jeweils ergebenden Raum explorieren kann, und somit lokal optimale jedoch global suboptimale Aktionen vermeidet. Auch sind die Entscheidungen besser nachvollziehbar, da als Bewertungsgrundlage immer ein ganzer Plan bis zum gesteckten Ziel betrachtet werden kann.

Auf Basis der definierten Ziele und der Rahmenbedingungen aus der Systemeinbettung werden nun Anforderungen an diese Manöverplanung abgeleitet.

Zielerreichung

Die Manöverplanung selbst muss zielführend sein. Das heißt, sie muss das Fahrzeug seinem Ziel näher bringen. Ein Fahrziel in Form eines geometrischen Punktes auf der Karte oder einem bestimmten Fahrspursegment kann z. B. vom Fahrer vorgegeben werden oder von einer Routenplanung (z. B. Navigationsgerät) als Teilziel bereitgestellt werden.

Vollständigkeit

Um eine Funktion in allen Verkehrssituationen sicherzustellen, muss die Manöverplanung unter gegebenen Randbedingungen wie Sicherheit und Beschränkungen der Fahrzeugdynamik vollständig sein. Das heißt, falls es eine sinnvolle Manöversequenz gibt, muss auch eine Manöversequenz gefunden werden. In Kombination mit einer anschließenden geometrischen Planung ist dies besonders relevant.

Globale Optimalität

Um eine zufriedenstellende Fahrperformanz zu ermöglichen und auch im

Bereich der Fahrerassistenz einem erfahrenen Fahrer einen Mehrwert bieten zu können, muss die Manöverplanung bezüglich einer gewünschten Metrik wie z. B. Fahrzeit global optimale Pläne erzeugen.

Manöversequenzalternativen

Aufgrund von unvorhergesehenen Veränderungen in der Umgebung können geplante Manöversequenzen zur Ausführungszeit nicht mehr fahrbar sein. Für diesen Fall ist es vorteilhaft, wenn der Trajektorienplaner auf eine alternative für die neue Situation passende Manöversequenz wechseln kann. Daher muss die Manöverplanung in der Lage sein, für eine Situation mehrere Manöversequenzen bereitzustellen.

Dynamische Fahrbarkeit

Die bereitgestellten Manöversequenzen sollen im Allgemeinen dynamisch fahrbar sein. Eine dynamische Fahrbarkeit kann nicht immer zugesichert werden, da dies bedeuten würde, eine genaue geometrische Planung der Manöversequenz durchzuführen. Für die Performanz des Gesamtsystems spielt eine präzise Bewertung der Fahrbarkeit der vorgeschlagenen Manöversequenzen dennoch eine entscheidende Rolle.

Angemessene Laufzeit

An die Manöverentscheidung werden im Vergleich zur Regelung und Trajektorienplanung geringere Laufzeitanforderungen gestellt. Ein Manöver ist in der Regel längerfristig, also für den Zeitraum mehrerer Sekunden, gültig. Um ein reaktives Fahrverhalten in einer dynamischen Umgebung zu ermöglichen, muss jedoch während der Ausführung eines Manövers immer wieder die Manöverentscheidung überprüft und aktualisiert werden. Als eine sinnvolle Frequenz für die Manöverentscheidung wird daher mindestens 1 Hz angenommen.

Aufgabe der Manöverplanung ist also in einer gegebenen Verkehrssituation in angemessener Zeit eine oder wenn möglich mehrere sichere, dynamisch fahrbare, zielführende Fahrmanöversequenzen abzuleiten.

3.4 Struktur des Gesamtsystems

Aufbauend auf den konzeptionellen Anforderungen (Kapitel 3.3) wird nun die Struktur einer semantischen Manöverplanung im Kontext eines Gesamtsystems vorgestellt. Abbildung 3.2 veranschaulicht hierzu die grundlegenden Ideen und Konzepte. Aus einer geometrischen Fahrscenenrepräsentation (unten) werden mittels einer Ontologie Entitäten und Relationen (blau) abgeleitet. Darauf aufbauend wird ein diskreter semantischer Zustandsraum als gerichteter Graph abgeleitet (Mitte). Die auf diesem Graph geplante semantische Manöversequenz wird zur Validierung der Dynamik in die Raum-Zeit-Ebene (oben) projiziert. Darin wird ein mögliches Geschwindigkeitsprofil (rot) berechnet. Resultierende Fahrhinweise

3 Semantische Planung zur Manöverentscheidung

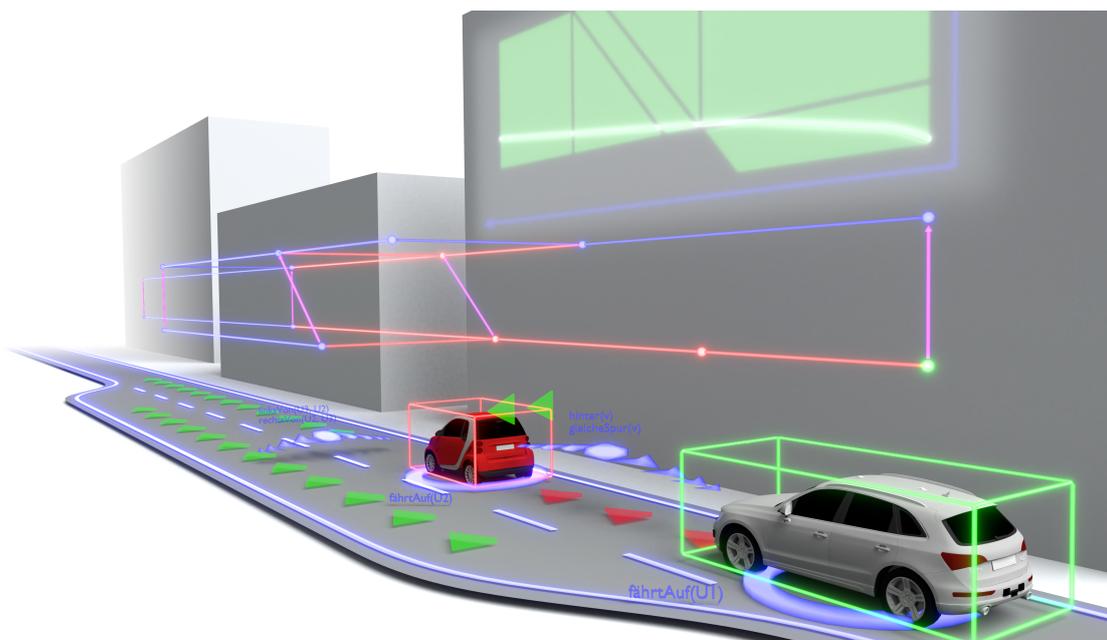


Abbildung 3.2: Konzeptionelle Darstellung der semantischen Manöverplanung. Dargestellt sind eine geometrische Fahrszenenrepräsentation (unten) und abgeleitete Relationen (blau), ein diskreter semantischer Zustandsraum als gerichteter Graph (Mitte), eine Projektion eines semantischen Plans zur Validierung der Dynamik in die Raum-Zeit-Ebene (oben) sowie ein darauf abgeleitetes Geschwindigkeitsprofil (rot).

können dem Fahrer durch Augmentierung der Umgebung angezeigt (Dreiecke unten) und für hochautomatisierte Fahrfunktionen verwendet werden.

Die Planung von Fahrmanöversequenzen ist somit in vier Teilaspekte untergliedert. Diese fügen sich harmonisch in die Gesamtsystemarchitektur ein. Abbildung 3.3 zeigt eine entsprechende Detaillierung der Systemarchitektur. Diese zeigt die in Kapitel 3.1 angesprochene Gliederung in eine geometrische und eine semantische Repräsentations- und Verarbeitungsebene. Die Trennung der Architektur in diese zwei Ebenen ist nicht scharf möglich und bietet im einzelnen Interpretationsspielraum. Die Komponenten sind hier jeweils nach ihren grundlegenden Eigenschaften eingeordnet.

Das Umweltmodell gliedert sich in zwei Teilbereiche, ein geometrisches und ein semantisches Umweltmodell. Auch die Manöverentscheidung wurde in zwei Komponenten detailliert: Eine semantische Planung und eine Manöverbewertung in Form einer Dynamikvalidierung. Wie in Kapitel 3.1 angedeutet bietet diese Detaillierung die Möglichkeit weitere Informationen in Form von passiven Aktionshinweisen oder aktiver Aktionsunterstützung an den Fahrer bzw. die Trajektorienplanung weiterzugeben.

Die einzelnen Komponenten werden im Folgenden vorgestellt. Dabei wird jeweils

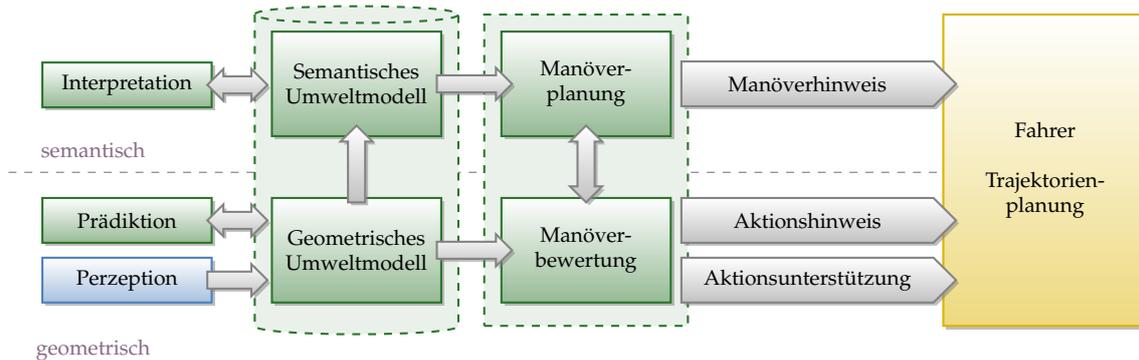


Abbildung 3.3: Detaillierung der Systemarchitektur. Die Module zur kognitiven Entscheidung (grün) sind jeweils in eine geometrische und eine semantische Komponente aufgeteilt.

eine kurze funktionale Übersicht gegeben, Anforderungen abgeleitet und eine Umsetzung vorgeschlagen.

3.4.1 Geometrische Umweltmodellierung

Das Wissen über das Ego-Fahrzeug und dessen Umfeld wird in einem Umweltmodell zusammengefasst. Die Repräsentation des Ego-Fahrzeugs wurde in den Grundlagen (Kapitel 2.3.1) schon ausführlich dargestellt. Daher liegt der Fokus im Folgenden auf der geometrischen Modellierung des Umfelds. Das geometrische Umfeldmodell kann grob in zwei Teilaspekte gegliedert werden. Statische Informationen z. B. über die Topologie und Geometrie der Straßen (siehe Kapitel 2.4) und dynamische Informationen über bewegliche Szenenobjekte wie z. B. andere Verkehrsteilnehmer (siehe Kapitel 2.5).

Statische Informationen – Karten

Statische topologische und geometrische Informationen werden in der Regel in Karten repräsentiert. An die Kartendarstellung werden aus der nachfolgenden Verarbeitung der Daten in den Komponenten zur Manöverentscheidung folgende Anforderungen gestellt:

Karteneinpassung (map matching)

Die Kartendarstellung muss die Möglichkeit bieten, dynamische Objekte auf der Karte verorten zu können. Dabei müssen Beziehungen zu Kartenobjekten wie z. B. Straßenabschnitten herstellbar sein.

Routenplanung (Routing)

Des Weiteren muss die Topologie der Karte so repräsentiert werden, dass das Ableiten einer fahrbaren Route von einem Startpunkt aus zu einem Zielpunkt möglich ist, wenn eine solche Route existiert.

Granularität

Die Granularität der Kartendarstellung muss so gestaltet sein, dass einzelne Elemente der Straßenrepräsentation eindeutige Relationen zueinander aufweisen. Das heißt, dass Relationen zwischen Straßensegmenten nicht widersprüchlich sind und jeweils für das ganze Segment und nicht nur Teile dessen gelten.

Als Kartenrepräsentation werden zwei Ansätze verwendet: Eine Repräsentation auf Straßenebene (OSM) sowie eine auf Fahrspurebene (Lanelets [BZS14]). Wie in Kapitel 2.4.1 dargestellt, können Straßennetze in dem von dem Projekt OSM verwendeten Datenformat dargestellt werden. Diese bieten den Vorteil, dass diese Karten vielerorts bereits vorhanden sind und von den Kartenservern heruntergeladen und weiterverarbeitet werden können. Durch die Darstellung von Straßen als georeferenzierte Polygonzüge bietet das Kartenmaterial die Möglichkeit der Zuordnung von Objektpositionen zu Straßen (Karteneinpassung). Da Fahrspuren häufig nur als Eigenschaften der Straßen modelliert werden, ist eine spurgenaue Karteneinpassung nur bedingt möglich. Eine der Geometrie überlagerte Straßentopologie ermöglicht das Planen von Routen (Routenplanung). Durch die grobe Granularität der Darstellung können häufig keine eindeutigen örtlichen Relationen zwischen Fahrspursegmenten abgeleitet werden. Diese müssen somit konservativ abgeschätzt werden. Kreuzungen von Straßen oder Fahrspuren werden als Punkte dargestellt. Ihre Geometrie ist nicht genau bekannt. Dies führt dazu, dass sie eine separate Behandlung in der semantischen Planung erfordern, z. B. muss die dort eingesetzte Ontologie entsprechend modifiziert werden. Details hierzu können der entsprechenden Veröffentlichung entnommen werden [KBS+14].

Spurgenaue Karten können mit Hilfe sogenannter Lanelets modelliert werden (vergleiche Kapitel 2.4.4). Kleinere Kartenausschnitte können per Hand auf der Basis von Luftbildern erstellt werden. Als Datengrundlage können auch sogenannte HD-Karten dienen, wie sie von Kartierungsdienstleistern wie HERE¹ oder Atlatec² bereitgestellt werden. Wie in Kapitel 2.7 dargestellt können jedoch auch aus OSM-Kartenmaterial spurdetailierte Karten abgeleitet werden. Dabei muss jedoch die Geometrie der Kreuzung geschätzt werden.

Durch die geometrische Repräsentation der einzelnen Fahrspursegmente ist eine genaue Karteneinpassung von Objekten wie Verkehrsteilnehmern möglich. Die topologische Ergänzung der geometrischen Fahrspursegmente erlaubt eine Routenplanung. Das Ergebnis des Routings kann ein Pfad oder ein Korridor sein.

Lanelet-Pfad

Ein Lanelet-Pfad ist eine Menge von Lanelets, die befahren werden müssen, um von einem Start-Lanelet zu einem Ziel-Lanelet zu gelangen. Dieser Pfad genügt einem Optimalitätskriterium (z.B. kürzeste Länge).

¹ HD Maps for Autonomous Driving and Driver Assistance | HERE: www.here.com/products/automotive/hd-maps (26.03.2020)

² Atlatec: www.atlatec.de (26.03.2020)

Lanelet-Korridor

Ein Lanelet-Korridor erweitert einen Lanelet-Pfad um parallele Fahrspuren. Er enthält somit auch Fahrspuren, die alternativ zu denen des Pfads verwendet werden können, um z. B. andere Fahrzeuge zu überholen. Somit repräsentiert ein Korridor die gesamte Route, die auf Straßenebene geplant wird.

Die Granularität der Kartendarstellung ist nicht genau spezifiziert. Sie muss und kann bei der Modellierung der Karte so gewählt werden, dass für eine semantische Repräsentation eindeutige Relationen zwischen einzelnen Straßensegmenten existieren. Die genauen Relationen werden später in der Verkehrsszenenontologie definiert (vergleiche Kapitel 4.1). Anschaulich gesprochen darf sich ein Fahrspursegment z. B. nicht links und recht von einem anderen befinden. Auch müssen zwei benachbarte Fahrspursegmente auf ganzer Länge aneinander angrenzen. Um dies zu gewährleisten, können Straßensegmente bei Uneindeutigkeiten an entsprechenden Stellen aufgeteilt werden und somit eindeutige Relationen erreicht werden.

Dynamische Objekte

Dynamische Szenenobjekte werden häufig durch Objektlisten oder Belegtheitsgitter repräsentiert. Für die weitere Verwendung muss diese Repräsentation folgenden Anforderungen genügen:

Temporale Konsistenz

Objekte müssen innerhalb des gesamten Zeitintervalls ihrer Existenz im Modell über mehrere Zeitschritte hinweg eindeutig identifizierbar sein.

Klassifikation

Objekte müssen in Klassen eingeteilt werden können (z. B. Fußgänger, Autos, Randbebauung).

Dynamik

Die Bewegung eines Objekts muss bekannt sein. Zumindest seine aktuelle Geschwindigkeit muss vorliegen.

Prädiktion

Die zukünftige Trajektorie und mögliche gefahrene Routen von Verkehrsteilnehmern müssen ableitbar sein.

Aufgrund dieser Anforderungen wird für die geometrische Modellierung dynamischer Objekte des Umfelds eine objektbasierte, parametrische Darstellung verwendet. Diese kann so ausgelegt werden, dass sie den oben beschriebenen Anforderungen genügt. Gegenüber gridbasierten Ansätzen erlaubt diese eine leichtere Ableitung von semantischen Eigenschaften und Relationen.

Die Bewegung anderer Fahrzeuge und ggf. auch des Ego-Fahrzeugs wird als Trajektorie beschrieben.

Definition 3.4.1: Trajektorie

Eine Trajektorie $p(t)$ beschreibt alle Posen p eines Objektes o in Abhängigkeit der Zeit t über einem bestimmten Zeitintervall $[t_s, t_e]$. Auf der Erde kann diese z. B. durch 2-dimensionale Geokoordinaten dargestellt werden:

$$p_o : [t_s, t_e] \rightarrow \mathbb{R} \times \mathbb{R} \times \mathbb{R}, \quad t \mapsto (lon, lat, \Psi) \quad (3.1)$$

Dabei beschreiben lon und lat die geographische Länge und Breite sowie Ψ die Ausrichtung des Fahrzeugs bzw. Objekts jeweils als Winkel.

Bewegt sich ein Verkehrsteilnehmer entlang oder relativ zu einer gewissen Route, so kann dessen Position auch in Abhängigkeit zu dieser dargestellt werden (Karteneinpassung). Dies entspricht im Wesentlichen der Transformation in ein Frenet-Serret-Koordinatensystem entlang der Route. Jede Route R definiert eine Abbildung von einer eindimensionalen Position x auf der Route zu einer georeferenzierten Pose:

$$R : x \mapsto (lon, lat, \Psi) \quad (3.2)$$

oder Position:

$$R' : x \mapsto (lon, lat) \quad (3.3)$$

Diese stellt einen typischen Pfad auf der Route dar, z. B. jeweils die Mitte der befahrenen Fahrspur oder Straße. Der Verlauf von Ψ ist tangential zu R . Abbildung 3.4 veranschaulicht die geometrischen Zusammenhänge. Dargestellt ist eine Straße mit einem Fahrzeug. Mittig auf der in Fahrtrichtung rechten Spur liegt die Funktion $R(x)$ (blau). Beispielhaft ist für x_0 auch die entsprechende Ausrichtung Ψ_0 eingezeichnet.

Die Skalierung von x ist dabei so zu wählen, dass $x_1 - x_2$ der Länge der zurückgelegten Strecke zwischen $R(x_1)$ und $R(x_2)$ entspricht. In der Abbildung ist x_d in einem Abstand von d (lila) von x_0 entfernt dargestellt.

Bewegt sich ein Objekt nicht exakt auf diesem Pfad, so kann dessen Pose als Position auf der Route x , lateralem Offset y sowie einem Gierwinkel-Offset ψ dargestellt werden. In Abbildung 3.4 hat das Fahrzeug einen lateralen Offset von y_v sowie eine Gierwinkel-Offset von ψ_v .

Die Abbildung R und R' lässt sich somit entsprechend erweitern:

$$(x, y, \psi) \mapsto_R (lon, lat, \Psi) \quad (3.4)$$

Für genügend kleine y ist die Umkehrung dieser Funktion wohldefiniert³:

$$(lon, lat, \Psi) \mapsto_{R^{-1}} (x, y, \psi) \quad (3.5)$$

³ Im Kontext von Straßen ist dies für Positionen auf der Straße und nahe der Straße (z. B. auf dem Gehweg) im Regelfall gegeben

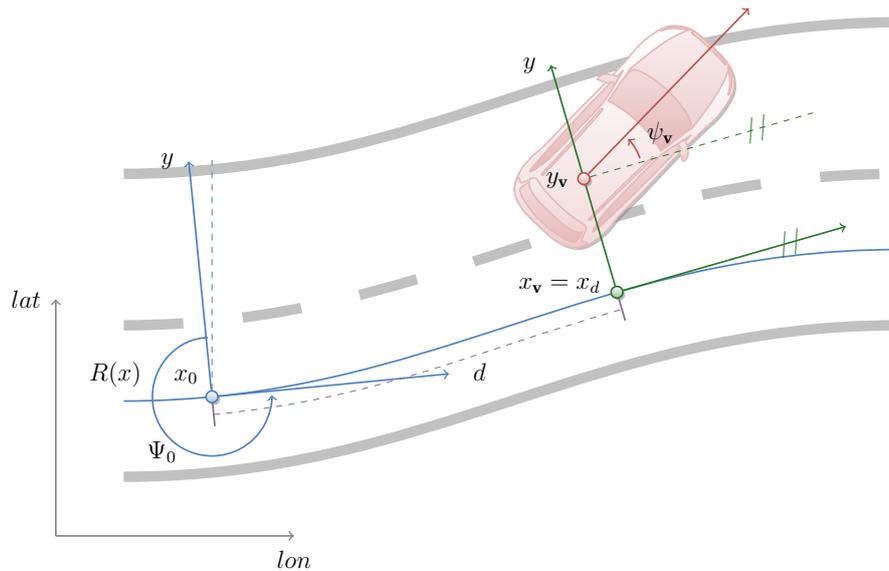


Abbildung 3.4: Beschreibung der Fahrzeugpose relativ zu einer Route $R(x)$ (blau). Die Fahrzeugpose kann in einem Frenet-Koordinatensystem, das entlang der Route verläuft, dargestellt werden.

und

$$(lon, lat) \mapsto_{R^{-1}} (x, y) \quad (3.6)$$

Eine Trajektorie lässt sich somit auch als Position und Offset zu einer Route darstellen:

$$p_o : (t) \mapsto_p (lon, lat, \Psi) \mapsto_{R^{-1}} (x, y, \psi) \quad (3.7)$$

3.4.2 Semantisches Umweltmodell

Aufbauend auf der geometrischen Szenenrepräsentation wird eine semantische Repräsentation abgeleitet. Mittels einer Ontologie wird die Verkehrsszene beschrieben. Dabei werden aus der geometrischen Darstellung Entitäten und Relationen zwischen diesen ermittelt und weiter inferiert. Für die Ableitung von Fahrmanöversequenzen wird ein semantischer Zustandsraum auf Basis der Ontologie aufgebaut. In diesem sind mögliche Zustände des Ego-Fahrzeugs als diskrete Zustände dargestellt. Transitionen zwischen den Zuständen geben mögliche Manöver an und beschreiben z. B. Verkehrsregeln.

Aus den allgemeinen konzeptionellen Zielen und Anforderungen werden für die semantische Szenenrepräsentation folgende Anforderungen abgeleitet:

Diskreter Zustandsraum

Um die Planungskomplexität zu reduzieren, soll ein diskreter Zustandsraum bereitgestellt werden, der es ermöglicht, Abfolgen von Manövern wie das

3 Semantische Planung zur Manöverentscheidung

Einfahren in eine Kreuzung oder einen Kreisverkehr, Spurwechsel oder Überholmanöver zu planen.

Informationserhaltend

Ergebnisse der Planung müssen an einen Fahrer oder eine nachgelagerte Trajektorienplanung im Kontext der Verkehrsszene kommuniziert werden können. Daher muss die Abbildung in einen diskreten Zustandsraum bezüglich dieser Anwendungen informationserhaltend sein. Es muss also domänenspezifisches Wissen über die Struktur der Szene zur Ableitung der diskreten Zustände verwendet werden wie Informationen über Fahrspuren, Kreuzungen und andere Verkehrsteilnehmer.

Nicht äquidistante Diskretisierung

Die auf dem Zustandsraum aufbauende semantische Planung soll vollständig sein. Daher muss bei der Abbildung des kontinuierlichen Fahrzeugzustands in diskrete Zustände die Diskretisierung nicht äquidistant erfolgen, um negative Effekte durch zu feine oder zu grobe Abtastung zu vermeiden. Um die Eigenschaft der Informationserhaltung zu erfüllen, soll sich die Diskretisierung somit örtlich an geometrisch-topologischen Grenzen wie z. B. Fahrspurbegrenzungen oder Sicherheitsabständen zu Verkehrsteilnehmern sowie zeitlich an Ereignissen in der Szene orientieren.

Im Folgenden wird das Konzept eines semantischen Zustandsraums eingeführt, das die oben genannten Anforderungen erfüllt. An dieser Stelle sollen zum besseren Verständnis zunächst nur die grundlegenden Konzepte des semantischen Zustandsraums dargestellt werden. Detaillierte Definitionen und ein strukturiertes Vorgehen zum Aufbau des Zustandsraumes werden in Kapitel 4 erarbeitet.

Der semantische Zustandsraum \mathbf{S} beschreibt mögliche diskrete Zustände $s \in \mathbf{S}$, die ein Ego-Fahrzeug **ego** in einer strukturierten Verkehrsszene annehmen kann. Dabei stellt ein semantischer Zustand eine mögliche relationale Beziehungen von **ego** zu anderen Objekten in der Verkehrsszene da. Aktionen bzw. hier im Speziellen Manöver, die **ego** von einem Zustand in den nächsten überführen, erfolgen entlang sogenannter Transitionen (*start* \rightarrow *ziel*). An Transitionen können Bedingungen $C_{(start \rightarrow ziel)}$ geknüpft sein, die definieren, ob und wann eine Transition gültig ist, also von **ego** ausgeführt werden kann. Die Verwendung von Transitionsbedingungen erlaubt unter anderem die Definition von Verkehrsregeln. Zustände \mathbf{S} und Transitionen \mathbf{T}_S bilden einen gerichteten Graph $G = (\mathbf{S}, \mathbf{T}_S)$, wobei \mathbf{S} als Knoten und \mathbf{T}_S als Kanten mit Eigenschaften C interpretiert werden. Es können auch Zustände modelliert werden, die von **ego** nicht erreicht werden sollen oder können, z. B. Zustände in denen es zu einer Kollision mit einem anderen Verkehrsteilnehmer käme.

Das Konzept soll im Folgenden an einem einfachen Beispiel erläutert werden. Abbildung 3.5 zeigt auf der linken Seite eine einfache Straßenszene mit einer zweispurigen Straße. Da keine weiteren Objekte in der Szene vorhanden sind, beinhaltet der Zustandsraum (rechts) genau zwei Zustände:

$$\mathbf{S} = \{\text{aufLinkerSpur}, \text{aufRechterSpur}\}.$$

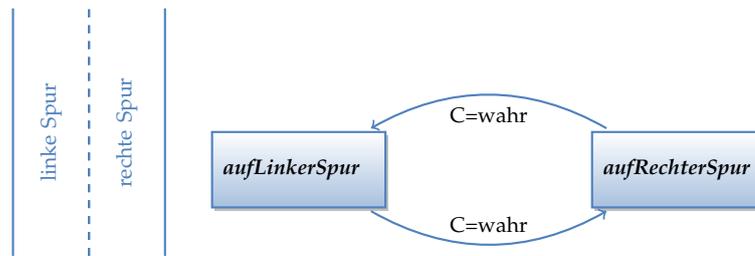


Abbildung 3.5: Einfache Beispielszene - Straße mit zwei Fahrspuren, Spurwechsel erlaubt. Graphenrepräsentation des entsprechenden semantischen Zustandsraums und der Transitionen.

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas et al.)

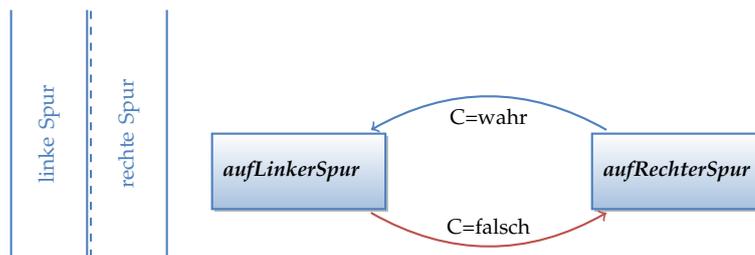


Abbildung 3.6: Einfache Beispielszene - Straße mit zwei Fahrspuren, Spurwechsel eingeschränkt. Graphenrepräsentation des entsprechenden semantischen Zustandsraums und der Transitionen.

Ein Spurwechsel ist in beide Richtungen möglich, repräsentiert durch die beiden Transitionen. In Abbildung 3.6 ist der Übergang von der linken zur rechten Spur nicht erlaubt. Dies wird durch die invalide Transition (rot) repräsentiert.

Das hier vorgestellte Konzept eines semantischen Zustandsraums erfüllt die geforderten Eigenschaften. Die vorgestellte Zustandsraumrepräsentation ist diskret. Die Diskretisierung erfolgt entlang szenenspezifischer Grenzen und ist somit nicht äquidistant. Durch eine geschickte Wahl der semantischen Beschreibung der Zustände ist es möglich, notwendige domänenspezifische Informationen über die Situation im Zustandsraum zu repräsentieren. Zur Beschreibung der semantischen Zustände kann eine Straßenszenen-Ontologie eingesetzt werden. Eine genaue Definition dieser Ontologie sowie eine strukturierte Bildungsvorschrift für den semantischen Zustandsraum wird in Kapitel 4 dargestellt. Dort wird auch die Eigenschaft der Komplexitätsreduktion untersucht (Kapitel 4.4.5).

3.4.3 Semantische Planung

Der semantische Zustandsraum kann zur Planung von Manöversequenzen eingesetzt werden. Diese Planung muss entsprechend der Anforderungen an das Gesamtkonzept zielführend und vollständig sein. Bezüglich einer gewählten Kostenfunktion sollen ein global optimaler Plan oder, um alternative Manöversequenzen

3 Semantische Planung zur Manöverentscheidung

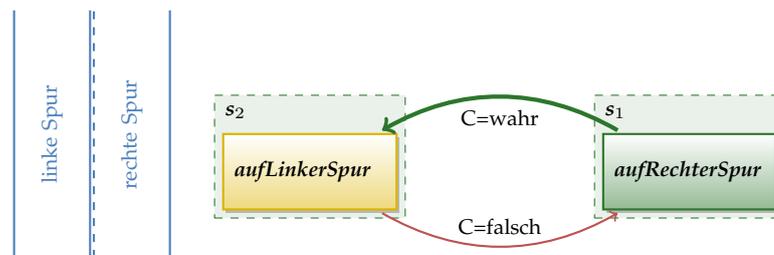


Abbildung 3.7: Einfache Beispielszene - Straße mit zwei Fahrspuren. Graphenrepräsentation des entsprechenden semantischen Planungsraums.

zen bereitstellen zu können, mehrere bzw. alle möglichen Pläne gefunden werden können.

Um eine semantische Planung durchführen zu können, wird ein entsprechender Planungsraum abgeleitet. Wie oben beschrieben, bilden Zustände \mathbf{S} und Transitionen \mathbf{T}_S einen gerichteten Graph $G = (\mathbf{S}, \mathbf{T}_S)$. Zusammen mit einem Startzustand *start* und einer Menge von Zielzuständen $Z \subset \mathbf{S}$ bildet dieser den Planungsraum $(\mathbf{S}, \mathbf{T}_S, \text{start}, Z)$. Häufig wird dieses Tupel als Zustandsraum bezeichnet. Um eine Abgrenzung zur Menge aller Zustände \mathbf{S} zu verdeutlichen, wird hier wenn notwendig der Begriff des Planungsraums verwendet.

Durch die Darstellung des Planungsraums als einen gerichteten Graphen, können zur Planung von Manöversequenzen Algorithmen zur Graphensuche eingesetzt werden. So kann z. B. ein Pfad von einem Startzustand in dem sich *ego* befindet in einen angestrebten Endzustand bestimmt werden. In dem einfachen Beispiel lässt sich so ein Spurwechsel planen. In Abbildung 3.7 ist ein resultierender Plan von *start* = *rechteSpur* (gelb) zu *linkeSpur* $\in Z$ (grün) dargestellt. Er besteht aus zwei Zuständen (s_1 und s_2) und einer Transition (grün).

Durch die Wahl eines geeigneten Algorithmus zur Graphensuche ist es möglich die oben genannten Anforderungen zu erfüllen. Verschiedene Ansätze und spezifische Erweiterungen werden in Kapitel 5 vorgestellt und bewertet.

3.4.4 Dynamikvalidierung und -evaluierung

Durch die Abstraktion des Planungsraums auf eine semantische Ebene ist die dynamische Fahrbarkeit der geplanten Manöverketten nicht direkt ableitbar. Dynamische Aspekte der Manöverketten werden daher anschließend in einer Dynamikvalidierung berücksichtigt. Primäres Ziel der Dynamikvalidierung ist es, die geplanten semantischen Manöverketten auf ihre dynamische Fahrbarkeit hin zu überprüfen. Zusätzlich können die Ergebnisse der Dynamikvalidierung als Dynamikevaluierung in die Bewertung einer Manöverkette einfließen. Daher ergeben sich folgende Anforderungen an die Dynamikvalidierung:

Vollständigkeit

Um die Eigenschaft der Vollständigkeit der semantischen Planung nicht zu

verlieren, muss auch die Dynamikevaluierung diesem Anspruch genügen. Falls eine Manöversequenz dynamisch fahrbar ist, so soll diese auch als solche validiert werden. Hierbei ist anzumerken, dass eine exakte Validierung einer genauen Trajektorienplanung entsprechen würde, und somit erst durch diese abschließend vorgenommen werden kann. Die Dynamikvalidierung muss also so ausgelegt sein, dass sie in Grenzfällen dynamische Fahrbarkeit annimmt. Im Hinblick auf Systemperformanz ist die Menge an Grenzfällen gering zu halten.

Laufzeit

Da die Dynamikvalidierung potentiell auf viele Manöversequenzen oder auch Teilsequenzen angewendet werden wird, spielt die Laufzeit hier eine wesentliche Rolle. Je nach verwendetem Suchalgorithmus in der semantischen Planung müssen hunderte bis tausende Pläne validiert werden (vergleiche Tabelle 5.1). Daher muss die maximale Laufzeit der Dynamikvalidierung in der Größenordnung von Millisekunden liegen.

Global optimal

Im Falle einer Dynamikevaluierung wird diese Anforderung direkt von der Planung übernommen, da die Dynamikevaluierung direkten Einfluss auf die Optimalität der generierten semantischen Pläne hat. An die reine Validierung wird diese Eigenschaft nicht gestellt.

Um eine Manöversequenz auf dynamische Fahrbarkeit hin zu untersuchen, wird diese zunächst in eine geeignete geo-temporale Repräsentation überführt, die Randbedingungen für ein Geschwindigkeitsprofil definiert. In einem zweiten Schritt wird falls vorhanden ein innerhalb der Randbedingungen gültiges Geschwindigkeitsprofil berechnet. Für die Evaluierung der Manöverkette muss dieses hinsichtlich verschiedener Kriterien optimal sein. Die Gewichtung der Optimierungskriterien kann dabei je nach Fahrerwunsch und Fahrzeugzustand variiert werden. Hierzu wird eine manöverbasierte Optimierung von Geschwindigkeitsprofilen verwendet. Als Längsmanöver wurden Beschleunigen, Geschwindigkeit halten, Ausrollen/Segeln, Abbremsen sowie bei Hybrid- und Elektrofahrzeugen Rekuperieren identifiziert.

In Kapitel 6 werden verschiedene Verfahren zur Geschwindigkeitsprofilgenerierung vorgestellt bewertet und ein entsprechend der Anforderungen geeignetes Verfahren ausgewählt.

3.5 Zusammenfassung und Fazit

Die Manöverplanung soll es ermöglicht, Abfolgen von Manövern wie das Einfahren in eine Kreuzung oder einen Kreisverkehr, Spurwechsel oder Überholmanöver zu planen. Die Definition 3.2.1 eines Fahrmanövers ermöglicht es, dies auf semantischer Ebene, also im Raum von Entitäten und Relationen durchzuführen. Eine genauere Betrachtung des Konzepts der semantischen Manöverplanung zeigt,

3 Semantische Planung zur Manöverentscheidung

dass diese die gesetzten Ziele erfüllt und sich in eine bestehende Systemarchitektur für automatisiertes Fahren integrieren lässt.

Die semantische Repräsentation der betrachteten Zustände impliziert eine Diskretisierung des im Allgemeinen kontinuierlichen Zustands einer Verkehrsszene. Diese Diskretisierung stellt eine Reduktion der Planungskomplexität gegenüber dem Einsatz einer rein geometrischen Trajektorienplanung dar. Je nach Auslegung der verwendeten Relationen kann die Komplexität des Planungsraums und somit auch der Planung gezielt beeinflusst werden.

Um Aktionen im Planungsraum direkt bestimmten Fahrmanövern zuordnen zu können, wird domänenspezifisches Wissen über die Struktur der Szene verwendet wie Informationen über Fahrspuren, Kreuzungen und andere Verkehrsteilnehmer. Dies kann über eine entsprechend modellierte Ontologie gewährleistet werden.

Das hier vorgestellte Konzept der semantischen Manöverentscheidung wurde hinsichtlich der in Kapitel 3.3 beschriebenen Anforderungen abgeleitet. Anforderungen wie Zielerreichung, Vollständigkeit, globale Optimalität und Manöversequenzalternativen konnten bereits auf konzeptioneller Ebene adressiert werden. Die verwendete Verkehrsszenen-Ontologie und ein strukturiertes Vorgehen zum Aufbau des vorgestellten semantischen Planungsraums wird in Kapitel 4 detailliert beschrieben. Die Sicherstellung der dynamischen Fahrbarkeit sowie die Eignung des Laufzeitverhaltens für reale Anwendungen wird nach Erarbeitung der semantischen Planung in Kapitel 5 und der Dynamikvalidierung in Kapitel 6 gezeigt.

4 Semantische Szenenmodellierung

Ziel der semantischen Szenenmodellierung ist es, einen diskreten Planungsraum für das Ego-Fahrzeug aufzubauen. Die Idee des verwendeten semantischen Zustandsraums wurde bereits in Kapitel 3.4.2 anhand eines einfachen Beispiels vorgestellt. Im Folgenden wird nun die Ableitung des Planungsraums basierend auf dem geometrischen Umweltmodell beschrieben. Dazu wird die verwendete Verkehrsszenenontologie vorgestellt (Kapitel 4.1) und ein systematisches Vorgehen zur Erstellung des Planungsraums beschrieben (Kapitel 4.3). Die Eigenschaften des Planungsraums werden anhand verschiedener Beispiele analysiert und eine theoretische Herleitung der Komplexitätsklasse der Zustandsraumgröße dargestellt (Kapitel 4.4). Zusätzlich werden Modellierungsmöglichkeiten und Auswirkungen probabilistischer Effekte diskutiert (Kapitel 4.5).

4.1 Relationale Szenenmodellierung – Verkehrsszenenontologie

Die relationale Szenenmodellierung wird aus der geometrischen Umfeldmodellierung abgeleitet, wie sie in Kapitel 3.4.1 beschrieben wurde. Dabei werden die geometrischen Entitäten wie Fahrspuren oder Umfeldobjekte sowie ihre räumlichen und logischen Beziehungen in einer Ontologie repräsentiert. Die Ontologie besteht aus Entitäten und Relationen. Es werden zweidimensionale Relationen $r(\mathbf{o}_k, \mathbf{o}_l)$ zwischen zwei Objekten \mathbf{o}_k und \mathbf{o}_l verwendet. Diese Relationen können als Prädikate aufgefasst werden, die wahr sind, wenn die Relation zwischen den Objekten besteht.

Die verwendete Ontologie ist in Abbildung 4.1 dargestellt. Die Ontologie ist zu großen Teilen eine Domänenontologie, die die Zusammenhänge zwischen Verkehrsteilnehmern und Infrastruktur abbildet. Sie ist so gestaltet, dass sie zur Generierung eines semantischen Planungsraums verwendet werden kann. Ein gemeinsames Elternelement der Entitäten wird implizit angenommen.

Als Entitäten werden statische und dynamische Szenenobjekte unterschieden. Statische Objekte sind vor allem Fahrspursegmente. Je nach Modellierung und verwendeter geometrischer Repräsentation ist es möglich Kreuzungen als separate Entität zu modellieren (vergleiche [KBS+14]). Als dynamische Objekte werden hier nur allgemein Fahrzeuge betrachtet. Eine Erweiterung und Detaillierung der Ontologie um weitere Verkehrsteilnehmer wie Fußgänger, Rad- oder Motorradfahrer als Kind von dynamischen Szenenobjekten ist jedoch möglich. Als

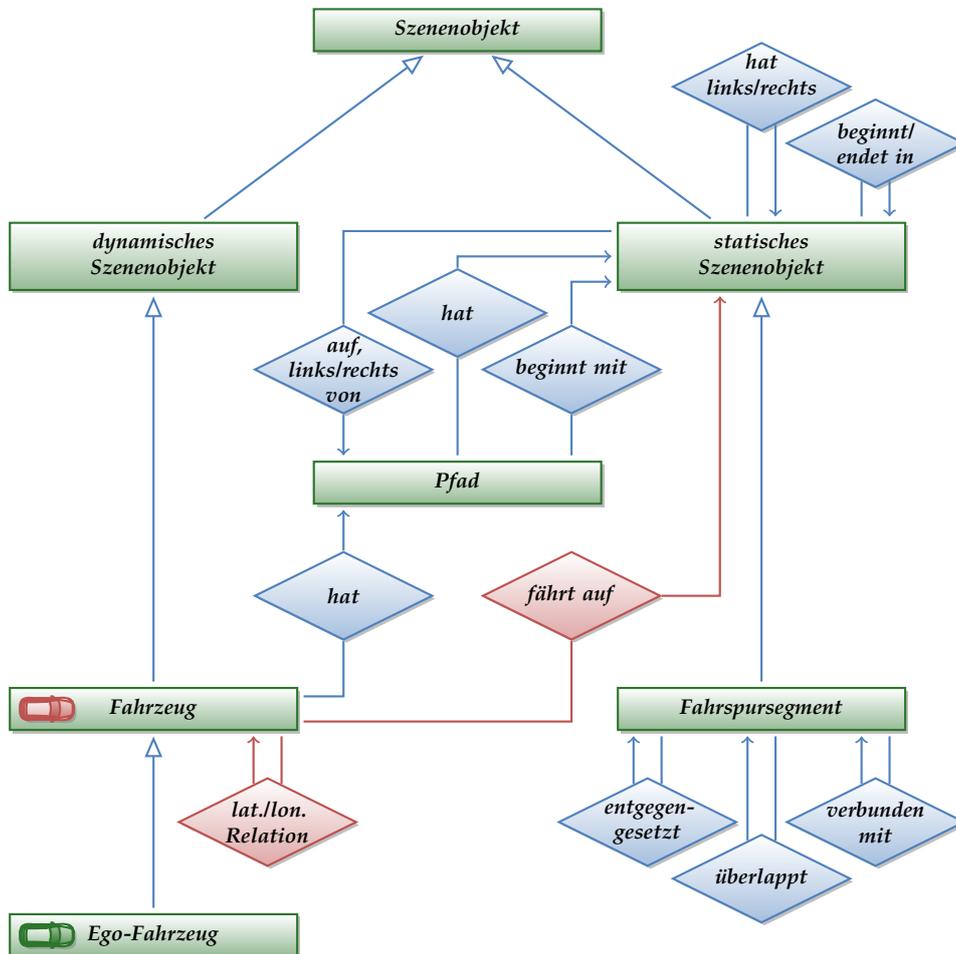


Abbildung 4.1: TBox der verwendeten Ontologie zur Verkehrsszenenbeschreibung. Relationen zwischen dem Ego-Fahrzeug und statischen Szenenobjekten (rot) werden für die Bildung des semantischen Zustandsraums verwendet.

Ähnlich veröffentlicht in [KHS+15] (Kohlhaas et al.)

spezielles Fahrzeug wird das Ego-Fahrzeug modelliert. Der *Pfad* ist als einzige Entität kein Szenenobjekt. Er repräsentiert die von einem Fahrzeug gefahrene Strecke als Menge von statischen Szenenobjekten (z. B. Fahrspursegmente).

Die meisten Relationen lassen sich direkt aus der geometrischen Szenenbeschreibung ableiten. Einige andere können über prädikatenlogische Regeln inferiert werden. Im Folgenden wird auf die Bedeutung und die Ableitung der Relationen einzeln eingegangen.

4.1.1 Relationen zwischen statischen Szenenobjekten

Die Relationen zwischen statischen Szenenobjekten (z. B. zwei Fahrspursegmenten U_k, U_l) dienen hauptsächlich der topologischen Verortung dieser.

beginnt/endet in(U_k, U_l)

Diese Relation repräsentiert die Folge von Straßensegmenten. Sind zwei Segmente mit dieser Relation verbunden, so kann von dem einen auf das andere gefahren werden.

hat links/rechts(U_k, U_l)

Die laterale Lage der Straßensegmente wird über diese Relation angegeben. Eine Befahrbarkeit wird nicht direkt impliziert (vergleiche *verbunden mit*).

Für Fahrspursegmente werden weitere Relationen zur Darstellung der Straßentopologie definiert.

verbunden mit(U_k, U_l)

Ist es möglich, zwischen zwei Fahrspursegmenten die Spur zu wechseln, so wird dies durch diese Relation angezeigt.

entgegengesetzt(U_k, U_l)

Um darzustellen, dass eine benachbarte Fahrspur in die entgegengesetzte Richtung zu befahren ist, wird diese Relation verwendet. Sie ist besonders wichtig für die Inferenz der relativen Lage zu semantischen Pfaden.

überlappt(U_k, U_l)

Überlappen sich zwei Fahrspursegmente, wird dies durch diese Relation modelliert. Diese Information kann entweder aus der Geometrie der Fahrspursegmente abgeleitet werden oder im Falle der Verwendung von Lanelets aus der entsprechenden Rolle „cross“ der Lanelets abgeleitet werden (vergleiche Kapitel 2.4.4).

Abbildung 4.2 zeigt eine Beispielszene und ausgewählte Relationen der abgeleiteten ABox. Es wird deutlich, wie die Relationen die Straßentopologie abbilden und Befahrbarkeitsanalysen auf Objektebene zulassen.

4 Semantische Szenenmodellierung

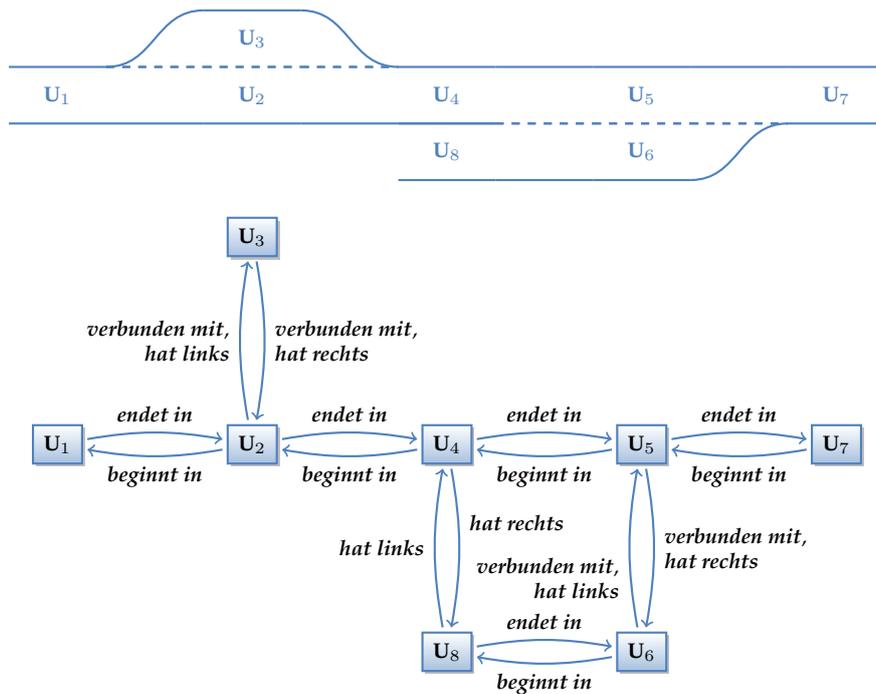


Abbildung 4.2: Beispiel ausgewählter Relationen zwischen statischen Szenenobjekten

4.1.2 Relationen zum Pfad

Der Pfad p repräsentiert den von einem Fahrzeug v gefahrenen Weg.

$hat(v, p)$

Jedem Fahrzeug wird genau ein Pfad zugeordnet. Dieser wird als statisch angenommen, kann sich jedoch je nach Wahrnehmung und Szeneninterpretation z. B. bei veränderten Gegebenheiten ändern.

$hat(p, U)$

Ein Pfad besteht aus mehreren statischen Szenenobjekten. Diese müssen zusammenhängend sein. Das heißt, jedes Element hat einen Vorgänger, der auch zum Pfad gehört, oder es ist ein Anfangselement (vergleiche *beginnt mit*). Es können auch parallel verlaufende Fahrspuren zu einem Pfad gehören, auch wenn das Fahrzeug nur auf einer der beiden Fahrspuren fährt. Formal lässt sich dieser Sachverhalt wie folgt darstellen. Sei $\mathbb{U} = \{U | hat(p, U)\}$, dann gilt:

$$\forall U \in \mathbb{U} : \textit{beginnt mit}(p, U) \vee \exists U' \in \mathbb{U} : (\textit{verbunden mit}(U, U') \vee \textit{beginnt in}(U, U'))$$

Für das Ego-Fahrzeug wird die gesamte Menge an Szenenobjekten, die auf dem Weg zu einem Navigationsziel befahren werden können, modelliert (vergleiche Lanelet-Korridor in Kapitel 3.4.1). Für alle anderen Fahrzeuge wird ein Pfad mithilfe einer Routenprädiktion bestimmt.

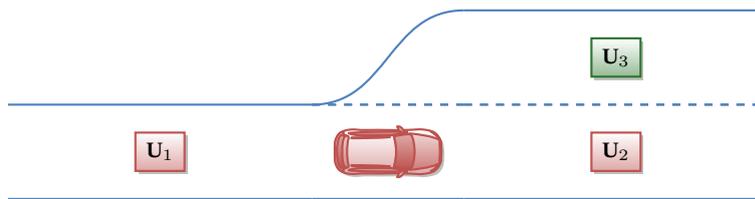


Abbildung 4.3: Straßenszenario mit einem Fahrzeug (\mathbf{v}). Entsprechend der Relationen zu seinem Pfad ($\mathbf{p} = \{\mathbf{U}_1, \mathbf{U}_2\}$) sind die statischen Szenenobjekte eingefärbt: rot - *liegt auf*(\mathbf{U}, \mathbf{p}), grün - *links von*(\mathbf{U}, \mathbf{p}).

beginnt mit(\mathbf{p}, \mathbf{U})

Um die Richtung des Pfads zu kodieren, wird das erste Element des Pfads mittels dieser Relation definiert. Häufig ist das entsprechende Fahrspursegment auch das, auf dem sich das zugehörige Fahrzeug befindet (vergleiche *fährt auf*).

auf, links/rechts von(\mathbf{U}, \mathbf{p})

Für jedes statische Szenenobjekt wird angegeben, ob es sich auf dem Pfad (*liegt auf*(\mathbf{U}, \mathbf{p})), links (*links von*(\mathbf{U}, \mathbf{p})) oder rechts (*rechts von*(\mathbf{U}, \mathbf{p})) von einem Pfad befindet. Diese Information wird verwendet, um die lateralen Relationen zwischen Fahrzeugen zu inferieren.

Da \mathbf{p} durch eine Menge von statischen Szenenobjekten repräsentiert wird, wird die folgende Mengenschreibweise für die Definition eines Pfads eingeführt:

$$\mathbf{p} = \{\mathbf{U} \mid \mathit{hat}(\mathbf{p}, \mathbf{U})\} \quad (4.1)$$

Abbildung 4.3 zeigt eine Straßenszene mit einem Fahrzeug, das der rechten Spur weiter folgt. Somit ergibt sich der Pfad $\mathbf{p} = \{\mathbf{U}_1, \mathbf{U}_2\}$. Die Fahrspursegmente des Pfads haben die Relation *liegt auf*(\mathbf{U}, \mathbf{p}) zum Pfad. Das Objekt \mathbf{U}_3 liegt links vom Pfad. In Abbildung 4.4 ist ein Kreuzungsszenario zwischen zwei einspurigen Straßen dargestellt. Die Kreuzung wird durch zwei sich überlappende Fahrspursegmente dargestellt ($\mathbf{U}_2, \mathbf{U}_3$). Der Pfad des Fahrzeugs besteht aus $\mathbf{p} = \{\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_4\}$. Durch die Relation *überlappt*($\mathbf{U}_2, \mathbf{U}_3$) kann beispielsweise auch *liegt auf*(\mathbf{U}_3, \mathbf{p}) abgeleitet werden.

4.1.3 Relationen zu Fahrzeugen

Neben der schon erwähnten Relation (*hat*) eines Fahrzeugs zu genau einem Pfad werden für Fahrzeuge weitere Relationen definiert.

fährt auf(\mathbf{v}, \mathbf{p})

Diese Relation gibt an, auf welchem statischen Szenenelement das Fahrzeug gerade fährt. Dabei kann das Fahrzeug als punktförmig angenommen werden. Eine Zuordnung zu mehreren Fahrspursegmenten kann jedoch sinnvoll sein, wenn die genaue Position des Fahrzeugs sensorisch nicht genau bestimmt werden kann (siehe Kapitel 4.5.2). Bei sich überlappenden

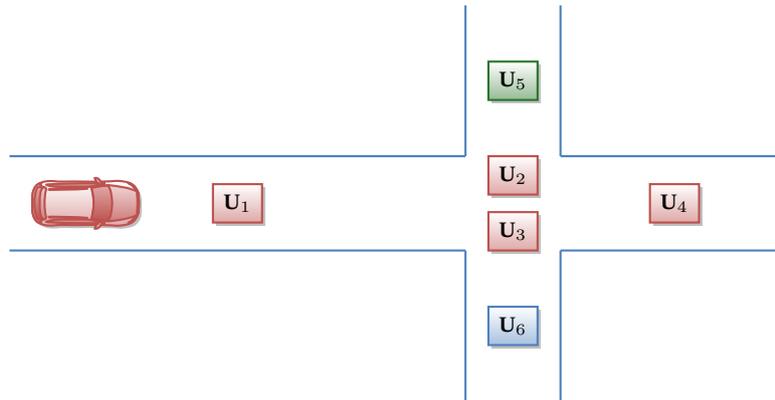


Abbildung 4.4: Kreuzungsszenario mit einem Fahrzeug (\mathbf{v}). Entsprechend der Relationen zu seinem Pfad ($\mathbf{p} = \{\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_4\}$) sind die statischen Szenenobjekte eingefärbt: rot - *liegt auf*(\mathbf{U}, \mathbf{p}), grün - *links von*(\mathbf{U}, \mathbf{p}), blau - *rechts von*(\mathbf{U}, \mathbf{p}).

<i>links von</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>hinten</i> ($\mathbf{v}_k, \mathbf{v}_l$)	<i>links von</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>gleiche Höhe</i> ($\mathbf{v}_k, \mathbf{v}_l$)	<i>links von</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>vor</i> ($\mathbf{v}_k, \mathbf{v}_l$)
<i>gleiche Spur</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>hinten</i> ($\mathbf{v}_k, \mathbf{v}_l$)	<i>gleiche Spur</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>gleiche Höhe</i> ($\mathbf{v}_k, \mathbf{v}_l$)	<i>gleiche Spur</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>vor</i> ($\mathbf{v}_k, \mathbf{v}_l$)
<i>rechts von</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>hinten</i> ($\mathbf{v}_k, \mathbf{v}_l$)	<i>rechts von</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>gleiche Höhe</i> ($\mathbf{v}_k, \mathbf{v}_l$)	<i>rechts von</i> ($\mathbf{v}_k, \mathbf{v}_l$) <i>vor</i> ($\mathbf{v}_k, \mathbf{v}_l$)

Abbildung 4.5: Mögliche Kombinationen von Relationen eines Fahrzeugs \mathbf{v}_k zu einem anderen Fahrzeug \mathbf{v}_l (rot)

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas et al.)

Fahrspursegmenten ist die Zuordnung nicht nur von der aktuellen Position, sondern auch von der gefahrenen Route abhängig.

laterale/longitudinale Relation($\mathbf{v}_k, \mathbf{v}_l$)

Longitudinale und laterale Relationen geben die relativen Positionen zweier Fahrzeuge ($\mathbf{v}_k, \mathbf{v}_l$) zueinander an. Es gibt drei laterale sowie drei longitudinale Relationen. Longitudinal werden *hinten*(\mathbf{v}), *gleiche Höhe*(\mathbf{v}) und *vor*(\mathbf{v}), lateral *links von*(\mathbf{v}), *gleiche Spur*(\mathbf{v}) sowie *rechts von*(\mathbf{v}) unterschieden. Für die Beschreibung der relativen Position wird im Regelfall genau eine longitudinale und eine laterale Relation verwendet. Eine Übersicht ist in Abbildung 4.5 schematisch dargestellt.

Die lateralen Relationen zweier Fahrzeuge wird über Inferenz auf der Ontologie

bestimmt.

$$\begin{aligned}
 \textit{links von}(\mathbf{v}_l, \mathbf{v}_k) &:= \exists \mathbf{U}, \mathbf{p} : \textit{fährt auf}(\mathbf{v}_l, \mathbf{U}) \wedge \textit{hat}(\mathbf{v}_k, \mathbf{p}) \wedge \textit{links von}(\mathbf{U}, \mathbf{p}) \\
 \textit{rechts von}(\mathbf{v}_l, \mathbf{v}_k) &:= \exists \mathbf{U}, \mathbf{p} : \textit{fährt auf}(\mathbf{v}_l, \mathbf{U}) \wedge \textit{hat}(\mathbf{v}_k, \mathbf{p}) \wedge \textit{rechts von}(\mathbf{U}, \mathbf{p}) \\
 \textit{gleiche Spur}(\mathbf{v}_l, \mathbf{v}_k) &:= \exists \mathbf{U}, \mathbf{p} : \textit{fährt auf}(\mathbf{v}_l, \mathbf{U}) \wedge \textit{hat}(\mathbf{v}_k, \mathbf{p}) \wedge \textit{liegt auf}(\mathbf{U}, \mathbf{p})
 \end{aligned}
 \tag{4.2}$$

Ein Fahrzeug \mathbf{v}_l ist z. B. links von einem anderen Fahrzeug \mathbf{v}_k wenn es auf einem Fahrspursegment \mathbf{U} fährt, das links vom Pfad \mathbf{p} des anderen Fahrzeugs \mathbf{v}_k liegt.

Die Relation ist sowohl für Fahrzeuge auf der gleichen Route wie auch für Fahrzeuge auf sich kreuzenden Routen definiert. Dabei spiegelt sie jeweils die intuitive relative Position zweier Fahrzeug wieder. So ist ein Fahrzeug, das bei einer Kreuzung von rechts kommt auch rechts von dem anderen Fahrzeug und umgekehrt. Dabei ist zu beachten, dass aus der Relation zwischen Fahrzeug \mathbf{v}_k und \mathbf{v}_l nicht direkt auf die Relation zwischen $\mathbf{v}_k, \mathbf{v}_l$ geschlossen werden kann. Je nach Streckentopologie und Position der Fahrzeuge sind alle Kombinationen für diese Relationen möglich. Kommen sich z. B. zwei Fahrzeuge im Rechtsverkehr entgegen sind sie jeweils links vom anderen Fahrzeug.

Die longitudinale Relation wird entsprechend des gewünschten Sicherheitsabstands zwischen Fahrzeugen definiert. Dabei wird die geometrische Grenze zwischen den Relationen so gewählt, dass *gleiche Höhe*($\mathbf{v}_k, \mathbf{v}_l$) gilt, wenn ein Fahrzeug \mathbf{v}_k den Sicherheitsabstand von \mathbf{v}_l verletzt und beide auf der gleichen Spur fahren. Longitudinale Relationen sind wie laterale Relationen auch sowohl für Fahrzeuge auf der gleichen Route wie auch für Fahrzeuge auf sich kreuzenden Routen einheitlich definiert. Je nach dem, ob sich ein Fahrzeug auf der gleichen, einer entgegenkommenden oder einer kreuzenden Spur befindet, kann der Sicherheitsabstand unterschiedlich gewählt werden.

Für die Bestimmung der jeweiligen longitudinalen Relation werden die in Kapitel 3.4.1 definierten Abbildungen verwendet. Im Folgenden werden zwei Fahrzeuge \mathbf{v}_1 und \mathbf{v}_2 betrachtet, die auf zwei sich kreuzenden oder teilweise überlappenden nicht notwendigerweise verschiedenen Routen R_1 und R_2 fahren. Die zwei Abbildungen $R'_1(x)$ und $R'_2(x)$ entsprechend Formel (3.3) seien ohne Beschränkung der Allgemeinheit so gewählt, dass:

$$\exists x \in D_{R'_1} \cap D_{R'_2} : R'_1(x) = R'_2(x)
 \tag{4.3}$$

Dabei ist $D_{R'}$ der Definitionsbereich der Funktion R' .

Für jede longitudinale Relation $\mathbf{r}(\mathbf{v})$ zu einem anderen Fahrzeug werden zwei Grenzfunktionen für die untere $f_{\mathbf{r}(\mathbf{v})}^{lb}(t)$ und obere Schranke $f_{\mathbf{r}(\mathbf{v})}^{ub}(t)$ der Form

$$f : t \mapsto x
 \tag{4.4}$$

definiert. Die Relation kann somit definiert werden als:

$$\mathbf{r}(\mathbf{v}) := (f_{\mathbf{r}(\mathbf{v})}^{lb}(t) \leq x_{\text{ego}}(t) < f_{\mathbf{r}(\mathbf{v})}^{ub}(t))
 \tag{4.5}$$

4 Semantische Szenenmodellierung

Um Wohldefiniertheit der longitudinalen Relationen zu gewährleisten, gilt für die longitudinalen Relationen der gewählten Verkehrsontologie:

$$\begin{aligned}
 f_{\text{hinterv}}^{\text{lb}}(t) &:= -\infty \\
 f_{\text{hinterv}}^{\text{ub}}(t) &:= f_{\text{gleiche Höhe(v)}}^{\text{lb}}(t) := x_{\mathbf{v}}(t) - A_1 \\
 f_{\text{gleiche Höhe(v)}}^{\text{ub}}(t) &:= f_{\text{vor(v)}}^{\text{lb}}(t) := x_{\mathbf{v}}(t) + A_2 \\
 f_{\text{vor(v)}}^{\text{ub}}(t) &:= \infty,
 \end{aligned} \tag{4.6}$$

wobei A_i mit $i \in 1,2$ jeweils den Sicherheitsabstand modelliert und abhängig von weiteren Parametern wie z. B. der Fahrzeuggeschwindigkeit und der gewählten Routen sein kann.

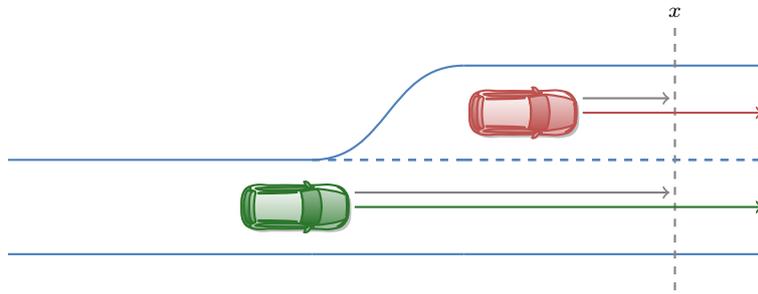
Allgemein kann $x_{\mathbf{v}}(t)$ eine beliebige stetige Funktion sein. Für das weitere Vorgehen beim Aufbau eines semantischen Zustandsraumes ist es jedoch erforderlich, an die Funktionen $f^{\text{lb}}(t)$ und $f^{\text{ub}}(t)$ und somit an $x_{\mathbf{v}}(t)$ weitere Anforderungen zu stellen. So müssen sich für jedes Paar von Fahrzeugen die Schnittpunkte der beiden Funktionen analytisch berechnen lassen.

Um das Konzept klarer darstellen und genauere Aussagen über Eigenschaften des semantischen Zustandsraums treffen zu können, wird im Folgenden von linearen Funktionen ausgegangen. Das heißt, dass alle Fahrzeuge mit konstanter Geschwindigkeit entlang ihrer Route prädiiziert werden. Viele in dieser Arbeit eingesetzte Perzeptions- und Prädiktionsverfahren legen diese Annahme ebenfalls zugrunde, so dass sich in diesen Fällen hieraus keine praktische Einschränkung ergibt. Lediglich das in Kapitel 2.6.1 vorgestellte Verfahren zur manöverbasierten Prädiktion liefert nichtlineare Prädiktionen. Wie diese auf Manöverebene verwendet werden können, wird exemplarisch in Kapitel 4.4.2 gezeigt.

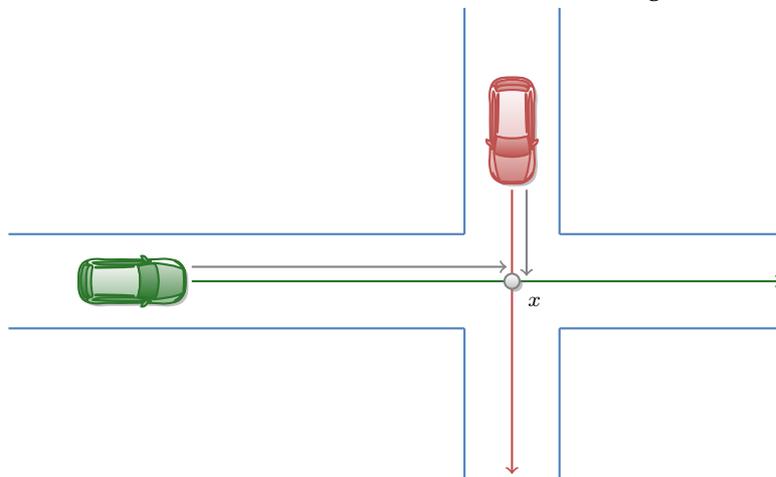
Die Bestimmung der longitudinalen Relationen soll an einem Beispiel verdeutlicht werden. In Abbildung 4.6 sind zwei Szenen dargestellt. Befinden sich zwei Fahrzeuge auf der gleichen Route (Landstraßenszenario, oben), so wird ihr Abstand zu einem beliebigen Punkt (x) auf der Route ermittelt. Sind die Fahrzeuge auf sich kreuzenden Routen (Kreuzungsszenario, Mitte), so wird der jeweilige Abstand zum Kreuzungspunkt gemessen. In beiden Fällen werden die Abstände verglichen und entsprechend des aktuellen Sicherheitsabstands die longitudinale Relation bestimmt (unten).

Für den Aufbau des semantischen Zustandsraums werden die Relationen des Ego-Fahrzeugs zu allen relevanten Szenenobjekten verwendet. Diese sind in Abbildung 4.1 in rot dargestellt. Diese Relationen haben die Form $r(\mathbf{ego}, \mathbf{Szenenobjekt})$. Im Sinne einer übersichtlicheren Darstellung wird im Folgenden eine vereinfachte Notation verwendet:

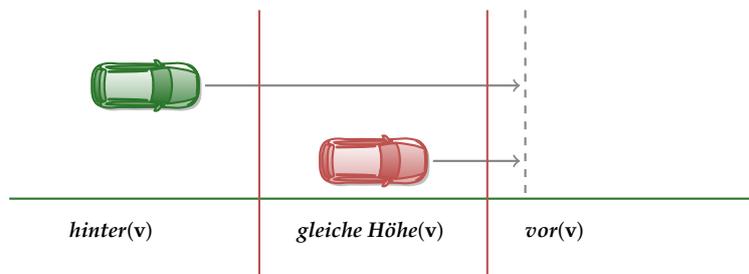
$$r(\mathbf{Szenenobjekt}) := r(\mathbf{ego}, \mathbf{Szenenobjekt}). \tag{4.7}$$



(a) Landstraßenszenario mit zwei Fahrzeugen



(b) Kreuzungsszenario mit zwei Fahrzeugen



(c) Resultierende Ableitung der longitudinalen Relation zweier Fahrzeuge

Abbildung 4.6: Konsistente Bestimmung der longitudinalen Relation in zwei exemplarischen Szenarien

4.2 Semantischer Planungsraum

Auf Basis der semantischen Szenenbeschreibung wird ein Zustandsraum zur Planung von Manövern für das Ego-Fahrzeug abgeleitet. Ein semantischer Zustand in diesem Zustandsraum beschreibt eine Menge gültiger Relationen. Eine Aktion des Ego-Fahrzeugs kann dabei nur eine direkte Auswirkung auf seine Relationen zu anderen Szenenobjekten haben. Daher werden zum Aufbau des Zustandsraums diese Relationen verwendet.

Definition 4.2.1: Semantischer Zustand

Ein semantischer Zustand s des Ego-Fahrzeugs **ego** ist definiert als die Menge aller gültiger Relationen zu allen relevanten Szenenobjekten O :

$$s = \{r(\mathbf{ego}, o) \mid o \in O \wedge r(\mathbf{ego}, o) = \text{wahr}\}$$

Der Begriff der Relevanz kann und muss an dieser Stelle nicht abschließend definiert werden. Die Beschränkung auf relevante Objekte führt zu einer Reduktion der Komplexität des Zustandsraums und ist daher im Hinblick auf Ausführungszeit und Ressourcenverbrauch zu befürworten. Die Miteinbeziehung nicht relevanter Objekte hat auf das konzeptionelle Vorgehen keine Auswirkungen. Später werden Verfahren vorgestellt, die auch die resultierenden praktischen Effekte minimieren (siehe Kapitel 5.2).

Auf Basis der Definition eines semantischen Zustands kann der semantischer Zustandsraum definiert werden.

Definition 4.2.2: Semantischer Zustandsraum

Der Zustandsraum S mit $s \in S$ ist die Menge von möglichen semantischen Zuständen s die **ego** theoretisch annehmen kann.

Um eine Bewegung in dem semantischen Zustandsraum darzustellen wird die Definition einer Nachbarschaftsbeziehung zwischen Zuständen benötigt.

Definition 4.2.3: Transition

Transitionen ($start \rightarrow ziel$) zwischen zwei semantischen Zuständen $start$ und $ziel$ zeigen an, dass es für **ego** prinzipiell möglich ist direkt von $start$ in den Zustand $ziel$ zu wechseln. An Transitionen können Bedingungen geknüpft sein (siehe Definition 4.3.1), die definieren, ob und wann eine Transition gültig ist, also von **ego** als semantische Aktion ausgeführt werden kann. Die Menge aller Transitionen wird mit T_S bezeichnet.

Das Ego-Fahrzeug kann sich entlang der Transitionen semantische Aktionen ausführen.

Definition 4.2.4: Semantische Aktion

Eine semantische Aktion ist der Übergang zwischen zwei semantischen Zu-

ständen. Sie erfolgt immer entlang einer Transition. Eine semantische Aktion führt zur Veränderung mindestens einer Relation zwischen Ego-Fahrzeug und anderen Szenenobjekten.

Um semantische Aktionen klassifizieren zu können werden den zugrundeliegenden Transitionen verschiedene Typen zugeordnet.

Definition 4.2.5: Transitionstypen

Basierend auf der gewählten Ontologie gibt es drei verschiedene Typen von Transitionen:

$$\text{Type}(s_a \rightarrow s_b) \in \{\textit{longitudinal}, \textit{lateral}, \textit{temporal}\} \quad (4.8)$$

Ändert sich bei einer Transition eine longitudinale Relation zu einem anderen Fahrzeug, so hat diese den Typ longitudinal, ändert sich hingegen eine laterale Relation, so handelt es sich um eine laterale Transition. Bei Änderungen der Relation *fährt auf*(U_a) zu *fährt auf*(U_b) hängt der Typ der Transition von der Relation zwischen U_a und U_b ab. Gilt *hat links*(U_a, U_b) oder *hat rechts*(U_a, U_b) so handelt es sich um eine laterale Transition. Bei *beginnt mit*(U_a, U_b) oder *endet in*(U_a, U_b) liegt eine longitudinale Transition vor. Temporale Transitionen werden im folgenden definiert (siehe Definition 4.2.8).

Änderungen von Relationen, an denen das Ego-Fahrzeug nicht beteiligt ist, können nicht direkt von diesem beeinflusst werden und stellen somit keine Aktionen im Zustandsraum dar. Änderungen der Szene wirken sich allerdings indirekt auf die Zustände im Zustandsraum aus. So kann sich z. B. die Menge der möglichen Zustände des Ego-Fahrzeugs durch die temporale Entwicklung der Szene verändern. Zustände fallen weg oder neue Zustände entstehen.

Dies geschieht zu bestimmten Ereignissen, also an bestimmten Zeitpunkten. Für das Zeitintervall zwischen zwei aufeinanderfolgenden Ereignissen ist der Zustandsraum statisch. Er basiert auf einer bestimmten semantischen Anordnung der Umfeldobjekte, einer Konstellation.

Definition 4.2.6: Konstellation

Der partielle semantische Zustandsraum zwischen zwei zeitlich aufeinanderfolgenden Ereignissen wird als Konstellation bezeichnet. Eine Konstellation ist somit für ein gewisses Zeitintervall gültig. Die enthaltenen Zustände basieren auf den in diesem Zeitintervall gültigen und aufgrund der Abwesenheit von Ereignissen sich zeitlich nicht verändernden Relationen.

Der semantische Zustandsraum besteht somit aus mehreren Konstellationen. Mit Hilfe des Begriffs der Konstellation, kann die Definition des semantischen Zustands konkretisiert werden.

Definition 4.2.7: Semantischer Zustand II

Ein semantischer Zustand s des Ego-Fahrzeugs *ego* ist definiert als die Menge

aller gültiger Relationen zu allen relevanten Szenenobjekten O in einer Konstellation k .

$$s_k = \{r(\mathbf{ego}, \mathbf{o}) \mid \mathbf{o} \in O \wedge r(\mathbf{ego}, \mathbf{o}) = \text{wahr}\}$$

Zur klareren Darstellung wird im Folgenden auf den Index zur Beschreibung der Konstellation, wenn nicht benötigt, verzichtet.

Zwischen Zuständen benachbarter Konstellationen werden zeitliche Transitionen eingeführt. Diese erlauben es dem Ego-Fahrzeug bei der Planung in der Zeit voranzuschreiten.

Definition 4.2.8: Temporale Transition

Zeitliche Transitionen verbinden Zustände, die durch genau die selben semantischen Relationen beschreiben werden. Da Bewegungen entgegen der Zeit als nicht möglich angenommen werden, sind die Transitionen nur in Richtung der Konstellation mit späterem Zeitintervall möglich.

Prinzipiell kann jede Kombination aus Relationen zu Szenenobjekten einen semantischen Zustand repräsentieren. Jedoch führen nicht alle Kombinationen aus Relationen zu sinnvollen Zuständen. Viele Kombinationen können durch logische Betrachtungen ausgeschlossen werden. So können die Relationen *vor(v)* und *hinter(v)* nicht zusammen in einem Zustand auftreten. Andere Relationskombinationen widersprechen der Szenengeometrie. So ist es für das Ego-Fahrzeug z. B. nicht möglich, in einer Szene mit zwei weiteren hintereinander fahrenden Fahrzeugen vor dem vorderen und hinter dem hinteren zu sein.

Um die Menge von Zuständen im Zustandsraum auf ein Minimum zu beschränken, wird im Folgenden ein systematisches Vorgehen beschrieben, alle gültigen Zustände des Zustandsraums aus der semantischen Szenenbeschreibung abzuleiten. Dabei werden alle für das Ego-Fahrzeug theoretisch erreichbaren Zustände erzeugt. Es werden auch solche Zustände erzeugt, die aufgrund von physikalischen Limitierungen nicht erreichbar sind. So werden z. B. auch Zustände generiert, die zu einer Kollision mit Verkehrsteilnehmern führen würden. Eine Verwendung dieser Zustände wird durch Bedingungen auf den Transitionen, die zu diesen Zuständen hinführen, für die Planung einer Manöverkette ausgeschlossen. Eine Modellierung dieser Zustände kann jedoch unter anderem für die Darstellung von Unsicherheiten eingesetzt werden (vergleiche Kapitel 4.5.2).

4.3 Aufbau des Zustandsraums

Der Aufbau des Zustandsraums aus der semantischen Szenenbeschreibung lässt sich in vier Phasen gliedern, die im Folgenden beschrieben werden. Zum besseren Verständnis wird der gesamte Prozess anschließend als Pseudocode visualisiert.

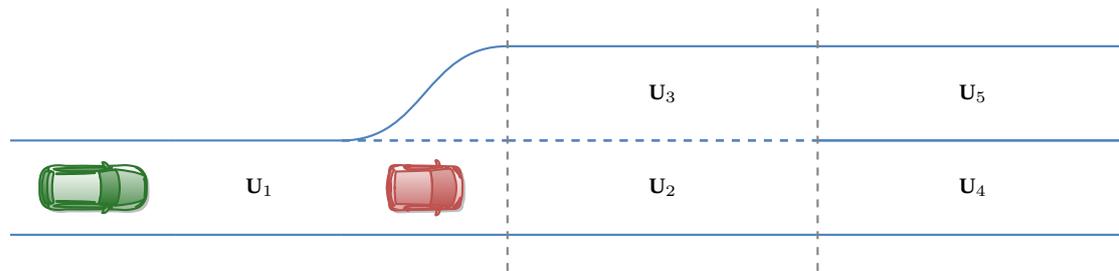


Abbildung 4.7: Schematische Darstellung des Bundesstraßenszenarios. Die Straße wird zweispurig. Ein anderes Fahrzeug v (rot) fährt vor ego (grün) und kann ggf. überholt werden.

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas et al.)

Zur Veranschaulichung des Prozesses wird eine Beispielszene verwendet, die in Abbildung 4.7 dargestellt ist. Hier ist ein Bundesstraßenszenario abgebildet. Das Ego-Fahrzeug (grün) fährt hinter einem anderen Fahrzeug (rot) her. Kurz vor dem Ego-Fahrzeug wird die Straße zweispurig. Weiter vorne teilt sich die Straße in zwei separate Fahrspuren auf. Aufgrund der geplanten Route des Ego-Fahrzeugs möchte dieses am Ende des dargestellten Straßenabschnitts rechts abbiegen. Somit möchte es auf der rechten der beiden Fahrspuren (U_4) weiterfahren, hat jedoch die Möglichkeit das andere Fahrzeug im zweispurigen Bereich zu überholen.

4.3.1 Betrachtung der statischen Szenenobjekte

Für eine gegebene Konstellation, z. B. die aktuelle Konstellation, werden zunächst semantische Zustände auf Basis von Relationen zu statischen Szenenobjekten gebildet. Hierzu werden alle möglichen Relationen *fährt auf* für alle statischen Szenenobjekte des Pfads von ego gebildet. Somit ergibt sich für den Zustandsraum die Menge von Zuständen als:

$$S = \{\{f\ddot{a}hrt\ auf(U)\} \mid U \text{ ist vom Typ } \textit{statisches Szenenobjekt}\}$$

Die Transitionen zwischen den Zuständen werden aus den Relationen *hat links/rechts*, *beginnt/endet in* und *verbunden mit* inferiert. Treten für zwei statische Szenenobjekte die Relation *hat links/rechts* zusammen mit der Relation *verbunden mit* auf, so entstehen laterale Transitionen zwischen den entsprechenden Zuständen. Diese werden im Folgenden grün dargestellt. Die Relation *beginnt/endet in* impliziert bereits eine befahrbare Verbindung und führt daher zur einer longitudinalen Transition. Diese wird im Folgenden blau dargestellt. Für die Beispielszene ist dies in Abbildung 4.8 dargestellt.

4 Semantische Szenenmodellierung

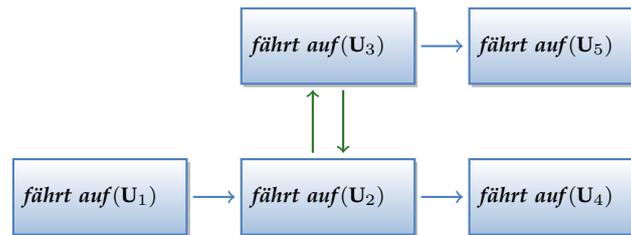


Abbildung 4.8: Zustandsraum unter Berücksichtigung der Relationen zu statischen Szenenobjekten. Jedes statische Szenenobjekt wird durch einen Zustand repräsentiert. Laterale (grün) und longitudinale (blau) Transitionen zwischen den Zuständen leiten sich aus den Relationen der zugehörigen Szenenobjekte ab.

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas et al.)

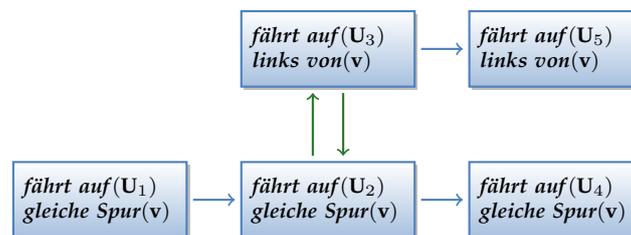


Abbildung 4.9: Zustandsraum unter Berücksichtigung der lateralen Relationen zu dynamischen Szenenobjekten. Die Zustände werden mit lateralen Relationen angereichert. Die Größe und Topologie des Zustandsraums bleibt erhalten.

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas et al.)

4.3.2 Betrachtung der dynamischen Szenenobjekte

In Phase zwei und drei werden die dynamischen Szenenobjekte in den Aufbau des Zustandsraums miteinbezogen. Dabei werden zunächst die lateralen Relationen zu Szenenobjekten berücksichtigt. Wie in (4.2) dargestellt ist die laterale Relation zu einem dynamischen Szenenobjekt stark an die Relation zu einem statischen Szenenobjekt geknüpft. Daher können laterale Relationen durch Anreicherung der Zustände mit weiteren Relationen ergänzt werden. Für jeden Zustand und jedes dynamische Szenenobjekt können die Gleichungen (4.2) ausgewertet werden und die entsprechende Relation im Zustand ergänzt werden.

Abbildung 4.9 zeigt den für die Beispielszene resultierenden Zustandsraum. In diesem Beispiel wird davon ausgegangen, dass das rote Fahrzeug der Fahrspur weiter folgt. Sein Pfad p_v besteht also aus drei Fahrspursegmenten:

$$p_v = \{U_1, U_2, U_4\}$$

In Phase drei werden die longitudinalen Relationen in den Zustandsraum integriert. Den vorhandenen Zuständen wird pro Objekt eine weitere Relation (*vor*, *gleiche Höhe*, *hinter*) zugewiesen. Dabei ist es im Allgemeinen notwendig,

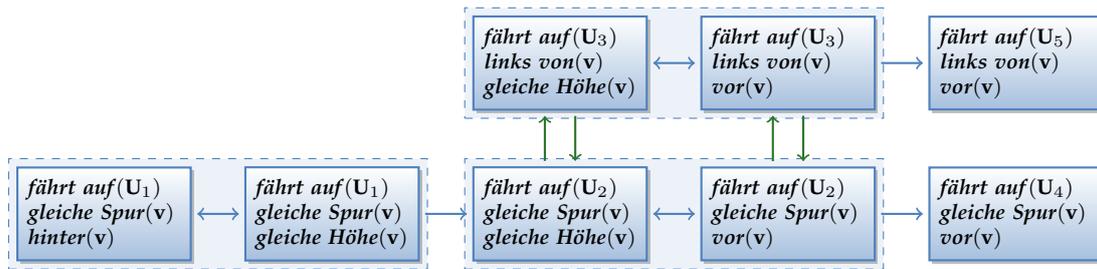


Abbildung 4.10: Zustandsraum unter Berücksichtigung der longitudinalen Relationen zu dynamischen Szenenobjekten. Entsprechend der longitudinalen Relationen zu anderen Verkehrsteilnehmern werden Zustände aufgeteilt. Geteilte Zustände sind durch gestrichelte Kästen hervorgehoben und jeweils mit longitudinalen Transitionen verbunden.

mehrere Zustände zu neuen Zuständen aufzuteilen, um verschiedene mögliche Kombinationen von Relationen abzubilden. Die longitudinalen Relationen haben eine geometrische Interpretation (vergleiche Abbildung 4.6 und Kapitel 4.3.3), die es erlaubt, die Zustände zu identifizieren. Zwischen duplizierten Zuständen werden longitudinale Transitionen in beiden Richtungen eingefügt. Dies erlaubt eine freie Relativbewegung zwischen Fahrzeugen. Eine Einschränkung auf bestimmte Richtungen, um z. B. Überholen auf der rechten Seite zu verbieten, kann durch Bedingungen (siehe Kapitel 4.3.4) modelliert werden. In Abbildung 4.10 ist der resultierende Zustandsraum dargestellt. Aus einem Zustand entstandene Zustände sind jeweils gekennzeichnet.

4.3.3 Temporale Fortsetzung

Der bisher generierte Zustandsraum bildet lediglich eine Konstellation ab, bzw. eine Szene ohne topologische Veränderungen der Konstellation um das Ego-Fahrzeug. Um auch die dynamischen Entwicklungen der Szene zu berücksichtigen, muss der Zustandsraum mehrere Konstellationen abdecken. In Phase 4 werden zunächst die Ereignisse, die die Konstellationen begrenzen, ermittelt sowie die Konstellationen erstellt und verbunden.

Konstellationen sind für den Zeitraum zwischen zwei Ereignissen gültig. Diese Ereignisse können anhand der durch die Relationen induzierten Grenzfunktionen bestimmt werden. Dabei werden die Grenzfunktionen der longitudinalen Relationen sowie Grenzfunktionen der Relationen zu statischen Szenenobjekten verwendet. Letztere sind analog zu ersteren definiert, sind jedoch statisch in der Zeit. Grenzfunktionen sind entsprechend Gleichung (4.4) in der Ort-Zeit-Ebene (x, t) definiert.

Für die Menge aller Grenzfunktionen $f \in F$ werden nun alle möglichen Schnittpunkte t_i ermittelt. Zwei in einer zeitlichen Ordnung benachbarte Punkte t_i, t_{i+1}

4 Semantische Szenenmodellierung

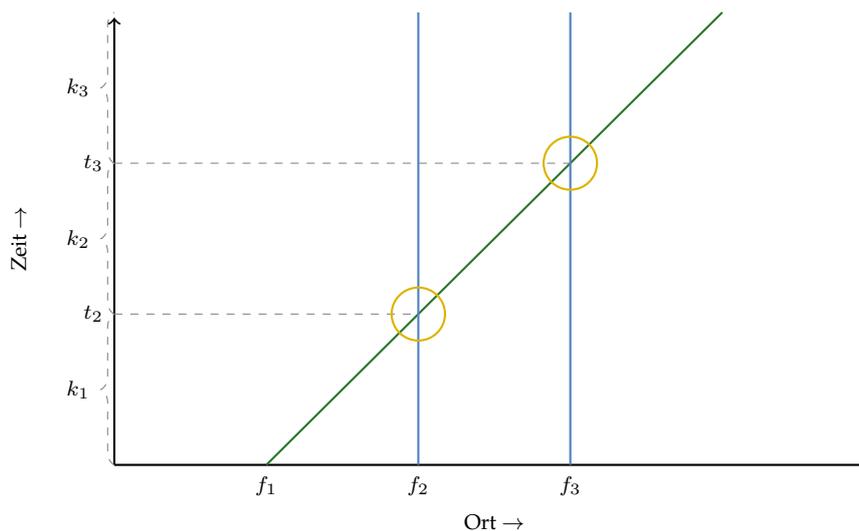


Abbildung 4.11: Beispielhafte Berechnung von Konstellationsintervallen: Die drei Grenzfunktionen weisen zwei Schnittpunkte auf. Diese definieren zwei Ereignisse bei t_2 und t_3 . Die Zeit wird in an diesen Ereignissen in insgesamt drei Intervalle geteilt, die jeweils eine Konstellation induzieren. In jedem Intervall sind die Grenzfunktionen in Richtung der Ortsachse eindeutig geordnet.

In Anlehnung an [Bit14]

begrenzen ein Intervall, das eine Konstellation k_i definiert. Auf dem offenen Intervall $(t_i : t_{i+1})$ bilden die Grenzfunktionen in Richtung der Ortsachse eine absolute Ordnung. Für einen Zeitpunkt innerhalb diesen Intervalls kann also eine Instanz der Verkehrsszenenontologie gebildet werden, die auf dem gesamten Intervall gültig ist, und auf Basis dieser ein Zustandsraum abgeleitet werden. Dieser Zustandsraum ist eine Konstellation. Abbildung 4.11 zeigt, wie die Schnittpunkte dreier Grenzfunktionen f_1 bis f_3 die Zeitachse in insgesamt drei Intervalle aufteilt. Jedes Intervall induziert eine Konstellation k_1 bis k_3 .

Je nach Anwendung und nach Art der verwendeten Grenzfunktionen können alle oder auch nur eine beschränkte Anzahl von Konstellationen gebildet werden. Für lineare Grenzfunktionen ist die Anzahl der Konstellationen beschränkt. Werden an den Aufbau des Zustandsraums und die anschließende Planung Anforderungen bezüglich der Laufzeit gestellt, kann es sinnvoll sein, die Anzahl der berechneten Konstellationen oder das maximale Zeitintervall, in dem Konstellationen erstellt werden, zu beschränken. Hierdurch kann eine sehr gute Skalierbarkeit und Anpassbarkeit des Ansatzes an entsprechende Anforderungen erreicht werden.

Zustände mit der gleichen Relationsmenge können in verschiedenen Konstellationen vorkommen. Zwischen zwei Zuständen mit der gleichen Relationsmenge in benachbarten Konstellationen wird jeweils eine temporale Transition eingefügt. Diese Transitionen verbinden die Konstellationen zu einem Zustandsraum. Sie ermöglichen es, bei der Planung über ein oder mehrere Ereignisse hinweg die Relationen zu anderen Szenenobjekten nicht zu verändern. Eine Bewegung in der

Zeit ist nur in positiver Richtung sinnvoll. Daher verlaufen temporale Transitionen nur von einer früheren zu einer späteren Konstellation.

Abbildung 4.12 zeigt die temporale Erweiterung des Zustandsraums über drei Konstellationen hinweg. Temporale Transitionen sind in lila dargestellt. Es ist zu erkennen, dass viele Zustände jeweils einen Folgezustand in der nächsten Konstellation haben. Das Ego-Fahrzeug kann also die Relationen zu anderen Szenenobjekten beibehalten. Für einige Zustände existieren jedoch keine Folgezustände in der jeweils nächsten Konstellation. Hier muss in der Planung eine Aktion vor Ende des Zeitintervalls der Konstellation erfolgen.

4.3.4 Anwendung von Regeln – Modellierung von Verkehrsregeln

Nicht alle Transitionen zwischen Zuständen bilden eine gewünschte Aktion ab. Einige Transitionen führen in unsichere Zustände, z. B. solche, in denen ein Sicherheitsabstand verletzt oder sogar eine Kollision möglich wird. Diese Transitionen werden mittels Bedingungen als ungültig markiert.

Definition 4.3.1: Transitionsbedingung

Transitionsbedingungen definieren ob eine Transition als semantische Aktion in der Planung verwendet werden kann.

$$C_{(start \rightarrow ziel)} = \{wahr, falsch\} \quad (4.9)$$

Sie werden mittels prädikatenlogischen Formeln z. B. auf Basis der Relationen der an der Transition beteiligten Zustände definiert. Nur wenn die Bedingung erfüllt ist, darf eine Transition passiert werden.

Mit diesem Mechanismus können auch Verkehrsregeln wie das Verbot rechts zu überholen modelliert werden. Im Folgenden wird die Verwendung von Bedingungen an zwei Beispielen gezeigt.

Zunächst sollen Übergänge in Zustände ausgeschlossen werden, in denen eine Kollision mit einem anderen Fahrzeug **vehicle** auftreten können: Die Menge Q_1 beschreibt den Sachverhalt, dass sich **ego** an der gleichen Stelle wie **vehicle** befindet.

$$Q_1 = \{sameLevel(\mathbf{vehicle}), sameLane(\mathbf{vehicle})\} \quad (4.10)$$

Die Bedingungen zur Vermeidung dieses Zustands können dann über folgende Formel abgeleitet werden:

$$\forall s_s, s_t : (s_t \supseteq Q_1) \Rightarrow C_{(s_s \rightarrow s_t)} = falsch \quad (4.11)$$

Es werden also alle Transitionen invalidiert, deren Endzustand die beiden Relationen aus Q_1 beinhaltet. Gleichmaßen können auch Transitionen aus diesen Zuständen invalidiert werden:

$$\forall s_s, s_t : (s_s \supseteq Q_1) \Rightarrow C_{(s_s \rightarrow s_t)} = falsch \quad (4.12)$$

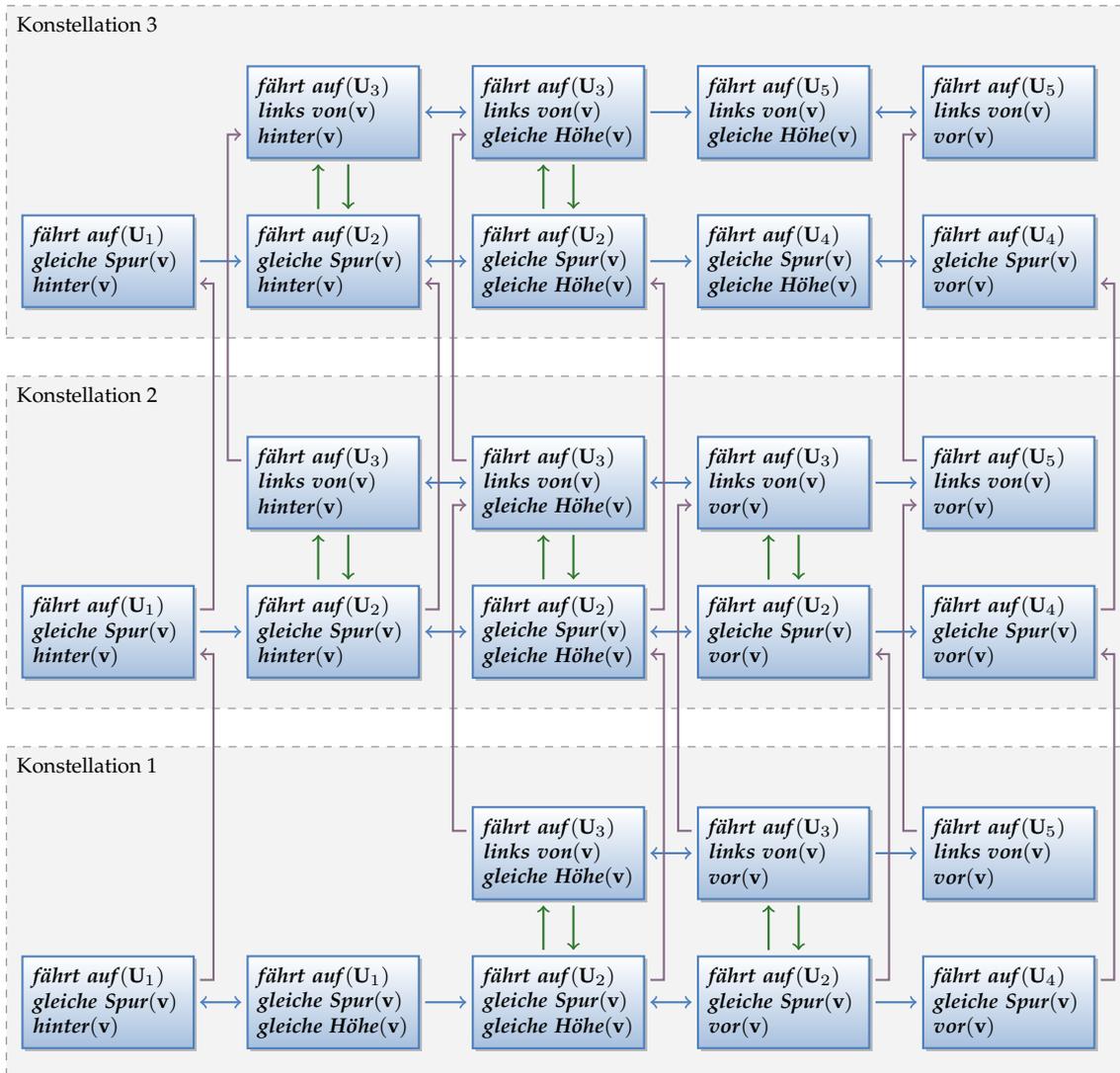


Abbildung 4.12: Temporale Fortsetzung des Zustandsraums über drei Konstellationen. Basierend auf der Szenenprädiktion ergeben sich mehrere Konstellationen. Korrespondierende Zustände sind durch zeitliche Transitionen (lila) verbunden.

Dies ist nicht zwingend notwendig, da der Ausgangszustand sowieso niemals erreicht werden sollte.

Ähnlich lässt sich auch ein Überholverbot auf der rechten Seite modellieren:

$$\begin{aligned} Q_2 &= \{\mathit{sameLevel}(\mathbf{vehicle}), \mathit{rightOf}(\mathbf{vehicle})\} \\ Q_3 &= \{\mathit{infrontOf}(\mathbf{vehicle}), \mathit{rightOf}(\mathbf{vehicle})\} \end{aligned} \quad (4.13)$$

$$\forall s_s, s_t : (s_s \supseteq Q_2) \wedge (s_t \supseteq Q_3) \Rightarrow C_{(s_s \rightarrow s_t)} = \mathit{falsch} \quad (4.14)$$

In Abbildung 4.13 ist das Ergebnis der Anwendung dieser Regeln auf den Zustandsraum dargestellt. Invalidierte Transitionen sind rot dargestellt. Zusätzlich sind zur besseren Verständlichkeit Zustände, in denen Kollisionen mit einem anderen Fahrzeug auftreten, rot eingefärbt.

4.3.5 Zusammenfassung des Ablaufs

Zur besseren Verständlichkeit der Abläufe werden diese als Pseudocode (Algorithmus 1) zusammengefasst. Als Eingabe werden der zu fahrende Pfad \mathbf{p} , eine Liste mit dynamischen Szenenobjekten (Fahrzeugen) \mathbb{V} , die Menge an semantischen Relationen \mathbb{R} , sowie die Verkehrsregeln in Form von Bedingungen \mathbb{C} erwartet. Zunächst wird der statische Zustandsraum aufgebaut (Zeile 2) und für jedes dynamische Szenenobjekt mit lateralen Relationen angereichert (Zeile 5). Anschließend wird für jedes dynamische Szenenobjekt (Zeile 6) ermittelt, zu welchen Zeitpunkten es neue Konstellationen induziert (Zeile 9) und welche Konstellation dazu dupliziert werden muss (Zeile 10). Abschließend werden für jedes dynamische Szenenobjekt die betroffenen semantischen Zustände geteilt (Zeile 13) und alle Zustände mit den dann eindeutigen longitudinalen Relationen angereichert (Zeile 14).

4.4 Analyse

Zur Analyse der semantischen Szenenmodellierung wird diese an verschiedenen Beispielszenen dargestellt. Für diese werden auch die entstehenden Zustandsräume gezeigt. Die Szenarien sind in der Komplexität so gewählt, dass die resultierenden Zustandsräume visuell darstellbar und qualitativ nachvollziehbar bleiben. Komplexere Szenen und größere Zustandsräume werden quantitativ betrachtet, um die Skalierung der Zustandsraumgröße mit der Komplexität der Szene darstellen zu können. Die Komplexitätsklasse der Zustandsraumgröße wird ebenfalls theoretisch hergeleitet.

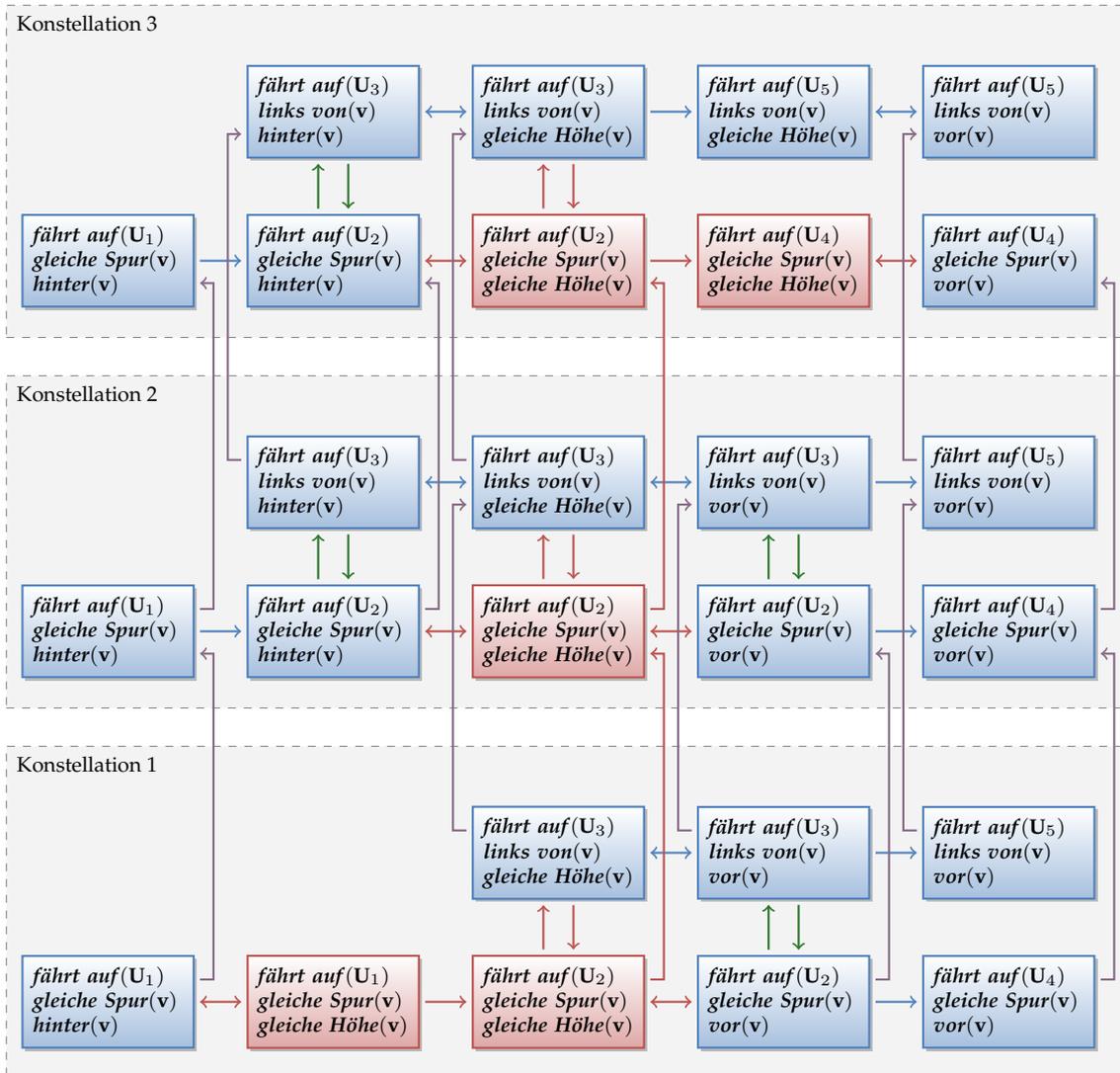


Abbildung 4.13: Anwendung von Regeln zur Einhaltung des Sicherheitsabstands zu anderen Fahrzeugen. Transitionen mit nicht erfüllter Bedingung werden invalidiert (rot). Zustände, die zu einer Kollision mit einem anderen Verkehrsteilnehmer führen würden, sind zur besseren Lesbarkeit ebenfalls rot dargestellt.

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas et al.)

Algorithmus 1 Aufbau Zustandsraum

Input: path: \mathbf{p} , vehicles: \mathbb{V} , relations: \mathbb{R} , conditions: \mathbb{C}

- 1: **procedure** BUILDSTATESPACE(\mathbf{p} , \mathbb{V} , \mathbb{R} , \mathbb{C})
- 2: $S[0] \leftarrow \text{BUILDSTATICSTATESPACE}(\mathbf{p}, \mathbb{R})$ ▷ see: 4.3.1
- 3: **for each** \mathbf{v} **in** \mathbb{V} **do**
- 4: $\mathbb{R}_{\mathbf{v}}^{lat} \leftarrow \text{FINDLATERALRELATIONS}(\mathbb{R}, \mathbf{v})$
- 5: $S[0] \leftarrow \text{ENRICHLATERALRELATIONS}(S[0], \mathbb{R}_{\mathbf{v}}^{lat})$ ▷ see: 4.3.2
- 6: **for each** \mathbf{v} **in** \mathbb{V} **do** ▷ see: 4.3.3
- 7: $\mathbb{R}_{\mathbf{v}}^{lon} \leftarrow \text{FINDLONGITUDINALRELATIONS}(\mathbb{R}, \mathbf{v})$
- 8: **for each** \mathbf{r} **in** $\mathbb{R}_{\mathbf{v}}^{lon}$ **do**
- 9: $t \leftarrow \text{CALCEVENTTIME}(S, \mathbf{r})$
- 10: $k \leftarrow \text{FINDCONTAININGCONFIGURATION}(t)$
- 11: $S[t] \leftarrow S[k]$
- 12: **for each** $S[i]$ **in** S **do**
- 13: $S[i] \leftarrow \text{SPLITNODES}(S[i], \mathbf{r})$
- 14: $S[i] \leftarrow \text{ENRICHLONGITUDINALRELATIONS}(S[i], \mathbf{r})$
- 15: $S \leftarrow \text{ADDTEMPORALTRANSITIONS}(S)$
- 16: **for each** c **in** \mathbb{C} **do**
- 17: $S \leftarrow \text{APPLYRULE}(S, c)$ ▷ see: 4.3.4
- 18: **return** S

4.4.1 Beispielszenarien

Die Anwendung der dargestellten Ontologie wird anhand von verschiedenen Straßenabschnitten exemplarisch gezeigt. Die Szenarien sind so gewählt, dass sie häufig auftretende Straßentopologien wie Kreuzungen und mehrspurige Abschnitte repräsentieren und interessante Aspekte der Szenenmodellierung wie z. B. Skalierbarkeit sowie Einsatz und Abdeckung der verschiedenen Relationen berücksichtigen. Des Weiteren sind sie bezüglich ihrer Komplexität und Größe so überschaubar gestaltet, dass sich die resultierenden semantischen Zustandsräume nicht nur quantitativ sondern auch qualitativ bewerten lassen.

Das erste Szenario ist eine Variation des in Bundesstraßenszenarios (siehe Abbildung 4.7) und ist in Abbildung 4.14 dargestellt. Um unterschiedlich große Szenarien betrachten zu können, kann dieses Szenario an der rechten Seite um weitere Fahrspursegmente ergänzt werden. Dieses Szenario wurde als Kartenausschnitt mittels der Lanelet-Repräsentation als OSM-Datei modelliert. Eine Darstellung der Karte ist in Abbildung 4.15 gezeigt. Als zweites Beispiel dient eine doppelte T-Kreuzung (Abbildung 4.16). Beispielhaft sind hier die Fahrwege von zwei Fahrzeugen dargestellt. Dieses Szenario wurde ebenfalls als Lanelet-Karte umgesetzt (Abbildung 4.17).

4 Semantische Szenenmodellierung

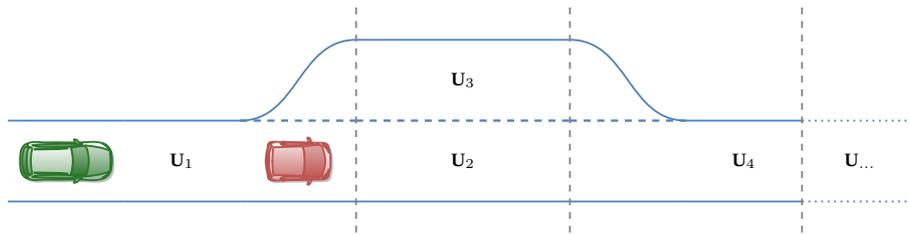


Abbildung 4.14: Schematische Darstellung eines skalierbaren Bundesstraßenszenarios. Die Szene kann bei Bedarf rechts um weitere Fahrspursegmente ergänzt werden.

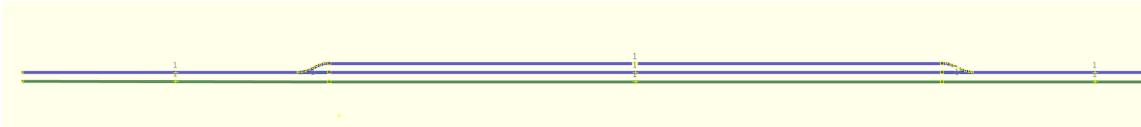


Abbildung 4.15: Kartenausschnitt des Bundesstraßenszenarios als Lanelet-Karte

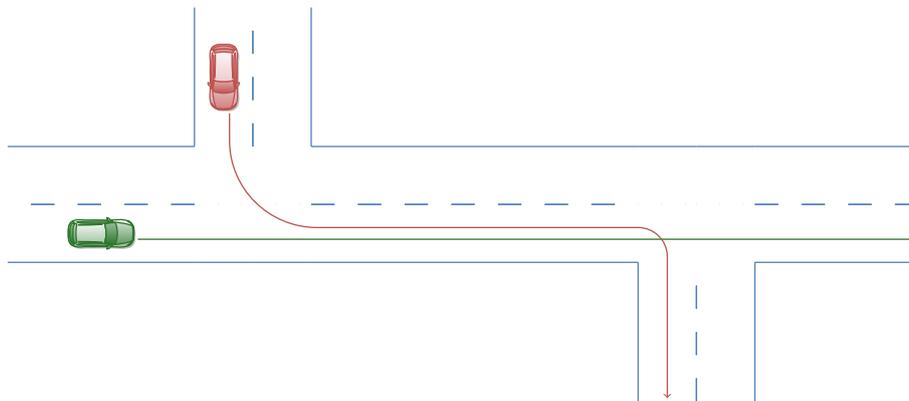


Abbildung 4.16: Schematische Darstellung eines Kreuzungsszenarios mit zwei T-Kreuzungen und zwei exemplarischen Fahrwegen

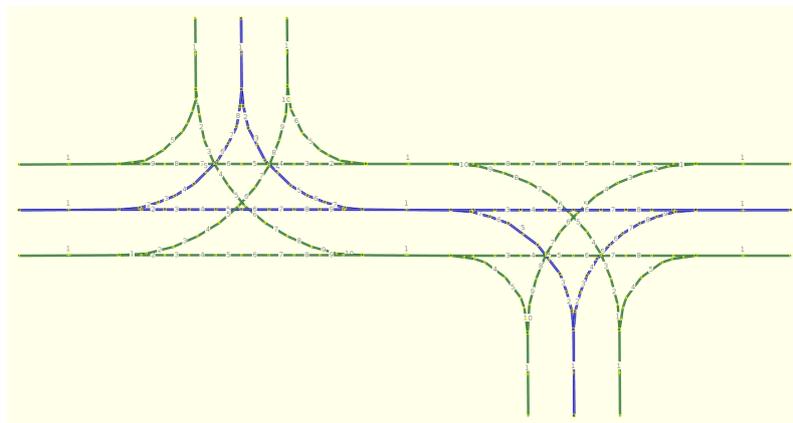


Abbildung 4.17: Kartenausschnitt des Kreuzungsszenarios als Lanelet-Karte

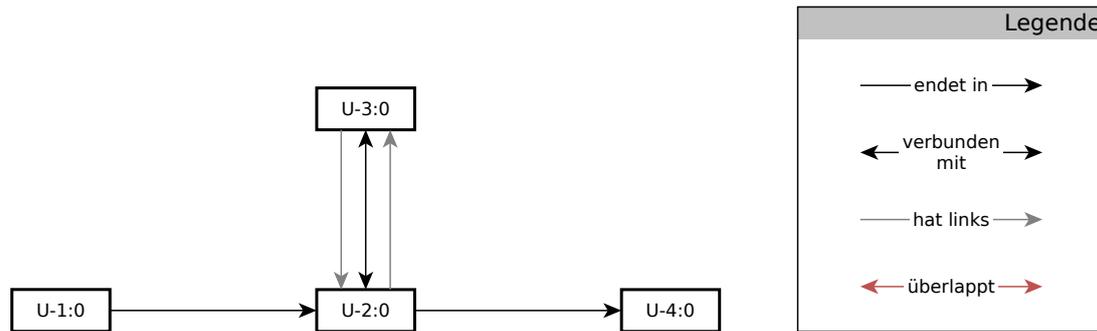


Abbildung 4.18: Ausgewählte Entitäten und Relationen aus der ABox des Bundesstraßenszenarios

4.4.2 Relationale Szenenmodellierung

Basierend auf den Lanelet-Karten und der Ontologie wird eine relationale Szenenmodellierung erstellt. Hierzu wird zunächst für das erste Szenario die ABox erstellt. Die Szenenbeschreibung des ersten Beispielszenarios besteht aus vier statischen Szenenobjekten. Eine graphische Darstellung der resultierenden ABox ist in Abbildung 4.18 dargestellt.

Die hier gezeigten Darstellungen wurden mittels der im Rahmen dieser Arbeit entwickelte Softwareimplementierung erstellt und weichen daher visuell von den konzeptionellen Visualisierungen ab. Aufgrund von Randbedingungen bei der Implementierung ergeben sich auch leichte Unterschiede in der Notation für statische Szenenobjekte. Sie sind entsprechend der ihnen zugrunde liegenden Straßensegmenten nummeriert. Bei der Verwendung von Kartenausschnitten ergibt sich somit ggf. eine nicht fortlaufende Nummerierung. Des Weiteren erlaubt die Implementierung sowohl die Verwendung von Lanelet-Karten (vergleiche Kapitel 3.4.1) als auch OSM-Karten. Um in beiden Fällen eine einheitliche Nomenklatur zu erreichen, werden statische Szenenobjekte wie folgt benannt:

U-<Lanelet-Nummer/OSM-Weg-Nummer>:<Spurnummer>.

Liegt der Szenenbeschreibung eine OSM-Karte zugrunde, so werden die verschiedenen Spuren eines Weges durch die Spurnummer unterschieden. Bei der Verwendung von Lanelets ist die Spurnummer immer 0, da ein Lanelet immer nur eine Spur repräsentiert.

Im zweiten Beispiel, dem Kreuzungsszenario (Abbildung 4.19), ist gut zu erkennen, dass sich die verschiedenen Spuren innerhalb einer Kreuzung schneiden oder überlappen und dementsprechend mit den Relationen *überlappt* verbunden sind.

Fügt man der Szene Fahrzeuge und entsprechende Pfade p hinzu, so spiegelt sich das auch in der jeweiligen ABox wieder. In Abbildung 4.20 sind ausgewählte Relationen zu einem Pfad eines Fahrzeugs, das auf der rechten Spur fährt, farbiger dargestellt. Die Umgebungsobjekte U_1 , U_2 und U_4 sind Teil des Pfads ($hat(p, U_1)$)

4 Semantische Szenenmodellierung

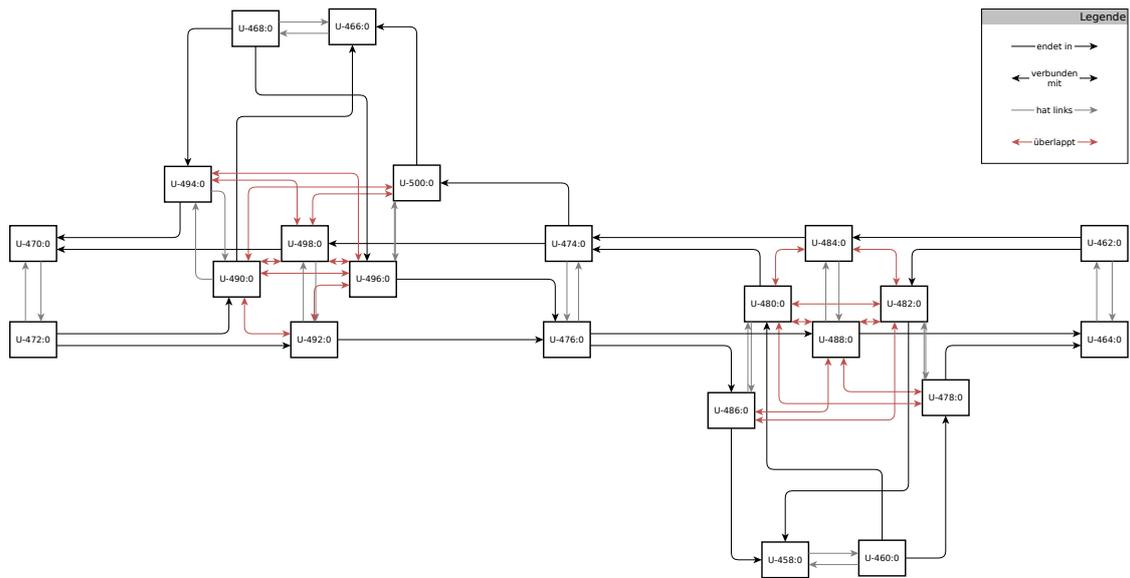


Abbildung 4.19: Ausgewählte Entitäten und Relationen aus der ABox des Kreuzungsszenarios

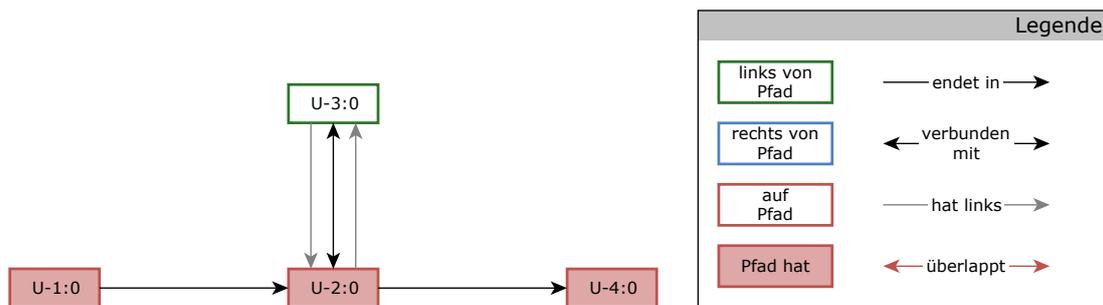


Abbildung 4.20: Ausgewählte Entitäten und Relationen aus der ABox des Bundesstraßenszenarios. Relationen zu einem Pfad eines Fahrzeugs sind farblich markiert.

und somit rot gefüllt. Die Relationen *auf* (rot), *links von* (grün) und *rechts von* (blau) werden durch die jeweilige Rahmenfarbe repräsentiert.

Zur besseren Illustration der Auswirkungen verschiedener Pfadverläufe werden im Kreuzungsszenario drei unterschiedliche Pfade betrachtet. Der erste Pfad verläuft von links nach rechts durch die Szene (grüner Pfad in Abbildung 4.16). Es gibt somit 5 Fahrspursegmente, die zum Pfad gehören (*hat(p, U)*, rot gefüllt in Abbildung 4.21). Kreuzende Fahrspursegmente gehören zwar nicht direkt zum Pfad, werden jedoch als auf dem Pfad liegend (*auf(U, p)*, roter Rahmen) modelliert. Straßensegmente in Straßen links und rechts vom Pfad werden entsprechend als links oder rechts des Pfads markiert. Dies ermöglicht die Modellierung von Verkehrsregeln wie „Rechts vor links“.

Der hier gezeigte Pfad wird in weiteren Beispielen der des Ego-Fahrzeugs sein. Er

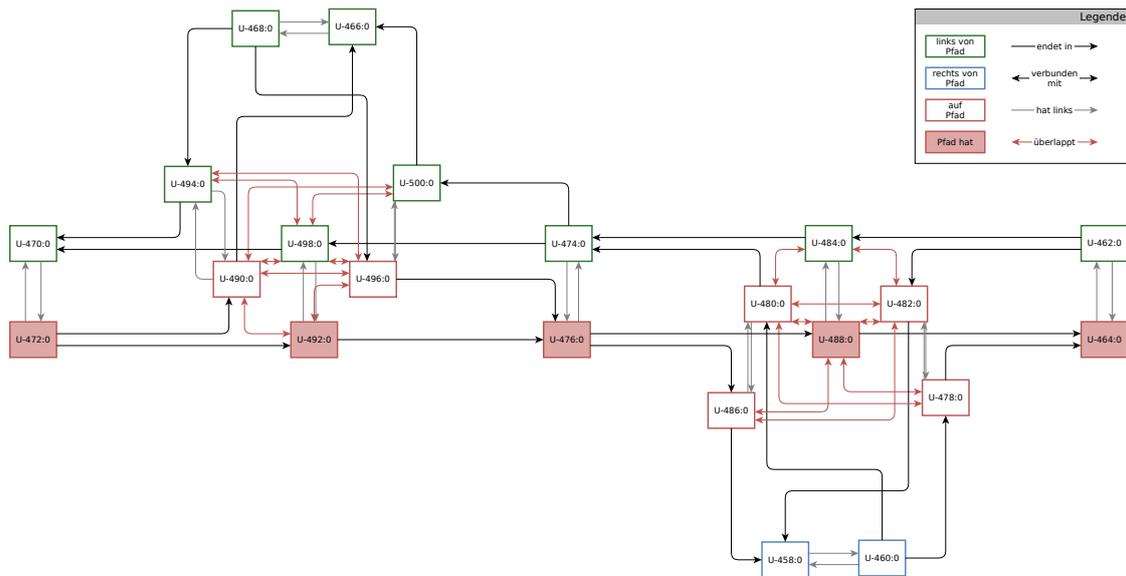


Abbildung 4.21: Ausgewählte Entitäten und Relationen aus der ABox des Kreuzungsszenarios. Relationen zu einem Pfad eines geradeaus fahrenden Fahrzeugs sind farblich markiert.

wird wie in Kapitel 4.3.1 zum Aufbau des Zustandsraums verwendet. Außerdem kann er innerhalb der Ontologie zur Ableitung von Verkehrsregeln und dem Verhalten anderer Verkehrsteilnehmer dienen.

Der zweite Pfad, verläuft von oben links nach unten rechts und biegt an der ersten Kreuzung zunächst links ab (roter Pfad in Abbildung 4.16). Abbildung 4.22 zeigt einen Ausschnitt der ABox der entstehenden Szenenbeschreibung. Auch hier sind wieder die Straßensegmente links und rechts des Pfads farblich gekennzeichnet.

Treffen zwei Fahrzeuge auf den hier gezeigten Pfaden an der Kreuzung aufeinander, so muss das Fahrzeug, das von oben kommt Vorfahrt gewähren und vor der Kreuzung abbremsen. Dies kann z. B. durch einen entsprechend eingesetzten IMM-Filter in der Szenenprädiktion vorhergesagt werden. Der relevanter Pfad des Fahrzeuges beschränkt sich somit auf das Fahrspursegment U_{468} .

Auch dies kann in der relationalen Szenenbeschreibung dargestellt werden. Abbildung 4.23 zeigt einen solchen verkürzten Pfad. Aufgrund der Kürze des Pfades ist es nicht möglich, alle Umgebungsobjekte eindeutig einer linken oder einer rechten Seite zuzuordnen. Die Relationen können dennoch auf Basis von topologischen und geometrischen Eigenschaften festgelegt werden. Für die weitere semantische Planung ergeben sich hieraus keine negativen Einschränkungen. Es ist jedoch zu beachten, dass für eine verlässliche Aussage über die relative Position eines Fahrzeugs zu einem anderen Fahrzeug dessen geplanter Pfad über die hierfür relevante Länge vollständig bekannt sein muss. Die relative Position anderer Verkehrsteilnehmer (z. B. für die Ableitung von Verkehrsregeln) kann unter Kenntnis des eigenen geplanten Pfades bestimmt werden.

4 Semantische Szenenmodellierung

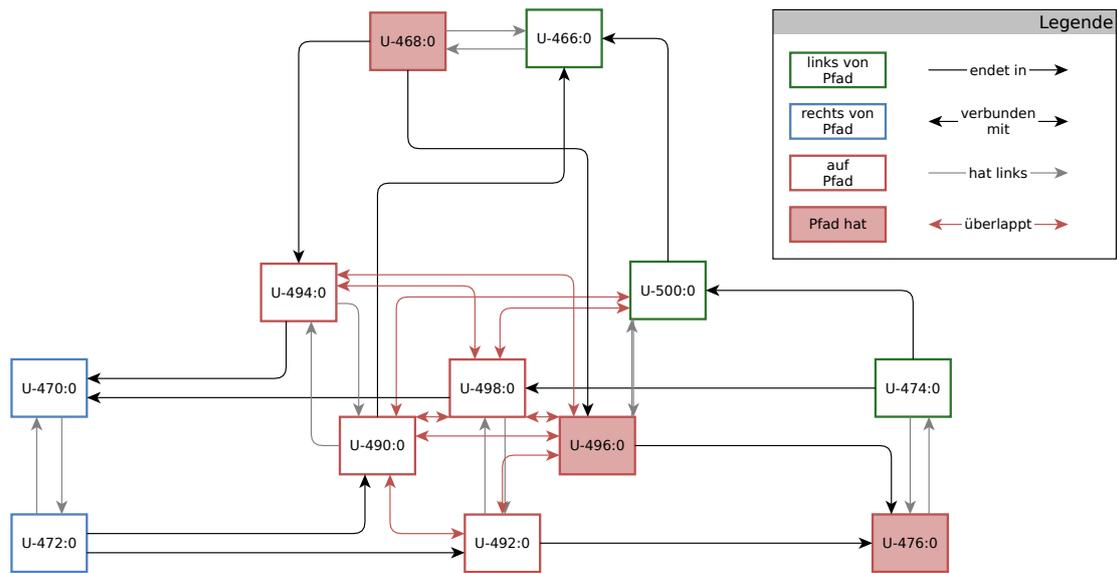


Abbildung 4.22: Ausgewählte Entitäten und Relationen aus der ABox des Kreuzungsszenarios. Relationen zu einem Pfad eines abbiegenden Fahrzeugs sind farbig markiert.

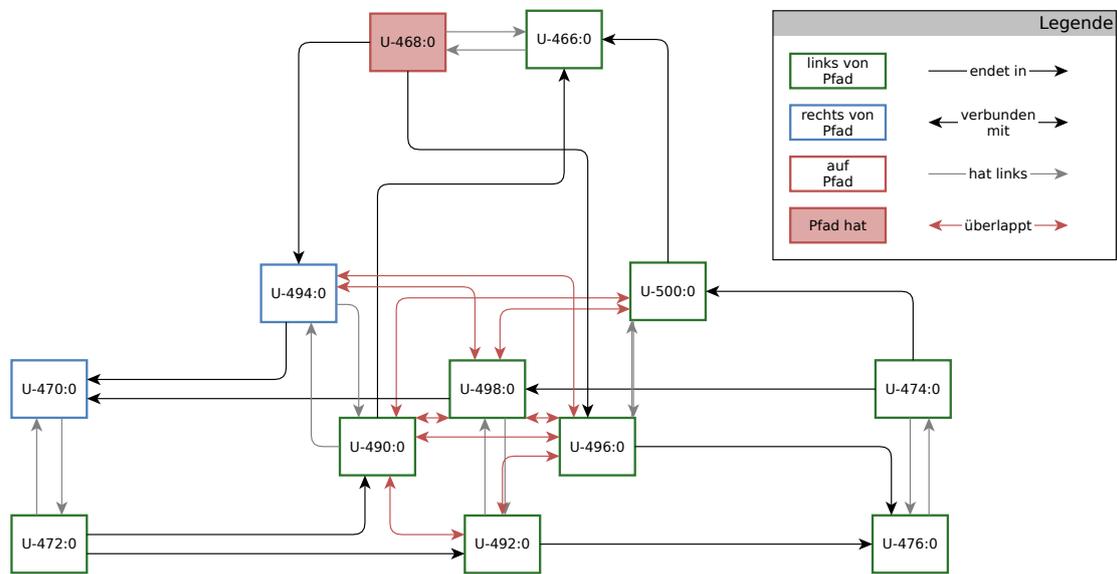


Abbildung 4.23: Ausgewählte Entitäten und Relationen aus der ABox des Bundesstraßenszenarios. Relationen zu einem Pfad eines stoppenden Fahrzeugs sind farbig markiert.

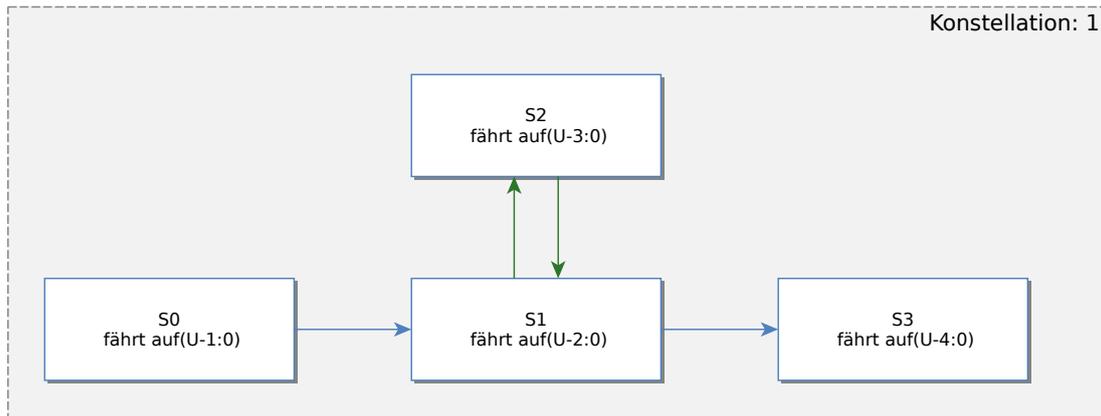


Abbildung 4.24: Resultierender Zustandsraum für das Bundesstraßenszenario ohne andere Verkehrsteilnehmer

4.4.3 Semantischer Zustandsraum

Auf Basis der semantischen Szenenrepräsentation kann der Zustandsraum für die Planung automatisiert abgeleitet werden. Der resultierende Zustandsraum für das Bundesstraßenszenario ohne andere Verkehrsteilnehmer ist in Abbildung 4.24 dargestellt. Die betrachtete Szene stimmt bis auf das Fehlen von U_5 mit der Beispielszene aus Abbildung 4.7 überein. Daher entspricht der Zustandsraum auch bis auf das Fehlen der zu U_5 korrespondierenden Zustände dem in Abbildung 4.8 gezeigten.

Der resultierende Zustandsraum für das Bundesstraßenszenario mit einem anderen Fahrzeug v_1 ist in Abbildung 4.25 dargestellt. Aufgrund der Bewegung des Fahrzeugs ergeben sich 5 Konstellationen. Die ersten drei Konstellationen entsprechen dabei erwartungsgemäß den in Abbildung 4.13 konzeptionell hergeleiteten (bis auf das Fehlen der zu U_5 korrespondierenden Zustände). In Kapitel 4.3.4 wurden zu Gunsten der Übersichtlichkeit nur die ersten drei Konstellationen dargestellt.

Soll der Zustandsraum, wie in Kapitel 4.3.3 besprochen, zeitlich beschränkt werden, so ist es in der Regel nicht sinnvoll lediglich Konstellationen zu streichen. Eine nachgelagerte semantische Planung wäre ggf. nicht mehr in der Lage einen Pfad zu einem Zielzustand zu finden. Würde man den hier betrachteten Zustandsraum z. B. auf zwei Konstellationen beschränken, so wäre ein Überholen des Fahrzeugs nicht mehr abbildbar.

Um auch in diesen Fällen korrekte semantische Pfade zu finden, kann das Abstraktionsniveau der semantischen Szenenbeschreibung angepasst werden. In einer höheren Abstraktion werden z. B. nur Relationen zu statischen Szenenobjekten berücksichtigt. Die Ontologie ist hier so formuliert, dass in einem Zustandsraum basierend auf diesen Relationen immer ein Pfad zu einem Ziel gefunden werden

4 Semantische Szenenmodellierung

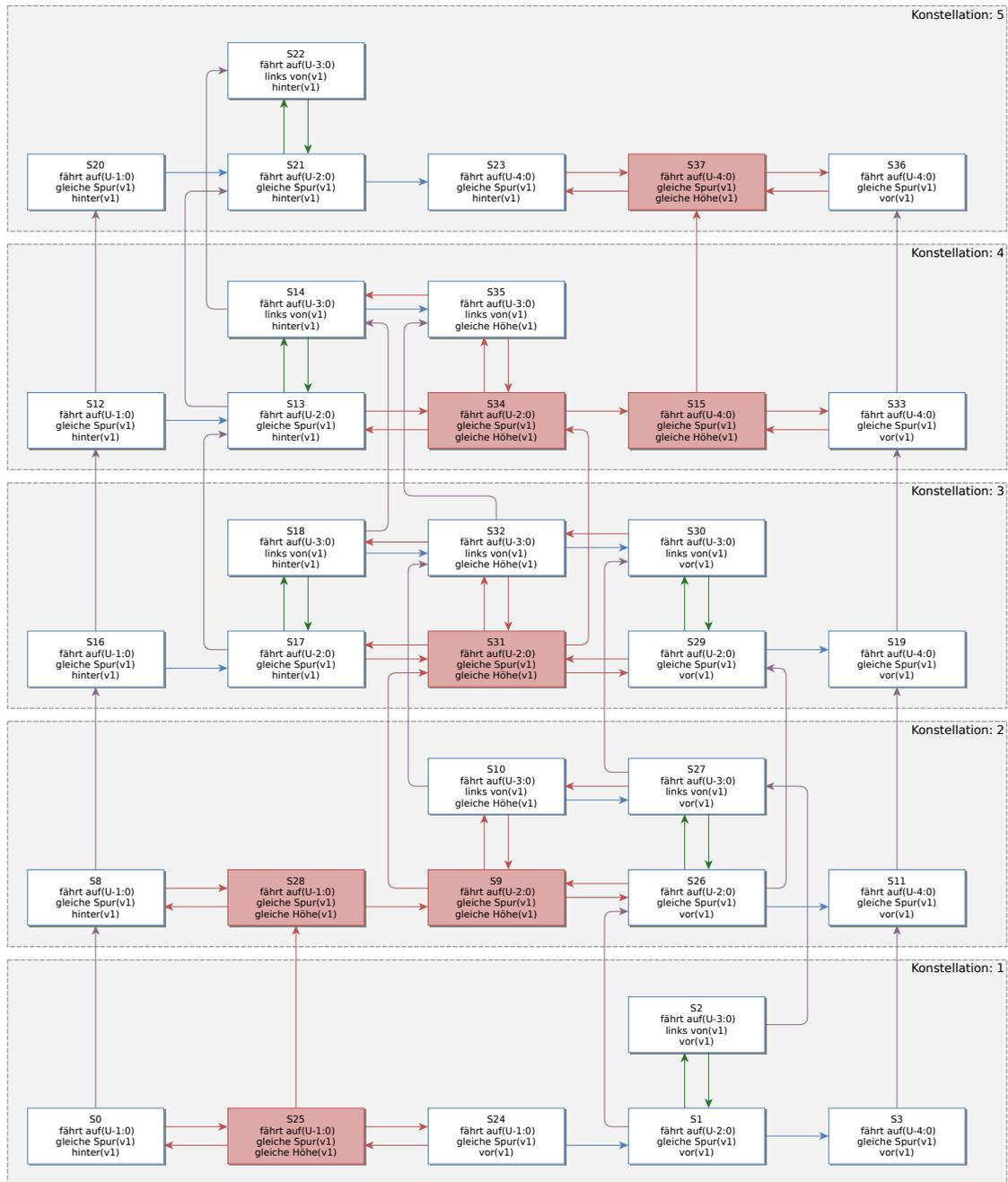


Abbildung 4.25: Resultierender Zustandsraum für das Bundesstraßenszenario mit einem anderen Fahrzeug

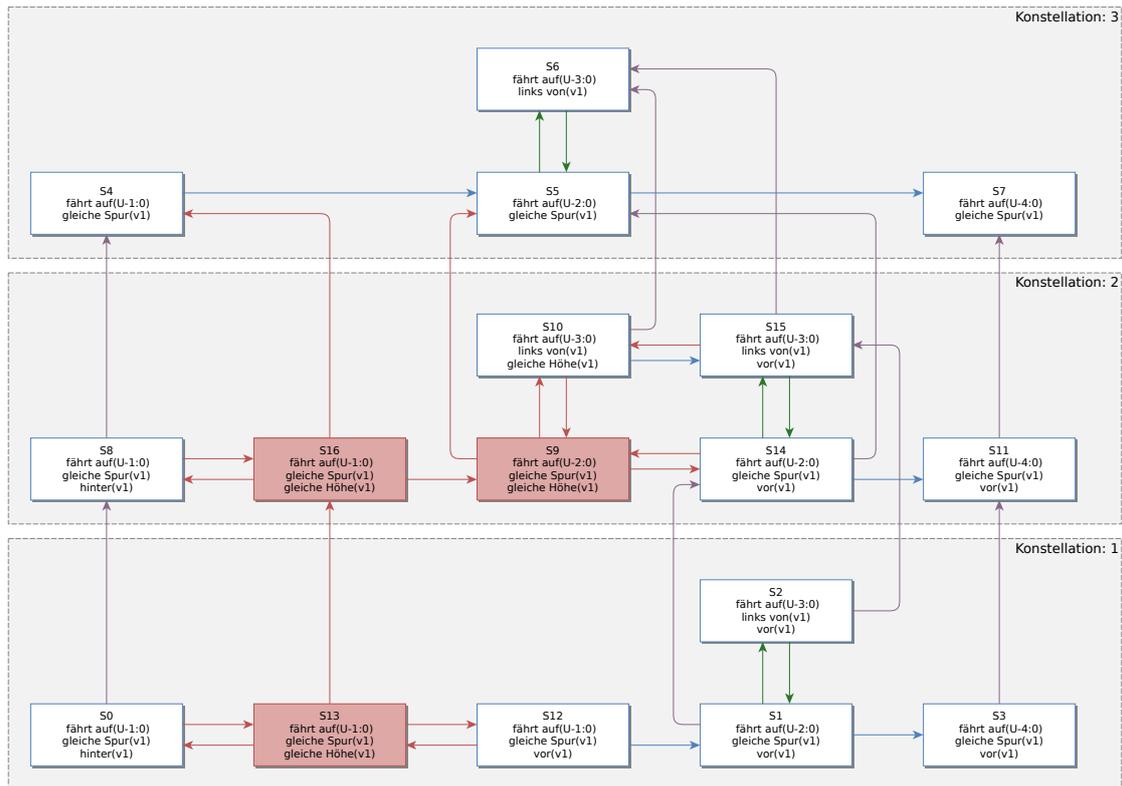


Abbildung 4.26: Resultierender Zustandsraum für das Bundesstraßenszenario mit einem anderen Fahrzeug beschränkt auf einen Zeithorizont von 10 s

kann, wenn dieser auch unter Berücksichtigung der gesetzten Regeln in in der Realität existiert.

Die bei der zeitlichen Beschränkung gestrichelten Konstellationen können durch eine Konstellation mit dieser höheren Abstraktion ersetzt werden. In dieser werden keine dynamischen Verkehrsteilnehmer mehr berücksichtigt. Sie wird daher im Folgenden als statische Konstellation bezeichnet. Durch den Einsatz einer statische Konstellation wird eine Planung über längere Strecken mit beschränktem Zeithorizont ermöglicht. Ein auf 10 s beschränkter Zustandsraum ist in Abbildung 4.26 dargestellt. Die dritte Konstellation ist eine statische Konstellation. Bei der Verbindung der letzten dynamischen Konstellation (Konstellation 2) mit der statischen Konstellation durch temporale Transitionen ist zu beachten, dass auch Zustände mit zusätzlichen Relationen bezüglich v_1 mit korrespondierenden Zuständen ohne diese verbunden werden.

Auch für das Kreuzungsszenario können semantische Zustandsräume generiert werden. Beispielhaft ist hier der Zustandsraum für **ego** dargestellt, das sich von links nach rechts durch die Szene bewegt (vergleiche Abbildung 4.21). Ein weiteres Fahrzeug v_1 fährt entlang des in Abbildung 4.22 dargestellten Pfads. Der vollständige Zustandsraum ist in Abbildung 4.27 dargestellt. Abbildung 4.28 zeigt

4 Semantische Szenenmodellierung

eine zeitlich auf drei Konstellationen beschränkte Version.

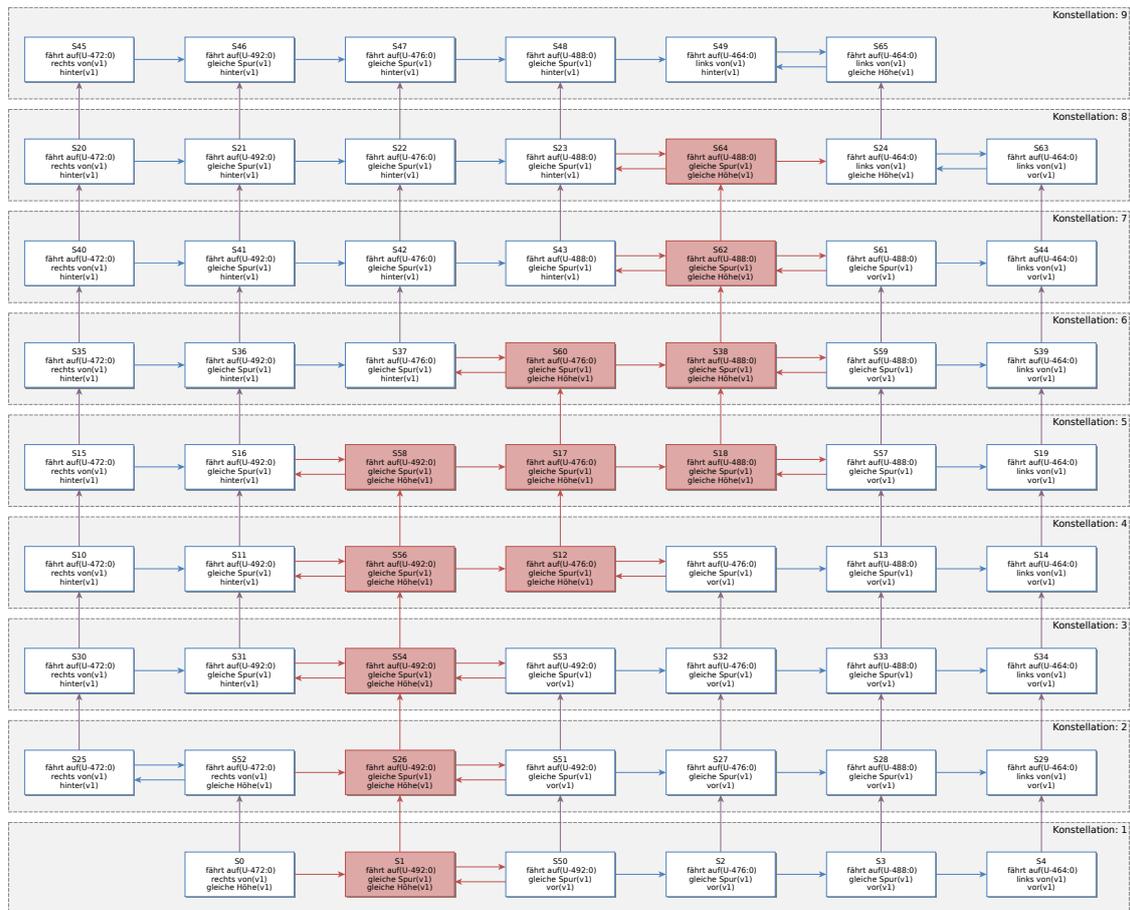


Abbildung 4.27: Resultierender Zustandsraum für das Kreuzungsszenario mit einem anderen Fahrzeug

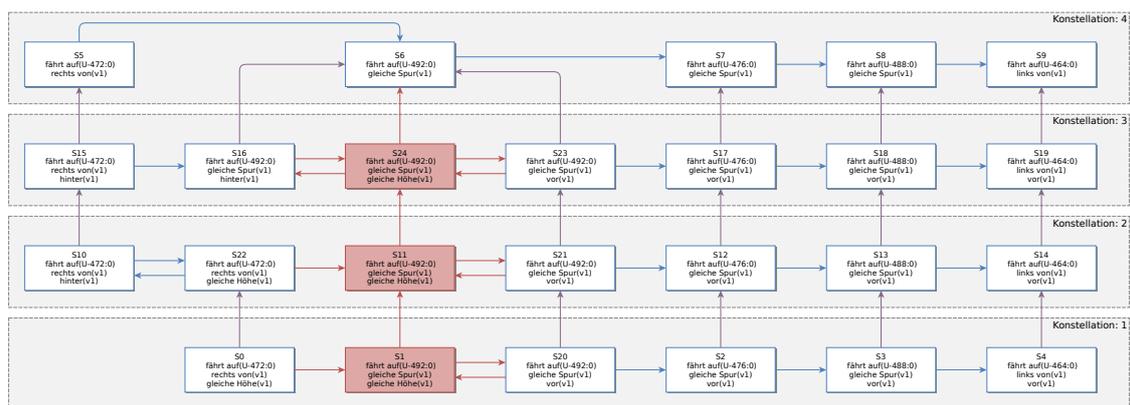


Abbildung 4.28: Resultierender Zustandsraum für das Kreuzungsszenario mit einem anderen Fahrzeug beschränkt auf drei dynamische Konstellationen

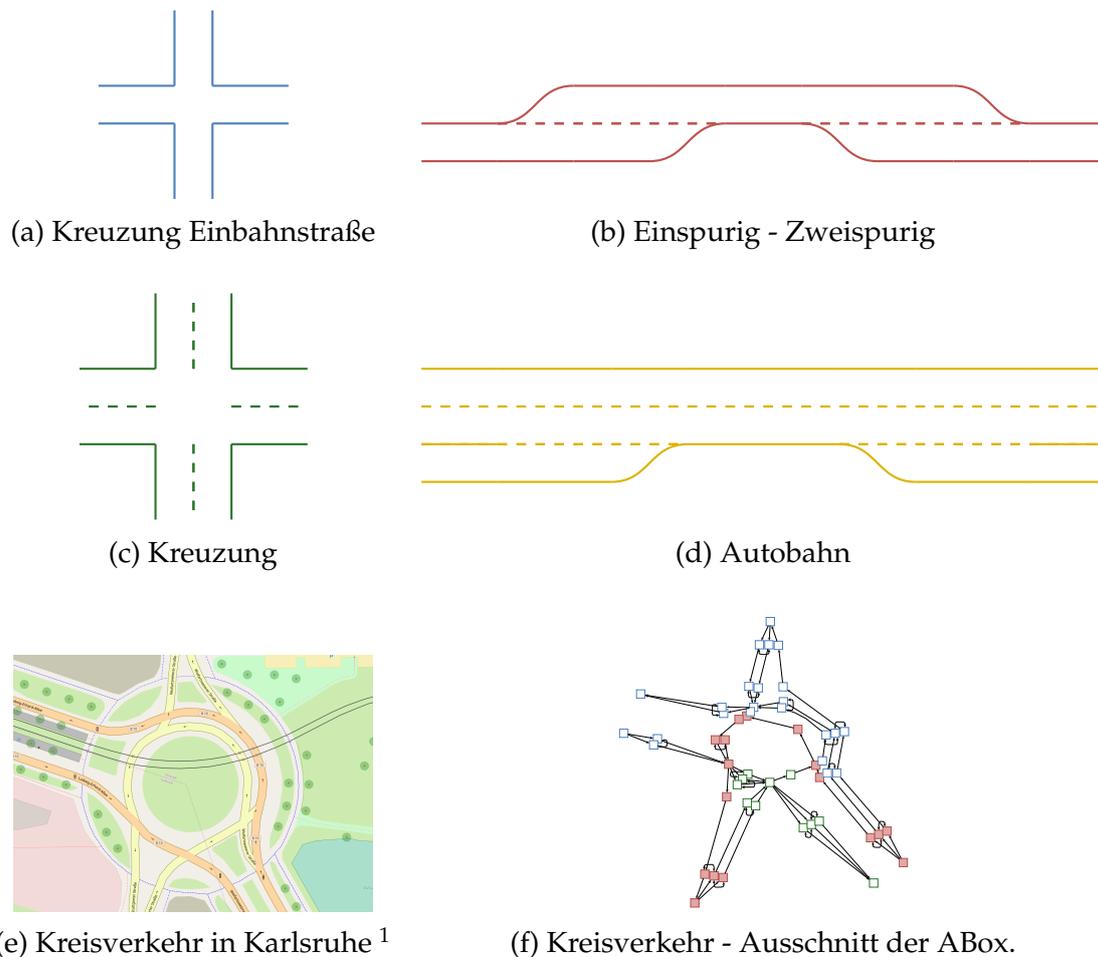


Abbildung 4.29: Beispielszenarien für die Analyse der Zustandsraumgröße

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas et al.)

4.4.4 Größe des Zustandsraums

Eines der konzeptionellen Ziele dieser Arbeit ist die nicht äquidistante ereignis- und strukturbasierte Diskretisierung des Zustandsraums. Demzufolge ist die Größe des Zustandsraums abhängig von der Komplexität der Szene. Die Größe des Zustandsraums wurde in [KBS+14] experimentell anhand von fünf verschiedenen Szenarien untersucht. Diese sind in Abbildung 4.29 abgebildet.

Es wurden zwei Kreuzungsszenarien verwendet. Eines mit zwei sich kreuzenden einspurigen Einbahnstraßen und eines mit zwei zweispurigen Straßen. Das Szenario „Einspurig - Zweispurig“ weist einspurige und zweispurige Streckenabschnitte auf, so wie sie z. B. auf Bundesstraßen vorkommen. Das Autobahnszenario zeigt eine zweispurige Autobahn mit einer Auf- und einer Abfahrt. Als letztes Beispiel wird ein Ausschnitt aus der OSM Straßenkarte von Karlsruhe verwendet. Dieser zeigt den „Oststadtkreisel“ am Schloss Gottesaue. Daneben ist zur Veranschaulichung ein Ausschnitt aus der ABox des Szenarios gezeigt. Darin ist die Trajektorie

¹ OSM Kartenausschnitt. © OpenStreetMap contributors. www.openstreetmap.org

4 Semantische Szenenmodellierung

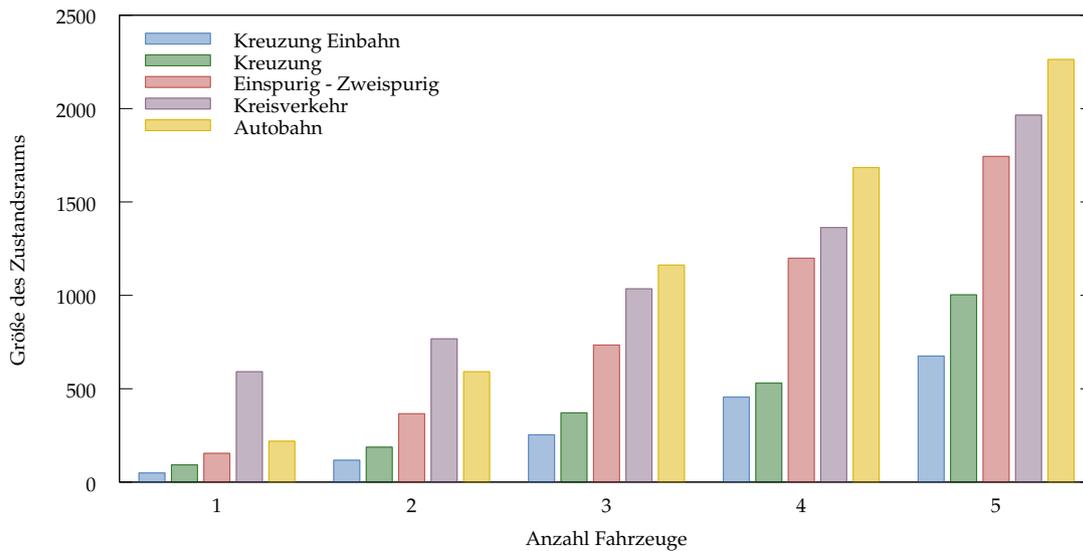


Abbildung 4.30: Vergleich der Größen der Zustandsräume für verschiedene Szenarien und unterschiedlich viele Fahrzeuge

Ähnlich veröffentlicht in [KBS+14] (Kohlhaas, Bittner et al.)

des Ego-Fahrzeugs rot markiert. Es fährt also von Süd-Osten her kommend durch den Kreisverkehr und verlässt diesen in Richtung Süd-Westen.

Alle Szenarien wurden in OSM modelliert und in eine semantische Szenenrepräsentation überführt. Dabei wurden verschieden viele andere Fahrzeuge zufällig in der Szene platziert. Die sich ergebenden Zustandsraumgrößen sind in Abbildung 4.30 dargestellt. Neben der Auswirkung der unterschiedlichen Komplexität der Szenarien auf die Größe des Zustandsraums ist auch ein Anstieg der Größe mit steigender Anzahl von Fahrzeugen zu erkennen. Dieser ist überproportional und entspricht augenscheinlich einem polynomiellen Verlauf. Um dies genauer zu analysieren, wird im Folgenden eine theoretische Betrachtung der Komplexitätsklasse der Zustandsraumgröße durchgeführt.

4.4.5 Komplexitätsklasse der Zustandsraumgröße

Für die Bestimmung der Komplexitätsklasse der Zustandsraumgröße S werden einige Zwischengrößen betrachtet:

U : Anzahl der Umgebungsobjekte bzw. Fahrspursegmente

U_x : Ausdehnung der statischen Szene in longitudinaler Richtung
also entlang der Trajektorie von **ego**

U_y : Maximale Ausdehnung der statischen Szene in lateraler Richtung

D : Anzahl der dynamischen Objekten in der Szene

$S_{kx}(U_x, D)$: Anzahl der Zustände in longitudinaler Richtung in einer Konstellation

$S_k(U_x, U_y, D)$: Anzahl der Zustände in einer Konstellation

$K(U_x, D)$: Anzahl der Konstellationen im Zustandsraum.

Die Anzahl der Zustände innerhalb einer Konstellation hängt von der Anzahl der Zustände des statischen Zustandsraums sowie der Anzahl der dynamischen Objekte ab. Zunächst wird die Ausdehnung in longitudinaler Richtung betrachtet. Da jedes Fahrzeug zwei zusätzliche longitudinale Transitionen einführt, die den Zustandsraum in longitudinaler Richtung um jeweils einen Zustand verlängern, ergibt sich für die Anzahl der Zustände einer Konstellation in longitudinaler Richtung:

$$S_{kx}(U_x, D) \leq U_x + 2D \quad (4.15)$$

Zusammen mit der Ausdehnung des statischen Zustandsraums in lateraler Richtung ist die Anzahl der Zustände innerhalb einer Konstellation wie folgt beschränkt.

$$S_k(U_x, U_y, D) \leq U_y(U_x + 2D) \quad (4.16)$$

Bei der Abschätzung der Anzahl der Konstellationen werden die in Kapitel 4.1 getroffenen Annahmen ausgenutzt. Dies ist im speziellen unter anderem die lineare Prädiktion anderer Verkehrsteilnehmer entlang eines vorgegebenen semantischen Pfades. Die Anzahl der Konstellationen kann somit wie folgt abgeschätzt werden.

$$K(U_x, D) \leq 2D^2 + 2DU_x - 4D + 1 \quad (4.17)$$

Dies wird im folgenden mittels vollständiger Induktion über die Anzahl der dynamischen Szenenobjekte bewiesen.

Beweis :

Als Induktionsanfang wird eine Szene ohne dynamische Objekte (also $D = 0$) verwendet. Hier ergibt sich unabhängig von der Anzahl der statischen Szenenobjekte und somit auch der Ausdehnung der Szene in Längsrichtung U_x genau eine Konstellation $K(U_x, 0) = 1$. Damit ergibt sich mit $2 \cdot 0^2 + 2 \cdot 0 \cdot U_x - 4 \cdot 0 + 1 = 1$

$$K(U_x, 0) \leq 2 \cdot 0^2 + 2 \cdot 0 \cdot U_x - 4 \cdot 0 + 1. \quad (4.18)$$

Es wird nun als Induktionsvoraussetzung die Abschätzung der Anzahl der Konstellationen für D dynamische Objekte wie in Gleichung (4.17) als gegeben angenommen. Die Anzahl der Konstellationen für $D + 1$ dynamische Objekte kann darauf aufbauend durch eine Betrachtung der Grenzfunktionen hergeleitet werden. Ein dynamisches Objekt definiert zwei lineare Grenzfunktionen. Jede dieser Grenzfunktionen hat maximal einen

4 Semantische Szenenmodellierung

Schnittpunkt mit einer der bestehenden $S_{kx}(U_x, D) - 1$ Grenzfunktionen und induziert somit maximal ebenso viele neuen Konstellationen (vergleiche Abbildung 4.11). Somit kann die Anzahl der Konstellationen für $D + 1$ dynamische Objekte wie folgt abgeschätzt werden:

$$K(U_x, D + 1) \leq K(U_x, D) + 2(S_{kx}(U_x, D) - 1) \quad (4.19)$$

Mit Hilfe dieser Abschätzung, der Gleichung (4.16) und der Induktionsvoraussetzung (IV) kann nun im Induktionsschritt die Induktionsbehauptung

$$K(U_x, D + 1) \leq 2(D + 1)^2 + 2(D + 1)U_x - 4(D + 1) + 1 \quad (4.20)$$

für $D + 1$ dynamische Objekte gezeigt werden:

$$\begin{aligned} & K(U_x, D + 1) \leq K(U_x, D) + 2(S_{kx}(U_x, D) - 1) \\ \stackrel{(4.16)}{\Rightarrow} & K(U_x, D + 1) \leq K(U_x, D) + 2((U_x + 2D) - 1) \\ \stackrel{(IV)}{\Rightarrow} & K(U_x, D + 1) \leq 2D^2 + 2DU_x - 4D + 1 + 2U_x + 4D - 2 \\ \Rightarrow & K(U_x, D + 1) \leq 2D^2 + 2DU_x + 2U_x - 1 \quad (4.21) \\ \Rightarrow & K(U_x, D + 1) \leq 2((D + 1)^2 - 2D - 1) + 2(D + 1)U_x - 1 \\ \Rightarrow & K(U_x, D + 1) \leq 2(D + 1)^2 + 2(D + 1)U_x - 4D - 2 - 1 \\ \Rightarrow & K(U_x, D + 1) \leq 2(D + 1)^2 + 2(D + 1)U_x - 4(D + 1) + 1 \end{aligned}$$

Somit gilt die Gleichung (4.17) für eine beliebige positive Anzahl dynamischer Objekte $D \geq 0$. ■

Die Anzahl der Zustände des Zustandsraums lässt sich dann abschätzen durch:

$$\begin{aligned} S(U_x, U_y, D) & \leq K(U_x, D) \cdot S_k(U_x, U_y, D) \\ & \leq 4D^3U_y + 6D^2U_xU_y - 8D^2U_y + 2DU_x^2U_y - 4DU_xU_y + 2DU_y + U_xU_y \end{aligned} \quad (4.22)$$

Mit

$$U_x \leq U \quad (4.23)$$

$$U_y \leq U \quad (4.24)$$

$$U_xU_y \leq U \quad (4.25)$$

folgt

$$S(U_x, U_y, D) \leq 4D^3U + 6D^2U + 2DU^2 + 2DU + U. \quad (4.26)$$

Damit liegt die Größe des Zustandsraums in der Komplexitätsklasse

$$\mathcal{O}(D^3U + DU^2 + U). \quad (4.27)$$

4.5 Probabilistische Betrachtung der semantischen Szenenbeschreibung

Der Zustandsraum wächst also polynomiell mit der Größe der statischen Szene und der Anzahl der dynamischen Objekte. Genauer betrachtet wächst er kubisch mit der Anzahl der dynamischen Objekte und quadratisch mit der Anzahl der statischen Umgebungsobjekte:

$$\begin{aligned} \mathcal{O}(D^3) &\text{ für konstantes } U \\ \mathcal{O}(U^2) &\text{ für konstantes } D \end{aligned} \tag{4.28}$$

Lediglich bei Abwesenheit jeglicher dynamischer Objekte wächst er linear mit der Anzahl der statischen Umgebungsobjekte:

$$\mathcal{O}(U) \tag{4.29}$$

Durch Beschränkung der Konstellationen kann eine niedrigere Komplexitätsklasse erreicht werden. Die kann z. B. durch die Einschränkung auf die Betrachtung eines limitierten Zeithorizonts und die Annahme, dass nur eine beschränkte Anzahl von Ereignissen innerhalb dieses eintreten können, erreicht werden. Beschränkt man die Anzahl der Konstellationen liegt die Größe des Zustandsraums in der Komplexitätsklasse $\mathcal{O}(U_y(U_x + 2D))$ oder durch Ausnutzung von Gleichung (4.25) in $\mathcal{O}(U + DU)$ und ist somit linear in beiden Eingangsgrößen.

4.5 Probabilistische Betrachtung der semantischen Szenenbeschreibung

Aufgrund der beschränkten Wahrnehmung und dem daraus resultierenden unvollständigen und unscharfen Wissen über das Verkehrsumfeld ist die Verwendung von probabilistischen Verfahren in der kognitiven Verarbeitungskette eines automatisierten Fahrzeugs notwendig. Durch das hohe Abstraktionsniveau der semantischen Manöverplanung werden probabilistische Effekte aus vorgelagerten Perzeptionskomponenten reduziert, so dass diese für viele Anwendungen auch vernachlässigt werden können. Dennoch sollen im Folgenden Möglichkeiten zur Integration von Unsicherheiten und Unschärfen in der Szenenbeschreibung in den semantischen Zustandsraum, wie sie zum Teil in [Bat15] erarbeitet wurden, vorgestellt werden. Zunächst werden verschiedene Möglichkeiten der Unsicherheit in den Eingangsdaten besprochen. Darauf aufbauend werden verschiedene Modellierungsmöglichkeiten dieser Unsicherheiten im Zustandsraum vorgestellt. Abschließend wird die Auswirkung dieser Anpassungen auf eine spätere Manöverplanung diskutiert.

4.5.1 Varianten der Unsicherheit

Bei der Wahrnehmung und Interpretation einer Verkehrsszene treten verschiedene Formen der Unsicherheit und Unschärfe auf:

Unschärfe in der Wahrnehmung

Aufgrund von Sensorungenauigkeiten und Rauschen in den Eingangssignalen können die physikalischen Größen wie Position und Geschwindigkeit anderer Verkehrsteilnehmer nicht exakt bestimmt werden (Abbildung 4.31). Dies führt zu einer Unschärfe in den zugehörigen Daten.

Unsicherheit in der Wahrnehmung

Bei der Wahrnehmung der Umgebung können auch Unsicherheiten auftreten. So führen z. B. Fehldetektionen zur Annahme, dass Objekte in der Szene vorhanden seien, die in der Realität nicht existieren. Andersrum können real existierende Objekte auch in der Umfeldrepräsentation fehlen, weil sie außerhalb der Sensorabdeckung liegen, durch andere Objekte verdeckt sind oder einfach durch die Sensorik- und Perzeptionskomponenten nicht erkannt wurden. Abbildung 4.32 zeigt den Fall, in dem die Existenz eines Fahrzeugs aufgrund von Verdeckung durch ein anderes Fahrzeug nicht festgestellt werden kann.

Unsicherheit in der Prädiktion

Selbst bei perfekter Umfeldwahrnehmung ist eine exakte Prädiktion der an der Verkehrsszene beteiligten Agenten nicht möglich. Dies führt zu Unsicherheiten und Unschärfen in den prädizierten Daten. Beispielsweise lässt sich ein Spurwechsel oder Abbiegevorgang häufig nicht direkt aus der Umfeldbeobachtung herleiten. In dem in Abbildung 4.33 dargestellten Fall hat die unbekannte Routenwahl des Fahrzeugs starken Einfluss auf die Fahrentscheidung des Ego-Fahrzeugs.

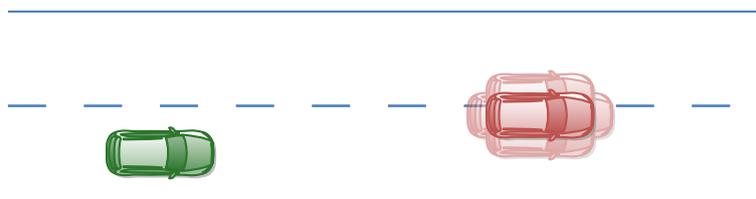


Abbildung 4.31: Unschärfe in der Wahrnehmung – die Position des vorausfahrenden Fahrzeugs kann nicht exakt bestimmt werden.

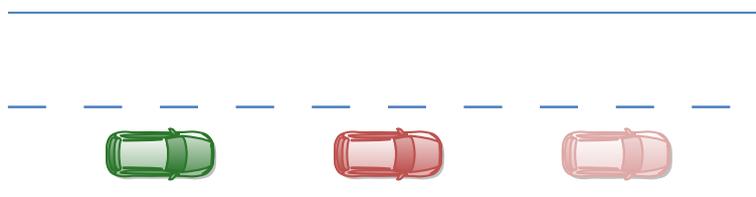


Abbildung 4.32: Unsicherheit in der Wahrnehmung – durch ein vorausfahrendes Fahrzeug ist die Sicht versperrt.

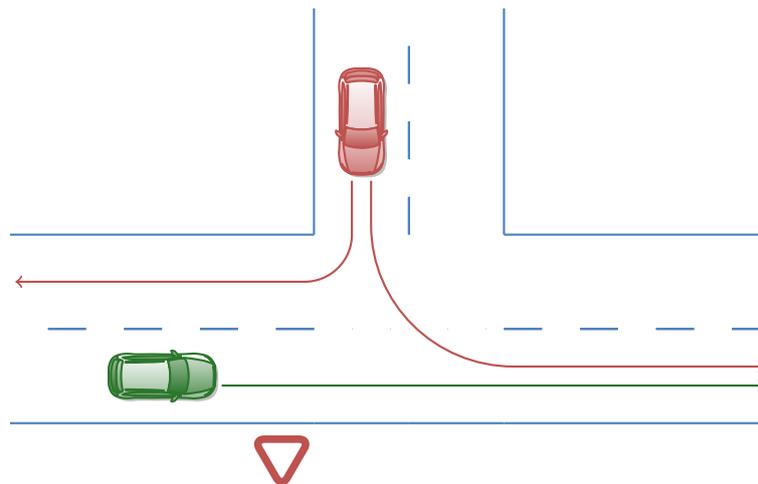


Abbildung 4.33: Unsicherheit in der Prädiktion – Beispiel an einer Kreuzung mit abknickender Vorfahrt.

4.5.2 Ansätze zur probabilistischen Modellierung des Zustandsraums

Im Folgenden werden für die verschiedenen Arten von Unsicherheiten Modellierungsvorschläge vorgestellt und ihre Anwendbarkeit diskutiert.

Erschöpfende Kombinatorik

Wird davon ausgegangen, dass die Unschärfe der Positionsbestimmung anderer Verkehrsteilnehmer durch unendlich ausgedehnte Wahrscheinlichkeitsverteilungen wie z. B. Gauß-Verteilungen modelliert wird, so können örtliche Relationen zwischen zwei Objekten beliebige Ausprägungen annehmen. Es kann also davon ausgegangen werden, dass jegliche mögliche Kombination von Relationen als Zustand im Zustandsraum realisierbar ist.

Aufgrund der angenommenen Wahrscheinlichkeitsverteilungen können diesen Zuständen dann Existenzwahrscheinlichkeiten zugeordnet werden. Eine Aufteilung des Zustandsraums in einzelne Konstellationen ist somit nicht mehr möglich. Vielmehr müsste die Existenzwahrscheinlichkeit der Zustände in Abhängigkeit der Zeit modelliert werden.

Dieser Ansatz der Modellierung ist sehr mächtig. Jedoch ist die zu erwartende Struktur des Zustandsraums für eine anschließende Manöverplanung in Form einer Graphensuche nicht geeignet. Daher wird dieser Ansatz hier nicht weiter verfolgt. Für andere Methoden zur Findung von Fahrentscheidungen z. B. durch Ansätze des maschinellen Lernens kann dieser Ansatz jedoch von größerem Interesse sein.

4 Semantische Szenenmodellierung

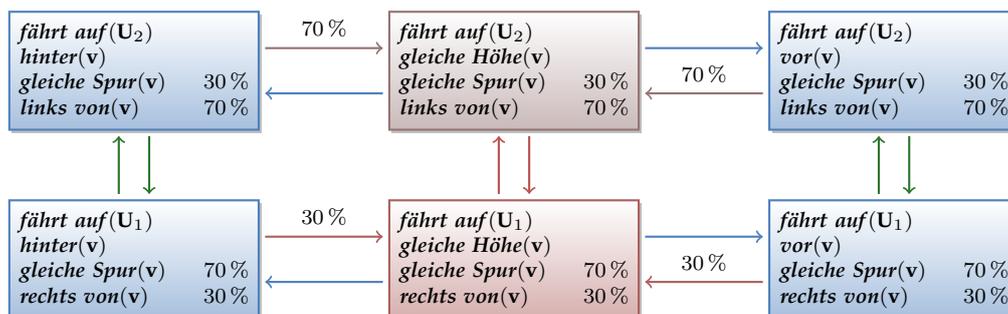


Abbildung 4.34: Beispielhafter Zustandsraum mit wahrscheinlichkeitsbehafteten Relationen in den Zuständen basierend auf der Szene in Abbildung 4.31

In Anlehnung an [Bat15]

Wahrscheinlichkeitsbehaftete Relationen

Laterale Relationen zu anderen Verkehrsteilnehmern haben keinen Einfluss auf die Struktur des Zustandsraums, sondern nur auf die an den Zuständen beteiligten Relationen und somit auf die Validität von Transitionen. Daher kann eine Unsicherheit in der Existenz dieser Relationen direkt in den Zuständen modelliert werden, indem alle möglichen lateralen Relationen mit ihren jeweiligen Wahrscheinlichkeiten in die Zustandsrepräsentation aufgenommen werden.

Abbildung 4.34 zeigt beispielhaft eine mögliche Konstellation für das Szenario aus Abbildung 4.31. Auf die Anwendung von Verkehrsregeln wurde zur besseren Verständlichkeit verzichtet. Die beiden mittleren Zustände können zu einer bestimmten Wahrscheinlichkeit (hier 30 % bzw. 70 %) nicht befahren werden. Diese Wahrscheinlichkeiten werden in den Bedingungen der entsprechenden Transitionen berücksichtigt. Invalide Transitionen (rot) haben demnach Bedingungen mit Wahrscheinlichkeit $P(C = \text{wahr}) = 0\%$. In der Abbildung wurden zur besseren Übersichtlichkeit Transitionen mit Bedingungswahrscheinlichkeit 100 % und 0 % nicht beschriftet. Bei der Planung kann so durch Multiplikation der Bedingungswahrscheinlichkeiten eine Wahrscheinlichkeit für die Durchführbarkeit eines Plans ermittelt werden.

Besonderes Augenmerk ist auf korrelierte Wahrscheinlichkeiten zu legen. Die beiden lateralen Transitionen zwischen den mittleren Zuständen hätten entsprechend der Wahrscheinlichkeiten ihrer Zielzustände nur eine Bedingungswahrscheinlichkeit von 30 % bzw. 70 %. Aufgrund der Korrelation der Wahrscheinlichkeiten der beiden Zustände ist jedoch unter der Bedingung, dass sich das Ego-Fahrzeug erfolgreich in einem der beiden Zustände befindet, der Übergang in den jeweils anderen Zustand unmöglich.

Unschärfe Transitionen

Die Unschärfe von Objektpositionen und -geschwindigkeiten in Längsrichtung hat ebenfalls Auswirkungen auf den Zustandsraum. Im Folgenden wird von örtlich beschränkten Wahrscheinlichkeitsverteilungen (also z. B. keine Gaußverteilungen) mit geringer Varianz, im Vergleich zu Fahrzeugabständen und Fahrbahnlängen, ausgegangen. Die Struktur des Zustandsraums bleibt dadurch erhalten.

Die Unschärfen können entweder durch Sensorungenauigkeiten oder aber durch Verdeckungen des Sichtbereichs resultieren. Eine Unschärfe in der Position oder Geschwindigkeit eines Fahrzeugs kann direkt auf eine Unschärfe der Grenzfunktionen zur Transitionsbildung abgebildet werden. Ist z. B. die Sicht nach vorne durch ein vorausfahrendes Fahrzeug (wie in Abbildung 4.32) versperrt, kann die Existenzunsicherheit eines zweiten Fahrzeugs als Unschärfe in der Länge der vorausfahrenden Kolonne aufgefasst werden. Die Kolonnenlänge kann ggf. aus der aktuellen Verkehrsdichte und statistischen Informationen zur Strecke geschätzt werden. In diesem Fall wird für die Generierung des Zustandsraums nicht ein einzelnes Fahrzeug mit Sicherheitsabständen betrachtet, sondern eine ganze Fahrzeugkolonne. Dies führt dann zu einer Unschärfe der Grenzfunktion zum vorderen Sicherheitsabstand hin.

Die Unschärfe einer Grenzfunktion wirkt sich auch auf die Zeitpunkte zwischen zwei Konstellationen aus. Geht man von linearen Bewegungsmodellen der Fahrzeuge aus, so ist die Wahrscheinlichkeitsverteilung der Konstellationsgrenzen linear abhängig von denen der Grenzfunktionen. Es ergeben sich also auch zeitlich unscharfe Grenzen zwischen den Zuständen.

Die hier beschriebene Modellierung von Unsicherheiten wurde exemplarisch in die Algorithmik zur Zustandsraumerstellung integriert und an verschiedenen Szenen analysiert. Abbildung 4.35 zeigt die Auswirkung einer Unschärfe in der Objektwahrnehmung. Die Position des vorausfahrenden Fahrzeugs (blauer Rahmen) kann nicht exakt bestimmt werden. Dies führt zu einer Unschärfe in den zugehörigen Grenzfunktionen, was sich wiederum auf eine Unschärfe in den Konstellationsgrenzen auswirkt. Der aktuelle Zeitpunkt liegt im Bereich der Wahrscheinlichkeitsverteilung einer Konstellationsgrenze. Somit ergibt sich eine Verteilung des aktuellen Zustands des Ego-Fahrzeugs über zwei Konstellationen hinweg. Die Verteilung ist durch entsprechende Grünschattierungen sowie eine Wahrscheinlichkeit an den entsprechenden Zuständen dargestellt.

Unschärfe Eigenpositionierung

Die Position des Ego-Fahrzeugs hat keine Auswirkung auf den Zustandsraum. Sie beeinflusst lediglich die Wahl des Startzustands bei der anschließenden Planung. Sowohl unscharfe Zustandsgrenzen als auch eine unscharfe Verortung des Ego-Fahrzeugs führen dazu, dass im Allgemeinen nicht ein eindeutiger Startzustand zugeordnet werden kann. Hier ergibt sich je nach Modellierung der Unschärfen

4 Semantische Szenenmodellierung



Abbildung 4.35: Anfahrt auf eine Ampel – Unschärfe in der Wahrnehmung wirkt sich auf die Konstellationsgrenzen aus. Der aktuelle Zeitpunkt und somit der Startzustand (grün) wird zwei Konstellationen zugeordnet.

eine Verteilung über mehrere Startzustände ggf. in mehreren Konstellationen. Auf die Struktur des Zustandsraums hat dies zwar keine Auswirkung, angeschlossene Planungsverfahren müssen hingegen mit der Mehrdeutigkeit des Startzustands umgehen können.

Auch die Unschärfe in der Eigenlokalisierung wurde exemplarisch analysiert. Abbildung 4.36 zeigt das Ego-Fahrzeug bei einem Spurwechsel. Aufgrund der Positionsunschärfe verteilt sich der aktuelle Zustand des Fahrzeugs auf zwei Fahrspuren und somit auf zwei Zustände (grün).

Szenarienbasierte Zustandsräume

Je nach Ausprägung der Prädiktion ist es möglich für eine gegebene Szene verschiedene Entwicklungen dieser zu präzisieren. Dies resultiert in einer Menge von möglichen Szenarien. Der Vorteil dieser Darstellung ist, dass Korrelationen zwischen verschiedenen wahrscheinlichkeitsbehafteten Größen schon durch die Szenarien mitberücksichtigt werden können. So hängt das Geschwindigkeitsprofil

4.5 Probabilistische Betrachtung der semantischen Szenenbeschreibung

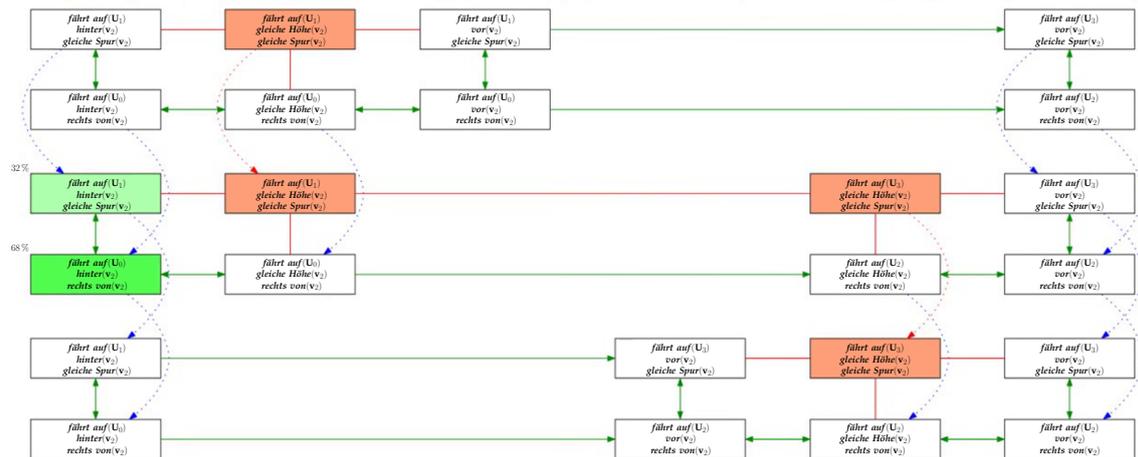


Abbildung 4.36: Anfahrt auf eine Ampel – Unschärfe in der Eigenpositionierung wirkt sich auf den aktuelle der Startzustand (grün) aus.

eines Fahrzeugs maßgeblich von dessen gefahrenen Manöver ab. Ein Fahrzeug, das einem anderen folgt wird z. B. langsamer fahren, als wenn es dieses überholt. Diese Art Information kann z. B. von der in Kapitel 2.6.1 vorgestellten Prädiktion generiert werden.

Liegt eine Prädiktion in Form von solchen Szenarien vor, so liegt es nahe, auch für jedes dieser Szenarien einen eignen Zustandsraum zu erzeugen und auf diesem zu planen. In der in Abbildung 4.33 gezeigten Situation würden demnach zwei Zustandsräume für die beiden Routen des roten Fahrzeugs erzeugt. Ähnlich kann auch zur Berücksichtigung von Existenzwahrscheinlichkeiten einzelner Objekte verfahren werden. So würde in der Situation aus Abbildung 4.32 ein Zustandsraum mit einem und einer mit zwei Fahrzeugen erzeugt werden.

Während diese Modellierung auf die Bildung des Zustandsraums und die Planung auf diesem keinen Einfluss hat, liegt eine Herausforderung in der Interpretation der Planungsergebnisse. Für eine finale Fahrentscheidung müssen mehrere Pläne zusammengefasst und ein optimales Manöver aus den erstellten Manöverketten gefunden werden. Eine mögliche Umsetzung dieses Ansatzes in Zusammenspiel mit einer szenarienbasierten Prädiktion und einem Trajektorienplaner wird in Kapitel 7 dargestellt.

Zusammenfassungen von semantischen Pfaden

Für eine sichere, passive Fahrweise kann es ausreichend sein, alle möglichen Fahrwege der anderen Verkehrsteilnehmer gemeinsam zu betrachten. So ist eine Berücksichtigung der Prädiktionsunsicherheit in der semantischen Szenendarstellung und somit auch in dem daraus erzeugten Zustandsraum möglich. Dazu werden alle Fahrspursegmente, die wahrscheinlich durch das Fahrzeug befahren werden zu seinem Pfad zugehörig modelliert, auch wenn diese nicht einen sondern mehrere verschiedene Fahrwege abbilden, wie z. B. in Abbildung 4.33 dargestellt. Dies führt dazu, dass mehrere Zustände mit der Relation *gleiche Spur(vehicle)* angereichert werden. Gleiches ist auch bei Unschärfen in der Wahrnehmung möglich.

In der Situation aus Abbildung 4.31 kann das Fahrzeug nicht eindeutig einer Spur zugeordnet werden. Fasst man die beiden resultierenden Pfade zusammen, wird die gesamte Straße durch das Fahrzeug blockiert. Dies ist zum einen ein gewünschtes Verhalten z. B. bei Spurwechseln oder an Kreuzungen. Auf der anderen Seite können bei dieser Modellierung von Unsicherheiten auch potentielle Lösungen nicht gefunden werden, weil die Korrelation der Wahrscheinlichkeitsverteilungen verloren gehen. So kann z. B. auch unter der Annahme, dass das vorausfahrende Fahrzeug auf einer der beiden Spuren fährt, bei entsprechender Unschärfe der Wahrnehmung kein Überholmanöver geplant werden.

Auch dieses Verfahren kommt in der in Kapitel 7 beschriebenen Umsetzung zum Einsatz. Hier kann z. B. in einem Kreuzungsszenario das Abbiegen in verschiedene Richtungen zusammengefasst werden und so lediglich zwischen einem Anhalten an der Kreuzung und einem Einfahren in die Kreuzung unterschieden werden.

4.6 Zusammenfassung und Fazit

Aufgabe der semantischen Szenenmodellierung ist die Ableitung eines diskreten Planungsraums für das automatisierte Ego-Fahrzeug aus der geometrischen Umweltmodellierung. Dieser lässt sich als gerichteter Graph darstellen, wobei Zustände als Knoten und mögliche Aktionen so genannte Transitionen als Kanten repräsentiert werden.

Die aktuelle Verkehrsszene wird mit Hilfe einer relationalen Szenenmodellierung in Form einer Verkehrsszenenontologie repräsentiert. Ausgehend von den Relationen des Ego-Fahrzeugs zu anderen Elementen der Verkehrsszene wird ein Planungsraum aufgebaut. In diesem vierschrüttigen Verfahren werden zunächst alle Relationen zu statischen Szenenobjekten und anschließend zu allen dynamischen Szenenobjekten betrachtet. Die Prädiktion der dynamischen Szenenobjekten führt zu Veränderungen in der Szenen die in sogenannten Konstellationen, Subgraphen mit beschränkter zeitlicher Gültigkeit, modelliert werden. Verkehrsregeln und

andere Beschränkung werden in Form von prädikatenlogischen Bedingungen den Transitionen zugeordnet.

Anhand von verschiedenen Verkehrsszenarien wurden qualitative und quantitative Eigenschaften des Planungsraums wie z. B. die Größe in Abhängigkeit der Szenenkomplexität dargestellt. Eine Auswertung verschiedener Verkehrsszenen mit variierender Anzahl an dynamischen Objekten legt eine polynomielle Abhängigkeit zwischen Größe des Szenarios bzw. Anzahl der enthaltenen dynamischen Objekte sowie der Größe des Zustandsraums nahe. Dieser Zusammenhang wurde durch eine Betrachtung der entsprechenden Komplexitätsklassen theoretisch bestätigt.

Im semantischen Planungsraum können auch unscharfe oder unsichere Informationen aus der geometrischen Umweltmodellierung repräsentiert werden. Hierzu wurden verschiedene Ansätze vorgestellt und bewertet. Zwei dieser Verfahren kommen in der abschließenden Evaluation (Kapitel 7) zum Einsatz.

Der semantische Planungsraum bietet die Möglichkeit zur Planung von semantischen Manöverketten. Im folgenden Kapitel werden daher mögliche Planungsverfahren vorgestellt und in verschiedenen Verkehrsszenen qualitativ und quantitativ evaluiert.

5 Planung auf semantischen Zuständen

Bisher wurde gezeigt, wie ein semantischer Zustandsraum aufgebaut werden kann. Nun sollen auf diesem Manöverketten geplant werden [KHS+15]. Ziel ist es dabei, in der gegebenen Graphenstruktur einen gültigen Pfad von einem Startzustand zu einem Zielzustand oder mehreren Zielzuständen zu berechnen. Hierfür können gängige Algorithmen zur Graphensuche eingesetzt werden. Abbildung 5.1 zeigt einen möglichen Plan als Pfad in der Graph-Darstellung des semantischen Zustandsraums für das Bundesstraßenszenario aus Abbildung 4.7. Zustände des Plans ($s_1 - s_8$) sind grün hinterlegt.

Es ergeben sich mehrere Herausforderungen, die bei der Graphensuche algorithmisch berücksichtigt werden müssen. Ziel der Suche ist, einen optimalen Plan hinsichtlich einer gegebenen Bewertungsfunktion zu finden. Da diese Bewertung nicht ausschließlich auf der semantischen Darstellung erfolgen kann, ist es notwendig, zunächst mehrere semantische Pläne (ggf. auch alle möglichen Pläne) zu finden, und diese in einem zweiten Schritt zu bewerten. Somit können Optimierungskriterien wie Zeit- oder Energieoptimalität aber auch Bedingungen an Komfort erfüllt werden.

Da der semantische Zustandsraum eine Abstraktion von der Geometrie der Szene darstellt, kann die Fahrbarkeit einer geplanten Manöverkette unter dynamischen und physikalischen Randbedingungen nicht gewährleistet werden. Daraus ergibt sich die Notwendigkeit, einen erhaltenen semantischen Plan auf seine Fahrbarkeit hin zu validieren.

Des Weiteren werden an den Planungsprozess Anforderungen hinsichtlich seiner Laufzeit gestellt. Gerade im Zusammenhang mit dem Ziel, alle möglichen Pläne zu finden, ist hier eine Beschleunigung der Suche gegenüber einer erschöpfenden Breiten- oder Tiefensuche notwendig.

Als Lösungsansätze werden im Folgenden verschiedene Algorithmen zur Graphensuche (Kapitel 5.1) und entsprechende Spezialisierungen und Erweiterungen vorgestellt. Eine Beschleunigung des Suchvorgangs wird durch die Verwendung einer Heuristik-basierten Suche und die Einführung und Verwendung von Homotopieklassen (Kapitel 5.2) erreicht. Die Planungsverfahren werden in Kapitel 5.3 qualitativ und quantitativ evaluiert und Anwendungsbeispiele gezeigt. Auf die notwendige Validierung der geplanten Manöverketten wird in Kapitel 6 genauer eingegangen.

5 Planung auf semantischen Zuständen

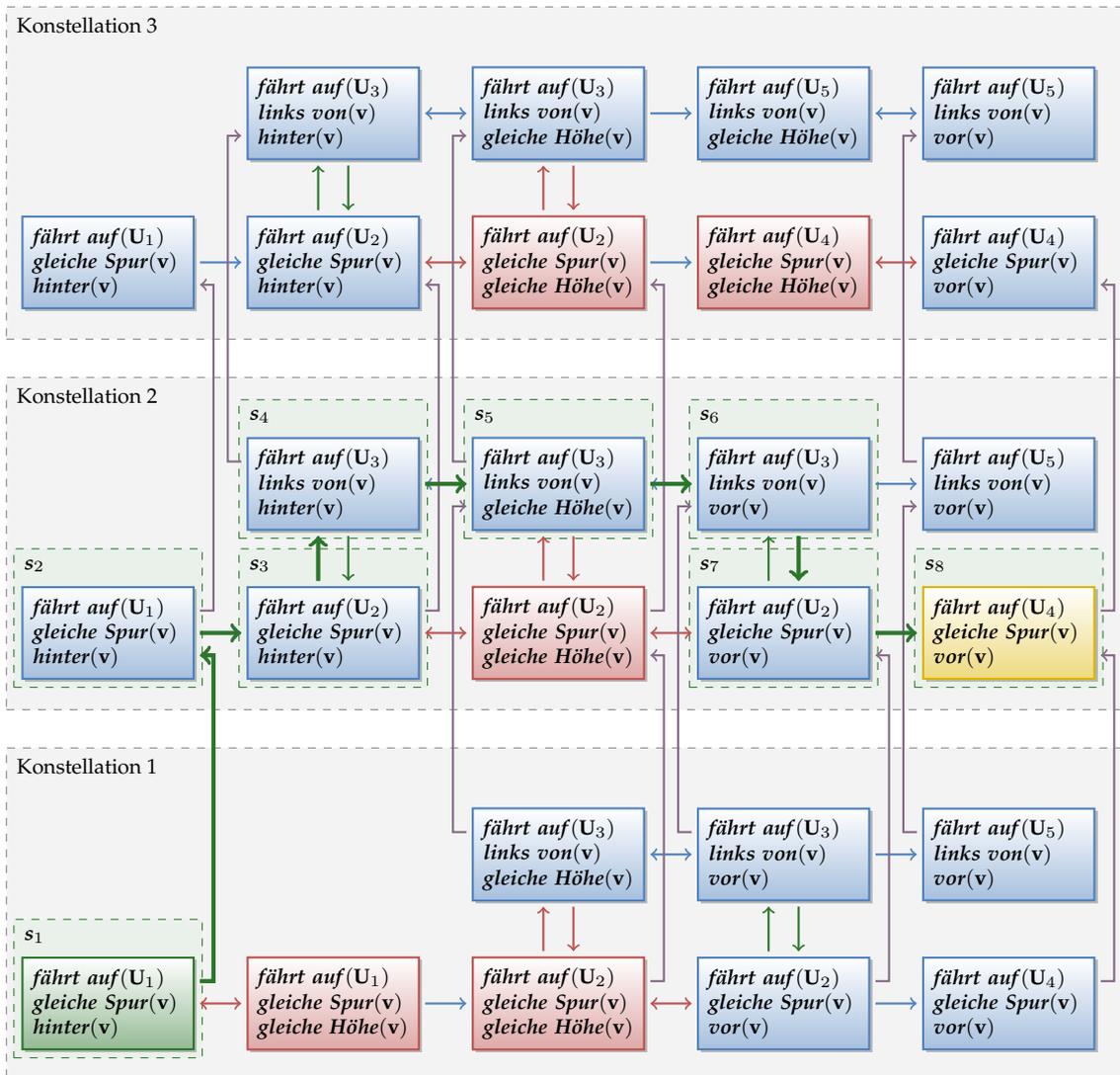


Abbildung 5.1: Möglicher Plan (grün) auf dem semantischen Zustandsraum des Bundesstraßenszenarios

Ähnlich veröffentlicht in [KHS+15] (Kohlhaas et al.)

5.1 Graphensuche

Da der semantische Zustandsraum eine Graphenstruktur besitzt, können bei der Manöverplanung Algorithmen zur Graphensuche verwendet werden. Einfache Verfahren wie Breitensuche (BFS) oder Tiefensuche (DFS) finden einen Pfad von einem Startknoten *start* zu einem Endknoten *ziel*.

Im Allgemeinen liegt bei der semantischen Planung jedoch eine Menge von Zielzuständen $Z \subset \mathbf{S}$ vor. Die Zielzustände können z. B. von einer übergeordneten Routenplanung her stammen und das Erreichen eines Fahrspursegments \mathbf{U} darstellen.

$$Z = \{s \in \mathbf{S} \mid \text{fährt auf}(\mathbf{U}) \in s\} \quad (5.1)$$

Beide Suchalgorithmen können auf die Verwendung einer Zielmenge angepasst werden, indem als Zielbedingung nicht mit einem sondern mit mehreren Zuständen verglichen wird.

Des Weiteren sollen alle Pfade vom Startzustand in die Zielmenge gefunden werden. Hier können die beiden Suchverfahren leicht modifiziert werden, indem nicht beim Finden eines Pfads abgebrochen wird, sondern alle teilweise explorierten Pfade bis zu einem Zielzustand oder einem Zustand ohne Folgezustand weiterverfolgt werden. Pfade, die bereits einen Zielzustand beinhalten, sollen jedoch nicht weiter expandiert werden, da hier die Zielmenge schon erreicht wurde. Auch können Knoten in verschiedenen Pfaden verwendet werden. Ein Knoten darf nur nicht mehr als einmal in einem Pfad vorkommen. Hierzu muss in der Implementierung statt einer Liste zu expandierender Knoten eine Liste zu expandierender Pfade mitgeführt werden. Die Algorithmen werden im Folgenden als vollständige Tiefensuche (engl.: exhaustive depth-first search, EDFS) bzw. vollständige Breitensuche (engl.: exhaustive breadth-first search, EBFS) bezeichnet.

Auf die Laufzeit der beiden Suchalgorithmen wirken sich diese Veränderungen negativ aus. Die Suche nach allen Pfaden sorgt dafür, dass immer die maximale Laufzeit $\mathcal{O}(|V| + |E|)$ (vergleiche Gleichung (2.8)) erreicht wird. Dürfen Knoten in verschiedenen Pfaden vorkommen, steigt die Laufzeit exponentiell in $|V|$ an. Quantitative Ergebnisse zum Laufzeitverhalten werden in Kapitel 6.4.3 vorgestellt. Eine Möglichkeit der Beschleunigung durch Reduktion der zu expandierenden Pfade mittels Homotopieklassen wird in 5.2 vorgestellt.

Eine weitere Möglichkeit die Suche zu beschleunigen ist die Verwendung einer Heuristik-basierten Suche. Dabei wird der nächste zu expandierende Pfad auf Basis einer Heuristik ausgewählt, so dass die erwarteten Kosten auf dem Pfad minimal sind. Die A*-Suche hat diese Eigenschaften und wurde in 2.9.1 vorgestellt. Er findet genau einen Plan von einem Start- zu einem Zielzustand. Dieser Plan ist bezüglich einer definierten Kostenfunktion optimal, solange gewisse Kriterien von der verwendeten Heuristik erfüllt werden.

Um den A*-Algorithmus anwenden zu können, muss also eine Kostenfunktion und eine zugehörige Heuristik aufgestellt werden. Für die Definition der bisherigen Kosten werden die Kosten für den Übergang von einem zum nächsten Zustand definiert als $c(s_i \rightarrow s_j)$. Als Kostenfunktion kann z. B. die Aufenthaltsdauer im Ausgangszustand, also die benötigte Zeit bis zur Transition herangezogen werden, um einen zeitoptimalen Plan zu erhalten. Als Heuristik $h(s)$ wird die Zeitdauer bis zum Erreichen der Zielkonstellation (Konstellation in der sich der Zielzustand befindet) verwendet. Es ist auch möglich eine dynamisch Validierung in die Kostenfunktion mit aufzunehmen und so lediglich dynamisch fahrbare Pläne zu erhalten. Dies wird in Kapitel 6.3.2 genauer beleuchtet.

In dieser Heuristik wird genau ein Zielzustand vorausgesetzt. Um die Suche auf mehrere Zielzustände zu erweitern, wird diese mehrfach, für jeden Zielzustand einmal, ausgeführt. Sind alle Zielzustände erreichbar, erhält man also genau so viele Pläne wie Zielzustände. Aus dieser Menge kann dann das Optimum durch Vergleich der jeweiligen Kosten bestimmt werden.

5.2 Homotopieklassen

Um ein umfassendes Bild der möglichen Manöver in einer Situation zu erlangen, ist der optimale Plan alleine nicht ausreichend. Stattdessen sollen alle relevanten Manövermöglichkeiten in entsprechenden Plänen abgebildet werden. Gleichzeitig soll die Anzahl der Pläne gering gehalten werden. Beispielsweise sollten in einer Fahrsituation wie dem Bundesstraßenszenario in Abbildung 4.7 nur zwei verschiedene Pläne existieren: Hinter dem Fahrzeug bleiben oder Überholen.

EDFS und EBFS führen jedoch je nach Szene zu deutlich mehr Plänen, die sich teilweise stark ähneln. So kann ein Überholmanöver z. B. etwas früher oder später eingeleitet werden. Die Zahl aller möglichen Pläne in einem semantischen Zustandsraum wird mit der Komplexität der Szene und damit des Raums größer. Da aus den Plänen in einem zweiten Schritt (lokal) optimale Pläne bestimmt werden sollen, ist es wünschenswert die Anzahl der Pläne frühzeitig zu reduzieren, ohne dabei die Optimalität zu vernachlässigen.

Ein gutes Maß für die Ähnlichkeit zweier Pläne ist die Zugehörigkeit zur selben Homotopieklasse. Für geometrische Trajektorien sind Homotopieklassen wie folgt definiert:

Definition 5.2.1: Homotopieklasse (geometrisch)

Zwei Trajektorien, die die gleichen Start- und Endpunkte verbinden, liegen genau dann in der gleichen Homotopieklasse, wenn die eine kontinuierlich in die andere überführt werden kann, ohne dabei ein Hindernis zu schneiden.[BLK12]

Diese Definition kann nicht direkt auf semantische Pläne angewendet werden, da sich diese nicht kontinuierlich modifizieren lassen. Die Idee hinter der kontinuierlichen Überführung ist, dass bei dieser keine Hindernisse übersprungen werden können (ohne diese zu schneiden). Für semantische Pläne muss also eine atomare Transformation definiert werden, die auch diese Eigenschaft besitzt. Außerdem muss sich durch sie jeder Plan in jeden anderen (mit gleichem Start- und Endpunkt) überführen lassen. Durch die Gitterstruktur des semantischen Zustandsraums lässt sich diese atomare Transformation durch eine Begrenzung der sich dabei verändernden Transitionen beschreiben.

Definition 5.2.2: Atomare Transformation

Ein atomare Transformation verändert einen semantischen Plan durch das Hinzufügen oder Streichen von in Summe maximal vier Transitionen.

Es ist zu beachten, dass diese vier Transitionen nicht beliebig über den semantischen Plan verteilt sein können, da der Plan sonst keine zusammenhängende Transitions-kette mehr wäre. Es gibt konkret nur vier verschiedene atomare Transformationen sowie ihre Umkehrungen:

- Hinzufügen von zwei aufeinanderfolgenden Transitionen mit gleichem Start- und Zielzustand
- Hinzufügen von vier aufeinanderfolgenden Transitionen mit gleichem Start- und Zielzustand
- Ersetzen einer Transition durch drei aufeinanderfolgende Transitionen
- Ersetzen von zwei aufeinanderfolgenden Transitionen durch zwei aufeinanderfolgende Transitionen

Damit lassen sich für semantische Pläne Homotopieklassen auf ähnliche Weise wie oben definieren:

Definition 5.2.3: Homotopieklasse (semantisch)

Zwei semantische Pläne, die die gleichen Start- und Endzustände verbinden, liegen genau dann in der gleichen Homotopieklasse, wenn der eine in den anderen durch die Anwendung von atomaren Transformationen überführt werden kann, ohne dabei einen Plan zu verwenden, der invalide Transitionen enthält.

Experimente zur A*S haben gezeigt, dass durch die Mehrfachausführung der Suche häufig alle Homotopieklassen in einer Szene durch Pläne repräsentiert werden. Es kann jedoch an einfachen Beispielen gezeigt werden, dass dies nicht immer der Fall ist. Beispielsweise wird auf einer dreispurigen Straße mit einem Fahrzeug auf der mittleren Spur nur ein Plan zum Überholen gefunden, obwohl (z. B. nach amerikanischen Verkehrsregeln) ein Überholen auf der linken sowie auf der rechten Seite möglich wäre.

5 Planung auf semantischen Zuständen

Daher sollen ausgehend von der Menge aller möglichen Pläne, wie sie z. B. von EDFs und EBFS gefunden werden, Homotopieklassen identifiziert und jeweils durch einen Plan repräsentiert werden. Dabei werden folgende Ziele verfolgt:

- Die Anzahl der Pläne soll reduziert werden, indem aus jeder Homotopieklasse nur ein Repräsentant bereitgestellt wird.
- Der jeweils optimale Plan einer Homotopieklasse soll erhalten bleiben. Da die Optimalitätskriterien während der semantischen Planung im allgemeinen nicht bekannt sind, sollen die Pläne verallgemeinert werden, um die Optimalität zu erhalten.
- Die Laufzeit des Planungsverfahrens soll verringert werden.

5.2.1 Homotopiebausteine

Um festzustellen, ob zwei Pläne in der gleichen Homotopieklasse liegen, werden sogenannte Homotopiebausteine definiert. Ein Homotopiebaustein beschreibt eine atomare Transformation zwischen zwei Plänen sowie Regeln zur Fusion dieser Pläne. Dabei werden mehrere Zustände zu einem Metazustand zusammengefasst. Der resultierende Plan gehört dabei der gleichen Homotopieklasse an.

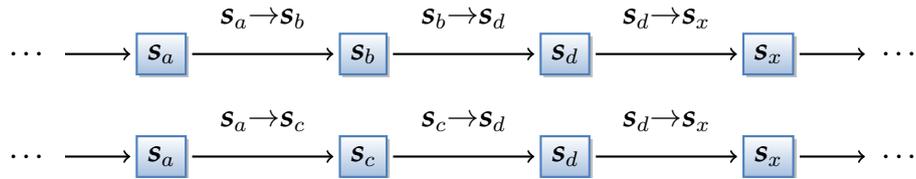
Die Zusammenführung von Plänen kann bei der EBFS schon zur Laufzeit angewendet werden, da hier frühzeitig Teilpläne vorliegen, die auf ihre Zugehörigkeit zu Homotopieklassen untersucht werden können. Dadurch wird der bei der Suche aufgespannte Baum reduziert und das Verfahren beschleunigt.

Abbildung 5.2 zeigt die Anwendung eines Homotopiebausteins auf zwei ähnliche Pläne (5.2a). Die beiden Pläne unterscheiden sich nur in den Knoten s_b und s_c . Fasst man beide Pläne zu einem Graphen zusammen, entsteht in diesem Beispiel eine rautenartige Struktur, die den Homotopiebaustein charakterisiert (5.2b). Die beiden Pläne können zusammengefasst werden. Dabei werden zwei Paare von Zuständen zu jeweils einem Metazustand zusammengefasst (5.2c).

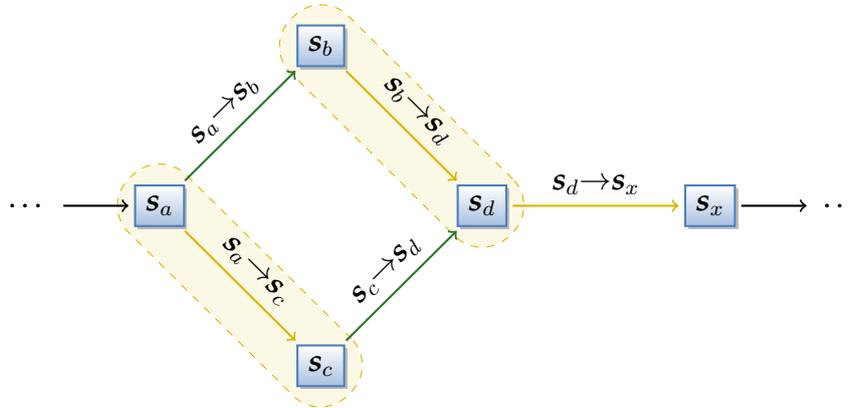
Im Folgenden werden verschiedenen Bausteine vorgestellt:

Rautenschema

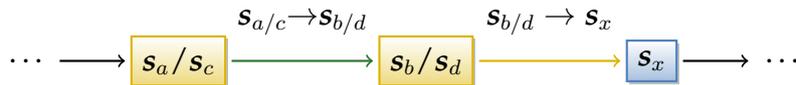
Wenn sich zwei Pläne nur durch jeweils einen Zustand unterscheiden, entsteht ein Rautenschema, wie in Abbildung 5.2b dargestellt, als Charakterisierung des Homotopiebausteins. In einer solchen Raute können jeweils zwei der vier Zustände zusammengefasst werden. Es muss jedoch noch bestimmt werden, ob und welche der Zustände zusammengefasst werden können. Hierbei spielen der Typ der Transitionen (vergleiche Definition 4.2.5) zwischen den Zuständen eine wichtige Rolle.



(a) Zwei Pläne die sich nur in den Zuständen s_b und s_c unterscheiden



(b) Rautenschema für die beiden Pläne



(c) Resultierender Plan mit kombinierten Metazuständen $s_{a/c}$ und $s_{b/d}$

Abbildung 5.2: Anwendung eines Homotopiebausteins auf zwei Pläne, die sich nur in einem Zustand unterscheiden

Ähnlich veröffentlicht in [KHS+15] (Kohlhaas, Hammann et al.)

Betrachtet man die beiden Pläne aus Abbildung 5.2, dann können diese nur dann zusammengefasst werden wenn folgende Bedingungen erfüllt sind:

$$\begin{aligned} \text{Type}(s_a \rightarrow s_b) &= \text{Type}(s_c \rightarrow s_d) \\ \text{Type}(s_a \rightarrow s_c) &= \text{Type}(s_b \rightarrow s_d) \end{aligned}$$

Abhängig vom konkreten Typ lassen sich drei verschiedene Arten von Homotopiebaustein unterscheiden:

Temporaler Homotopiebaustein

Ein temporaler Homotopiebaustein liegt vor, wenn

$$\text{Type}(s_a \rightarrow s_c) = \text{Type}(s_b \rightarrow s_d) = \text{Type}(s_d \rightarrow s_x) = \text{temporal}.$$

Die anderen beiden Transitionen ($s_a \rightarrow s_c$, $s_b \rightarrow s_d$) müssen dann von Typ lateral oder longitudinal sein. Wie auch bei den folgenden Homotopiebausteinen ist der symmetrische Fall mit

$$\text{Type}(s_a \rightarrow s_b) = \text{Type}(s_c \rightarrow s_d) = \text{Type}(s_d \rightarrow s_x) = \text{temporal}$$

ebenfalls möglich.

5 Planung auf semantischen Zuständen

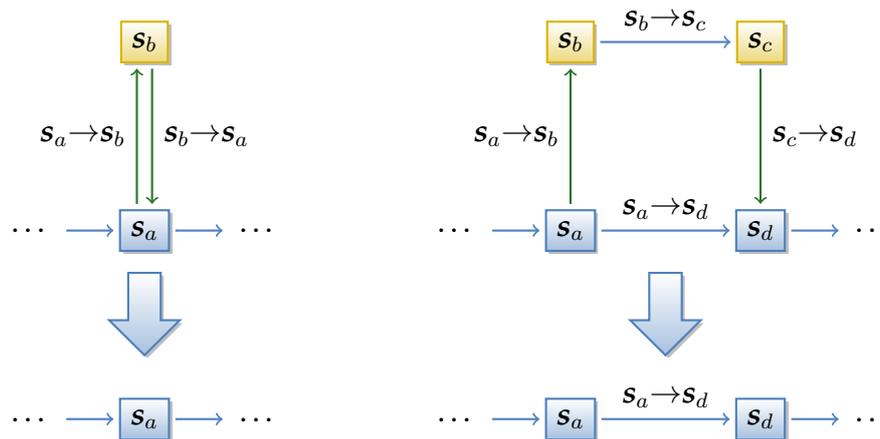


Abbildung 5.3: Homotopiebausteine zur Darstellung eines unnötigen Spurwechsels. Der Plan mit Spurwechsel kann jeweils verworfen werden.

In Anlehnung an [Ham15]

Longitudinaler Homotopiebaustein

Ein longitudinaler Homotopiebaustein liegt vor, wenn

$$\text{Type}(s_a \rightarrow s_b) = \text{Type}(s_c \rightarrow s_d) = \text{Type}(s_d \rightarrow s_x) = \text{longitudinal.}$$

Die anderen beiden Transitionen müssen dann von Typ lateral oder temporal sein.

Lateraler Homotopiebaustein

Ein lateraler Homotopiebaustein liegt vor, wenn

$$\text{Type}(s_a \rightarrow s_b) = \text{Type}(s_c \rightarrow s_d) = \text{Type}(s_d \rightarrow s_x) = \text{lateral.}$$

Die anderen beiden Transitionen müssen dann von Typ longitudinal oder temporal sein.

Doppelter Spurwechsel ohne Überholen

Weitere Homotopiebausteine beschreiben Paare von Spurwechseln, die wieder zurück auf die Ausgangsspur führen und nicht Teil eines Überholmanövers sind. Je nach Anwendung und anschließender geometrischer Planung kann es sinnvoll sein, diese Spurwechsel aus den Plänen zu streichen oder auch beide Spuren für die weitere Planung freizugeben (z. B. um Raum für Ausweichmanöver in kritischen Situationen freizugeben). Im Folgenden wird davon ausgegangen, dass diese Spurwechsel unnötig sind und Pläne, die solche enthalten, zu Gunsten von Plänen derselben Homotopieklasse, die diesen Spurwechsel nicht enthalten, verworfen werden können. Abbildung 5.3 zeigt zwei Homotopiebausteine zur Erkennung dieses Sachverhalts. Im linken Beispiel erfolgt der Spurwechsel direkt zurück in den Ausgangszustand. Dieser Spurwechsel würde genau so wie auch andere Zyklen mit mehr Transitionen auch von einer Zyklenerkennung während

der Planung vermieden werden. Dies ist beim rechten Beispiel nicht der Fall. Durch eine weitere longitudinale oder temporale Transition wird der Zyklus aufgelöst.

$$\begin{aligned} \text{Type}(s_a \rightarrow s_b) &= \text{Type}(s_b \rightarrow s_a) = \text{Type}(s_c \rightarrow s_d) = \text{lateral} \\ \text{Type}(s_b \rightarrow s_c) &= \text{Type}(s_a \rightarrow s_d) \in \{\text{longitudinal, temporal}\} \end{aligned}$$

Dennoch liegt ein vermeidbarer Spurwechsel vor, der durch Anwendung dieses Homotopiebausteins während der Planung erkannt werden kann. Gleiches kann auch für unnötige longitudinale Positionswechsel mit anderen Fahrzeugen in Kombination mit Spurwechseln modelliert werden.

Zusammenfassung von Zuständen

Durch die Zusammenfassung von Zuständen entstehen neue Metazustände, die jeweils eine Menge von Zuständen beinhalten. Gerade an den Rändern des Planungsraums kann es passieren, dass zwei Teilpläne (P_A und P_B) prinzipiell gleich sind, jedoch in einem Plan (P_A) zwei oder mehrere Zustände zu einem Metazustand zusammengefasst sind, in dem anderen (P_B) jedoch nicht. Diese Pläne können dennoch zusammengefasst werden, wenn ein entsprechender Homotopiebaustein vorliegt. Dazu werden die Zustände in P_B zunächst entsprechend der Metazustände aus P_A zusammengefasst. Danach kann dann der Homotopiebaustein angewendet und die Pläne vereinigt werden.

5.2.2 Anwendung der Homotopiebausteine

Die Homotopiebausteine und Zusammenfassung von Zuständen kann während der Planungsphase der EBFS eingesetzt werden. In diesem Algorithmus wird bei der Exploration von Teilplänen jeweils ein weiterer Zustand an einen Plan angefügt. Hier kann die Homotopiebetrachtung ansetzen. Der Algorithmus 2 skizziert den Ablauf der Zusammenfassung von Teilplänen aus der gleichen Homotopiekategorie.

Ausgangspunkt ist ein Teilplan p sowie ein semantischer Zustand s , der an diesen angefügt werden soll (Zeile 1). Da bei der Breitensuche die anderen Teilpläne \mathbb{P} gleich weit expandiert sind (ggf. einen Zustand weniger weit) und der hier gezeigte Algorithmus schon in den vorherigen Expansionsschritten angewendet wurde, ist es ausreichend, Pläne mit gleichem Endzustand zu vergleichen (Zeile 3). Zuerst wird s an p angefügt (Zeile 2). Für alle anderen Pläne p_i wird überprüft, ob sie zusammen mit p am Ende des Plans einen Homotopiebaustein bilden und ansonsten identisch sind (Zeile 5). Falls ja, können die beiden Pläne entsprechend des Blocks zusammengeführt werden (Zeile 6) und der entsprechende Plan aus der Menge der zu expandierenden Pläne gelöscht werden (Zeile 7). Der zusammengeführte Plan p_m kann dann als Ergebnis der Breitensuche übergeben werden.

5 Planung auf semantischen Zuständen

Sind die Anfänge der Pläne unterschiedlich, wird versucht sie durch Zusammenfassen von Zuständen, die im jeweils anderen Plan schon zusammengefasst waren, anzugleichen (Zeile 9). Danach wird die Homotopiebausteinsuche auf den angeglichenen Plänen wiederholt (Zeile 10ff). Ist auch diese nicht erfolgreich, wird lediglich der expandierte Plan p zur Weiterverarbeitung zurückgegeben (Zeile 14).

Algorithmus 2 Anwendung der Homotopiebausteine

```
1: procedure EXTENDPATH( $p, s, \mathbb{P}$ )
2:   ADDSTATETOPLAN( $p, s$ )
3:    $\mathbb{P}_s \leftarrow$  SUBSETENDSWITHSTATE( $\mathbb{P}, s$ )
4:   for each  $p_i$  in  $\mathbb{P}_s$  do
5:     if HASHHOMOTOPYBUILDINGBLOCK( $p, p_i$ ) then
6:        $p_m \leftarrow$  MERGEPLANS( $p, p_i$ )
7:       REMOVEPLAN( $\mathbb{P}, p_i$ )
8:       return  $p_m$ 
9:      $p_t, p_it \leftarrow$  MERGESTATES( $p, p_i$ )
10:    if HASHHOMOTOPYBUILDINGBLOCK( $p_t, p_it$ ) then
11:       $p_m \leftarrow$  MERGEPLANS( $p_t, p_it$ )
12:      REMOVEPLAN( $\mathbb{P}, p_i$ )
13:      return  $p_m$ 
14:   return  $p$ 
```

Die hier beschriebene Variante wird im Folgenden als vollständige Breitensuche mit Berücksichtigung von Homotopieklassen (EBFSH) bezeichnet. Es handelt sich dabei um einen gierigen Algorithmus, der nur ein lokales Optimum erreicht. Die Reihenfolge der Expansion der Pläne entscheidet hier über mögliche Kombinationen der Pläne. Es kann also nicht gewährleistet werden, dass alle Pläne einer Homotopieklasse zusammengefügt werden. Der Algorithmus findet hier dennoch Anwendung, da er sich gut in die Struktur der EBFS integrieren lässt und die angestrebte Reduzierung der betrachteten Pläne bei gleichzeitiger Erhaltung des optimalen Plans erreicht. Eine Laufzeitverbesserung aufgrund der geringeren Anzahl von Plänen wird später in Kapitel 6.4.3 experimentell nachgewiesen.

5.3 Evaluation

Um die Planungsverfahren qualitativ und quantitativ zu evaluieren, wurden sie auf die aus Kapitel 4.4 bekannten Szenarien und Zustandsräume angewendet. Des Weiteren werden in Kapitel 5.3.2 Beispielanwendungen für die semantische Planung vorgestellt.

5.3.1 Planungsverfahren

Im Folgenden werden folgende Planungsansätze verglichen:

- vollständige Breitensuche (EBFS)
- vollständige Tiefensuche (EDFS)
- A*-Suche (A*S)
- vollständige Breitensuche mit Berücksichtigung von Homotopieklassen (EBFSH)

Abbildung 5.4 und Abbildung 5.5 zeigen für das Bundesstraßenszenario die beiden resultierenden Pläne mittels A*S. Die Pläne repräsentieren die Möglichkeiten das Fahrzeug zu überholen oder hinter diesem zu verbleiben. Zustände, die zur geplanten Manöverkette gehören, sind grün umrandet und entsprechend ihrer Reihenfolge im Plan mit sn gekennzeichnet. Der Startzustand ist grün gefüllt dargestellt, die Zielzustände sind gelb gefüllt. Wie aus der Darstellung des Zustandsraums bekannt, sind Zustände, in denen es zu einer Kollision mit einem anderen Fahrzeug kommen könnte, rot gefärbt.

Als Ziel des Ego-Fahrzeugs wurde das Fahrspursegment U_4 ausgewählt. Um die Menge der Zielzustände nicht unnötig zu vergrößern, wurden lediglich die Zustände als Zielzustände ausgewählt, die eine direkte Transition von einem anderen Fahrspursegment in das Zielsegment aufweisen. So ergibt sich pro Konstellation genau ein Zielzustand. Es werden genau zwei Pläne (Abbildung 5.4 und 5.5) gefunden. Dies entspricht der Menge der erreichbaren Zielzustände. Die Zielzustände in Konstellation 1, 2 und 4 sind aufgrund der Position des vorausfahrenden Fahrzeugs nicht erreichbar.

In Abbildung 5.4 ist das Überholmanöver dargestellt. Der Plan entspricht dem in Abbildung 5.1 gezeigten Plan. Allerdings beginnt der Zustandsraum dort erst bei Konstellation 2 (also zeitlich ein Ereignis später) und es sind nur die relevanten Konstellationen dargestellt.

Die Ergebnisse der Breiten- bzw. Tiefensuche (EBFS und EDFs) sind aufgrund der Eigenschaften der Planungsverfahren identisch. Exemplarisch sind in Abbildung 5.6 die resultierenden Pläne der EBFS dargestellt. Aus Gründen der Lesbarkeit wurde auf eine Zustandsbeschriftung verzichtet. Plan 5.6a zeigt das Überholmanöver. Die Pläne 5.6b bis 5.6h verbleiben hinter dem Fahrzeug. Dabei weisen die Pläne 5.6e bis 5.6h jeweils einen unnötigen Spurwechsel hinter dem Fahrzeug auf.

Auch für das Kreuzungsszenario können Pläne erzeugt werden. Abbildung 5.7 zeigt exemplarisch einen Plan der mittels EBFS erstellt wurde. Es ist zu erkennen, wie **ego** zunächst vor der Kreuzung wartet (s_1 bis s_8) und das andere Fahrzeug passieren lässt, dann diesem folgt (s_9 bis s_{13}) und in der letzten Konstellation nachdem das andere Fahrzeug an der zweiten Kreuzung abgebogen ist, freie Fahrt in den Zielzustand (s_{14}) hat. Auffällig ist, dass **ego** beim Übergang zwischen

5 Planung auf semantischen Zuständen

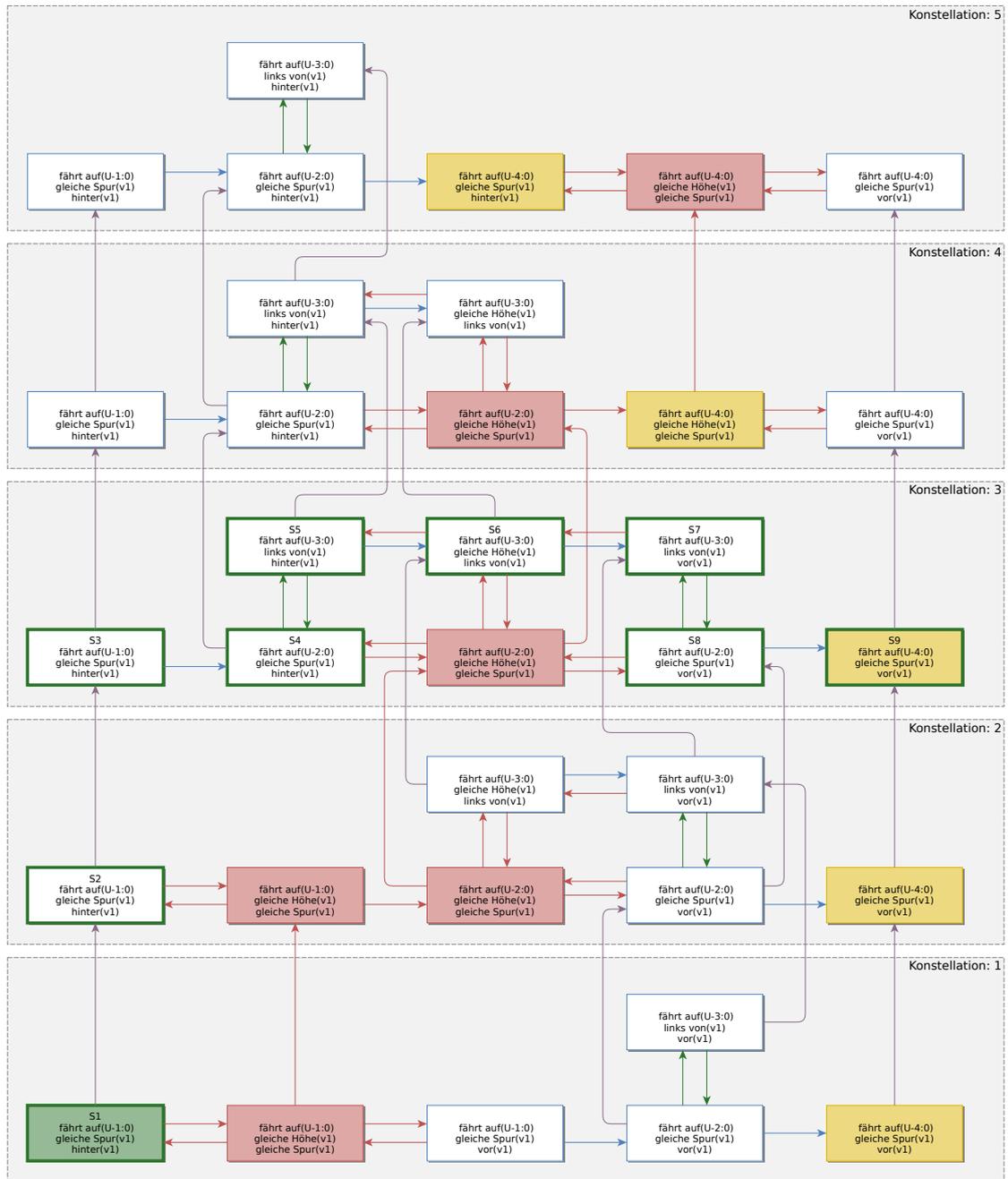


Abbildung 5.4: Resultierender Plan für das Bundesstraßenszenario mit A*S. Das Fahrzeug v_1 wird in Konstellation 3 überholt.

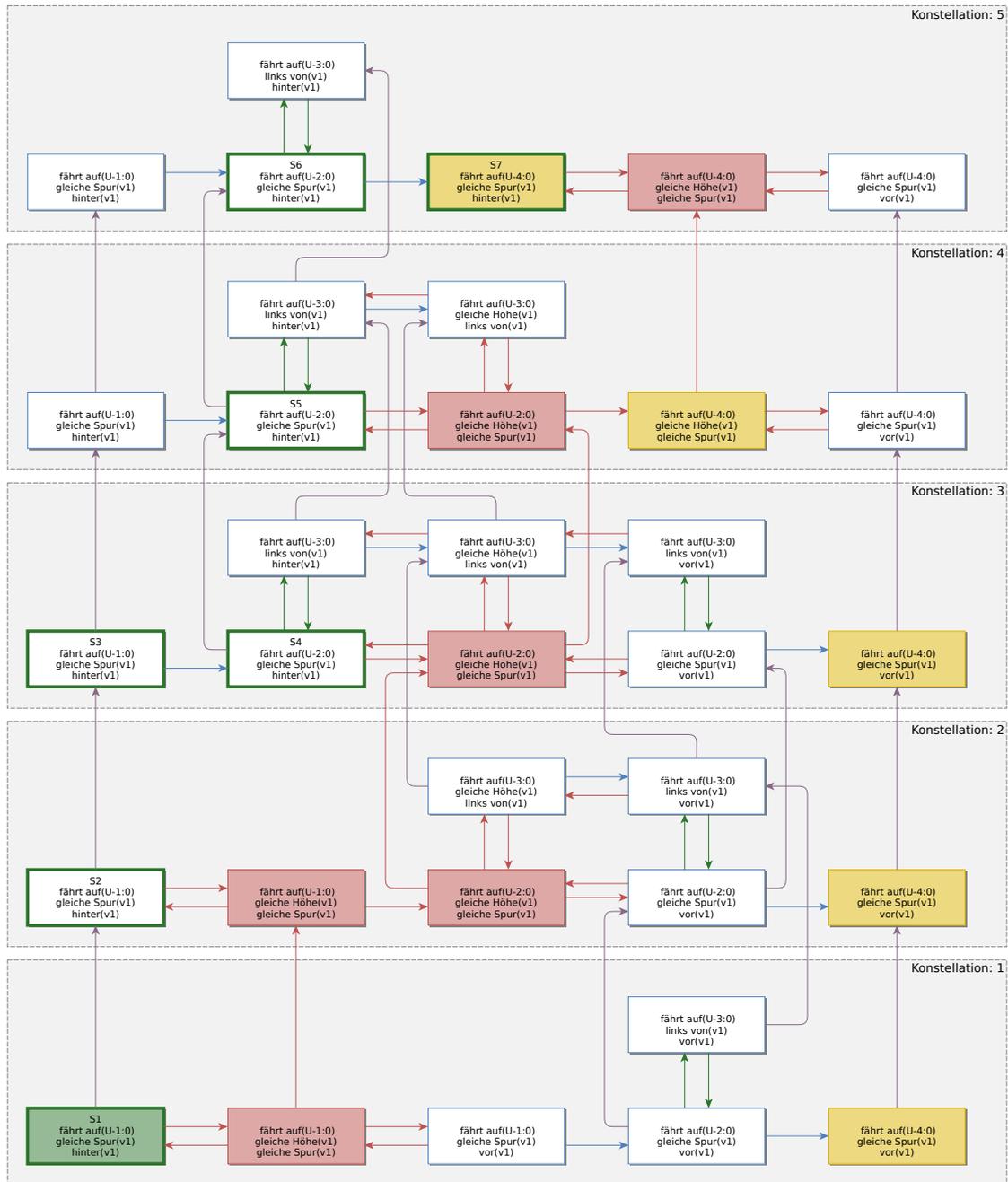


Abbildung 5.5: Resultierender Plan für das Bundesstraßenszenario mit A*S. Das Ego-Fahrzeug bleibt hinter dem Fahrzeug v₁.

5 Planung auf semantischen Zuständen



Abbildung 5.6: Resultierende Pläne für das Bundesstraßenszenario mit EBFS

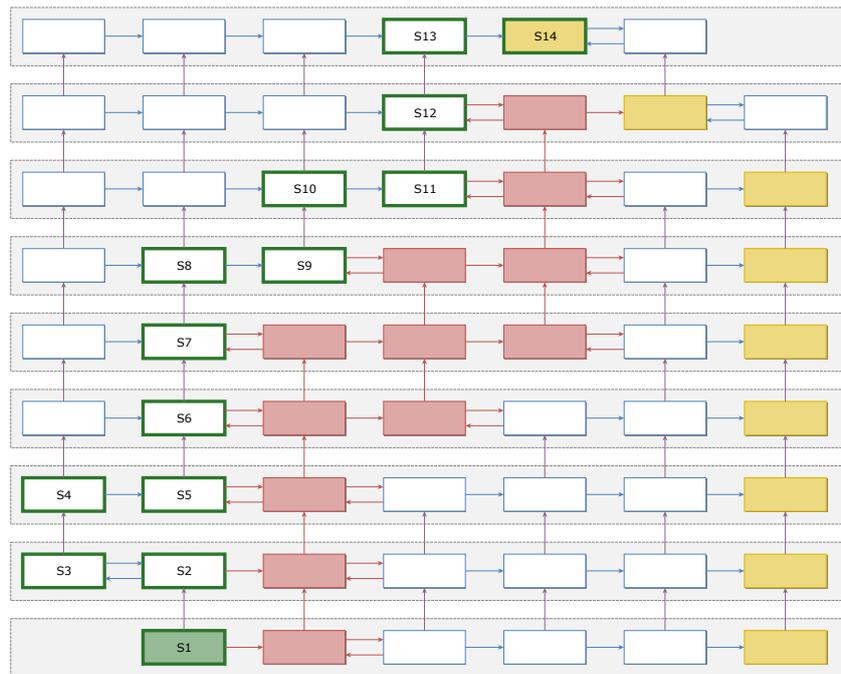


Abbildung 5.7: Resultierender Plan für das Kreuzungsszenario mit EBFS. Das Ego-Fahrzeug gibt dem anderen Fahrzeug Vorfahrt und bleibt hinter diesem.

s_2 und s_3 in der graphischen Darstellung vermeintlich einen Zustandswechsel entgegen der Fahrtrichtung macht. Die Transition ($s_2 \rightarrow s_3$) ist zwar entgegen der Fahrtrichtung gerichtet, dies bedeutet jedoch nicht, dass **ego** hier real rückwärts fahren muss. Die Transition beschreibt lediglich den Übergang der Relation von *gleiche Höhe*(v_1) zu *hinter*(v_1). Da sich v_1 auf die Kreuzung zu bewegt, reicht es aus, dass **ego** eine geringere Geschwindigkeit annimmt, um sich gegenüber v_1 zurückfallen zu lassen, bevor die Kreuzung erreicht wird.

Homotopiebausteine

Wie in Kapitel 5.2.1 beschreiben, können Pläne der selben Homotopieklasse während der Suche zusammengefasst werden. Im vorherigen Beispiel des Bundesstraßenszenarios (siehe Abbildung 5.6) gehören z. B. 7 der 8 von der EBFS gefundenen Pläne zu einer Homotopieklasse. Sie beschreiben alle das Verbleiben von **ego** hinter dem anderen Fahrzeug. Wird auf dieses Szenario die EBFSH angewendet, so reduziert sich die Anzahl der Pläne auf die zu erwartenden zwei Pläne: Hinter dem Fahrzeug bleiben und das Fahrzeug überholen. Die Pläne sind in Abbildung 5.8 dargestellt.

Abbildung 5.8a zeigt dabei den Überholvorgang. Zu Gunsten der Übersichtlichkeit wurde auf die Bezeichnung der Zustände verzichtet. Zustände, die zum Plan gehören, sind wie zuvor grün umrandet und entsprechend der Reihenfolge im

5 Planung auf semantischen Zuständen

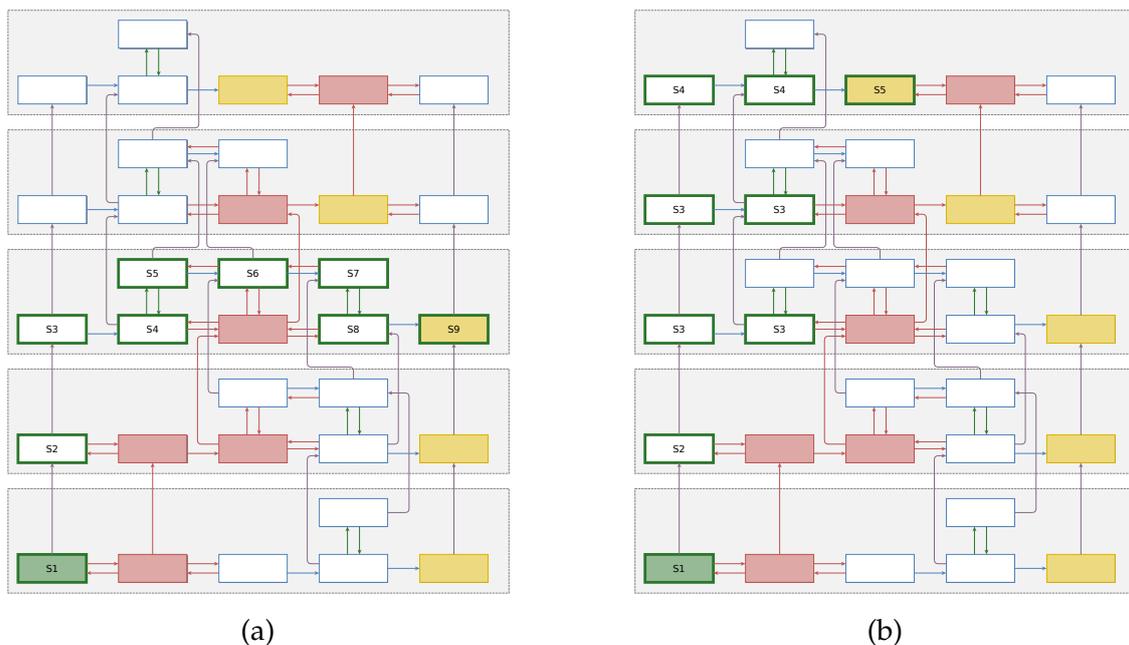


Abbildung 5.8: Resultierende Pläne für das Bundesstraßenszenario mit EBFSH

Plan nummeriert. Der Plan besteht also aus 9 Zuständen und endet in der dritten Konstellation. Dieser Plan ist der einzige Repräsentant seiner Homotopieklasse. Es gab somit keine Zusammenfassung von Plänen.

In Abbildung 5.8b ist der Plan dargestellt, bei dem **ego** hinter v_1 bleibt. Hier wurden mehrere Pläne zu einem zusammengefasst und somit auch mehrere semantische Zustände zu einem Metazustand zusammengefasst. Zusammengefasste Zustände sind mit der selben Nummer bezeichnet. Die Zustände s_3 und s_4 sind durch zweifache Anwendung des Rautenschemas entstanden. Dabei werden zunächst die Pläne in Abbildung 5.6b und 5.6c durch Anwendung eines temporalen Homotopiebausteins zusammengefasst. Der resultierende Plan kann dann mit dem Plan aus Abbildung 5.6d durch Anwendung eines longitudinalen Homotopiebausteins vereinigt werden. Hier ist gut zu erkennen, dass die semantische Information erhalten bleibt, dass zunächst hinter dem Fahrzeug gefahren werden muss, bis dieses in der fünften Konstellation das Ziel-Fahrspursegment freigibt. Dies wird durch die Transition von s_3 zu s_4 dargestellt. Des Weiteren ist erkennbar, dass unnötige Spurwechsel auf die zweite Spur erfolgreich erkannt wurden und somit nicht mehr in den Plänen vorkommen. Die EBFSH liefert in diesem Beispiel genau die beiden gewünschten Pläne und räumt dabei durch die Zusammenfassung von Zuständen nachgelagerten geometrischen Planungs- und Ausführungskomponenten maximale Flexibilität ein.

In einigen Szenarien führt die Anwendung der Homotopiebausteine jedoch nicht zur Erkennung und Zusammenführung aller Homotopieklassen. Es werden also mehr repräsentative Pläne gefunden, als es notwendig wäre. Dies wird am Kreuzungsszenario (vergleiche Abbildung 4.27) veranschaulicht.

5 Planung auf semantischen Zuständen

Straßensegmente an das Szenario wird dessen Länge von ursprünglich drei Segmenten auf bis zu 9 Segmente erweitert. Hierdurch nehmen nicht nur die Anzahl der Zustände in jeder Konstellation zu, sondern durch die Bewegung des roten Fahrzeugs auch die Anzahl der Konstellationen. Zielsegment für die Planung ist jeweils das am weitesten rechts liegende Straßensegment. Da nur ein Zielstraßensegment existiert, entspricht die Anzahl der Zielzustände der Anzahl der Konstellationen.

In Tabelle 5.1 sind zu den Szenariolängen jeweils die Anzahl resultierender Zustände im Planungsraum sowie die der Zielzustände aufgetragen. Die Anzahl der Zustände wächst, wie in Kapitel 4.4.5 hergeleitet, quadratisch mit der Anzahl der statischen Szenenobjekte und somit in diesem Fall mit der Szenariolänge an.

Szenario- länge	Anzahl Zielzustände	Anzahl Zustände	Anzahl Pläne		
			EBFS/EDFS	EBFSH	A*S
3	5	34	8	2	2
4	7	53	59	8	4
5	9	76	443	15	6
6	11	103	3404	39	8
7	13	134	–	85	10
8	15	169	–	203	12
9	17	208	–	429	14

Tabelle 5.1: Auswertung der Zustandsraumgröße im Bundesstraßenszenario

Zusätzlich ist für die betrachteten Planungsverfahren die Anzahl der resultierenden Pläne angegeben. Für die beiden Planungsverfahren EBFS und EDFFS ergeben sich logischerweise identische Zahlen, da jeweils alle möglichen Pläne gefunden werden. Dementsprechend stark ist auch der Anstieg bei wachsender Szenariolänge. Ab einer Länge von sieben Segmenten konnte die Planung nicht in sinnvoller Rechenzeit abgeschlossen werden.

Wie erwartet zeigt die Ausnutzung von Homotopieklassen eine starke Reduktion der Pläne. Jedoch zeigt sich auch hier, wie schon zuvor für das Kreuzungsszenario dargestellt, dass mehr Pläne entstehen, als im Szenario zu erwarten sind. Die wenigsten Pläne werden durch A*S erzeugt. Hier steigt die Anzahl der Pläne mit der Anzahl der Zielzustände und ist auch durch diese beschränkt, da für jeden Zielzustand genau ein Plan gesucht und in den meisten Fällen auch gefunden wird. In einigen Anwendungen kann es jedoch sinnvoll sein, mehr oder auch allgemeinere Pläne zu erzeugen, so dass dann eines der erschöpfenden Suchverfahren zum Einsatz kommen muss.

5.3.2 Anwendungen

Um die Anwendbarkeit der semantischen Planung darzustellen, wurden zwei Beispielanwendungen implementiert. Diese werden im Folgenden vorgestellt.

Überholassistent

Bei dieser Anwendung handelt es sich um ein hinweisgebendes, passives Assistenzsystem auf dem SAE-Level 0 (vergleiche Kapitel 2.1). Es soll dem Fahrer dabei helfen, zu entscheiden, ob ein Überholmanöver möglich ist oder nicht. Dabei wird die aktuell gewählte Route (z. B. aus einem Navigationssystem) sowie der vorausfahrende Verkehr berücksichtigt.

Mittels einer Monokamera wird der vorausfahrende Verkehr beobachtet und Fahrzeuge detektiert. Hierbei kommt das in Kapitel 2.5.1 beschriebene Verfahren zur Fahrzeugdetektion zum Einsatz. Es basiert auf einer Perzeption mittels Haar-Wavelets und Ada-Boosting, einer Klassifikation mittels HOG und SVM sowie einem lokalen Kalman-Filter für die temporale Fusion. Die Lage und Geschwindigkeit des Ego-Fahrzeugs wird mittels eines hochgenauen GPS mit IMU bestimmt. Durch die Verwendung einer hochgenauen Karte kann die Verortung auf dieser rein geometrisch erfolgen.

Auf Basis dieser geometrischen Umfelddarstellung wird dann ein semantischer Zustandsraum aufgebaut. Gibt es einen Plan zum gewählten Zielsegment, der mit einem Spurwechsel beginnt, so wird dieser dem Fahrer vorgeschlagen. Die Signalisierung erfolgt mittels Augmentierung mit grünen Pfeilen im Kamerabild.

Die Anwendung wurde auf dem Versuchsfahrzeug CoCar (vergleiche Kapitel 2.2.2) integriert und auf verschiedenen Straßenabschnitten getestet. Im Folgenden werden Ergebnisse von einer Fahrt auf der B35 nahe Bruchsal gezeigt. Der befahrene Straßenabschnitt ist in Abbildung 5.10 dargestellt. Das Ego-Fahrzeug fährt von der A5 kommend Richtung Osten auf die B35 und dann weiter auf die L556 Richtung Süden nach Karlsruhe.

Eine schematische Darstellung der Szene ist in Abbildung 5.11 dargestellt. Dabei bilden U_4 und U_7 die Abbiegespur auf die L556. Da die Fahrt auf L556 weitergeführt werden soll, wird U_7 hier als Zielsegment verwendet.

Die Ausgabe des Verfahrens sowie die erzeugten Zustandsräume sind in Abbildung 5.12 dargestellt. Der obere Bereich der einzelnen Grafiken zeigt dabei das Videobild der Fahrzeugkamera. Vom Fahrzeugdetektor erkannte Fahrzeuge sind mit einem blauen Rechteck umrahmt. Im unteren Bereich ist der aktuell erzeugte Zustandsraum dargestellt. Aus Gründen der Übersichtlichkeit ist dieser auf maximal drei Konstellationen beschränkt. Der aktuelle Zustand des Ego-Fahrzeugs ist in grün, durch andere Fahrzeuge blockierte Zustände in rot dargestellt.

5 Planung auf semantischen Zuständen

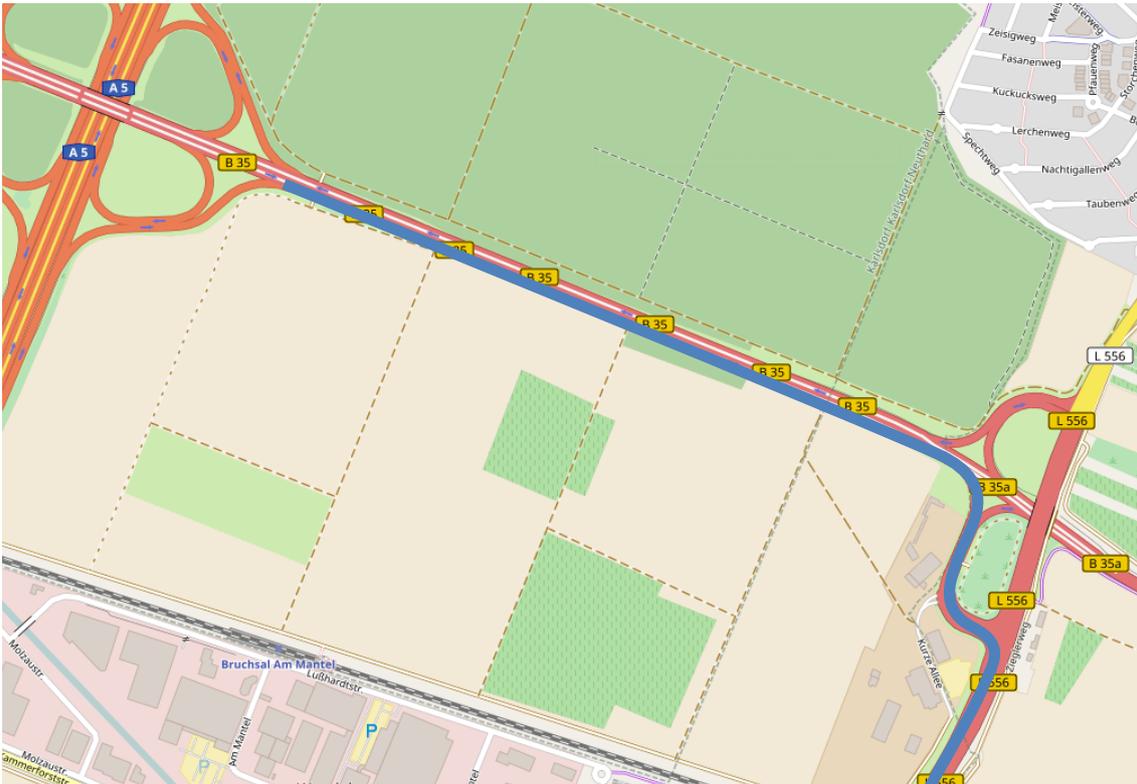


Abbildung 5.10: Kartenausschnitt bei Bruchsal. Die befahrene Strecke ist blau markiert.

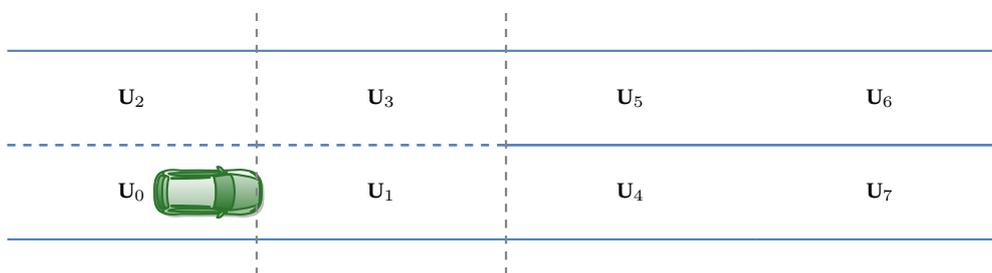
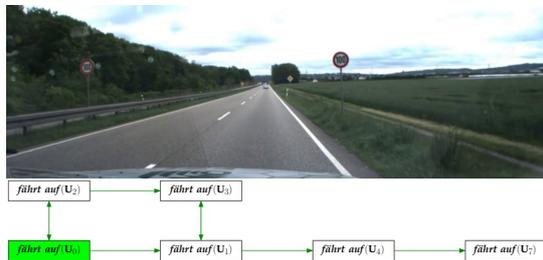
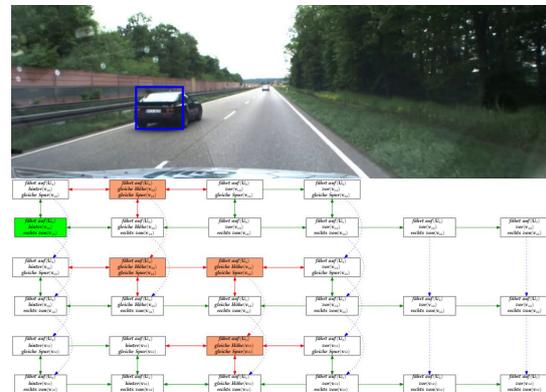


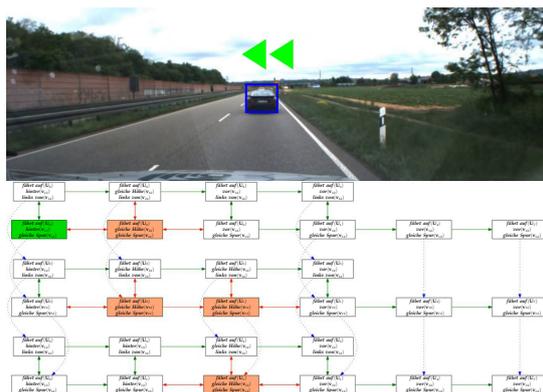
Abbildung 5.11: Schematische Darstellung des Streckenabschnitts



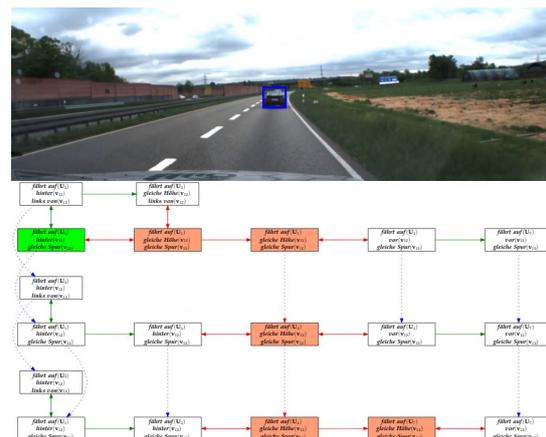
(a) Statische Szene ohne Fahrzeuge



(b) Das Fahrzeug auf der linken Spur kann nicht überholt werden



(c) Nachdem das Fahrzeug nach rechts gewechselt hat, kann es überholt werden. Grüne Pfeile signalisieren dies dem Fahrer.



(d) Das Fahrzeug kann nicht mehr überholt werden, da ein Einscheren nicht mehr vor der Abzweigung möglich wäre

Abbildung 5.12: Spurwechselassistent auf realen Fahrzeugdaten. Kamerabild mit Fahrzeugdetektion (oben) und semantischer Zustandsraum (unten).

5 Planung auf semantischen Zuständen

Wird kein Fahrzeug erkannt, enthalten die Zustände lediglich Relationen zu den statischen Umgebungsobjekten. Dies ist in Abbildung 5.12a dargestellt. Das entfernte, vorausfahrende Fahrzeug befindet sich außerhalb der Detektionsreichweite des verwendeten Fahrzeugdetektors. Wird ein Fahrzeug detektiert, so wird dieses in den Zustandsraum mit aufgenommen. In Abbildung 5.12b wird ein Fahrzeug erkannt. Dieses darf jedoch nicht auf der rechten Seite überholt werden. Erst nach einem Spurwechsel des vorausfahrenden Fahrzeugs (siehe Abbildung 5.12c) existiert ein Plan, der ein Überholmanöver beinhaltet. Ein entsprechender Spurwechselhinweis wird in Form von zwei grünen Pfeilen angezeigt. Vor Erreichen der Abzweigung wird der Spurwechselhinweis nicht mehr gegeben (siehe Abbildung 5.12d), da nun kein Plan mehr existiert, bei dem das Fahrzeug überholt wird und der dennoch auf Fahrspursegment U_7 endet.

Es ist ersichtlich, dass nahe der Abzweigung ein Überholen trotz gegebener Spurwechselempfehlung nicht mehr möglich ist, ohne die Ausfahrt zu verpassen, da sich vor dem zu überholenden Fahrzeug noch ein weiteres Fahrzeug befindet. Eine Überholempfehlung wird dennoch gegeben, da zum einen das zweite vorausfahrende Fahrzeug nicht erkannt wird und somit nicht in die Planung mit aufgenommen wird. Hier könnten Ansätze zur Integration von Perzeptionsunsicherheiten (siehe Kapitel 4.5.2) Verbesserungen zeigen. Zum andern geht durch die Reduktion des Planungsproblems auf den semantischen Zustandsraum Information über die dynamische Fahrbarkeit eines Plans verloren. Hier zeigt sich deutlich der Bedarf einer Dynamikvalidierung der semantischen Pläne, wie sie in Kapitel 6 vorgestellt wird.

Automatisierter Spurwechsel

Die semantische Spurwechsellinformation kann auch im Bereich des automatisierten Fahrens eingesetzt werden. Um dies zu zeigen, wurde eine Funktion zum automatisierten Spurwechsel an die Simulationsumgebung CarMaker¹ angebunden. Diese Simulationsumgebung bietet den Vorteil, dass das simulierte Fahrzeug bereits Basisfahrfunktionen, wie das Folgen einer Fahrspur, beherrscht. Des Weiteren stehen abstrakte Sensormodelle zur Verfügung, die eine Erkennung anderer Verkehrsteilnehmer leicht möglich machen.

Auf Basis der Simulationsdaten wurde eine Funktion ähnlich dem Spurwechselasistenten implementiert. Die Spurwechselempfehlung wird dabei jedoch an das automatisierte Fahrzeug weitergegeben, das dann einen Spurwechsel ausführt. Zusätzlich wird die Spurwechselempfehlung in einem head-up-display (HUD) angezeigt. Abbildung 5.13 zeigt einen beispielhaften Ablauf eines automatisierten Spurwechsels in der Simulation. Zu den gezeigten Zeitpunkten ist jeweils eine

¹ CarMaker | IPG Automotive: ipg-automotive.com/de/produkte-services/simulation-software/carmaker/ (23.11.2017)



(a) Ein Spurwechselhinweis wird gegeben um das blaue Fahrzeug zu überholen



(b) Das automatisierte Fahrzeug wechselt die Spur



(c) Die Spur wird gehalten und das blaue Fahrzeug wird überholt



(d) Erst nachdem genügend Sicherheitsabstand zum blauen Fahrzeug aufgebaut wurde, wird ein Spurwechsel zurück auf die rechte Spur durchgeführt

Abbildung 5.13: Automatisierter Spurwechsel in der Simulation

Außenansicht des Fahrzeugs sowie oben rechts eine Sicht aus dem Fahrzeuginneren dargestellt. Hier ist auch der Spurwechselhinweis als grüner Pfeil im HUD des Fahrzeugs zu erkennen.

5.4 Zusammenfassung und Fazit

Da der semantische Zustandsraum die Form eines gerichteten Graphen aufweist, können zur semantischen Pfadplanung Standardverfahren der Graphensuche wie A*-, Breiten- oder Tiefensuche verwendet werden. Für die weitere Verarbeitung der semantischen Pläne ist es notwendig, dass nicht nur ein Plan, sondern mehrere bzw. alle Pläne zu den verschiedenen möglichen Zielzuständen gefunden werden. Daher sind Anpassungen an den jeweiligen Verfahren notwendig. Eine A*-Suche für alle Zielzustände ist schnell, kann jedoch zu suboptimalen Plänen führen. Die Suchverfahren EBFS und EDFs liefern hingegen alle Pläne, haben jedoch Nachteile bei der Laufzeit.

Um die Anzahl der Pläne bei diesen Verfahren zu reduzieren, ohne die optimal Lösung zu verlieren, werden Homotopieklassen auf semantischen Plänen defi-

5 Planung auf semantischen Zuständen

niert und ausgenutzt. Durch die Anwendung sogenannter Homotopiebausteine während der Graphensuche können Pläne aus der gleichen Homotopieklasse zusammengeführt werden und somit weniger und allgemeinere Pläne erzeugt werden. Die Evaluation zeigt, dass dadurch die Anzahl an gefundenen Plänen signifikant reduziert werden kann.

Die beschriebenen Suchverfahren werden auch in zwei Anwendungen beispielhaft evaluiert. In einem Überholassistenten wird dem Fahrer basierend auf Kamerabildern eines Versuchsfahrzeugs und einer bildbasierten Fahrzeugdetektion eine Möglichkeit zum Überholen als Spurwechselhinweis gegeben. Ein ähnliches Verfahren lässt sich auch in Kombination mit einer automatisierten Längs- und Querführung für das automatisierte Fahren verwenden, wie simulativ gezeigt werden konnte. Die Evaluation zeigt die Anwendbarkeit der gezeigten semantischen Planungsverfahren, verdeutlicht jedoch auch, dass eine Planung rein auf semantischer Ebene nicht ausreichend ist, da nicht garantiert werden kann, dass die gefundenen Pläne auch mit der beschränkten Dynamik eines realen Fahrzeugs gefahren werden können. Daher wird im folgenden Kapitel eine Validierung und Evaluierung der semantischen Pläne auf Basis der Fahrzeugdynamik vorgestellt.

6 Dynamikvalidierung und -evaluierung

Durch die Abstraktion des Planungsraums auf eine semantische Ebene ist die dynamische Fahrbarkeit der geplanten Manöverketten nicht direkt ableitbar. Um die dynamische Fahrbarkeit einer Manöverkette zu validieren, wird entsprechend dem vorgestellten Konzept ein optimales Geschwindigkeitsprofil erstellt, das den Randbedingungen der Manöverkette entspricht. Die hierzu eingesetzte Geschwindigkeitsprofiloptimierung muss gewisse Eigenschaften aufweisen, die sich aus den im Konzept in Kapitel 3.4.4 definierten Anforderungen an die Dynamikvalidierung ableiten: Vollständigkeit und geringe Laufzeit. Für die Dynamikevaluierung kommt noch die globale Optimalität hinzu.

Im Folgenden werden zunächst ausgehend von den Anforderungen verschiedene Eigenschaften der Geschwindigkeitsprofiloptimierung abgeleitet (Kapitel 6.1). Dann werden in Kapitel 6.2 verschiedene Optimierungsverfahren vorgestellt, um optimale Geschwindigkeitsprofile unter gegebenen Randbedingungen abzuleiten. Diese werden anhand ihrer Eigenschaften für die Anwendung im Rahmen der semantischen Planung bewertet und es wird ein geeignetes Verfahren ausgewählt. Um dieses anwenden zu können, wird in Kapitel 6.3 ein Vorgehen zur Darstellung eines semantischen Plans in der Ort-Zeit-Ebene vorgestellt. Es ist möglich, schon zur Planungszeit Validierungsschritte auf Teilplänen durchzuführen. Daher wird anschließend ein Suchverfahren beschrieben, das die Dynamikvalidierung bereits während der semantischen Planung integriert (Kapitel 6.3.2). Um die Anwendbarkeit des Verfahrens zu zeigen, wird dieses in Kapitel 6.4 auf die bekannten synthetischen Szenarien angewendet, das Laufzeitverhalten evaluiert und erwartete Verbesserungen in der Anwendung Überholassistentz gezeigt.

6.1 Anforderungen und Eigenschaften der Geschwindigkeitsprofiloptimierung

Aufgabe einer Geschwindigkeitsprofiloptimierung ist es, bei gegebener Zielfunktion $f(x(t))$ bzw. $f(v(x))$ die Parameter der Funktion, also eine 1D-Trajektorie $x(t)$ oder ein Geschwindigkeitsprofil $v(x)$ zu finden, so dass f minimal ist. Die Funktion f wird häufig auch als Kostenfunktion bezeichnet. Zur besseren Lesbarkeit wird häufig nur von Geschwindigkeitsprofil gesprochen, auch wenn die Aussagen entsprechend auch für Trajektorien gelten.

Um ein Verfahren zur Geschwindigkeitsprofiloptimierung zur Bewertung und Validierung von Manöversequenzen anzuwenden, muss es gewisse Eigenschaften aufweisen. Diese lassen sich aus den konzeptionellen Anforderungen an die Dynamikvalidierung und -evaluation ableiten. Im Folgenden werden daher verschiedene Eigenschaften der Verfahren vorgestellt, kurz beschrieben und ihre Auswirkung auf die Erfüllung der Anforderungen dargestellt.

6.1.1 Repräsentation des Lösungsraums

Wie oben erwähnt, lässt sich die Längsdynamik eines Fahrzeugs als Trajektorie, also als Funktion der Position x des Fahrzeugs entlang der Strecke über die Zeit innerhalb eines definierten Zeitintervalls $t \in [t_{\min}, t_{\max}]$ darstellen: $x(t)$. Eine alternative, zeitunabhängige Darstellung ist die des Geschwindigkeitsprofils als Darstellung der Geschwindigkeit v über der Position auf einer gegebenen Strecke $x \in [x_{\min} : x_{\max}]$: $v(x)$. Die Wahl der Repräsentation des Lösungsraums hat Auswirkungen auf die Vollständigkeit des Verfahrens. Da bei einem Geschwindigkeitsprofil genaue Dauern von Standzeiten mit $v = 0$ nicht modelliert werden können, hat dies negative Auswirkungen auf die Vollständigkeit des Verfahrens, da ggf. Lösungen mit Standzeiten nicht gefunden werden können.

6.1.2 Diskretisierung des Zustands- und Aktionsraums

Um die Berechnung des Geschwindigkeitsprofils handhabbar zu machen, wird in einigen Verfahren der Zustandsraum oder auch der Aktionsraum diskretisiert. Häufig kommen dabei äquidistante Diskretisierungen zum Einsatz. So werden Trajektorien oft in äquidistanten Zeitschritten repräsentiert. Andere Verfahren repräsentieren den Zustandsraum in Form eines äquidistanten Gitters z. B. in den Dimensionen Ort, Zeit und Geschwindigkeit.

Auch der Aktionsraum kann diskretisiert werden. Für die Geschwindigkeitsprofiloptimierung wird das Fahrzeugmodell auf die Modellierung der Längsdynamik beschränkt. Somit ergibt sich eine natürliche Form der Diskretisierung durch die verschiedenen Betriebsmodi eines Fahrzeugs, wie sie in Kapitel 2.3.1 dargestellt werden. Einige Betriebsmodi wie z. B. Bremsen und Beschleunigen können jedoch kontinuierlich parametrisiert werden. Eine Einschränkung auf eine diskrete Zahl von Aktionen kann in Zusammenhang mit der Wahl der Zustandsraumrepräsentation zu einer zu starken Einschränkung der Lösungsmöglichkeiten führen.

Eine Diskretisierung von Zustands- und/oder Aktionsraum kann sich also negativ auf die Vollständigkeit des Verfahrens auswirken. Bei einer zu groben äquidistanten Diskretisierung können z. B. mögliche Lösungen nicht mehr darstellbar sein. Daher sind Verfahren mit einer kontinuierlichen Repräsentation von Zustands- und Aktionsraum potentiell besser geeignet.

6.1.3 Mögliche Verkehrsszenenklassen

Entsprechend ihrer Randbedingungen sind verschiedene Verfahren für verschiedene Klassen von Verkehrsszenen geeignet. Eine der einfachsten Formen von Verkehrsszenen ist die statische Szene, in der sich keine bewegten oder sich ändernden Elemente befinden. Es wird also nur der Straßenverlauf ggf. mit Steigungen, Kurvenkrümmungen und Geschwindigkeitsbeschränkungen betrachtet. Dynamische Verkehrsteilnehmer wie andere Fahrzeuge werden nicht modelliert.

Eine Erweiterung der statischen Szene ist die semistatische Szene, in der sich ändernde, aber nicht bewegliche Elemente hinzu kommen. Dies können z. B. Ampelanlagen sein, die eine feste Position, jedoch zeitlich variable Zustände haben. Eine dynamische Szene enthält hingegen auch bewegliche Elemente wie andere Fahrzeuge oder Fußgänger und Radfahrer.

Die Randbedingungen der Manöverkettenvalidierung ähneln denen der dynamischen Verkehrsszene. Daher sind Verfahren, die mit dieser Klasse von Verkehrsszenen umgehen können, besser geeignet.

6.1.4 Art des Optimums

Eine wichtige Eigenschaft eines Optimierungsverfahrens ist die Art des bestimmten Optimums. In der Regel weist f neben dem globalen Optimum, also dem Punkt an dem f seinen kleinsten Funktionswert annimmt, mehrere Punkte im Parameterraum auf, die lokal optimal sind. Das heißt, dass eine lokale Veränderung der Parameter zu einer Verschlechterung von f führt. Man spricht in diesem Fall von einem lokalen Optimum. Optimierungsverfahren unterscheiden sich in der Art des gefundenen Optimums. Viele Verfahren sind lediglich in der Lage, ein lokales Optimum in der Nähe eines initialen Parametersatzes zu finden. Somit hängt die Güte des Verfahrens auch von der Wahl dieses initialen Parametersatzes ab. Andere Verfahren sind hingegen in der Lage, das globale Optimum unabhängig von einer Initialisierung zu finden.

6.1.5 Zielfunktion

Bei der Optimierung können verschiedene Ziele verfolgt werden. Typische Optimierungskriterien sind Reisezeit, mittlere Geschwindigkeit, Komfort oder Energieverbrauch in Form von Litern Treibstoff bzw. Kilowattstunden elektrischer Energie. Die gleichzeitige Optimierung mehrerer dieser Kriterien ist nur eingeschränkt möglich. Es kann lediglich ein so genanntes Pareto-Optimum erreicht werden, bei dem sich eine Zielgröße verschlechtert, wenn eine andere verbessert wird.

6 Dynamikvalidierung und -evaluierung

Als Zielfunktion kann somit eine gewichtete Summe der einzelnen Zielkriterien verwendet werden:

$$f(x(t)) = \sum_i w_i f_i(x(t)), \quad (6.1)$$

wobei w_i das jeweilige Gewicht der ein Kriterium bewertenden Funktion f_i ist.

Eine Möglichkeit die Gewichte der Zielfunktion zu bestimmen ist die Anpassung dieser an den (Mit-)Fahrer. Verschiedene Fahrer haben unterschiedliche Vorlieben bezüglich Fahrkomfort und auch die präferierte Gewichtung zwischen Reisezeit und Energieverbrauch kann von Fahrt zu Fahrt variieren. Gerade bei Fahrerassistenzfunktionen spielt die Anpassung der Gewichtung an den Fahrer für die Akzeptanz und Effektivität der Hinweise eine große Rolle [BNK+11].

Im Folgenden wird an einigen Stellen eine spezielle Form der Zielfunktion verwendet, die die Kriterien Reisezeit f_R und Verbrauch f_V berücksichtigt:

$$f_{RV}(x(t)) = \alpha \cdot k_R f_R(x(t)) + (1 - \alpha) \cdot k_V f_V(x(t)) \quad (6.2)$$

Dabei entspricht $\alpha = 0$ einer verbrauchsorientierten Optimierung ohne Berücksichtigung der Zeit und $\alpha = 1$ einer Optimierung der Reisezeit ohne Berücksichtigung des Verbrauchs. Die beiden Faktoren k_{Zeit} und $k_{Verbrauch}$ dienen der Normierung. Als Kostenfunktion f_R wird die zeitliche Gesamtlänge der Trajektorie verwendet. Für die Berechnung des Verbrauchs f_V kommen die in Kapitel 2.3.1 vorgestellten Verbrauchsmodelle zum Einsatz.

Die Form der Zielfunktion bestimmt das zu findende Optimum. Einige Optimierungsverfahren sind auf die Verwendung bestimmter Zielfunktionen beschränkt, oder es wird ihre Eigenschaft der globalen Optimalität nur für gewisse Funktionen zugesichert. Gerade im Zusammenspiel mit heuristikbasierten Planungsverfahren wie der A*-Suche ist die Form der Optimierungsfunktion bei der Evaluierung der Pläne relevant. Für die reine Pfadvalidierung spielt sie keine Rolle.

6.1.6 Laufzeit

Eine wichtige Eigenschaft für die Anwendbarkeit der Verfahren ist das Laufzeitverhalten. Im Kontext der Suche nach geeigneten Geschwindigkeitsprofilen für Fahrerassistenzsysteme und automatisiertes Fahren können zwei verschiedene Laufzeiten unterschieden werden: Die Laufzeit zur Vorberechnung einer Verkehrsszene und die Laufzeit für die Bestimmung eines optimalen Geschwindigkeitsprofils bzw. einer optimalen Aktion bei gegebenem Startzustand des Ego-Fahrzeugs innerhalb dieser Szene. Entsprechend der angestrebten Anwendung und des verwendeten Verfahrens kann Rechenzeit auf beide Bereiche verteilt werden.

Eine umfangreichere Vorberechnung auf der Verkehrsszene zu Gunsten kürzerer Rechenzeiten bei der Bestimmung einzelner optimaler Aktionen bietet sich vor allem für statische und semistatische Szenen an oder, wenn der dynamische

Verlauf der Szene im Vorfeld bekannt ist oder über lange Zeit prädiziert werden kann. Kurze Rechenzeiten für die Bestimmung einer optimalen Aktion sind vor allem bei Fahrerassistenzfunktionen wünschenswert, da hier nicht gewährleistet ist, dass der Fahrer der vorgeschlagenen Aktion folgt. Durch die kurze Rechenzeit kann reaktiv auf das Fahrverhalten des Fahrers reagiert werden. Bei Funktionen für das automatisierte Fahren hingegen ist davon auszugehen, dass das Fahrzeug den ermittelten Aktionen folgt. Somit können ganze Aktionsketten (Trajektorien) vorberechnet werden, was längere Zeit in Anspruch nehmen darf.

Für die Validierung semantischer Manöverketten sind beide Laufzeiten von Bedeutung, da jede Kette neue Randbedingungen definiert und somit eine neue Vorberechnung erfordert und ein Geschwindigkeitsprofil abgeleitet werden muss.

6.2 Optimierungsverfahren

Im Rahmen dieser Arbeit wurden verschiedene Optimierungsverfahren auf ihre Anwendbarkeit zur Geschwindigkeitsprofiloptimierung sowie zur dynamischen Validierung von Manöverketten hin untersucht. Im Folgenden werden verschiedene Optimierungsverfahren vorgestellt, ihre Eigenschaften anhand der oben beschriebenen Kriterien herausgestellt und ihre Anwendbarkeit im Bereich der Geschwindigkeitsprofiloptimierung an Beispielanwendungen gezeigt (vergleiche [KSZ15]). Im Anschluss (Kapitel 6.2.6) kann dann auf Basis ihrer Eigenschaften und der definierten Anforderungen ein Verfahren für die Dynamikvalidierung und Dynamikevaluierung ausgewählt werden.

6.2.1 Dynamisches Programmieren

Das dynamische Programmieren ist ein globales Optimierungsverfahren. Es basiert auf der Annahme, dass sich eine optimale Gesamtlösung aus optimalen Teillösungen zusammensetzt. Das Verfahren kann sowohl kontinuierlich also auf einem kontinuierlichen Aktions- und Zustandsraum sowie auf diskreten Räumen angewendet werden. Aufgrund der verwendeten Fahrzeugmodelle und der Komplexität des betrachteten Problems kommt hier das diskrete Verfahren zum Einsatz. Details hierzu können der Arbeit [Car13] entnommen werden.

Dazu wird der Zustandsraum des Fahrzeugs bezüglich Position, Geschwindigkeit und Zeit modelliert und geeignet meist äquidistant diskretisiert. Ausgehend vom örtlichen Ende des Szenarios werden für jeden diskreten Punkt im Raum eine Bewertung sowie die zugehörige lokal optimale Aktion bestimmt. Für einen Zustand s wird für jede mögliche Aktion A_i der resultierende Zustand $s_i = A_i(s)$ ermittelt. Jeder Aktion kann somit eine Bewertung $B(A_i)$ als Summe der Kosten für die Aktion A_i und der Bewertung $B(s_i)$ des Folgezustands zugeordnet werden.

6 Dynamikvalidierung und -evaluierung

Die Bewertung des Zustands $B(s)$ ist dann das Minimum über alle Aktionsbewertungen. Die Aktion mit der kleinsten Bewertung wird als Strategie (engl.: policy) in diesem Zustand festgehalten.

Durch die Anwendung einer Aktion auf einen Fahrzeugzustand wird in der Regel ein Zustand erreicht, der nicht Teil des diskretisierten Zustandsraums ist. In diesem Fall kann die Bewertung des Folgezustands aus den Bewertungen der umliegenden Zustände durch Interpolation ermittelt werden.

Als Kostenfunktion kann z. B. das oben beschriebene gewichtete Mittel aus Energie- und Zeitbedarf $f_{RV}(x(t))$ (vergleiche Gleichung (6.2)) verwendet werden. Verkehrsregeln wie die gültige Maximalgeschwindigkeit, der Sicherheitsabstand zu anderen Verkehrsteilnehmern sowie das Verbot eine rote Ampel zu überfahren fließen ebenfalls als sehr hohe Kosten in die Bewertungsfunktion mit ein. So können zum einen Regelverstöße praktisch ausgeschlossen werden, jedoch auch leichte Regelverstöße wie Fahren mit leicht überhöhter Geschwindigkeit zu Gunsten der Gesamtbewertung in Grenzen zugelassen werden.

Nach der Bewertung aller Zustände im betrachteten Zustandsraum liegt für jeden Zustand eine optimale Aktionswahl (policy) vor. Entsprechend dieser kann vom aktuellen Fahrzeugzustand aus ein optimales Aktionsprofil und somit ein optimales Geschwindigkeitsprofil abgeleitet werden.

Im Folgenden wird das Verfahren an zwei Beispielen, einer Annäherung an eine Geschwindigkeitsbeschränkung sowie eine Annäherung an eine Ampel, gezeigt.

Beispiele und Anwendung

Das erste Beispiel soll die Auswirkung verschiedener Gewichtungen in der Zielfunktion auf das optimale Geschwindigkeitsprofil verdeutlichen. In Abbildung 6.1 ist das Ergebnis einer Optimierung mittels dynamischem Programmieren für verschiedene Gewichtungen der Kostenkriterien dargestellt. Als Szenario wurde die Anfahrt auf eine Geschwindigkeitsbeschränkung von 60 km/h auf 30 km/h an der Position 300 m gewählt. Die Ausgangsgeschwindigkeit beträgt 50 km/h. Es ist jeweils die optimale Policy (farbige Punkte) sowie eine daraus resultierende optimale Trajektorie ist als farbige Linie dargestellt. Die Farbe repräsentiert dabei die gewählte Aktion. Da das Szenario zeitunabhängig ist, wurde auf eine explizite Modellierung der Zeit verzichtet.

Wird bei der Optimierung lediglich der Energieverbrauch berücksichtigt ($\alpha = 0$), so schreibt die Policy einen Ausrollvorgang bis zum Stillstand vor. Da das Erreichen des Ziels in endlicher Zeit in diesem Fall kein Optimierungskriterium darstellt, wird dieses auch nicht erreicht. Eine ausgewogenere Fahrweise wird durch die Wahl von $\alpha = 0.5$ bzw. $\alpha = 0.75$ erreicht. Erstere stellt eine zwar zielorientierte jedoch effiziente Fahrweise dar, bei der auf die Geschwindigkeitsbeschränkung hin ausgerollt wird. Das Geschwindigkeitsprofil für $\alpha = 0.75$ zeigt ein Verhalten, bei dem vor einer Geschwindigkeitsbeschränkung zunächst etwas

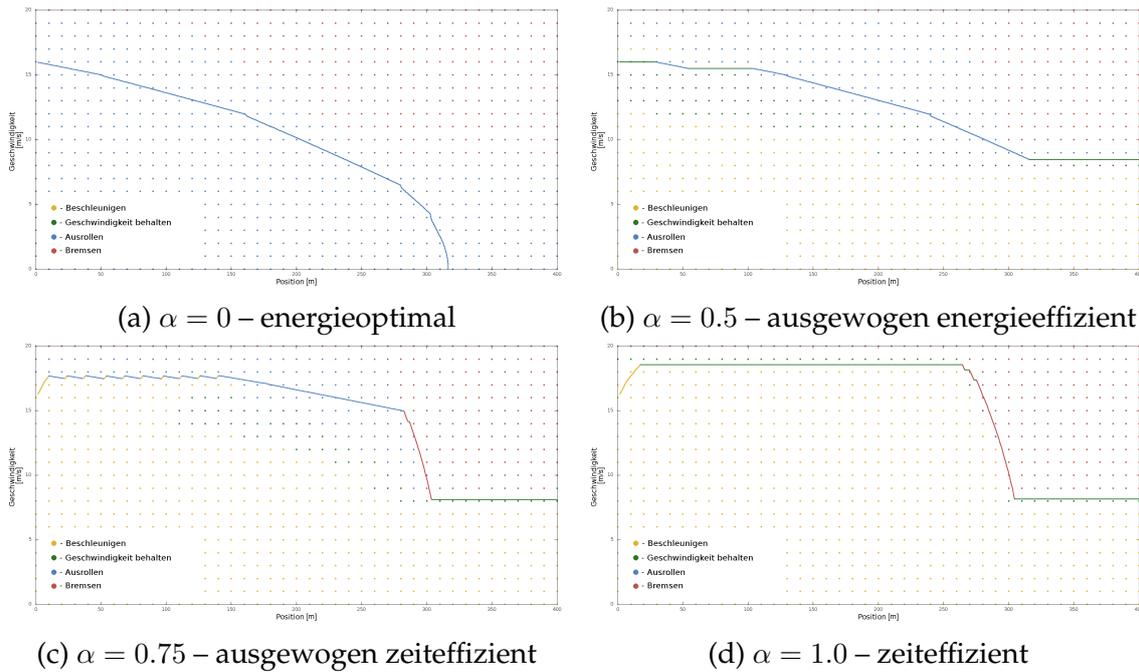


Abbildung 6.1: Beispiel Dynamische Programmierung: Anfahrt auf eine Geschwindigkeitsbeschränkung mit variierender Gewichtung der Kosten für Zeit- und Energieverbrauch

Bildquelle: [Car13]

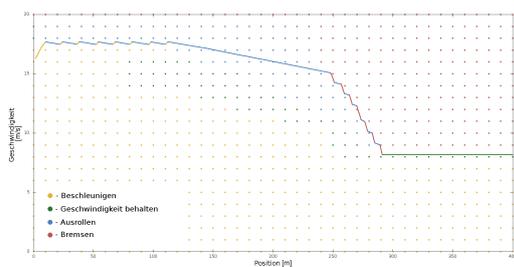
ausrollen gelassen und dann kurz vor der Beschränkung gebremst wird. Wie zu erwarten spiegelt $\alpha = 1.0$ ein Fahrverhalten wieder, bei dem zu Beginn auf die maximal zulässige Geschwindigkeit von 60 km/h beschleunigt wird um diese möglichst lange zu halten und erst kurz vor dem 30er-Schild stark abzubremsen.

Die Auswirkung der Bestrafung für Fahren mit überhöhter Geschwindigkeit ist in Abbildung 6.2 dargestellt. Beide Berechnungen wurden mit $\alpha = 0.75$ durchgeführt. Im linken Diagramm fließt eine Überschreitung der Maximalgeschwindigkeit sehr stark in die Bewertungsfunktion mit ein. Es ist zu erkennen, dass das Fahrzeug die Geschwindigkeitsbeschränkung mit der vorgegebenen Geschwindigkeit erreicht und diese auch im weiteren Verlauf einhält.

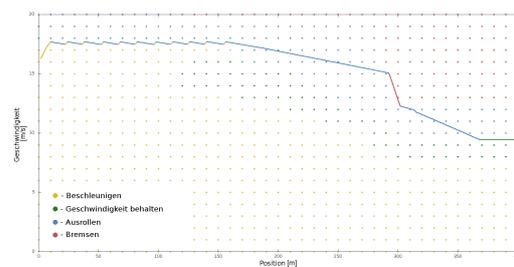
Bei deutlich geringerer Bewertung (rechts) wird das 30er-Schild mit deutlich höherer Geschwindigkeit erreicht und erst danach gebremst. Auch wird nicht bis auf die dann zulässige Geschwindigkeit heruntergebremst, sondern zugunsten der Zeit- und Energieeffizienz weiter ausrollen gelassen. Die Reisegeschwindigkeit ist auch etwas höher als die zulässige Geschwindigkeit.

Ein anderer hier zu beobachtender Effekt sind Fehler, die aufgrund der Diskretisierung auftreten. Im linken Diagramm ist zwischen 250 m und 300 m ein schneller Wechsel zwischen Bremsen und Ausrollen zu beobachten. Dieses Verhalten ist bei der verwendeten Bewertungsfunktion suboptimal. Da Bremsen und Ausrollen in diesem Modell energetisch keinen Unterschied in der Bewertung ausmachen, wäre ein Geschwindigkeitsprofil, das zunächst weiter ausrollt und dann bremst

6 Dynamikvalidierung und -evaluierung



(a) Hohe Kosten für überhöhte Geschwindigkeit



(b) Niedrige Kosten für überhöhte Geschwindigkeit

Abbildung 6.2: Beispiel Dynamische Programmierung: Anfahrt auf eine Geschwindigkeitsbeschränkung mit variierender Bewertung für überhöhte Geschwindigkeit.

Bildquelle: [Car13]

zeitlich und somit auch gesamt besser bewertet. Die schnellen Wechsel sind auf die notwendige Interpolation bei der Berechnung der Policy zurückzuführen.

Das gezeigte Verfahren lässt sich auch auf dynamische Szenen anwenden, solange sie entweder zeitlich begrenzt oder zyklisch in der Zeit sind. Ein Beispiel für ein zyklisches Szenario ist das Passieren einer Ampel. Hier wird ein dreidimensionaler Zustandsraum aus Ort, Geschwindigkeit und Zeit verwendet, wobei die zeitliche Dimension auf die Länge eines Ampelzyklusses beschränkt wird und diese als zyklisch angenommen wird.

Abbildung 6.3 zeigt eine Menge von optimalen Trajektorien zur Durchfahrt einer Ampel projiziert auf die Ort-Zeit-Ebene. An Position 400 m befindet sich eine Ampel. Diese ist 60 s lang grün und 60 s lang rot, dargestellt durch die rote bzw. grüne vertikale Linie. Des Weiteren sind 24 Trajektorien dargestellt, die zu unterschiedlichen Zeiten im Ampelzyklus bei Position 0 m starten.

Die Farbe der Linie gibt den an dieser Stelle gewählten Betriebsmodus wieder. Zusätzlich zu den im vorigen Beispiel gewählten Betriebsmodi Bremsen (rot), Ausrollen (blau), Konstantfahrt (grün) und Beschleunigen (gelb) wurde hier auch die Rekuperation (hell grün) als Option berücksichtigt.

Es ist zu erkennen, dass alle Trajektorien, die Ampel in einer Grünphase durchfahren. In der Mitte der Trajektorienschar gibt es viele, die mit konstanter Geschwindigkeit die Ampel passieren. Trajektorien, die zeitlich davor oder danach starten verzögern durch eine Kombination aus zunächst Rekuperation und Ausrollen. Diese passieren die Ampel zum frühest möglichen Zeitpunkt mit möglichst hoher Geschwindigkeit. Dieses Muster ist weitestgehend unabhängig von der gewählten Gewichtung zwischen Reisezeit und Energieverbrauch.

Auch in diesem Beispiel sind suboptimale Ergebnisse aufgrund von Diskretisierungseffekten zu erkennen. Deutlichstes Beispiel sind die Trajektorien mit Startzeitpunkt 85 s und 90 s. Hier wäre es möglich und optimal, die Ampel mit konstanter Geschwindigkeit zu passieren. Auch andere Suboptimalitäten, wie der

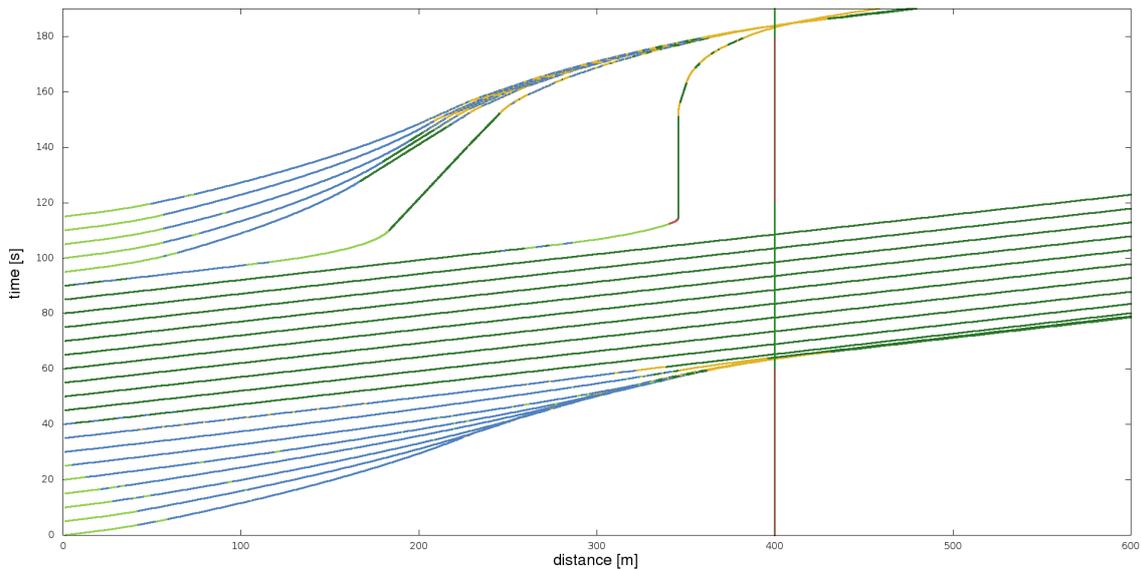


Abbildung 6.3: Beispiel Dynamische Programmierung: Optimales Passieren einer Ampel

Erstveröffentlichung in: [KSZ15] (Kohlhaas et al.)

schnelle Wechsel zwischen ausrollen und Beschleunigen sowie Beschleunigungen nach der Ampeldurchfahrt sind auf die Diskretisierungseffekte zurückzuführen.

Fazit

Dynamische Programmierung liefert ein global optimales Geschwindigkeitsprofil, jedoch treten aufgrund der Diskretisierung des Zustandsraums Fehler auf, die z. B. zu schnell iterierenden Aktionsketten führen. Für ein hinweisgebendes, passives Assistenzsystem ist dieses Verfahren daher nicht geeignet, da die häufigen Aktionswechsel nicht an den Fahrer kommunizierbar sind. Des Weiteren ist trotz Verwendung von performanten Längsdynamikmodellen eine Berechnung der Policy auf großen Zustandsräumen nicht in Echtzeit möglich.

Stärken zeigt das Verfahren bei der Behandlung von komplexen Szenarien, da keine speziellen Annahmen an die Art der Szenarien oder Bewertungsfunktionen gestellt werden. Da im Regelfall Lösungen nahe der globalen Optima gefunden werden, eignet sich das Verfahren gut für die initiale Untersuchung von Szenarien. Wie in den zwei Beispielen gezeigt, kann z. B. das prinzipielle Erscheinungsbild einer Policy, also die Abfolge der einzelnen Betriebsmodi, in einem Szenario festgestellt werden.

Ist die Policy einer Szene einmal berechnet, lassen sich schnell entsprechende Trajektorien generieren. Dies ermöglicht die Anwendung für das automatisierte Fahren in bekannten vorberechenbaren Szenarien. Außerdem können für gegebene Szenarien mit gewissen Variationen in den Startbedingungen des Fahrzeugs statistische Aussagen über Einsparpotentiale gewonnen werden.

6.2.2 Modellbasierte lokale Optimierung

Für ein gegebenes Szenario kann mit entsprechendem Vorwissen ein parametrisches Modell für die gewünschte Trajektorie aufgestellt und optimiert werden. Als Ausgangspunkt können hierzu z. B. die Erkenntnisse aus der Optimierung mit dynamischer Programmierung (siehe Kapitel 6.2.1) verwendet werden. Im Regelfall ist die Anzahl der Parameter in diesen Modellen gering, so dass Standardverfahren zur Optimierung wie z. B. ein Gradientenabstiegsverfahren eingesetzt werden können. Auch ist häufig genügend Vorwissen über die Parameter des Modells verfügbar, so dass schon in der Nähe des Optimums mit der Optimierung gestartet werden kann. Die Vermeidung von Nebenmaxima ist somit von geringerer Bedeutung.

Ist die Reihenfolge der in der optimalen Trajektorie verwendeten Betriebsmodi bekannt, so kann eine Trajektorie eindeutig durch die Zeitpunkte der Wechsel zwischen den Modi dargestellt werden. Das Modell umfasst somit nur wenige Parameter, die zu optimieren sind.

Beispiele und Anwendungen

Um das Verfahren zu veranschaulichen, wird dieses auf das Ampelbeispiel von oben angewendet. Hier konnte durch die Optimierung mittels dynamischer Programmierung die Reihenfolge der Betriebsmodi für eine optimale Trajektorie festgestellt werden. Die Ergebnisse legen nahe, dass die optimale Trajektorie aus den Betriebsmodi Rekuperation, Ausrollen, Beschleunigen und Konstantfahrt in dieser Reihenfolge besteht. Wird dies als Modellannahme getroffen so müssen lediglich die Zeitpunkte der Moduswechsel geschätzt werden. Da am Ausgang der Szene mit der zulässigen Maximalgeschwindigkeit gefahren werden soll, lässt sich der Übergang zwischen Beschleunigen und Konstantfahrt aus den anderen beiden Zeitpunkte berechnen. Das zu optimierende Modell ist also ein Tupel zwei Zeitpunkten (t_a, t_b) .

Dieses Modell kann mittels Gradientenabstiegsverfahren optimiert werden. Als Kostenfunktion kommt wieder das oben beschriebene gewichtete Mittel aus Energie- und Zeitbedarf $f_{RV}(x(t))$ (vergleiche Gleichung (6.2)) zum Einsatz. Zusätzlich wird beim Passieren einer Ampel der Zeitabstand hin zu einer Grünphase bestraft. Der Gradient wird numerisch durch eine kleine Variation in jeweils einer der beiden Dimensionen bestimmt und normiert. Die Schrittweite beträgt zu Beginn eine Sekunde und wird, falls kein Abstieg mehr möglich ist, sukzessive halbiert, bis zu einer minimalen Schrittweite von 0,001 s.

Abbildung 6.4 zeigt eine Auswahl von Schritten des Gradientenabstiegsverfahrens. Zu Beginn sind die Übergangszeiten sehr früh gewählt, so dass eine Trajektorie mit praktisch konstanter Geschwindigkeit erzeugt wird. Diese passiert die Ampel während einer Rotphase. Schon nach 8 Schritten wird eine Trajektorie gefunden, die die Ampel während der Grünphase passiert. Diese wird in den weiteren

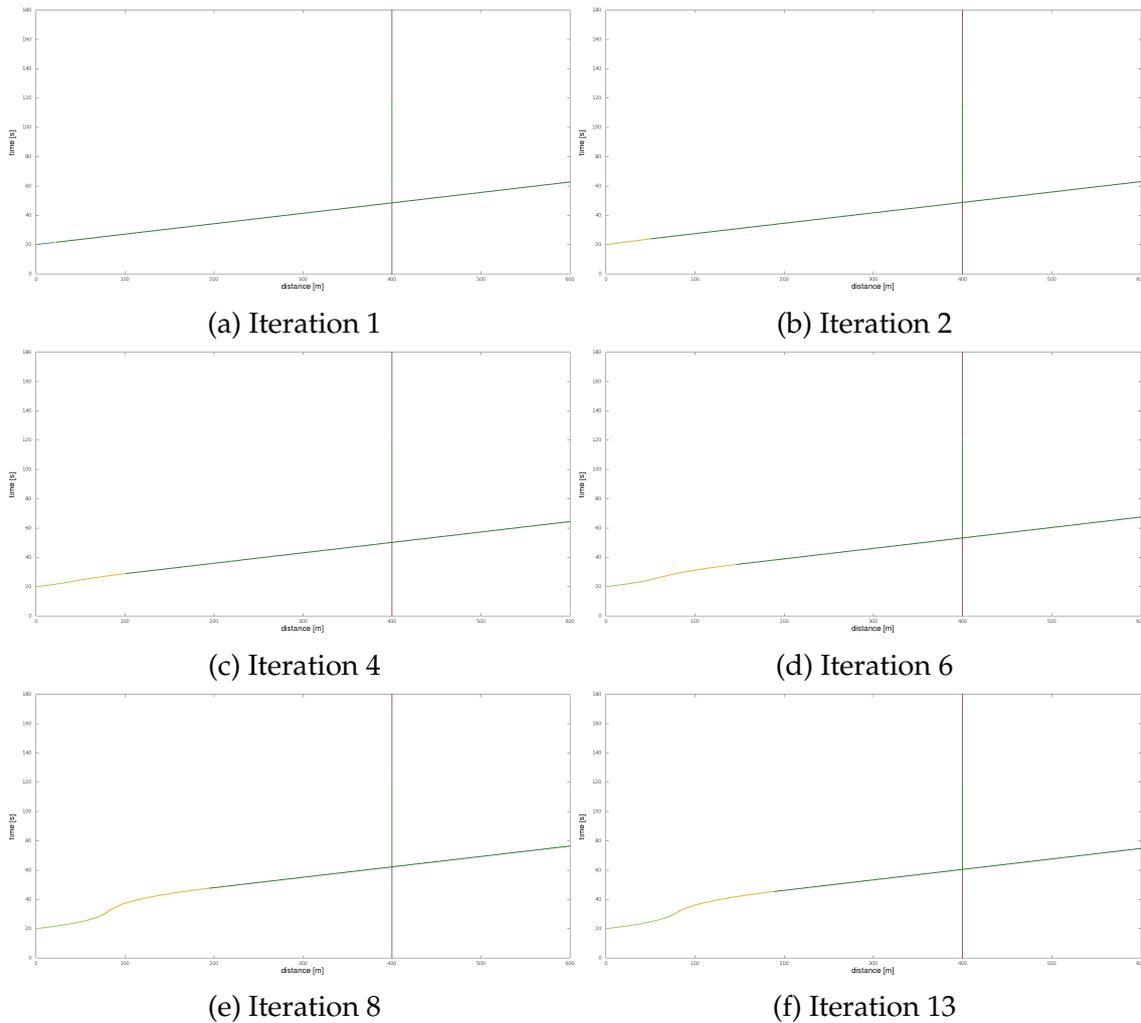


Abbildung 6.4: Beispiel Gradientenabstieg: Optimales Passieren einer Ampel

Erstveröffentlichung in: [KSZ15] (Kohlhaas et al.)

Schritten weiter optimiert, so dass bei Schritt 13 das finale Optimum gefunden wird.

Fazit

Das Gradientenabstiegsverfahren optimiert ein parametrisches Modell der Trajektorie. Es muss also gewährleistet sein, dass ein solches aufgestellt werden kann und sich die optimale Trajektorie mit diesem abbilden lässt. Für allgemeine Szenarien, bei denen sich ein solches Modell nur schwer bestimmen lässt, ist das Verfahren daher schlecht geeignet. Auch müssen gute Startwerte für die Optimierung gefunden werden, damit diese nicht in einem lokalen Optimum konvergiert. Diese Voraussetzungen sind im allgemeinen in der praktischen Anwendung der Fahrerassistenz oder des automatisierten Fahrens nicht gegeben. Sind diese Voraussetzungen jedoch erfüllt, ist es ein schnelles Verfahren, das ohne künstliche

Diskretisierung des Zustandsraums auskommt.

6.2.3 Vorberechnete optimale Geschwindigkeitsprofile

Für statische Verkehrsszenen kann die Berechnung einer Policy stark vereinfacht werden. Als Randbedingungen sind hier nur Geschwindigkeitsbeschränkungen relevant. Sie können aus den vorgegebenen Höchstgeschwindigkeiten oder aufgrund von Kurven abgeleitet sein. Aus der Optimierung mit dynamischer Programmierung (siehe Kapitel 6.2.1) kann abgeleitet werden, dass die Reihenfolge der Betriebsmodi für die Annäherung an eine Geschwindigkeitsbeschränkung durch die Größe der entsprechenden Verzögerungsbeschleunigung bestimmt wird: Konstantfahrt, Ausrollen, Rekuperation, Bremsen. Es ist zu beachten, dass dies entgegen der Reihenfolge bei der Annäherung auf dynamische Beschränkungen wie Ampeln und vorausfahrende Fahrzeuge ist, da in diesen Fällen eine höhere Geschwindigkeit bei der Annäherung keinen zeitlichen Vorteil, sondern nur einen energetischen Mehrverbrauch bedeuten würde. Ist keine Geschwindigkeitsbeschränkung in relevanter Nähe und das Fahrzeug langsamer als durch die aktuell gültige Maximalgeschwindigkeit vorgegeben, sollte mit Hinblick auf eine geringe Fahrzeit beschleunigt werden.

Um eine Policy aufzustellen, reicht es für eine Geschwindigkeitsbeschränkung, gegeben durch einen Sprung in $v_{max}(s)$, ein Annäherungsprofil für jeden der Betriebsmodi Ausrollen $v_A(s)$, Rekuperation $v_R(s)$ und Bremsen $v_B(s)$ zu berechnen, das die neue Geschwindigkeitsbeschränkung mit der richtigen Geschwindigkeit erreicht. Abbildung 6.5 zeigt schematisch die Ausrollprofile für die verschiedenen Betriebsmodi. Für ein Fahrzeug mit der Geschwindigkeit v an der Position s kann dann folgende Policy abgeleitet werden:

$$\begin{aligned}v &\geq v_B(s) \rightarrow \text{Bremsen} \\v < v_B(s) \wedge v &\geq v_R(s) \rightarrow \text{Rekuperation} \\v < v_R(s) \wedge v &\geq v_A(s) \rightarrow \text{Ausrollen} \\v < v_A(s) \wedge v &< v_{max}(s) \rightarrow \text{Beschleunigen} \\&\text{sonst} \rightarrow \text{Konstantfahrt}\end{aligned}$$

In der Praxis können die Vergleiche, wie später in den Beispielen und Anwendungen beschrieben, mit einer Hysterese verbunden werden, um schnelle Moduswechsel zu vermeiden.

Die Annäherungstrajektorien können mit Hilfe des in Kapitel 2.3.1 beschriebenen Fahrzeugmodells für beliebige Strecken berechnet werden. Eine Iteration entgegen der Fahrtrichtung ermöglicht eine effiziente Berechnung der Geschwindigkeitsprofile. Dabei wird das Modell ausgehend von einer Endgeschwindigkeit am Ende der Strecke rückwärts iterativ angewendet. Bei aufeinanderfolgenden Geschwindigkeitsbeschränkungen, wie sie z. B. vor Ortseinfahrten auftreten, kann es sein, dass

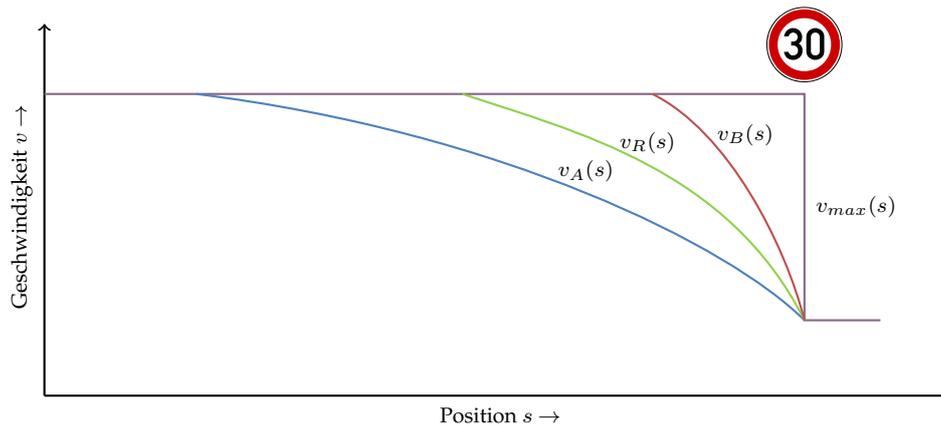


Abbildung 6.5: Berechnung der Policy zur Annäherung an statische Geschwindigkeitsbeschränkungen

Erstveröffentlichung in: [KSZ15] (Kohlhaas et al.)

die berechneten Geschwindigkeitsprofile an der vorderen Begrenzung eine geringere Geschwindigkeit aufweisen als diese. In diesem Fall ist die Beschränkung für dieses Geschwindigkeitsprofil nicht relevant und wird nicht berücksichtigt.

Beispiele und Anwendungen

Das hier beschriebene Optimierungsverfahren eignet sich gut, um die Geschwindigkeit des Fahrzeugs an statische Beschränkungen wie Kurven anzupassen. Dazu werden über die geplante Route die benötigten Geschwindigkeitsprofile vorberechnet. Basierend auf diesen und der aktuellen Fahrzeuggeschwindigkeit kann die optimale Policy hergeleitet werden. Da eine Policy für alle Fahrzeugzustände schnell abgeleitet werden kann, ist eine Anwendung für Assistenzfunktionen möglich, bei denen der Fahrzeugzustand vom Fahrer abhängt.

In einer exemplarischen Implementierung wird dem Fahrer der aktuell beste Betriebsmodus über ein Display kommuniziert. Abbildung 6.6 zeigt eine mögliche Darstellung der Hinweise im Kombi-Instrument des Fahrzeugs. Die Anzeige (Abbildung 6.6a) wurde auf Basis von Testfahrten mit Probanden iterativ optimiert und besteht aus mehreren Elementen. Oben wird die vom Fahrer durchzuführende Aktion (Wahl des Betriebsmodus des Fahrzeugs) textuell dargestellt. Diese dient vornehmlich dem Erlernen des Umgangs mit dem Assistenzsystem. Der Hinweis wird zusätzlich farbkodiert im Hintergrund der Anzeige dargestellt, z. B. blau für Ausrollen (coast) oder rot für Bremsen (brake). Dies dient der schnelleren Wahrnehmung der Hinweise durch den Fahrer. Die unteren zwei Zeilen dienen im gezeigten Beispiel lediglich in rundenbasierten Fahrversuchen mit Probanden als Rückmeldung an den Fahrer. Sie könnten in einem Assistenzsystem entfallen oder als Vergleich ähnlicher Situationen oder Fahrstrecken dienen.



(a) Brems- bzw. Ausrollhinweis wegen Geschwindigkeitsbeschränkung auf 50 km/h bzw. einer Kreuzung ohne Vorfahrt

(b) Integration in das Kombi-Instrument des Versuchsfahrzeugs

Abbildung 6.6: Fahrerassistenz für energieeffizientes Fahren auf statische Geschwindigkeitsbeschränkungen

Erstveröffentlichung in: [KSZ15] (Kohlhaas et al.)

Häufig werden Fahrhinweise schon gegeben, wenn der Fahrer den Grund dafür noch nicht sehen kann, weil z. B. ein Stoppschild hinter einer Kurve steht. Erste Probandentests haben gezeigt, dass die Akzeptanz der Hinweise größer ist, wenn der Grund für die Aktion mitgeteilt wird. Daher wird in der Mitte des Displays der Hinweisgrund schematisch dargestellt. Geschwindigkeitsbeschränkungen werden durch das entsprechende Verkehrszeichen visualisiert. Für Ampeln, Kreuzungen, Kurven und Abbiegungen werden entsprechende Schilder und Symbole verwendet.

Im oberen Bereich ist ein adaptiver Tacho dargestellt. Dieser zeigt die aktuell gefahrene Geschwindigkeit mit einer roten Tachonadel an. Der Tacho zeigt nur einen Ausschnitt des möglichen Geschwindigkeitsbereichs an. Die aktuell empfohlene Geschwindigkeit befindet sich dabei immer in der Mitte der Anzeige. Für eine effiziente Fahrweise muss der Fahrer also die Tachonadel ebenfalls in der Mitte der Anzeige halten. Bei optimalen Ausrollvorgängen verbleibt die Tachonadel in der Mitte der Anzeige, während sich die Geschwindigkeitsskala unter ihr hinwegbewegt. Durch die Bewegung des Tachos können Ausrollhinweise häufig vom Fahrer antizipiert werden.

Um bevorstehende Bremshinweise anzukündigen, dient die farbige Leiste oberhalb des Tachos. Hier wird der Geschwindigkeitsbereich rot markiert, in dem eine Bremsaktion notwendig wird. Bei der Anfahrt auf eine Geschwindigkeitsbeschränkung wird dieser Bereich von rechts her größer. Erreicht er die Tachonadel, so wird ein Bremshinweis gegeben.

Die Anzeige kann in das Kombi-Instrument eines Fahrzeugs eingeblendet werden. Abbildung 6.6b zeigt die Anzeige im Versuchsfahrzeug CoCar (vergleiche Kapitel 2.2.2). Hier ist ein Ausrollhinweis aufgrund eines bevorstehenden Abbiegevorgangs dargestellt.

Um dem Fahrer optimale Fahraktionen empfehlen zu können, werden entsprechende Geschwindigkeitsprofile für die geplante Strecke vorberechnet. Abbildung 6.7 zeigt beispielhaft die berechneten Geschwindigkeitsprofile für eine rund 12 km lange Landstraßenstrecke mit mehreren Ortsdurchfahrten. Es sind die Geschwindigkeitsprofile für die Maximalgeschwindigkeit aus Kurvengeschwindigkeiten und Geschwindigkeitsbeschränkungen (lila) sowie für Ausrollen (blau) und Bremsen (rot) dargestellt.

Vor allem vor Passagen mit starkem Gefälle kann es vorkommen, dass das optimale Geschwindigkeitsprofil zum Ausrollen sehr kleine Geschwindigkeiten vorgibt. Dies würde zu Verkehrsbehinderungen und einer Verringerung der Akzeptanz führen. Daher wird in dieser Anwendung auch eine Minimalgeschwindigkeit (gelb) verwendet. Als Minimalgeschwindigkeit wird in der Regel ein fester Bruchteil der Maximalgeschwindigkeit angenommen. Lediglich in Bereichen vor einer Reduzierung der Maximalgeschwindigkeit wird die Minimalgeschwindigkeit linear angepasst, so dass eine Verzögerung durch Ausrollen oder Bremsen möglich wird. In größerer Entfernung zu Geschwindigkeitsbeschränkungen wird somit eine zu geringe Geschwindigkeit vermieden.

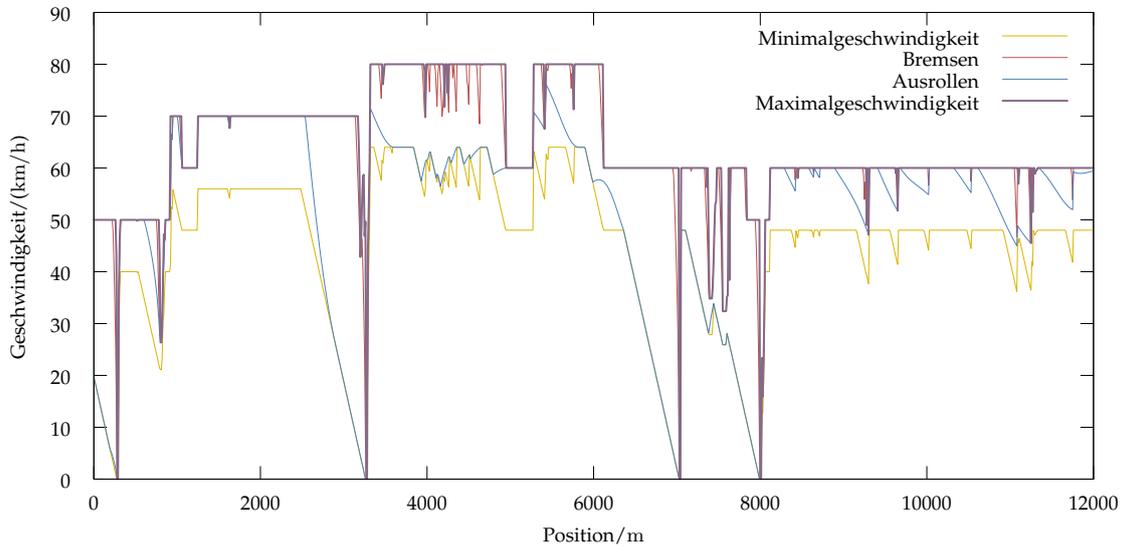
Abbildung 6.7b zeigt einen Ausschnitt der Strecke vor einer Kurve. Anhand einer auf den Fahrer einstellbaren Querbeschleunigung in der Kurve wird die maximale Kurvengeschwindigkeit berechnet. Aufgrund von Diskretisierungseffekten im Kartenmaterial ist die Maximalgeschwindigkeit im Kurveneingang (bei 790 m) sprunghaft. Das Geschwindigkeitsprofil für die Aktion Bremsen weist somit zwei Verzögerungsprofile auf. Bei der Aktion Ausrollen hingegen gibt es nur einen Verzögerungsvorgang, weil die Maximalgeschwindigkeit im Kurvenmittelpunkt zu einer früheren Verzögerung führt, als die im Kurveneingang.

Bei der Ableitung der Aktionshinweise wird jeweils anhand der aktuellen Fahrzeugposition und -geschwindigkeit die an dieser Stelle auszuführende Aktion bestimmt. In der Praxis kann es gerade bei sehr schneller Reaktion auf einen Ausrollhinweis und aufgrund von Ungenauigkeiten in der Strecken- und Fahrzeugmodellierung dazu kommen, dass das reale Verzögerungsprofil um das Ausrollprofil schwankt. Um ein schnelles Alternieren der Fahrhinweise in dieser Situation zu vermeiden, wird eine Hysterese angewendet. Wird z. B. ein Ausrollhinweis gegeben, so wird dieser auch weiterhin kommuniziert, auch wenn die aktuelle Geschwindigkeit innerhalb eines kleinen Geschwindigkeitsintervall unterhalb des Ausrollprofils liegt. In der Praxis hat sich ein Intervall von rund 5 km/h als ausreichend herausgestellt, um diesen Effekt weitestgehend zu vermeiden.

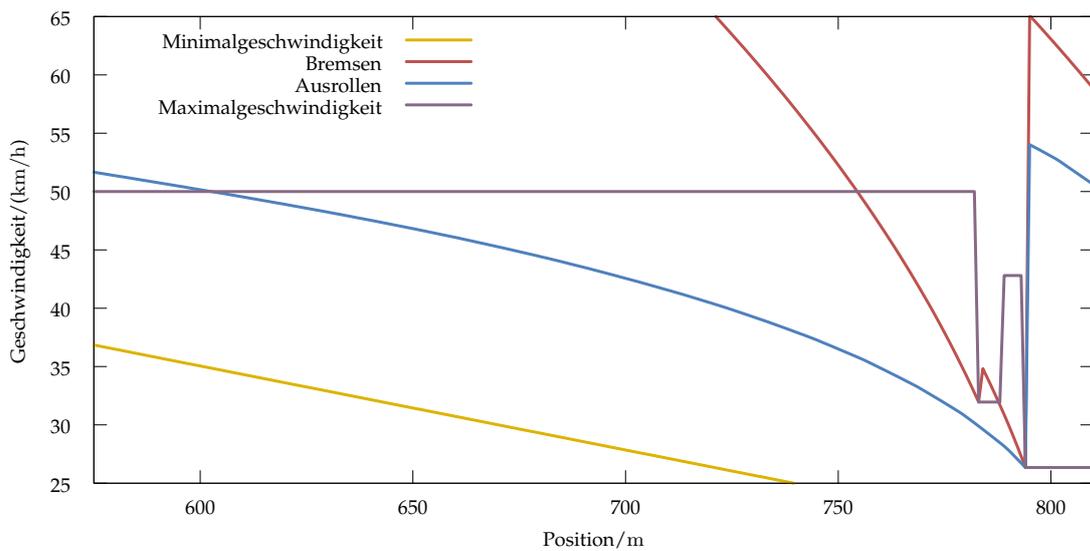
Benutzerstudien im Fahrsimulator und im realen Fahrversuch zeigen, dass mit Hilfe eines solchen hinweisgebenden System Kraftstoffersparnisse von 5 % bis zu 15 % erreicht werden können. Dabei verlängert sich die Fahrzeit lediglich um wenige Prozent (vergleiche [BKZ+11]).

Neben einer Darstellung im Kombi-Instrument des Fahrzeugs ist auch eine direkte Einblendung der Hinweise auf die Fahrbahn denkbar. Diese können z. B.

6 Dynamikvalidierung und -evaluierung



(a) Gesamte Strecke



(b) Ausschnitt Kurvenannäherung

Abbildung 6.7: Vorberechnetes Geschwindigkeitsprofil für ein Landstraßenszenario



Abbildung 6.8: Aktionshinweise als Augmentierung an die Straße

direkt im Sichtfeld des Fahrers durch ein Head-up-Display bzw. eine Augmented-Reality-Brille dargestellt werden [ZKB+14] oder in das Videobild einer nach vorne gerichteten Kamera eingeblendet werden. Abbildung 6.8 zeigt eine mögliche Darstellung als Augmentierung auf die Straße. Der Fahrhinweis wird in Form von ortsfesten farbigen Kreisen auf die Fahrbahn projiziert. Dazu wird die optimale Aktion in mehreren Abständen vor dem Fahrzeug ausgewertet und ein entsprechend farbiger Kreis in das Kamerabild projiziert.

In diesem Beispiel wurde die bereits bekannte Farbkodierung für die Aktionen verwendet. Als zusätzliche Aktion wird hier das Rekuperieren (grün) berücksichtigt. Diese Aktion wird vom Fahrzeug automatisiert ausgeführt, wenn der entsprechende Aktionshinweis angezeigt wird und der Fahrer das Fahrzeug ausrollen lässt. Im Gegensatz zum vorherigen Beispiel wird hier der Fahrer aktiv vom Fahrzeug bei der Fahraufgabe unterstützt.

Im Bild 6.8a ist ein Ausrollhinweis (blaue Kreise) zu sehen. Außerdem ist in einigen Metern Entfernung ein grüner Kreis zu erkennen. Der Fahrer kann somit leicht abschätzen, wann das Fahrzeug mit der Rekuperation beginnen wird. Im Bild 6.8b wird ein Bremshinweis gegeben. Am Verlauf der Kreise kann auch die zu fahrende Route abgelesen werden. In diesem Beispiel soll links abgebogen werden. Hier ist eine Kombination aus Navigationshinweisen und Fahraktionshinweisen möglich. Da die Darstellung mehr Platz benötigt, wurde diese wie in Bild 6.8c dargestellt im multi media interface (MMI) des Fahrzeugs integriert.

Eine weitere Anwendungsmöglichkeit ist die vollständige Automatisierung der Längsführung. Hierbei werden die Fahrhinweise direkt in Steuersignale für das Fahrzeug umgesetzt. Dies ermöglicht es, ein ACC auf die Anwendung im Innenstadtbereich zu erweitern, in dem neben anderen Fahrzeugen enge Kurven, Kreuzungen und Ampeln die Geschwindigkeit beeinflussen [KSL+13].

Im Gegensatz zu Autobahnen und Landstraßen sind im innerstädtischen Bereich die Kurvenradien deutlich kleiner und bei mehrspurigen Straßen auch abhängig von der gewählten Fahrspur. Daher wird spurgenaues Kartenmaterial benötigt. In der hier vorgestellten Anwendung wurden die Kartendaten mit Hilfe von Messfahrten basierend auf einem hochgenauen GPS erstellt. Für jede Fahrspur auf der zu fahrenden Route werden wie in der Anwendung zuvor Geschwindigkeitsprofile für die verschiedenen Aktionen erstellt. In dieser Anwendung lag der

6 Dynamikvalidierung und -evaluierung

Fokus lediglich auf dem komfortablen Fahren, so dass nur ein Profil, das einer komfortablen Bremsung entspricht, hinterlegt wurde.

Um auf dynamische Ampelschaltungen reagieren zu können, wird jedes Profil in einer zweiten Version vorgehalten, bei der von einem Anhalten an der Stopplinie der Ampel ausgegangen wird. Dieses Profil wird jeweils verwendet, wenn die nächste Ampel rot oder gelb zeigt. Um ein unnötiges, den nachfolgenden Verkehr gefährdendes, starkes Abbremsen kurz vor einer Ampel zu vermeiden, deren Zustand gerade auf gelb gewechselt hat, wird eine maximale Bremsbeschleunigung angenommen und das entsprechende Bremsprofil ebenfalls hinterlegt. Ist die aktuelle Geschwindigkeit größer als die von diesem Profil gegebene Geschwindigkeit so ist davon auszugehen, dass die Ampel vor kurzem ihren Zustand auf gelb gewechselt hat und es wird keine unnötige, starke Bremsung durchgeführt. Tests haben gezeigt, dass mit dieser Technik gelbe Ampeln ohne Bremsung passiert werden können, während an roten Ampeln zuverlässig angehalten wird.

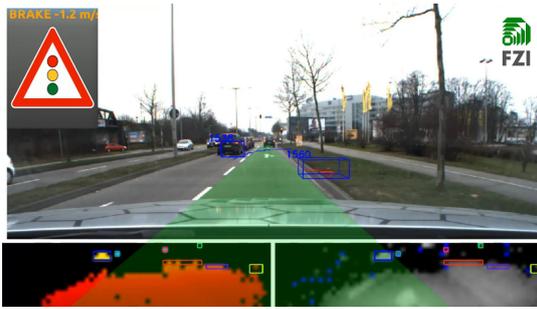
Durch die engen Kurvenradien ist eine Auswahl des relevanten vorausfahrenden Fahrzeugs für die Funktionalität des ACC nicht trivial lösbar. Im Gegensatz zu Autobahnscenarien reicht hier der geschätzte Fahrverlauf auf Basis des aktuellen Lenkwinkels nicht aus, um eine Zuordnung von vorausfahrenden Fahrzeugen zur eigenen oder zu Nachbarspuren zuverlässig durchführen zu können. Mit Hilfe der spurgenaue Karteninformation ist es jedoch möglich, Fahrzeuge im Fahrzeugumfeld den einzelnen Fahrspuren zuzuordnen und somit ihre Relevanz für die Funktionalität des ACC zu bestimmen.

Abbildung 6.9 zeigt exemplarisch drei Szenen aus einer Erprobungsfahrt der automatisierten Längsführung. In den Bildern sind jeweils verschiedene Aspekte der Szeneninterpretation und Aktionsgenerierung dargestellt. Unten links ist das Tiefenbild der verwendeten TOF-Kamera zu sehen. Rote Pixel stehen für nahe Messungen während blaue Pixel weiter entfernte Objekte repräsentieren. In schwarzen Bereichen ist die Reflektanz der Szene zu gering, um eine zuverlässige Tiefenmessung durchzuführen. Rechts daneben sind die Reflektanzwerte als Graustufenbild dargestellt. Farbige Boxen visualisieren das Ergebnis des angewendeten Segmentierungsverfahrens.

Im oberen Bereich ist das Videobild einer RGB-Kamera dargestellt, in das die segmentierten Objekte sowie die aus der Karte extrahierte aktuelle Fahrspur (grün) projiziert wird. Für die ACC-Funktionalität relevante Objekte sind rot dargestellt, unwichtige Objekte in blau. Oben links wird die jeweilige Aktion und der dazu gehörige Grund dargestellt.

Fazit

Das hier vorgestellte Verfahren eignet sich sehr gut zur Ableitung optimaler longitudinaler Fahraktionen in statischen Verkehrsszenen. Ist die Route des Fahrzeugs bekannt, so kann seine Geschwindigkeit auf statische Geschwindigkeitsbeschränkungen z. B. durch Kurven mittels optimaler Aktionen angepasst werden. Durch



(a) Verzögerung auf eine Ampel



(b) ACC Funktionalität auf ein vorausfahrendes Fahrzeug. Das relevante Fahrzeug wird durch einen roten Rahmen markiert.



(c) Verzögerung aufgrund einer Kurve. Das Fahrzeug vor dem Ego-Fahrzeug wird mit Hilfe der Spurinformati- on als nicht relevant für die ACC Funktionalität eingestuft.

Abbildung 6.9: Drei exemplarische Situationen der automatisierten Längsführung im innerstädtischen Bereich

Erstveröffentlichung in [KSL+13] (Kohlhaas, Schamm, Lenk et al.) © 2013 IEEE

die Bereitstellung einer Policy für alle Geschwindigkeitsbereiche entlang der Strecke kann das Verfahren auch für die Fahrerassistenz eingesetzt werden, bei dem die Geschwindigkeit des Fahrzeugs maßgeblich vom Fahrverhalten des Fahrers abhängt.

Eine Erweiterung des Verfahrens auf semi-statische Geschwindigkeitsbeschränkungen wie Ampeln ist durch die Verwendung mehrerer Profile möglich. Auch kann dieses Verfahren gut mit anderen Geschwindigkeit beschränkenden Funktionen wie einem ACC verknüpft werden.

6.2.4 Modellprädiktion

Häufig ist die Verkehrsszene nicht statisch oder nicht so präzise prädizierbar wie bei der Anfahrt an Ampeln mit bekannter Schaltzeit. In diesen Fällen stoßen

Verfahren, die eine Policy vorberechnen, an die Grenzen der Berechenbarkeit. Hier können reaktive Verfahren zum Einsatz kommen.

Die modellprädiktive Regelung (MPC) ist ein sehr reaktives Verfahren aus dem Bereich der Regelungstechnik. Das Verfahren kombiniert Ansätze aus der Planung mit der Reaktivität einer Regelung. Dabei werden für den aktuellen Systemzustand verschiedene mögliche zukünftige Verhalten auf Basis verschiedener Eingabegrößen prädiziert und bewertet. Die Eingabegröße, die den besten zukünftigen Verlauf verspricht, wird dann als Steuergröße im aktuellen Zeitschritt verwendet.

Dieses Verfahren kann auch für das Finden von lokal optimalen Geschwindigkeitsprofilen eingesetzt werden. Häufig wird eine große Zahl von möglichen Steuergrößen und somit Aktionen berücksichtigt. Wie in [KSN+11] gezeigt, kann für spezielle Szenarien bei geschickter Wahl auch eine stark eingeschränkte Menge von Aktionen zu sehr guten Ergebnissen führen. Dieser Ansatz wird im Folgenden kurz vorgestellt.

Beispiele und Anwendung

In diesem Beispiel wird eine effiziente automatisierte Längsführung unter Berücksichtigung von Überholmöglichkeiten vorausfahrender Fahrzeuge mittels samplingbasierter MPC umgesetzt. Dazu werden drei Aktionen für das MPC verwendet: Ausrollen, Bremsen, Normalfahrt (Beschleunigen auf Sollgeschwindigkeit, anschließend Konstantfahrt). Bei der Anfahrt auf langsamere vorausfahrende Fahrzeuge spielt die zeitliche Komponente der Manöverbewertung keine Rolle, solange das Fahrzeug nicht überholt werden kann. Daher gestaltet sich die Bewertung der drei Aktionen relativ einfach. Solange kein vorausfahrendes Fahrzeug die Fahrt behindert ist die Normalfahrt zu bevorzugen. Eine Behinderung durch ein vorausfahrendes Fahrzeug ist dann gegeben, wenn durch die Aktion Ausrollen der Sicherheitsabstand zum vorausfahrenden Fahrzeug verletzt würde. Ist dies der Fall und liegt keine kritische Situation vor, ist die Aktion Ausrollen zu wählen. Ist die Situation kritisch, würde also selbst durch Bremsen der Sicherheitsabstand zum vorausfahrenden Fahrzeug verletzt, muss gebremst werden.

Um die Aktionen bewerten zu können, wird jeweils ausgehend vom aktuellen Fahrzeugzustand das Verhalten des Fahrzeugs für die Aktionen Ausrollen und Bremsen bis zum Erreichen des Sicherheitsabstands oder der Geschwindigkeit des vorausfahrenden Fahrzeugs prädiziert.

Abbildung 6.10 zeigt den Kraftstoffverbrauch eines Fahrzeugs während einer simulierten Fahrt auf einer Bundesstraße. Nach einer Fahrstrecke von rund 500 m wird das Fahrzeug durch ein vorausfahrendes, langsames Fahrzeug behindert. Bei 3200 m ergibt sich die Möglichkeit zu überholen. Dargestellt ist der Verbrauch eines Fahrers, der vor Erreichen des langsameren Fahrzeugs abbremst und mit geringem Sicherheitsabstand folgt, sowie der eines assistierten Fahrers, der entsprechend dem MPC frühzeitig ausrollen lässt. Es ist gut zu erkennen, wie durch

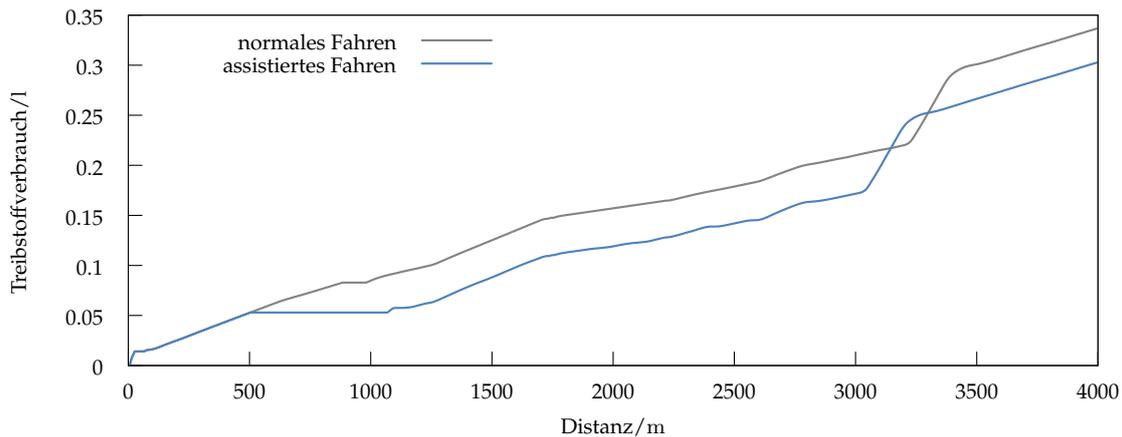


Abbildung 6.10: Anfahrt auf ein langsames vorausfahrendes Fahrzeug: Vergleich einer Fahrt ohne Assistenz mit einer Fahrt optimiert durch MPC.

Ähnlich veröffentlicht in [KSN+11] (Kohlhaas et al.)

das frühzeitige Ausrollen in der assistierten Fahrt Kraftstoff eingespart werden kann. Wie erwähnt hat dies keine Auswirkungen auf die Fahrzeit, da der Fortschritt auf der Strecke durch das vorausfahrende Fahrzeug limitiert wird.

Auch beim anschließenden Überholen können Methoden des MPC zum Einsatz kommen. Im oben gezeigten Beispiel werden während der Folgefahrt die Aktionen Folgen und Normalfahrt betrachtet, um einen optimalen Zeitpunkt zum Beschleunigen vor einer Überholmöglichkeit abzupassen. Dies führt im Beispiel zu einem etwas früheren Überholen und zu einem leichten Zeitgewinn, jedoch auch zu einem geringen Kraftstoffmeherverbrauch. Dieser fällt jedoch gegenüber den Einsparungen bei der Annäherung vernachlässigbar klein aus. Über die gesamte Strecke konnte durch Einsatz des auf MPC basierenden hinweisgebenden Systems eine Kraftstoffersparnis von gut 10 % erreicht werden.

Fazit

MPC ist reaktives Verfahren, das Planungskomponenten in die Regelung eines Systems integriert. Bei der Optimierung von Geschwindigkeitsprofilen ist es aufgrund des Regelungsansatzes gut geeignet, wenn die Komplexität der Szene vor allem durch dynamische oder schwer prädizierbare Elemente bestimmt wird. Durch den eingeschränkten Planungshorizont kann das Verfahren in komplexen Szenen versagen und keine korrekte Lösung bereitstellen. Obwohl keine globale Policy errechnet wird, ist das Verfahren für den Einsatz in Assistenzsystemen geeignet, da es schnell auf Abweichungen in der Ausführung der Hinweise reagieren kann. Auch ist das Verfahren gut geeignet, eine kleine, diskrete Menge von Aktionen zu verarbeiten. Schnelles Alternieren von Hinweisen kann auftreten. Dies lässt sich jedoch durch entsprechende Hysteresen bei der Aktionsauswahl verhindern.

6.2.5 Sichtgraphenverfahren

Zur Geschwindigkeitsprofiloptimierung kann auch das in Kapitel 2.9.2 vorgestellte Sichtgraphenverfahren [JH13] eingesetzt werden. Der Pfad-Geschwindigkeit-Zeit-Planer (PVT-Planer) erzeugt nicht direkt ein Geschwindigkeitsprofil, sondern einen Pfad in der Ort-Zeit-Ebene. Aus diesem kann dann durch Bildung der Ableitung ein Geschwindigkeitsprofil abgeleitet werden.

Das Verfahren ist auf Szenen spezialisiert, in denen die Fahrt durch dynamische Objekte beschränkt wird. Diese Objekte können in die Ort-Zeit-Ebene projiziert werden und bilden dort Polygone. Ein notwendiger Sicherheitsabstand zu den Objekten sowie die Größe des eigenen Fahrzeugs muss bei der Erstellung der Polygone berücksichtigt werden.

Zwischen diesen Polygonen wird dann ein Sichtgraph zur Planung aufgebaut. Bei diesem wird bereits die Dynamik des Fahrzeugs berücksichtigt, so dass es nur Verbindungen zwischen Punkten gibt, die unter Berücksichtigung der maximalen Beschleunigungswerte erreichbar sind. Auf dem Sichtgraph kann eine optimale Trajektorie ermittelt werden. Die Trajektorie weist eine stückweise konstante Beschleunigung auf, so dass sie effizient mittels Tupeln aus Position, Geschwindigkeit und Zeit ($x_i = (p_i, v_i, t_i)$) beschrieben werden kann. Die konstante Beschleunigung auf einem Teilabschnitt lässt sich direkt aus den jeweiligen Begrenzungspunkten errechnen.

Beispiele und Anwendung

Für dieses Verfahren werden in [JH13] zwei Anwendungen vorgestellt: eine automatisierte Fahrt durch eine belebte Straße und ein Assistenzsystem, bei dem der Fahrer vor Kollisionen mit anderen Fahrzeugen gewarnt wird. Für das Beispiel des automatisierten Fahrens werden für einen geplanten Pfad alle kreuzende Hindernisse wie Fußgänger, Radfahrer und Autos in die Ort-Zeit-Ebene eingetragen und ein Geschwindigkeitsprofil geplant. Bei der Anwendung als Assistenzsystem wird fortlaufend unter Berücksichtigung aller Hindernisse ein Geschwindigkeitsverlauf auf dem geplanten Pfad berechnet. Wird kein gültiger Geschwindigkeitsverlauf mehr gefunden, so wird eine Warnung ausgegeben. In diesem Fall ist eine Kollision bei Einhaltung des Pfads unvermeidbar. In beiden Beispielen werden keine Spurwechsel des Ego-Fahrzeugs berücksichtigt. Durch die Projektion der Hindernisse auf den geplanten Pfad funktionieren beide Systeme jedoch auch in Kreuzungssituationen.

Fazit

Das Verfahren ermöglicht die Planung eines Geschwindigkeitsprofils auf einem gegebenen Pfad mit kreuzenden bzw. ein- und ausscherenden Fahrzeugen. Randbedingungen an das Profil können durch Polygone in der Ort-Zeit-Ebene definiert

werden. Eine Beschränkung der Maximalgeschwindigkeit sowie der Maximalbeschleunigungen ist nur global möglich. Als Aktionen werden Beschleunigen und Abbremsen mit stückweise konstanter Beschleunigung sowie Konstantfahren berücksichtigt. Eine Optimierung erfolgt hinsichtlich des Zeitbedarfs.

Das Sichtgraphverfahren ist vollständig und optimal. Wenn eine Lösung für das Planungsproblem existiert, wird diese auch gefunden. Die gefundene Lösung ist hinsichtlich des betrachteten Optimierungskriteriums optimal. Durch die Verwendung eines Sichtgraphen und die geringe Komplexität der Berechnung der Sichtlinien, ist das Verfahren sehr schnell. Es eignet sich somit für die Anwendung beim automatisierten Fahren sowie für die Fahrerassistenz.

6.2.6 Zusammenfassung

Die hier vorgestellten Verfahren unterscheiden sich in ihren Eigenschaften und somit auch in ihren Anwendungsmöglichkeiten. In Tabelle 6.1 sind ihre Eigenschaften dargestellt und hinsichtlich ihrer Eignung für den Einsatz der Dynamikvalidierung im Rahmen der semantischen Planung bewertet. Die ersten drei Eigenschaften entsprechen den in Kapitel 3.4.4 abgeleiteten Anforderungen und sind somit als sehr wichtig zu bewerten.

Hier ist klar erkennbar, dass das Sichtgraphverfahren unter Annahme der ausgewählten Kriterien am besten für die Validierung und Bewertung semantischer Manöverketten geeignet ist. Einschränkungen gibt es bei den möglichen Szeneigenschaften, da hier die Betrachtung von verschiedenen Tempolimits in der verwendeten Implementierung nicht möglich ist. Es kann pro Szene nur eine Höchstgeschwindigkeit angegeben werden. Da die Dynamikvalidierung in der semantischen Manöverplanung lediglich für Zeithorizonte von wenigen Sekunden relevant ist, kann das Geschwindigkeitsprofil der zulässigen Höchstgeschwindigkeiten im entsprechenden Streckenintervall durch das Maximum konstant approximiert werden. Ob stückweise Definitionen von Höchstgeschwindigkeiten konzeptionell behandelbar sind, wurde daher in dieser Arbeit nicht untersucht.

Auch bei der Wahl der Kostenfunktion ist das Verfahren eingeschränkt. Es wird lediglich eine Kostenfunktion unterstützt, die die Fahrzeit berücksichtigt. Für die Dynamikvalidierung ist dies jedoch vollkommen ausreichend. Eine Bewertung des resultierenden Energieverbrauchs kann dennoch auf Basis des Geschwindigkeitsprofils vorgenommen werden und so eine Indikation für eine energetische Bewertung der Manöversequenzen geben. Eine energieoptimale Fahrweise innerhalb der Manöversequenz ist bei Bedarf eine Anforderung an den nachgelagerten Trajektorienplaner.

Das Sichtgraphverfahren ist somit für die Dynamikvalidierung von Manöverketten geeignet. Daher wird dieses Verfahren in das Gesamtsystem integriert. Die dazu nötigen Schritte werden im Folgenden beschrieben.

6 Dynamikvalidierung und -evaluierung

Kriterium	Dynamisches Programmieren	Modellbasierte Optimierung	Vorberechnete Profile	Modellprädiktion	Sichtgraphverfahren
Vollständigkeit	-	-	-	-	+
geringe Laufzeit	-/+ ¹	+	-/+ ²	+	+
global optimal	(+) ³	-	-	-	+
kontinuierlicher Aktionsraum	-	-	-	-	+
kontinuierlicher Zustandsraum	-	+	+/- ⁴	+	+
dynamische Hindernisse	+	+	-	+	+
Geschwindigkeitsbeschränkungen	+	+	+	+	o ⁵
Kostenfunktionen	+	+	o	+	o ⁶

- ¹ Die Berechnung der Policy ist sehr aufwändig. Die Ableitung von Aktionen in einem vorberechneten Szenario ist sehr schnell möglich.
- ² Die Vorbereitung der Geschwindigkeitsprofile ist aufwändig, kann jedoch für statische Szenen offline erfolgen. Die Ableitung von Aktionen ist sehr schnell möglich.
- ³ Aufgrund von Diskretisierung und Diskretisierungsfehlern wird das Optimum nicht erreicht.
- ⁴ Hybrider Zustandsraum mit diskreter Position und kontinuierlicher Geschwindigkeit.
- ⁵ Verschiedene Geschwindigkeitsbeschränkungen sind in der Implementierung [JH13] nicht vorgesehen.
- ⁶ Die Implementierung unterstützt lediglich zeitoptimale Geschwindigkeitsprofile.

Tabelle 6.1: Bewertung der Verfahren zur Dynamikvalidierung

6.3 Dynamikvalidierung von semantischen Plänen

Um eine semantische Manöverkette auf ihre dynamische Fahrbarkeit hin zu überprüfen und zu bewerten, soll für eine Manöverkette ein optimales Geschwindigkeitsprofil abgeleitet werden. Eine Manöverkette besteht zunächst aus semantischen Zuständen, die keine direkte geometrische Interpretation zulassen. Das ausgewählte Verfahren zur Geschwindigkeitsprofilvalidierung benötigt jedoch eine Repräsentation der Szene als eine geo-temporalen Darstellung von Hindernisse in einer Ort-Zeit-Ebene. Die semantische Zustandskette muss also in eine solche überführt werden.

Die Geschwindigkeitsprofilvalidierung kann auf finalen Plänen der semantischen Manöverplanung angewendet werden oder auch während des Planungsprozesses zur Validierung von Teilplänen eingesetzt werden (siehe Kapitel 6.3.2).

6.3.1 Semantischer Plan als Ort-Zeit-Schlauch

Über die zeitlichen Eigenschaften der Konstellationen und der die Zustände beschränkenden Transitionen ist eine Ableitung einer geo-temporalen Interpretation eines semantischen Zustands möglich. Wie in Kapitel 4.3 dargestellt, werden die semantischen Zustände aus einer geometrischen Repräsentation abgeleitet. Es ist daher möglich jedem semantischen Zustand geometrische Informationen zuzuordnen, die seine Ausdehnung auf der Ort-Zeit-Ebene definieren.

Jeder semantische Zustand s ist genau einer Konstellation k zugeordnet und ist demnach zeitlich eindeutig auf das Intervall $[t_{Start}(s) : t_{Ende}(s)]$ begrenzt. Die Ausdehnung des Zustands entlang der Ortsachse kann aus den dem Zustand zugeordneten Relationen *fährt auf*(\mathbf{U}) sowie den longitudinalen Relationen zu anderen Verkehrsteilnehmern abgeleitet werden. Jede longitudinale Relation r_i ($i \in \{1, 2 \dots n\}$) definiert zwei Grenzfunktionen f_i^{lb} und f_i^{ub} . Auch die Relation *fährt auf*(\mathbf{U}) definiert diese Grenzfunktionen f_0^{lb} und f_0^{ub} als statische Funktionen über die Zeit (vergleiche Kapitel 4.3.3). Die Ausdehnung des semantischen Zustands kann somit bestimmt werden als der Schnitt über alle Intervalle $[f_i^{lb} : f_i^{ub}]$.

$$[f^{lb} : f^{ub}] = \bigcap_{i \in \{0, 1 \dots n\}} [f_i^{lb} : f_i^{ub}] \quad (6.3)$$

Dies ist wohldefiniert, da für das durch die Konstellation gegebene Intervall $[t_{Start}(s) : t_{Ende}(s)]$ die Grenzfunktionen eine absolute Ordnung aufweisen. Als weitere Einschränkung kann die Fahrzeugposition als untere Schranke verwendet werden, um Rückwärtsbewegungen zu vermeiden.

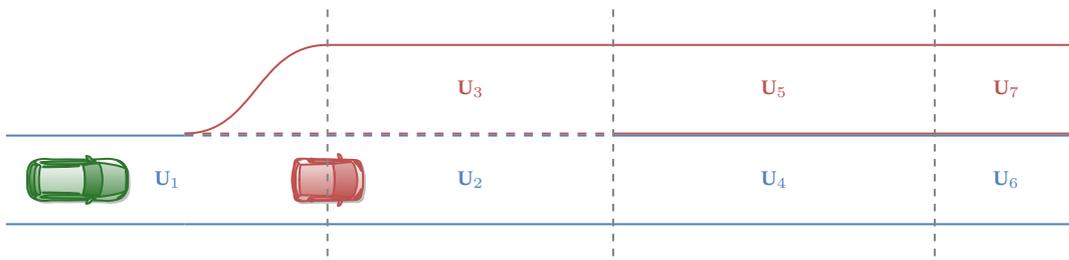
Werden als Grenzfunktionen lineare Funktionen verwendet, so kann einem semantischen Zustand eine trapezförmige Fläche zugeordnet werden. Die Eckpunkte dieser Fläche berechnen sich als:

$$\begin{aligned} x_1 &= (f^{lb}(t_{Start}), t_{Start}) \\ x_2 &= (f^{ub}(t_{Start}), t_{Start}) \\ x_3 &= (f^{lb}(t_{end}), t_{end}) \\ x_4 &= (f^{ub}(t_{end}), t_{end}) \end{aligned} \quad (6.4)$$

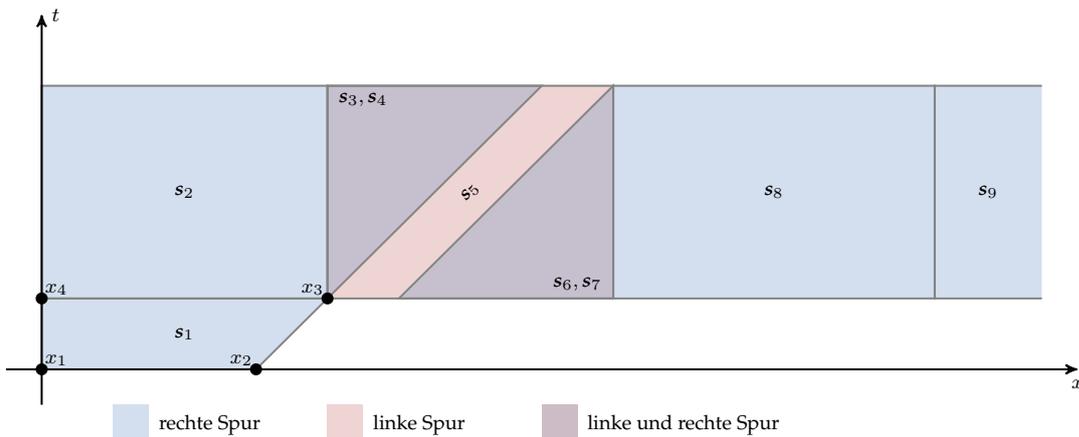
Ein Beispiel hierzu wird in Abbildung 6.11 gegeben. Im oberen Bereich ist nochmals das Bundesstraßenszenario dargestellt. Hier wurde das aus Abbildung 4.7 bekannte Szenario um die Straßensegmente \mathbf{U}_6 und \mathbf{U}_7 erweitert. Die beiden Fahrspuren sind farblich markiert. Analog zum in Abbildung 5.1 dargestellten Plan existiert auch hier eine Manöverkette zum Überholen des anderen Fahrzeugs. Aufgrund des längeren Szenarios ist diese auch um einen Zustand s_9 , der das Erreichen von \mathbf{U}_6 darstellt, erweitert.

In Abbildung 6.11b ist die Projektion dieses Plans auf die Ort-Zeit-Ebene dargestellt. Für den Zustand s_1 sind die Eckpunkte x_1 bis x_4 gekennzeichnet. Die Flächen

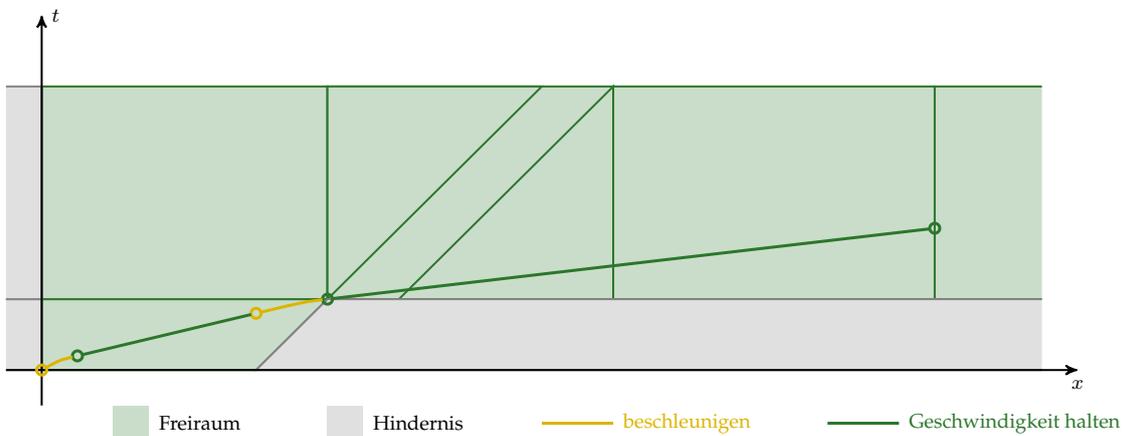
6 Dynamikvalidierung und -evaluierung



(a) Schematische Darstellung des Bundesstraßenszenarios. Die rechte Spur ist grün, die linke rot dargestellt.



(b) Projektion der Erweiterung des semantischen Plans aus Abbildung 5.1 als Freiraum in der Ort-Zeit-Ebene. Die Zustände des Plans ($s_1 - s_9$) werden durch Vierecke repräsentiert wie in Formel (6.4) dargestellt.



(c) Konvertierung des Freiraums (grün) in Hindernisse (grau) und zeitoptimale Trajektorie des PVT-Planers.

Abbildung 6.11: Dynamikvalidierung eines semantischen Plans mit Hilfe des PVT-Planers

Ähnlich veröffentlicht in [KHS+15] (Kohlhaas, Hammann et al.)

der Zustände, die zu Fahrspursegmenten auf der linken Spur korrespondieren, sind rot gefärbt, die der rechten Spur blau. Bei einigen Zuständen (z. B. s_3) ist das Trapez zu einem Dreieck degeneriert. Bei geplanten Spurwechseln kommt es zu Überlappungen mehrerer Zustände. Aufgrund der topologischen Eigenschaften des semantischen Zustandsraums und der Planungsverfahren wird die gesamte Manöverkette auf eine zusammenhängende Fläche auf der Ort-Zeit-Ebene abgebildet. Diese kann als Freiraum für eine anschließende Geschwindigkeitsprofiloptimierung und somit zur Dynamikvalidierung verwendet werden.

Um das Sichtgraphverfahren zur Trajektoriengenerierung auf den Freiraum anwenden zu können, muss dieser durch angrenzende Hindernisse beschrieben werden. Dazu wird für jedes Konstellations-Zeitintervall ein linkes und ein rechtes beschränkendes Hindernis generiert. Der Freiraum sowie die beschränkenden Hindernisse des Beispielszenarios sind in Abbildung 6.11c dargestellt.

Mittels der aktuellen Fahrzeugposition und -geschwindigkeit als Ausgangspunkt und der linken Seite des Zielzustands (im Beispiel s_9) als Zielposition kann eine Trajektorienuche mit dem Sichtgraphverfahren durchgeführt werden. Diese ermittelt auf dem durch die Hindernisse definierten Ort-Zeit-Schlauch ein optimales Geschwindigkeitsprofil in Form einer Trajektorie in der Ort-Zeit-Ebene. Eine beispielhafte Trajektorie ist in Abbildung 6.11c dargestellt.

Das Ergebnis der Geschwindigkeitsprofilplanung kann dann als Bewertung und Validierung des semantischen Plans verwendet werden. Existiert z. B. kein Geschwindigkeitsprofil, so ist die geplante Manöverkette dynamisch nicht fahrbar und kann verworfen werden. Die zeitliche Position des letzten Profilverpunkts, also die zeitliche Ausdehnung des Profils $t(\mathbf{p})$, kann als Bewertung des Plans $g(\mathbf{p}) = t(\mathbf{p})$ für eine fahrzeitoptimale Lösung genutzt werden.

6.3.2 Einbindung in Heuristik-Suche

Wie bereits in Kapitel 5.1 angesprochen kann die Geschwindigkeitsprofilvalidierung auch auf semantische Teilpläne angewendet und somit schon während der Suche eingesetzt werden. Bei der A*-Suche wird eine Kostenfunktion pro Zustand $f(s)$ verwendet bestehend aus den Kosten bis zu diesem Zustand $g(s)$ und einer Schätzung (Heuristik) der Kosten von diesem bis zum Zielzustand $h(s)$ (vergleiche Formel (2.9)). Durch die Geschwindigkeitsprofilvalidierung von Teilplänen ist es möglich, die Kosten $g(s)$ (Zeitbedarf) bis zu einem Zustand genau zu bestimmen. Auch können, wenn kein gültiges Geschwindigkeitsprofil bei der Validierung gefunden wird, die Kosten auf einen sehr hohen Wert gesetzt werden oder der Plan nicht weiter expandiert werden.

Entgegen der Annahme bei der A*-Suche sind die Kosten, die durch die Geschwindigkeitsprofilvalidierung bestimmt werden, nicht nur vom Zustandsübergang und den Kosten des Vorgängerzustands sondern auch vom genauen Plan bis zu diesem abhängig. Gleichung (2.10) ist hier also nicht gültig. Dies führt dazu, dass

A*S in gewissen Situationen einen gültigen Pfad nicht finden kann, wenn es für einen Teilpfad eine günstigere Alternative gibt, die jedoch zu keinem gültigen Pfad gehört.

Um in allen Situationen korrekte Ergebnisse zu erhalten, wird ein weiteres Suchverfahren eingeführt, das die A*-Suche für die Verwendung der Geschwindigkeitsprofilvalidierung in der Suche anpasst. Ähnlich der Erweiterung des Suchverfahrens von BFS zu EBFS wird auch hier statt einer Liste zu expandierender Knoten eine Liste von Teilplänen verwendet. Verschiedene Teilpläne können teilweise die selben Knoten beinhalten und insbesondere auch im selben Knoten enden. Somit wird keine Bewertung der Knoten wie in Gleichung (2.9) sondern eine Bewertung der Teilpläne \mathbf{p}_i benötigt:

$$f(\mathbf{p}_i) = g(\mathbf{p}_i) + h(\mathbf{p}_i) \quad (6.5)$$

Die Kosten des Teilplans sind analog zu den Kosten eines Plans definiert als $g(\mathbf{p}_i) = t(\mathbf{p}_i)$ und können somit direkt aus der Geschwindigkeitsprofilvalidierung bestimmt werden. Als Heuristik der noch ausstehenden Kosten kann die Zeitdifferenz zwischen $t(\mathbf{p}_i)$ und dem Beginn der Zielkonstellation t_z angenommen werden. Wird die Maximalgeschwindigkeit des Fahrzeugs v_{\max} und die zu fahrende Reststrecke d_i in die Heuristik mit einbezogen, so lässt sich diese noch weiter präzisieren:

$$h(\mathbf{p}_i) = \max \left(t_z - t(\mathbf{p}_i), \frac{d_i}{v_{\max}} \right) \quad (6.6)$$

Der hieraus resultierende Suchalgorithmus wird im Folgenden analog zu den anderen vollständigen Suchverfahren vollständige A*-Suche (engl.: exhaustive A*-search, EA*S) genannt.

6.4 Evaluation

Das oben beschriebene Verfahren zur Dynamikvalidierung der semantischen Pläne wird im Folgenden an ausgewählten Planungsergebnissen evaluiert. Dabei werden zunächst detaillierte qualitative Ergebnisse anhand eines Beispiels dargestellt. Dabei wird vor allem auch die Funktionsweise der neu eingeführten EA*S beleuchtet. Wichtig für die Anwendbarkeit der Algorithmen in realen Verkehrssituationen ist das Laufzeitverhalten, das für die verschiedenen Planungsansätze evaluiert wird. Abschließend werden die positiven Auswirkungen der Dynamikvalidierung auf die Beispielanwendung des Überholassistenten dargestellt.

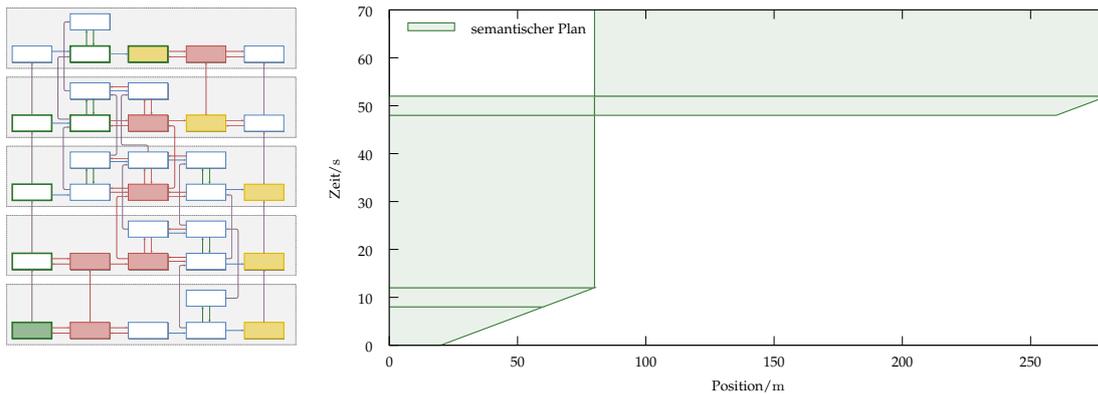


Abbildung 6.12: Bundesstraßenszenario - Projizierter semantischer Plan aus der EBFS. Die Zustände (grün) des semantischen Plans (links) werden in die Ort-Zeit-Ebene projiziert (rechts) und bilden dort einen Fahrerschlauch.

6.4.1 Dynamikvalidierung von semantischen Plänen

Die Projektion eines semantischen Plans in die Ort-Zeit-Ebene wird nun beispielhaft an einem semantischen Plan für das Bundesstraßenszenario, der mit Hilfe der EBFS erstellt wurde, gezeigt (vergleiche Abbildung 5.6c).

Abbildung 6.12 zeigt links erneut den semantischen Zustandsraum und den durch EBFS erzeugten Plan in grün. Auf der rechten Seite ist das Ergebnis der Projektion der Zustände des Plans auf die Ort-Zeit-Ebene dargestellt. Jedem Zustand des Plans wird dabei entsprechend seiner zugehörigen Konstellation und seiner ihn beschränkenden longitudinalen Transitionen ein Vier- bzw. Dreieck zugeordnet. Konstellationen sind, wie in Kapitel 4.3.3 dargestellt, für ein bestimmtes Zeitintervall gültig. Diese Intervalle sind in der Ort-Zeit-Ebene als horizontale Strukturen gut zu erkennen. Die durch Zustände belegte Fläche bildet einen möglichen Fahrerschlauch, in dem ein Geschwindigkeitsprofil geplant werden kann.

Um die in Kapitel 6.2.6 ausgewählte Geschwindigkeitsprofilplanung anzuwenden, wird ein Hindernisraum benötigt, der sich als Komplement des Fahrerschlauchs ergibt. Die Konvertierung des Freiraums in den Hindernisraum ist in Abbildung 6.13 visualisiert. Objekte sind dabei in grau, der Freiraum in weiß dargestellt. Im konkreten Beispiel wurden zwei Objekte erzeugt. Eines (unten rechts) beinhaltet unter anderem den durch das andere Fahrzeug blockierten Bereich. Das Objekt oben links repräsentiert einen durch den Planer nicht genutzten Zustand.

Auf dieser Hindernisdarstellung kann dann der PVT-Planer angewendet werden. Das Ergebnis dieser Geschwindigkeitsprofilplanung ist in Abbildung 6.14 dargestellt. Es ist deutlich zu erkennen, dass der hier beispielhaft gewählte Plan in einem nur suboptimalen Geschwindigkeitsprofil resultiert. Das Profil beginnt mit einem starken Bremsvorgang (rot) und beinhaltet daher eine relativ lange Beschleunigungsphase (gelb). Auch wird das Ende des Szenarios erst bei etwas

6 Dynamikvalidierung und -evaluierung

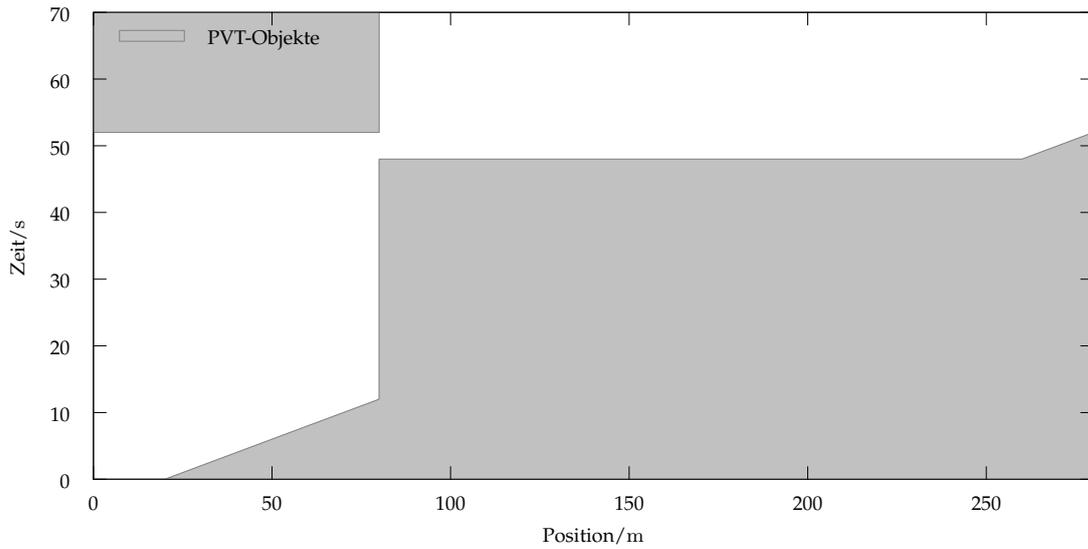


Abbildung 6.13: Bundesstraßenszenario - Konvertierung des Fahrschlauchs in PVT-Objekte

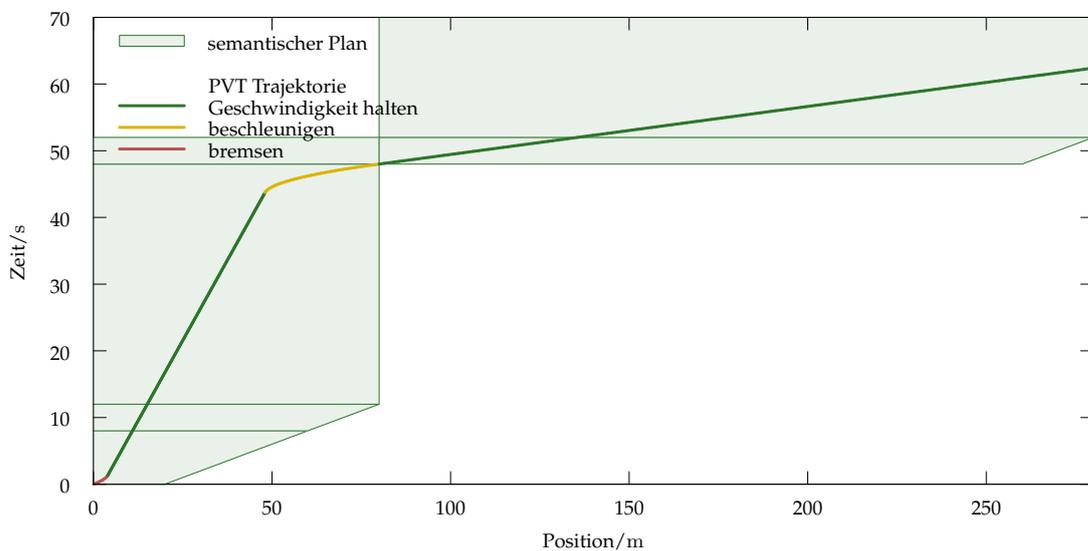


Abbildung 6.14: Bundesstraßenszenario - Optimales Geschwindigkeitsprofil im projizierten Fahrschlauch mittels PVT-Planer

mehr als einer Minute Fahrzeit erreicht. Um den optimalen Plan zu finden, ist bei Einsatz des BFS eine dynamische Validierung aller Pläne notwendig.

Beispielhaft ist in Abbildung 6.15 die dynamische Validierung aller 8 durch BFS gefundenen, semantischen Pläne dargestellt. Die Reihenfolge der Pläne ist so wie in Abbildung 5.6 gewählt. In diesem speziellen Szenario wird für jeden semantischen Plan auch ein fahrbares Geschwindigkeitsprofil gefunden. Dies ist im Allgemeinen jedoch nicht immer der Fall. Wird kein fahrbares Geschwindigkeitsprofil gefunden, so ist der semantische Plan definitiv nicht ausführbar. Dies wird durch die Eigenschaft des PVT-Planers vollständig zu sein sicher gestellt.

Wie zu erwarten ist, weist der Plan, bei dem das Fahrzeug überholt wird, die kleinste Fahrzeit von 26,4 s auf (Abbildung 6.15a). Ohne Überholmanöver liegt die optimale Fahrzeit bei 52,0 s.

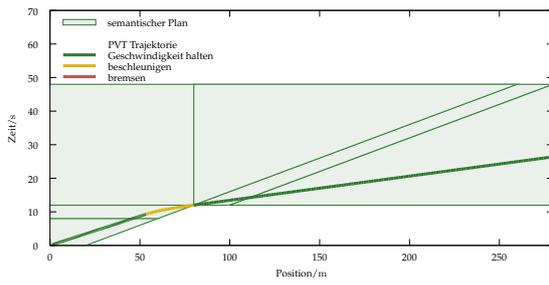
6.4.2 Integrierte Dynamikvalidierung - EA*S

Durch eine kontinuierliche Dynamikvalidierung von Teilplänen zur Planungszeit, wie sie von EA*S durchgeführt wird, kann die Anzahl der gefundenen Pläne verringert werden, ohne die optimale Lösung zu verlieren. Im hier gezeigten Beispielszenario werden nur zwei Pläne gefunden. Abbildung 6.16 und 6.17 sowie 6.18 und 6.19 zeigen jeweils die Projektion der semantischen Pläne, den korrespondierenden Hindernisraum sowie das optimale Geschwindigkeitsprofil.

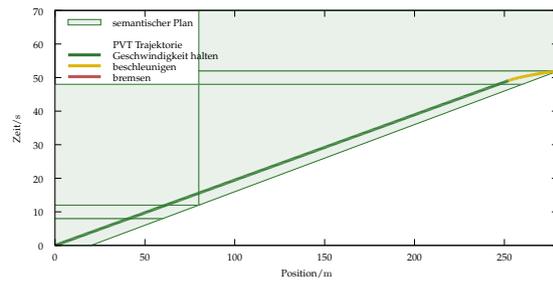
Die zwei Pläne entsprechen den Plänen (a) und (b) des EBFS. Wie aus dem Vergleich der Fahrzeiten ersichtlich ist, wurde jeweils der zeitoptimale Plan mit und ohne Überholmanöver gefunden. Die Pläne (e), (f) und (g) sind zwar auch zeitoptimal, weisen jedoch unnötige Spurwechsel auf. Diese werden von EA*S aufgrund der größeren Pfadlänge nicht ausgewählt.

Sowohl bei den Plänen der EBFS als auch bei denen mit EA*S ist in den Fällen ohne Überholmanöver am Ende des resultierenden Geschwindigkeitsprofils eine Beschleunigung zu erkennen. Unter Kenntnis der Szene ist hier zu erwarten, dass das Ego-Fahrzeug nach Ende des betrachteten Zeithorizonts den Sicherheitsabstand zum vorausfahrenden Fahrzeugs verletzt wird und es ggf. sogar zu einer Kollision kommen könnte. Aufgrund des beschränkten Zeithorizonts ist für den PVT-Planer dieser Zusammenhang jedoch nicht zu erkennen. Hier wird ersichtlich, dass für die sichere Ausführung eines Planes, der geplante Zeithorizont den Zeithorizont der Ausführung übersteigen muss. Das heißt, dass auch hier, ähnlich wie bei anderen Planungsverfahren, mit ausreichendem Abstand vor Erreichen eines Planendes ein neuer Plan erstellt werden muss.

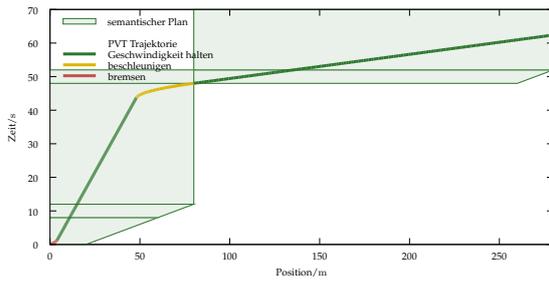
6 Dynamikvalidierung und -evaluierung



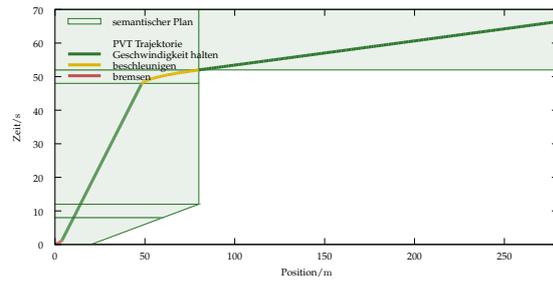
(a) Dauer: 26,4 s



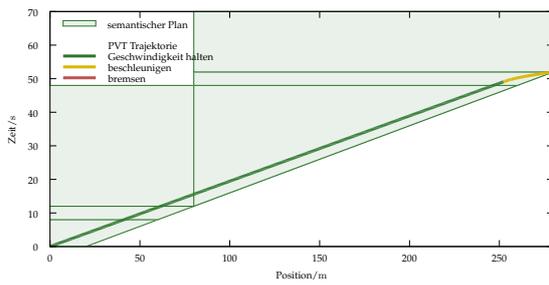
(b) Dauer: 52,0 s



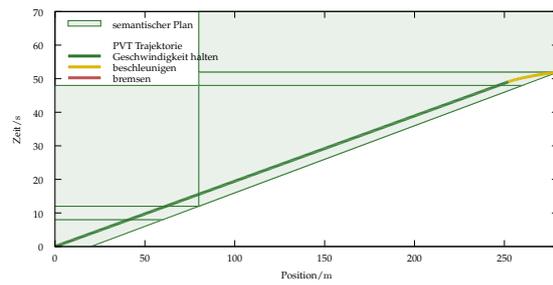
(c) Dauer: 62,4 s



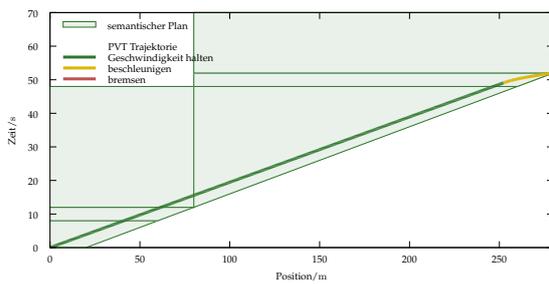
(d) Dauer: 62,4 s



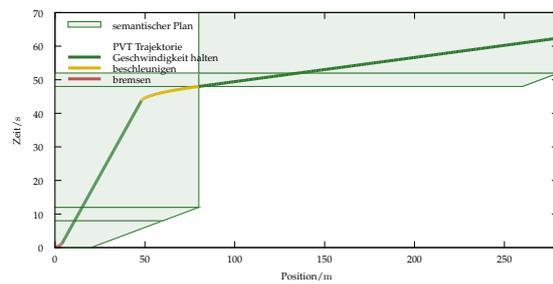
(e) Dauer: 52,0 s



(f) Dauer: 52,0 s



(g) Dauer: 52,0 s



(h) Dauer: 62,4 s

Abbildung 6.15: Bundesstraßenszenario - Dynamikvalidierung der acht mittels EBFS gefundenen Pläne

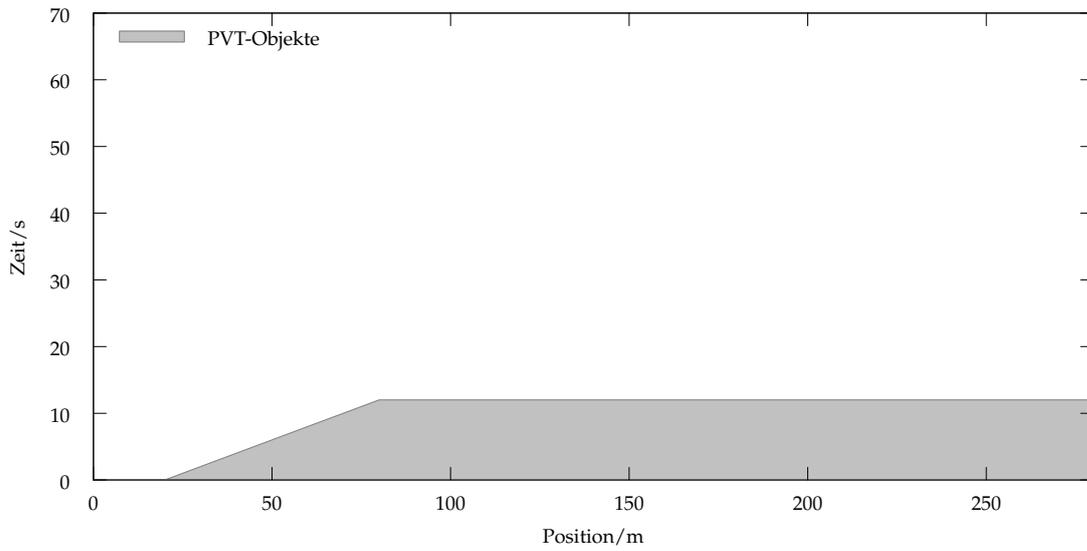


Abbildung 6.16: Bundesstraßenszenario - PVT-Objekte für Plan mit Überholung

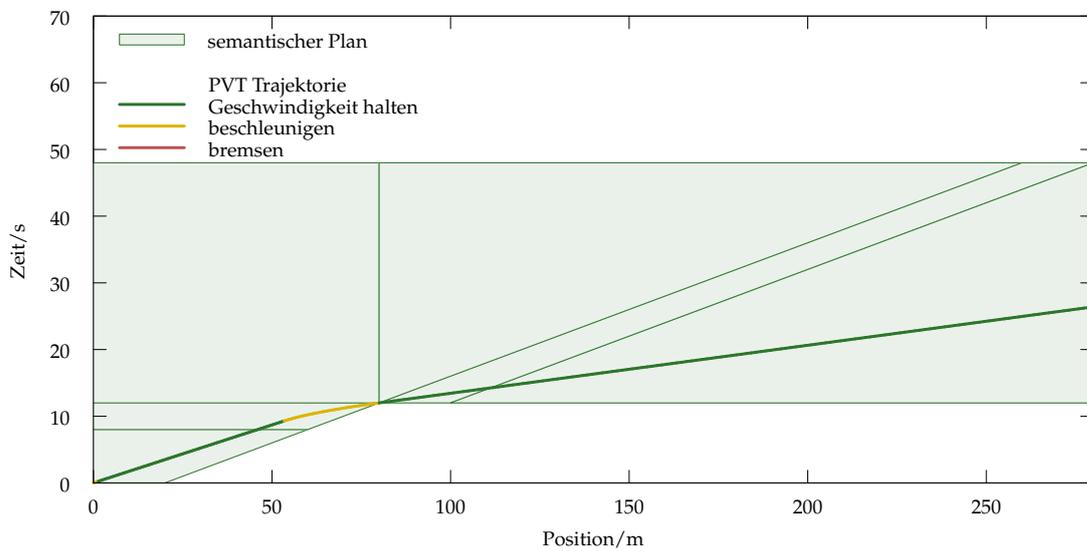


Abbildung 6.17: Bundesstraßenszenario - Plan mit Überholung erstellt durch EA*S

6 Dynamikvalidierung und -evaluierung

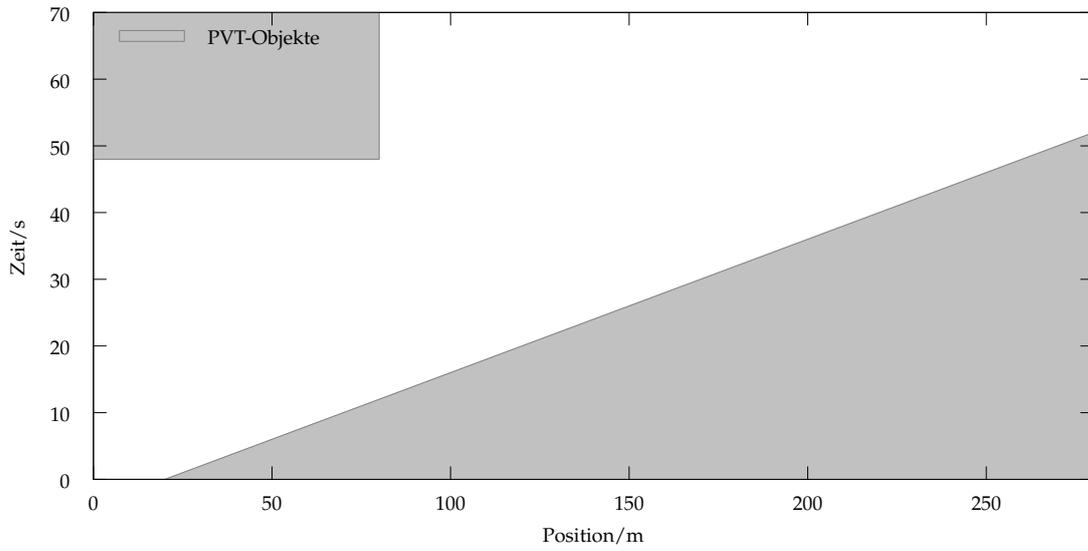


Abbildung 6.18: Bundesstraßenszenario - PVT-Objekte für Plan ohne Überholung

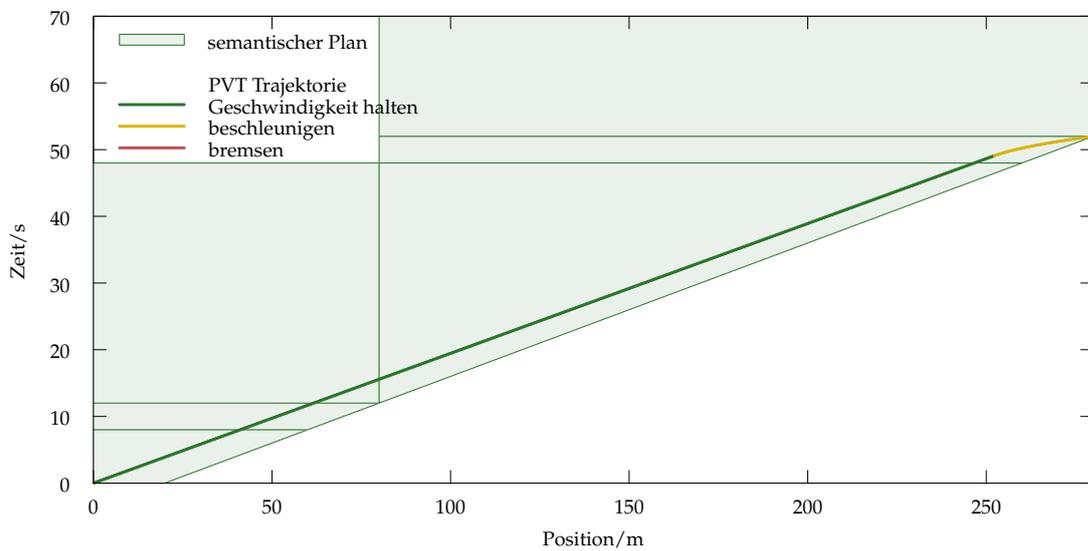


Abbildung 6.19: Bundesstraßenszenario - Plan ohne Überholung erstellt durch EA*S

6.4.3 Laufzeitverhalten

Das Laufzeitverhalten der verschiedenen Planungsalgorithmen wird anhand des Bundesstraßenszenarios mit variabler Länge (vergleiche Abbildung 4.14) verglichen. Quantitative Zahlen zu Zustandsraumgröße und gefundenen Plänen sind in Kapitel 5.3.1 aufgeführt. Die Laufzeit wurde auf einem Desktop-PC mit einem Intel® Core™ i7-5820K Prozessor, 3,30 GHz und 15,6 GB Arbeitsspeicher gemessen. Tabelle 6.2 zeigt die Laufzeit ausgewählter Planungsverfahren für verschiedene Szenariolängen gemittelt über 100 Ausführungen.

Szenario- länge	Laufzeit der Planung				
	EBFS	EBFSH	A*S	A*S (D)	EA*S (D)
3	1 ms	2 ms	0 ms	2 ms	3 ms
4	10 ms	5 ms	1 ms	5 ms	14 ms
5	347 ms	41 ms	1 ms	13 ms	176 ms
6	– ms	128 ms	2 ms	28 ms	2,242 ms
7	– ms	950 ms	3 ms	52 ms	– ms
8	– ms	5,356 ms	4 ms	94 ms	– ms
9	– ms	– ms	6 ms	158 ms	– ms

Tabelle 6.2: Auswertung der Laufzeit der semantischen Planung im Bundesstraßenszenario. Algorithmen mit integrierter Dynamikvalidierung sind mit (D) gekennzeichnet.

Einfache Suchverfahren wie EBFS kommen auch bei kleinen Szenariogrößen schnell an Rechenzeitgrenzen, so dass sie für einen Einsatz im Fahrzeug nicht geeignet sind. Eine starke Reduktion der Rechenzeit wird durch die Ausnutzung von Homotopieklassen (EBFSH) erreicht. Dies erlaubt einen Einsatz für kleine Szenarien oder beschränkte Planungshorizonte.

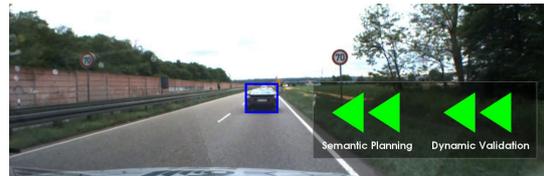
Sehr kleine Laufzeiten werden durch die A*-Verfahren erreicht. Verzichtet man auf eine Dynamikvalidierung zur Planungszeit, so können extrem kurze Planungszeiten im Millisekundenbereich erzielt werden. Dabei ist jedoch nicht gewährleistet, dass die gefundenen Pläne auch dynamisch fahrbar sind. Im konkreten Beispiel sind dennoch die Mehrzahl der Pläne auch dynamisch fahrbar.

Wird die Dynamikvalidierung schon zur Laufzeit eingesetzt (A*S (D) und EA*S (D)) steigen die Planungszeiten, da die Validierung in jedem Planungsschritt ausgeführt wird. Dennoch lassen sich vor allem in Kombination mit der A*-Suche Planungszeiten erzielen, die den Online-Einsatz des Verfahrens ermöglichen. Eine Kombination aus A*-Suche und der Berücksichtigung von Homotopieklassen

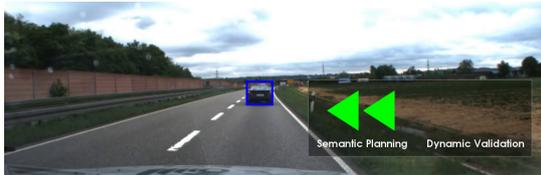
6 Dynamikvalidierung und -evaluierung



(a) Solange das Fahrzeug noch auf der linken Spur fährt, ist kein Überholmanöver möglich.



(b) Das Fahrzeug kann jetzt überholt werden.



(c) Vor Erreichen der Abzweigung ist zwar ein Überholen semantisch noch möglich. Die maximale Beschleunigung und Geschwindigkeit des Fahrzeugs erlaubt ein Überholen hier jedoch nicht mehr.



(d) Deutlich später ist ein Überholen aufgrund der Semantik der Szene nicht mehr möglich.

Abbildung 6.20: Spurwechslassistent mit und ohne Dynamikvalidierung. Es werden zwei Hinweise dargestellt (grüne Pfeile): Links auf Basis der semantischen Planung, rechts zusätzlich unter Berücksichtigung der Dynamikvalidierung.

scheint großes Potential zu bieten und könnte Teil zukünftiger Forschungsarbeiten sein.

6.4.4 Anwendung – Überholassistent

Wie in Kapitel 5.3.2 bereits erwähnt, besteht bei der beispielhaften Anwendung Überholassistent der Bedarf, die semantischen Pläne auch dynamisch zu validieren. Die hier vorgestellte Dynamikvalidierung wurde daher auch in den Überholassistenten integriert. Als Resultat werden nur Überholvorgänge vorgeschlagen, die unter Berücksichtigung der wahrgenommenen Verkehrssituation, der geplanten Route und der erwarteten Dynamik des eigenen Fahrzeugs fahrbar sind.

Abbildung 6.20 zeigt beispielhaft den Mehrwert der zusätzlichen Dynamikvalidierung der semantischen Pläne. Dabei wird die Ausgabe der rein semantischen Planung (grüne Pfeile links), wie sie in Kapitel 5.3.2 vorgestellt wurde, mit der um die Dynamikvalidierung erweiterten Planung (grüne Pfeile rechts) verglichen. Wird kein Pfad zum Überholen des Fahrzeugs gefunden, so ergeben sich keine Unterschiede (Abbildung 6.20a). Auch bei genügend langem Weg zum Überholen empfehlen beide Verfahren ein Überholmanöver (Abbildung 6.20b). Vor

Erreichen der Abzweigung ist es nicht mehr sinnvoll einen Überholvorgang zu starten (Abbildung 6.20c). Hier ist zwar semantisch noch ein Überholvorgang möglich, die Algorithmik mit Dynamikvalidierung gibt jedoch richtigerweise keine Überholempfehlung mehr. Kurz vor Erreichen der Verzweigung gibt es dann auch aufgrund der Semantik der Szene keine Überholmöglichkeit mehr und beide Verfahren geben keine Überholempfehlung mehr (Abbildung 6.20d). Durch die Integration der Dynamikvalidierung ist es möglich, dem Fahrer sinnvollere Hinweise zu geben.

6.5 Zusammenfassung und Fazit

Für die Dynamikvalidierung und -evaluierung wird eine Geschwindigkeitsprofiloptimierung innerhalb der zu bewertenden Manöversequenzen eingesetzt. Ausgehend von den in Kapitel 3.4.4 definierten Anforderungen an die Dynamikvalidierung wurden relevante Eigenschaften der Geschwindigkeitsprofiloptimierung herausgestellt und ihr Einfluss auf die Anforderungserfüllung dargestellt. Es wurden mehrere Verfahren zur Geschwindigkeitsprofiloptimierung vorgestellt: Dynamische Programmierung, modellbasierte lokale Optimierung, vorberechnete Geschwindigkeitsprofile, Modellprädiktion sowie ein Sichtgraphverfahren. Es wurden jeweils Anwendungen für diese gezeigt und ihre Eigenschaften beschrieben.

Das Sichtgraphverfahren hat sich im Vergleich als am besten geeignet für die Validierung von Manöverketten herausgestellt. Daher wurde dieses in das Gesamtsystem integriert. Es wurde gezeigt, dass semantische Manöverketten mit diesem Verfahren auf dynamische Fahrbarkeit hin bewertet werden können. Auch kann diese Bewertung in einer heuristikbasierten Suche eingesetzt werden, um dynamische Fahrbarkeit schon frühzeitig im Planungsprozess zu gewährleisten und die Planungsergebnisse zu verbessern. Dies wurde exemplarisch am Beispiel eines Überholassistenten gezeigt. Eine Anwendung im Rahmen des automatisierten Fahrens ist ebenfalls möglich und wird im Folgenden genauer dargestellt.

7 Evaluation im Gesamtsystem

Nachdem die einzelnen Komponenten der semantischen Manöverplanung in den jeweiligen Kapiteln 4.4, 5.3 und 6.4 evaluiert wurden, soll hier nun auch das Gesamtsystem im realen Fahrversuch erprobt werden. Im Rahmen des Projekts RobustSENSE¹ wurden die hier beschriebenen Verfahren in einem Rahmenwerk zum automatisierten Fahren eingebettet und in einer realen Verkehrsszene getestet. Als Versuchsträger kam das in Kapitel 2.2.2 vorgestellte Fahrzeug CoCar zum Einsatz.

Im Folgenden wird zunächst das betrachtete Szenario vorgestellt (Kapitel 7.1). Anschließend werden in Kapitel 7.2 die Einbettung der semantischen Manöverplanung in die bestehende Systemarchitektur sowie die verwendeten Komponenten und Schnittstellen beschrieben. Zur Darstellung der Ergebnisse wird die Visualisierungsumgebung Robot Operating System Visualization (RViz) verwendet. Diese wird in Kapitel 7.3 vorgestellt. Abschließend werden Ergebnisse der semantischen Manöverplanung anhand des Demonstrationsszenarios vorgestellt (Kapitel 7.4).

7.1 Szenario

Die in dieser Arbeit vorgestellten Verfahren wurden im Rahmen einer Projektdemonstration des Projekts RobustSENSE im realen Fahrversuch erprobt. Da der Fokus der Demonstration auf der Erkennung und Behandlung nichtregelkonformen Verhaltens lag, wurde ein Kreuzungsszenario für die Demonstration ausgewählt, bei der das automatisiert fahrende Ego-Fahrzeug Vorfahrt hat. Ein zweites Fahrzeug, das sich von rechts der Kreuzung nähert, müsste Vorfahrt gewähren. Die Demonstration wurde auf dem Gelände der Daimler TTS GmbH in Ulm an einer Kreuzung der Wilhelm-Runge-Straße² durchgeführt. Abbildung 7.1 zeigt eine Übersicht der Situation in mehreren Zeitschritten.

7.2 Systemarchitektur und Schnittstellen

Die hier vorgestellten Verfahren zur semantischen Manöverplanung sollen in ein Gesamtsystem zum automatisierten Fahren eingebettet werden. Hierzu muss das

¹ RELIABLE, SECURE, TRUSTABLE SENSORS. FOR AUTOMATED DRIVING: robustsense.eu (26.03.2020)

² 48°25'18,8"N 9°56'24,3"E



Abbildung 7.1: Demonstrationsszenario – Überblick: Zwei Fahrzeuge nähern sich einer Kreuzung. Das automatisierte Ego-Fahrzeug (silbern) von links hat Vorfahrt. Das Fahrzeug von rechts (schwarz) muss Vorfahrt gewähren.

System an die angestrebte Zielplattform angepasst und in das verwendete Rahmenwerk eingebunden werden. Als Zielplattform wird hier das Forschungsfahrzeug CoCar (siehe Kapitel 2.2.2) des FZI verwendet. Als Integrationsumgebung und Kommunikationsstruktur kommt auf diesem Fahrzeug das Robot Operating System (ROS) zum Einsatz. Durch die modulare Struktur der in ROS verwendeten Komponenten in Form von sogenannten Nodes ist es leicht möglich, neue Komponenten einzubinden und in die bisherige Struktur zu integrieren. Des Weiteren steht für den Test der Algorithmik eine Simulationsumgebung auf Basis der freien Robotersimulation Gazebo³ zur Verfügung.

Die im Projekt RobustSENSE verwendete Systemarchitektur weist einige projektspezifische Eigenschaften auf, auf die hier im Detail nicht eingegangen werden soll. Im Wesentlichen besteht sie aus den in Kapitel 3.1 vorgestellten Komponenten:

Perzeption und Egopose

Perzeptionsverfahren liegen in dieser Anwendung nicht im Fokus. Daher wird die Perzeption aus einer Kombination im Fahrzeug verbauter Laserscanner sowie einer präzisen C2C-Kommunikation der Fahrzeugposition und -geschwindigkeit gebildet. Der aktuelle Fahrzeugzustand also die Pose und Geschwindigkeit wird von einer Lokalisierungskomponente in Form einer GPS-Nachricht bereitgestellt.

Interpretation und Prädiktion

Zur Interpretation und Prädiktion der Verkehrsszene kommt das in Kapitel 2.6.1 vorgestellte Verfahren zum Einsatz. Es erlaubt das Verhalten der Verkehrsteilnehmer probabilistisch zu schätzen und jeweils entsprechende Bewegungsprofile zu präzisieren. Dabei wird pro Verhalten eine Entwicklung der Szene sowie ihre Wahrscheinlichkeit geschätzt. Die prädizierten Objektdaten werden jeweils in Form einer Objektliste der semantischen Planung übergeben.

Manöverentscheidung

Zur Manöverentscheidung wird die in dieser Arbeit vorgestellte semantische

³ Gazebo: gazebosim.org (26.03.2020)

Manöverplanung verwendet. Entsprechend der szenenbasierten Prädiktion werden zur Behandlung der Unsicherheit in der Prädiktion eine Kombination der in Kapitel 4.5.2 dargestellten Methode der szenenbasierten Zustandsräume sowie der Zusammenfassung von semantischen Pfaden verwendet. Dies erlaubt mehrere Szenenentwicklungen wie „links Abbiegen“ und „geradeaus fahren“ eines Fahrzeugs zu einem Verhalten „in Kreuzung einfahren“ zusammenzufassen und anschließend pro prädictiertem Verhalten eine Manöversequenz, falls vorhanden, zu generieren. Als Suchverfahren wird das in Kapitel 6.4.2 vorgestellte heuristikbasierte Suchverfahren EA*S(D) ausgewählt.

Trajektorienplanung

Die durch die semantische Planung erstellten Manöverketten sollen in einem geometrischen Planungsverfahren als Eingabe verwendet werden. Hierzu wird der semantische Plan in geo-temporale Grenzen umgewandelt, so dass eine Repräsentation des befahrbaren Fahrschlauchs in örtlicher und zeitlicher Dimension entsteht. Als Trajektorienplanung kommt in diesem Fall ein optimierungsbasierter Algorithmus zum Einsatz.

Karte/Routing

Zur Festlegung der zu fahrenden Route wird auf Basis einer global verfügbaren Laneletkarte eine Route als Lanelet-Korridor in Form von Lanelet-IDs generiert. Diese wird von einer zentralen Instanz, einem Lanelet-Server, bereitgestellt. Für den Aufbau eines semantischen Zustandsraums wird neben der zu befahrenden Route auch das Umfeld dieser Route benötigt. Dieses wird durch eine iterative Erweiterung des Lanelet-Korridors durch benachbarte und kreuzende Lanelets erstellt.

Zur Integration in das Rahmenwerk ROS werden die Softwarekomponenten zur semantischen Planung und Manöverkettenvalidierung in einen ROS-Node integriert. Zur Kommunikation mit den anderen Komponenten der Systemarchitektur sind entsprechende Schnittstellen implementiert. Diese werden in ROS in Form von Nachrichten umgesetzt. Hier kann zum Teil auf bestehende Nachrichtendefinitionen zurückgegriffen werden.

7.3 Visualisierung der Verarbeitungsschritte

Zur Visualisierung der Ergebnisse wird die Visualisierungsumgebung RViz verwendet. Es werden der verwendete Lanelet-Korridor, die für die semantische Szenenbeschreibung verwendeten Lanelets, der entstehende Zustandsraum und der darauf erstellte Plan in einer dreidimensionalen Ansicht visualisiert.

Im Folgenden werden die Visualisierungskomponenten zusammen mit den jeweiligen Verarbeitungsschritten beispielhaft an einer Fahrt über den Campus Ost des KIT dargestellt. Die Straßen und Kreuzungen des Campus Ost sind in einer

7 Evaluation im Gesamtsystem

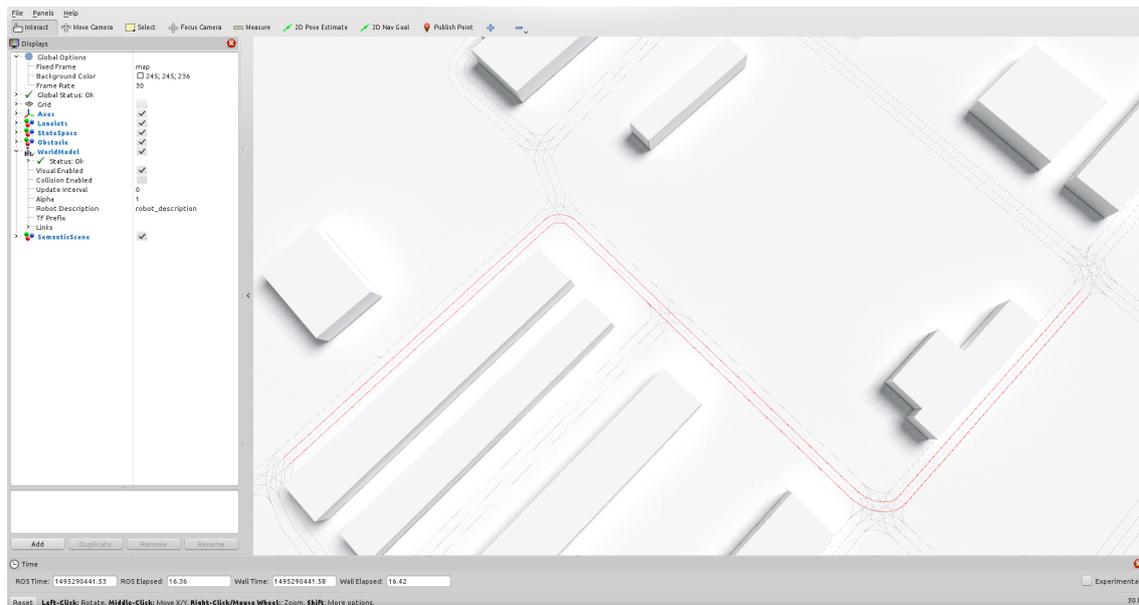


Abbildung 7.2: Visualisierung in RViz – Laneletkarte mit Korridor (rot) auf dem Campus Ost des KIT

detaillierten Lanelet-Karte modelliert. Diese steht den verschiedenen Softwarekomponenten zur Verfügung. Abbildung 7.2 zeigt einen Ausschnitt der Lanelet-Karte in Form von Begrenzungslinien in der Visualisierungsumgebung RViz.

Zunächst wird auf Basis der aktuellen Fahrzeugposition und der Zielvorgabe durch einen Lanelet-Server eine Route ermittelt und in Form eines Lanelet-Korridors an die semantische Planungskomponente übergeben. Der ermittelte Lanelet-Korridor verläuft in Abbildung 7.2 von links nach rechts über zwei Kreuzungen. Er ist rot hervorgehoben.

Auf Basis des Korridors werden alle für die semantische Szenendarstellung benötigten Fahrspursegmente ermittelt. Diese sind in Abbildung 7.3 grün hervorgehoben. Hierzu werden ausgehend von der gewählten Route sukzessiv angrenzende und kreuzende Fahrspursegmente bis zu einer bestimmten Maximaldistanz berücksichtigt.

Zusammen mit der Position von anderen Verkehrsteilnehmern wird ein semantischer Zustandsraum aufgebaut. In diesem Beispiel fährt ein weiterer Verkehrsteilnehmer die mittlere Straße diagonal von oben links nach unten rechts entlang (nicht visualisiert). Abbildung 7.4 zeigt den resultierenden semantischen Zustandsraum. Die Zustände sind als farbige Kugeln dargestellt. Sie sind entsprechend der geometrischen Position der ihnen zugrundeliegenden Fahrspursegmente angeordnet. Konstellationen sind in z-Richtung von unten nach oben aufgetragen. Zustände, in denen Kollisionen mit anderen Fahrzeug auftreten können, sind wie gewohnt rot dargestellt.

Eine auf dem Zustandsraum geplante Manöversequenz ist grün dargestellt. Wie in Kapitel 4.4.4 gezeigt, wird auch hier der Zustandsraum zeitlich beschränkt,

7.3 Visualisierung der Verarbeitungsschritte

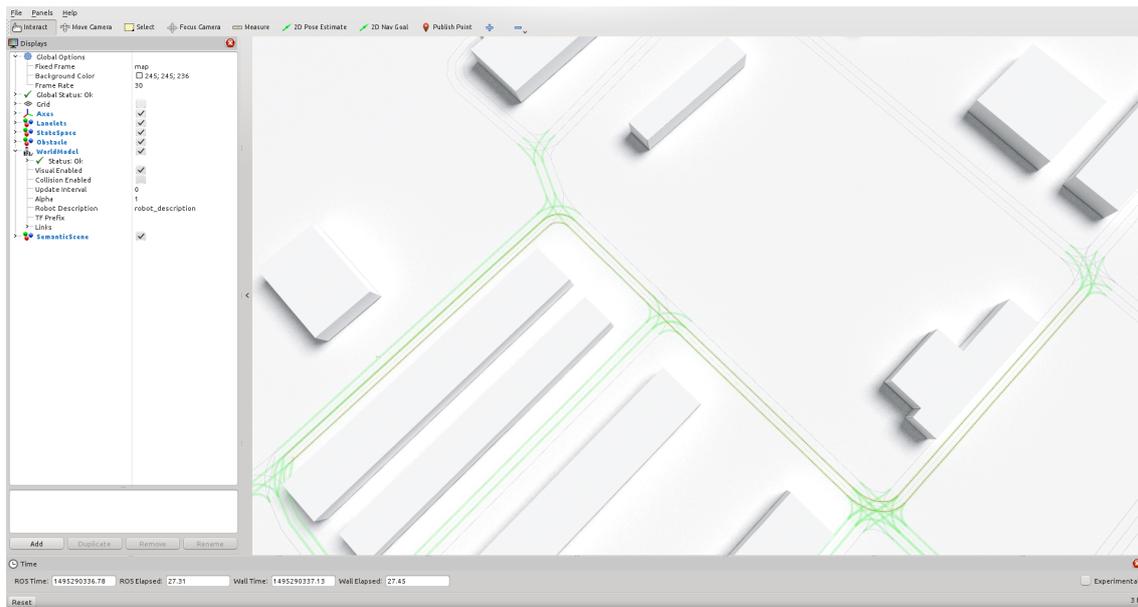


Abbildung 7.3: Visualisierung in RViz – Erweiterter Korridor (grün) als Grundlage für die semantische Szenenbeschreibung

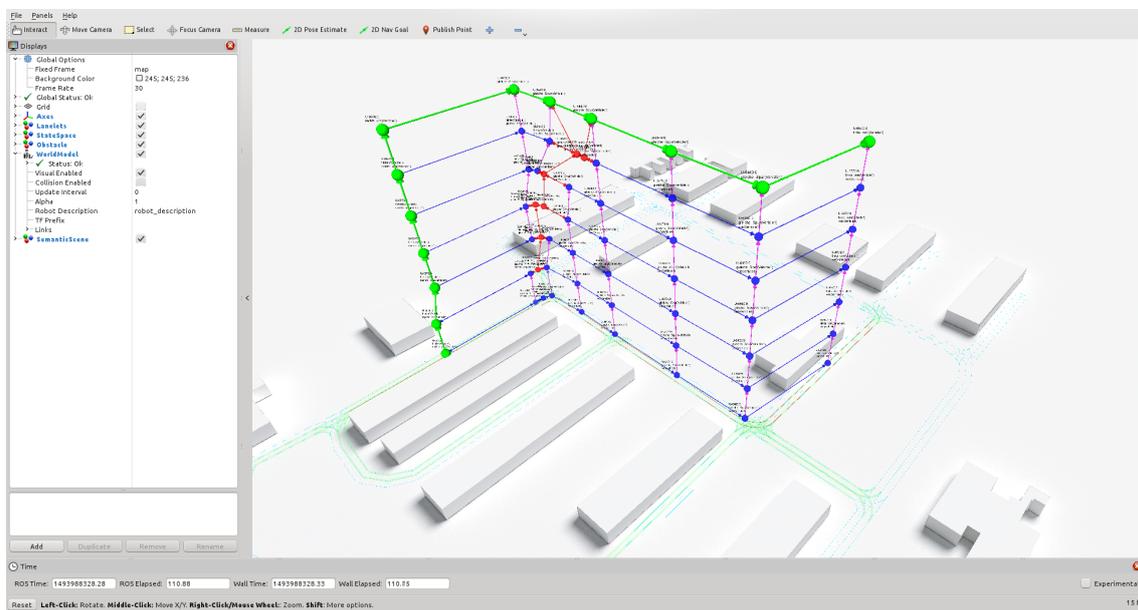


Abbildung 7.4: Visualisierung in RViz – Semantischer Zustandsraum für den geplanten Korridor mit Manöverplan



Abbildung 7.5: Demonstrationsszenario – Detailansicht. Das schwarze Fahrzeug gewährt erst sehr spät Vorfahrt. Das automatisierte, silberne Fahrzeug soll sich der Kreuzung entsprechend defensiv nähern.

so dass die letzte, in der Darstellung oberste, Konstellation keine dynamischen Objekte mehr berücksichtigt.

7.4 Demonstration und Ergebnisse

Der Verlauf der Demonstration ist so gewählt, dass das Ego-Fahrzeug situativ verschiedene Verhalten zeigen muss. Das schwarze Fahrzeug stellt eine Situation nach, in der der Fahrer erst kurz vor Erreichen der Kreuzung erkennt, dass er Vorfahrt gewähren muss. Das Ego-Fahrzeug soll trotz eigener Vorfahrt diese Situation erkennen und sich entsprechend defensiv verhalten. Nachdem der Fahrer des schwarzen Fahrzeugs durch starkes Verzögern die Situation verkehrsregelkonform löst, soll das Ego-Fahrzeug die Kreuzung zügig passieren. In Abbildung 7.5 ist die Situation kurz vor Erreichen der Kreuzung gezeigt. Das schwarze Fahrzeug verzögert stark, so dass das Ego-Fahrzeug die Kreuzung passieren kann.

In Abbildung 7.6 ist exemplarisch eine Anfahrt auf die Kreuzung zu drei Zeitpunkten dargestellt⁴. Auf dem Display im Fahrzeug auf der Beifahrerseite ist die Ausgabe des online im Fahrzeug laufenden Gesamtsystems zu erkennen. Zur besseren Lesbarkeit ist jeweils auf der rechten Seite ein Bildschirmfoto der Visualisierung der semantischen Manöverplanung dargestellt⁵.

Hier ist wie in Kapitel 7.3 beschrieben die Laneletkarte (blaue Fahrspurbegrenzungen), der verwendete Korridor inklusive Erweiterung für die semantische

⁴ Die Aufnahmen stammen aus Vorbereitungsfahrten am Vortag der Projektabschlussveranstaltung.

⁵ Die Bilder aus dem Fahrzeuginneren und aus der Visualisierung stammen von zwei verschiedenen Versuchsfahrten. Abweichungen in den jeweiligen Fahrzeugpositionen sind daher möglich.

Szenenbeschreibung (grüne Fahrspuren) sowie der semantische Zustandsraum (Netzwerk aus farbigen Kugeln) dargestellt. Für jeden prädizierten Szenenverlauf wird jeweils ein Zustandsraum und ein optimaler semantischer Plan erstellt. Es wird zur besseren Übersichtlichkeit jeweils nur der Zustandsraum des in diesem Zeitpunkt als am wahrscheinlichsten angenommenen Szenenverlaufs dargestellt. Zusätzlich sind in dieser Visualisierung die Position des Ego-Fahrzeugs (graues Auto), die aus der Perzeption ermittelte Objektliste mit Objekten (orange Boxen) und ihren Geschwindigkeiten (graue Pfeile) dargestellt. Das aktuell geschätzte Verhalten des anderen Fahrzeugs wird durch türkise Kugeln visualisiert: Eine Kugel zwischen den Fahrzeugen mit verbindenden Linien zu beiden Fahrzeugen symbolisiert die Annahme, dass das Fahrzeug von rechts Vorfahrt gewährt. Eine Kugel über dem Fahrzeug visualisiert die Annahme, dass das Fahrzeug uneingeschränkt fährt, also nicht auf das andere Fahrzeug reagiert.

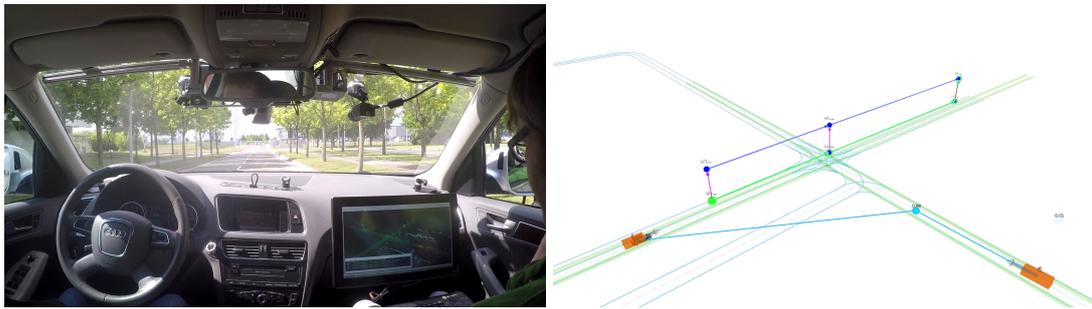
In Abbildung 7.6a sind beide Fahrzeuge noch weit von der Kreuzung entfernt. Die Komponenten zur Interpretation und Prädiktion haben eine hohe Wahrscheinlichkeit (99 %) für regelkonformes Verhalten ermittelt. Der dargestellte semantische Zustandsraum basiert auf der Annahme, dass das Fahrzeug von rechts vor der Kreuzung abbremsen und anhalten wird. Er besteht aus zwei Konstellationen, einer dynamischen und einer statischen. Da auch im dynamischen Fall keine relevanten Objekte in der Szene vorhanden sind, wird ein einfacher Plan zum Überfahren der Kreuzung generiert.

Abbildung 7.6b zeigt die Szene einige Sekunden später. Das Fahrzeug von rechts nähert sich weiter mit unverminderter Geschwindigkeit. Da zu diesem Zeitpunkt normalerweise schon eine Verzögerung des Fahrzeugs zu erwarten ist, ist die Wahrscheinlichkeit für ein nicht regelkonformes Verhalten auf annähernd 100 % angestiegen. Entsprechend zeigt der dargestellte semantische Zustandsraum, dass es bei Einfahrt in die Kreuzung innerhalb der ersten Konstellation zu einer Kollision kommen kann (rote Kugeln). Die semantische Pfadplanung gibt daher einen Pfad vor, bei dem die Kreuzung erst nach dem anderen Fahrzeug passiert werden kann.

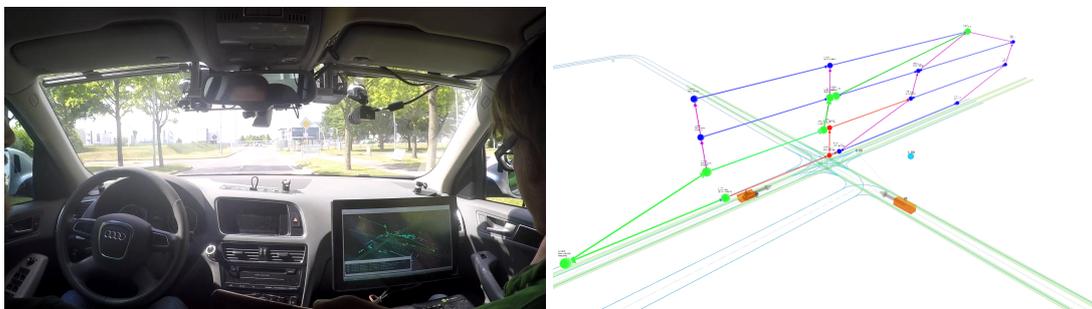
Kurz darauf bremst das Fahrzeug von rechts sehr stark ab. Obwohl es in der in Abbildung 7.6c dargestellten Szene noch eine Restgeschwindigkeit aufweist, kann aus der starken Verzögerung geschlossen werden, dass es die Vorfahrt des Ego-Fahrzeugs beachten wird. Entsprechend kann wieder der dargestellte semantische Plan, der das Fahrzeug die Kreuzung ohne Einschränkungen passieren lässt, an die Trajektorienplanung als sehr wahrscheinlich ausführbar weitergegeben werden.

Im Rahmen der Abschlussveranstaltung des Projekts RobustSENSE im Mai 2018 konnte das hier gezeigte System einen Tag lang erfolgreich einem breiten Fachpublikum demonstriert werden. Dabei hat sich das in dieser Arbeit vorgestellte Verfahren zur semantischen Manöverplanung als funktionale und robuste Möglichkeit zur Manöverentscheidung auf Basis einer probabilistischen Szenenbeschreibung erwiesen. Das automatisierte Versuchsfahrzeug wurde dadurch befähigt in der ausgewählten Situation mit uneindeutigem Verhalten anderer Verkehrsteilnehmer sichere und performante Fahrentscheidungen zu treffen.

7 Evaluation im Gesamtsystem



(a) Für das Fahrzeug von rechts wird regelkonformes Verhalten prädiziert, **ego** (von links) kann die Kreuzung ohne Beeinträchtigung passieren.



(b) Für das Fahrzeug von rechts wird regelwidriges Verhalten prädiziert, **ego** (von links) verzögert, um eine Kollision zu vermeiden.



(c) Das Fahrzeug von rechts bremst stark ab. Es wird wieder regelkonformes Verhalten prädiziert, **ego** (von links) kann die Kreuzung passieren.

Abbildung 7.6: Kreuzungsanfahrt - Sicht aus dem Fahrzeug und semantische Planung

8 Zusammenfassung und Ausblick

In dieser Arbeit wurden grundlegende Methoden zur Bereitstellung von abstrakten Fahrmanöversequenzen für automatisiertes Fahren sowie zur passiven Unterstützung des Fahrers entwickelt und evaluiert. Sie tragen dazu bei, aktuelle Herausforderungen des automatisierten Fahrens, das automatisierte Verstehen von komplexen Verkehrssituationen und darauf aufbauend die Generierung von optimalen Fahrstrategien, zu bewältigen. Exemplarisch wurden diese Methoden in eine bestehende Systemarchitektur eines automatisierten Versuchsfahrzeugs integriert. Die Eignung der hier vorgestellten Verfahren konnte in verschiedenen synthetischen und aufgezeichneten Szenarien sowie in einer realen Fahrdeemonstration am Beispiel einer Anfahrt auf eine Kreuzung erfolgreich präsentiert werden.

8.1 Zusammenfassung der Ergebnisse

Als theoretische Grundlage wurde zunächst ein Konzept zur Abstraktion der Verkehrssituation durch eine relationale, semantische Szenenrepräsentation sowie darauf aufbauend Methoden zur effizienten Planung und dynamischen Validierung von Fahrmanöverketten entwickelt. Ausgehend von konzeptionellen Anforderungen wurde ein systematisches Vorgehen zur Ableitung des Planungsraums auf Basis einer geometrischen Umfeldrepräsentation sowie zur Planung und Validierung von optimalen Fahrmanöversequenzen vorgestellt.

In dem erarbeiteten Vorgehen werden aus einer geometrische Verkehrsszenenrepräsentation mittels einer Ontologie Entitäten und Relationen abgeleitet. Darauf aufbauend wird ein diskreter semantischer Zustandsraum als gerichteter Graph gebildet. Der zeitliche Verlauf der Szene wird in Form von Subgraphen, sogenannten Konstellationen, berücksichtigt, die jeweils für die Zeit zwischen zwei aufeinanderfolgenden Ereignissen gültig sind. Übergänge zwischen Zuständen sind an Regeln gebunden, die prädikatenlogisch formuliert und zur Darstellung z. B. von Verkehrsregeln verwendet werden können.

Auf der so gebildeten Graphstruktur können mittels Verfahren zur Graphensuche semantische Manöversequenzen geplant werden. Die Rückprojektion der semantischen Pläne in eine geometrische Darstellung als Raum-Zeit-Ebene ermöglicht die dynamische Validierung und Optimierung resultierender Pläne hinsichtlich Energieeffizienz und Komfort. Aus den Plänen abgeleitete Fahrhinweise können dem Fahrer durch Augmentierung der Umgebung angezeigt und für hochautomatisierte Fahrfunktionen verwendet werden.

Durch eine relationale Modellierung der Verkehrsszene kann der Planungsraum für Fahrmanöver an semantischen Grenzen diskretisiert werden. Hierdurch bildet sich die Komplexität der Szene direkt in der Größe des semantischen Zustandsraums ab. Dies wurde experimentell und auch theoretisch durch die Herleitung der Komplexitätsklasse nachgewiesen. Des Weiteren wurden Einflüsse von probabilistischen Effekten in der geometrischen Umfeldrepräsentation auf den semantischen Zustandsraum sowie Möglichkeiten derer Modellierung untersucht.

Der semantische Zustandsraum abstrahiert von der Verkehrsszene, so dass zur Planung generische sowie spezialisierte Verfahren zur Graphensuche eingesetzt werden können. Eine Auswahl bekannter Suchverfahren wurde auf die Anforderungen der Manöverplanung angepasst. Die Suchverfahren wurden um Konzepte zur Beschleunigung der Suche wie die Ausnutzung von Homotopieklassen und spezielle Heuristiken erweitert. Die resultierenden Verfahren wurden evaluiert, so dass für verschiedene Anwendungen passende Verfahren ausgewählt werden können.

Um die semantischen Pläne auf dynamische Fahrbarkeit hin zu überprüfen, wurden verschiedene Methoden zur Dynamikvalidierung vorgestellt, jeweils typische Anwendungsszenarien gezeigt und die Methoden anhand ausgewählter Kriterien evaluiert. Als geeignetes Verfahren stellte sich ein modifiziertes Sichtgraphverfahren heraus, da dieses die Kernanforderungen der Dynamikvalidierung hinsichtlich Optimalität und Laufzeit erfüllt. Es wird gezeigt, wie dieses in die semantische Planung eingebunden werden kann, so dass für erzeugte Manöverketten auch eine dynamische Fahrbarkeit gewährleistet werden kann.

Die vorgestellten Verfahren zur semantischen Manöverplanung wurden implementiert und in verschiedenen Anwendungsbeispielen zur passiven und aktiven Fahrerassistenz auf synthetischen sowie realen Daten, in der Simulation und auf aufgezeichneten Sensordaten evaluiert. Somit wurde ihre Anwendbarkeit nachgewiesen. Es wurde gezeigt, wie die semantische Manöverplanung in ein Gesamtframework zum automatisierten Fahren eingebettet werden kann.

Dies ermöglichte die Anwendung der entwickelten Verfahren im realen Fahrversuch. Als Versuchsträger kam das mit Sensoren und Aktuatoren ausgestattete Versuchsfahrzeug CoCar des FZI zum Einsatz. Auf diesem Fahrzeug konnte in einem beispielhaft gewählten Szenario, der Anfahrt auf eine Kreuzung mit kreuzendem Verkehr, eine robuste Fahrentscheidung auf Basis realer Umfelddaten getroffen werden. Dieses Szenario wurde im Rahmen der Abschlussveranstaltung des Projekts RobustSENSE erfolgreich demonstriert.

8.2 Ausblick

Das hier gezeigte Verfahren zur Manöverplanung kann zur Manöverentscheidung beim automatisierten Fahren eingesetzt werden. Die abgeleiteten semantischen,

abstrakten Fahrbefehle können im Regelfall jedoch nicht direkt zur Steuerung eines Fahrzeugs verwendet werden. Sie müssen in einem weiteren Schritt aufbereitet und konkretisiert werden. Dies konnte zwar im Rahmen der Fahrdemonstration prototypisch gezeigt werden, allgemein ist es jedoch erforderlich hier flexiblere Verfahren wie eine dynamische Trajektorienplanung zu verwenden. Diese Verfahren können den semantischen Manöverplan als Randbedingung für eine Planung verwenden.

Die kombinatorische Komplexität des Planungsprozesses wird bereits von der Manöverplanung aufgelöst. Hierdurch ist es möglich, für die Trajektoriengenerierung ein breites Spektrum an Planungsalgorithmen, z. B. lokale Verfahren wie Optimierer, einzusetzen. Auch kann das erzeugte Geschwindigkeitsprofil aus der Dynamikvalidierung als Ausgangsbasis für den zeitlichen Verlauf der Trajektorie verwendet werden, was je nach Verfahren die Planungszeit verkürzen kann.

Auch die aktuell erfolgreichen Algorithmen auf Basis tiefer neuronaler Netze, die rein auf Sensordaten eine Fahrentscheidung treffen, können von einer Manöverentscheidung profitieren. Diese Verfahren können bereits autonom Spurwechsel und Abbiegevorgänge ausführen. Das Wissen darüber, unter welchen Randbedingungen, wann und in welche Richtung dies sinnvoll ist, kann durch die Manöverplanung ergänzt werden.

Im Bereich der in dieser Arbeit vorgestellten Planungsverfahren hat sich die Kombination der A*-Suche mit der Betrachtung von Homotopieklassen als vielversprechende Weiterführung der Forschungsarbeiten herausgestellt. Die positiven Effekte beider Verfahren könnten im besten Fall zu einem effizienten Planungsverfahren führen, das durch die Kombination von Teilplänen möglichst generelle Pläne für die Weiterverwendung in einer Trajektorienplanung bereitstellt.

Die semantische Szenenbeschreibung in Form der entwickelten Ontologie kann als Grundlage zum verteilten Szenenverständnis verwendet werden. Durch die von den erfassten Sensordaten stark abstrahierte Darstellung der Szene bietet sie großes Potential für die Kommunikation über car to infrastructure (C2X) zwischen verschiedenen Fahrzeugsystemen mit beschränkter Bandbreite. So stellt sie einen großen Mehrwert bei der verteilten und kooperativen Manöverplanung dar, wie sie u. a. im SPP 1835: Kooperativ interagierende Automobile erforscht wird¹.

Ein weiterer Anwendungsbereich der semantischen Szenenbeschreibung liegt in Verfahren zum maschinellen Lernen für das automatisierte Fahren. Eine Repräsentation der erzeugten Relationen als Tensor und die anschließende Verarbeitung durch neuronale Netze zur Generierung von Fahrentscheidungen ist Gegenstand aktueller Forschungsvorhaben [PAK+18]. Durch die Reduktion der Szenenkomplexität durch die Abstraktion von geometrischen Details ist mit einer besseren Generalisierungsfähigkeit bei weniger Lernbeispielen zu rechnen. Da die Anwendung von Lernverfahren in der Regel keine Planung darstellt, muss die Prädiktion

¹ Institut AIFB - SPP 1835: Kooperativ interagierende Automobile: www.aifb.kit.edu/web/SPP_1835:_Kooperativ_interagierende_Automobile (26.03.2020)

8 Zusammenfassung und Ausblick

des zeitlichen Verlaufs in diesem Fall implizit berücksichtigt werden. Hierzu können dynamische Aspekte der Szene in die Ontologie und somit in die semantische Szenenbeschreibung zusätzlich mit aufgenommen werden.

Abbildungsverzeichnis

1.1	Schematische Darstellung einer Verkehrssituation in der Innenstadt	2
1.2	Schematische Darstellung des Aufbaus der Arbeit.	5
2.1	Ein kognitives System interagiert mit seiner Umgebung	7
2.2	CoCar – Sensorsetup des Forschungsfahrzeugs des FZI	13
2.3	Messungen eines Fahrzeugs zur Bildung eines Dynamik- und Verbrauchsmodells	18
2.4	Applikationsontologie einer Kreuzung für ein Fahrerassistenzsystem	23
2.5	Graphische Darstellung einer einfachen Ontologie zur Beschreibung einer Verkehrsszene	24
2.6	Vergleich verschiedener geometrisch topologischer Repräsentationsformen von Straßennetzen	25
2.7	Verschiedene Repräsentationsmöglichkeiten einer T-Kreuzung in OSM	26
2.8	Kartenausschnitt dargestellt mit Lanelets	28
2.9	Kartendarstellung mit Lanelets	29
2.10	Typische Verarbeitungskette eines videobildbasierten Objektdetektors	30
2.11	Videobasierte Detektion von Autos auf einer semivirtuellen Autobahnszene	31
2.12	Objekterkennung auf einem Tiefenbild einer TOF-Kamera	33
2.13	Verarbeitungskette eines kombinierten Objektdetektors für 2D- und 3D-Bilder	34
2.14	Objektdetektion und Klassifikation auf Basis eines Tiefenbilds einer TOF-Kamera und eines Videobilds	34
2.15	Schematische Darstellung der wichtigsten Begriffe des Szenenmodells	36
2.16	Blockdiagramm eines IMM-Filters mit zwei Modellen	37
2.17	Unterschied in der Suchreihenfolge von Breiten- und Tiefensuche	44
2.18	Anwendung einer Heuristik in der A*-Suche	45
2.19	Bestimmung eines zeitoptimalen Geschwindigkeitsprofils mittels eines Sichtgraph-Verfahrens	48
2.20	Manövervarianten für ein Ego-Fahrzeug in einer Verkehrsszene mit drei weiteren Fahrzeugen	49
2.21	Beispielhafte Skelettierung eines Planungsraums	49
3.1	Horizontale Sicht auf die konkretisierte Systemarchitektur	52

3.2	Konzeptionelle Darstellung der semantischen Manöverplanung	56
3.3	Detaillierung der Systemarchitektur	57
3.4	Beschreibung der Fahrzeugpose relativ zu einer Route	61
3.5	Einfache Beispielszene - Straße mit zwei Fahrspuren, Spurwechsel erlaubt	63
3.6	Einfache Beispielszene - Straße mit zwei Fahrspuren, Spurwechsel eingeschränkt	63
3.7	Einfache Beispielszene - Straße mit zwei Fahrspuren	64
4.1	TBox der verwendeten Ontologie zur Verkehrsszenenbeschreibung	68
4.2	Beispiel ausgewählter Relationen zwischen statischen Szenenobjekten	70
4.3	Straßenszenario mit einem Fahrzeug	71
4.4	Kreuzungsszenario mit einem Fahrzeug	72
4.5	Mögliche Kombinationen von Relationen eines Fahrzeugs zu einem anderen Fahrzeug	72
4.6	Konsistente Bestimmung der longitudinalen Relation in zwei exemplarischen Szenarien	75
4.7	Schematische Darstellung des Bundesstraßenszenarios	79
4.8	Zustandsraum unter Berücksichtigung der Relationen zu statischen Szenenobjekten	80
4.9	Zustandsraum unter Berücksichtigung der lateralen Relationen zu dynamischen Szenenobjekten	80
4.10	Zustandsraum unter Berücksichtigung der longitudinalen Relationen zu dynamischen Szenenobjekten	81
4.11	Beispielhafte Berechnung von Konstellationsintervallen	82
4.12	Temporale Fortsetzung des Zustandsraums über drei Konstellationen	84
4.13	Anwendung von Regeln zur Einhaltung des Sicherheitsabstands zu anderen Fahrzeugen	86
4.14	Schematische Darstellung eines skalierbaren Bundesstraßenszenarios	88
4.15	Kartenausschnitt des Bundesstraßenszenarios als Lanelet-Karte	88
4.16	Schematische Darstellung eines Kreuzungsszenarios	88
4.17	Kartenausschnitt des Kreuzungsszenarios als Lanelet-Karte	88
4.18	Ausgewählte Entitäten und Relationen aus der ABox des Bundesstraßenszenarios	89
4.19	Ausgewählte Entitäten und Relationen aus der ABox des Kreuzungsszenarios	90
4.20	Ausgewählte Entitäten und Relationen aus der ABox des Bundesstraßenszenarios	90
4.21	Ausgewählte Entitäten und Relationen aus der ABox des Kreuzungsszenarios mit Pfad eines Fahrzeugs	91
4.22	Ausgewählte Entitäten und Relationen aus der ABox des Kreuzungsszenarios mit Pfad eines abbiegenden Fahrzeugs	92

4.23	Ausgewählte Entitäten und Relationen aus der ABox des Bundesstraßenszenarios mit Pfad eines stoppenden Fahrzeugs	92
4.24	Resultierender Zustandsraum für das Bundesstraßenszenario ohne andere Verkehrsteilnehmer	93
4.25	Resultierender Zustandsraum für das Bundesstraßenszenario mit einem anderen Fahrzeug	94
4.26	Resultierender Zustandsraum für das Bundesstraßenszenario mit einem anderen Fahrzeug beschränkt auf einen Zeithorizont von 10 s	95
4.27	Resultierender Zustandsraum für das Kreuzungsszenario mit einem anderen Fahrzeug	96
4.28	Resultierender Zustandsraum für das Kreuzungsszenario mit einem anderen Fahrzeug beschränkt auf drei dynamische Konstellationen	96
4.29	Beispielszenarien für die Analyse der Zustandsraumgröße	97
4.30	Vergleich der Größen der Zustandsräume für verschiedene Szenarien und unterschiedlich viele Fahrzeuge	98
4.31	Unschärfe in der Wahrnehmung	102
4.32	Unsicherheit in der Wahrnehmung	102
4.33	Unsicherheit in der Prädiktion	103
4.34	Beispielhafter Zustandsraum mit wahrscheinlichkeitsbehafteten Relationen in den Zuständen	104
4.35	Anfahrt auf eine Ampel – Unschärfe in der Wahrnehmung	106
4.36	Spurwechsel bei Anfahrt auf eine Ampel – Unschärfe in der Eigenpositionierung	107
5.1	Möglicher Plan auf dem semantischen Zustandsraum des Bundesstraßenszenarios	112
5.2	Anwendung eines Homotopiebausteins auf zwei Pläne	117
5.3	Homotopiebausteine zur Darstellung eines unnötigen Spurwechsels	118
5.4	Resultierender Plan für das Bundesstraßenszenario mit A*S (1)	122
5.5	Resultierender Plan für das Bundesstraßenszenario mit A*S (2)	123
5.6	Resultierende Pläne für das Bundesstraßenszenario mit EBFS	124
5.7	Resultierender Plan für das Kreuzungsszenario mit EBFS	125
5.8	Resultierende Pläne für das Bundesstraßenszenario mit EBFSH	126
5.9	Resultierende Pläne für das Kreuzungsszenario mit EBFSH	127
5.10	Kartenausschnitt bei Bruchsal	130
5.11	Schematische Darstellung des Streckenabschnitts	130
5.12	Spurwechselassistent auf realen Fahrzeugdaten	131
5.13	Automatisierter Spurwechsel in der Simulation	133
6.1	Beispiel Dynamische Programmierung: Variierende Kosten für Zeit- und Energieverbrauch	141
6.2	Beispiel Dynamische Programmierung: Variierende Bewertung für überhöhte Geschwindigkeit	142

6.3	Beispiel Dynamische Programmierung: Optimales Passieren einer Ampel	143
6.4	Beispiel Gradientenabstieg	145
6.5	Berechnung der Policy zur Annäherung an statische Geschwindigkeitsbeschränkungen	147
6.6	Fahrerassistenz für energieeffizientes Fahren auf statische Geschwindigkeitsbeschränkungen	148
6.7	Vorberechnetes Geschwindigkeitsprofil für ein Landstraßenszenario	150
6.8	Aktionshinweise als Augmentierung an die Straße	151
6.9	Drei exemplarische Situationen der automatisierten Längsführung im innerstädtischen Bereich	153
6.10	Anfahrt auf ein langsames vorausfahrendes Fahrzeug	155
6.11	Dynamikvalidierung eines semantischen Plans mit Hilfe des PVT-Planers	160
6.12	Bundesstraßenszenario - Projizierter semantischer Plan aus der EBFS	163
6.13	Bundesstraßenszenario - Konvertierung des Fahrschlauchs in PVT-Objekte	164
6.14	Bundesstraßenszenario - Optimales Geschwindigkeitsprofil im projizierten Fahrschlauch mittels PVT-Planer	164
6.15	Bundesstraßenszenario - Dynamikvalidierung der acht mittels EBFS gefundenen Pläne	166
6.16	Bundesstraßenszenario - PVT-Objekte für Plan mit Überholung	167
6.17	Bundesstraßenszenario - Plan mit Überholung erstellt durch EA*S	167
6.18	Bundesstraßenszenario - PVT-Objekte für Plan ohne Überholung	168
6.19	Bundesstraßenszenario - Plan ohne Überholung erstellt durch EA*S	168
6.20	Spurwechselassistent mit und ohne Dynamikvalidierung	170
7.1	Demonstrationsszenario – Überblick	174
7.2	Visualisierung in RViz – Laneletkarte mit Korridor	176
7.3	Visualisierung in RViz – Erweiterter Korridor	177
7.4	Visualisierung in RViz – Semantischer Zustandsraum	177
7.5	Demonstrationsszenario – Detailansicht Kreuzung	178
7.6	Kreuzungsanfahrt - Sicht aus dem Fahrzeug und semantische Planung	180

Tabellenverzeichnis

5.1	Auswertung der Zustandsraumgröße im Bundesstraßenszenario	128
6.1	Bewertung der Verfahren zur Dynamikvalidierung	158
6.2	Auswertung der Laufzeit im Bundesstraßenszenario	169

Abkürzungsverzeichnis

A*S A*-Suche

ABox assertional box

ACC Abstandsregeltempomat (engl.: adaptive cruise control)

BASt Bundesamt für Straßenwesen

BFS Breitensuche (engl.: breadth-first search)

C2C car to car

C2X car to infrastructure

CNN convolutional neural network

CoCar Cognitive Car

DARPA Defense Advanced Research Projects Agency des US-amerikanischen Verteidigungsministeriums

DFS Tiefensuche (engl.: depth-first search)

DP Dynamische Programmierung

EA*S vollständige A*-Suche (engl.: exhaustive A*-search)

EBFS vollständige Breitensuche (engl.: exhaustive breadth-first search)

EBFSH vollständige Breitensuche mit Berücksichtigung von Homotopieklassen

ECU Steuergerät (engl.: electronic control unit)

EDFS vollständige Tiefensuche (engl.: exhaustive depth-first search)

ERM entity-relationship-model

ESP elektronisches Stabilitätsprogramm

FZI FZI Forschungszentrum Informatik

GCDC Grand Cooperative Driving Challenge

GPS Globales Positionsbestimmungssystem (engl.: global positioning system)

HMM hidden Markow model

HUD head-up-display

HOG histogram of oriented gradients

Abkürzungsverzeichnis

- ICE** Verbrennungsmotor (engl.: internal combustion engine)
- IDM** intelligent driver model
- IMM-Filter** interacting multiple model filter
- IMU** inertielle Messeinheit (engl.: inertial measurement unit)
- KIT** Karlsruher Institut für Technologie
- KF** Kalman-Filter
- MDP** Markov decision process
- MLN** Markov logic network
- MMI** multi media interface
- MPC** modellprädiktive Regelung (engl.: model predictive control)
- NHTSA** National Highway Traffic Safety Administration
- OSM** OpenStreetMap
- PMD** photonic mixing device
- POMDP** partially observable Markov decision process
- PVT-Planer** Pfad-Geschwindigkeit-Zeit-Planer (engl.: path-velocity-time-planner)
- RNDF** route network definition file
- ROS** Robot Operating System
- RRT** rapidly-exploring random tree
- RViz** Robot Operating System Visualization
- SAE** Society of Automotive Engineers
- SFB/TR 28** Sonderforschungsbereich/Transregio 28 Kognitive Automobile
- SVM** support vector machine
- SWRL** semantic web rule language
- TBox** terminological box
- TOF** Licht-Laufzeit-Verfahren (engl.: time of flight)
- UML** unified modeling language

Quellennachweise

Veröffentlichungen

- [KSN+11] Ralf Kohlhaas, Thomas Schamm, Dennis Nienhüser und J. Marius Zöllner. „Anticipatory energy saving assistant for approaching slower vehicles“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2011.
- [BNK+11] Tobias Bär, Dennis Nienhüser, Ralf Kohlhaas und J. Marius Zöllner. „Probabilistic driving style determination by means of a situation based analysis of the vehicle data“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2011.
- [ZKB+14] Marc René Zofka et al. „Robust, Marker-Based Head Tracking for Testing Cognitive Vehicles in the Loop“. In: *Advances in Visual Computing, ISVC*. Springer, 2014.
- [KKS+15] Florian Kuhnt, Ralf Kohlhaas, Thomas Schamm und J. Marius Zöllner. „Towards a Unified Traffic Situation Estimation Model – Street-dependent Behaviour and Motion Models –“. In: *IEEE International Conference on Information Fusion, Fusion*. IEEE, 2015.
- [BKZ+11] Tobias Bär, Ralf Kohlhaas, J. Marius Zöllner und Kay-Ulrich Scholl. „Anticipatory driving assistance for energy efficient driving“. In: *IEEE Forum on Integrated and Sustainable Transportation Systems, FISTS*. IEEE, 2011.
- [KKJ+14a] Florian Kuhnt et al. *Particle Filter Map Matching and Trajectory Prediction Using a Spline Based Intersection Model*. Techn. Ber. FZI Forschungszentrum Informatik, 2014.
- [ZKS+14] Marc Rene Zofka, Ralf Kohlhaas, Thomas Schamm und J. Marius Zöllner. „Semivirtual simulations for the evaluation of vision-based ADAS“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2014.
- [KSL+13] Ralf Kohlhaas, Thomas Schamm, Dominik Lenk und J. Marius Zöllner. „Towards driving autonomously: Autonomous cruise control in urban environments“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2013.
- [KSS+10] Ralf Kohlhaas, Thomas Schamm, Marcus Strand und J. Marius Zöllner. „Flexible fusion of 2D images and 3D time of flight data using plane segmentation“. In: *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*. IEEE, 2010.

- [SSG+09] Thomas Schamm et al. „Vision and ToF-based driving assistance for a personal transporter“. In: *International Conference on Advanced Robotics, ICAR*. IEEE, 2009.
- [KPZ+16] Florian Kuhnt et al. „Robust environment perception for the audi autonomous driving cup“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2016.
- [KKJ+14b] Florian Kuhnt et al. „Particle Filter Map Matching and Trajectory Prediction using a Spline Based Intersection Model * (Extended Abstract)“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2014.
- [KBS+14] Ralf Kohlhaas, Thomas Bittner, Thomas Schamm und J. Marius Zöllner. „Semantic state space for high-level maneuver planning in structured traffic scenes“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2014.
- [KHS+15] Ralf Kohlhaas, Daniel Hammann, Thomas Schamm und J. Marius Zöllner. „Planning of High-level Maneuver Sequences on Semantic State Spaces“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2015.
- [KSZ15] Ralf Kohlhaas, Thomas Schamm und J. Marius Zöllner. „Manöverbasierte Optimierung von Geschwindigkeitsprofilen für Fahrerassistenz und Automatisiertes Fahren“. In: *7. Tagung Fahrerassistenz*. TÜV Süd, 2015.
- [NBK+12] Dennis Nienhüser et al. „Energy Efficient Driving and Operation Strategies Based on Situation Awareness and Reasoning“. In: *it - Information Technology*. Oldenbourg Wissenschaftsverlag, 2012.
- [Koh08] Ralf Kohlhaas. „Gemeinsame Kalibrierung von 2D-Farb- und 3D-Tiefenbildkameras mit 2D Kalibrierungsmustern“. Studienarbeit. KIT, 2008.
- [Koh09] Ralf Kohlhaas. „Informationsfusion mit dynamischen Bayes’schen Netzen zur Zustandsschätzung von Verkehrsteilnehmern“. Diplomarbeit. KIT, 2009.
- [KSG+09] Ralf Kohlhaas, Ferdinand Szekeresch, Tobias Gindele und Rüdiger Dillmann. „Dynamic Bayesian Network Library: A C++ framework for calculations on dynamic Bayesian networks“. In: *Autonome Mobile Systeme, AMS*. Springer, 2009.
- [ZSN+10] Johann Marius Zöllner et al. „Advanced technical cognitive systems for mobility assistance“. In: *KIT-Nachwuchswissenschaftler-Symposium*. 2010.
- [ZKK+15] Marc René Zofka et al. „Data-Driven Simulation and Parametrization of Traffic Scenarios for the Development of Advanced Driver Assistance Systems“. In: *IEEE International Conference on Information Fusion, Fusion*. IEEE, 2015.

- [KEO+16] Sebastian Klemm et al. „Autonomous multi-story navigation for valet parking“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2016.
- [ZKK+16] Marc René Zofka, Florian Kuhnt, Ralf Kohlhaas und J. Marius Zöllner. „Simulation framework for the development of autonomous small scale vehicles“. In: *IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots, SIMPAR*. IEEE, 2016.
- [KTH+16] J. Kaiser et al. „Towards a framework for end-to-end control of a simulated vehicle with spiking neural networks“. In: *IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots, SIMPAR*. IEEE, 2016.
- [ZEF+18] Marc Zofka, Marc Essinger, Tobias Fleck, Ralf Kohlhaas und Johann Marius Zöllner. „The Sleepwalker Framework : Verification and Validation of Autonomous Vehicles by Mixed Reality LiDAR Stimulation“. In: *International Conference on Simulation, Modeling, and Programming for Autonomous Robots, SIMPAR*. IEEE, 2018.
- [FDW+18] Tobias Fleck et al. „Towards Large Scale Urban Traffic Reference Data: Smart Infrastructure in the Test Area Autonomous Driving Baden-Württemberg“. In: *Conference on Intelligent Autonomous Systems*. Springer, 2018.
- [BDD+19] Jürgen Beyerer et al. „General Fail-Safe Emergency Stopping for Highly Automated Vehicles“. In: *9. Tagung Automatisiertes Fahren*. TÜV Süd, 2019.

Studentische Arbeiten

- [Ham15] Daniel Philipp Hammann. „Planung von Fahrmanöversequenzen auf semantischen Zustandsräumen unter Berücksichtigung der Fahrdynamik“. Masterarbeit. KIT, 2015.
- [Bit14] Thomas Bittner. „Semantische Modellierung von Straßenverkehrsszenen zur Ableitung von Fahrmanövern“. Masterarbeit. KIT, 2014.
- [Bat15] Pierre Batram. „Berücksichtigung von unsicherem Wissen in der semantischen Manöverplanung“. Studienarbeit. KIT, 2015.
- [Car13] Ecaterina Carabadjac. „Geschwindigkeitsprofiloptimierung auf semistatischen Fahrsituationen“. Bachelorarbeit. KIT, 2013.
- [Göt11] Isabel Götzenberger. „Klassifikation von Bildausschnitten zur videobasierten Fußgängererkennung für Fahrerassistenzsysteme“. Diplomarbeit. KIT, 2011.

- [Mun13] Silviu Munteanu. „Entwurf und Implementierung einer Mensch-Maschine-Schnittstelle für ein autonomes Fahrzeug“. Diplomarbeit. KIT, 2013.
- [Len13] Dominik Lenk. „Entwicklung einer Fahrzeug-Längsführung für automatisiertes Fahren im urbanen Straßenverkehr“. Masterarbeit. KIT, 2013.

Literatur

- [TKZ+16] Omer Sahin Tas, Florian Kuhnt, J. Marius Zöllner und Christoph Stiller. „Functional system architectures towards fully automated driving“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2016.
- [Nat13] National Highway Traffic Safety Administration (NHTSA). *Preliminary Statement of Policy Concerning Automated Vehicles*. Techn. Ber. 2013.
- [GAA+12] Tom M. Gasser et al. „Rechtsfolgen zunehmender Fahrzeugautomatisierung“. In: *Berichte der Bundesanstalt für Straßenwesen, Unterreihe „Fahrzeugsicherheit“*. Bundesanstalt für Straßenwesen (BASt), 2012.
- [On-16] On-Road Automated Vehicle Standards Committee. „Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles“. In: *SAE International, Standard J3016*. 2016.
- [Smi13] Bryant Walker Smith. *Summary of Levels of Driving Automation for On-Road Vehicles*. 2013.
- [DGN87] Ernst Dieter Dickmanns, V. Graefe und W. Niegel. *Abschlußbericht Definitionsphase PROMETHEUS, Pro-Art*. Techn. Ber. Universität der Bundeswehr München, 1987.
- [TSP+18] Omer Sahin Tas et al. „Making Bertha Cooperate—Team AnnieWAY’s Entry to the 2016 Grand Cooperative Driving Challenge“. In: *IEEE Transactions on Intelligent Transportation Systems*. IEEE, 2018.
- [WWZ16] Michael Weber, Peter Wolf und J. Marius Zöllner. „DeepTLR: A single deep convolutional network for detection and classification of traffic lights“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2016.
- [KOH+15] Sebastian Klemm et al. „RRT*-Connect: Faster, asymptotically optimal motion planning“. In: *IEEE International Conference on Robotics and Biomimetics, ROBIO*. IEEE, 2015.
- [RS40] Paul Riekert und Theo Ernst Schunck. „Zur Fahrmechanik des gummbereiften Kraftfahrzeugs“. In: *Ingenieur-Archiv* 11. 1940.
- [GU10] Frank Gauterin und Hans-Joachim Unrau. *Grundlagen der Fahrzeugtechnik I - Scriptum zur Vorlesung nach Prof. Dr.-Ing. R. Gnadler*. Institut für Fahrzeugsystemtechnik, Lehrstuhl für Fahrzeugtechnik, KIT, 2010.

- [RSK+11] Christian Raubitschek, Nico Schütze, Evgeny Kozlov und Bernard Bäker. „Predictive driving strategies under urban conditions for reducing fuel consumption based on vehicle environment information“. In: *IEEE Forum on Integrated and Sustainable Transportation Systems, FISTS*. IEEE, 2011.
- [BGV+09] Sebastian Brechtel, Tobias Gindele, Jan Vogelsang und Rüdiger Dillmann. „Probabilistisches Belegtheitsfilter zur Schätzung dynamischer Umgebungen unter Verwendung multipler Bewegungsmodelle“. In: *Autonome Mobile Systeme, AMS*. Springer, 2009.
- [GBS+09] Tobias Gindele, Sebastian Brechtel, Joachim Schröder und Rüdiger Dillmann. „Bayesian occupancy grid filter for dynamic environments using prior map knowledge“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2009.
- [STT+13] Sayanan Sivaraman, Mohan M. Trivedi, Mario Toppelhofer und Trevor Shannon. „Merge recommendations for driver assistance: A cross-modal, cost-sensitive approach“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2013.
- [SMJ12] Andreas Schindler, Georg Maier und Florian Janda. „Generation of high precision digital maps using circular arc splines“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2012.
- [HW11] Kai Homeier und Lars Wolf. „RoadGraph: High level sensor data fusion between objects and street network“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2011.
- [KH10] J. Knaup und K. Homeier. „RoadGraph - Graph based environmental modelling and function independent situation analysis for driver assistance systems“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2010.
- [SBF98] Rudi Studer, V. Richard Benjamins und Dieter Fensel. „Knowledge Engineering: Principles and Methods“. In: *Data & Knowledge Engineering*. 1998.
- [HZW11] Michael Hülsen, J. Marius Zöllner und Christian Weiss. „Traffic intersection situation description ontology for advanced driver assistance“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2011.
- [AFIG14] Alexandre Armand, David Filliat und Javier Ibanez-Guzman. „Ontology-based context awareness for driving assistance systems“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2014.
- [HPSB+04] Ian Horrocks et al. „SWRL : A Semantic Web Rule Language Combining OWL and RuleML“. In: *W3C Member submission 21*. 2004.
- [DKP+06] Pedro Domingos, Stanley Kok, Hoifung Poon, Matthew Richardson und Parag Singla. „Unifying Logical and Statistical AI“. In: *National Conference on Artificial Intelligence, AAAI*. AAAI, 2006.

- [KPV+07] Sören Kammel, Benjamin Pitzer, Stefan Vacek und Joachim Schröder. *DARPA Urban Challenge Team AnnieWAY Technical System Description*. Techn. Ber. KIT, 2007.
- [HTL07] Britta Hummel, Werner Thiemann und Irina Lulcheva. „Description Logic for Vision-Based Intersection Understanding“. In: *Cognitive systems with interactive sensors, COGIS*. Stanford University, 2007.
- [Thr98] Sebastian Thrun. „Learning metric-topological maps for indoor mobile robot navigation“. In: *Artificial Intelligence*. 1998.
- [Fre12] Christian Frese. *Planung kooperativer Fahrmanöver für kognitive Automobile*. KIT Scientific Publishing, 2012.
- [BZS14] Philipp Bender, Julius Ziegler und Christoph Stiller. „Lanelets: Efficient map representation for autonomous driving“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2014.
- [ZBS+14] Julius Ziegler et al. „Making bertha drive—an autonomous journey on a historic route“. In: *IEEE Intelligent Transportation Systems Magazine*. IEEE, 2014.
- [PPJ+18] Fabian Poggenhans et al. „Lanelet2 : A high-definition map framework for the future of automated driving“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2018.
- [NGZ11] Dennis Nienhüser, Thomas Gump und J. Marius Zöllner. „Relevance Estimation of Traffic Elements Using Markov Logic Networks“. In: *IEEE International Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2011.
- [VJ03] Paul Viola und Michael J. Jones. „Detecting Pedestrians Using Patterns of Motion and Appearance“. In: *IEEE International Conference on Computer Vision, ICCV*. IEEE, 2003.
- [Vap98] Vladimir N. Vapnik. *Statistical learning Theory*. John Wiley & Sons, 1998.
- [Kal60] Rudolph Emil Kalman. „A new approach to linear filtering and prediction problems“. In: *Transactions of the ASME, Journal of Basic Engineering*. 1960.
- [SZV+08] Thomas Schamm, J. Marius Zöllner, Stefan Vacek, Joachim Schröder und Rüdiger Dillmann. „Obstacle detection with a Photonic Mixing Device-camera in autonomous vehicles“. In: *International Journal of Intelligent Systems Technologies and Applications*. Inderscience Publishers, 2008.
- [SRZ10] Thomas Schamm, Arne Rönnau und J. Marius Zöllner. „Fast 3D Time of Flight Data Segmentation using the U-V-Histogram Approach“. In: *Intelligent Robotics and Applications*. Springer, 2010.
- [PPS+18] Florian Piewak et al. „Boosting LiDAR-based Semantic Labeling by Cross-Modal Training Data Generation“. In: *The European Conference on Computer Vision: Workshops, ECCV*. Springer, 2018.

- [BSLK01] Yaakov Bar-Shalom, X. Rong Li und Thiagalingam Kirubarajan. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. John Wiley & Sons, 2001.
- [KSS+16] Florian Kuhnt, Jens Schulz, Thomas Schamm und Johann Marius Zöllner. „Understanding Interactions between Traffic Participants based on Learned Behaviors“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2016.
- [GNL+09] Thomas Gump, Dennis Nienhüser, Rebecca Liebig und J. Marius Zöllner. „Recognition and tracking of temporary lanes in motorway construction sites“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2009.
- [GOZ12] Thomas Gump, Jan Oberländer und J. Marius Zöllner. „Probabilistic hierarchical detection, representation and scene interpretation of lanes and roads“. In: *IEEE International Conference on Vehicular Electronics and Safety, ICVES*. IEEE, 2012.
- [Töp14] Daniel Töpfer. „On Compositional Hierarchical Models for holistic Lane and Road Perception in Intelligent Vehicles“. Diss. KIT, 2014.
- [KOK+16] Florian Kuhnt, Stefan Orf, Sebastian Klemm und J. Marius Zöllner. „Lane-precise Localization of Intelligent Vehicles Using the Surrounding Object Constellation“. In: *IEEE International Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2016.
- [HSS15] Toni Heidenreich, Jens Spehr und Christoph Stiller. „LaneSLAM-Simultaneous Pose and Lane Estimation Using Maps with Lane-Level Accuracy“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2015.
- [ALB+02] Jan Christian Albiez, Tobias Luksch, Karsten Berns und Rüdiger Dillmann. „An Activation-Based Behavior Control Architecture for Walking Machines“. In: *International Journal of Robotics Research*. SAGE Publications, 2002.
- [THH00] Martin Treiber, Ansgar Hennecke und Dirk Helbing. „Congested traffic states in empirical observations and microscopic simulations“. In: *Physical Review*. American Physical Society, 2000.
- [DP04] R. Dittmar und B.-M. Pfeiffer. *Modellbasierte prädiktive Regelung: Eine Einführung für Ingenieure*. Oldenbourg, 2004.
- [KZZ18] Karl Kurzer, Chenyang Zhou und J. Zöllner. „Decentralized Cooperative Planning for Automated Vehicles with Hierarchical Monte Carlo Tree Search“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2018.
- [LF09] M. Likhachev und D. Ferguson. „Planning Long Dynamically Feasible Maneuvers for Autonomous Vehicles“. In: *The International Journal of Robotics Research*. 2009.
- [Zie15] Julius Ziegler. *Optimale Trajektorienplanung für Automobile*. KIT Scientific Publishing, 2015.

- [SP14] Wilko Schwarting und Patrick Pascheka. „Recursive Conflict Resolution for Cooperative Motion Planning in Dynamic Highway Traffic“. In: *IEEE International Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2014.
- [Vac09] Stefan Vacek. *Videogestützte Umfelderkennung zur Interpretation von Verkehrssituationen für kognitive Automobile*. Universitätsverlag Karlsruhe, 2009.
- [BGD14] Sebastian Brechtel, Tobias Gindele und Rüdiger Dillmann. „Probabilistic Decision-Making under Uncertainty for Autonomous Driving using Continuous POMDPs“. In: *IEEE International Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2014.
- [HBW+17] Christian Hubschneider, André Bauer, Michael Weber und J Marius Zöllner. „Adding Navigation to the Equation: Turning Decisions for End-to-End Vehicle Control“. In: *IEEE International Conference on Intelligent Transportation Systems: Workshop, ITSC*. IEEE, 2017.
- [UM13] Simon Ulbrich und Markus Maurer. „Probabilistic Online POMDP Decision Making for Lane Changes in Fully Automated Driving“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2013.
- [UM14] Simon Ulbrich und Markus Maurer. „Evaluation einer taktischen Verhaltensentscheidungsfindung für Fahrstreifenwechsel beim vollautomatisierten Fahren in Städten“. In: *9. Workshop Fahrerassistenzsysteme, FAS*. TÜV Süd, 2014.
- [BTD+16] Mariusz Bojarski et al. „End to End Learning for Self-Driving Cars“. In: *arXiv*. 2016.
- [HBD+17] Christian Hubschneider et al. „Integrating End-to-End Learned Steering into Probabilistic Autonomous Driving“. In: *IEEE International Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2017.
- [KZP+09] Sören Kammel et al. „Team AnnieWAY’s autonomous system for the DARPA Urban Challenge 2007“. In: *Springer Tracts in Advanced Robotics*. Springer, 2009.
- [CLR+09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest und Clifford Stein. *Introduction to Algorithms*. MIT Press, 2009.
- [HNR68] P.E. Hart, N.J. Nilsson und B. Raphael. „A Formal Basis for the Heuristic Determination of Minimum Cost Paths“. In: *IEEE Transactions on Systems Science and Cybernetics*. IEEE, 1968.
- [TS89] O. Takahashi und R.J. Schilling. „Motion planning in a plane using generalized Voronoi diagrams“. In: *IEEE Transactions on Robotics and Automation*. IEEE, 1989.
- [JH13] Jeff Johnson und Kris Hauser. „Optimal longitudinal control planning with moving obstacles“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2013.

- [BLK12] S. Bhattacharya, M. Likhachev und V. Kumar. „Topological constraints in search-based robot path planning“. In: *Autonomous Robots*. Springer, 2012.
- [BTZ+15] Philipp Bender, Omer Sahin Tas, Julius Ziegler und Christoph Stiller. „The Combinatorial Aspect of Motion Planning: Maneuver Variants in Structured Environments“. In: *IEEE Intelligent Vehicles Symposium, IV*. IEEE, 2015.
- [Ben15] Philipp Bender. „Trajektorienplanung : Manöveridentifikation anhand der Topologie des Freiraums“. In: *10. Workshop Fahrerassistenzsysteme, FAS*. Uni-DAS e. V, 2015.
- [Sch09] Joerg Henning Schneider. „Modellierung und Erkennung von Fahr-situationen und Fahrmanövern für sicherheitsrelevante Fahrerassis-tenzsysteme“. Diss. Technischen Universität Chemnitz, 2009.
- [PAK+18] Dominik Petrich, Darius Azarfar, Florian Kuhnt und J. Marius Zöllner. „The Fingerprint of a Traffic Situation: A Semantic Relationship Tensor for Situation Description and Awareness“. In: *IEEE Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2018.