Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

Reihe C | Dissertationen | Heft Nr. 824

Oliver Bertin Röth

Extraktion von hochgenauer Fahrspurgeometrie und -topologie auf der Basis von Fahrzeugtrajektorien und Umgebungsinformationen

München 2018

Verlag der Bayerischen Akademie der Wissenschaften

ISSN 0065-5325 | ISBN 978-3-7696-5236-9

Veröffentlichungen der DGK

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften

Reihe C | Dissertationen | Heft Nr. 824

Extraktion von hochgenauer Fahrspurgeometrie und -topologie auf der Basis von Fahrzeugtrajektorien und Umgebungsinformationen

Von der Fakultät für Bauingenieurwesen und Geodäsie
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades
Doktor-Ingenieur (Dr.-Ing.)
genehmigte Dissertation

Vorgelegt von

M. Sc. Oliver Bertin Röth

Geboren am 28.12.1988 in Essen

München 2018

Verlag der Bayerischen Akademie der Wissenschaften

ISSN 0065-5325 ISBN 978-3-7696-5236-9

Diese Arbeit ist gleichzeitig veröffentlicht in:
Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Universität Hannover
ISSN 0174-1454, Nr. 331, Hannover 2018

Adresse der DGK:

Ausschuss Geodäsie der Bayerischen Akademie der Wissenschaften (DGK)

Alfons-Goppel-Straße 11 ● D – 80 539 München

Telefon +49 – 331 – 288 1685 ● Telefax +49 – 331 – 288 1759

E-Mail post@dgk.badw.de ● http://www.dgk.badw.de

Prüfungskommission:

Vorsitzender: Prof. Dr.-Ing. habil. Jürgen Müller

Referent: Prof. Dr.-Ing. Claus Brenner

Korreferenten: Prof. Dr.-Ing. Christian Heipke
Prof. Dr.-Ing. Christoph Stiller (Karlsruher Institut für Technologie)

Tag der mündlichen Prüfung: 11.07.2018

ISSN 0065-5325 ISBN 978-3-7696-5236-9

Thesis Outline

Regarding current and future systems such as autonomously driving vehicles or complex driver assistance systems, highly accurate maps gain more and more importance. While conventional navigation maps are widely available today, the fully automated generation of highly accurate maps is subject to many research activities. Those maps are incorporating geometrical and topological information on a lane-level in the order of centimeters. Currently, a lot of different high-precision sensors like laser scanners are used to collect surrounding data which are finally fused semi-automatically with a huge amount of human post processing into a map. In order to create those maps more economically, the generation of highly accurate maps based on cheap sensor data is in the limelight of many scientific contributions.

In this thesis, a new method is developed which describes streets and crossings on a lane-level with appropriate models. These models are then optimized regarding a set of training data by using a reversible jump Markov Chain Monte Carlo approach. One advantage of this method is that not only the training data can be taken into account but also some prior knowledge about the shape of such models. By using specified operations, modifications of the models and therefore of their state can be proposed which are accepted or rejected with a certain acceptance probability. The development of the models, the specification of the operations, the calculation of the acceptance probability, and the evaluation of the algorithm are parts of this thesis.

Based on three trajectory datasets recorded by a vehicle fleet the new approach is applied. Those datasets each differ in accuracy levels with distinct dimensions of positioning errors. The results are evaluated against comparable approaches from the literature and against a high-precision ground-truth map. To this end, appropriate evaluation methods to point out the suitability and performance of the fully automated construction method are used. The result of those evaluations implies that the new approach sets the new benchmark concerning the geometrical quality and topological completeness with respect to each precision level of the data. As a supplement to the prior mentioned datasets, more trajectories originating from other road users are added. Those datasets are created by observing other road users with a camera. By using those datasets, the quality of the results can be improved and the number of required passings through a street can be reduced.

Keywords: reversible jump Markov Chain Monte Carlo methods, swarm trajectories, sensor and surrounding information, highly accurate street maps, fully automated reconstruction

Kurzfassung der Dissertation

Im Kontext aktueller und zukünftiger Systeme, wie beispielsweise autonom fahrende Fahrzeuge oder komplexe Fahrerassistenzsysteme, gewinnen hochgenaue digitale Straßenkarten immer mehr an Bedeutung. Während herkömmliche Navigationskarten mittlerweile bei diversen Anbietern flächendeckend zugänglich sind, ist die automatisierte Generierung hochgenauer Karten derzeit Gegenstand der Forschung. Dabei umfasst derartiges Kartenmaterial Informationen bezüglich der Geometrie und Topologie von Straßen auf Fahrspurebene im Zentimeterbereich. Aktuell werden zur Erzeugung die Informationen verschiedenster hochgenauer Sensoren, wie beispielsweise Laserscanner, in aufwändiger teil-automatisierter Arbeit fusioniert. Um derartige Karten wirtschaftlich nutzen zu können, muss der Aufwand der Generierung reduziert werden. Daher liegt ein Schwerpunkt der Forschung auf der automatischen Generierung derartiger Karten aus verschiedenen Sensordaten.

In dieser Arbeit wird ein neues Verfahren entwickelt, welches in der Lage ist eine Straße bzw. Kreuzung auf Fahrspurebene mit geeigneten Modellen zu beschreiben. Diese Modelle werden anschließend unter Verwendung eines Reversible Jump Markov Chain Monte Carlo Verfahrens hinsichtlich einer Menge von Eingabedaten optimiert. Ein Vorteil dieser Herangehensweise ist, dass das Verfahren einerseits die Daten adaptieren und andererseits beliebiges Vorwissen über die *normale* Gestalt der Modelle berücksichtigen kann. Anhand festgelegter möglicher Operationen werden Änderungen an den Zuständen der Modelle vorgeschlagen und eine Akzeptanzwahrscheinlichkeit bestimmt, anhand welcher entschieden wird, ob die Änderungen angenommen oder verworfen werden. Die Entwicklung der Modelle, die Festlegung der Operationen, die Berechnungsvorschriften der Akzeptanzwahrscheinlichkeiten und die Evaluierung des Verfahrens sind Teile dieser Arbeit.

Basierend auf den Trajektorien einer Fahrzeugflotte, welche im Rahmen dieser Arbeit in verschiedenen Datensätzen mit unterschiedlichen Positionierungsfehlern aufgezeichnet wurden, wird der neue Ansatz zur Anwendung gebracht. Anschließend werden die Ergebnisse mit geeigneten Bewertungsmethoden gegen vergleichbare Verfahren aus der Literatur, sowie gegen eine hochgenaue Referenzkarte verglichen, um den Mehrwert der vollautomatisierten Generierung zu verdeutlichen. Die dabei gewonnene Erkenntnis ist, dass das neue Verfahren, basierend auf allen verschiedenen Eingabedatensätzen, Ergebnisse mit geringer Abweichung zur Referenz hinsichtlich der geometrischen Präzision und topologischen Vollständigkeit erzeugen kann. Zusätzlich hinzugenommen wurde außerdem noch eine Erweiterung der Datensätze um Trajektorien, welche von anderen Verkehrsteilnehmern stammen, die mit Hilfe einer Kamera beobachtet wurden. Dadurch konnte einerseits die Qualität weiter gesteigert und andererseits die benötigte Anzahl an Durchfahrten für eine gute Rekonstruktion gesenkt werden.

Schlagwörter: reversible jump Markov Chain Monte Carlo Verfahren, Schwarm-Trajektorien, Sensor- und Umgebungsinformationen, hochgenaue Straßenkarten, automatische Rekonstruktion

Inhaltsverzeichnis

Anhang

1. Einleitung

1.1. Motivation

Wer heutzutage an die Navigation eines Fahrzeugs denkt, hat in der Regel keine Assoziation mehr mit einer gedruckten Karte oder einem Kompass, sondern eher mit dem im Fahrzeug verbauten Bordcomputer oder seinem Smartphone. Der lange Weg zur satellitengestützten Navigation wird von Hofmann-Wellenhof u. a. (2007) beschrieben: 1958 wurde mit der Entwicklung des ersten Satellitennavigationssystem „Navy Navigation Satellite System“ (NNSS) durch die US-Marine begonnen (Defense Advanced Research Projects Agency, 2018). Das System verwendete sechs Satelliten um eine Genauigkeit von 500 m bis 15 m zu erreichen und wurde bis 1996 militärisch und zivil genutzt. 1973 begann das US-Verteidigungsministerium mit der Entwicklung des „Global Positioning System“ (GPS), welches 1985 das NNSS ablöste und 1995 voll funktionsfähig wurde. Dabei war das GPS für militärische Zwecke voll zugänglich, während für zivile Zwecke zunächst nur ein künstlich verschlechtertes Signal „Selective Availability“ (SA) mit einer Genauigkeit von bis zu 100 m zur Verfügung gestellt wurde. SA wurde im Jahre 2000 abgeschaltet, sodass auch für zivile Zwecke eine Positionierungsgenauigkeit von bis zu 10 m erreicht werden konnte. Das System nutzt heute 30 Satelliten in einer Flughöhe von ca. 20000 km.

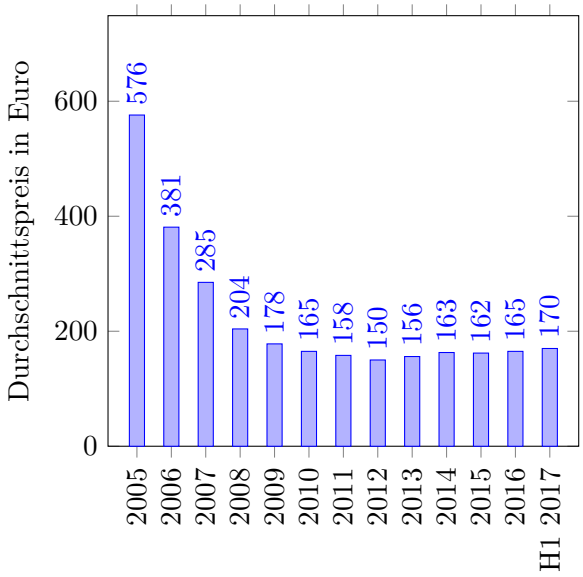


Abbildung 1.1.: Durchschnittspreis für verkaufte Navigationsgeräte in Deutschland von 2005 bis zum 1. Halbjahr 2017 (in Euro) (Quelle: statista.com)

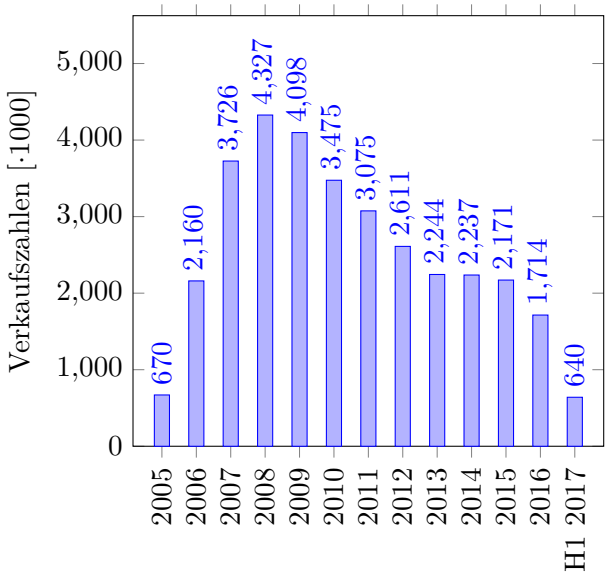


Abbildung 1.2.: Absatz von Navigationsgeräten in Deutschland von 2005 bis zum 1. Halbjahr 2017 (in 1000*) (Quelle: statista.com)*

Wie aus der Statistik in Abbildung 1.1 hervorgeht, wurde durch die Weiterentwicklung der Navigationstechnik und der Hardwarekomponenten das Navigationssystem über die Jahre preislich für den Einzelverbraucher erschwinglich. Im Jahr 2005 kostete ein Navigationsgerät noch durchschnittlich 576 Euro, während es im Jahr 2017 nur noch 170 Euro waren. Ab dem Jahr 2009 hält sich der Preis, trotz der allgemeinen Tendenz zum Preisverfall elektronischer Geräte, auf einem annähernd konstanten Niveau. Somit scheint ein gesellschaftlich akzeptabler Preis erreicht worden zu sein. Aus den in Abbildung 1.2 abgebildeten Absatzzahlen resultiert die in Abbildung 1.3 dargestellte Entwicklung des Umsatzes mit Navigationsgeräten, welcher durch die Kombination aus Preis und Absatz in den Jahren 2007 und 2008 am höchsten lag. Dies wird auch in Abbildung 1.4 deutlich, welche darstellt, wie groß der Anteil an privaten Haushalten mit einem Navigationssystem ist. Hier ist vom Jahr 2007 zu 2008 der größte relative Anstieg überhaupt verzeichnet.

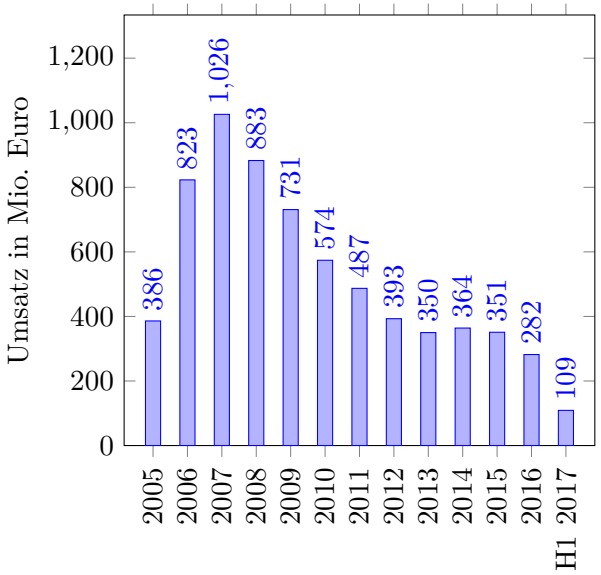


Abbildung 1.3.: Umsatz mit Navigationsgeräten in Deutschland im Zeitraum von 2005 bis zum 1. Halbjahr 2017 (in Millionen Euro) (Quelle: statista.com)

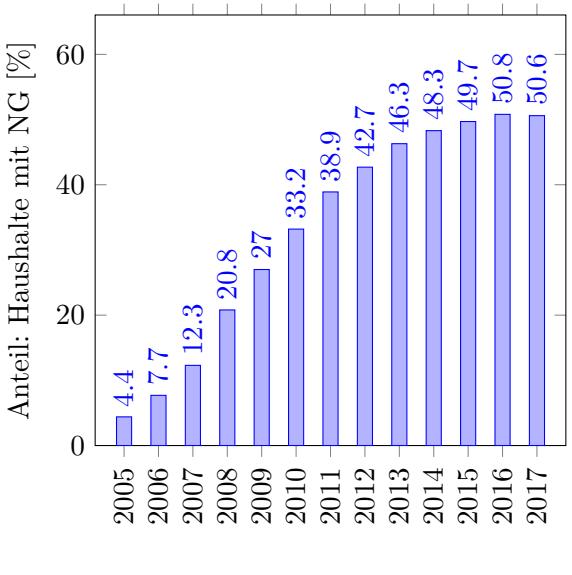


Abbildung 1.4.: Anteil der privaten Haushalte in Deutschland mit einem Navigationsgerät von 2005 bis 2017 (Quelle: statista.com)

Mittlerweile gehören Navigationssysteme zur Routenführung und Informationsdarbietung zur Grundausstattung eines neuen Fahrzeugs. Für die Integration zukünftiger Assistenzsysteme mit höheren Anforderungen an Präzision und Aktualität der Informationen müssen neue Wege begangen werden. Die Herausforderungen erstrecken sich von der spurgenauen Navigation, über vorausschauende Assistenzsysteme bis hin zum autonomen Fahren von Fahrzeugen. Letzteres ist derzeit ein Forschungsschwerpunkt vieler Unternehmen. Aus Abbildung 1.5 geht hervor, dass die Robert Bosch GmbH mit 958 angemeldeten Patenten zum Thema derzeit führend ist. Ebenso wird ersichtlich, wie viele verschiedene Unternehmen an dem Thema interessiert sind. Nicht überraschend ist daher das in Abbildung 1.6 prognostizierte stetige Wachstum, woraus hervorgeht, dass autonome Fahrfunktionen bis zum Jahr 2025 ein weltweites Marktvolumen von 26 Mrd. Euro bieten. Dabei sollen bis zum Jahr 2022 Fahrfunktionen, welche eine vollautomatisierte Fahrweise mit seltenen

Eingriffen des Fahrers ermöglichen (sog. „Level-4“), erreicht sein. Des Weiteren sollen bis zum Jahr 2025 Systeme auf dem Markt verfügbar sein, welche autonome Fahrfunktionen ohne Eingriffe des Fahrers (sog. „Level-5“) bieten.

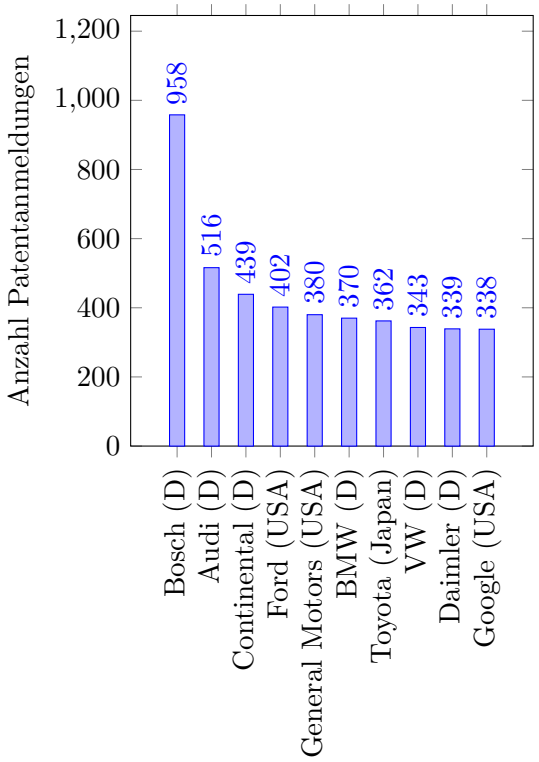


Abbildung 1.5.: Anzahl der Patentanmeldungen zum autonomen Fahren nach Unternehmen und Herkunftsland im Zeitraum der Jahre 2010 bis Juli 2017 (Quelle: statista.com)

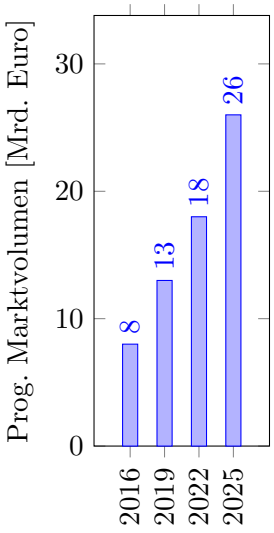


Abbildung 1.6.: Prognostiziertes weltweites Marktvolumen von autonomen Fahrfunktionen und Fahrassistenzsystemen in den Jahren 2016 bis 2025 (in Mrd. Euro) (Quelle: statista.com)

Um diesen Herausforderungen entgegen treten zu können, ist einer der Schlüssel das Vorhandensein hochgenauen Kartenmaterials mit Informationen bezüglich der Topologie und Geometrie im Zentimeterbereich. Um wirtschaftlich sinnvoll genutzt werden zu können, ist es von großem Belang, wie derartiges Kartenmaterial erzeugt werden kann. Aktuell werden hochspezialisierte Fahrzeuge genutzt, um mit Laserscanner, Radaren und anderen Sensoren Umfeldinformationen aufzuzeichnen. Diese werden teilweise automatisiert, allerdings auch mit viel händischer Bearbeitung zu einer hochgenauen Karte fusioniert. Somit ist sowohl die Datenerhebung, als auch die Auswertung sehr aufwändig und kostenintensiv. Aus diesem Grund befasst sich die vorliegende Arbeit mit der Entwicklung eines Verfahrens zur automatisierten Generierung einer hochgenauen Straßenkarten aus Sensorinformationen von Fahrzeugflotten. Diese *Crowd Sourcing* Daten werden von den meisten Fahrzeugen heutzutage standardmäßig aufgezeichnet und verwendet und sind somit in großem Stile kostengünstig verfügbar.

1.2. Zielsetzung der Arbeit

Zukünftig werden Fahrzeuge situationsabhängig Ego-Trajektorien unterschiedlicher Genauigkeit bereitstellen. Diese Trajektorien können einerseits in einer heutzutage üblichen Positionierungsgenauigkeit von $\sim 10\,\mathrm{m}$ und andererseits, durch hochgenaue Lokalisierung, von bis zu $\sim 10\,\mathrm{cm}$ übermittelt werden. Zusätzlich können diese Fahrzeuge unter Verwendung eines Kamerasystems andere Verkehrsteilnehmer beobachten und somit deren Trajektorie in ähnlicher Genauigkeit bestimmen.

Ziel dieser Arbeit ist die Ableitung impliziten Wissens aus derartigen Bewegungsdaten. Das bedeutet, es sollen Verfahren entwickelt werden, welche in der Lage sind topologische und geometrische Eigenschaften des Fahrzeugumfeldes aus den Daten zu extrahieren, ohne dieses Umfeld direkt sensorisch zu überwachen. Dabei sollen einerseits statische Informationen hinsichtlich der Straßen- und Kreuzungskonnektivität gewonnen werden, welche den Zusammenhang zwischen Fahrspuren einer Straßen bzw. ein- und ausführenden Fahrspuren einer Kreuzung beschreiben. Andererseits sollen individuelle, meist lokal gültige Informationen bezüglich der Fahrspurgeometrie, wie beispielsweise die Spurbreite oder -krümmung, generiert werden.

Als Eingabedaten für die zu entwickelnden Verfahren liegen die Ego-Trajektorien einer *Fahrzeugflotte*, aufgenommen auf dem Hildesheimer Stadtgebiet, vor. Die Messdaten wurden dabei in drei unterschiedlichen Genauigkeitsstufen mit verschiedenen Positionierungsfehlern aufgezeichnet. Aufgrund dieser Daten soll untersucht werden, welcher Detailgrad an abgeleiteten Informationen auf Grundlage welcher Eingabedaten erzielt werden kann. Ebenso Gegenstand der Untersuchungen ist der potentielle Qualitätsgewinn auf Grund der Beobachtungen anderer Verkehrsteilnehmer. Des Weiteren sollen Versuche bezüglich der benötigten Anzahl an Eingabedaten durchgeführt werden, um zu bestimmen, welches Datenvolumen für eine derartige Extraktion nötig ist.

Zusätzlich sollen Evaluationverfahren aus der Literatur gewählt oder entwickelt werden, welche die Qualität der topologischen und geometrischen Informationen im Vergleich zu einer hochgenauen Referenzkarte bewerten können.

1.3. Wissenschaftliche Beiträge

Das in dieser Arbeit entwickelte Verfahren ist in der Lage das beschriebene Wissen bezüglich eines Straßennetzwerkes auf Fahrspurebene zu extrahieren. Ähnliche Verfahren wie beispielsweise von Chen u. a. (2010) beschreiben bestimmte Merkmale, wie z. B. die Fahrspurmittellinie, durch primitive Formen als Linien und Kreisbögen. Der neue Algorithmus hingegen verwendet skalierbare Modelle, welche für die Beschreibung von Straßen und Kreuzungen ausgelegt sind und eine umfassende Abbildung diverser Parameter ermöglichen. In verschiedenen Arbeiten wie z. B. von Rogers u. a. (1999) werden lokale Probleme beispielsweise unter der Anwendung von Clusteringverfahren gelöst und anschließend vereinigt. Dabei werden häufig umfangreiche Annahmen getroffen, um eine Konvergenz des Verfahrens zu gewährleisten. Das neue Verfahren verwendet eine Markov Chain Monte Carlo Methode, um die Aufgabe der Rekonstruktion global durch die Formulierung als

Optimierungsproblem zu lösen. Dabei werden in Abhängigkeit der Eingabedaten mehrere Modelle initialisiert und optimiert, wobei lediglich wenige Annahmen zur Einschränkung der Dimension der Modelle getroffen werden. Bei der Optimierung werden Abhängigkeiten zwischen verschiedenen Modellen berücksichtigt, um diese nicht nur an die lokale Situation anzupassen, sondern um übergreifend gewisse Charakteristika eines Modells in ein benachbartes fortzupflanzen.

1.4. Gliederung

In dieser Arbeit werden in Kapitel 2 und 3 zunächst verschiedene Grundlagen vermittelt. Kapitel 2 bezieht sich auf Verfahren und Definitionen, welche im Rahmen dieser Arbeit verwendet bzw. für die Spezifikation und Ableitung späterer Modelle benötigt werden. Kapitel 3 befasst sich zunächst mit der Fragestellung, weshalb der beschriebenen Aufgabenstellung mit einem Reversible Jump Markov Chain Monte Carlo Verfahren begegnet wird und warum sich *simplere* Vorgehen dieser Art nicht eignen. Anschließend werden die mathematischen Grundlagen zum Verständnis der Verfahren dargelegt.

In Kapitel 4 wird der Stand der Forschung bezüglich der Rekonstruktion digitaler Straßenkarten präsentiert. Dabei werden zunächst Vertreter behandelt, welche sich mit der Ableitung herkömmlicher fahrbahngenauer Straßenkarten von verschiedenen Sensordaten, im Speziellen GNSS Fahrzeugtrajektorien, befassen. Anschließend werden Algorithmen betrachtet, welche in der Lage sind Informationen auf Spurebene abzuleiten. Dabei werden die Verfahren unterschieden in jene, die lokal gültige Informationen für Fahrerassistenzsysteme generieren und solche die fahrspurgenaue digitalen Straßenkarten ableiten können.

Kapitel 5 beschreibt die drei Schritte des neuen, im Rahmen dieser Arbeit entwickelten Algorithmus zur vollautomatischen Konstruktion digitaler Straßenkarten. Dabei wird zunächst ein Verfahren präsentiert, welches die Eingabedaten in Form von Fahrzeugtrajektorien in verschiedene Verkehrssituationen segmentiert. Diese Unterteilung wird im zweiten Schritt genutzt, um, basierend auf einem Reversible Jump Markov Chain Monte Carlo (RJMCMC) Ansatz, Modelle anzulernen, welche eine herkömmliche fahrbahngenaue Karte repräsentieren. Abschließend werden, basierend auf der fahrbahngenauen Darstellung, Modelle eingeführt, welche eine derartige Karte auf Fahrspurebene beschreiben können. Diese werden ebenfalls mit einem RJMCMC Ansatz an die Eingabedaten angelernt.

Abschließend werden in Kapitel 6 die mit dem neuen Verfahren erzeugten Ergebnisse evaluiert. Dazu werden zunächst Bewertungen eingeführt, welche sowohl zur Beurteilung der fahrbahngenauen, als auch der fahrspurgenauen Straßenkarten herangezogen werden können. Anschließend werden zunächst für beide Repräsentationen Parameterstudien durchgeführt, um die benötigten optimalen Freiheitsgrade zu bestimmen. Dabei werden diese nicht auf die verwendeten Datensätze, sondern für die allgemeine Verwendung im Kontext der Anwendung optimiert. Abschließend werden die Ergebnisse erzeugt und jeweils gegen die Lösungen anderer Verfahren aus der Literatur verglichen. Zusätzlich werden Studien bezüglich der benötigten Menge an Eingabedaten und bezüglich der Qualität der Daten durchgeführt.

In Kapitel 7 werden die Ergebnisse und Erkenntnisse der Arbeit zusammengefasst und ein Ausblick auf mögliche weitere Arbeiten gegeben.

2. Grundlagen

Dieses Kapitel beschreibt die Grundlagen verschiedener mathematischer Bereiche, welche benötigt werden, um die im Verlaufe dieser Arbeit entwickelten Verfahren zu definieren bzw. deren Arbeitsweise nachzuvollziehen. In Abschnitt 2.1 werden zunächst die Grundbegriffe der Graphentheorie und in Abschnitt 2.1.2 mit dem Algorithmus von Dijkstra ein Verfahren zur Bestimmung optimaler Wege auf derartigen Strukturen dargelegt. Beides wird für die Definition und Verwendung von digitalen Straßenkarten benötigt. Anschließend wird in Abschnitt 2.2 der Algorithmus von Viterbi eingeführt, ein Verfahren zur Bestimmung des wahrscheinlichsten Pfades durch ein Hidden-Markov-Modell, welcher in dieser Arbeit verwendet wird, um zu rekonstruieren, von welchem Punkt einer digitalen Straßenkarte eine Position in Weltkoordinaten aufgezeichnet wurde. Abschließend wird das Simulated-Annealing-Verfahren eingeführt, ein Optimierungsverfahren zur Bestimmung eines globalen Optimums einer Funktion mit begrenztem Wissen über die Funktion selbst.

2.1. Graphentheorie

Dieser Abschnitt beinhaltet das für das Verständnis der Arbeit nötige Basiswissen zur Graphentheorie. In Abschnitt 2.1.1 werden die grundlegenden Begriffe und Strukturen zur Definition eines allgemeinen Graphen beschrieben, wobei sich die Darstellung auf die für diese Arbeit relevanten Bereiche beschränkt. Abschnitt 2.1.2 beinhaltet den *Algorithmus von Dijkstra*, welcher zur Suche von optimalen Pfaden in einem Graphen verwendet wird. Für weiterführende Erläuterungen wird auf die Arbeit von Pahl und Damrath (2000) verwiesen, woran sich die Ausführungen in diesem Abschnitt inhaltlich und strukturell orientieren.

2.1.1. Struktur

Ein *Graph* ist ein mathematisches Konstrukt, welches Beziehungen zwischen verschiedenen Elementen abbildet. Ein *Element* oder *Knoten* repräsentiert dabei ein allgemeines *Objekt*. Die Summe aller $P \in \mathbb{N}$ Knoten wird als *Knotenmenge* $V = \{v_1,...,v_P\}$ bezeichnet, welche den mathematischen Regeln der Mengenlehre unterliegt. Eine *Relation* $R \subseteq V \times V$ beschreibt die Beziehungen zwischen den Elementen der Knotenmenge. Im Rahmen dieser Arbeit werden *homogene binäre Relationen* betrachtet, welche immer genau zwei gleichartige Elemente in Beziehung setzen. Ein einzelner Eintrag einer derartigen Relation $r_{ij} = (v_i,v_j)$ wird als *gerichtete Kante*, v_i als *Vorgänger* von v_j und v_j als *Nachfolger* von v_i bezeichnet. Existiert die umgekehrte Beziehung $r_{ji} = (v_j,v_i)$ ebenfalls, wird die Kante *ungerichtet* genannt. Da die Menge der Knoten endlich ist, kann die Relation vollständig durch die binäre Matrix $M \in B^{PxP}$ mit $B = \{0,1\}$ beschrieben werden. Es existieren folgende Spezialfälle von Relationen:

Definition 2.1 (Spezielle Relationen)**.**

$$\begin{aligned} \textit{Nullrelation } & R = \emptyset,\ M = 0^{P\times P} \\ \textit{Identitätsrelation } & R = \{(v,v)|v \in V\},\ M = I_P \\ \textit{Allrelation } & R = V \times V,\ M = 1^{P\times P} \\ \textit{Komplement } & \bar{R} = (V \times V)\backslash R \\ \textit{Transponierte } & R^T = \{(v,u)|(u,v) \in R\} \end{aligned}$$

Mithilfe dieser Definitionen können verschiedene Eigenschaften von Relationen definiert werden. Ist die Identitätsrelation I in der Relation R enthalten wird diese *reflexiv* genannt. Ist die Identitätsrelation I im Komplement $\bar{R}$ der Relation enthalten, wird diese als *antireflexiv* bezeichnet. Entspricht eine Relation ihrer Transponierten, $R = R^T$, wird sie *symmetrisch* genannt. Ist die Schnittmenge $R \cap R^T = \emptyset$ wird die Relation als *asymmetrisch* bezeichnet. Die Nullrelation wird als *leer* und die Allrelation als *vollständig* bezeichnet.

Auf Grundlage der Eigenschaften der Relation eines Graphen können folgende Graphentypen definiert werden:

Definition 2.2 (Schlichter Graph)**.** *Ein schlichter Graph $G = (V,R)$, bestehend aus der Knotenmenge V und der Relation R, ist die einfachste Form eines Graphen und unterliegt keinerlei Einschränkungen.*

Definition 2.3 (Einfacher Graph)**.** *Ein schlichter Graph $G = (V,R)$, bestehend aus der Knotenmenge V und einer antireflexiven und symmetrischen Relation R wird einfacher Graph genannt. Ein schlichter Graph kann durch die symmetrische Ergänzung der Relation und Entfernen von Schlingen (Definition 2.7) in einen einfachen Graphen überführt werden.*

Definition 2.4 (Gerichteter Graph)**.** *Ein schlichter Graph $G = (V,R)$, bestehend aus der Knotenmenge V und einer antireflexiven Relation R ohne Symmetrieeigenschaften (weder symmetrisch noch asymmetrisch) wird gerichteter Graph genannt.*

Definition 2.5 (Gewichteter Graph)**.** *Ein beliebiger Graph $G = (V,R,W)$ mit einer zusätzlichen Bewertungsmenge W wird gewichteter Graph genannt. Dabei wird jeder Kante der Relation ein Element der Bewertungsmenge zugewiesen, dessen Bedeutung im Kontext interpretiert werden muss.*

Zur algorithmischen Behandlung eines Graphen (z. B. zur Wegfindung) ist es nötig, diesen in eine gewisse *Struktur* zu überführen bzw. vorhandene Strukturen zu identifizieren. Bevor diese Strukturen definiert werden können, werden folgende Definitionen eingeführt:

Definition 2.6 (Weg)**.** *Eine Sequenz von Kanten $w_{[ae]} = \{(v_a,v_b),(v_b,v_c)...,(v_d,v_e)\}$ wird als Weg oder Kantenzug von v_a nach v_e bezeichnet, wobei die Länge die Anzahl an enthaltenen Kanten beschreibt. Die Menge der verbundenen Knoten $\{v_a,...,v_e\}$ heißt Knotenfolge.*

Definition 2.7 (Zyklus)**.** *Ein Weg $w_{[aa]} = \{(v_a,v_b),(v_b,v_c)...,(v_d,v_a)\}$ mit gleichem Start- und Endknoten heißt Zyklus. Ein Zyklus mit einer Länge größer eins wird als echter Zyklus, andernfalls als Schlinge bezeichnet. Wenn jeder Weg eines Graphen ein Zyklus ist, wird der Graph als zyklisch bezeichnet. Gibt es keinen Zyklus, so wird der Graph azyklisch genannt.*

Definition 2.8 (Zusammenhang)**.** *Existiert zwischen jedem Knotenpaar eines einfachen Graphen ein Weg, wird der Graph einfach zusammenhängend genannt. Gilt der Sachverhalt in einem gerichteten Graphen, wird der Graph streng zusammenhängend genannt. Ist jedes Knotenpaar von einem dritten Knoten aus erreichbar, wird der Graph als quasistreng zusammenhängend bezeichnet. Ein gerichteter Graph ist schwach zusammenhängend, wenn der aus ihm erzeugte einfache Graph einfach zusammenhängend ist.*

Im Folgenden werden unter Verwendung dieser Definitionen die relevanten Strukturen eingeführt. Ein einfacher, azyklischer Graph mit einfachem Zusammenhang ist ein *Baum* und besteht aus n Knoten und $k = n - 1$ Kanten. In einem Baum ist jeder Weg eindeutig. Existiert in einem schwach zusammenhängenden Graphen ein Knoten der von allen Knoten aus erreichbar ist, wird dieser als *Wurzel* bezeichnet, welche eindeutig ist, falls der Graph azyklisch ist. Verfügt ein Graph über mindestens eine Wurzel, wird er *Wurzelgraph* mit quasistrengem Zusammenhang genannt. Hat in einem Wurzelgraph mit genau einer Wurzel jeder Knoten höchstens einen Vorgänger wird er als *Wurzelbaum* bezeichnet.

2.1.2. Der Algorithmus von Dijkstra

Das bekannteste Verfahren zur Suche von optimalen Pfaden in gewichteten Graphen $G = (V,R,W)$ wurde von Dijkstra (1959) eingeführt, wobei das zu optimierende Kriterium durch die Art der Gewichtung im Graphen gegeben ist. Dabei wird der Graph von einem Startknoten s ausgehend in einen Wurzelbaum (vgl. Abschnitt 2.1.1) überführt, welcher solange aufgebaut wird, bis der Zielknoten z eingefügt wurde. Durch die Eindeutigkeit des Weges $w_{[sz]}$ von der Wurzel s zum Ziel z, ist die gefundene Lösung die gesuchte. Zum Aufbau des Baumes wird eine im Kontext des Graphen gültige Distanzfunktion benötigt, welche in der Lage ist, den *Abstand* eines Knotens zur Wurzel als Summe der Kantengewichte zu bewerten.

$$d(v) = \sum_{k \in w_{sv}} W(k) \tag{2.1}$$

In Algorithmus 1 ist der Ablauf des Algorithmus als Pseudocode dargestellt und soll im Folgenden detailliert betrachtet werden. Zur Initialisierung des Algorithmus befinden sich alle Knoten des Graphen, mit Ausnahme des Startknotens s, in der Menge noch nicht besuchter Knoten N. Außerdem befindet sich der Startknoten s in der Kandidatenliste K und steht als Wurzel im Suchbaum S. Die Kandidatenliste hat dabei die Struktur einer *Prioritätswarteschlange*, welche die vorhandenen Knoten anhand der Distanzfunktion (2.1) aufsteigend sortiert. Iterativ wird im Algorithmus immer das erste Elemente k aus der Kandidatenliste entfernt und alle Nachfolger im Graphen betrachtet (vgl. Zeile 15). Sollte der Nachfolger n noch nicht betrachtet worden sein, wird er aus N entfernt, mit Hilfe des Distanzfunktion in die Kandidatenliste übertragen und in den Suchbaum S eingegliedert (vgl. Zeilen 17 - 19). In Zeile 20 wird geprüft, ob sich der betrachtete Nachfolger in der Kandidatenliste befindet. Sollte dies der Fall sein, ist es möglich, dass sich der Knoten bereits vor der Iteration im Suchbaum befunden hat und über den aktuellen Knoten k optimaler erreichbar ist. Sollte dies der Fall sein, wird die bisherige Beziehung aus dem Baum

entfernt und der Nachbar neu eingefügt (vgl. Zeilen 21 - 23). Wird während dieser Iteration der Zielknoten z aus der Kandidatenliste entfernt, ist der optimale Weg w_{sz} gefunden (vgl. Zeilen 8 - 14).

Da die Iteration abbricht, sobald der Zielknoten gefunden wurde, wird in den meisten Fällen nur ein Teil des Graphen in einen Wurzelbaum überführt. Die asymptotische Laufzeit des Algorithmus hängt von der Implementierung der Kandidatenliste ab. Die beste Laufzeit wird bei der Verwendung eines *Fibonacci Heaps* erzielt und ist $O(n \cdot \log(n) + m)$, mit der Knotenanzahl n und der Kantenanzahl m.

Algorithmus 1 : Der Algorithmus von Dijkstra.

Eingabe : Gewichteter Graph $G = (V,R,W)$, mit $W : R \to \mathbb{R}$

Eingabe : Startknoten s, Zielknoten z

```
1  N = V/s                                          // Menge noch nicht besuchter Knoten
2  K = {s}                                          // Kandidatenliste
3  d : V → ℝ, mit d(s) = 0                          // Distanzfunktion
4  S = ({s}; {})                                    // Suchbaum
5  w_sz = {z}                                       // Weg s → z
6  while K nicht leer do
7      k ← entferne k aus K
8      if k = z then
9          v = z
10         while v besitzt einen Vorgänger do
11             v = Vorgänger von v in S
12             füge v an erster Stelle von w_sz ein
13         end
14         break                                    // Abbruch
15     for alle Nachfolger n von k in G do
16         if n ∈ N then
17             n ← entferne n aus N
18             füge n in K sowie n und (k,n) in S ein
19             d(n) = d(k) + W((k,n))
20         if n ∈ K then
21             if d(k) + W((k,n)) < d(n) then
22                 d(n) = d(k) + W((k,n))
23                 lösche n aus S und füge n sowie (k,n) in S ein
24     end
25 end
26 return S
```

2.2. Der Algorithmus von Viterbi

Der Algorithmus von Viterbi (Viterbi, 1967) ist ein Verfahren zur Bestimmung der wahrscheinlichsten Abfolge verborgener Zustände eines *Hidden-Markov-Models* aus einer Sequenz emittierter Ausgangssymbole.

Ein Hidden-Markov-Model (HMM) ist ein stochastisches Modell, welches ein System λ mit Hilfe einer Markov-Kette (vgl. Abschnitt 3.3.1) modelliert. Das System kann dabei mit gewissen Wahrscheinlichkeiten zwischen seinen Zuständen Z wechseln, wobei ein Teil der Zustände versteckt (*hidden*) ist und somit nicht explizit beobachtet werden kann. Stattdessen werden mit bestimmten Wahrscheinlichkeiten von jedem Zustand verschiedene Ausgangssymbole emittiert. Durch die Modellierung als Markov-Kette hängt der Übergang zwischen zwei Zuständen nicht von den vorherig eingenommenen Zuständen ab. Somit kann zu jedem emittierten Symbol mit einer bestimmten Wahrscheinlichkeit gefolgert werden, in welchem Zustand sich das System befunden hat.

Definition 2.9 (Hidden-Markov-Model)**.** *Ein Hidden-Markov-Model ist gegeben durch das 5-Tupel* $\lambda = (Z,E,A,B,\pi)$*, wobei*

$$
\begin{aligned}
Z &= \text{Menge der verborgenen Zustände} \\
E &= \text{Emissionsalphabet} \\
A &= \text{Zustandsübergangsmatrix} \\
B &= \text{Emissionssmatrix} \\
\pi &= \text{Anfangsverteilung}
\end{aligned}
$$

Wenn A und B sich nicht über die Zeit ändern, heißt das HMM stationär.

Die Aufgabe des Algorithmus besteht darin, aus einer Sequenz emittierter Ausgangssymbole $e = \{e_1,...,e_T\} \in E$ die wahrscheinlichste Abfolge durchlaufener Zustände $z = \{z_1,...,z_T\} \in Z$ zu bestimmen. Das bedeutet, dass die Wahrscheinlichkeit $P(z|e)$ maximiert werden muss. Da in der bedingten Wahrscheinlichkeit

$$P(z|e) = \frac{P(e,z)}{P(e)}$$

der Nenner nicht von den durchlaufenen Zuständen abhängt, ist die Aufgabenstellung äquivalent zur Maximierung der Wahrscheinlichkeit $P(e,z)$ in Abhängigkeit von z.

Der Algorithmus besteht aus drei Phasen, zunächst werden alle benötigten Anfangswahrscheinlichkeiten bestimmt (Initialisierung), als nächstes die möglichen Pfade durch das System berechnet und abschließend der gewählte Pfad zurückverfolgt (Backtracking). Die *lokale Pfadwahrscheinlichkeit* $\vartheta_t(i)$ gibt dabei die Wahrscheinlichkeit an, dass sich das System zum Zeitpunkt t im Zustand z_i befindet.

$$\vartheta_t(i) = \max_{z \in Z} P(e_1,...,e_t; z_1,...,z_t = z)$$

Des Weiteren wird in $\psi_t(i)$ gespeichert, welcher Vorgängerzustand $\vartheta_t(i)$ maximiert hat.

Initialisiert werden die lokalen Pfadwahrscheinlichkeiten durch die Emissionswahrscheinlichkeiten des Ausgangssymbols von den jeweiligen Zuständen und deren Startwahrscheinlichkeiten:

$$\vartheta_1(i) = b_i(e_1) \cdot \pi_i$$

Für jeden Zustand und Zeitpunkt wird während der Pfadbestimmung die lokale Pfadwahrscheinlichkeit ermittelt. Diese ergibt sich aus der Pfadwahrscheinlichkeit des vorherigen Zustandes, der Übergangswahrscheinlichkeit zum aktuellen Zustand und der Emissionswahrscheinlichkeit des Ausgangssymbols. Da es mehrere Pfade geben kann, wird über diesen Ausdruck maximiert:

$$\vartheta_t(i) = \max_j \Big(\vartheta_{t-1}(j) \cdot a_{ji} \cdot b_i(e_t)\Big)$$

Der Zustand, welcher zur Maximierung beigetragen hat, wird in $\psi_t(i)$ abgelegt.

Wurden alle Pfadwahrscheinlichkeiten berechnet, wird für den letzten Zeitpunkt $t = T$ der Zustand z_T mit der größten Wahrscheinlichkeit ausgewählt und der Pfad mit Hilfe von ψ_T rückwärts (*Backtracking*) ermittelt.

Zur Laufzeitbestimmung hängen die Anzahl der Operationen von der Menge der Zustände $|Z|$ und von der Länge der Sequenz $|e|$ bzw. T ab. Die Initialisierung benötigt einmalig $|Z|$ und jede Pfadberechnung $|Z|^2$ Operationen. Somit werden insgesamt $(T-1)|Z|^2 + |Z|$ Operationen benötigt, welche in $O(T \cdot |Z|^2)$ liegen.

2.3. Simulated Annealing

Simulated Annealing (Kirkpatrick u. a., 1983) ist ein stochastisches Verfahren zur Suche eines Parametersatzes $x = (x_1,...x_n) \in X$ welcher eine Funktion $f(x)$ optimiert. Diese Funktion wird Kostenfunktion genannt und bewertet die Güte einer Konfiguration von Parametern hinsichtlich einer Problemstellung. Diese Kostenfunktion kann dabei beliebig komplex und hochdimensional sein. Eine klassische Herangehensweise zur Lösung solcher Probleme ist beispielsweise der *Bergsteigeralgorithmus*, welcher sich entlang der Kostenfunktion in Richtung des nächsten Optimums bewegt, dabei allerdings nicht in der Lage ist, lokale Optima zu verlassen. Im Simulated Annealing ist die Möglichkeit gegeben derartige Optima durch die Annahme suboptimalerer Lösungen zu verlassen. Die Idee ist angelehnt an den physikalischen Abkühlungsprozess (*annealing*) eines Systems, wobei sich das System seinem energetisch günstigsten Zustand (optimalen) annähert. Hierbei werden in Abhängigkeit der Temperatur τ energetisch schlechtere Zustände akzeptiert, um die Möglichkeit zu erhalten, den energetisch besten Zustand zu erreichen. Physikalisch hat das Metall in einem beliebigen Zustand die Energie e, welche sich bei einer Veränderung um Δe ändert. Gilt $\Delta e < 0$ ist ein energetisch günstigerer Zustand erreicht, welcher akzeptiert wird. In Abhängigkeit der Temperatur ist es allerdings *möglich* (modelliert durch eine Wahrscheinlichkeit), dass $\Delta e > 0$ gilt und dieser energetisch höhere Zustand erreicht wird. Modelliert wird diese Möglichkeit mit der

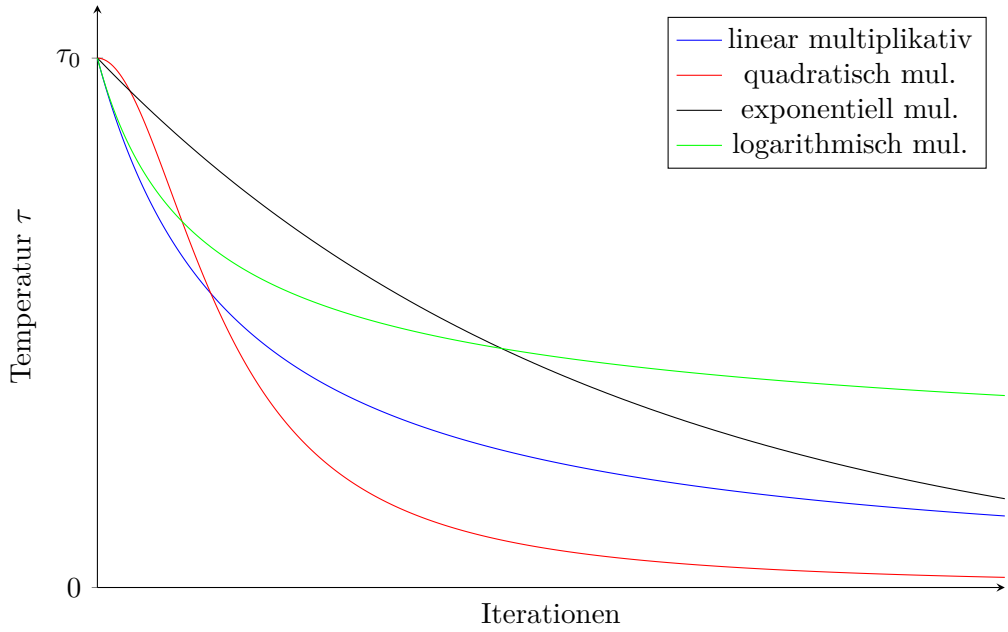


Abbildung 2.1.: Beispielhafte Abkühlfunktionen für das Simulated Annealing.

Boltzmann Konstante κ, so dass ein derartiger Zustand mit Wahrscheinlichkeit $e^{-\frac{\Delta e}{\kappa\ \tau}}$ angenommen wird:

$$e^{-\frac{\Delta e}{\kappa\ \tau}} > x,\ x \sim \text{Unif}^{a}(0,1)$$

In einer nicht-physikalischen Anwendung wird die Temperatur ersetzt durch eine monoton fallende Funktion und die Energie durch den Funktionswert der Kostenfunktion. Der Algorithmus wird an einem zufälligen Startpunkt $x \in X$ initialisiert. Anschließend werden iterativ zufällige neue Lösungen

$$x^{t+1} = x^t + u, \text{ mit } u \in U$$

in einer definierten Umgebung U um die aktuelle Konfiguration generiert. Ist der Funktionswert der Kostenfunktion $f(x^{t+1})$ besser als $f(x^t)$ wird die neue Konfiguration als aktuelle Konfiguration übernommen, andernfalls wird sie nur mit einer gewissen *Akzeptanzwahrscheinlichkeit*

$$A(x^t,x^{t+1}) = \exp\Big(-\frac{f(x^{t+1}) - f(x^t)}{\tau_t}\Big)$$

übernommen. Dabei ist τ_t eine monoton fallende Folge mit dem Startwert τ_0, welche die Temperatur des System repräsentiert. Die Akzeptanzwahrscheinlichkeit ist somit für kleinere Verschlechterungen größer und nimmt mit der Zeit generell ab. Durch diese Möglichkeit Verschlechterungen zu akzeptieren, ist das Verfahren in der Lage lokale Optima der Kostenfunktion zu überwinden. Im Kontext der Verwendung muss festgelegt werden, wie sich das System, ausgedrückt durch τ_t, abkühlt. Beispielsweise kann eine Abkühlfunktion verwendet werden, bei der die initiale Temperatur

[a] vgl. Anhang B.2

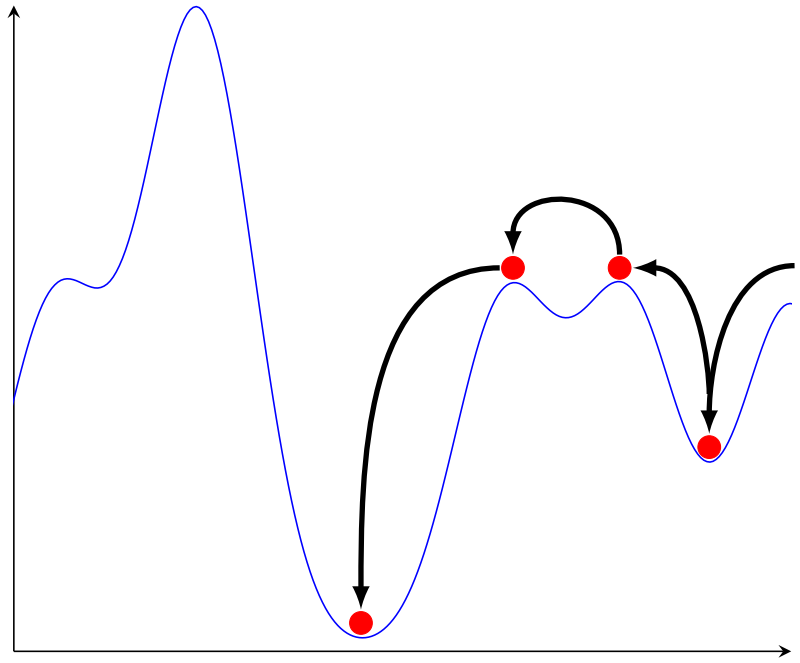


Abbildung 2.2.: Visualisierung einer Minimierungsaufgabe mit Simulated Annealing.

τ_0 mit einem Faktor multipliziert wird, welcher sich invers proportional zur Anzahl an Iterationen bzw. zum Quadrat der Anzahl an Iterationen verhält:

$$\tau_t^{\text{lin}} = \tau_0 \frac{1}{1+\alpha t},\ \alpha > 0$$
$$\tau_t^{\text{quad}} = \tau_0 \frac{1}{1+\alpha t^2},\ \alpha > 0$$

Alternativ wird von Kirkpatrick u. a. (1983) bzw. von Aarts und Korst (1989) eine Abkühlfunktion beschrieben, in welcher sich der multiplikative Faktor exponentiell bzw. logarithmisch mit der Anzahl an Iterationen entwickelt:

$$\tau_t^{\text{exp}} = \tau_0 \cdot \alpha^t,\ 0 > \alpha > 1$$
$$\tau_t^{\text{log}} = \tau_0 \frac{1}{1+\alpha \log(1+t)},\ \alpha > 0$$

Die vier beschriebenen Abkühlfunktionen sind beispielhaft in Abbildung 2.1 mit $\alpha = 0.8$ dargestellt.

Das allgemeine Vorgehen ist in Abbildung 2.2 für eine Minimierungsaufgabe mit einem Parameter visualisiert. Die Kostenfunktion ist in blau und die Konfigurationen in rot dargestellt. Dabei wird zunächst ein lokales Minimum erreicht, welches überwunden werden kann, um das globale Minimum zu erreichen.

3. Unüberwachtes Lernen

Die Aufgabe des unüberwachten Lernens besteht darin, zu einer erhobenen Datenmenge X eine mathematische Modellierung zu finden, welche die Daten möglichst gut abbildet und gleichzeitig eine Generalisierung der Daten ermöglicht. Dies bedeutet, dass die Modellierung auch nicht erhobene Daten aus der selben Quelle abbilden muss. In diesem Kapitel werden Verfahren betrachtet, welche in der Lage sind die Daten durch entsprechende Modelle zu beschreiben. Die Darstellungen in diesem Kapitel orientieren sich inhaltlich und strukturell an den Ausarbeitungen von Bishop (2006), Lauer (2004) und Plehn (2014).

3.1. Dichteschätzung

Allgemein kann die Modellierungsaufgabe durch die Beschreibung eines Modells als Wahrscheinlichkeitsverteilung, aus welcher die Trainingsdaten generiert wurden, gelöst werden. Dies kann beispielsweise mit Hilfe einer *Dichteschätzung* geschehen. Je nach Anforderungen an das Modell und die Art der Daten kommen spezialisiertere Verfahren wie die Cluster- (Kaufman und Rousseeuw, 1990) und Ausreißeranalyse (Beniger u. a., 1980) o.ä. zum Einsatz.

Bei der Dichteschätzung wird versucht, zu einer Stichprobe $X = \{x_1,...,x_n\}$, deren Elemente unabhängige und identisch verteilte Realisierungen aus einer nicht bekannten Verteilung P mit Dichte $p(\cdot)$ sind, eine neue Wahrscheinlichkeitsverteilung Q mit Dichte $q(\cdot)$ zu erzeugen, welche sich möglichst geringfügig von P unterscheidet. Der Unterschied kann über die Dichten mit Hilfe der Kullback-Leibler-Divergenz (Vapnik, 1995) ausgedrückt werden:

$$\mathrm{KL}(p,q) = \int p(t) \cdot \log \frac{p(t)}{q(t)} \mathrm{d}t \geq 0 \tag{3.1}$$

Es kann gezeigt werden, dass die erforderliche Minimierung von Gleichung (3.1) gleichbedeutend ist mit der Maximierung des Daten-Log-Likelihood zwischen der Stichprobe und der Dichte q:

$$LL(x_1,...,x_n|q) = \sum_{i=1}^{n} \log\, q(x_i) \tag{3.2}$$

Bei der Lösung gestaltet sich allerdings das in Abbildung 3.1 dargestellte *Overfitting*-Phänomen als problematisch, welches eine unbegrenzte Annäherung des Daten-Log-Likelihood an die Stichprobe beschreibt. Abbildung 3.1 zeigt eine Folge von drei Dichten, welche der Stichprobe (Punkte) immer weiter angepasst werden. Während Abbildung 3.1a keine *visuelle* Zuordnung zwischen den Daten und der Dichte ermöglicht, zeigt Abbildung 3.1b eine Verteilung, welche die Stichprobe gut abbildet. In Abbildung 3.1c ist eine überangepasste Dichte dargestellt, da die Verteilung immer, außer auf den

Daten der Stichprobe, eine Wahrscheinlichkeit von Null angibt. Dadurch ist keine Generalisierung der Daten gegeben.

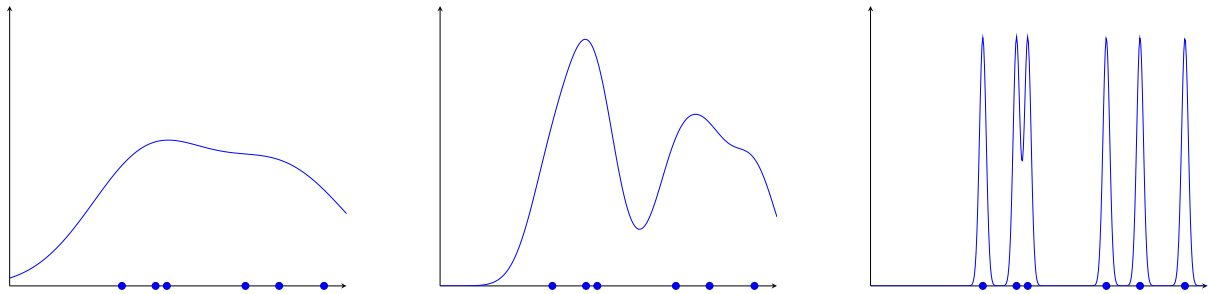


Abbildung 3.1.: Drei mögliche Dichten (a)-(c), welche bei der Maximierung des Daten-Log-Likelihood betrachtet werden könnten, wobei das Ziel die Adaption der erzeugenden Verteilung der Stichprobe X (Punkte) ist. Es ist ersichtlich, dass die in (c) dargestellte Dichte die Daten und nicht die gesuchte Verteilung beschreibt und somit überangepasst ist.

Diesem Verhalten kann mit zwei Maßnahmen entgegengewirkt werden:

1. Der Hypothesenraum H beschreibt die Menge aller Dichten q, welche als Lösung des Problems zulässig sind. Dieser Raum kann durch die Wahl geeigneter Grenzen so begrenzt werden, dass zum Overfitting neigende Dichten nicht ausgewählt werden können.
2. Mit der Einführung von Regularisierungstermen ist es möglich, zum Overfitting neigende Dichten nicht komplett auszuschließen, sondern zu benachteiligen. Dies geschieht durch die Addition bestimmter Terme zum Daten-Log-Likelihood. Diese Terme müssen so gewählt werden, dass zum Overfitting neigende Dichten bestraft werden und stochastisch seltener als Lösung ausgewählt werden können.

Zur Lösung des Problems unter Berücksichtigung des Anpassungsproblems werden im Folgenden mehrere mögliche Herangehensweisen dargelegt.

3.1.1. Parametrisierte Schätzer

Bei der Verwendung parametrisierter Dichten zur Dichteschätzung wird der Hypothesenraum H definiert über alle zulässigen Dichten: $H = \{q(\cdot|\vartheta)|\vartheta \in \Theta\}$. Dabei ist ϑ der Parametersatz einer parametrisierten Wahrscheinlichkeitsdichte $q(\cdot|\vartheta)$ aus einer Menge aller möglicher Parametersätze Θ. Aufgabe der parametrisierten Schätzer ist die Bestimmung des optimalen Parametersatzes.

3.1.1.1. Maximum-Likelihood-Schätzer

Ziel ist es nach wie vor das Daten-Log-Likelihood zwischen der Dichtefunktion q und der Stichprobe X zu maximieren. Bei der Verwendung des Maximum-Likelihood-Schätzers (ML) geschieht dies durch die Bestimmung des besten Parametersatzes $\hat{\vartheta} \in \Theta$ der parametrisierten Dichte:

$$\hat{\vartheta}_{\mathrm{ML}} := \arg\max_{\vartheta \in \Theta} \sum_{i=1}^{n} \log\, q(x_i|\vartheta)$$

Damit ist die Dichte $q(\cdot|\hat{\vartheta}_{\mathrm{ML}})$ jene mit der höchsten Daten-Log-Likelihood. Dabei wird das Overfitting durch die konkrete Einschränkung des Hypothesenraums bezüglich der möglichen Parametersätze vermieden. Somit liefert diese Methode nur eine gute Lösung, wenn der Hypothesenraum passend gewählt wurde.

Z.B. kann durch die Annahme, dass eine Stichprobe univariat normalverteilt (vgl. Anhang B.2) ist der Hypothesenraum auf die entsprechenden Dichten eingeschränkt werden:

$$H = \{\mathrm{Norm}(\mu,\sigma^2)|\mu \in \mathbb{R},\sigma^2 \in \mathbb{R}_{>0}\}$$

Der ML-Schätzer kann auf Basis der Stichprobe $X = \{x_1,...,x_n\}$ bestimmt werden:

$$\hat{\mu}_{\mathrm{ML}} = \frac{1}{n}\sum_{i=1}^{n} x_i$$
$$\sigma^2_{\mathrm{ML}} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu}_{\mathrm{ML}})^2$$

Overfitting tritt hierbei ein, wenn die Stichprobe zu klein ist, in diesem Fall tendiert die Varianz gegen Null: $\sigma^2_{\mathrm{ML}} \to 0$ und die Normalverteilung entartet zu einem Peak.

3.1.1.2. Maximum-a-posteriori-Schätzer

Die zweite Möglichkeit der Regularisierung besteht in der Einführung passender Bewertungsterme. Beim Maximum-a-posteriori-Schätzer (MAP) werden diese genutzt, um bestimmte Ausprägungen des Parametersatzes zu bestrafen. Die Schätzung basiert auf der Posterioriverteilung $P(\vartheta|x_1,...,x_n)$ der Parameter, welche die Wahrscheinlichkeit für das Auftreten eines Parametersatzes unter der Voraussetzung einer Menge von Trainingsdaten beschreibt. Unter Verwendung des Satzes von Bayes erhält man folgende Formulierung:

$$P(\vartheta|x_1,...,x_n) = P(x_1,...,x_n|\vartheta) \cdot \frac{P(\vartheta)}{P(x_1,...,x_n)}$$

wobei $P(x_1,...,x_n|\vartheta)$ der Daten-Log-Likelihood entspricht und $P(x_1,...,x_n)$ unabhängig von dem zu optimierenden Parametersatz ϑ ist und somit ignoriert werden kann.

Bei der MAP-Schätzung wird die logarithmierte Posterioriwahrscheinlichkeit maximiert:

$$\hat{\vartheta}_{\text{MAP}} := \arg\max_{\vartheta\in\Theta}(\sum_{i=1}^{n}\log\, q(x_i|\vartheta) + \log\, P(\vartheta))$$

wobei der erste Teil der Funktion die Maximierung der Daten-Log-Likelihood und der zweite Teil die Maximierung der Prioriwahrscheinlichkeit beschreibt. Die Prioriwahrscheinlichkeit beschreibt eine Verteilung über den Raum der Parameter und ermöglicht damit die Bestrafung und Begünstigung bestimmter Ausprägungen. Für eine konstante Prioriwahrscheinlichkeit, also ohne Bevorzugung von Daten, ergibt sich der Maximum-Likelihood-Schätzer als Spezialfall.

Äquivalent zum oben betrachteten Beispiel der Normalverteilung kann die Aufgabe als MAP-Schätzung formuliert und die Neigung zum Overfitting vermieden werden. Dazu wird eine Prioriwahrscheinlichkeit eingeführt, welche die Verteilung von σ^2 beschreibt und somit kleinen Varianzwerten kleine Auswahlwahrscheinlichkeiten zuordnen kann. Dazu kann z. B. die Inverse Gammaverteilung (vgl. Anhang B.2) $\sigma^2 \sim \Gamma^{-1}(\alpha,\beta)$ genutzt werden, womit sich folgende Aufgabe und Lösung ergibt:

$$(\hat{\mu}_{\text{MAP}},\hat{\sigma}^2_{\text{MAP}}) = \arg\max_{\mu\in\mathbb{R},\sigma\in\mathbb{R}_{>0}}\Big(\sum_{i=1}^{n}\log \text{Norm}(\mu,\sigma^2) - (\alpha+1)\log\sigma^2 - \frac{\beta}{\sigma^2}\Big)$$

$$\hat{\mu}_{\text{MAP}} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\hat{\sigma}^2_{\text{MAP}} = \frac{\sum_{i=1}^{n}(x_i - \hat{\mu}_{\text{MAP}})^2 + 2\beta}{n + 2 + 2\alpha}$$

Ist der Stichprobenumfang groß, unterscheiden sich ML- und MAP-Schätzer wenig, für kleine Stichproben jedoch bestimmen die Parameter α und β und weniger die Daten den MAP-Schätzer.

Die betrachteten parametrisierten Schätzer kommen erfolgreich zum Einsatz, wenn ein Verteilungsmodell für die Daten gegeben ist. Wenn die Trainingsdatenmenge nicht zu klein ist, kommen die Verfahren bei kurzer Laufzeit und geringer Affinität zum Overfitting zu guten Ergebnissen. Problematisch wird es, wenn keine Informationen über die Datenquelle vorliegen, da somit bei der Wahl eines Modells die Möglichkeit besteht, ein zu einfaches Modell zu wählen. Durch die resultierende *Unteranpassung* des Modells kann die Stichprobe in der Regel nur mit großen Fehlern und mangelnder Generalisierung beschrieben werden. Andererseits kann das Modell auch zu komplex gewählt werden und zum Overfitting führen. Weiterführende Informationen zur Verwendung parametrisierter Dichte sind beispielsweise in der Arbeit von Stahel (2002) zu finden.

3.1.2. Nichtparametrisierte Schätzer

Alternativ zu den parametrisierten gibt es auch die Klasse der nichtparametrisierten Schätzer, dazu zählen beispielsweise die als Parzen-Fenster-Technik (Parzen, 1962) bekannten *kernbasierten* Schätzverfahren. Dabei wird nicht ein optimaler Parametersatz einer Dichte gesucht, sondern diese

wird durch eine Glättung der gegebenen Stichprobe unter Verwendung einer *Kernfunktion* $K(\cdot)$ erzeugt. Dabei gilt:

$$K(x) \geq 0$$
$$\int K(x)\ \mathrm{d}x = 1$$

Über einen Parameter $h \in \mathbb{R}_{>0}$, welcher den Grad der Glättung festlegt und die gegebene Kernfunktion K ergibt sich die Dichte über eine d-dimensionalen Trainingsdatensatz zu:

$$q_{\text{Parzen}}(x) = \frac{1}{n \cdot h^d} \sum_{i=1}^{n} K\Big(\frac{x - x_i}{h}\Big) \tag{3.3}$$

Wie in Abbildung 3.2 ersichtlich, ist die korrekte Wahl von h entscheidend für die Qualität des Ergebnisses, wobei die Auswahl immer ein individuelles Problem darstellt und der Parameter nur empirisch bestimmt werden kann. Dadurch neigen derartige Verfahren stark zum Overfitting.

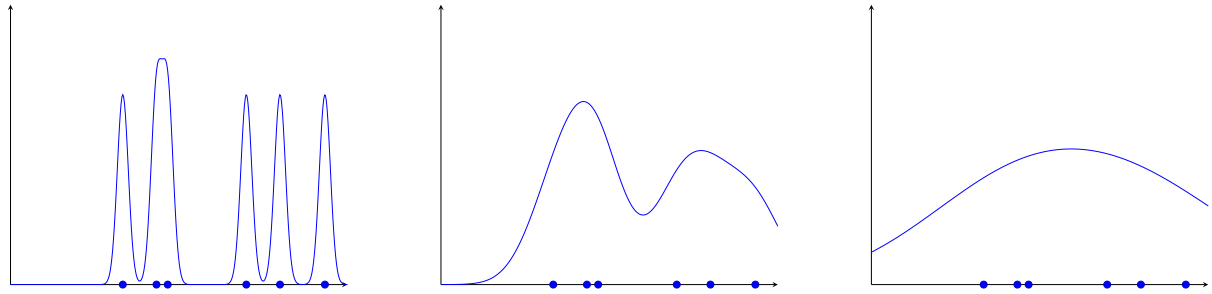


Abbildung 3.2.: Kernschätzung mit drei verschiedenen Größen des Glättungsparameters h. Wie in (a) ersichtlich wird, neigt das Verfahren bei einer zu kleinen Glättung zum Overfitting, während bei einer zu großen Glättung, wie in (c) dargestellt, keine Details der Stichprobe mehr wahrnehmbar sind.

Ein entscheidender Unterschied zu den parametrisierten Schätzern liegt darin, dass der durch Gleichung (3.3) induzierte Hypothesenraum von den Trainingsdaten und nicht von einem variablen Parametersatz abhängig ist.

Weiterführende Informationen zu den nichtparametrisierten Schätzern sind beispielsweise in den Arbeiten von Duda und Hart (1973), Devroye (1987) und Silverman (1986) zu finden.

3.1.3. Mischverteilungen

Die bisher betrachteten Verfahren haben individuelle Vor- und Nachteile. Mit den sogenannten *Mischverteilungen* werden die beschriebenen Stärken vereint, um ein parametrisiertes Modell ohne konkretes Wissen über die Verteilung der Daten erzeugen zu können.

Mischverteilungen sind gewichtete Summen von Wahrscheinlichkeitsdichten:

$$p(x) = \sum_{i=1}^{k} w_i \cdot p_i(x)$$
$$\text{mit } p(x) \geq 0, \int p(x)\mathrm{d}x = 1$$
$$\text{wobei } k \in \mathbb{N},\ w_1,...,w_k \in [0,1],\ \sum_{i=1}^{k} w_i = 1$$

Die verwendeten Einzeldichten p_i sind dabei beliebig. Um eine parametrisierte Mischverteilung erzeugen zu können werden im Folgenden die Einzeldichten auf parametrisierte Dichten beschränkt:

$$p(x|k,w_1,...,w_k,\vartheta_1,...,\vartheta_k) = \sum_{i=1}^{k} w_i \cdot p_i(x|\vartheta_i)$$

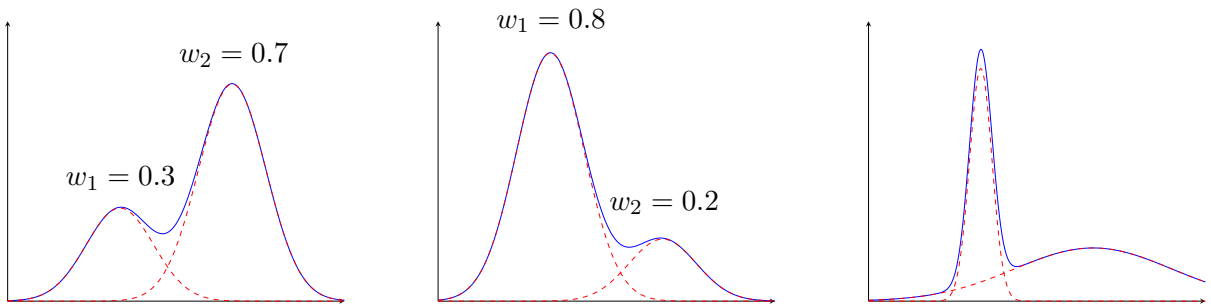


Abbildung 3.3.: Drei beispielhafte Mischverteilungen. Die Mischverteilungen sind in blau, die Einzeldichten in rot dargestellt.

In Abbildung 3.3 sind drei verschiedene Mischverteilungen dargestellt. In 3.3a und 3.3b wird ersichtlich, dass eine derartige Verteilung sehr flexibel ist und durch einfaches Verschieben der Gewichte stark variiert werden kann. Durch das Verändern der Parameter der Einzeldichten können auch beispielsweise breite Verteilungen, wie in Abbildung 3.3c abgebildet, erzeugt werden.

Anhand Gleichung (3.3) können die wesentliche Vorteile von Mischverteilungen verdeutlicht werden. Im Gegensatz zu den nichtparametrisierten Schätzern hängt die Anzahl der Einzeldichten nicht von der Größe des Trainingsdatensatzes ab, wodurch die Modellierung weniger komplex und der Hypothesenraum stark eingeschränkt wird. Zusätzlich kann jede der einzelnen Dichten andersartig sein und deren Parameter unabhängig von den Daten bestimmt werden.

Weiterführende Informationen zu Mischverteilungen sind beispielsweise in den Ausarbeitungen von Titterington u. a. (1985), Hand u. a. (1989) und McLachlan und Peel (2000) zu finden.

3.1.4. Zusammenfassung

In den Abschnitten 3.1.1 - 3.1.3 wurden verschiedene Realisierungen der Dichteschätzung betrachtet. Zunächst wurden parametrisierte Schätzer untersucht, welche einerseits schnell anzulernen sind und nicht zum Overfitting neigen, allerdings andererseits eine Modellvorgabe benötigen und somit nicht flexibel sind. Anschließend wurde ein nichtparametrisierter Schätzer betrachtet, welcher sich einerseits ohne Lernprozess flexibel an die Trainingsdaten anpassen lässt, andererseits jedoch einen schwer zu studierenden Glättungsparameter benötigt und stark zum Overfitting neigt. Um die Vorteile beider Schätzer zu vereinen wurden abschließend Mischverteilungen untersucht, welche zum einen sehr flexibel sind und jede beliebige Verteilung erzeugen können, zum anderen jedoch sehr aufwändig anzulernen sind, Overfitting nicht ausschließen können und eine Modellgröße als Vorgabe benötigen.

Um diese Eigenschaften der Mischverteilungen noch zu verbessern ist es von großem Belang wie der Anlernprozess durchgeführt wird. Daher wird in den nächsten Abschnitten die Modellbildung basierend auf Monte Carlo Methoden eingeführt. Die grundlegende Idee dabei ist, dass die zu lernenden Parameter der Einzeldichten jeweils einer Wahrscheinlichkeitsverteilung unterliegen, welche durch die Trainingsdaten impliziert wird. Somit erfolgt das Lernen der Parameter durch die Realisierung von Stichproben aus diesen nicht bekannten Verteilungen.

3.2. Monte Carlo Methoden

Die Monte-Carlo-Simulation ist ein Verfahren zur numerischen Lösung von Problemen und basiert auf dem *Gesetz der großen Zahlen*. Das bedeutet, dass das numerisch erzielte Ergebnis bei einer Vielzahl an experimentellen Auswertungen gegen die korrekte Lösung konvergiert. Die Einsatzmöglichkeiten derartiger Methoden sind vielfältig, im Rahmen dieser Arbeit werden sie eingesetzt, um die Verteilungseigenschaften von Zufallsvariablen zu untersuchen.

In diesem Abschnitt werden zunächst mit der *Inversions-* und der *Verwerfungsmethode* zwei Vertreter der Methoden betrachtet, welche in der Lage sind Stichproben aus Wahrscheinlichkeitsverteilungen zu generieren.

3.2.1. Inversionsmethode

Die Inversionsmethode (Schmeiser und Devroye, 1988)(Mueller-Gronbach u. a., 2012) ist ein Verfahren, welches in der Lage ist, Zufallszahlen aus einer gegeben Wahrscheinlichkeitsverteilung zu erzeugen. Die grundlegende Idee dabei ist, einen Zusammenhang zwischen der Inversen dieser Verteilung und einer Gleichverteilung (vgl. Abschnitt B.2) auf dem Intervall [0,1] herzustellen, sodass eine gleichverteilte Zufallsvariable auf die gegebene Wahrscheinlichkeitsverteilung transformiert werden kann.

Gegeben sei eine Verteilungsfunktion $F : \mathbb{R} \to \mathbb{R}$ einer Zufallsvariablen X. Zu dieser Funktion muss die Inverse, welche auch als *Quantilfunktion* bezeichnet wird, bestimmt werden:

$$F^{-1}(x) = \inf\{u \in \mathbb{R} | F(u) \geq x\}$$

Zusätzlich wird eine gleichverteilte Zufallsgröße u auf dem Intervall [0,1] benötigt, welche als Funktionswert der Inversen verwendet wird. Auf diese Weise wird x erhalten

$$x := F^{-1}(u)$$

wobei x nun eine zufällige Realisierung von F darstellt. Eingeschränkt wird die Anwendbarkeit durch die benötigte Umkehrfunktion, da diese in vielen Fällen nicht, oder sehr schwierig, bestimmt werden kann.

3.2.2. Verwerfungsmethode

Die Verwerfungsmethode (Schmeiser und Devroye, 1988)(Mueller-Gronbach u. a., 2012) ist ein Verfahren zur Erzeugung von Zufallszahlen aus einer gegebenen Verteilung, kann allerdings im Gegensatz zur Inversionsmethode (vgl. Abschnitt 3.2.1) auch Verteilungen verwenden, deren Inverse nicht bestimmt werden kann.

Gegeben sei eine Verteilungsfunktion $F : \mathbb{R} \to \mathbb{R}$ einer Zufallsvariablen X mit Dichte f, sowie eine Verteilungsfunktion H mit Dichte h für die H^{-1} existiert, sodass unter Anwendung der Inversionsmethode Zufallszahlen generiert werden können. Zur Bestimmung einer Zufallszahl werden nun solange gleichverteilte Standardzufallszahlen a_i und Zufallszahlen b_i von H bestimmt, bis gilt:

$$a_n < p$$
$$p = \frac{f(b_n)}{k \cdot h(b_n)}$$
$$\text{wobei } k \in \mathbb{R}, \text{ so dass } f(x) \leq k \cdot h(x) \; \forall x \in \mathbb{R}$$

Alle Zufallszahlen die diesem Kriterium nicht genügen werden *verworfen*. Auf diese Weise kann eine Stichprobe erzeugt werden, welche F genügt. Der Vorfaktor k wird benötigt, da es sich bei f und h um Dichten handelt und die Fläche unter einer Dichte immer 1 ist. Somit würde es ohne den Vorfaktor Werte geben, für welche $f(x) > h(x)$ gilt. Problematisch gestaltet sich bei der Verwerfungsmethode die Bestimmung einer passenden Verteilungsfunktion H, da der Zusammenhang $f(x) \leq k \cdot h(x)$ sichergestellt werden muss.

3.3. Markov-Chain-Monte-Carlo Methoden

Markov-Chain-Monte-Carlo (MCMC) Methoden kommen zum Einsatz, wenn die betrachtete Verteilungsfunktion zu komplex und hochdimensional ist, um eine Zufallsstichprobe durch direktes

Auswerten, beispielsweise durch die in Abschnitt 3.2 eingeführten Verfahren, zu erzeugen. Im Folgenden werden zunächst einige Eigenschaften von Markov-Ketten und anschließend Vertreter der MCMC Methoden betrachtet.

3.3.1. Markov-Ketten

Eine Markov-Kette ist ein spezieller stochastischer Prozess, welcher zur Beschreibung sequentieller, stochastisch unabhängiger Zufallsfolgen benötigt wird. Dabei ist die Folge auf einer Grundgesamtheit S definiert, zwischen dessen Zuständen mit einer bestimmten Wahrscheinlichkeit gewechselt werden kann. In dieser Arbeit beschränkt sich die Betrachtung auf homogene (Übergangswahrscheinlichkeiten zeitlich unabhängig) Markov-Ketten erster Ordnung (die Zukunft des System hängt nur vom gegenwärtigen Zustand ab). Es wird zunächst der einfachere Fall endlicher Zustandsräume betrachtet.

Definition 3.1 (Markov Kette)**.** *P sei eine $k \times k$ Matrix mit den Einträgen $\{p_{i,j} \geq 0 : i,j = 1,...,k\}$, wobei $\sum_{j=1}^{k} p_{ij} = 1$ für alle $i \in \{1,...,k\}$ gilt. Ein Zufallsprozess $(X_0,X_1,...)$ mit endlichem Zustandsraum $S = \{s_1,...,s_k\}$ wird Markov-Kette mit Übergangsmatrix P genannt, wenn für alle $n \in \mathbb{N}_0$ und alle $i_0,...i_{n-1} \in \{1,...,k\}$ gilt:*

$$P(X_{n+1} = s_j | X_0 = s_{i_0}, X_1 = s_{i_1}, ..., X_{n-1} = s_{i_{n-1}}, X_n = s_i) = P(X_{n+1} = s_j | X_n = s_i) = p_{ij}$$

Der Startzustand X_0 der Markov-Kette wird gemäß einer durch einen Zeilenvektor dargestellten Startverteilung

$$\mu^{(0)} = (P(X_0 = s_1), ..., P(X_0 = s_k))$$

gezogen, wobei für alle Folgezustände gilt:

$$\mu^{(n)} = \mu^{(0)} P^n$$

Für diese Arbeit relevant sind Markov-Ketten, welche unabhängig von ihrer Startverteilung gegen eine Grenzverteilung konvergieren. Diese Eigenschaft wird *Ergodizität* genannt.

Definition 3.2 (Ergodizität)**.** *Die Matrix P^∞ sei definiert als $\lim_{e \to \infty} P^e$. Eine Markov-Kette heißt ergodisch, wenn*

(1) P^∞ existiert

(2) Die Zeilen von P^∞ identisch sind

(3) Die Einträge von P^∞ echt positiv sind: $\forall i,j \in \{1,...,k\} : p_{ij} > 0$

(4) Die Zeilen von P^∞ eine Wahrscheinlichkeitsverteilung bilden:
$\forall i \in \{1,...,k\} : \sum_{j=1}^{k} p_{ij}^{\infty} = 1$

Die Ergodizität einer Markov-Kette wird durch die Eigenschaften der *Irreduzibilität* und der *Aperiodizität* impliziert, welche einfacher nachzuweisen sind. Irreduzibilität bedeutet, dass jedes Element s_i des Zustandsraumes S, in welchem die Markov-Kette definiert ist, von jedem anderen Element s_j aus in einem oder mehreren Schritten erreicht werden kann:

Definition 3.3 (Irreduzibilität). *Sei* $(X_0,X_1,...)$ *eine Markov-Kette mit dem Zustandsraum* $S = \{s_1,...,s_k\}$ *und der Übergangsmatrix* P*. Der Zustand* s_i *kommuniziert mit dem Zustand* s_j *(geschrieben* $s_i \to s_j$*), wenn der Pfad* $(s_i,...,s_j)$ *mit einer echt positiven Wahrscheinlichkeit auftritt, das heißt es existiert ein* n*, so dass gilt:*

$$P(X_{m+n} = s_j | X_m = s_i) > 0$$

Die Markov-Kette heißt irreduzibel, wenn für alle $s_i,s_j \in S$ *gilt, dass* $s_i \to s_j$ *und* $s_j \to s_i$*. Andernfalls heißt die Markov-Kette reduzibel.*

Aperiodizität bedeutet, dass es keine systematische Rückkehr zu einem Zustand gibt:
Definition 3.4 (Aperiodizität). *Für eine Markov-Kette* $(X_0,X_1,...)$ *mit dem Zustandsraum* $S = \{s_1,...,s_k\}$ *ist die Periode* d_s *eines Zustandes* s *definiert als:*

$$d_s := \mathrm{ggT}^a\Big\{e \in \mathbb{N} \mid \exists(s_1,...,s_e) \text{ mit } p_{s_e s_1} > 0,\, p_{s_i s_{i+1}} > 0\ \forall i \in \{1,...,e-1\}\Big\}$$

Der Zustand s *heißt aperiodisch, wenn* $d_s = 1$ *gilt. Die Markov-Kette heißt aperiodisch, wenn jeder Zustand aperiodisch ist. Andernfalls heißt die Kette periodisch.*
Definition 3.5 (Stationarität). *Sei* $(X_0,X_1,...)$ *eine Markov-Kette mit dem Zustandsraum* $S = \{s_1,...,s_k\}$ *und der Übergangsmatrix* P*. Eine Wahrscheinlichkeitsverteilung* $r = (r_1,...,r_k)$ *auf dem Zustandsraum* S *ist eine stationäre Verteilung der Markov-Kette, wenn gilt:*

$$r = rP$$

Durch die eingeführten Definitionen lässt sich das Konvergenzverhalten von Markov-Ketten wie folgt zusammenfassen:
Satz 3.6 (Konvergenz von Markov-Ketten).

(1) Eine Markov-Kette ist genau dann ergodisch, wenn sie irreduzibel und aperiodisch ist.

(2) Zu jeder ergodischen Markov-Kette existiert genau eine stationäre Verteilung.

(3) Die Matrix P^∞ *definiert die stationäre Verteilung einer ergodischen Markov-Kette.*

(4) Jede ergodische Markov-Kette konvergiert gegen ihre stationäre Verteilung, unabhängig von der Startverteilung.

Der umfängliche Beweis dieses Satzes findet sich z. B. bei Feller (1968).

Bisher wurden ausschließlich Markov-Ketten aus diskreten Zustandsräumen betrachtet. Für ausführliche Beschreibungen zur Übertragung der Definition von Markov-Ketten auf kontinuierliche Zustandsräume wird auf die Arbeiten von Gilks u. a. (1996) oder Robert und Casella (2004) verwiesen. Bei Markov-Ketten aus kontinuierlichen Zustandsräumen wird die Übergangsmatrix P

[a] größter gemeinsamer Teiler

zu einem Übergangskern $T(s',s)$ und nach (Andrieu u. a., 2003) werden die stationären Verteilungen zu Eigenfunktionen des Übergangkerns, für welche gilt:

$$r(s') = \int T(s|s')r(s)\mathrm{d}s \tag{3.4}$$

Nach (Robert und Casella, 2004) bleiben die Konvergenzeigenschaften von ergodischen Markov-Ketten im kontinuierlichen Fall erhalten. Des Weiteren ist es möglich, komplexe Übergangskerne durch eine Verkettung einfacherer Übergangskerne darzustellen.

Satz 3.7 (Verkettung von Übergangskernen)**.** *T_1 und T_2 seien Übergangskerne mit stationärer Verteilung r. Dann ist*

$$T(s'|s) := \int T_2(s'|s'') \cdot T_1(s''|s)\mathrm{d}s''$$

der Übergangskern einer Markov-Kette mit stationärer Verteilung r.

Außer dem Verketten von Übergangskernen ist es auch möglich, Übergangskerne zweier Markov-Ketten zu mischen.

Satz 3.8 (Mischen von Übergangskernen)**.** *T_1 und T_2 seien Übergangskerne mit stationärer Verteilung r und m_1,$m_2 \geq 0$ mit $m_1 + m_2 = 1$. Dann ist*

$$T(s'|s) := m_1 \cdot T_1(s'|s) + m_2 \cdot T_2(s'|s)$$

Eine hinreichende Bedingung dafür, dass ein Übergangskern eine stationäre Verteilung r besitzt ist die sogenannte *detaillierte Gleichgewichtsbedingung*[b]. Da in vielen Veröffentlichungen der englische Begriff verwendet wird, wird dies aus Konformitätsgründen im Folgenden übernommen.

Satz 3.9 (Detailed Balance Condition)**.** *Eine Markov-Kette mit Übergangskern T erfüllt die Detailed Balance Condition, wenn eine Verteilung r existiert, sodass für alle s,s' gilt*

$$r(s)T(s|s') = r(s')T(s'|s) \tag{3.5}$$

Wenn die Verteilung r existiert, ist sie nach Kelly (2011) die stationäre Verteilung der Markov-Kette. Durch Integration von Gleichung (3.5) kann Gleichung (3.4), die Definition der stationären Verteilung, erreicht werden:

$$\int_S r(s)T(s|s')\mathrm{d}s = \int_S r(s')T(s'|s)\mathrm{d}s = r(s')\int_S T(s'|s)\mathrm{d}s = r(s')$$

3.3.2. Der Metropolis-Hastings-Algorithmus

Der Metropolis-Hastings-Algorithmus (Metropolis u. a., 1953)(Hastings, 1970) ist ein allgemeines MCMC Verfahren. Das grundlegende Vorgehen besteht darin, einen Übergangskern einer Markov-Kette so zu bestimmen, dass die stationäre Verteilung der Kette der gegebenen Zielfunktion

[b] Englisch: Detailed Balance Condition

entspricht. Um dies zu erreichen wird eine Vorschlagsverteilung $q(s^{(n+1)}|s^{(n)})$ verwendet, um Folgezustände für die Markov-Kette unabhängig von der Zielfunktion zu generieren und den neuen Zustand in einem Folgeschritt zu verwerfen oder zu akzeptieren, ähnlich wie bei der Verwerfungsmethode (vgl. Abschnitt 3.2.2).

Sei zusätzlich zur genannten Suchverteilung π die Dichte der Zielverteilung. In jeder Iteration des Algorithmus wird zunächst ein neuer Zustand s' erzeugt. Für die meisten Zustände wäre die Detailed Balance Condition (3.5) nicht erfüllt:

$$\pi(s)q(s,s') \neq \pi(s')q(s',s)$$

Damit diese erfüllt werden kann, wird eine *Akzeptanzwahrscheinlichkeit* eingeführt, welche so gewählt wird, dass die Detailed Balance Condition erfüllt ist. Mit dieser korrigierten Wahrscheinlichkeit wird ausgewertet, ob der neue Zustand als Folgezustand akzeptiert und in die Markov-Kette aufgenommen, oder verworfen wird. Im zweiten Fall wird der Ausgangszustand als Folgezustand beibehalten. In Algorithmus 2 ist der Ablauf des Algorithmus nach Chib und Greenberg (1995) als Pseudocode dargestellt.

Algorithmus 2 : Metropolis-Hastings-Algorithmus Pseudocode. Zielverteilung π und Suchverteilung q.

```
Wähle s^(0)                                          // Initialer Zustand
for n ← 1 to N do
    Ziehe s' ~ q(·|s^(n))
    Ziehe u ~ Unif^c(0,1)
    if u < π(s')q(s^(n)|s') / π(s^(n))q(s'|s^(n)) then
        Setze s^(n+1) := s'
    else
        Setze s^(n+1) := s^(n)
    end
end
```

Auf diese Weise wird die Markov-Kette generiert und die Konvergenz hängt maßgeblich von der Wahl von q ab. Wird die Suchfunktion so gewählt, dass die resultierende Markov-Kette aperiodisch und irreduzibel ist, ist π die stationäre Verteilung der Kette mit folgendem Übergangskern:

$$\begin{aligned} T_{MH}(s^{(n+1)}|s^{(n)}) &= q(s^{(n+1)}|s^{(n)})A(s^{(n+1)}|s^{(n)}) \\ &\quad + \delta(s^{(n+1)} - s^{(n)}) \int q(s'|s^{(n)})(1 - A(s'|s^{(n)}))\mathrm{d}s' \qquad (3.6) \\ \text{mit } A(x|y) &= \min\Big\{1, \frac{\pi(x)q(y|x)}{\pi(y)q(x|y)}\Big\} \end{aligned}$$

[c] vgl. Anhang B.2

3.4. Reversible Jump MCMC Methoden

MCMC Algorithmen sind lediglich in der Lage, über einen Zustandsraum konstanter Dimension zu arbeiten. Um die Möglichkeit zu eröffnen, Modelle unterschiedlicher Größe in Erwägung zu ziehen, werden in diesem Abschnitt die von Green (1995) eingeführten *Reversible Jump MCMC* (RJMCMC) Methoden vorgestellt.

Seien S_m verschiedene Zustandsräume und π_m die Zielwahrscheinlichkeiten in den entsprechenden Räumen. Der kombinierte Zustandsraum ergibt sich zu

$$S = \bigcup_{m=1}^{M} \Big(\{m\} \times S_m \Big),$$

wobei ein Paar (m,s) einen Zustand $s \in S_m$ bezeichnet. Ein Übergangskern T_m mit der stationären Verteilung $r_m(\cdot)$, der auf dem beschriebenen kombinierten Zustandsraum definiert ist, lässt sich wie folgt beschreiben:

$$T((m',s'),(m,s)) = \begin{cases} T_m(s'|s) & \text{wenn } m = m' \\ 0 & \text{sonst} \end{cases} \tag{3.7}$$

Um den Übergangskern zu komplettieren, müssen Übergänge zwischen den Teil-Zustandsräumen ermöglicht werden. Für die Darstellung in Gleichung (3.7) ist die Detailed Balance Condition in Gleichung (3.5) erfüllt, diese geht allerdings davon aus, dass der potentielle Übergangskern T_m in S_m der einzig mögliche ist. Mit der Erweiterung ist die Möglichkeit gegeben, dass ein Übergang aus einem Teil-Zustandsraum S_i in S_j mit $i \neq j$ durchgeführt werden kann. Daher werden alle Kombinationen der Teil-Zustandsräume durch eigene Übergangskerne beschrieben, so sind einem Wechsel aus dem m-ten Raum M Übergangskerne zugeordnet, um in jeden beliebigen Raum wechseln zu können: $\{T_{1|m},...,T_{M|m}\}$. Somit können alle nicht zulässigen Übergangskerne wie folgt gesetzt werden:

$$T_{\mu'|\mu}((m',s')|(m,s)) = 0, \text{ wenn } \mu' \neq m', \mu \neq m, s' \notin S_{m'}, s \notin S_m$$

Der Übergangskern $T_{m'|m}$ wird mit einer bestimmten Wahrscheinlichkeit $w_{m'|m}$ ausgewählt, wobei $\sum_{m'=1}^{M} w_{m'|m} = 1$ gilt.

Für einen derartige Übergang werden die Teil-Zustandsräume in einen gemeinsamen Zustandsraum eingebettet, welcher die Zustandsvektoren um die Komponenten U auf eine Dimension ergänzt:

$$\begin{aligned} S' &= (\{i\} \times S_i \times U_i) \cup (\{j\} \times S_j \times U_j) \\ \dim(S_i) + \dim(U_i) &= \dim(S_j) + \dim(U_j), \forall i,j \in \mathbb{N}_0 \end{aligned} \tag{3.8}$$

Auf diesem neuen Zustandsraum wird eine bijektive Abbildung definiert, welche die Übergänge zwischen zwei Elementen $\Omega_i = (i,s_i,u_i)$ und $\Omega_j = (j,s_j,u_j)$ aus S' beschreibt:

$$\begin{aligned} \tau &: S_i \times U_i \to S_j \times U_j \\ \tau(s_i,u_i) &= (s_j,u_j) \end{aligned} \tag{3.9}$$

Die Rücktransformation erfolgt unter Anwendung von τ^{-1}. Die zusätzlichen Komponenten U und die Abbildung τ sind problemspezifische Parameter, wodurch es möglich ist, einen Zustand s_i mit verschiedenen s_j zu assoziieren. Für die Generierung der zusätzlichen Komponenten U werden Wahrscheinlichkeitsverteilungen $u_x \sim q_x(\cdot|s_x)$ vorgegeben. Algorithmus 3 beschreibt den Übergangskern zum Wechseln zwischen zwei Teil-Zustandsräumen als Pseudocode.

Algorithmus 3 : Pseudocode eines Übergangskerns, welcher zwischen Zuständen aus S' wechselt.

```
Ausgangszustand (i,s_i)
Ziehe u_i ~ q_i(·|s_i)
Transformiere s' = (s_j,u_j) = τ(s_i,u_i) ∈ S'
Ziehe w ~ Unif(0,1)
if w < A(s_j,u_j|s_i,u_i) then          // A: siehe Gleichung (3.13)
    Setze s^(n+1) := s'
else
    Setze s^(n+1) := s^(n)
end
```

Mit der Vorgabe der Verteilungen für die zusätzliche Komponenten wird eine Wahrscheinlichkeitsverteilung $P(\Omega_j|\Omega_i)$ induziert. Durch diese Verteilung muss garantiert werden, dass der Übergang zwischen beiden Zuständen möglich ist, indem die Akzeptanzwahrscheinlichkeit des Übergangskern so gewählt wird, dass die Detailed Balance Condition erfüllt ist. Da, wie oben beschrieben, in Gleichung (3.5) noch von einem eindeutigen Übergangskern T ausgegangen wird, muss unter Berücksichtigung der oben eingeführten Auswahlwahrscheinlichkeiten der Übergangskerne die Detailed Balance Condition erweitert werden:

$$\pi_m(s) w_{m'|m} T_{m'|m}((m',s')|(m|s)) = \pi_{m'}(s') w_{m|m'} T_{m|m'}((m,s)(m',s') \tag{3.10}$$

Die Übergangswahrscheinlichkeiten zwischen den Zuständen Ω_i und Ω_j mit der Transformation τ sind:

$$P(\Omega_i \to \Omega_j) = \pi_i(s_i) \cdot w_{j|i} \cdot q_i(u_i|s_i) \tag{3.11}$$

$$P(\Omega_j \to \Omega_i) = \pi_j(s_j) \cdot w_{i|j} \cdot q_j(u_j|s_j) \cdot |\Im_\tau(s_i,u_i)| \tag{3.12}$$

$$\text{mit } \Im_\tau(s_i,u_i) = \frac{\partial \tau(s_j,u_j)}{\partial(s_i,u_i)}$$

Gemäß Gleichung (3.8) sind die Gleichungen über den gleichen Raum, allerdings in unterschiedlichen Variablen definiert, wodurch auch die Proportionalitätskonstanten verschieden sind. Um diese zu vereinheitlichen, wird gemäß des Transformationssatzes (vgl. (Heuser, 2004)) in Gleichung (3.12) die Determinante der Jacobimatrix der Transformation τ berücksichtigt.

Damit ergibt sich analog zum Metropolis-Hastings-Algorithmus in Gleichung (3.6) die Akzeptanzwahrscheinlichkeit für die Übergänge zu:

$$A(s_j,u_j|s_i,u_i) = \min\Big\{1,\frac{\pi_j(s_j)\cdot w_{i|j}\cdot q_j(u_j|s_j)\cdot|\Im_\tau(s_i,u_i)|}{\pi_i(s_i)\cdot w_{j|i}\cdot q_i(u_i|s_i)}\Big\} \tag{3.13}$$

$$A(s_i,u_i|s_j,u_j) = \min\Big\{1,\frac{\pi_i(s_i)\cdot w_{j|i}\cdot q_i(u_i|s_i)}{\pi_j(s_j)\cdot w_{i|j}\cdot q_j(u_j|s_j)\cdot|\Im_\tau(s_i,u_i)|}\Big\} \tag{3.14}$$

Damit hat der in Algorithmus 3 beschriebene Übergangskern mit den Akzeptanzwahrscheinlichkeiten (3.13), respektive (3.14), die vorgegebene Zielverteilung. Durch Mischen der Übergangskerne gemäß Algorithmus 3 und gemäß Gleichung (3.7) nach Satz 3.8 können auf diese Weise ergodische Markov-Ketten generiert werden.

4. Stand der Forschung

In diesem Kapitel wird der Stand der Forschung bezüglich der automatisierten Rekonstruktion von Straßenkarten auf Fahrbahn-, sowie Fahrspurniveau aus Fahrzeugtrajektorien dargelegt. Dazu wird zunächst beschrieben, wie eine Straßenkarte digital repräsentiert werden kann. Anschließend werden in Abschnitt 4.2 die Verfahren zur Erzeugung einer fahrbahngenauen digitalen Straßenkarte erläutert, während in Abschnitt 4.3 die Verfahren zur fahrspurgenauen Rekonstruktion dargelegt sind.

Die Möglichkeit der Aufzeichnung von Fahrzeugtrajektorien über herkömmliches GPS ist noch keine zwei Jahrzehnte möglich, in dieser Zeit wurden allerdings viele Verwendungsfelder identifiziert und erschlossen. Zu Beginn dieser Zeit wurden von Wilson u. a. (1998) viele dieser Felder bereits prognostiziert, unter anderem die Verwendung der Positionsdaten zur automatisierten Erzeugung von digitalen Straßenkarten.

4.1. Digitale Straßenkarten

Eine *digitale Straßenkarte* ist die modellhafte Beschreibung eines realen Straßen- oder Verkehrsnetzwerkes mithilfe der in Abschnitt 2.1 eingeführten Graphentheorie. Für die Beschreibung und Repräsentation der Karte gibt es anwendungsabhängig verschiedene etablierte Standards wie beispielsweise NDS[a] oder GDF[b].

Eine allgemeine digitale Straßenkarte ist definiert als ein gerichteter gewichteter Graph (vgl. Definition 2.4 und 2.5)

$$G = (V,E,W,A_v,A_e)$$

Mit jedem Knoten in V wird eine geographische Koordinate assoziiert, wodurch der Graph in die Ebene bzw. in den Raum eingebettet wird. Die tatsächliche topologische und geometrische Konfiguration eines Straßennetzes kann dadurch mithilfe des Graphen approximiert werden. Üblicherweise befinden sich Knoten an jenen Stellen, an welchen sich der Straßen- oder Spurverlauf signifikant ändert (Geometrieänderungen) oder eine Vereinigung bzw. Trennung von Straßen oder Spuren erfolgt (Topologieänderung). Die Kantenmenge E enthält Paare von Knoten und entspricht somit der im Kapitel 2 definierten Relation R. In digitalen Straßenkarten werden üblicherweise die Kanten E implizit mit einer zugehörigen Geometrie assoziiert, zumeist mit geradlinigen Verbindungen zwischen den beteiligten Knoten, wobei auch kompliziertere Geometrien, z. B. in Form von Splines, möglich sind. Die Bewertungsmenge W weist jeder Kante $e \in E$ eine oder mehrere kontextbezogene Bewertungen zu, welche zur algorithmischen Behandlung, zum Beispiel

[a] https://www.nds-association.org
[b] International Organization for Standardization: ISO 14825:2011

durch den Algorithmus von Dijkstra (vgl. Abschnitt 2.1.2), genutzt können. Beispielsweise kann beschrieben werden, wie viel Zeit benötigt wird, um die Kante zu passieren oder wie hoch der durchschnittliche Kraftstoffverbrauch dazu wäre. Zusätzlich weist die Abbildung A_v jedem Knoten $v \in V$ eine Menge von *Attributen* zu. Außerdem weist die Abbildung A_e jeder Kante $e \in E$ eine Menge von Attributen zu, welche beispielsweise einen Straßennamen oder Geschwindigkeitslimits beinhalten können. Eine beispielhafte Darstellung einer fahrspur- und fahrbahngenauen digitalen Straßenkarte in Überlagerung zum entsprechenden Satellitenbild ist in Abbildung 4.1 gegeben.

(a) Repräsentation eines Straßennetzes als fahrbahngenauer Graph in Überlagerung zu einem Satellitenbild (Quelle: Microsoft Bing). Knoten sind in Rot und Kanten in Schwarz dargestellt.

(b) Repräsentation eines Straßennetzes als fahrspurgernauer Graph in Überlagerung zu einem Satellitenbild (Quelle: Microsoft Bing). Knoten sind in Rot und Kanten in Schwarz dargestellt

Abbildung 4.1.: Beispielhafte Darstellung einer fahrbahn- bzw. fahrspurgenauen Repräsentation eines Straßennetzes in Überlagerung zu einem Satellitenbild (Quelle: Microsoft Bing).

Vor allem bei der Verwendung fahrspurgenauer digitaler Straßenkarten sind häufig mehr Informationen als die Spurmittellinie von Interesse. In diesem Fall kann entweder generisch die Abbildung A_E benutzt werden, um die Fahrspuren mit beispielsweise weiteren geometrischen Informationen zu besetzen oder es kann eine individuelle Erweiterung des Datenformats entwickelt werden. Dadurch können jeder Fahrspur zusätzliche Attribute wie beispielsweise eine Funktion zur Beschreibung der Spurbreite oder der baulichen Trennung zwischen den Fahrtrichtungen zugewiesen werden.

Zur allgemeinen algorithmischen Behandlung, beispielsweise durch den Algorithmus von Dijkstra zur Routenplanung, können derartige graph-basierte digitale Straßenkarten sowohl auf Fahrbahn-, als auch auf Fahrspurebene verwendet werden.

4.2. Rekonstruktion fahrbahngenauer Straßenkarten

Ziel der für diesen Abschnitt relevanten Verfahren ist es, aus einer Menge von aufgezeichneten Fahrzeugtrajektorien automatisiert eine fahrbahngenaue digitale Straßenkarte zu rekonstruieren. Während weiterführende Verfahren diverse Sensorinformationen wie Radare oder Kameras verwen-

den, sind für diese Betrachtung nur Daten des GNSS-Systems relevant. Zwei der Verfahren werden in Kapitel 6 zur Evaluierung der Arbeit verwendet, daher werden diese im Abschnitt 4.2.4 und 4.2.5 detaillierter betrachtet.

Mittlerweile existiert eine Vielzahl derartiger Algorithmen mit vielen verschiedenen Ansätzen sowie individuellen Vor- und Nachteilen. In (Ahmed u. a., 2015) wird eine Klassifizierung in drei Kategorien dargelegt, welche in diesem Abschnitt als Grundlage dient und mit dem aktuellen Stand der Forschung ergänzt wird. Die drei Kategorien sind die *Punktwolkenanalyse*, die *inkrementelle Trajektorien Analyse* und die *Analyse charakteristischer Punkte*. Zu den im Folgenden dargelegten Verfahren existieren Algorithmen, welche in der Lage sind die erzeugten Ergebnisse nachträglich zu verbessern, indem sie bestimmte Charakteristika ausprägen. Ein derartiges Verfahren ist beispielsweise in der Veröffentlichung von Roeth u. a. (2015) zu finden.

4.2.1. Punktwolkenanalyse

In der Kategorie der Punktwolkenanalyse (PWA) sind Verfahren enthalten, welche die Menge der Eingabedaten als Messpunktwolke betrachten und unter Anwendung von Cluster-Algorithmen Straßen und Kreuzungen identifizieren. Dabei werden nochmals zwei Unterkategorien eingeführt: Einerseits existieren Verfahren, welche in einem ersten Schritt Kreuzungen finden und anschließend die Straßenzüge entlang dieser Punkte identifizieren. Andererseits gibt es Algorithmen, welche die Eingabedaten beispielsweise als Bild betrachten und mit Techniken der Bildverarbeitung die Straßenkarte extrahieren.

In Bentley und Ottmann (1979) ist ein Verfahren zur Identifizierung von Schnittstellen zwischen Segmenten (bspw. Straßen oder Trajektorien) unter Verwendung eines *Sweep-Line*-Algorithmus dargelegt. Dieser kann benutzt werden, um die Segmente in einen ungerichteten Graphen zu überführen. Dieses Verfahren wurde von Edelkamp und Schrödl (2003) erweitert, um zunächst derartige Schnittpunkte basierend auf einer Abstandsmetrik zu bewerten und anschließend mit der Anwendung des *k-means*-Algorithmus, eines nichtparametrisierten Schätzers (vgl. Abschnitt 3.1.2), diese Punkte zu gruppieren. Anhand dieser Gruppierungen wird anschließend die Straßenmittellinie detektiert. Ein recht einfacher Ansatz wird von Guo u. a. (2007) verwendet. Dabei wird eine Menge von Messpunkten unter Annahme einer Normalverteilung der Daten über den Straßenquerschnitt gruppiert und anschließend durch einen Spline beschrieben. Dies führt vor allem bei benachbarten Straßenzügen zu Problemen, falls der Positionierungsfehler zu groß wird. Dieser Umstand wird durch die Berücksichtigung des Headings in der Arbeit von Worrall und Nebot (2007) verbessert. Die Identifizierung der Fahrbahn erfolgt in diesem Fall unter Verwendung des *Least-Squares-Fitting* von Shakarji (1998). Liu u. a. (2012) unterteilen die Trajektorien zunächst in Bereiche mit linearem Fahrverhalten (keine Kurven etc.). Anschließend wird auf diese Bruchstücke ein Clustering Algorithmus angewandt, um die Abschnitte auf gleichen Straßen zu bündeln. Zur Extraktion der Straßenmittellinie wird das *B-Spline-Fitting* aus (Piegl und Tiller, 1996) verwendet. In den Ausführungen von Jang u. a. (2010) werden die Trajektorien zunächst in eine Kachelgeometrie getrennt. In jeder Kachel werden anschließend Verkehrsschwerpunkte geclustert und verbundene Cluster über

die Kachelgrenzen hinweg verbunden. Abschließend werden die entstandenen Straßenzüge geglättet, um marginales Vorwissen über einen typischen Straßenverlauf einfließen zu lassen.

Andere Verfahren überführen die Eingabedaten mittels einer Kerndichteschätzung (KDS), vgl. Abschnitt 3.1.2, in ein diskretes Bild, welches über die Farbintensität das relative Datenaufkommen wiedergibt. Chen und Cheng (2008) präsentieren ein Verfahren, welches ohne Datentransformation, dafür allerdings nur auf kleinen Datensätzen auskommt. Zunächst werden die Daten den Bildverarbeitungsoperationen *Dilatation* und *Schließung* unterzogen. Anschließend wird mit Hilfe des *Ausdünnungs-Algorithmus* von Naccache und Shinghal (1984) die Straßenmittellinie extrahiert. Dieses Verfahren arbeitet auf einer binären Darstellung der Dilatation, daher entstehen Probleme bei unterschiedlich hohem Verkehrsaufkommen. Biagioni und Eriksson (2012) (vgl. Abschnitt 4.2.4) erweitern dieses Verfahren und erzeugen bei der Dilatation ein Graustufenbild, um unterschiedliche Verkehrsaufkommen bzw. fehlerhafte Messungen zu identifizieren. Abschließend wird das Bild binarisiert und ausgedünnt, um den Straßenverlauf zu detektieren. Davies u. a. (2006) erzeugen bei der Dilatation ebenfalls ein Graustufenbild, um aus diesem die Kontur nach (Yokoi u. a., 1975) zu extrahieren. Somit wird eine Art *Bounding-Box* erzeugt, in welcher die Straße liegt. Abschließend wird zu der Kontur das Voronoi Diagramm generiert, welches die Straßenmittellinie repräsentiert. Weitere Verfahren mit diesem Ansatz und verschiedenen Ansprüchen an die Anzahl der Eingabedaten sind beispielsweise in (Steiner und Leonhardt, 2011) und (Shi u. a., 2009) dargelegt. Des Weiteren geben Davies u. a. (2006) und Wang u. a. (2013) Möglichkeiten an, ihre Verfahren zur Aktualisierung von vorhandenem Kartenmaterial zu verwenden.

4.2.2. Inkrementelle Trajektorien Analyse

Bei der inkrementellen Spuranalyse werden die Eingabedaten iterativ ausgewertet. Zu Beginn ist die zu erzeugende Straßenkarte leer und mit jeder hinzugefügten Trajektorie werden neue Eigenschaften erzeugt oder vorhandenen Informationen aktualisiert.

Cao und Krumm (2009) (vgl. Abschnitt 4.2.5) wenden zunächst einen Vorverarbeitungsschritt an, in welchem künstliche physikalische Anziehungskräfte auf die Trajektorien aufgebracht werden. Dadurch gruppieren bzw. separieren sich die Eingabedaten auf den Straßen, sollten sie einen gewissen Abstand aufweisen. Anschließend werden die Daten iterativ verarbeitet, um eine Straßenkarte von Grund auf zu konstruieren. Ahmed und Wenk (2012) führen das Einfügen und Aktualisieren basierend auf der Fréchet - Distanz unter Verwendung der Arbeit von Buchin u. a. (2009) durch. Erzeugt wird ein ungerichteter Graph unter der Voraussetzung, dass jede abzuleitende Straße einen Bereich hat, der sich *eindeutig* identifizieren lässt. Brüntrup u. a. (2005) präsentieren ein Serversystem mit Integration von Algorithmen der künstlichen Intelligenz mit hohen Ansprüchen an die Eingabedaten. Zur skalierbaren Anwendung werden die Eingabedaten auf eine Kachelkarte verteilt und nach der Prozessierung vereinigt. Zhang u. a. (2010) präsentieren ein Verfahren zur Aktualisierung von vorhandenem Kartenmaterial. Dazu wird eine herkömmliche digitale Straßenkarte zunächst an charakteristischen Stellen senkrecht geschnitten, um diese inkrementell mit den gemessenen Trajektorien zu fusionieren. Als Ergebnis entsteht eine aktualisierte und präzisere Straßenkarte.

4.2.3. Analyse charakteristischer Punkte

Bei dieser Klasse von Verfahren werden auf den Eingabedaten zunächst charakteristische Punkte bezüglich des Fahrverhaltens, beispielsweise basierend auf der Geschwindigkeit oder dem Fahrtwinkel, identifiziert. Anschließend werden die Verbindungen zwischen diesen Punkten analysiert, um etwaige Straßenverläufe zu bestimmen.

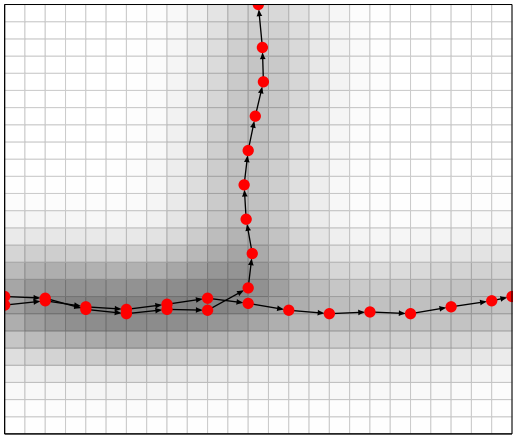
Im Verfahren von Karagiorgou und Pfoser (2012) werden zunächst *Abbiegepunkte* auf den Trajektorien über Grenzwerte in Fahrtrichtung und -geschwindigkeit bestimmt, welche anschließend über Distanzgrenzwerte geclustert werden. Dabei werden jedem Abbiegepunkt Metadaten zugeordnet, welche die Abbiegerichtung beinhalten. Somit kann über die geclusterten Punkte bestimmt werden, ob es sich um eine Kreuzung oder eine Kurve handelt. Die Cluster können abschließend hinsichtlich ihrer Konnektivität untersucht und somit in einen Straßengraph überführt werden. Fathi und Krumm (2010) beschreiben das Anlernen einer Schablone, um über die Verteilung der Eingabedaten um einen Punkt herum, Kreuzungen von Straßen zu unterscheiden. Abschließend werden die Verbindungen zwischen den gefundenen Kreuzungen erzeugt.

4.2.4. Verfahren nach Biagioni und Eriksson

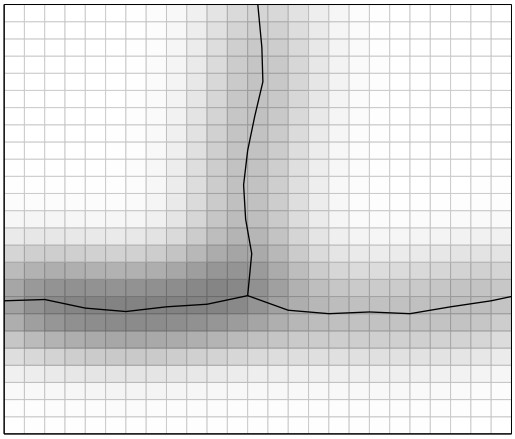
Als Vertreter der Punktwolkenanalyse wird in diesem Abschnitt das Verfahren nach Biagioni und Eriksson (2012) genauer betrachtet, da es zur Evaluierung in Kapitel 6 verwendet wird.

Zu Beginn wird das betrachtete Szenario in äquidistant große Zellen von $1\,\mathrm{m} \times 1\,\mathrm{m}$ diskretisiert. Durch das Auswerten und Akkumulieren der Anzahl an Trajektorien welche jede Zelle passiert haben entsteht ein zweidimensionales Grauwerthistogramm. Um den inhärenten Positionierungsfehler zu rekonstruieren, wird eine KDS (vgl. Abschnitt 3.1.2) durchgeführt. Dazu wird das Histogramm mit einer Normalverteilung $\mathrm{Norm}^{\mathrm{c}}(0,\sigma^2)$ gefaltet, wobei der Parameter σ vom zur Aufnahme verwendeten GNSS-System abhängt. Diese beiden Schritte sind in Abbildung 4.2a exemplarisch mit zwei Trajektorien dargestellt. Das ungefaltete Grauwerthistogramm ist dabei nicht abgebildet. Aus dem entstandenen Histogramm wird mit Hilfe des Skelettierungsalgorithmus (Zhang und Suen, 1984) die Mittellinie extrahiert. Die beispielhafte Anwendung ist in Abbildung 4.2b dargestellt. Zu diesem Zweck muss das Grauwerthistogramm binarisiert werden, was bedingt durch die Wahl eines passenden Schwellwertes Informationen über weniger häufig befahrene Straße verwerfen würde. Um dies zu vermeiden, wird für jedes Grauwertlevel zwischen $0 - 255$, beginnend beim höchsten, ein Skelett erzeugt und alle addiert. Anschließend wird auf dem erzeugten Gesamtskelett der *Combustion*-Algorithmus (Shi u. a., 2009) angewandt, um Kreuzungs- und Endstraßenpixel zu identifizieren. Abschließend werden mit Hilfe des Algorithmus von Viterbi (vgl. Abschnitt 2.2) und des *Douglas-Peucker*-Algorithmus (Douglas und Peucker, 1973) die Straßenzüge zwischen den identifizierten Punkten, wie in Abbildung 4.2c dargestellt, erzeugt.

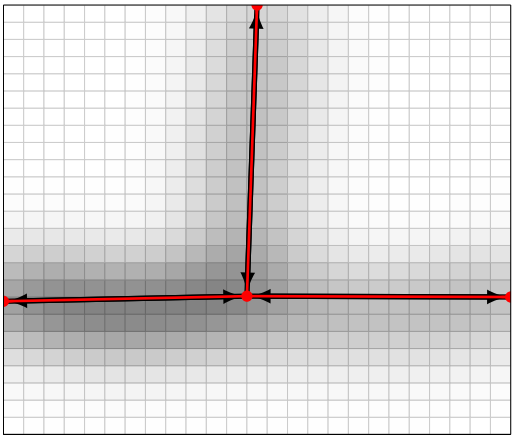
[c] vgl. Anhang B.2



(a) Überführung der Trajektorien (schwarz/rot) in ein Grauwerthistogramm (nicht dargestellt) und Faltung mit einer Normalverteilung (grau).



(b) Skelettierung (schwarz) der Faltung.



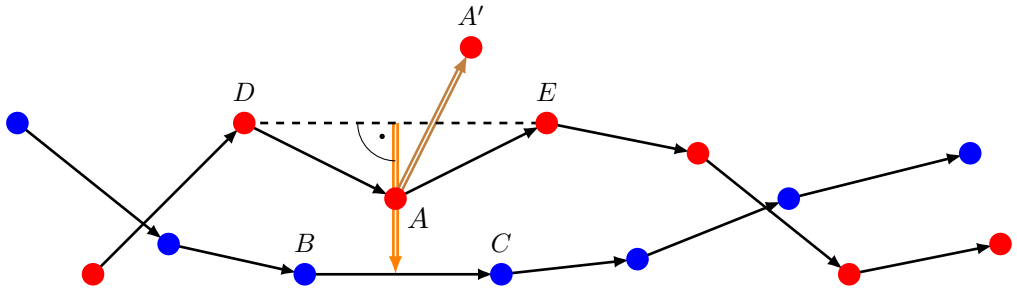
(c) Abgeleiteter fahrbahngenauer Straßengraph.

Abbildung 4.2.: Schritte zur Rekonstruktion einer fahrbahngenauen Straßenkarte nach Biagioni und Eriksson (2012).

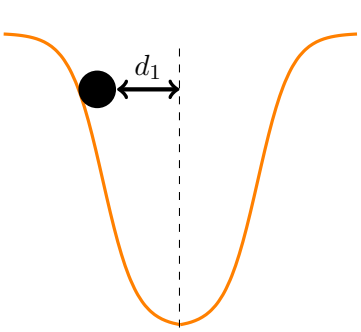
4.2.5. Verfahren nach Cao und Krumm

Als Vertreter der inkrementellen Trajektorienanalyse wird in diesem Abschnitt das Verfahren nach Cao und Krumm (2009) genauer betrachtet, da es zur Evaluierung in Kapitel 6 verwendet wird.

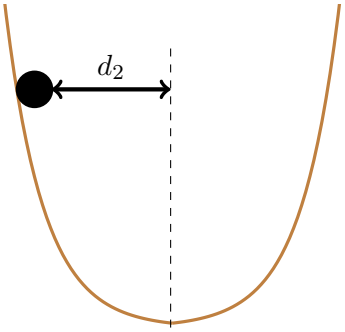
In diesem Verfahren werden zunächst die Eingabedaten vorverarbeitet, um die Identifizierung verschiedener Straßen einfacher zu gestalten. Dazu werden künstliche physikalische Kräfte auf die Messpunkte der Daten aufgebracht, um deren Position zu verändern. Zur Erläuterung sind in Abbildung 4.3a zwei Trajektorien mit je blauen bzw. roten Knoten und schwarzen Kanten dargestellt. Des Weiteren sind zwei verschiedene Arten von Kräften durch Pfeile mit doppelter Linie dargestellt: Einerseits wird jeder Messpunkt A von benachbarten Kanten wie BC, senkrecht zu gedachten, den betrachteten Knoten A überspringenden, Kanten wie DE angezogen, wobei die Kraft einer Anziehungskraft (orange) gleicht. Andererseits wird jeder Messpunkt A von seiner ursprünglichen Position A' exponentiell (braun) angezogen, sodass jeder Punkt nicht beliebig weit bewegt werden kann. In Abbildung 4.3b und 4.3c sind die Potentiale der beschriebenen Kräfte



(a) Wirkungsweise der künstlichen physikalischen Kräfte.



(b) Potential einer Anziehungskraft.



(c) Potential einer Exponentialkraft.

Abbildung 4.3.: Künstliche physikalische Kräfte im Modell. Die Abbildungen sind den Originalen in (Cao und Krumm, 2009) nachempfunden.

dargestellt. Optional kann in der Anziehungskraft noch die Fahrrichtung berücksichtigt werden, sodass nur Punkte gleichgerichteter Trajektorien angezogen, andernfalls abgestoßen werden.

Die Überführung der Daten in einen Straßengraphen erfolgt iterativ, wobei der initiale Graph leer ist. Für jeden Messpunkt einer jeden Trajektorie wird entschieden, ob er von einer bereits berücksichtigten Straße des Graphen aufgezeichnet wurde, oder ob ein neuer Knoten in den Graph eingefügt werden muss. Dabei wird auch jeweils das zeitliche Umfeld der Messung berücksichtigt, um beispielsweise zu entscheiden, ob der Messpunkt von einer potentiellen Kreuzung oder Brücke aufgezeichnet wurde.

4.3. Rekonstruktion fahrspurgenauer Straßenkarten

Während das im vorherigen Abschnitt beschriebene Thema über die letzten Jahre intensiv erforscht wurde und daher die beschriebenen Verfahren vielzählig und die Ansätze vielfältig sind, ist das Erzeugen von fahrspurgenauen digitalen Straßenkarten gegenwärtig im Fokus diverser Arbeiten. In diesem Abschnitt werden Verfahren beschrieben, welche in der Lage sind derartige Karten teilweise oder vollständig aus verschiedenen Sensordaten abzuleiten. Die Ziele der präsentierten Arbeiten sind dabei verschieden motiviert, da beispielsweise diverse Fahrerassistenzsysteme zwar fahrspurgenaue Informationen benötigen, diese allerdings nur im unmittelbaren Umfeld gültig sein müssen. Andererseits werden für das autonome Fahren flächendeckende und hochaktuelle Spurdaten benötigt, um den unterschiedlichsten Aufgaben von Routenplanung bis Durchführung gerecht werden zu können. Häufig können die Ansätze zur Ableitung lokaler Informationen für die

Generierung großflächiger Karten weiterverwendet werden, daher werden in diesem Abschnitt der Vollständigkeit halber auch derartige Beispiele aus der Literatur angeführt.

4.3.1. Fahrerassistenzsysteme

Bezüglich der Fahrerassistenzsysteme haben Dickmanns und Zapp (1987) Pionierarbeit geleistet und aus Kamerabildern die anstehende Krümmung der Fahrspur vorausgesagt, um diese für eine bessere Fahrzeugführung nutzen zu können. Klotz u. a. (2004) haben sich mit der gleichen Prädiktion, basierend auf Kameradaten in Kombination mit einem GNSS System beschäftigt, um die Steuerung des ACC[d] zu verbessern. Sparbert u. a. (2001) präsentieren ein Verfahren, um in Laserscandaten dynamische Hindernisse zu erkennen und zur besseren Spurdetektion temporär zu entfernen, um den Spurhalteassistenten zu verbessern. Goldbeck und Huertgen (1999) verbessern die kamerabasierte Prädiktion, um bei verschiedensten Wetterbedingungen gute Ergebnisse erzielen zu können. Cramer u. a. (2004) detektieren und klassifizieren Fahrbahnmarkierungen und -bordsteine, wobei das Tracking auch bei Spurwechseln funktioniert. Huertgen u. a. (1999) erweitern dieses System noch um die Erkennung und Klassifizierung von Verkehrsschildern. Weigel u. a. (2006) verwenden GNSS-, Kamera- und Radardaten, sowie Kartenvorwissen, um Informationen über empfohlene zulässige Höchstgeschwindkeiten auf Fahrspurebene zu generieren.

4.3.2. Naive Verfahren

Über die Ableitung von spurgenauen Umfeldinformationen hinaus gehen zunächst die *naiven* Verfahren. Dies bedeutet, dass diese Verfahren ohne ein globales Modell arbeiten, sondern lokal gültige Informationen *naiv* aus den Eingabedaten extrahieren und verknüpfen und dabei viele Annahmen und Vereinfachungen treffen. In diesem Rahmen sind sie allerdings in der Lage gute Ergebnisse bei kurzer Laufzeit erzielen zu können. Chen und Krumm (2010) gehen von einer fahrbahngenauen Straßenkarte aus und schneiden den Straßenverlauf in bestimmten Abständen senkrecht, um für jede Senkrechte die Schnittpunkte mit den Fahrzeugtrajektorien zu berechnen. Die Konstruktion dieser Querschnitte ist schematisch in Abbildung 4.4 dargestellt. Auf jedem derartigen Schnitt entsteht somit eine eindimensionale Clusteringaufgabe. Es wird angenommen, dass die GNSS Messungen der Fahrzeuge innerhalb einer jeden Fahrspur normalverteilt sind und somit auf jedem Schnitt die Fahrspuren durch eine Kombination von Normalverteilungen in Form eines *Gauß'schen Mischmodells* (GMM) beschrieben werden können. Um die Parameter des Modells zu lernen wird der *Expectation - Maximization* - Algorithmus (Dempster u. a., 1977) verwendet. Um die Anzahl an Fahrspuren, respektive die Anzahl der Komponenten im GMM zu bestimmen, wird zusätzlich ein Minimierungsproblem definiert, dessen Zielfunktion eine gute Übereinstimmung zwischen Modell und Daten und außerdem ein minimal komplexes Modell forciert. Edelkamp und Schrödl (2003) geben zusätzlich zu ihrem Verfahren zur Rekonstruktion einer fahrbahngenauen Karte eine Möglichkeit zur Spurschätzung an. Dabei wird ebenfalls der Straßenverlauf äquidistant senkrecht geschnitten (vgl. Abbildung 4.4) und die Trajektorien auf

[d] Adaptive Cruise Control

diesen Schnitten ausgewertet. Die Anzahl der Spuren wird aufgrund der Streuung bestimmt, indem diese durch die durchschnittliche Breite einer Spur geteilt wird. Anschließend wird der *k-means*-Algorithmus verwendet, um die Spurmittellinien zu finden. Über die Analyse aufeinander folgender Querschnitte können Spuraufteilungen bzw. -zusammenführungen identifiziert werden. Uduwaragoda u. a. (2013) beschreiben einen probabilistischen Ansatz, welcher in Kapitel 6 verwendet wird. Dabei werden ebenfalls nach Abbildung 4.4 senkrechte Schnitte auf dem Straßenverlauf konstruiert. Auf jedem Schnitt impliziert jede Fahrspur nun eine Wahrscheinlichkeitsverteilung, welche angibt wo Realisierungen eines Zufallsexperimentes (der Durchfahrt) liegen können. Umgekehrt bedeutet dies, dass jeder Schnittpunkt P zwischen einer Senkrechten und einer Trajektorie dieser nicht bekannten Wahrscheinlichkeitsverteilung unterliegt. Ziel der Arbeit ist es, mit Hilfe einer *Kerndichteschätzung* (vgl. Abschnitt 3.1.2) diese Verteilung zu rekonstruieren, um daraus die wahrscheinlichste Lage der Fahrspuren zu extrahieren.

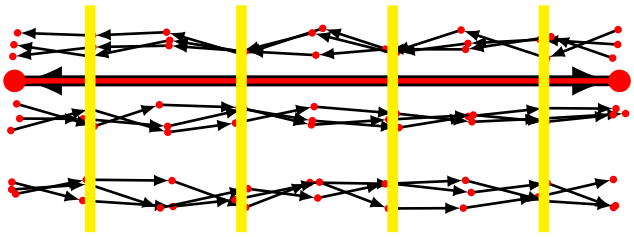


Abbildung 4.4.: Konstruktion der senkrechten Querschnitte (gelb) zum Straßenverlauf (rote Linie) und Auswertung der Trajektorien (schwarze Pfeile).

4.3.3. Komplexe Verfahren

Komplexe Verfahren sind Algorithmen, welche das Wissen über die Spurgeometrie und -topologie nicht direkt aus den Daten ableiten, sondern ein Modell beschreiben, welches auf verschiedene Arten an die Eingabedaten angelernt wird. Auf diese Weise kann eine entsprechende Karte sehr viel robuster aus fehlerbehafteten Daten erzeugt werden. Des Weiteren kann das Modell, in der Regel zu Lasten der Laufzeit, beliebig komplex gestaltet sein, um somit weniger bis keine Annahmen und Vereinfachungen treffen zu müssen.

In dieser Arbeit sind vornehmlich jene Verfahren von Interesse, welche aus Positionsdaten von Fahrzeugen spurgenaue digitale Straßenkarten erzeugen. Daher werden derartige Algorithmen detaillierter untersucht, während über Arbeiten basierend auf anderen Informationen lediglich eine Übersicht gegeben wird. Viel behandelt sind Ansätze, welche das spurgenaue Wissen aus stationären oder mobilen Kameras ableiten. Z.B. Kang und Jung (2003) und Kim (2008) detektieren Spurmarkierungen und Bordsteine, um diese relativ zum Fahrzeug aufzuzeichnen. Somit entsteht für jede befahrene Spur redundantes Wissen, welches zu einer Karte fusioniert werden kann. Verfahren wie von Dickmanns und Mysliwetz (1992) nutzen diese Informationen, um Fahrspurmodelle z.B. in Form von Klothoiden anzulernen. Mélo u. a. (2006) extrahieren aus stationären Kameras die Bewegungsprofile von Fahrzeugen auf Autobahnen, um entsprechendes spurgenaues Wissen zu extrahieren. Huang u. a. (2009) kombinieren mehrere Kameras und Laserscanner, um bei einer

Durchfahrt ein umfassendes Umfeldmodell, bestehend aus Linienmarkierungen und Fahrspurmittellinien, sowie Hindernissen zur späteren Lokalisierung zu erzeugen und zu tracken. Kaliyaperumal u. a. (2001) beschreiben einen Verfahren mit dem gleichen Ziel, basierend auf Radar Messungen.

Im Folgenden werden verschiedene Verfahren betrachtet, die aus Positionsdaten verschiedener Güte spurgenaues Wissen erzeugen. Rogers u. a. (1999) repräsentieren eine Straße als eine Menge von Straßensegmenten. Jedes Segment wird dargestellt als eine Mittellinie, bestehend aus einer Sequenz von Weltkoordinaten, zu welcher alle Fahrspuren parallel verlaufen. Zum Anlernen des Modells wird jede Mittellinie aus einer herkömmlichen fahrbahngenauen Straßenkarte entnommen und die entsprechenden Eingabedaten iterativ analysiert. Für jede neue Fahrzeugtrajektorie werden die Positionen der Knoten der Mittellinie neu berechnet, um somit eine verbesserte fahrbahngenaue Repräsentation zu erhalten. Anschließend werden die Abstände zwischen den Trajektorien und den Mittellinien berechnet und ein hierarchisches Clusteringverfahren genutzt, um die Abstände zwischen der Straßen- und den Spurmittellinien zu bestimmen. Diese Werte gelten auf dem gesamten Segment, um die Spurverläufe relativ zur Straße zu definieren. Die Annahmen, dass Fahrspuren immer parallel verlaufen und keine Angaben über die Behandlung von Spuraufteilungen bzw. -zusammenführungen oder Kreuzungen gemacht werden schränken das Verfahren ein. Des Weiteren werden die Spurverläufe, in Abhängigkeit der Straßenmittellinie, als diskrete Punkte und nicht als kontinuierliche Kurve beschrieben. Chen u. a. (2010) beschreiben ein Modell, in dem eine Fahrspur zunächst durch Stützknoten im Groben definiert und anschließend durch verschiedene geometrische Formen verfeinert wird. Betrachtet werden dabei Linien, Kreisbögen, Klothoiden und Splines, welche jeweils als parametrisierte Kurven auf den Stützknoten definiert werden. Linien und Bögen sind dabei simpel und somit effizient zu definieren, stellen allerdings in der Realität selten repräsentative Verläufe, sondern eher Approximationen dar. Klothoiden und Splines sind komplexer zu beschreiben, bieten dafür allerdings eine bessere Näherung realer Spurverläufe. Zum Anlernen der Modelle werden in dieser Beschreibung zunächst die Annahmen getroffen, dass die Eingabedaten in Form von Fahrzeugtrajektorien zwar durch das Fahrverhalten und die Positionierung fehlerbehaftet sind, diese Fehler sich allerdings im Zentimeterbereich bewegen. Die Betrachtungen der Arbeit beschränken sich auf eine Fahrspur, daher geschieht das Anlernen der Modelle durch eine analytische Berechnung der Parameter unter Berücksichtigung der Daten. Die Beschreibungen der Modelle werden im Rahmen der Arbeit nicht auf Kreuzungen ausgeweitet. Schrödl u. a. (2004) erweitern die Arbeit von Edelkamp und Schrödl (2003) und beschreiben Straßen wie Rogers u. a. (1999) als Segmente mit parallelen Fahrspuren, allerdings mit der Berücksichtigung von Spuraufteilungen und -zusammenführungen. In der Arbeit von Schrödl u. a. (2004) wird das Modell um die Repräsentation von Kreuzungen erweitert. Durch den *Versuch* die Eingabedaten von Kreuzungen in eine Straße zu überführen werden die potentiellen Übergänge von Straßensegmenten in eine Kreuzungsfläche detektiert. Über die Auswertung der Trajektorien werden zulässige Abbiegebeziehungen bestimmt und im Modell als Matrix abgelegt.

5. Verfahren zur automatischen Konstruktion hochgenauer Straßenkarten

5.1. Überblick

In diesem Kapitel werden die in dieser Arbeit entwickelten Algorithmen dargestellt. Als Eingangsdaten wurden Fahrzeugtrajektorien einer Fahrzeugflotte erfasst.

In Abschnitt 5.2 wird zunächst ein Verfahren dargelegt, welches in der Lage ist, Fahrzeugtrajektorien semantisch in verschiedene Verkehrssituationen aufzuteilen. Das Verfahren sucht dabei charakteristische Merkmale auf den Daten, welche kategorisiert und durch die Konstruktion einer Triangulation in eine räumliche Unterteilung überführt werden. Insgesamt unterscheidet der Algorithmus am Ende zwischen geraden Straßen, Kurven und Kreuzungen.

In Abschnitt 5.3 werden die auf diese Art segmentierten Eingabedaten benutzt, um mit einem neuen Ansatz eine fahrbahngenaue Straßenkarte zu erzeugen. Dabei erzeugt das Verfahren für jede unterschiedene Verkehrssituation ein eigenes Untermodell, welche unter Verwendung einer RJMCMC Methode (vgl. Abschnitt 3.4) bezüglich der Eingabedaten optimiert werden.

In Abschnitt 5.4 wird anschließend ein weiteres neues Verfahren beschrieben, welches unter Verwendung einer RJMCMC Methode in der Lage ist, eine fahrspurgenaue Straßenkarte zu erzeugen. Dabei wird eine mit dem vorherigen Verfahren erzeugte fahrbahngenaue Karte um geometrische und topologische Informationen angereichert. Dazu werden zunächst die Straßen und Kreuzungen aus der fahrbahngenauen Karte in detaillierte Modelle überführt. Diese Modelle beschreiben auf Fahrspurebene die Eigenschaften der Straße, wie beispielsweise die Spurbreiten der Fahrspuren und werden vom Algorithmus bezüglich der Daten optimiert.

5.2. Segmentierung von Fahrzeugtrajektorien

Die von einer Fahrzeugflotte gesammelten GNSS Fahrzeugtrajektorien, wie beispielsweise in Abschnitt 6.1 dargelegt, beschreiben in einer beliebigen Anzahl ein ebenso beliebig großes Verkehrsszenario. Diese Daten können zu verschiedenen Analysen und Vorhersagen verwendet werden. Um die Dimension der auszuwertenden Daten greifbar zu machen ist es nötig, die Trajektorien automatisiert gemäß einer bestimmten Logik aufzuteilen. In diesem Abschnitt wird ein, in Roeth u. a. (2016) veröffentlichtes Verfahren beschrieben, welches derartige Fahrzeugtrajektorien semantisch in verschiedene sogenannte *Verkehrssituationen* unterteilen kann. Eine Verkehrssituation beschreibt dabei eine Situation mit bestimmten topologischen und geometrischen Ausprägungen. Da die zu analysierenden Fahrzeugtrajektorien ein Verkehrsnetz abbilden, ist es möglich die Differenzierung der

Verkehrssituationen anhand charakteristischer Ausprägungen eines derartigen Netzes vorzunehmen. Somit lässt sich ein beliebiges Szenario aus drei verschiedenen Ausprägungen zusammensetzen:

1. Straße - Gradlinig verlaufender Streckenabschnitt ohne Krümmung
2. Kurve - Gekrümmter Streckenabschnitt
3. Kreuzung - Beliebig geartete Zusammenführung / Aufteilung von Streckenabschnitten

Der in diesem Abschnitt beschriebene Algorithmus ist in der Lage, einen beliebigen Datensatz von Fahrzeugtrajektorien vollautomatisch in die genannten Kategorien aufzuteilen, um daraus eine räumliche *Segmentierung* zu erzeugen. Eine Segmentierung S besteht aus einer Partition von *Raumzellen* C, welche jeweils eine Verkehrssituation enthalten:

$$S = \bigcup_{c \in C} c \text{ mit } c_i \cap c_j = \varnothing \; \forall i \neq j$$

In Abbildung 5.1 ist eine beispielhafte Segmentierung mit den oben genannten Ausprägungen auf einem Minimalbeispiel dargestellt.

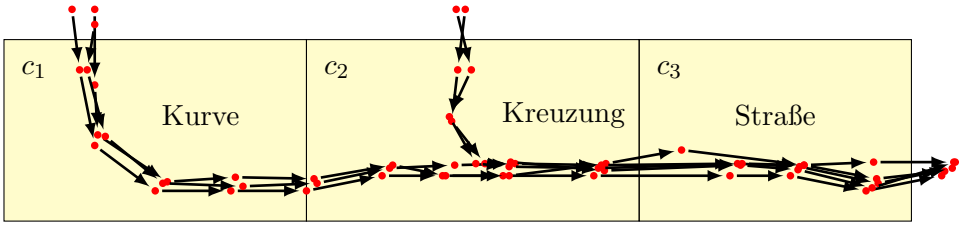


Abbildung 5.1.: Beispielhafte Unterteilung eines Szenarios in verschiedene Verkehrssituationen.

Im ersten Schritt des Algorithmus werden iterativ alle Fahrzeugtrajektorien analysiert, um zu bestimmen, ob ein Messpunkt auf einer Straße oder in einer Kurve bzw. einer Kreuzung aufgezeichnet wurde. Es wird ein zeitliches Fenster von drei aufeinanderfolgenden Messpunkten betrachtet und die Durchschnittsgeschwindigkeit, sowie der eingeschlossene Fahrwinkel berechnet (vgl. Abbildung 5.2). Wird eine minimale Fahrwinkeländerung von α und eine maximale Durchschnittsgeschwindigkeit von s_a unter- bzw. überschritten, wird ein Messpunkt (hier: P_2) als *Abbiegepunkt* kategorisiert.

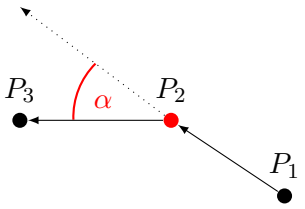


Abbildung 5.2.: Bestimmung der Fahrwinkeländerung α im Punkt P_2 (rot).

Anschließend werden die detektierten Abbiegepunkte mit einem *DBSCAN* Algorithmus (Ester u. a., 1996) mit den Parametern ϵ_{seg} und $\text{minPts}_{\text{seg}}$ in verschiedene Cluster aufgeteilt. Somit liegt jedes entstandene Cluster auf einer Kurve (vgl. Abbildung 5.3) oder auf einer Kreuzung. Fehlerhafte Abbiegepunkte, entstanden durch Messfehler einzelner Fahrzeugtrajektorien, werden vom DBSCAN

Algorithmus aufgrund der benötigten Mindestanzahl an Punkten pro Cluster nicht kategorisiert und somit vom Verfahren als Ausreißer ignoriert.

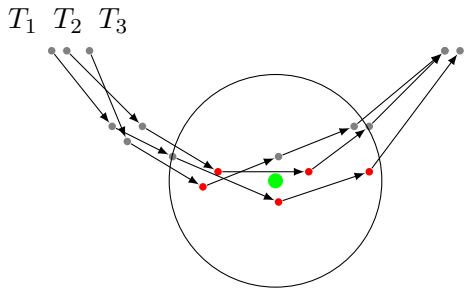


Abbildung 5.3.: Beispielhafte Clusterbestimmung (grün) auf einer Kurve, basierend auf den detektierten Abbiegepunkten (rot) dreier Fahrzeugtrajektorien (Messpunkte grau/rot).

Um aus der Summe von Clusterpunkten eine räumliche Diskretisierung zu erzeugen, wird im nächsten Schritt die Triangulierung (Lee und Schachter, 1980) dieser Punkte erzeugt. Zu Problemen kann hierbei eine fehlende Einschränkung der Ausdehnung führen. Da die finale Segmentierung die Straße einschließen soll (vgl. Abbildung 5.1), muss die Struktur dieser immer erhalten bleiben. Diese Prämisse ist allerdings schon im simpelsten Fall nicht gewährleistet. Beispielsweise kann eine lange Straße mit geringer Krümmung vom Algorithmus durch nur zwei Clusterpunkte an den jeweiligen Enden beschrieben werden, wobei eine derartige lineare Betrachtung der Struktur nicht gerecht werden kann. Um dies zu verhindern, muss die Größe eines Triangulationsdreiecks eingeschränkt werden. Daher werden die Clusterpunkte zunächst hinsichtlich ihrer Konnektivität untersucht. Hierzu wird anhand der Verläufe der einzelnen Fahrzeugtrajektorien mithilfe eines Distanzgrenzwertes d_{cl} geprüft, welche Clusterpunkte verbunden sind (vgl. Abbildung 5.4, grün). Um das oben genannte Problem zu vermeiden, werden zunächst die Strecken C_xC_y mithilfe eines

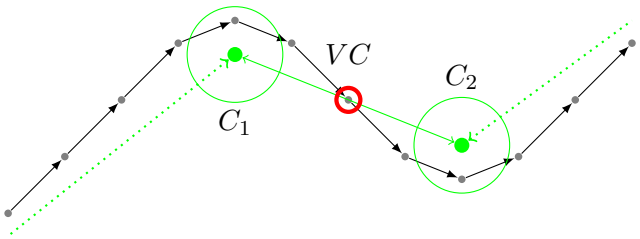


Abbildung 5.4.: Ermittlung der Konnektivität zwischen den Clusterpunkten C_1 und C_2 (grün) und dem virtuellen Clusterpunkt VC (rot) anhand einer Fahrzeugtrajektorie.

Grenzwertes d_{vc} durch *virtuelle* Clusterpunkte (VC) diskretisiert. Ein beispielhafter derartiger Punkt ist in Abbildung 5.4 als roter Kreis dargestellt. Abschließend wird die Triangulierung basierend auf der Summe aus virtuellen und nicht-virtuellen Punkten erzeugt. In Abbildung 5.5 ist die Triangulierung des Szenario aus Abbildung 5.4 dargestellt.

Im nächsten Schritt wird die Triangulierung in das entsprechende Voronoi-Diagramm (Lee und Schachter, 1980) überführt. Die Polygone dieser Zerlegung beschreiben folglich Regionen deren Zentren virtuelle oder nicht-virtuelle Clusterpunkte sind. Abbildung 5.6 zeigt einen Ausschnitt des Voronoi-Diagramms zur Triangulierung aus Abbildung 5.5. In Abbildung 5.6 wird ersichtlich,

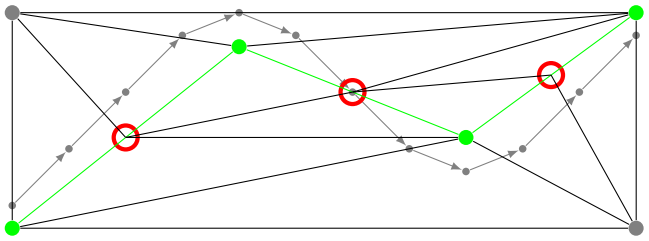


Abbildung 5.5.: Beispielhafte Triangulierung des Szenarios aus Abbildung 5.4, ergänzt um weitere (virtuelle) Cluster und zwei Rahmenpunkte (grau), welche in einer realen Anwendung durch angrenzende Clusterpunkte ersetzt werden würden.

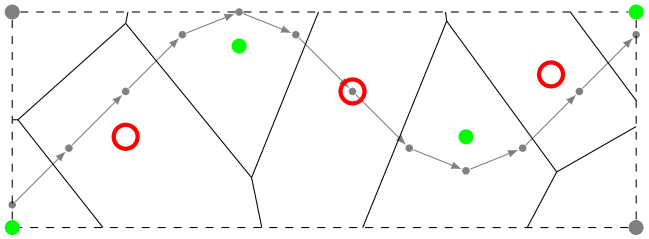


Abbildung 5.6.: Beispielhaftes Voronoi-Diagramm (Ausschnitt) der Triangulierung aus Abbildung 5.5.

dass durch das Einfügen der virtuellen Clusterpunkte die Polygone des Voronoi-Diagramms die Trajektorie möglichst rechtwinkelig schneiden. Bereits nach diesem Schritt beinhaltet jedes Polygon des Voronoi-Diagramms immer eine der definierten Verkehrssituationen. Je nach Beschaffenheit der Trajektorienverläufe sind die Polygone in der Regel unnötig groß und sollen im nächsten Schritt auf den wesentlichen Teil beschnitten werden. Dazu werden alle Trajektorien auf eine Breite von w_{vp} dilatiert und vereinigt. Das Resultat ist eine *Umgebung*, welche alle Trajektorien beinhaltet (vgl. Abbildung 5.7). Wird die entstandene Dilatation aus Abbildung 5.7 mit dem

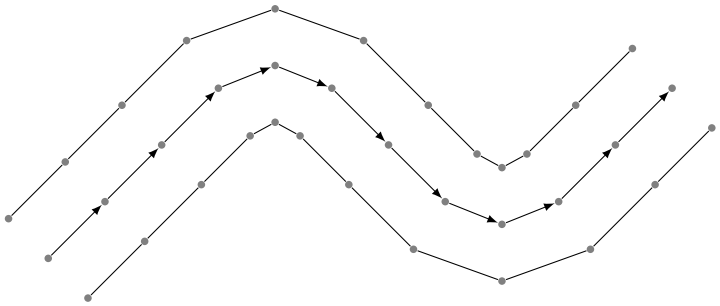


Abbildung 5.7.: Dilatation der Trajektorie.

Voronoi-Diagramm aus Abbildung 5.6 geschnitten, verbleiben nicht konvexe Polygone, welche in Abbildung 5.8 dargestellt sind. Die Schnittpunkte zwischen dem Voronoi-Diagramm und der Dilatation sind rot markiert.

Anschließend werden die Polygone durch ihre konvexe Hülle ersetzt. Dies erleichtert dem in Abschnitt 5.3 beschriebenen Optimierungsalgorithmus zu bestimmen, ob eine Änderung zu einem Ergebnis innerhalb des Polygons geführt hat. Entsprechende Änderungen sind in Abbildung 5.8 grün markiert. Durch das Einsetzen der virtuellen Clusterpunkte sind die Polygone, wie beabsichtigt,

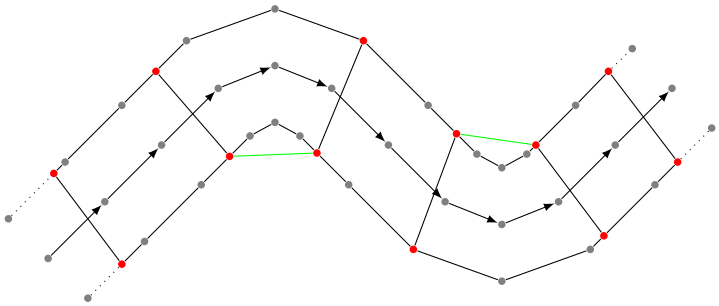


Abbildung 5.8.: Schnittmenge zwischen der Dilatation aus Abbildung 5.7 und dem Voronoi-Diagramm aus Abbildung 5.6. Schnittpunkte sind rot markiert. Änderungen, um die Polygon in konvexe Polygone zu überführen sind grün markiert.

auf die Straßenstruktur beschränkt, allerdings sind längere Straßen unnötig oft unterteilt. Daher werden abschließend alle benachbarten Polygone, welche aus virtuellen Clusterpunkten erzeugt wurden, vereinigt.

Spezialfall: Brücken

Da das beschriebene Verfahren zur Aufgabe hat, die Eingabedaten in verschiedene Verkehrssituationen zu unterteilen, ist eine semantische Überprüfung in einigen Fällen unerlässlich. Die Verkehrssituationen *Straße* und *Kurve* sind eindeutig identifizierbar, bei einer *Kreuzung* hingegen muss aufgrund der Eingabedaten entschieden werden, ob es sich um eine Kreuzung oder eine Brücke handelt. Aus diesem Grund werden in jeder detektierten Kreuzungszelle die entsprechen-

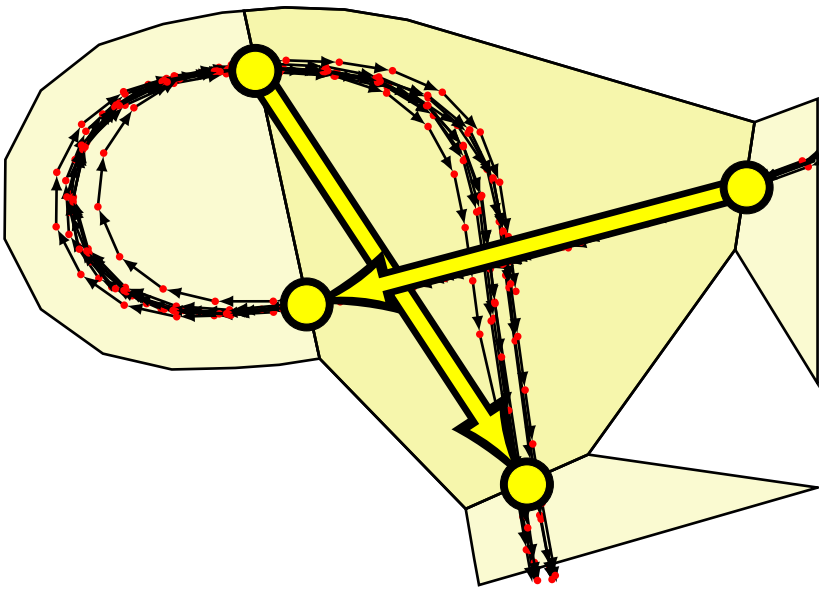


Abbildung 5.9.: Spezialfall Brücke.

den Trajektorien mit dem Zellrand geschnitten und mit einem einfachen Clustering-Algorithmus (DBSCAN mit ϵ_{br}, $\text{minPts}_{\text{br}}$) gruppiert. Anschließend wird aufgrund der inhärenten Trajektorien entschieden, welche Cluster verbunden sind. Somit wird abschließend bestimmt, ob es sich bei der

Verkehrssituation um eine Kreuzung (mehrere Verbindungen in den Clustern) oder um eine Brücke handelt. Abbildung 5.9 zeigt eine detektierte Brücke mit Fahrzeugtrajektorien in schwarz und den gefundenen Clustern und deren disjunkte Verbindungen in gelb.

5.3. Rekonstruktion fahrbahngenauer Straßenkarten

Zur Rekonstruktion fahrbahngenauer digitaler Straßenkarten können beispielsweise die in Kapitel 4 beschriebenen Algorithmen verwendet werden. Diese erzeugen oder aktualisieren mit verschiedenen Ansätzen derartige Karten unter Verwendung von Sensordaten von Fahrzeugflotten oder Einzelfahrzeugen. Eine derartige Karte wird vom, in Abschnitt 5.4 beschriebenen, Verfahren zur Rekonstruktion fahrspurgenauer Straßenkarten zur Initialisierung benötigt. Dabei ist es für das Verfahren von Vorteil gewisse Charakteristika, wie beispielsweise minimal viele Knoten, in der fahrbahngenauen Straßenkarte vorzufinden. Daher wird im Folgenden ein neuer, in Roeth u. a. (2016) veröffentlichter Ansatz zur vollautomatischen Ableitung einer fahrbahngenauen Straßenkarte aus Fahrzeugtrajektorien dargelegt.

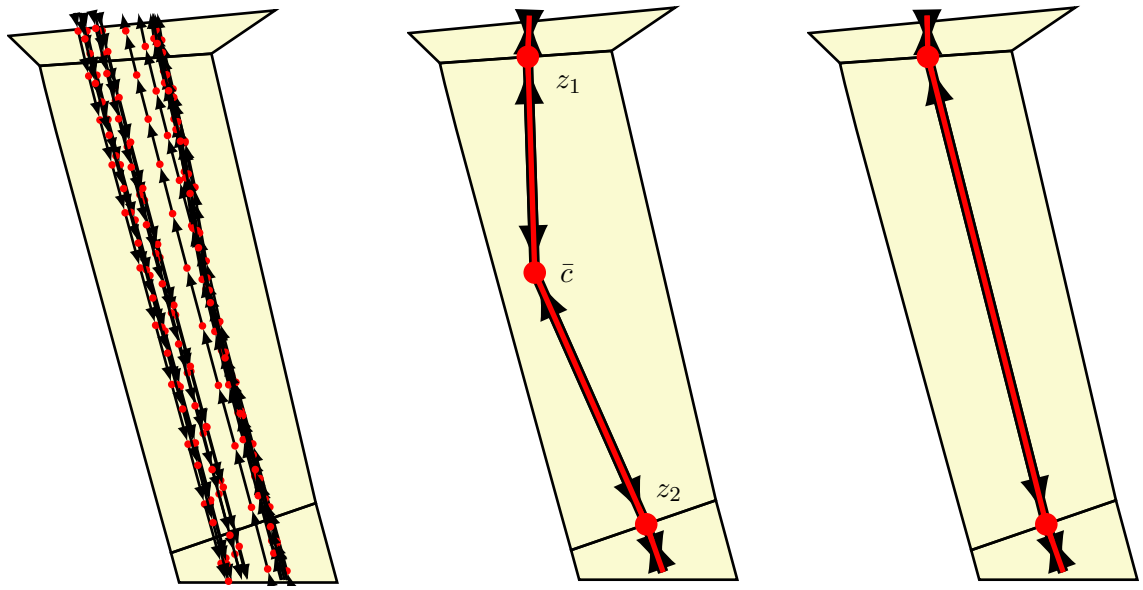
Die grundlegende Idee besteht darin, die Eingabedaten nach der im Abschnitt 5.2 eingeführten Methode zu segmentieren und für jede Zelle der Segmentierung ein Modell in Form eines Graphen (vgl. Abschnitt 2.1) zu erzeugen, welcher die entsprechende Verkehrssituation beschreibt. Jedes dieser Modelle wird anschließend mit Hilfe eines Reversible Jump Markov Chain Monte Carlo Verfahrens (vgl. Abschnitt 3.4) hinsichtlich einer Zielfunktion optimiert, welche die Übereinstimmung zwischen dem Graph und den Fahrzeugtrajektorien beschreibt.

5.3.1. Initialisierung

Der erste Schritt besteht in der Initialisierung des Straßengraphen G_c einer jeden Zelle c. Dazu werden für jede Zelle die Fahrzeugtrajektorien T mit dem Zellenrand geschnitten und die Schnittpunkte eindimensional mit dem DBSCAN Algorithmus (ϵ_r, $\mathrm{minPts}_\mathrm{r}$) zu Z_c gruppiert. Jedes Clusterzentrum $z \in Z_c$ wird als Knoten in den Straßengraphen eingefügt. Anschließend wird der Schwerpunkt $\bar{c}$ der Zelle als Knoten aufgenommen und mit jedem vorhandenen Knoten mit einer doppelt gerichteten Kante verbunden. Der Schwerpunkt als Knoten im Straßengraphen wird benötigt, da andernfalls im Falle einer Kreuzung ein Modell mit mehreren Straßen erzeugt werden müsste, um jeden Übergang zu ermöglichen.

$$G_{c,\mathrm{init}} = \Big(V = \Big\{Z_c, \bar{c}\Big\}, E = \Big\{ \bigcup_{z \in Z_c} (z \times \bar{c}, \bar{c} \times z) \Big\}\Big)$$

$$Z_c = \mathrm{DBSCAN}_{\epsilon_\mathrm{r}, \mathrm{minPts}_\mathrm{r}}\Big[\bigcup_{t \in T} \Big(t \cap c\Big)\Big]$$



(a) Fahrzeugtrajektorien und Segmentierung (*Straße*).

(b) Initialer Straßengraph.

(c) Finaler Straßengraph.

Abbildung 5.10.: Zwischenstufen einer realen Modellentwicklung.

In Abbildung 5.10a sind beispielhafte Trajektorien mit der entsprechenden Zelle der Segmentierung und in Abbildung 5.10b der korrespondierende initiale Graph

$$G_{c,\text{init},5.10\text{b}} = \left(\left\{z_1,z_2,\bar{c}\right\},\left\{(z_1,\bar{c}),(\bar{c},z_1),(z_2,\bar{c}),(\bar{c},z_2)\right\}\right)$$

dargestellt. Algorithmus 4 beschreibt die Initialisierungsprozedur als Pseudocode.

Algorithmus 4 : Initialisierung der Straßengraphen.

for $c \in S$ **do**
  $G_{c,\text{init}} = \{\}$
  $I_c \leftarrow \bigcup_{t \in T} (t \cap c)$ `// $I_c$: Schnittpunkte, $T$: Alle Trajektorien`
  $Z_c \leftarrow \text{DBSCAN}(\text{I}_\text{c})$
  for $z \in Z_c$ **do**
    Füge z als Knoten in $G_{c,\text{init}}$ ein
  end
  $\bar{c} \leftarrow \frac{1}{|c|} \sum_{c_p \in c} c_p$
  Füge $\bar{c}$ als Knoten in $G_{c,\text{init}}$ ein
  for $z \in Z_c$ **do**
    Füge $(z,\bar{c})$ und $(\bar{c},z)$ als Kanten in $G_{c,\text{init}}$ ein
  end
end

5.3.2. Reversible Jump Markov Chain Monte Carlo

Die initialen Straßengraphen jeder Zelle stellen die aktuellen Konfigurationen der Modelle dar. Die Parameter dieser Modelle sind die zweidimensionalen Koordinaten der einzelnen Knoten, sowie die Kanten zwischen ihnen. Diese sollen unter Anwendung der in Abschnitt 3.4 eingeführten RJMCMC Methodik variiert und hinsichtlich einer Bewertung (vgl. Gleichung (5.2)) optimiert werden. Wie in den Grundlagen beschrieben, stehen, in Abhängigkeit der aktuellen Modellkonfiguration, verschiedene Übergangskerne zur Verfügung. Jeder Übergangskern beschreibt eine *Operation*, welche die Konfiguration verändert. Im Folgenden wird zunächst eine Übersicht über diese Operationen gegeben, welche anschließend genauer betrachtet werden.

Die möglichen Operationen teilen sich in zwei Klassen. Zum einen gibt es Aktionen, die lediglich die Werte der vorhandenen Modellparameter variieren und zum anderen jene, die die Struktur und damit die Dimension des Modells verändern. Zur ersten Klasse gehört die *Move* Operation (vgl. Abbildung 5.11), welche die Koordinaten eines Knotens verändert und somit nicht die Dimension des Modells beeinflusst. Zur zweiten Klasse gehören die Operationen *Birth*, *Death*, *Split* und *Merge*. Beim Birth (vgl. Abbildung 5.12a) wird ein neuer Knoten auf einer Kante eingefügt und die Kante dadurch geteilt. Die Death Operation (vgl. Abbildung 5.12b) entfernt einen vorhandenen Knoten aus dem Graphen und bildet somit mit der Birth Operation ein reversibles Paar, dessen Auswahlwahrscheinlichkeiten unter Berücksichtigung der erweiterten Detailed Balance Condition in Gleichung (3.10) festgelegt werden müssen. Gleiches gilt für die Split Operation (vgl. Abbildung 5.13a), welche einen Kreuzungsknoten (Knoten mit mehr als zwei verbundenen Kanten) in zwei benachbarte Knoten auftrennt und die dazu gegensätzliche Merge Operation (vgl. Abbildung 5.13b), welche zwei benachbarte Knoten zu einem Kreuzungsknoten fusioniert.

Die Auswahlwahrscheinlichkeiten für die verschiedenen Operationen werden als identisch angenommen:

$$w_{\text{move}} = w_{\text{birth}} = w_{\text{death}} = w_{\text{split}} = w_{\text{merge}} \qquad (5.1)$$
$$\sum w_{\text{operation}} = 1$$

Dadurch müssen diese im Folgenden nicht mehr explizit berücksichtigt werden.

Die Move Operation

Da die Move Operation den Zustandsraum nicht wechselt, ist der Übergangskern von der Form nach Gleichung (3.6). Um keine Bewegungsrichtung zu präferieren wird die Suchfunktion als *Random Walk*[a] realisiert:

$$\delta m = (\delta x, \delta y) \text{ mit } \delta x, \delta y \sim q_{\text{move}}(s^{(n+1)}|s^{(n)}) = \text{Unif}^{\text{b}}(0, \sigma_{\text{move}})$$

[a] Stochastischer Prozess mit identisch verteilten unabhängigen Zuwächsen auf einem bestimmten Intervall.

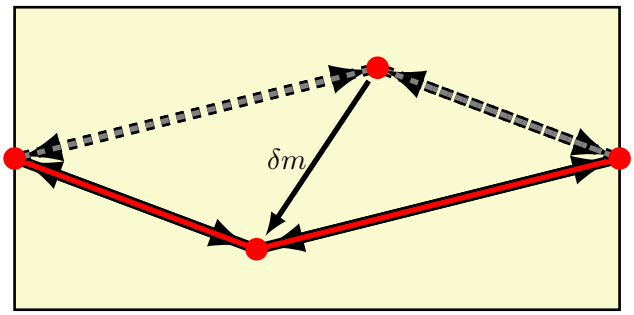


Abbildung 5.11.: Beispielhafte Visualisierung der Move Operation um δm.

Der Parameter σ_{move} beeinflusst dabei, wie weit ein Knoten maximal verschoben werden kann. Abbildung 5.11 zeigt eine beispielhafte Verschiebung eines Knotens um δm.

Die Birth und Death Operation

Die Übergangskerne der Birth und Death Operation werden nach Algorithmus 3 konstruiert. Da das Modell durch diese Operationen verändert wird, stammen die Konfigurationen s' und $s^{(n)}$ aus verschiedenen Zustandsräumen und somit muss eine Transformation gemäß Gleichung (3.9) definiert werden. Betrachtet werden dabei zwei benachbarte Knoten (x_1,y_1), $(x_2,y_2) \in s$ des Zellgraphen.

$$\tau_{\text{birth}}(x_1,y_1,x_2,y_2,u_1,u_2) = (x_1,y_1,x_2,y_2,\bar{x}+u_1,\bar{y}+u_2)$$
$$\bar{x} = \frac{x_1+x_2}{2},\ \bar{y} = \frac{y_1+y_2}{2}$$
$$u_1,u_2 \sim q_{\text{birth}}(s^{(n+1)}|s^{(n)}) = \text{Norm}^{\text{c}}(0,\sigma^2_{\text{birth}}) \in \mathbb{R}$$

Die Akzeptanzwahrscheinlichkeit der Birth Operation wird nach Gleichung (3.13) bestimmt, wobei die Auswahlwahrscheinlichkeiten w aufgrund der angenommenen Gleichheit gekürzt wurden.

$$A_{\text{birth}}(s'|s^{(n)},u_1,u_2) = \min\Big\{1,\frac{\pi(s')\cdot|\Im(\tau_{\text{birth}})|}{\pi(s^{(n)})\cdot q_{\text{birth}}(u_1,u_2|s^{(n)})}\Big\}$$
$$q_{\text{birth}}(u_1,u_2|s^{(n)}) = \frac{1}{2\pi\sigma^2_{\text{birth}}}\cdot\exp(-\frac{u_1^2+u_2^2}{2\sigma^2_{\text{birth}}})$$

[b] vgl. Anhang B.2
[c] vgl. Anhang B.2

Die Jacobi Matrix der Transformation lautet

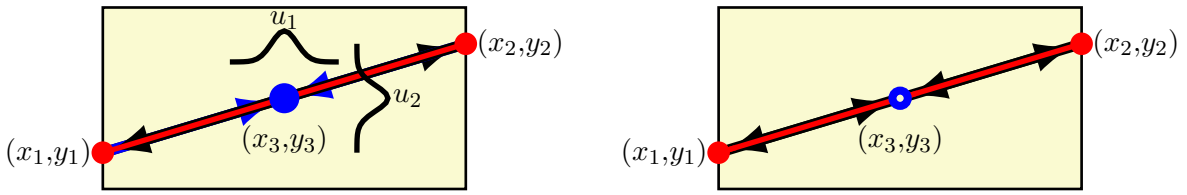
$$\Im(\tau_{\text{birth}}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 1 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 1 \end{pmatrix}$$

mit der Determinante $|\Im(\tau_{\text{birth}})| = 1$, da die Matrix Dreiecksform hat. Abbildung 5.12a zeigt eine beispielhafte Birth Operation.

Die Death Operation kann als umgekehrter Fall betrachtet werden, von drei aufgereihten Knoten wird der mittlere Knoten $(x_3,y_3) \in s$ entfernt. Die Akzeptanzwahrscheinlichkeit dafür wird nach Gleichung (3.14) berechnet, wobei keine zusätzlichen Komponenten gezogen, sondern berechnet werden.

$$A_{\text{death}}(s'|s^{(n)}) = A_{\text{birth}}(s^{(n)}|s',u_1,u_2)^{-1}$$
$$u_1 = |\bar{x} - x_3|,\ u_2 = |\bar{y} - y_3|$$
$$\bar{x} = \frac{x_1 + x_2}{2},\ \bar{y} = \frac{y_1 + y_2}{2}$$

Abbildung 5.12b zeigt eine beispielhafte Death Operation.



(a) Birth Operation *(b)* Death Operation

Abbildung 5.12.: Beispielhafte Visualisierung der Birth und Death Operation. Änderungen sind in Blau dargestellt.

Die Split und Merge Operation

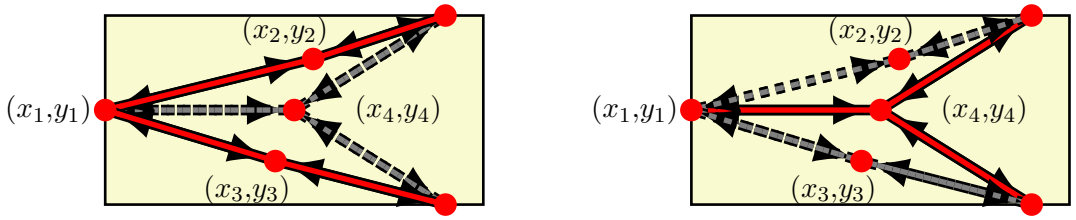
Die Übergangskerne der Split und Merge Operation werden ebenfalls nach Algorithmus 3 konstruiert, wobei die Transformation zwischen den Zustandsräumen gemäß Gleichung (3.9) definiert wird. Dazu wird bei der Split Operation ein Kreuzungsknoten $(x_4,y_4) \in s$, wie in Abbildung 5.13a dargestellt, betrachtet, welcher in zwei nicht verbundene Knoten (x_2,y_2), $(x_3,y_3) \in s'$ zerfällt.

$$\tau_{\text{split}}(s_{\setminus(x_4,y_4)},x_4,y_4,u_1,u_2) = (s_{\setminus(x_4,y_4)},x_4 + u_1,y_4 + u_2,x_4 - u_1,y_4 - u_2)$$
$$u_1,u_2 \sim q_{\text{split}}(s^{(n+1)}|s^{(n)}) = \chi^2_3 \in \mathbb{R} \text{ (vgl. Anhang B.2)}$$

Die Jacobi Matrix der Transformation lautet

$$\Im(\tau_{\text{split}}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{pmatrix}$$

mit der Determinante $|\Im(\tau_{\text{split}})| = 4$.



(a) Split Operation *(b)* Merge Operation

Abbildung 5.13.: Beispielhafte Visualisierung der Split und Merge Operation.

Die Akzeptanzwahrscheinlichkeit der Split Operation wird nach Gleichung (3.13) berechnet:

$$A_{\text{split}}(s'|s^{(n)},u_1,u_2) = \min\Big\{1, \frac{\pi(s') \cdot |\Im(\tau_{\text{split}})|}{\pi(s^{(n)}) \cdot q_{\text{split}}(u_1,u_2|s^{(n)})}\Big\}$$

$$q_{\text{split}}(u_1,u_2|s^{(n)}) = \frac{\sqrt{u_1 u_2}}{2\pi} \cdot \exp(-\frac{u_1+u_2}{2})$$

Die Merge Operation kann als umgekehrter Fall der Split Operation betrachtet werden, wobei zwei benachbarte, nicht verbundene Knoten zu einem Knoten $(x_4,y_4) \in s'$, wie in Abbildung 5.13b dargestellt, vereint werden.

$$A_{\text{merge}}(s'|s^{(n)}) = A_{\text{split}}(s^{(n)}|s',u_1,u_2)^{-1}$$

$$u_1 = |\bar{x} - x_2| = |\bar{x} - x_3|,\ u_2 = |\bar{y} - y_2| = |\bar{y} - y_3|$$

$$\bar{x} = \frac{x_2 + x_3}{2},\ \bar{y} = \frac{y_2 + y_3}{2}$$

5.3.3. Algorithmus

Mit der Segmentierung, den initialen Modellen und den RJMCMC-Operationen kann das Verfahren zur Optimierung der Modelle nach Algorithmus 5 beschrieben werden. Die in den nächsten Abschnitten definierte und zur Auswertung von $A(T)$ in Zeile 9 benötigte Zielfunktion

$$\pi = \beta\pi_d^s + (1-\beta)\pi_p^s \tag{5.2}$$

beschreibt dabei im Term π_d^s die logarithmische Wahrscheinlichkeit, dass die Daten vom aktuellen Modell generiert wurden und im Term π_p^s die vorhandene logarithmische a-priori Wahrschein-

lichkeiten des Modells. $\beta \in [0;1]$ bestimmt das Verhältnis zwischen Daten- und Modellwissen. Wie beispielsweise in (Tournaire u. a., 2010) beschrieben, ist dieser Parameter im Kontext der Anwendung zu interpretieren und zu wählen. Bezüglich der beschriebenen Verwendung bestimmt β vor allem im „Zweifelsfall“ welcher Datenursprung mehr Einfluss hat. Derartige Fälle treten einerseits bei Ausreißern in der Eingabedaten auf, andererseits bei Unterschieden zwischen Modell und Realität. Liegen nicht ausreichend viele Daten zur Rekonstruktion vor, dient π_p^s der Generalisierung. Liegen in den Daten Messfehler vor, dient der Term der Regularisierung und verhindert, dass zu komplexe Modelle erzeugt werden. Unterscheidet sich ein real vermessener Streckenabschnitt stark von der modellhaften Annahme, muss π_d^s in der Lage sein zu bewirken, dass das Modell in diese Richtung verändert wird, ohne dabei von der Modellannahme gehindert zu werden.

Algorithmus 5 : Pseudocode zum Algorithmus zur Rekonstruktion fahrbahngenauen Straßenkarten.

```
1  S ←Abschnitt 5.2                                   // Segmentierung der Eingabedaten
2  G ←Algorithmus 4                                   // Initialisierung der Straßengraphen
3  while ∑ t_SA^(c)(n) > 0 do                         // Abbruchkriterium
4      for c ∈ S do
5          for v ∈ G_c do
6              T ← Wähle zulässige Operation/Übergangskern gemäß Gleichung (5.1)
7              s' ← Erzeuge neuen Zustand gemäß T
8              Ziehe w ∼ Unif^d(0,1)
9              if w < A(T) mit π = π^(1/t_SA^(c)(n)) (Gl. (5.3)) then // A: Operationsabhängig
10                 Setze s^(n+1) := s'
11             else
12                 Setze s^(n+1) := s^(n)
13                 Kühle Funktion t_SA^(c)(n) ab
14             end
15         end
16     end
17 end
```

In Zeile 9 von Algorithmus 5 kommt das in Abschnitt 2.3 beschriebene Simulated Annealing Verfahren zum Einsatz. Motiviert durch die Prozedur in (Andrieu u. a., 2000) ist der Zweck der Anwendung, die Zielfunktion aus Gleichung (5.2) in Abhängigkeit der Laufzeit mit einem exponentiell multiplikativen Abkühlschema zu beeinflussen:

$$\pi \propto \pi^{1/t_{SA}^{(c)}(n)} \tag{5.3}$$

wobei die Funktion $t_{SA}^{(c)}(n)$ für jede Zelle c eine Abkühlfunktion mit $\lim_{n\to\infty} t_{SA}^{(c)}(n) = 0$ und $t_{SA}^{(c)}(0) = 1$ darstellt. Dies bewirkt, dass sich die Entstehung der Markov-Kette auf besser bewertete

[d] vgl. Anhang B.2

Regionen der Zielfunktion konzentriert. In der Praxis bedeutet dies, dass Operationen, welche die Bewertung verschlechtern, mit fortschreitender Laufzeit seltener akzeptiert werden. Der Wert der Abkühlfunktion sinkt, sobald eine vorgeschlagene Operation abgelehnt wurde (vgl. Zeile 13). Die Abkühlfunktion ist dabei eine exponentiell abnehmende Funktion:

$$t_{SA}^{(c)}(n) = \mathrm{e}^{-\lambda n} \in [0,1] \tag{5.4}$$

wobei der Parameter λ so gewählt wird, dass eine bestimmte Anzahl an Schritten s benötigt wird, um eine Temperatur $eps \approx 0$ zu erreichen. Dazu wird Gleichung (5.4) in eine Berechnungsvorschrift in Abhängigkeit der Schrittanzahl überführt:

$$\lambda = -\frac{\log(eps)}{s} \tag{5.5}$$

In Abbildung 5.10c ist beispielhaft der resultierende Straßengraph zum initialen Straßengraph in Abbildung 5.10b abgebildet.

5.3.4. Bewertung

In diesem Abschnitt werden die beiden Bestandteile der Zielfunktion in Gleichung (5.2) definiert.

5.3.4.1. Bewertung bezüglich der Daten

Um in der oben beschriebenen Anwendung π_d^s als Teil der Zielfunktion zu formulieren wird eine Bewertung benötigt, welche ausdrückt, wie gut die aktuellen Modelle die Daten abbilden bzw. wie wahrscheinlich es ist, dass die Eingabedaten von den aktuellen Modellen fehlerbehaftet aufgezeichnet wurden. Um dies zu formulieren, müssen zunächst die Trajektorien auf die einzelnen Straßengraphen abgebildet werden. Für diese Problemstellung ist in (Newson und Krumm, 2009) ein Verfahren beschrieben, um Fahrzeugtrajektorien auf eine Karte abzubilden. Für jedes Modell wird dabei ein *Hidden Markov Modell* (HMM) erzeugt, welches als versteckte Zustände die Kanten des Graphen und als Beobachtungen die Trajektorienpunkte besitzt. Für die HMMs werden anschließend mit Hilfe des Algorithmus von Viterbi (vgl. Abschnitt 2.2) die optimalen Zuordnungen gefunden. Abbildung 5.14 beschreibt beispielhaft die Zuordnung eines Trajektorienabschnitts zu einer Straße.

Die Verteilung π_d^s beschreibt für jede Zelle c die Wahrscheinlichkeit, dass der n-te Messpunkt einer Trajektorie t von jener Kante e des Straßengraphen G_c, auf welche er durch die Lösung des HMM abgebildet wurde, aufgenommen wurde. Diese Kante wird im Folgenden als e_n abgekürzt.

$$\pi_d^s = \prod_{c \in S} \prod_{t \in T_c} \prod_{n \in m} \left(\Upsilon_d \Big[G_c(t_n), t_n \Big] \cdot \Upsilon_\alpha \Big[\gamma(G_c(t_n)), \gamma(t_n) \Big] \right)$$

$$G_c(t_n) = \arg\max_{\bar{e} \in G_c} p(\bar{e}|t_n) = e_n$$

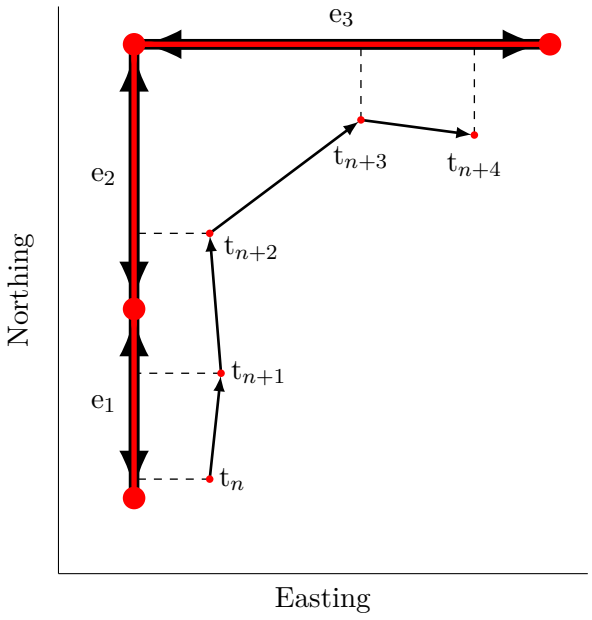


Abbildung 5.14.: Abbildung eines Trajektorienabschnittes auf einen Straßengraph.

Als zu bestimmende Fehlermerkmale werden dabei in Υ_d der euklidische Abstand und in Υ_α der eingeschlossene Fahrtwinkel γ zwischen der Trajektorie t und dem Straßengraphen G_c berücksichtigt und im Folgenden genauer betrachtet.

Um numerische Ungenauigkeiten zu vermeiden, werden die logarithmierten Werte der Wahrscheinlichkeiten summiert:

$$\Pi_d^s = \log(\pi_d^s) = \sum_{c \in S} \sum_{t \in T_c} \sum_{n \in m} \log\Big(\Upsilon_d\Big[G_c(t_n), t_n\Big]\Big) + \log\Big(\Upsilon_\alpha\Big[\gamma(G_c(t_n)), \gamma(t_n)\Big]\Big) \tag{5.6}$$

Um den Fehler Υ_d bezüglich der Distanz zwischen Trajektorie und Straßengraph, wie in Abbildung 5.15 dargestellt, zu formulieren, wird die Annahme getroffen, dass der Fehler in der Positionierung einer Normalverteilung unterliegt. Dementsprechend kann die Wahrscheinlichkeit $p(d_n)$, dass eine um mindestens d_n fehlerhafte Messung eines Trajektorienpunktes t_n von einer bestimmten Kante e_n des Straßengraphen aufgezeichnet wurde, mit Hilfe einer Normalverteilung berechnet werden:

$$\Upsilon_d(e_n, t_n) = p(t_n | e_n) = p(d_n) = P(X \in [d_n, \infty]) = 2 \cdot \int_{d_n}^{\infty} \frac{1}{\sqrt{2\pi\sigma_t^2}} \cdot \mathrm{e}^{-\frac{1}{2}\left(\frac{x}{\sigma_t}\right)^2} \mathrm{d}x \tag{5.7}$$

wobei der Parameter σ_d vom verwendeten GNSS System abhängt.

Bezüglich des Fehlers im Fahrtwinkel, wie in Abbildung 5.16 dargestellt, wird ebenfalls angenommen, dass dieser einer Normalverteilung unterliegt, da der Betrag unmittelbar vom Positionierungsfehler abhängt. $\gamma(\cdot)$ beschreibt dabei die Fahrtrichtung auf einer Kante bzw. die Ausgangsrichtung einer Kante aus einem Punkt. Dementsprechend kann die Wahrscheinlichkeit $p(\alpha_n)$, dass eine im Fahrtwinkel um mindestens $\alpha_n \in [0, \pi]$ fehlerhafte Messung eines Trajektorienpunktes t_n von einer

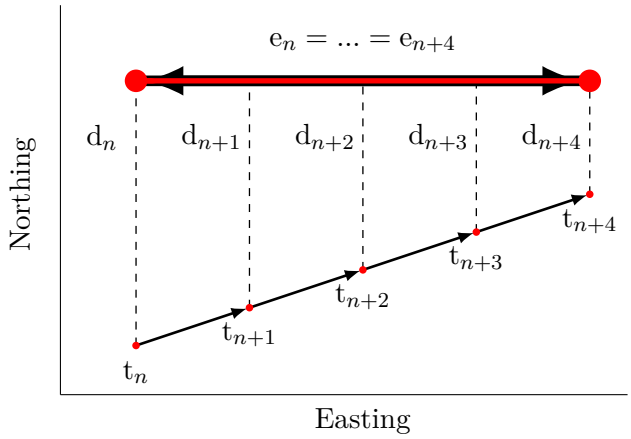


Abbildung 5.15.: Bewertungskriterium bzgl. des euklidischen Abstandes zwischen Trajektorie und Straßengraph.

bestimmten Kante e_n des Straßengraphen aufgezeichnet wurde mit Hilfe einer Normalverteilung berechnet werden:

$$\Upsilon_\alpha(\gamma(e_n),\gamma(t_n)) = p(\gamma(t_n)|\gamma(e_n)) = p(\alpha_n) = P(X \in [\alpha_n,\pi]) = 2 \cdot \int_{\alpha_n}^{\pi} \frac{1}{\sqrt{2\pi\sigma_\alpha^2}} \cdot \mathrm{e}^{-\frac{1}{2}\left(\frac{x}{\sigma_\alpha}\right)^2} \mathrm{d}x \quad (5.8)$$

Der Parameter σ_α hängt vom verwendeten GNSS-System ab.

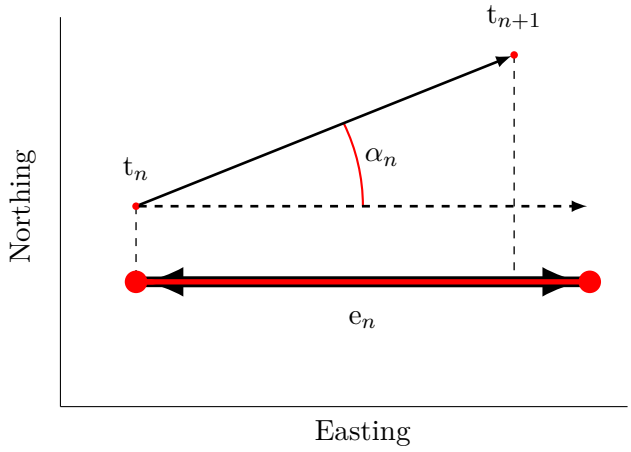


Abbildung 5.16.: Bewertungskriterium bzgl. des Fahrtwinkels zwischen Trajektorie und Straßengraph.

5.3.4.2. A-priori Informationen

Um das in Abschnitt 3.1 beschriebene Overfitting und unrealistische Straßennetze zu vermeiden wird in Gleichung (5.2) in π_p^s (bzw. logarithmiert Π_p^s) Vorwissen über das Modell berücksichtigt. Da im Kontext der Beschreibung von Straßenverläufen gelegentlich sehr spezielle Situationen auftreten können, ist es nicht sinnvoll, den Hypothesenraum des Modells auf bestimmte Bereiche einzuschränken. Stattdessen wird ein Regularisierungsterm eingeführt, welcher gewisse Ausprägungen bestraft, aber nicht vermeidet, sodass jedes Model stochastisch möglich ist.

Um Overfitting des Modells zu vermeiden, werden Knoten $v \in V$ des Graphen G, welche keine Fahrtwinkeländerung γ bewirken, bestraft. Dies bewirkt, dass so wenig Knoten wie möglich verwendet werden und diese hauptsächlich Kurven oder Kreuzungen beschreiben. Zusätzlich sollen Spitzkehren im Straßenverlauf bestraft werden, da das Modell durch die erste Regularisierung dazu neigt *Zick-Zack* Verläufe zu erzeugen. Daher wird als *Ziel* eines *optimalen* Knotens eine Fahrtwinkeländerung von 90 ° definiert.

$$\Pi_p^s = \sum_{v \in G} \log\Big(\frac{2}{\pi}(1 - \Big|1 - \frac{2\gamma(v)}{\pi}\Big|)\Big),\ \gamma(v) \in [0,\pi]$$

5.4. Rekonstruktion fahrspurgenauer Straßenkarten

Zukünftige Fahrfunktionen und Fahrerassistenzsysteme benötigen digitale Straßenkarten mit mehr Informationsgehalt als die bisher behandelten fahrbahngenauen Straßenkarten. Von der fahrspurgenauen Navigation bis hin zur vollautomatischen Steuerung von Fahrzeugen wird präzises Kartenmaterial benötigt, welches einerseits topologische Informationen über Anzahl und Konnektivität von Fahrspuren und andererseits geometrische Informationen über deren Breite oder Fahrbahnmarkierungen vorhält. In diesem Abschnitt wird ein, in Roeth u. a. (2017) veröffentlichtes Verfahren erläutert, welches in der Lage ist, eine fahrspurgenaue Straßenkarte zu erzeugen, indem eine fahrbahngenaue Karte, wie sie in Abschnitt 5.3 konstruiert wird, um die genannten Informationen angereichert wird.

5.4.1. Modell

Im fahrspurgenauen Modell zur Repräsentation einer Straßenkarte werden für eine Straße und eine Kreuzung je ein Untermodell definiert, da sich diese Verkehrssituationen grundlegend unterscheiden. In diesem Abschnitt werden die beiden Untermodelle beschrieben.

5.4.1.1. Straße

In der bisher betrachteten Repräsentation einer Straßenkarte wurde als Modell für eine Fahrbahn ein gerichteter gewichteter Graph verwendet, welcher maximal für einen Abschnitt gültige Informationen, wie beispielsweise Höchstgeschwindigkeiten, enthalten kann. Das Modell eines Verkehrsnetzes auf Fahrspurebene benötigt mehr Facetten, basiert allerdings als Referenz auf einem fahrbahngenauen Graphen und wird im Folgenden dargestellt.

Zunächst wird jede Straße des zu Grunde liegenden Fahrbahngraphen parametrisiert, sodass jeder Punkt der Straße durch einen Wert aus $[0; 1]$ beschrieben werden kann. In Abbildung 5.17 ist eine beispielhafte Parametrisierung dargestellt.

Mithilfe der Parametrisierung P kann eine Straße eindimensional in verschiedene Bereiche, wie in Abbildung 5.17 (unten) dargestellt, eingeteilt werden. Diese Einteilung wird verwendet, um individuelle Verkehrssituationen auf Fahrspurebene zu kapseln. Ein jeder solcher Bereich enthält

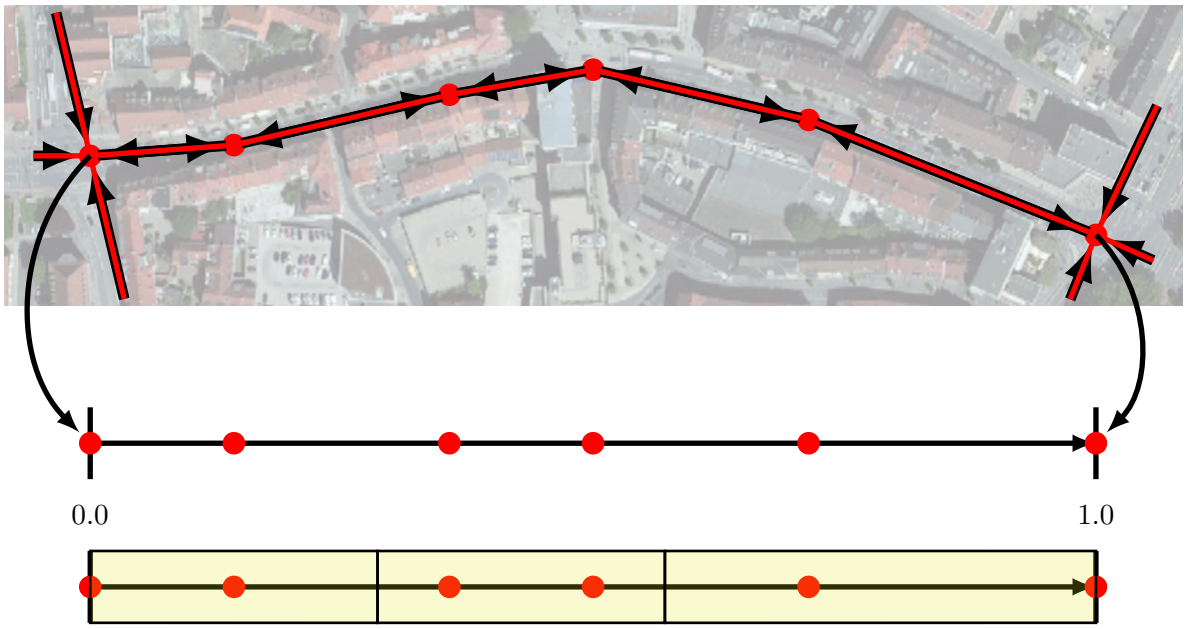


Abbildung 5.17.: Parametrisierung einer Straße mit einer beispielhaften Einteilung in Blöcke.

individuelle Informationen, womit ein mehrdimensionales Straßenmodell mit geometrischen und topologischen Informationen definiert wird. Ein derartiges Modell wird im Folgenden als *Block B* bezeichnet und umfasst die nachstehenden, in Abbildung 5.18 visualisierten Eigenschaften:

1. Anzahl der Fahrspuren L
2. Breite der einzelnen Fahrspuren W
3. Typ T der Fahrbahnmarkierungen jeder Fahrspur (geschlossen, gestrichelt,...)
4. Größe der baulichen Trennung G zwischen den entgegengesetzten Fahrspuren
5. Krümmung C

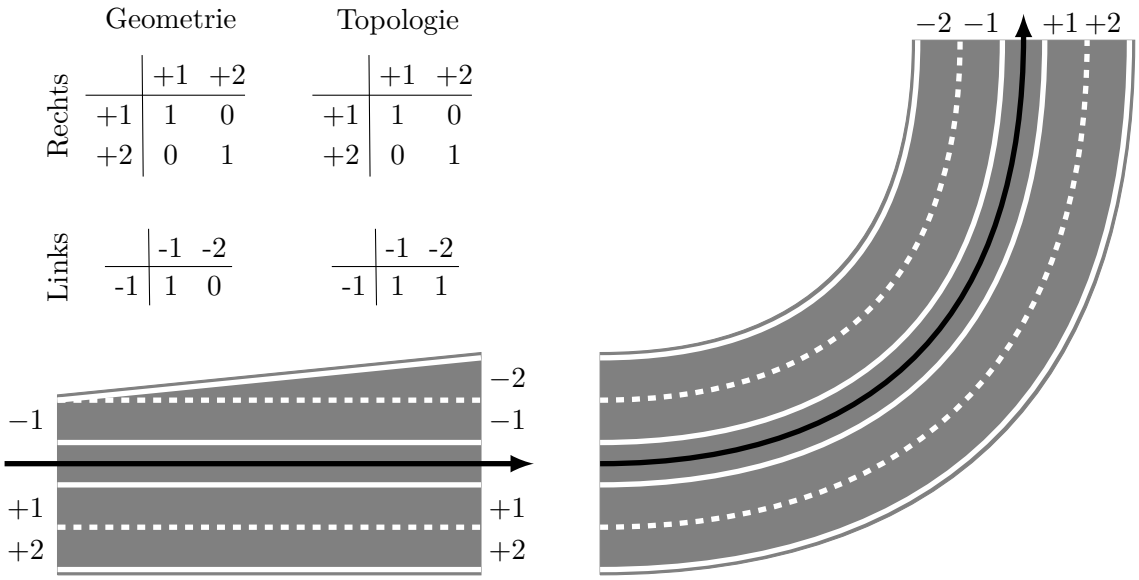
Ein allgemeiner Block ist somit definiert als:

$$B = (L,W,T,G,C)$$

Das Modell einer gesamten Straße besteht aus m Blöcken mit den Parametrisierungswerten $P = (p_1 = 0.0,...,p_{m+1} = 1.0)$ und ist definiert als:

$$\xi_s = (P,(B_1,...,B_m))$$

Um einen gekrümmten Block darzustellen, werden alle Verbindungen durch kubische Hermite Polynome beschrieben. Dadurch ist es möglich, dass ein Block nicht nur eine konstante Krümmung aufweist, sondern einen kubischen Verlauf nehmen kann, wobei lediglich die Randbedingungen der Stetigkeit und Differenzierbarkeit zwischen den Blöcken eingehalten werden müssen, um einen realistischen Straßenverlauf zu erzeugen. Die Randbedingungen werden durch die Vorgabe der



(a) Verbindungsblock *(b)* Straßenblock

Abbildung 5.18.: Spezifikationen eines allgemeinen Blockes.

Anschlusspunkte und der normierten Richtungsvektoren in den Anschlusspunkten eingebracht. Um den Verlauf der Polynome zu beeinflussen, ist ein Skalierungsfaktor der Richtungsvektoren als Parameter des Modells zugänglich.

Ein allgemeiner Block kann durch weitere Einschränkungen oder Ergänzungen spezifiziert werden. Ein *Straßenblock* B_s, wie in Abbildung 5.18b abgebildet, enthält die Einschränkung, dass die Anzahl der Fahrspuren über die Verkehrssituation konstant bleibt. Somit kann ein Straßenstück abgebildet werden, auf welchem sich lediglich die inhärenten Eigenschaften wie die Spurbreiten ändern.

Ein *Verbindungsblock* B_c, wie in Abbildung 5.18a dargestellt, beschreibt eine Verkehrssituation, in welcher sich die Anzahl der Spuren ändert und in der somit eine Zusammenführung oder Aufspaltung von Fahrspuren modelliert werden kann. Daher wird dieser Blocktyp durch ein Tupel $R = (R_{\mathrm{GR}},R_{\mathrm{TR}},R_{\mathrm{GL}},R_{\mathrm{TL}})$ ergänzt, welche sowohl geometrisch beschreibt, welche Fahrspur verschwindet bzw. erzeugt wird, als auch topologisch definiert, welche Fahrspuren verbunden sind. Wie in Abbildung 5.18a dargestellt, wird für jede Fahrtrichtung eine individuelle geometrische $R_{\mathrm{GR}},R_{\mathrm{GL}}$ und topologische $R_{\mathrm{TR}},R_{\mathrm{TL}}$ Matrix angegeben. Eine existierende Information (Verbundenheit) wird binär durch eine Eins verdeutlicht. Beispielsweise ist es in der dargestellten Situation topologisch möglich, auf der linken Seite in Fahrtrichtung von der Fahrspur -2 auf die angrenzende Fahrspur -1 zu wechseln, da eine Vereinigung stattfindet. Diese topologische Information ist nicht gleichbedeutend mit einem einfachen Spurwechselmanöver, diese Restriktion wird durch die Art der Fahrbahnmarkierung impliziert. Ein Verbindungsblock ist definiert als:

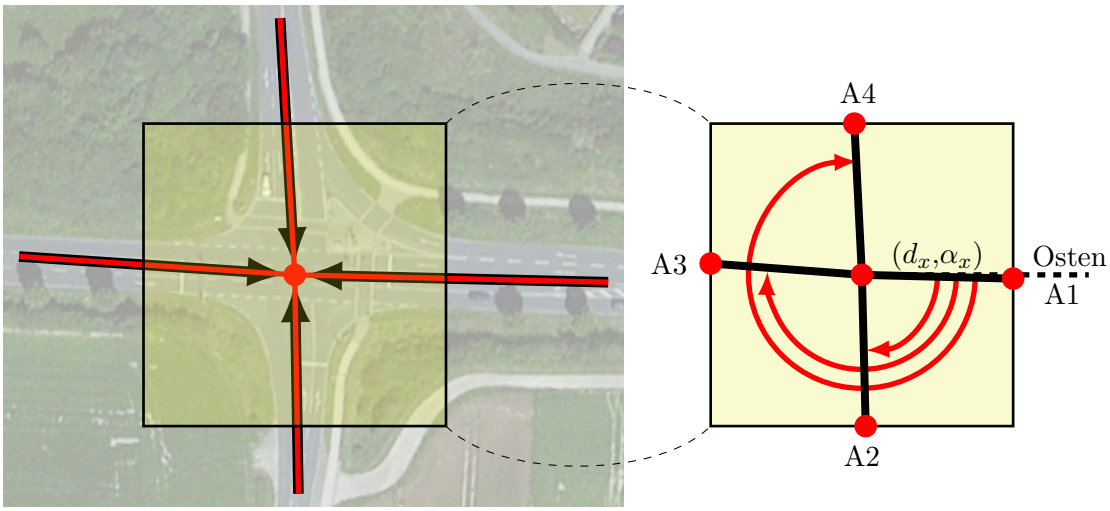
$$B_c = (B,R)$$

Bezüglich einer gesamten Straße ist zusätzlich vorgegeben, dass zwei Straßenblöcke durch einen Verbindungsblock getrennt sein müssen, um gegebenenfalls die Unterschiede zwischen den Straßen-

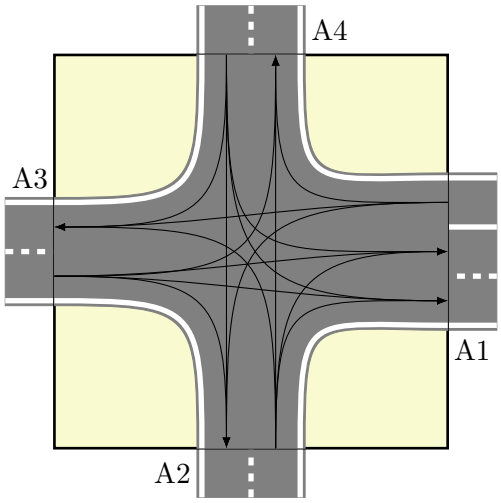
blöcken (z. B. bzgl. der Fahrspuranzahl) auszugleichen. Sollte keine Änderungen nötig sein, stellt der Verbindungsblock als Spezialfall einen Straßenblock dar.

5.4.1.2. Kreuzung

In der bisherigen Repräsentation einer Straßenkarte wurde eine Kreuzung durch einen Knoten, welcher mit mehr als zwei Kanten verbunden ist, dargestellt. Für die fahrspurgenaue Abbildung einer Kreuzung werden sowohl geometrische Informationen über die befahrbare Fläche, als auch topologische Informationen über die Konnektivität der ein- und ausführenden Spuren und deren Fahrweg über die Kreuzungsfläche benötigt. Das Kreuzungsmodell ξ_c setzt sich aus zwei verschiedenen Einzelmodellen zusammen.



(a) Äußeres Kreuzungsmodell.



	$A1_{-1}$	$A1_{-2}$	$A2_{-1}$	$A3_{-1}$	$A4_{-1}$
$A1_{+1}$	$F_{x,y}$	...	...	...	...
$A2_{+1}$	...	...	...	...	...
$A3_{+1}$	...	...	...	...	...
$A4_{+1}$	...	...	...	...	...

(b) Inneres Kreuzungsmodell mit Matrix.

Abbildung 5.19.: Das Kreuzungsmodell.

In Abbildung 5.19a ist das *äußere* Modell $\xi_{c,o}$ dargestellt. Zur Initialisierung werden die Knoten des fahrbahngenauen Graphen untersucht und anhand der verbundenen Kanten Kreuzungsknoten

identifiziert. Da eine große Kreuzung im Graphen durch mehrere Knoten beschrieben sein kann, werden mithilfe eines Distanzwertes d_{cr} gegebenenfalls mehrere Kandidaten zusammengefasst. Das äußere Modell wird auf dem Zentrum der beteiligten Knoten erzeugt. Anhand der identifizierten Kreuzungsknoten kann direkt die Information entnommen werden, wie viele Straßen mit der Kreuzung verbunden sind. Für jede Straße wird ein *Arm* erzeugt. Jeder Arm verfügt über einen Distanzparameter d, welcher den Abstand vom Zentrum zum Beginn der Kreuzungsfläche bezüglich dieses Armes beschreibt und einen Winkelparameter α, welcher relativ zum Ostvektor einen Drehwinkel definiert. Durch diese beiden Parameter ist für jeden Arm der Übergangspunkt von der Straße in die Kreuzungsfläche festgelegt.

Das *innere* Modell $\xi_{c,i}$ in Abbildung 5.19b beschreibt die Konnektivität der Fahrspuren. Jeder Arm verfügt dabei über die gleichen Informationen wie ein allgemeiner Block des Straßenmodells, wodurch die Geometrie der verbundenen Straßen festgelegt ist. Zwischen jeder ein- und ausführenden Fahrspur kann eine Verbindung bestehen, deren Verlauf durch ein kubisches Hermite Polynom beschrieben wird. Somit wird jede Verbindung durch einen Parameter beeinflusst, der den Verlauf über die Kreuzungsfläche prägt. Die Parameter werden in einer separaten Matrix gespeichert (vgl. Abbildung 5.19b), wobei ein Wert von 0 angibt, dass die Verbindung nicht existiert. Durch die Identifizierung der äußeren Spurverläufe wird zusätzlich die Begrenzung der Kreuzungsfläche definiert.

5.4.2. Initialisierung

Zur Initialisierung der Modelle wird der fahrbahngenaue Straßengraph in Straßen und Kreuzungen aufgeteilt. Auf jeder Kreuzung wird ein initiales Modell erzeugt, wobei die Anzahl der angeschlossenen Straßen und somit die Winkelparameter der Arme aus dem Graphen bestimmt werden. Aus der Erzeugung der fahrbahngenauen Straßenkarte ist bekannt, welche Eingabedaten auf die Kreuzung abgebildet werden[e]. Zur Bestimmung der initialen Armabstände werden für jeden Arm der Kreuzung die Eingabedaten vom zum Mittelpunkt nächsten Messpunkt soweit zurück traversiert, bis der durchschnittliche Trajektorienverlauf eine geringere Fahrtwinkeländerung als δ_{init} aufweist. Dieser Abstand oder ein Minimalwert von $d_{\text{init,min}}$ bzw. ein Maximalwert von $d_{\text{init,max}}$ wird als initialer Wert d_{init} verwendet. Anschließend werden die Straßenmodelle zwischen den Kreuzungen generiert, wobei eine Straße im Graph durch eine Sequenz $\{v_x,...v_y\}$ beschrieben wird. Im fahrspurgenauen Modell wird auf jedem Knoten ein Verbindungsblock und dazwischen ein Straßenblock erzeugt. Jede Straße beginnt und endet mit einem Verbindungsblock, welcher gegebenenfalls die Unterschiede zwischen dem angrenzenden Straßenblock und dem Kreuzungsanschluss korrigieren kann. Jede Straße wird als zweispurig, je eine Fahrspur pro Richtung, initialisiert. Die Gesamtheit der a Straßen- und b Kreuzungsmodelle wird im Folgenden als $\Phi = \{\xi_s^1,...,\xi_s^a,\xi_c^1,...,\xi_c^b\}$ bezeichnet.

[e] Alternativ kann diese Abbildung wie beispielsweise in Abschnitt 5.3.4.1 beschrieben erzeugt werden

5.4.3. Reversible Jump Markov Chain Monte Carlo

Die initialisierten Modelle Φ stellen die aktuelle Konfiguration des Gesamtmodells dar. Die Parameter dieser Modelle sind die beschriebenen Eigenschaften der Blöcke und Kreuzungen. Da diese analog zum Vorgehen in Abschnitt 5.3 variiert werden sollen, werden im Folgenden die möglichen Operationen und die entsprechenden Übergangskerne eingeführt.

Für alle Modelle gibt es einerseits Operationen, welche lediglich die Werte der vorhandenen Parameter beeinflussen und andererseits jene, welche die Dimension des Modells verändern.

5.4.3.1. Straße

Bezüglich eines Straßenmodells werden die Operationen in zwei Klassen unterteilt. Die Operationen *Split*, *Merge* und *Adjust Parameter*, wie in Abbildung 5.20 dargestellt, verändern das Modell auf Blockebene, das heißt die individuellen Eigenschaften werden nicht verändert, sondern nur die Anzahl der Blöcke und deren räumliche Ausdehnung. Bei der Split Operation wird ein vorhandener Straßenblock in zwei Straßenblöcke und einen Verbindungsblock aufgeteilt. Die Merge Operation verbindet entsprechend eine derartige Konstellation und bildet mit der Split Operation ein reversibles Paar, dessen Auswahlwahrscheinlichkeiten so gewählt werden müssen, dass die erweiterte Detailed Balance Condition (3.10) erfüllt ist. Bei der Aktion Adjust Parameter werden die Grenzen eines Blocks bzgl. der Parametrisierung des Straßenmodells verändert.

Die Operationen *Add Lane*, *Remove Lane*, *Adjust Middlegap*, *Adjust Lanewidth* und *Adjust Curvature*, wie in Abbildung 5.21 dargestellt, verändern die Eigenschaften eines Straßenblockes. Die Eigenschaften eines Verbindungsblockes können nicht aktiv, sondern nur passiv verändert werden. Diese passen ihre Parameter den angrenzenden Straßenblöcken an. Die Operationen Add und Remove Lane bilden dabei ebenfalls ein reversibles Paar, während die drei Adjustierungsoperationen lediglich die Werte der Parameter verändern.

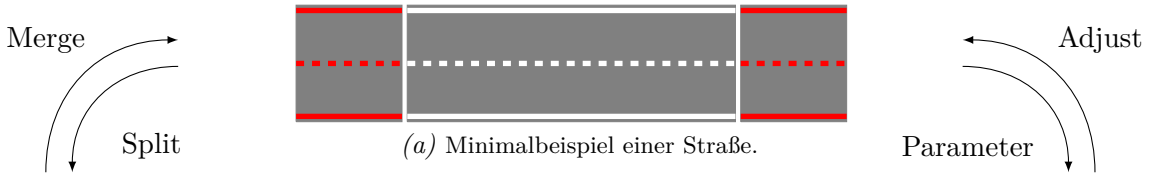
Zu jeder Operation gibt es eine Auswahlwahrscheinlichkeit. Da vor allem zu Beginn der Optimierung grundlegende Eigenschaften vor den Details gefunden werden müssen, werden in Abschnitt 5.4.4 verschiedene Phasen des Algorithmus mit verschiedenen Auswahlwahrscheinlichkeiten eingeführt, wobei immer gilt:

$$\sum w_{\text{operation}} = 1$$

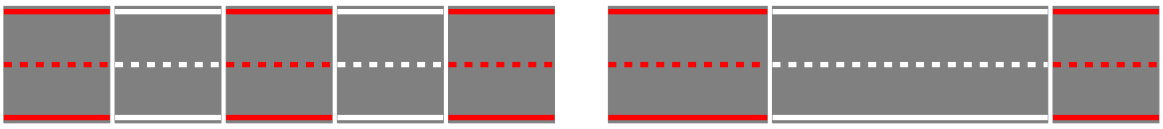
Im Folgenden werden die einzelnen Operationen genauer betrachtet.

Die Split und Merge Operation

Der Übergang von Abbildung 5.20a zu Abbildung 5.20b erfolgt mittels der Split Operation. Die longitudinale Ausdehnung des zu teilenden Straßenblocks B_s ist über die Parametrisierung P des Straßenmodells ξ_s mit den Werten $(p_s,p_e) \in P$ mit der parametrisierten Länge $p_l = p_e - p_s$ definiert. Zur Teilung werden zwei neue Werte $u_1,u_2 \in [0,p_l]$ auf dieser Länge mit $u_1 < u_2$ benötigt.



(a) Minimalbeispiel einer Straße.



(b) In den Straßenblock wurde ein Verbindungsblock eingefügt.

(c) Der longitudinale Parametrisierungswert des ersten Verbindungsblocks wurde geändert.

Abbildung 5.20.: Operationen auf Block Ebene (Die Fahrbahnmarkierungen der Verbindungsblöcke sind rot dargestellt).

Da die Dimension des Modells durch diese Operation verändert wird, wird der Übergangskern nach Algorithmus 3 konstruiert und es muss eine Transformation gemäß Gleichung (3.9) definiert werden:

$$\begin{aligned} \tau_{\text{split}}(p_s,p_e,u_1,u_2) &= (p_s,p_e,p_s+u_1,p_s+u_2) \\ \text{mit } P' &= \{p_1,\ldots,p_s,p_s+u_1,p_s+u_2,p_e,\ldots,p_{m+1}\} \\ u_1,u_2 \sim q_{\text{split}}(s^{(n+1)}|s^{(n)}) &= \text{Unif}^{\text{f}}(0,p_l) \in \mathbb{R} \end{aligned}$$

Nach Gleichung (3.13) wird die Akzeptanzwahrscheinlichkeit bestimmt:

$$A_{\text{split}}(s'|s^{(n)},u_1,u_2) = \min\Big\{1, \frac{\pi(s')}{\pi(s^{(n)})} \cdot \frac{|\Im(\tau_{\text{split}})|}{p_l^2}\Big\}$$

Die Jacobi Matrix der Transformation lautet

$$\Im(\tau_{\text{split}}) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

mit der Determinante $|\Im(\tau_{\text{split}})| = 1$, da die Matrix Dreiecksform hat.

Die Merge Operation kann als umgekehrter Fall angesehen werden. Die Konstellation *Straßenblock, Verbindungsblock, Straßenblock* wird zusammengefasst, wobei für die Transformation keine neuen Komponenten benötigt, sondern berechnet werden. Die Konstellation ist in der Parametrisierung P des Straßenmodells durch die Folge $\{p_a,p_b,p_c,p_d\}$ definiert. Die Akzeptanzwahrscheinlichkeit wird nach Gleichung (3.14) berechnet:

$$\begin{aligned} A_{\text{merge}}(s'|s^{(n)}) &= A_{\text{split}}(s^{(n)}|s',u_1,u_2)^{-1} \\ \text{mit } u_1 = p_b - p_a,\ u_2 &= p_c - p_a \end{aligned}$$

[f] vgl. Anhang B.2

Die Adjust Parameter Operation

Die Adjust Parameter Operation beschreibt als Übergang zwischen Abbildung 5.20a und Abbildung 5.20c die Veränderung eines der beiden Parametrisierungswerte eines Straßenblocks. Da dabei der Zustandsraum nicht gewechselt wird, ist der Übergangskern von der Bauart nach Gleichung (3.6). Der Parameter kann dabei maximal um die halbe parametrisierte Länge p_l des Blockes verändert werden. Um keine Bewegungsrichtung zu präferieren wird die Suchfunktion als Random Walk realisiert:

$$\delta p \sim q_{\text{parameter}}(s^{(n+1)}|s^{(n)}) = \text{Unif}^{\text{g}}(-\frac{\text{p}_\text{l}}{2},\frac{\text{p}_\text{l}}{2}) \in \mathbb{R}$$

Die Add Lane und Remove Lane Operation

Um, wie in Abbildung 5.21a dargestellt, eine neue Fahrspur mit der Add Lane Operation einzufügen, wird das Blockmodel um eine neue Fahrspurbreite ergänzt. Die neue Spurbreite wird aus einer Normalverteilung gezogen, deren Erwartungswert und Varianz sich aus straßenbaulichen Vorgaben ergeben. Diese müssen im Kontext des Szenarios bzw. der Art der Straße bestimmt werden. Für die Transformation folgt:

$$\tau_{add\ lane}(W,u) = (W,u)$$
$$u \sim q_{\text{add lane}}(s^{(n+1)}|s^{(n)}) = \text{Norm}^6(\mu_{\text{add lane}},\sigma^2_{\text{add lane}})$$

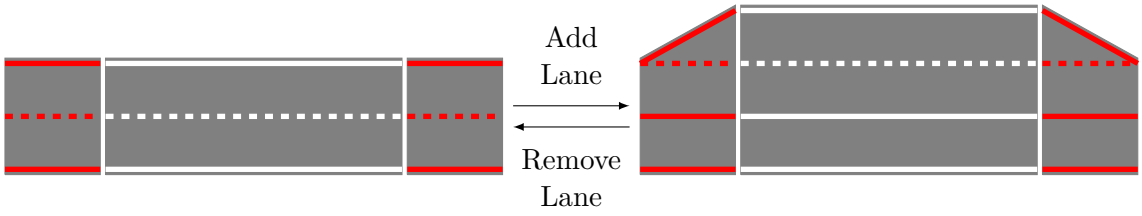
Die Determinante der Jacobimatrix ist $|\Im(\tau_{\text{add lane}})| = 1$ und die Akzeptanzwahrscheinlichkeit ergibt sich zu:

$$A_{\text{add lane}}(s'|s^{(n)},u) = \min\Big\{1,\frac{\pi(s')\cdot|\Im(\tau_{\text{add lane}})|}{\pi(s^{(n)})\cdot q_{\text{add lane}}(u|s^{(n)})}\Big\}$$
$$q_{\text{add lane}}(u|s^{(n)}) = \frac{1}{\sqrt{2\pi\sigma^2_{\text{add lane}}}}\cdot\exp^{-\frac{(u-\mu_{\text{add lane}})^2}{2\sigma^2_{\text{add lane}}}}$$

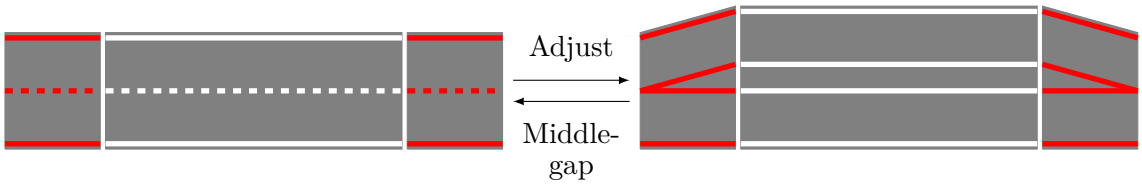
Die Akzeptanzwahrscheinlichkeit für die gegensätzliche Operation Remove Lane wird entsprechend berechnet, wobei die Komponente u die Spurbreite der zu entfernenden Fahrspur ist:

$$A_{\text{remove lane}}(s'|s^{(n)}) = A_{\text{add lane}}(s^{(n)}|s',u)^{-1}$$

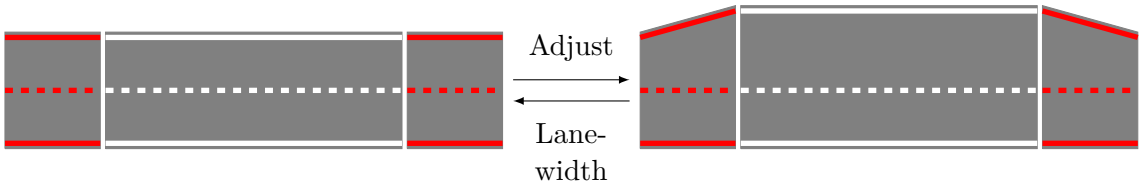
[g] vgl. Anhang B.2



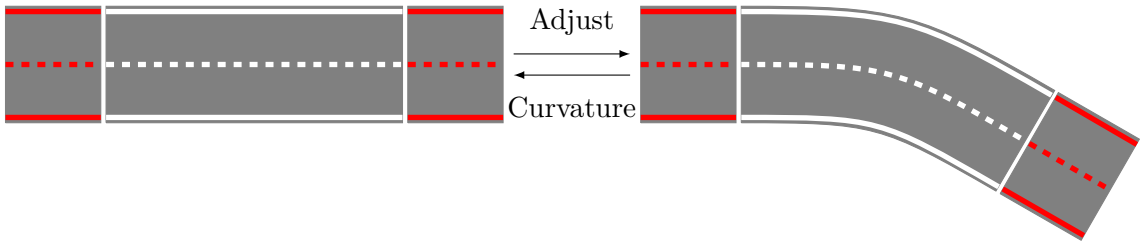
(a) Hinzufügen bzw. Entfernen einer Fahrspur.



(b) Änderung der Größe der baulichen Trennung zwischen den Fahrtrichtungen.



(c) Änderung der Breite einer Fahrspur.



(d) Änderung der Krümmung eines Blocks.

Abbildung 5.21.: Operationen auf Spur Ebene (Die Fahrbahnmarkierungen der Verbindungsblöcke sind rot dargestellt).

Die drei in Abbildung 5.21b - 5.21d dargestellten Operationen verändern die vorhandenen Parameter der Modelle. Daher werden die entsprechenden Suchfunktionen als Random Walks realisiert:

$$\delta g \sim q_{\text{middle gap}}(s^{(n+1)}|s^{(n)}) = \text{Unif}(0,\sigma_{\text{middle gap}})$$
$$\delta w \sim q_{\text{lane width}}(s^{(n+1)}|s^{(n)}) = \text{Unif}(0,\sigma_{\text{lane width}})$$
$$\delta c \sim q_{\text{curvature}}(s^{(n+1)}|s^{(n)}) = \text{Unif}(0,\sigma_{\text{curvature}})$$

5.4.3.2. Kreuzung

Da das Kreuzungsmodell aus zwei Modellen besteht, gibt es für jedes Untermodell verschiedene Operationen. Wie in Abbildung 5.19a dargestellt, beeinflussen die zwei Parameter *Distanz* d und *Winkel* α die Gestalt des äußeren Modells. Die Anzahl der Arme wird bereits im Initialisierungsprozess aus dem Straßengraphen extrahiert und wird nicht mehr verändert. Somit werden zwei Operationen definiert, welche die beiden Parameter, aber nicht die Dimension des Modells verändern.

Daher sind die Übergangskerne von der Bauart nach Gleichung (3.6) und die Suchfunktionen werden als Random Walks umgesetzt:

$$\delta d \sim q_{\text{distance}}(s^{(n+1)}|s^{(n)}) = \text{Unif}(0,\sigma_{\text{distance}}) \in \mathbb{R}$$
$$\delta\alpha \sim q_{\text{angle}}(s^{(n+1)}|s^{(n)}) = \text{Unif}(0,\sigma_{\text{angle}}) \in \mathbb{R}$$

Im inneren Modell verfügt nach der Definition in Abschnitt 5.4.1 jeder Arm über einen Anschlussquerschnitt, welcher die gleichen Eigenschaften wie ein Straßenblock hat. Da an jedem Arm ein Verbindungsblock angeschlossen ist, reagiert dieser auf die Änderungen des Armes genau wie auf Änderungen eines Straßenblocks. Daher sind die Operationen zum Variieren der Anschlusseigenschaften identisch zu den bereits definierten eines Straßenblocks. Die Beeinflussung der Skalierungsfaktoren aus Abbildung 5.19b geschieht nicht durch eine RJMCMC Operation. Stattdessen werden die Trajektorien auf das Modell abgebildet, um für jede Verbindungsmöglichkeit zu bestimmen, ob diese existiert. Ist dies der Fall, wird jeder Faktor iterativ, passend zum Trajektorienbündel der Verbindung gesucht.

5.4.4. Algorithmus

Nachdem alle Modelle initialisiert wurden, kann das Verfahren mithilfe der definierten RJMCMC Operationen nach Algorithmus 6 beschrieben werden.

Die im nächsten Abschnitt definierte und zur Auswertung von $A(T)$ in Zeile 9 bzw. Zeile 23 benötigte Zielfunktion

$$\pi = \delta\pi_d^l + (1-\delta)\pi_p^l \tag{5.9}$$

berücksichtigt dabei sowohl die Übereinstimmung zwischen den Daten und dem Modell in π_d^l, als auch vorhandenes Vorwissen, über zu vermeidende und zu bestärkende Entwicklungen des Modells in π_p^l. $\delta \in [0;1]$ bestimmt das Verhältnis zwischen dem Daten- und dem Modellwissen.

Der Algorithmus ist in eine Aufwärm- und eine Hauptphase unterteilt. In der Aufwärmphase in den Zeilen 4 - 16 stehen nicht alle Operationen aus Abschnitt 5.4.3 zur Verfügung, sondern es können nur die Adjust Middlegap Operationen der Blöcke bzw. Kreuzungen verwendet werden. Das mit dieser Maßnahme behandelte Problem tritt bei Straßen mit großer baulicher Trennung auf: Bei einer gleichberechtigten Auswahl der Operationen und einem initialen Modell ohne bauliche Trennung erzeugt das Verfahren schnell ein Modell mit vielen Fahrspuren. Es werden anschließend viele Iterationen dazu verwendet, die überflüssigen Fahrspuren gegen eine bauliche Trennung *zu tauschen*. Durch die Aufwärmphase wird eine bessere initiale Schätzung der Trennung in sehr wenigen Iterationen erreicht.

Bei der Auswertung in Zeile 9 bzw. Zeile 23 kommt erneut das Simulated Annealing Verfahren (vgl. Abschnitt 2.3) zum Einsatz. Jedem Modell $\xi \in \Phi$ wird dabei eine Abkühlfunktion $t_{SA}^{(\xi)}(n)$ mit exponentiell multiplikativem Abkühlschema, wie in Gleichung (5.4) beschrieben, zugeordnet.

Die Parameter der Abkühlfunktion werden für die Aufwärm- bzw. Hauptphase unterschiedlich initialisiert.

Algorithmus 6 : Pseudocode zum Algorithmus zur Konstruktion fahrspurgenauer Straßenkarten.

$G \leftarrow$Algorithmus 5 `// Generierung einer fahrbahngenauen Straßenkarte`
$\Phi \leftarrow$ Initialisiere Modelle auf G `// Abschnitt 5.4.2`
$t_{SA}^{(\xi)}(n) \leftarrow$ Initialisiere SA Funktionen für Aufwärmphase
while $\sum_{\xi \in \Phi} t_{SA}^{(\xi)}(n) > 0$ **do** `// Abbruchkriterium`
  for $\xi \in \Phi$ **do**
    $T \leftarrow$ Operation *Adjust Middlegap*
    $s' \leftarrow$ Erzeuge neuen Zustand gemäß T
    Ziehe $w \sim$ Unif(0,1)
    if $w < A(T)$ *mit* $\pi = \pi^{1/t_{SA}^{(\xi)}(n)}$ *(Gl. (5.3))* **then** `// A: Operationsabhängig`
      Setze $s^{(n+1)} := s'$
    else
      Setze $s^{(n+1)} := s^{(n)}$
      Kühle Funktion $t_{SA}^{(\xi)}(n)$ ab
    end
  end
end
$t_{SA}^{(\xi)}(n) \leftarrow$ Initialisiere SA Funktionen für Hauptphase
while $\sum_{\xi \in \Phi} t_{SA}^{(\xi)}(n) > 0$ **do** `// Abbruchkriterium`
  for $\xi \in \Phi$ **do**
    $T \leftarrow$ Wähle Operation/Übergangskern
    $s' \leftarrow$ Erzeuge neuen Zustand gemäß T
    Ziehe $w \sim$ Unif(0,1)
    if $w < A(T)$ *mit* $\pi = \pi^{1/t_{SA}^{(\xi)}(n)}$ *(Gl. (5.3))* **then** `// A: Operationsabhängig`
      Setze $s^{(n+1)} := s'$
    else
      Setze $s^{(n+1)} := s^{(n)}$
      Kühle Funktion $t_{SA}^{(\xi)}(n)$ ab
    end
  end
end

5.4.5. Bewertung

Für die Auswertung von Gleichung (5.9) wird einerseits ein Maß zur Bewertung der Übereinstimmung zwischen dem Modell und den Daten in π_d^l (bzw. logarithmiert Π_d^l) und andererseits Vorwissen über das Modell in π_p^l (bzw. logarithmiert Π_p^l) benötigt. Im Folgenden werden diese Maße dargelegt.

5.4.5.1. Daten

Im ersten Schritt werden die Fahrzeugtrajektorien auf die Fahrspuren abgebildet. Dazu wird zunächst jede Mittellinie eines jeden Blocks und jeder Kreuzung in einen Graphen überführt, wobei die Mittellinie kleinschrittig mit Knoten und Kanten diskretisiert wird. Jeder Graph wird anschließend mit dem *Douglas-Peucker-Algorithmus* (Douglas und Peucker, 1973) auf eine minimale Anzahl von Knoten optimiert. Abschließend werden diese Graphen zu einer Gesamtrepräsentation G fusioniert.

Im nächsten Schritt wird für jede Trajektorie ein Hidden-Markov-Modell (HMM) erzeugt, welches als latente Zustände die Kanten des generierten Graphen G und als emittierte Beobachtungen die Trajektorienpunkte hat. Mit Hilfe des Algorithmus von Viterbi (vgl. Abschnitt 2.2) wird das Optimum gesucht und somit die wahrscheinlichste Zuordnung eines jeden Messpunktes zu einer Fahrspur bestimmt.

Analog zur Bewertung der fahrbahngenauen Konstruktion in Abschnitt 5.3.4 werden als Fehlermaße der euklidische Abstand Υ_d zwischen Trajektorie und Fahrspur in Gleichung (5.7), sowie der eingeschlossene Fahrtwinkel Υ_α in Gleichung (5.8) ausgewertet. Zusätzlich wird ein Grenzwert für die minimale Anzahl an Durchfahrten eingeführt. Υ_f beschreibt im Rahmen dieser Arbeit eine Sprungfunktion, welche bewirkt, dass Fahrspuren mit zu wenigen Durchfahrten als Ausreißer eingestuft und die Konfiguration abgelehnt wird.

$$\Pi_d^l = \sum_{t\in T}\sum_{n\in m}\Big[\log\Big(\Upsilon_d\Big(G(t_n),t_n\Big)\Big) + \log\Big(\Upsilon_\alpha\Big(\gamma(G(t_n)),\gamma(t_n)\Big)\Big)\Big] + \sum_{\xi\in\Phi}\sum_{x\in L(\xi)}\log\Big(\Upsilon_f\Big(x,T\Big)\Big) \tag{5.10}$$

$$G(t_n) = \arg\max_{\bar{e}\in G} p(\bar{e}|t_n) = e_n$$

wobei $L(\xi)$ die Fahrspuren des Models ξ beschreibt. Die Sprungfunktion zur Berücksichtigung der Durchfahrten ist dabei definiert als:

$$\Upsilon_f(x,T) = \begin{cases} 1, & \text{falls } \varepsilon \geq \eta \\ 0, & \text{sonst} \end{cases}$$

wobei ε die Anzahl an Trajektorien, die auf die x-te Fahrspur des Modells abgebildet wurden und η die minimal zu erreichende Anzahl an Durchfahrten beziffert. Die Sprungfunktion kann bei einem ausreichend großem Datensatz durch eine Sigmoidfunktion ersetzt werden was im Rahmen dieser Arbeit allerdings nicht sinnvoll ist.

5.4.5.2. A-priori Informationen

In Π_p^l wird Vorwissen über die fahrspurgenauen Modelle berücksichtigt. Um die Blöcke und Kreuzungen realistisch zu halten werden Regularisierungsterme eingeführt, welche die gewissen Ausprägungen der Modelleigenschaften bestärken oder abschwächen sollen. Berücksichtigt werden dabei die Breite einer Fahrspur und die Länge eines Blocks:

$$\Pi_p^l = \Pi_{p,\text{lane width}}^l + \Pi_{p,\text{block size}}^l$$

Bezüglich der Spurbreite w wird eine Normalverteilung angenommen, dessen Parameter im Kontext des Szenario zu wählen sind (vgl. Abschnitt 6.5.1). Diese Parameter werden dabei aus dem Parametersatz der RJMCMC Add Lane Operation wiederverwendet.

$$\Pi_{p,\text{lane width}}^l = \log\Big(\frac{1}{\sqrt{2\pi\sigma_{lw}^2}} \cdot \exp^{-\frac{(w-\mu_{lw})^2}{\sigma_{lw}^2}}\Big)$$

$$\mu_{lw} = \mu_{\text{add lane}}$$

$$\sigma_{lw} = \sigma_{\text{add lane}}$$

Overfitting würde in diesem Verfahren durch eine Aufreihung sehr kurzer Blöcke entstehen. Um dem entgegen zu wirken wird eine Minimallänge κ für einen Block eingeführt, welche durch die Regulierung mit Hilfe einer Sprungfunktion realisiert wird:

$$\Pi_{p,\text{block size}}^l = \begin{cases} 0, & \text{falls Block Länge } > \kappa \\ -\infty, & \text{sonst} \end{cases}$$

6. Ergebnisse

In diesem Kapitel werden mit dem vorgestellten neuen Algorithmus zur fahrbahn- bzw. fahrspurgenauen Rekonstruktion digitaler Straßenkarten Ergebnisse erzeugt und evaluiert. Zunächst werden in Abschnitt 6.1 die verwendeten Eingabedaten und in Abschnitt 6.2 die zur Bewertung benötigte Vergleichskarte (Referenz) dargelegt. Um die konstruierte Karte mit der Referenzkarte zu vergleichen, werden in Abschnitt 6.3 verschiedene Bewertungsmöglichkeiten präsentiert. Zur Erzeugung der Ergebnisse wird zunächst für die fahrbahngenaue Rekonstruktion eine Parameterstudie in Abschnitt 6.4.1 durchgeführt, um die Parameter des Algorithmus auf die Problemstellung (nicht auf die Daten) einzustellen. Anschließend werden die Bewertungen zur Auswertung der Ergebnisse in Abschnitt 6.4.2 herangezogen. Das gleiche Vorgehen wird bezüglich der fahrspurgenauen Rekonstruktion angewandt, in Abschnitt 6.5.1 werden die Parameter eingestellt und in Abschnitt 6.5.2 die erzeugten Ergebnisse evaluiert.

6.1. Eingabedatensätze

Zur Erzeugung des benötigten Datensatzes wird die *Fahrzeugflotte* im Rahmen dieser Arbeit durch ein einzelnes Fahrzeug repräsentiert. Im Folgenden werden zunächst die verwendete Hard- sowie Software und anschließend die erhobenen Daten dargelegt.

Bei dem Testfahrzeug handelt es sich um einen VW Passat Variant 2.0 TDI. Das Fahrzeug ist ausgestattet mit dem POS LV 220 System der 5. Generation von Applanix[a]. Dabei handelt es sich um ein Gesamtsystem zur hochgenauen Positionierung, welches in der Lage ist diverse Fahrzeugsensoren auszuwerten und zur Positionsbestimmung zu vereinen. Zur grundlegenden Ortung wird ein GNSS Empfänger (Applanix GPS-17 Komponente) verwendet. Um typische Drift- und Sprungfehler zu verringern, werden zwei derartige Empfänger mit jeweils einer Trimble 540 AP Antenne benutzt und als ein *GNSS Azimuth Measurement Subsystem* (GAMS) verwendet. Dies bedeutet, dass über die bekannte relative und die per Satellit gemessene Position der Antennen die Schätzung der Fahrtrichtung stark verbessert werden kann, um einen glatten und realistischen Verlauf der Messung zu gewährleisten. Zur Verbesserung der Daten ist zusätzlich eine *Inertial Measurement Unit* (IMU) (Applanix IMU-42 Komponente) mit drei Beschleunigungssensoren und drei Gyroskopen zur Überwachung der Beschleunigung und Drehung des Fahrzeugs verbaut. Des Weiteren ist ein *Distance Measurement Indicator* (DMI) in Form eines externen Drehratensensors am Rad des Fahrzeuges angebracht, welcher die zurückgelegte Strecke misst, um die Drift des GNSS zu verringern und somit die Positionierungsgenauigkeit zu erhöhen. Die Daten aller Komponenten können mit einer Frequenz von bis zu 200 Hz aufgezeichnet werden. Ebenfalls Teil des Systems ist die *PosPac MMS*

[a] https : //www.applanix.com/pdf/poslv_brochure.pdf

Software, womit die Daten der beschriebenen Komponenten analysiert und diese oder weitere bei Bedarf nachträglich vereint werden können. Im Rahmen dieser Arbeit werden zusätzlich Daten des *Satellitenpositionierungsdienst*[b] (SAPOS) zur hochgenauen Positionierung verwendet. Dabei wird ein vorhandenes Netz geostationärer Referenzstationen genutzt, um über die bekannte und die bestimmte eigene Position atmosphärische und anderweitige Einflüsse zu messen und daraus lokal gültige Korrekturdaten zu berechnen. Mithilfe dieser Daten kann die Positionierungsgenauigkeit des GNSS System erheblich verbessert werden.

Zusätzlich ist das Fahrzeug mit einer *Bosch Multi Purpose Camera* (MPC2) in der Frontscheibe ausgestattet. Die Kamera verfügt über eine Bildfrequenz von 30 Hz, bei einem horizontalen Öffnungswinkel von 50 ° und einem vertikalen von 28 °. Die integrierte Software ist in der Lage, sich bewegende Objekte zu erkennen und dessen Spurverlauf relativ zur eigenen Position aufzuzeichnen. Somit ist es möglich, andere Verkehrsteilnehmer zu identifizieren und deren Trajektorie über die Position des aufzeichnenden Fahrzeugs zu bestimmen. Dabei hat die Kamera eine Arbeitsreichweite von 130 m und somit kann zusätzlich zu den Ego-Trajektorien eine Vielzahl von *Objekt-Trajektorien* generiert werden.

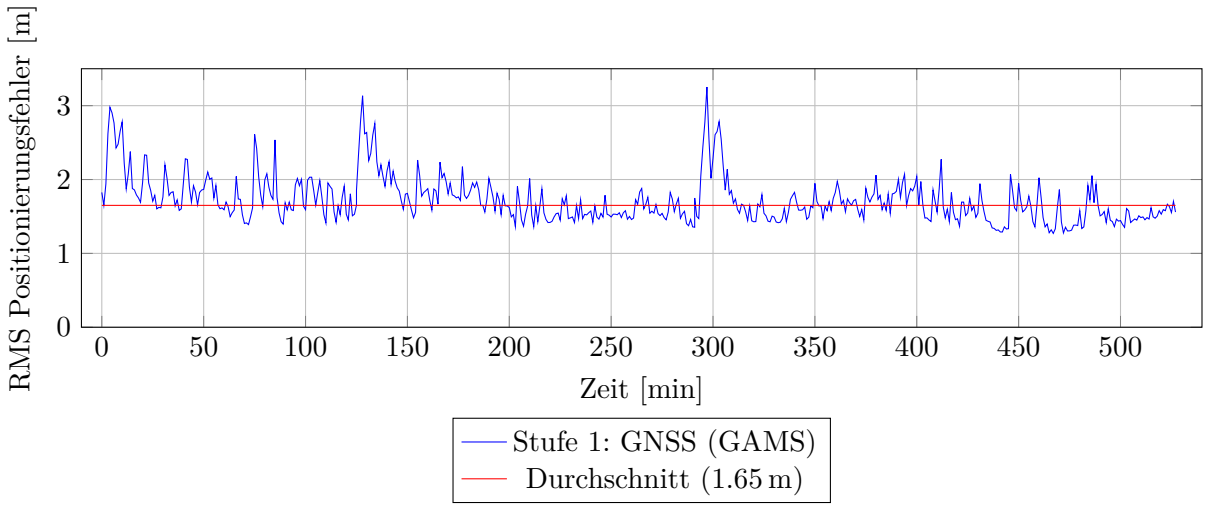


Abbildung 6.1.: Entwicklung des durchschnittlichen RMS Positionierungsfehlers des Applanix Systems bezüglich der Daten der ersten Qualitätsstufe über der Laufzeit des gesamten Experiments.

Das eingefahrene Szenario befindet sich auf dem Stadtgebiet von Hildesheim in Niedersachsen und erstreckt sich über eine Fläche von ca. 2.8 km^2. Die Durchfahrten sind so gewählt, dass sie einerseits die Natur einer zufällig durchfahrenden Fahrzeugflotte erhalten und andererseits jede Fahrspur der Straßen und jede topologische Verknüpfung der Kreuzungen mindestens einmal befahren wurde. Dabei wurden die Daten der oben beschriebenen Hardware Komponenten separat aufgezeichnet, um die Möglichkeit zu erhalten, den Datensatz in verschiedenen Genauigkeitsstufen auszuwerten. Die in der *ersten Stufe* enthaltenen Daten sind dabei nur mit dem GNSS/GAMS System aufgezeichnet und stellen somit den ungenausten Datensatz dar. Die Entwicklung des vom Applanix Systems ausgegebenen Root-Mean-Square (RMS) Positionierungsfehlers ist in Abbildung 6.1 über der Zeit

[b] http://www.sapos.de/

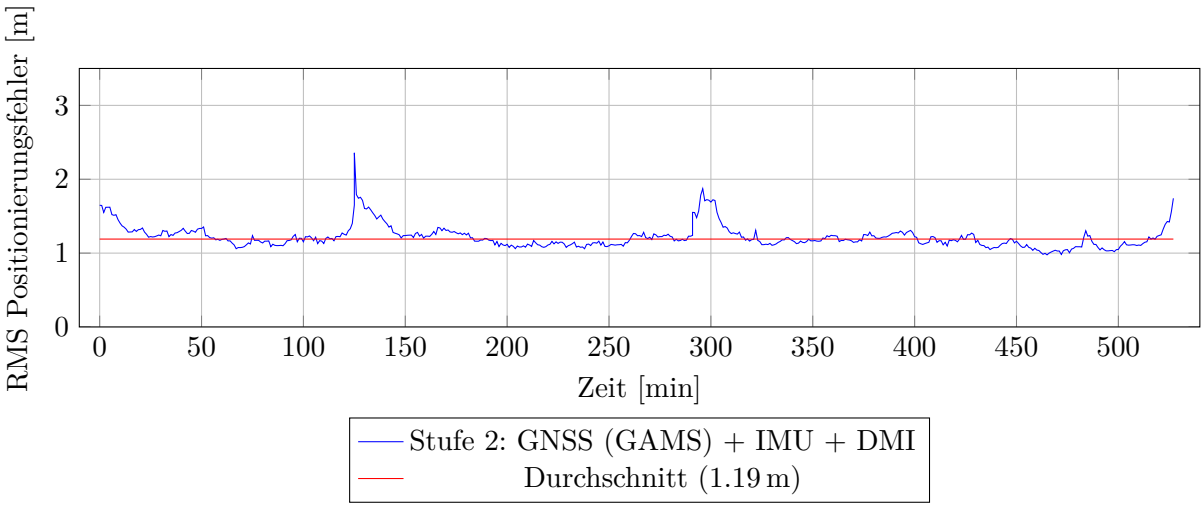


Abbildung 6.2.: Entwicklung des durchschnittlichen RMS Positionierungsfehlers des Applanix Systems bezüglich der Daten der zweiten Qualitätsstufe über der Laufzeit des gesamten Experiments.

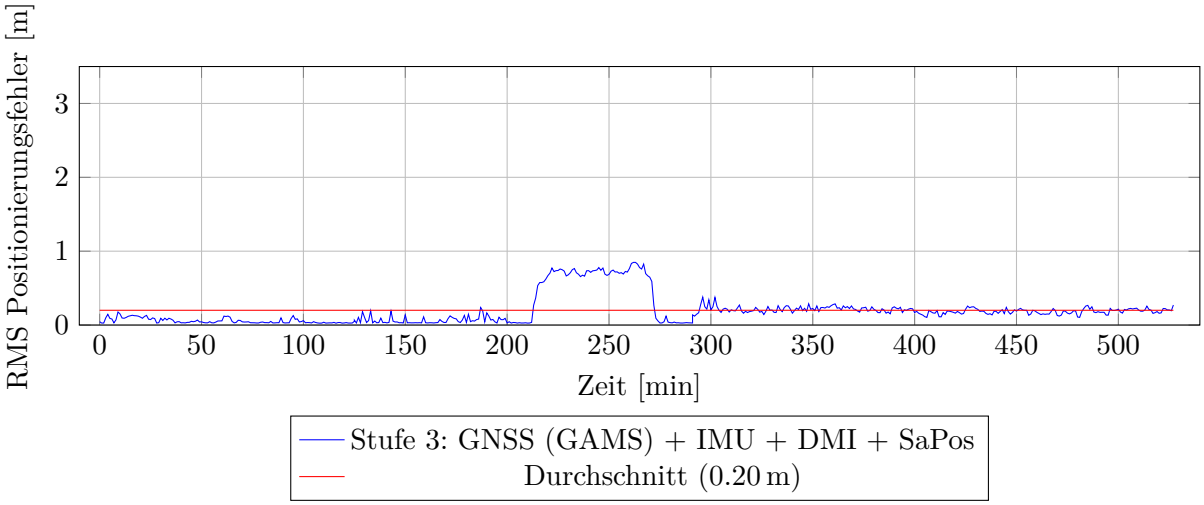


Abbildung 6.3.: Entwicklung des durchschnittlichen RMS Positionierungsfehlers des Applanix Systems bezüglich der Daten der dritten Qualitätsstufe über der Laufzeit des gesamten Experiments.

und zusätzlich der Durchschnitt (rot) dargestellt. Des Weiteren befindet sich im Anhang C.2 Abbildung C.1 eine räumliche Darstellung der Daten, überlagert mit einem Satellitenbild und eingefärbt nach dem Positionierungsfehler. Zur Erzeugung der *zweiten Stufe* werden diese Daten mit der IMU und dem DMI vereint, um den Positionierungsfehler zu senken. Die Entwicklung über die Zeit ist in Abbildung 6.2 und die räumliche Visualisierung in Anhang C.2 Abbildung C.2 abgebildet. In den Entwicklungen der ersten beiden Stufen sind jeweils bei ca. Minute 130 und Minute 300 Anomalien erkennbar. Die dargestellte Messung erfolgte über drei Tage hinweg und zu den beschriebenen Zeitpunkten beginnt ein neuer Tag mit einer Kalibrierungsphase, abseits des eigentlichen Szenarios, was zu den erhöhten Werten führt. Abschließend beinhalten die Messungen der *dritten Stufen* die zusätzliche Vereinigung mit den SAPOS Korrekturdaten, um die bestmögliche Genauigkeit zu erhalten, wie in Abbildung 6.3 bzw. Anhang C.2 Abbildung C.3 dargestellt. Durch die Korrekturdaten konnten die Kalibrierungsphasen abseits des Szenarios ausgebessert werden.

In diesem Datensatz ist allerdings zwischen Minute 210 und 270 ein höherer Positionierungsfehler erkennbar. Diese Fehler sind auf Messausfälle in den SAPOS Daten zurückzuführen[c].

Die finalen Datensätze beinhalten pro Genauigkeitsstufe 130 Trajektorien mit einer Gesamtlänge von 136.69 km und einer Fahrtzeit von 08 : 14 : 42 (hh:mm:ss).

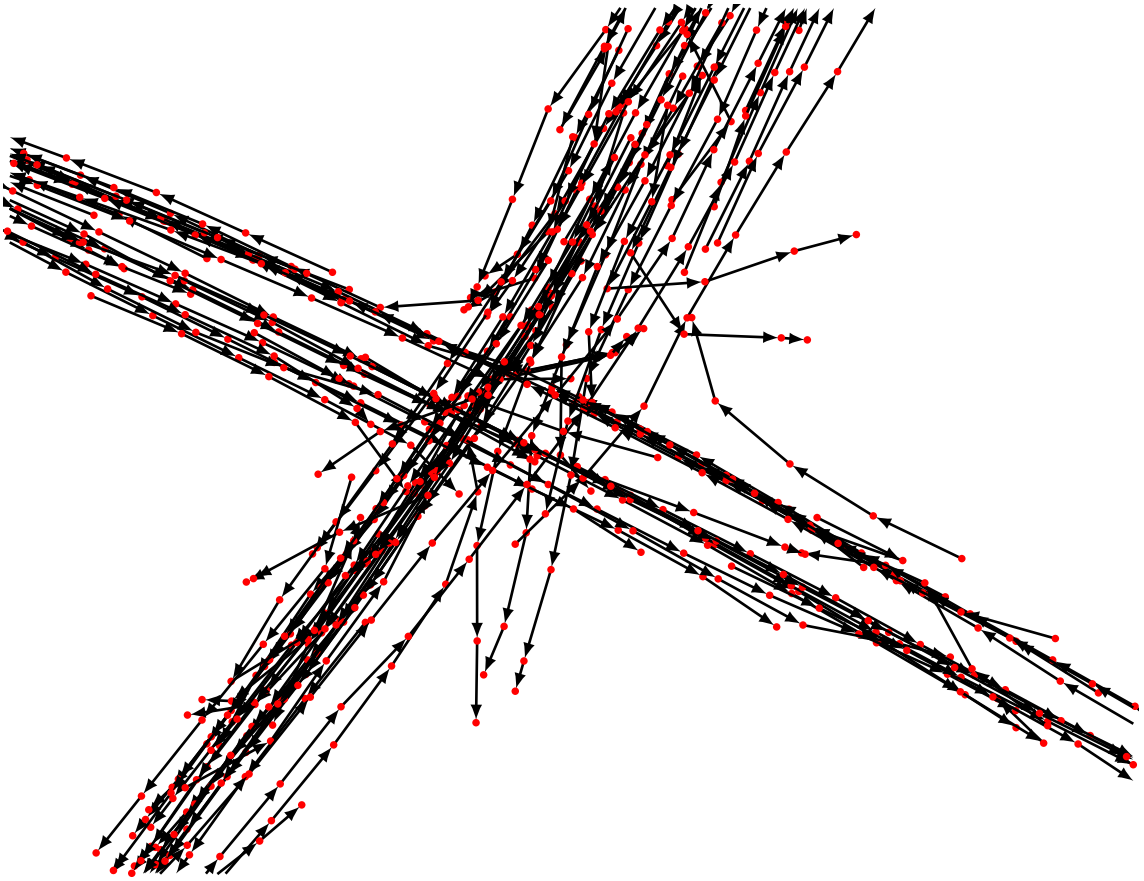


Abbildung 6.4.: Objekt-Trajektorien an einer Kreuzung basierend auf GNSS (GAMS).

Die aufgezeichneten Objekt-Trajektorien sind relativ zum Egofahrzeug vermessen, daher entsteht ein vom Ortungssystem unabhängiger Fehler durch die Kamera, welcher beim Transformieren in absolute Koordinaten auf den Positionierungsfehler des Ortungssystems in Form einer Varianzfortpflanzung fortgepflanzt wird. Durch verschiedene Einflüsse wie beispielsweise die Sonneneinstrahlung kommt es zu Fehlern und Ausreißern im Tracking und der Positionierung der beobachteten Objekte. In den aufgezeichneten Daten sind einige derartige Fehler ersichtlich. Daher stellt die Verwendung dieser Daten einerseits eine einfache Möglichkeit zur Erzeugung zusätzlicher Trajektorien und somit redundanter Informationen dar, andererseits steigen aufgrund der inhärenten starken Messfehler die Anforderungen an die Robustheit der verarbeitenden Systeme. In Abbildung 6.4 sind beispielhaft Objekt-Trajektorien auf einer Kreuzung dargestellt, wobei die Messfehler vor allem bei Abbiegungen deutlich werden. Als problematisch erweist sich hierbei die Berücksichtigung der Rotation des Egofahrzeugs durch die Kamera. Weitere Beispiele auf einer Straße und die Auswertungen basierend

[c]SAPOS Log: "Bei der Station 0663 ist es zu Datenausfällen gekommen. Bei den Daten von 17.03.2016 09:55:00 bis 17.03.2016 10:59:59 fehlen 180 Epochen."

auf den anderen Ortungssystemen sind im Anhang C.3 dargestellt. Die finalen Datensätze bestehen pro Genauigkeitsstufe aus 971 Trajektorien mit einer Gesamtlänge von 173.68 km.

Die beschriebenen Datensätze unterschiedlicher Genauigkeitsstufen und Sensoren sind in anonymisierter Form unter der CC0 1.0 Lizenz veröffentlicht und unter folgendem Link abrufbar: http://dx.doi.org/10.25835/LUIS.1

6.2. Referenz Datensatz

Um die erzeugten Ergebnisse in Form von spurgenauen digitalen Straßenkarten bewerten zu können wird in diesem Abschnitt eine Vergleichskarte beschrieben, welche in dieser Arbeit als bestmögliche Lösung angesehen wird. Die Referenzkarte beinhaltet das in Abschnitt 6.1 beschriebene Szenario. Die Karte wurde durch die Fusion von Laser-, Radar-, dGPS- und Kameradaten teilautomatisch und durch manuelle Nachbearbeitung erzeugt. Sie besteht aus insgesamt 137 Straßen mit einer Gesamtlänge von 24.4 km und 71 Kreuzungen. Als Grundlage dient eine fahrbahngenaue Straßenkarte im NDS-Standard[d]. Relativ zu dieser Repräsentation können beispielsweise Fahrspuren und längliche Fahrbahnmarkierungen als beliebig komplexe Funktionen und punktförmige Objekte wie beispielsweise Piktogramme, Ampeln und viele weitere definiert werden. Zusätzlich können topologische Informationen wie Verkehrs- und Vorfahrtregeln abgebildet werden. In Abbildung 6.5 ist beispielhaft eine Kreuzung dargestellt, auf welcher in Gelb die Spurmittellinien, in Weiß die Fahrbahnmarkierungen und Piktogramme, sowie weitere zusätzliche Objekte wie Verkehrsinseln in Rot erkennbar sind.

6.3. Bewertungen

In diesem Abschnitt werden die zur Auswertung der Ergebnisse verwendeten Bewertungskriterien eingeführt.

Die einfache *Distanz* Bewertung d_D beschreibt die durchschnittliche Abweichung zwischen einem Graphen G und den Eingabetrajektorien T. Die Bewertung trifft somit eine Aussage darüber, wie gut der Graph die Trajektorien abbildet. Dazu wird zunächst für jede Trajektorie $t \in T$ ein Hidden-Markov-Modell (HMM) erzeugt, welches als versteckte Zustände die Kanten $e \in E$ des Graphen G und als emittierte Beobachtungen die Trajektorienpunkte hat. Das HMM wird mit Hilfe des Algorithmus von Viterbi (vgl. Abschnitt 2.2) gelöst und ergibt die wahrscheinlichste Zuordnung $e(t_i)$ eines jeden Messpunktes t_i zu einer Kante e. Somit kann zu jeder Trajektorie der

[d] http://www.nds-association.org/

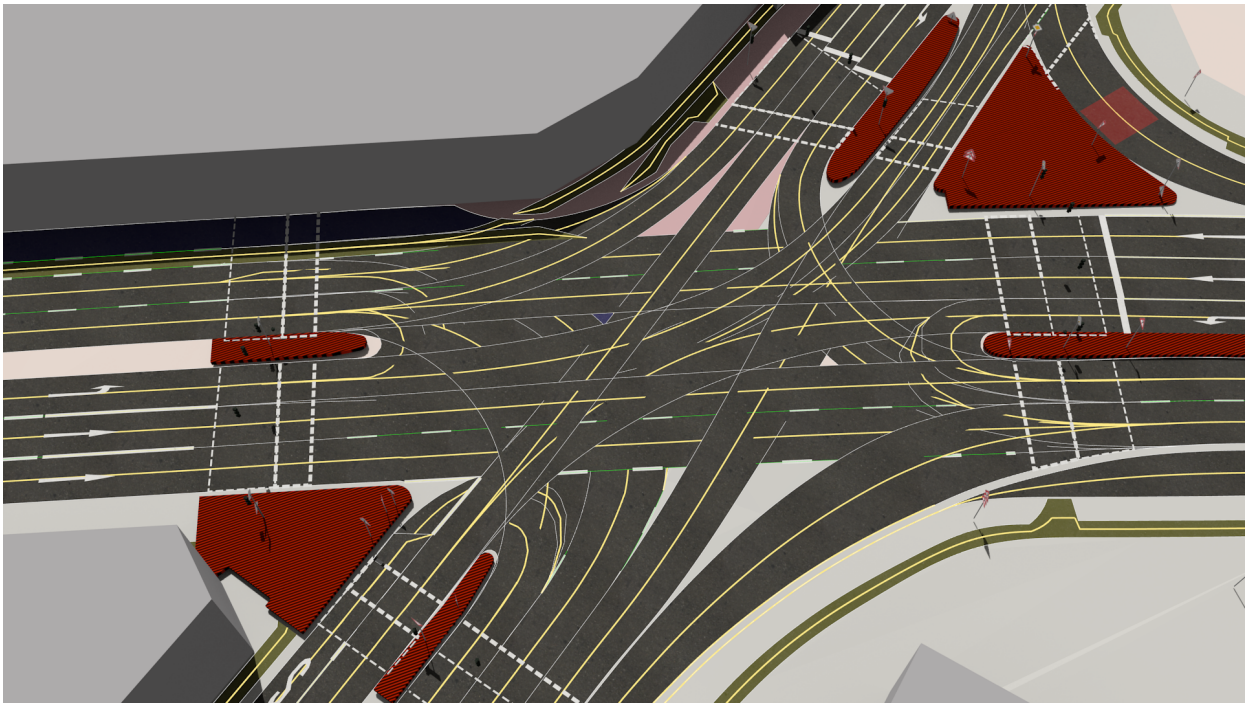


Abbildung 6.5.: Screenshot einer Visualisierung der Referenzkarte. Dargestellt sind Spurmittellinien der Straße, sowie Geh- und Radwege (gelb), Fahrbahnmarkierungen, Piktogramme, Fußgängerüberwege und Haltelinien (weiß), Verkehrsinseln (rot), Gebäude (grau). Quelle: Robert Bosch GmbH.

korrespondierende *Pfad* im Graphen bestimmt werden. Als Ergebnis dieser Bewertung wird über alle Trajektorienpunkte anschließend der kürzeste Abstand zur zugehörigen Kanten gemittelt:

$$d_D = \frac{1}{m} \sum_{t \in T} \sum_{i=1}^{|t|} \mathrm{d}(e(t_i), t_i)$$

$$e(t) = \arg\max_{\bar{e} \in G} p(\bar{e}|t)$$

wobei $\mathrm{d}(\cdot)$ der kleinste euklidische Abstand und m die Gesamtanzahl der Messpunkte aller Trajektorien ist. Eine Visualisierung der Distanzbestimmung kann Abbildung 6.6 entnommen werden.

Im Kontext der in den nächsten Abschnitten beschriebenen Auswertungen wird eine Trajektorie jeweils auf die konstruierte Straßenkarte und auf die korrespondierende Referenzkarte abgebildet. Somit stellen die beiden Pfade die wahrscheinlichsten Wege einer Trajektorie durch zwei digitale Straßenkarten (als Graph) dar.

Die *Pfadlängen-Differenz* d_{PL} beschreibt den absoluten und relativen longitudinalen Unterschied zwischen zwei Pfaden p_1 und p_2 (vgl. Abbildung 6.7) und beträgt somit im optimalen Fall 0 m bzw. 0 %. Diese Bewertung wird durch geometrische und topologische Unterschiede zwischen den Pfaden beeinflusst, ist daher ungleich Null, wenn derartige Fehler vorhanden sind. Sie erlaubt allerdings nicht den Umkehrschluss, da auch gänzlich unterschiedliche Pfade die gleiche Länge besitzen können. Daher ist bei der Verwendung sicherzustellen, dass die verglichenen Pfade die gleiche Situation beschreiben. Zur Ermittlung des absoluten Ergebnisses $d_{PL,a}$ wird der eindimensionale

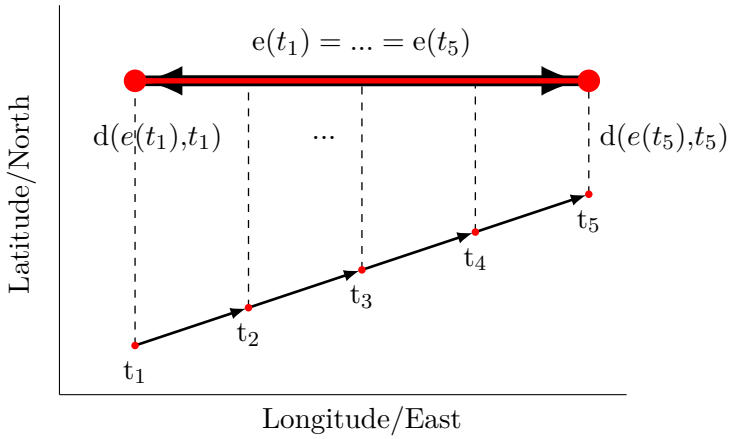
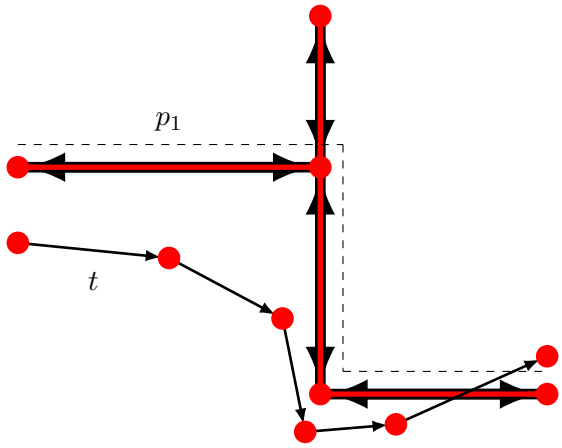
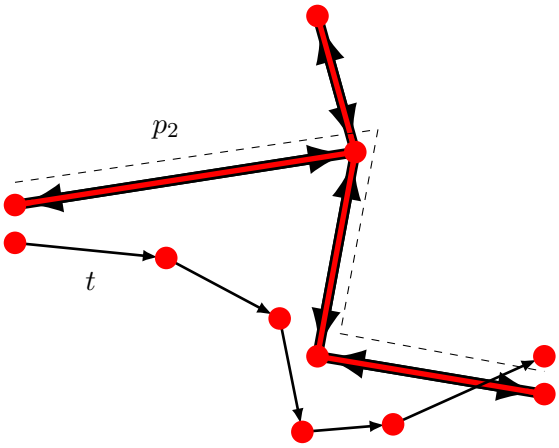


Abbildung 6.6.: Einfache Distanz Bewertung.



(a) Trajektorie t abgebildet als gestrichelter Pfad p_1 auf die Referenzkarte.



(b) Trajektorie t abgebildet als gestrichelter Pfad p_2 auf die Rekonstruktion.

Abbildung 6.7.: Die Pfadlängen-Differenz.

Längenunterschied zwischen den Pfaden berechnet. Dieser Unterschied kann zur jeweiligen Länge der beiden Pfade in Relation gesetzt werden, um ein relatives Ergebnis $d_{PL,r}$ zu erhalten. Auf diese Weise kann die Größenordnung der Pfade berücksichtigt werden, da ein *kleiner* Fehler bei einem *kurzen* Pfad in ein Verhältnis zu einem *großen* Fehler in einem *langen* Pfad gestellt werden kann.

$$d_{PL,a} = | \sum_{e \in p_1} ||e|| - \sum_{e \in p_2} ||e|| \; |$$

$$d^{(1)}_{PL,r} = \frac{d_{PL,a}}{\sum\limits_{e \in p_1} ||e||}$$

$$d^{(2)}_{PL,r} = \frac{d_{PL,a}}{\sum\limits_{e \in p_2} ||e||}$$

Die *Hausdorff*-Distanz d_{HD} basiert ebenfalls auf der Auswertung der auf die Graphen abgebildeten Pfade. Diese Bewertung beschreibt allerdings die absolute Abweichung in allen Dimensionen zwischen den Wegen. Die *Hausdorff-Distanz* (Nadler, 1978) beschreibt den Abstand zwischen zwei Mengen

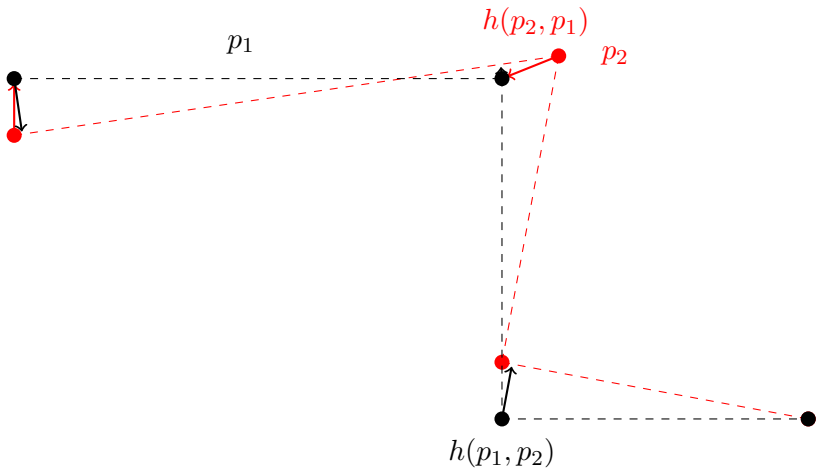


Abbildung 6.8.: Bestimmung der Hausdorff-Distanz bezüglich der in Abbildung 6.7 dargestellten Pfade.

A und B. Je kleiner der Wert, desto besser ist die gegenseitige Überdeckung der Mengen. Die Bewertung ist definiert als:

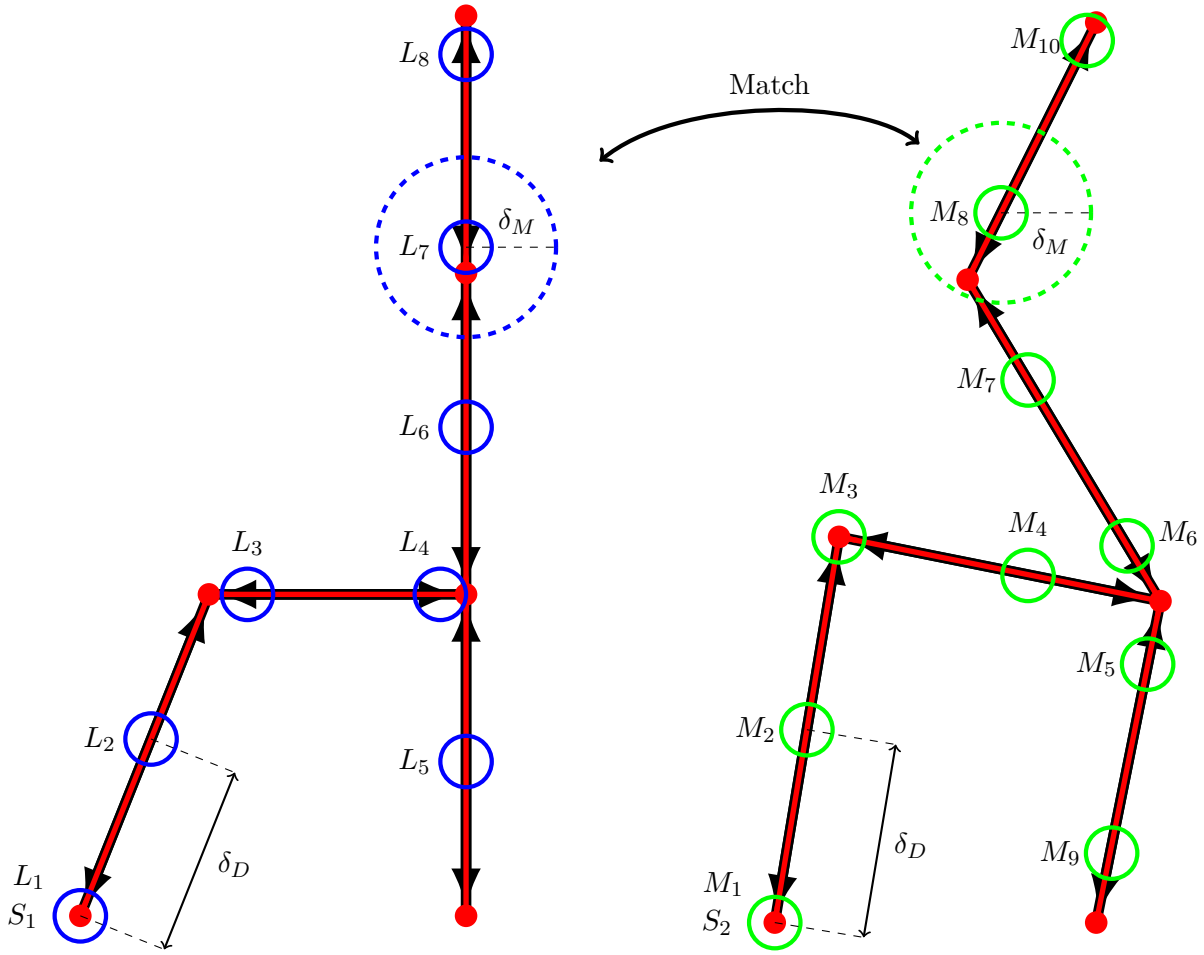
$$H(A,B) = \max(h(A,B),h(B,A))$$
$$h(A,B) = \max_{a \in A} \min_{b \in B} ||a - b||$$

wobei $||\cdot||$ die euklidische Norm ist. Die Hausdorff-Distanz, paarweise angewandt auf die beiden Pfade beschreibt somit deren gegenseitige Überdeckung, welche im optimalen Fall Null ist.

$$d_{HD} = H(p_1,p_2) = \max(h(p_1,p_2),h(p_2,p_1))$$

Die Bestimmung der Hausdorff-Distanz ist beispielhaft in Abbildung 6.8 dargestellt.

Der *F-Score* ist ein allgemeines Maß zur Beurteilung eines Klassifikators. Ein Klassifikator ordnet eine Menge von Objekten verschiedenen Klassen zu. Dabei ist die Zuordnung nicht immer eindeutig, wodurch es zu fehlerhaften Ergebnissen kommen kann. Die relativen Häufigkeiten dieser Fehler werden verwendet, um den Klassifikator zu beurteilen. In diesem Zusammenhang wird der F-Score verwendet, um die geometrische und topologische Ähnlichkeit zwischen zwei Graphen G_1 und G_2 zu beschreiben. Der *Klassifikator* ist in diesem Fall eine durch die Struktur vorgegebene Zuordnung von Punkten zu einem der beiden Graphen. Zur Bewertung werden Teile der zu vergleichenden Graphen in eine Baumstruktur (vgl. Abschnitt 2.1.1) überführt und verglichen. Die Erzeugung der Struktur erfolgt auf beiden Graphen parallel. Zunächst wird aus G_1 ein zufälliger Startknoten S_1 ausgewählt und der entsprechende Punkt S_2 aus G_2 mit kleinstem Abstand gesucht. Anschließend werden auf beiden Graphen Baumstrukturen erzeugt, indem iterativ die nächsten Nachbarn des betrachteten Knoten in den Baum mit entsprechender Distanzbewertung eingefügt werden. Die Überführung wird abgebrochen, sobald ein Distanzwert δ_F zum Startknoten erreicht ist. Anschließend wird jede Baumstruktur von der Wurzel abwärts durchlaufen und in longitudinalen Abständen von δ_D auf G_1 „Löcher“ und auf G_2 „Murmeln“ platziert. Abschließend *rollt* jede „Murmel“ in das nächste „Loch“, wobei ein „Loch“ maximal δ_M weit entfernt und nur einmal belegt sein darf. Zur abschließenden



(a) Verteilung der „Löcher“ (blau) über die Referenzkarte.

(b) Verteilung der „Murmeln“ (grün) über die Rekonstruktion.

Abbildung 6.9.: Die F-Score Bewertung.

Bewertung werden die relativen Häufigkeiten zweier Fehlermaße definiert. Die *Precision* P bestimmt das Verhältnis zwischen den in einem „Loch“ befindlichen „Murmeln“ und der gesamten Anzahl. Der *Recall* R bestimmt das Verhältnis zwischen gefüllten „Löchern“ und der gesamten Anzahl an „Löchern“. Der F-Score ist das harmonische Mittel dieser Größen:

$$P = \frac{\text{Murmeln in Löchern}}{\text{Murmeln Gesamt}}$$
$$R = \frac{\text{Löcher mit Murmel}}{\text{Löcher Gesamt}}$$
$$d_F = 2 \cdot \frac{P \cdot R}{P + R}$$

Dieser Prozess wird mehrmals mit unterschiedlichen Startknoten durchlaufen und für die finale Auswertung wird der Mittelwert über alle Durchläufe gebildet.

In Abbildung 6.9 ist beispielhaft eine Bestimmung des F-Score verdeutlicht. In Teil 6.9a ist ein Ausschnitt aus einer Referenzkarte, sowie die Verteilung der „Löcher“ in Blau dargestellt. 6.9b zeigt analog die Rekonstruktion des Straßengraphen, sowie die Verteilung der „Murmeln“ darüber.

Ausgangspunkt für die Verteilung sind die Knoten S_1 bzw. S_2. In Tabelle 6.1 ist die resultierende Zuordnung zwischen „Murmeln“ und „Löchern“ und in Tabelle 6.2 die entsprechenden Größen zur Bestimmung des F-Score dargestellt.

Loch	Murmel
L_1	M_1
L_2	M_2
L_3	M_3
L_4	M_6
L_5	M_9
L_6	M_7
L_7	M_8
L_8	M_{10}
–	M_4
–	M_5

Tabelle 6.1.: Zuordnung zwischen „Löchern“ und „Murmeln“.

Maß	Wert
Precision P	80 %
Recall R	100 %
F-Score F	$88.\overline{8}$ %

Tabelle 6.2.: Auswertung des F-Score.

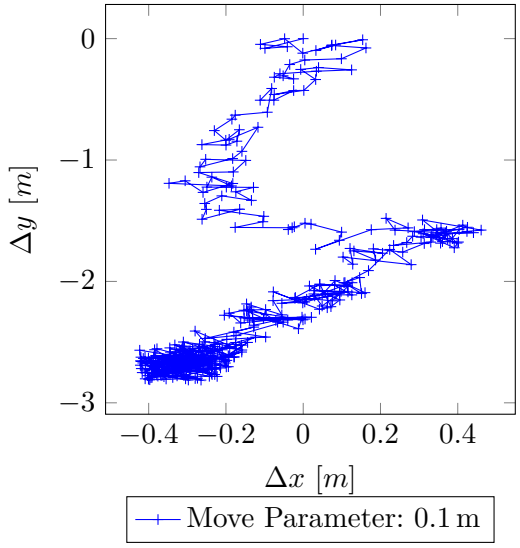
6.4. Fahrbahngenaue Rekonstruktion

In diesem Abschnitt wird das in Abschnitt 5.3 eingeführte Verfahren zur automatischen Generierung fahrbahngenauer Straßenkarten auf die im Abschnitt 6.1 dargestellten Eingabedaten angewandt. Um das bestmögliche Ergebnis zu erhalten, werden zunächst mehrere Parameterstudien durchgeführt. Mit dem ermittelten Parametersatz wird anschließend für jeden Datensatz eine fahrbahngenaue Straßenkarte erzeugt und diese anhand der in Abschnitt 6.3 definierten Bewertungen mit den Ergebnissen der in Abschnitt 4.2 beschriebenen Verfahren verglichen. Als Vergleichslösung für die Versuche wird die Referenzkarte aus Abschnitt 6.2 verwendet. Zusätzlich werden verschiedene Experimente bezüglich Typ und Anzahl der Eingabedaten durchgeführt.

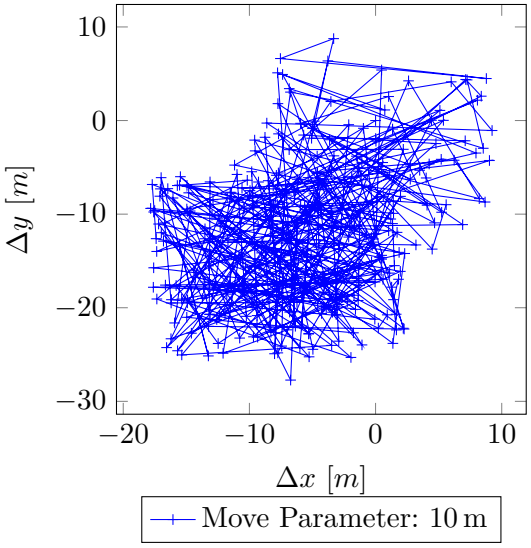
6.4.1. Analyse des Verfahrens

Das Verfahren zur Rekonstruktion fahrbahngenauer Straßenkarten wird nach der Beschreibung in Abschnitt 5.3 von drei Parametern maßgeblich beeinflusst. Zum einen beschreiben die Parameter σ_{move} und σ_{birth}, in welchem Maße sich die Eigenschaften des beschriebenen Modells verändern. Zum anderen wird das Verfahren durch den Laufzeitparameter des Simulated Annealing beeinflusst. In diesem Abschnitt werden verschiedene Studien durchgeführt, mit dem Ziel, die optimalen Werte dieser Parameter für die *allgemeine* Rekonstruktion einer Straßenkarte zu bestimmen. Um in der Zielfunktion (5.2) keinen der Anteile zu bevorzugen, wird der Verhältnisparameter zu $\beta = 0.5$ gesetzt. Für die Studien werden die Eingabedaten der dritten Stufe verwendet, um den Einfluss topologischer und geometrischer Fehler möglichst gering zu halten. Herauszustellen ist dabei, dass als Ergebnis für jeden Parameter eine begründete Dimension ermittelt wird, welche unabhängig

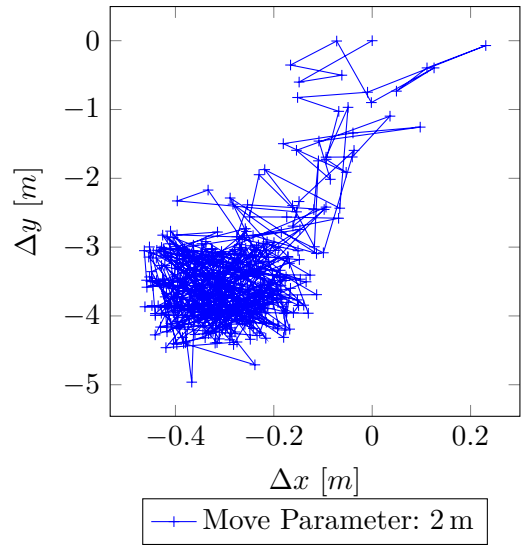
von der Qualität der Eingabedaten verwendet werden kann und somit nicht auf einen Datensatz zugeschnitten ist.



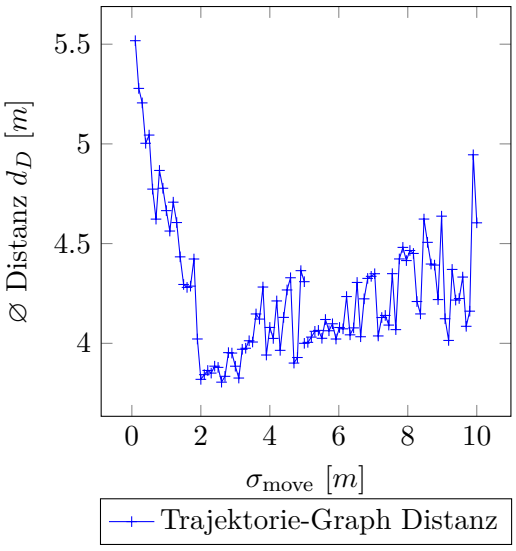
(a) Bewegung eines Knotens mit $\sigma_{\text{move}} = 0.1\,\text{m}$.



(b) Bewegung eines Knotens mit $\sigma_{\text{move}} = 10\,\text{m}$.



(c) Bewegung eines Knotens mit $\sigma_{\text{move}} = 2\,\text{m}$.

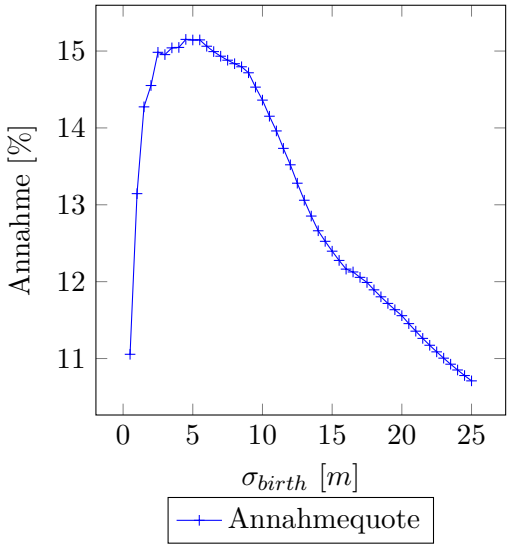


(d) Mehrfache Auswertung des Algorithmus auf dem Parameterbereich $\sigma_{\text{move}} \in [0.1,\, 5.0]\,\text{m}$.

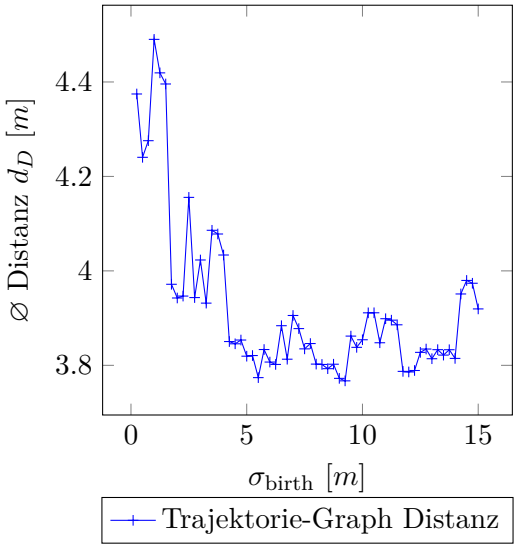
Abbildung 6.10.: Parameterstudie von σ_{move}.

Um die Einflüsse der Parameter und deren optimale Werte zu ermitteln, werden im Folgenden zunächst verschiedene Sonderfälle betrachtet. Als erstes wird der Parameter σ_{move} untersucht. Der Wert dieser Variablen beschreibt die Schrittweite in Metern, um welche ein Knoten des Straßengraphen bei der Operation „Move“ verschoben wird. Um das Konvergenzverhalten zu analysieren, werden im Algorithmus die Auswahlwahrscheinlichkeiten der Operationen „Birth“ und „Death“ zu Null gesetzt, da der entsprechende Parameter noch nicht optimiert ist. Zusätzlich wird eine Laufzeit des Simulated Annealing in Gleichung (5.5) von $s = 150$ angenommen, da ein optimaler Wert noch nicht ermittelt wurde und so eine Konvergenz des RJMCMC Verfahrens

durch die Überschätzung des Parameters wahrscheinlich wird. Zunächst werden die Extremfälle des Parameters betrachtet. In Abbildung 6.10a ist die zweidimensionale Bewegung eines Knotens mit einem *zu klein* gewählten Parameterwert abgebildet. Es ist ersichtlich, dass das System sehr langsam konvergiert, da der Knoten in vielen Iterationen in die entsprechende Richtung bewegt wird. Wird zusätzlich die Laufzeit des Verfahrens begrenzt, kann unter Umständen das Optimum nicht erreicht werden. Wird der Wert, wie in Abbildung 6.10b dargestellt, zu groß gewählt, konvergiert das System unter Umständen nicht, da das Optimum durch die große Schrittweite selten getroffen werden kann. Um einen passenden Wert zu bestimmen, wird der Algorithmus mit dem Parameterbereich $\sigma_{\text{move}} = [0.1,\, 10.0]\,\text{m}$ mehrfach durchlaufen. Zur Evaluierung der Ergebnisse, wird die eingeführte einfache Distanz Bewertung d_D verwendet. Die Entwicklung ist in Abbildung 6.10d dargestellt und beschreibt eine Verbesserung bis zu $\sigma_{\text{move}} = 2\,\text{m}$. Die entsprechende Bewegung eines Knotens ist in Abbildung 6.10c dargestellt. Es ist ersichtlich, dass das System schnell konvergiert und anschließend nicht weit streut.



(a) Annahmequote der Birth Operationen auf dem Parameterbereich $\sigma_{\text{birth}} \in [0.1,\, 25.0]\,\text{m}$.



(b) Mehrfache Auswertung des Algorithmus auf dem Parameterbereich $\sigma_{\text{birth}} \in [0.1,\, 15.0]\,\text{m}$.

Abbildung 6.11.: Parameterstudie von σ_{birth}.

Als nächstes wird der Parameter σ_{birth} auf die gleiche Weise untersucht, wobei der ermittelte Wert für $\sigma_{\text{move}} = 2\,\text{m}$ verwendetet wird. Ein neuer Knoten wird im arithmetischen Mittel zweier benachbarter Knoten erzeugt und um einen bestimmten Raumvektor verschoben, dessen Betrag und Richtung durch diesen Parameter beeinflusst wird. Zunächst werden die Extremfälle betrachtet. Wird der Parameter zu klein gewählt, wird der neue Knoten nah am berechneten Mittelwert erzeugt, was in der Bewertungsfunktion der a-priori Informationen in Abschnitt 5.3.4 bestraft wird, da dieser Knoten keine Änderungen beschreibt und somit nicht benötigt werden würde. Wird der Parameter zu groß gewählt, wird der neue Knoten häufig weit abseits des bisherigen Straßenverlaufes erzeugt und beschreibt meistens nicht mehr die Daten, was in der Bewertungsfunktion der Daten bestraft wird. Um einen angemessenen Wert zu bestimmen, wird der Algorithmus auf dem Parameterbereich $\sigma_{\text{birth}} \in [0.1,\, 25.0]\,\text{m}$ mehrfach durchlaufen. In Abbildung 6.11a ist die prozentuale Annahmequote

der vorgeschlagenen Birth Operationen über dem Parameterwert dargestellt. Es wird ersichtlich, dass die größte Annahmequote und somit die sinnvollsten Vorschläge im Bereich $\sigma_{\text{birth}} \in [4.0, 6.0]\,\text{m}$ erreicht wird. Zusätzlich ist in Abbildung 6.11b die Auswertung der einfachen Distanz Bewertung d_D über dem Parameterwert aufgetragen. Dabei tritt eine Verbesserung bis zum Wert $\sigma_{\text{birth}} = 5.5\,\text{m}$ ein, welcher als optimierter Wert angenommen wird.

Um abschließend den optimalen Wert für das Simulated Annealing zu bestimmen, werden mit den bisher bestimmten Parametern mehrere Durchläufe ausgewertet. Dabei wird λ gemäß Gleichung (5.5) in Abhängigkeit der Schrittanzahl $s \in [1{,}200]$ bestimmt. Zur Auswertung wird ebenfalls die einfache Distanz Bewertung d_D eingesetzt und das Ergebnis in Abbildung 6.12 dargestellt. Es wird ersichtlich, dass eine Verbesserung bis $s = 100$, respektive $\lambda = 0.92$, eintritt.

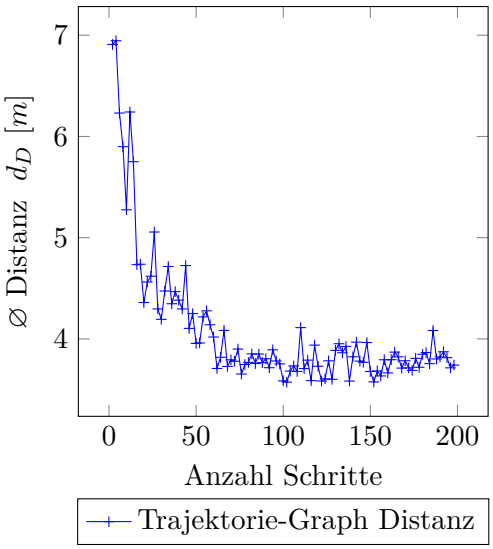


Abbildung 6.12.: Parameterstudie des Simulated Annealing.

6.4.2. Bewertung der Ergebnisse

Mit den ermittelten (vgl. Tabelle 6.3) und gewählten (vgl. Tabelle C.3) Parametern wird eine fahrbahngenaue Straßenkarte auf Grundlage der verschiedenen Eingabedaten aus Abschnitt 6.1 erzeugt. Basierend auf den identischen Daten werden entsprechende Karten unter Verwendung

Parameter	Wert
σ_{move}	2 m
σ_{birth}	5.5 m
s_{SA}	100

Tabelle 6.3.: Parameter des Algorithmus zur Rekonstruktion fahrbahngenauer Straßenkarten.

des Algorithmus nach Biagioni und Eriksson, sowie nach Cao und Krumm (vgl. Abschnitt 4.2) generiert. Um die geometrische und topologische Qualität der Straßenkarten zu beurteilen, werden in diesem Abschnitt die in Abschnitt 6.3 beschriebenen Bewertungen zum Vergleich zweier Graphen herangezogen, um die Ergebnisse mit der Referenzkarte aus Abschnitt 6.2 zu vergleichen. Als

geometrischer Fehler wird dabei ein von der Vergleichskarte abweichender geographischer Verlauf angesehen. Ein topologischer Fehler hingegen beschreibt das Fehlen von Verbindungen im Straßengraphen. Im Folgenden werden zunächst die Ergebnisse basierend auf den Ego-Trajektorien der verschiedenen Qualitätsstufen ausgewertet. Anschließend werden Experimente bezüglich des Einflusses der Objekt-Trajektorien betrachtet.

Ego-Trajektorien

Als erstes wird die Pfadlängen-Differenz d_{PL} ausgewertet. Nachdem alle Trajektorien auf die entsprechenden Graphen abgebildet wurden, können die absoluten und relativen Unterschiede zwischen den korrespondierenden Wegen ausgewertet werden. In Abbildung 6.13 bzw. Abbildung 6.14 sind die Ergebnisse als Boxplots[e] dargestellt. Die Abbildungen zeigen die Auswertungen der Bewertungen, angewandt auf die Ergebnisse der verschiedenen Algorithmen, basierend auf den drei unterschiedlichen Qualitätsstufen der Eingabedaten. Die Darstellung des Fehlers in absoluter und relativer Form erzeugt dabei ein besseres Verständnis einiger Werte, weil ein großer absoluter Fehler bezüglich einer *kurzen* Trajektorie gravierend, im Bezug auf eine *lange* Trajektorie allerdings weniger schwerwiegend ist.

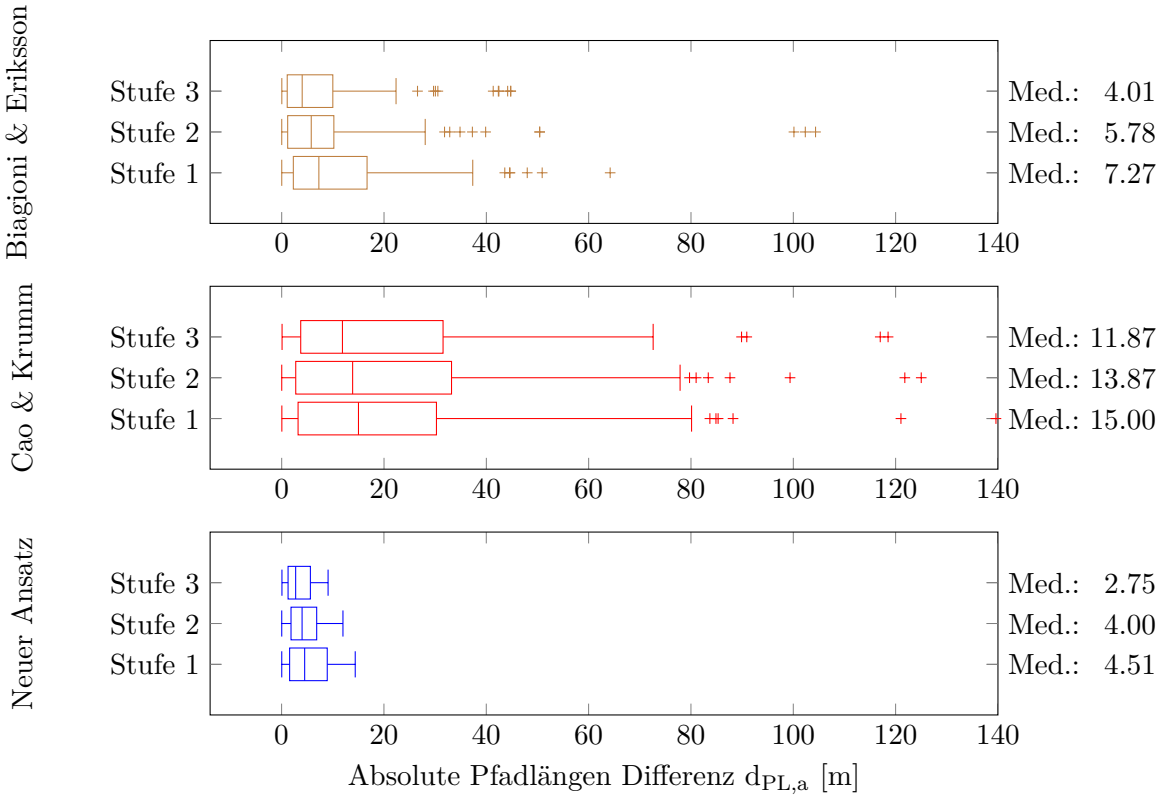


Abbildung 6.13.: Absolute Längenunterschiede zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen. Die rechte Beschriftung beschreibt den Wert des jeweiligen Medians.

[e]Datenreferenz: Median: 50 %, untere Box: 25 %, obere Box: 75 %, Interquartilsabstand (IQR): Länge der Box als Wert, Abstand Whisker: maximal 1.5·IQR

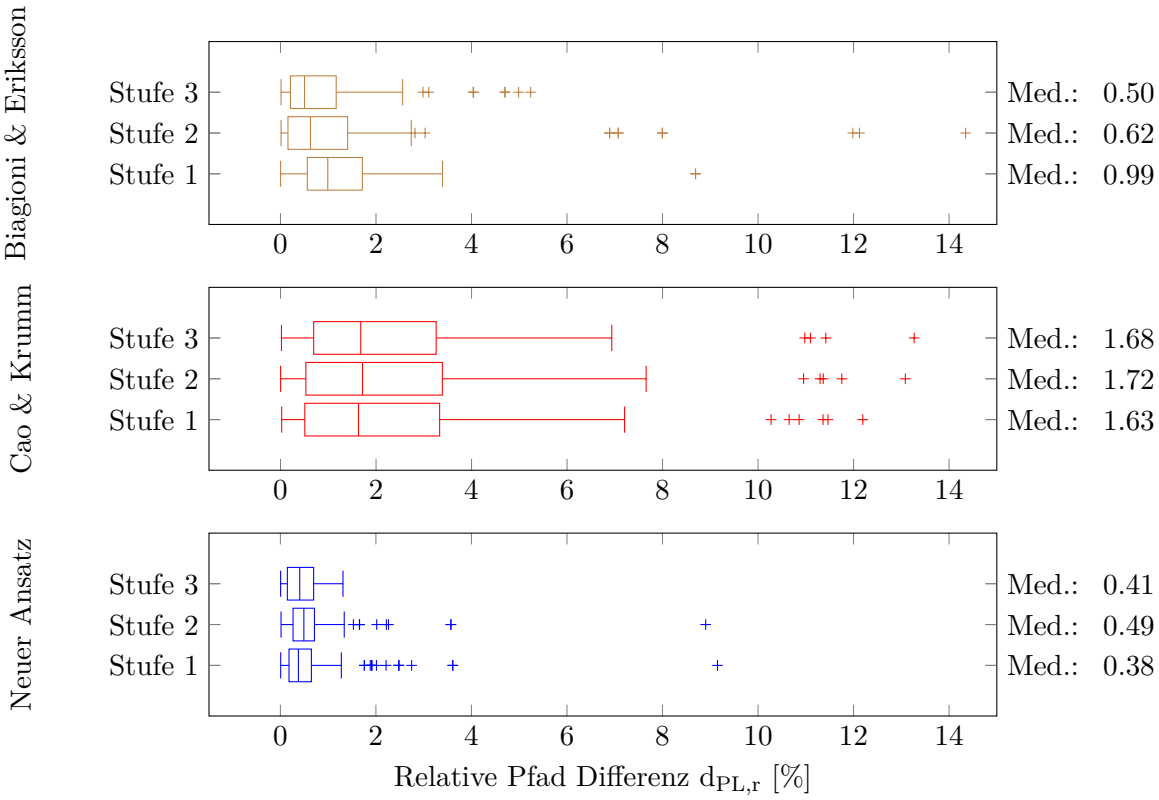


Abbildung 6.14.: Relative Längenunterschiede zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen. Die rechte Beschriftung beschreibt den Wert des jeweiligen Medians.

Bezüglich der absoluten Differenzen wird an den ausgewiesenen Werten zunächst allgemein ersichtlich, dass die Qualität der Ergebnisse mit jener der Eingabedaten steigt. Da die relativen Differenzen nicht im gleichen Verhältnis skalieren, wird deutlich, dass die Eingabedaten der verschiedenen Stufen durch den unterschiedlich großen Positionierungsfehler verschiedene Problemstellungen aufweisen.

Beim Verfahren nach Biagioni und Eriksson werden die in den Eingabedaten inhärenten Messungenauigkeiten durch eine Kerndichteschätzung rekonstruiert. Dadurch hängt die Qualität dieser Rekonstruktion und dem daraus erzeugten Straßengraphen unmittelbar von der Qualität der Daten ab. Dabei ist das Verfahren stark von individuellen Parametern geprägt, welche vor allem bei einem Szenario mit gemischten Straßenausprägungen große Auswirkungen haben. Für eine Rekonstruktion wird ein konstanter Parametersatz verwendet, allerdings würde bei einer mehrspurigen Straße ein anderer optimaler Parametersatz benötigt werden als für eine kleinere Straße. Durch die direkte Abhängigkeit von der Datenqualität sinkt allgemein der absolute und relative Pfadlängenfehler, dennoch treten aufgrund der beschriebenen Problematik Ausreißer in Dimensionen weit über dem Median auf. Da diese Bewertung eine eindimensionale Auswertung darstellt, beschreibt sie für jeden Pfad prinzipiell den *Umweg* im Vergleich zur Referenz. Dabei würden topologische Fehler in der Regel größere Umwege bedingen als geometrische. In der ersten Qualitätsstufe liegen 50 % der Ergebnisse unter $d_{PL,a} = 7.27\,\text{m}$ bzw. unter $d_{PL,r} = 1\,\%$. Diese Umwege sind aufgrund der Größe des Fehlers maßgeblich auf geometrische Ungenauigkeiten zurückzuführen. Auffällig sind

einige Ausreißer in der zweiten Qualitätsstufe im Bereich von $d_{PL,a} > 100\,\text{m}$ mit einer relativen Bewertung von $d_{PL,r} > 12\,\%$, welche auf einen topologischen Fehler zurückzuführen sind. Diese Situation tritt aufgrund eines fehlenden Streckenabschnitts auf. In der dritten Qualitätsstufe liegen $50\,\%$ der Werte unter $d_{PL,a} = 4\,\text{m}$ bzw. unter $d_{PL,r} = 0.5\,\%$ mit moderaten Ausreißern. Bei einer durchschnittlichen Trajektorienlänge von 1,05 km werden diese Ergebnisse als gut angesehen.

Im Verfahren nach Cao und Krumm werden die Trajektorien zunächst durch ein physikalisches Kräftemodell variiert, um jene, welche auf der selben Straßen aufgenommen wurden näher zueinander zu bewegen. Auch dieses Verfahren ist stark durch die Wahl der Parameter beeinflusst, was unter Umständen einerseits bedeuten kann, dass Trajektorien einer mehrspurigen Straßen nicht zueinander gebracht werden, oder andererseits, dass benachbarte kleinere Straßen fälschlicherweise zueinander gebracht werden. Beide Fehlerquellen können zu großen geometrischen Abweichungen und zu entweder irrtümlich vorhandenen oder fehlenden topologischen Verknüpfungen führen. Dieses Problem wird in den Ergebnissen deutlich, da sogar in der höchsten Qualitätsstufe $50\,\%$ der Ergebnisse nur unter $d_{PL,a} = 11.87\,\text{m}$ bzw. $d_{PL,r} = 1.68\,\%$ liegen und Ausreißer von bis zu $d_{PL,a} = 120\,\text{m}$ auftreten. Der generell in allen Stufen hohe absolute und relative Fehler ist maßgeblich auf den geometrischen Unterschied zurückzuführen, welcher durch die Veränderung der Eingabedaten entsteht. Zusätzlich entstehen große geometrische und einzelne topologische Fehler durch die beschriebene Problematik der Parameter.

Die Parameter des neuen Verfahrens müssen nicht im Kontext des Szenarios gewählt werden, sondern sind durch die durchgeführten Parameterstudien lediglich für die allgemeine Rekonstruktion getrimmt. Dadurch ist es in der Lage in den Zellen der Segmentierung Modelle zu erzeugen, welche individuell an die jeweilige Verkehrssituation angepasst werden können. Dies wird auch in den Ergebnissen deutlich, da diese in allen Qualitätsstufen verhältnismäßig sehr gut sind. Somit werden Spitzenwerte von $d_{PL,a} = 2.75\,\text{m}$ bzw. $d_{PL,r} = 0.41\,\%$ im Median der dritten Stufe und bereits $d_{PL,a} = 4.51\,\text{m}$ bzw. $d_{PL,r} = 0.38\,\%$ im Median der ersten Stufe erreicht. Auffällig ist in der ersten und zweiten Qualitätsstufe jeweils ein Ausreißer in den relativen Werten im Bereich von $d_{PL,r} = 9\,\%$. Da es keine korrespondierenden Ausreißer in den absoluten Werten gibt, sind diese auf ein fehlerhaftes Pfadmatching und nicht auf eine fehlerhafte Rekonstruktion zurückzuführen. Dabei sind die resultierenden Pfade annähernd gleich lang, im Verhältnis zur Trajektorie allerdings unverhältnismäßig.

Als nächstes werden die Pfade mit Hilfe der Hausdorff-Distanz verglichen. Die Ergebnisse sind in Abbildung 6.15 dargestellt. Bei dieser Bewertung wird für jeden Pfad der Knoten mit dem größten aller kleinsten Abstände zur Referenz gesucht und berücksichtigt im Gegensatz zur Pfadlängen-Differenz mehrere Dimensionen. Allgemein ist in allen Ergebnissen eine tendenzielle qualitative Verbesserung mit zunehmender Qualität der Eingabedaten erkennbar.

Da jede Trajektorie, respektive jeder Pfad, durch seine schlechteste Messung repräsentiert wird, sind die Ergebnisse maßgeblich durch die Schwachstellen des Systems geprägt. Das Verfahren nach Biagioni und Eriksson weist Schwierigkeiten an großen Kreuzungen auf. Da nicht jede topologische Verknüpfung gleich häufig durch die Trajektorien abgedeckt wird, entsteht durch die Kerndichteschätzung kein symmetrisches Bild mit Schwerpunkt auf der Kreuzungsmitte, sondern

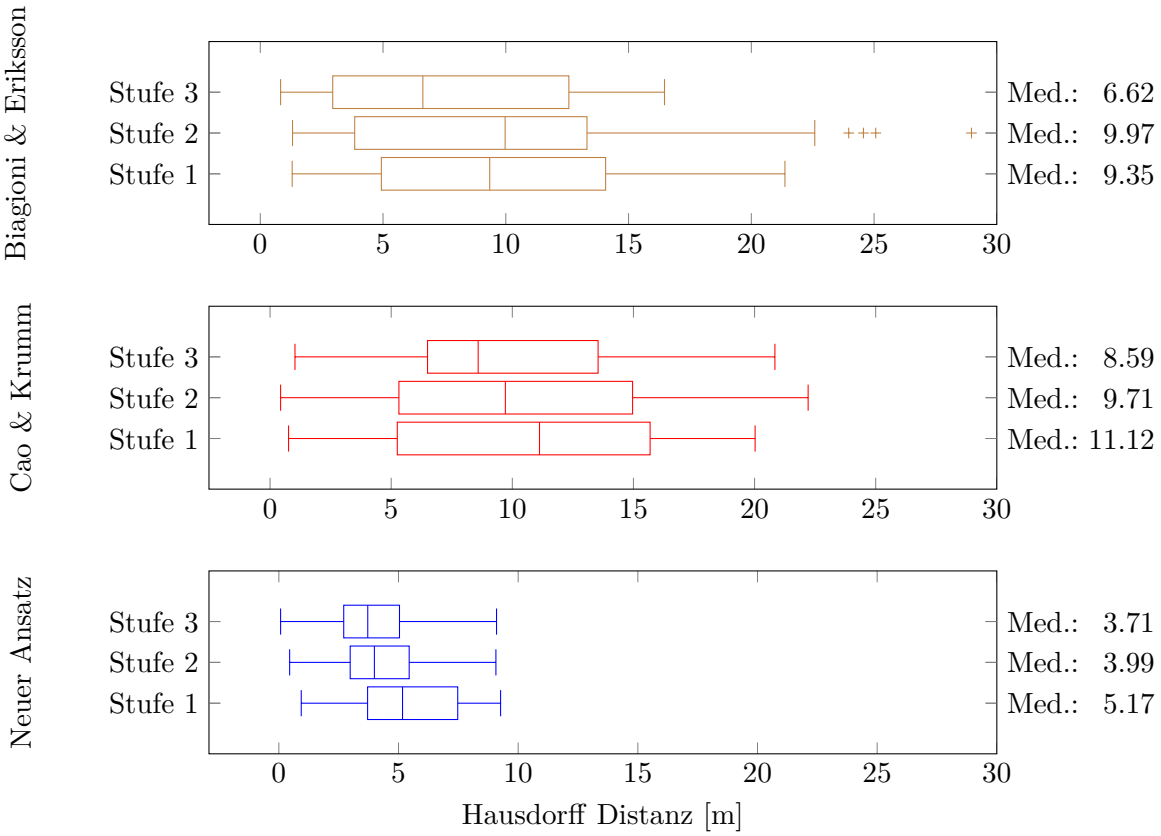


Abbildung 6.15.: Hausdorff-Distanz zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien- Qualitätsstufen.

eine schiefe Verteilung, beeinflusst durch die größte Häufung von Trajektorien. Daher sind die meisten der Bewertungen auf geometrische Ungenauigkeiten in derartigen Situationen zurückzuführen. Dabei liegen in der ersten Qualitätsstufe 50 % der Werte zwischen $d_{HD} = 5\,\text{m}$ und 14 m und in der dritten Stufe zwischen $d_{HD} = 3\,\text{m}$ und 13 m. Erneut auffällig sind die Ergebnisse mit Eingabedaten der zweiten Qualitätsstufe, welche bei einem Median von $d_{HD} = 9.97\,\text{m}$ und Ausreißern von bis zu 30m qualitativ unter jenen der ersten Qualitätsstufe liegen. Dabei beschreiben die Ausreißer die Distanz, in welcher der fehlende Streckenabschnitt umfahren werden muss. Diese Bewertungen decken sich mit jenen der Pfadlängen-Differenzen und sind auf topologische Fehler zurückzuführen.

Nach der Veränderung der Eingabedaten im Verfahren nach Cao und Krumm bilden die Trajektorien das oben beschriebene Problem in ähnlicher Weise ab. Das Kräftemodell konzentriert die Daten im Mittel der Trajektorien und neigt somit dazu, Kreuzungen zu verzerren. In den Ergebnissen liegt der Median der ersten Qualitätsstufe bei $d_{HD} = 11.12\,\text{m}$ und wird analog zur steigenden Qualität der Eingabedaten verbessert. In der dritten Stufe liegen 50 % der Werte zwischen $d_{HD} = 6\,\text{m}$ und 13 m bei einem Median von $d_{HD} = 8.59\,\text{m}$.

Das neue Verfahren konzentriert das Modell durch die Datenbewertung in Abschnitt 5.3.4 ebenfalls im Mittel der Daten, erhält durch die Berücksichtigung des Fahrtwinkels und der a-priori Bewertung die Struktur der Kreuzung und bildet somit den Mittelpunkt präziser ab, als die bisher beschriebenen Verfahren. Es wird ersichtlich, dass bereits alle Bewertungen auf Basis der ersten Qualitätsstufe

unter $d_{HD} = 10\,\text{m}$ liegen. Dabei werden die Ergebnisse in den nächsten Stufen im Mittel noch verbessert, der Bewertungsradius (Whisker des Boxplot) verändert sich allerdings kaum. Für die fahrbahngenaue Rekonstruktion können mit diesem Verfahren somit bereits auf reinen GPS - Daten hervorragende Ergebnisse erzielt werden. In der dritte Stufe liegen 50 % der Werte zwischen $d_{HD} = 3\,\text{m}$ und 5 m, wobei die beste Messung, mit einer Bewertung von $d_{HD} = 0.07\,\text{m}$, im Rahmen der Messgenauigkeit als fehlerfrei angesehen werden kann.

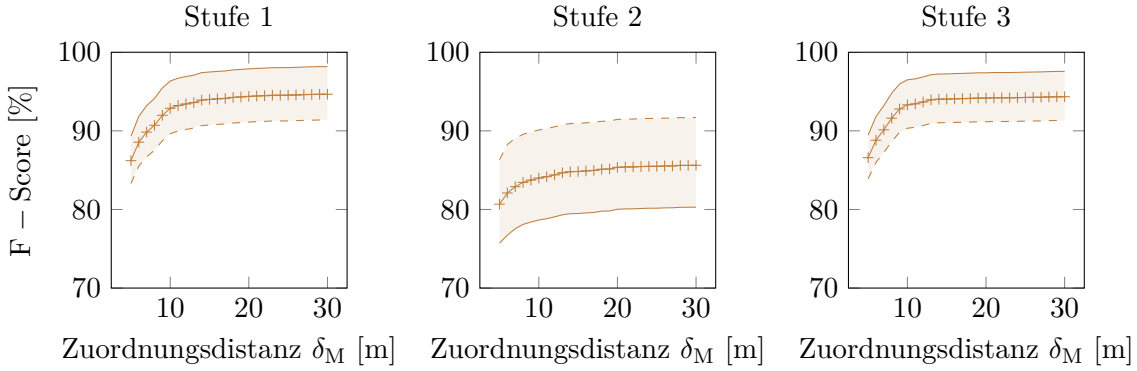


Abbildung 6.16.: Auswertung des F-Score auf den Ergebnissen nach Biagioni & Eriksson in allen Trajektorien Qualitätsstufen. Legende: Der F-Score wird mit + Markierungen, der Recall das durchgezogene und die Precision als gestrichelte Linie dargestellt.

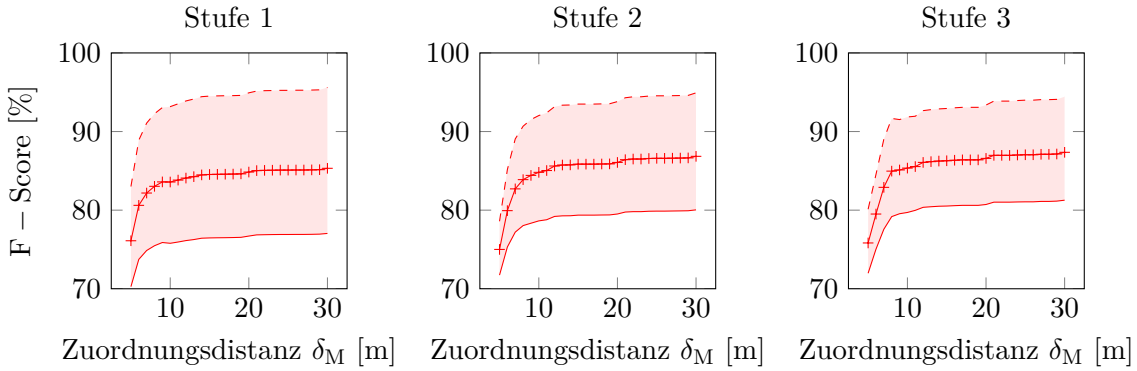


Abbildung 6.17.: Auswertung des F-Score auf den Ergebnissen nach Cao & Krumm in allen Trajektorien Qualitätsstufen. Legende: Der F-Score wird mit + Markierungen, der Recall das durchgezogene und die Precision als gestrichelte Linie dargestellt.

Als nächstes wird der F-Score ausgewertet. Zur allgemeinen Interpretation gilt zunächst, je höher die Werte, desto besser die Ergebnisse, da sich demnach die Rekonstruktion und die Referenz entlang ihrer Kanten im Graphen ähneln. Eine weitere Interpretation liefert die Diskrepanz zwischen Precision und Recall. Beide liefern Werte, welche das Fehlen von „Löchern“ bzw. „Murmeln“ beschreiben. Ein großer Unterschied zwischen den Werten beschreibt somit einen einseitigen Mangel, was bedeutet, dass beispielsweise systematisch zu viele „Murmeln“ erzeugt werden, weil die Geometrie oder Topologie der Rekonstruktion jener der Referenz zu wenig ähnelt. Ein geringer Abstand zwischen den Werten beschreibt punktuelle Fehler, wobei streckenweise eine gute Zuordnung existiert und vereinzelt ein grober Fehler oder eine Agglomeration von kleinen Fehlern zu einer fehlerhaften Zuordnung führt. Dadurch treten die Zuordnungsfehler häufig paarweise auf, wodurch

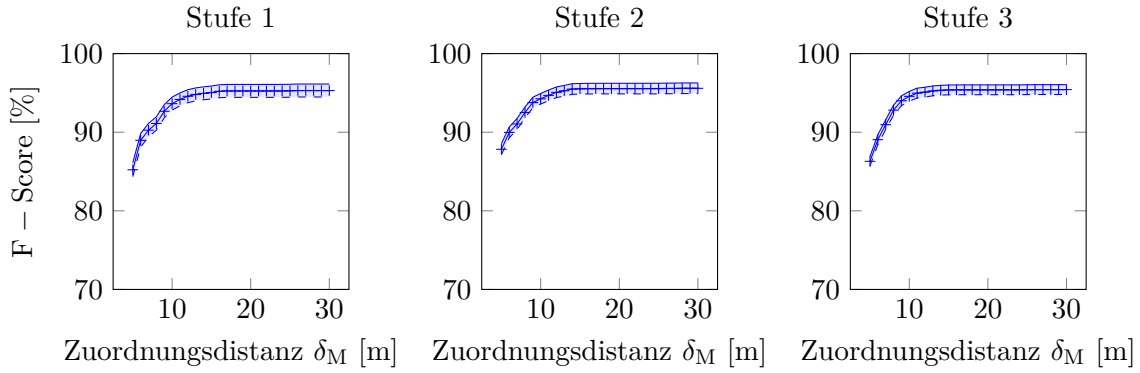


Abbildung 6.18.: Auswertung des F-Score auf den Ergebnissen nach dem neuen Ansatz in allen Trajektorien Qualitätsstufen. Legende: Der F-Score wird mit + Markierungen, der Recall das durchgezogene und die Precision als gestrichelte Linie dargestellt.

die Kennwerte nah beieinander liegen. Bei der Erzeugung der Ergebnisse des F-Score wurden pro Auswertung 1500 Durchläufe mit einem Distanzwert von $\delta_F = 100\,\text{m}$ und einem longitudinalen Abstand von $\delta_D = 10\,\text{m}$ durchgeführt. Die Zuordnungsdistanz δ_M wird in jeder Auswertung entsprechend der x-Achse variiert.

In den Ergebnissen nach Biagioni und Eriksson, dargestellt in Abbildung 6.16, konvergiert der F-Score in den Bewertungen der ersten und dritten Stufe jeweils gegen $d_F = 94\,\%$. Dabei wird in der ersten Stufe ein durchschnittlicher Precision-Recall-Korridor von 6.6 % und in der dritten Stufe von 6.1 % eingehalten. Da in beiden Fällen der Recall-Wert über der Precision liegt, sind mehr „Murmeln“ als „Löcher“ nicht zugeordnet worden. Da der Korridor sehr schmal ist, sind die fehlenden Zuordnungen auf die geometrischen Fehler zurückzuführen und durch punktuelle Mängel, z. B. an Kreuzungen, verursacht. Da in beiden Fällen bereits bei der kleinsten Zuordnungsdistanz ein Wert von $d_F = 86\,\%$ erreicht wird, sind die bereits identifizierten Schwachstellen gravierend, während der Rest der Rekonstruktion nur sehr wenig Mängel aufweist, da sonst ein sehr viel niedrigerer Wert erreicht werden würde. Wie bereits in den anderen Auswertungen sind die Ergebnisse der zweiten Stufe auffällig. Hier konvergiert der F-Score gegen $d_F = 85\,\%$ bei einem durchschnittlichen Korridor von 11.4 %, wobei in diesem Fall der Precision Wert über dem Recall liegt. Aufgrund des fehlenden Streckenabschnitts bleiben viele „Löcher“ auf der Referenzkarte leer, wodurch der Recall stark sinkt. Da sich die fehlenden „Murmeln“ nicht auswirken können, bleibt der Precision Wert ähnlich dem der anderen Stufen.

In den Ergebnissen nach Cao und Krumm, dargestellt in Abbildung 6.17, ist eine kontinuierliche Verbesserung der Werte mit steigender Datenqualität ersichtlich. Der F-Score konvergiert in den drei Stufen gegen $d_F = 85\,\%$, 86 % bzw. 87 % bei einem jeweiligen durchschnittlichen Korridor von 17.6 %, 13.6 %, bzw. 12.2 %. In allen Ergebnissen liegt der Recall unter der Precision, somit gibt es viele nicht gefüllte „Löcher“ auf der Referenzkarte. Da alle Precision-Werte gegen hohe Werte konvergieren, kann dieses Phänomen nicht nur auf topologische Mängel, wie es die ersten Auswertungen nahe legen, zurückgeführt werden. Begründet werden können diese geometrischen Fehler durch die Veränderung der Eingabewerte. In vielen Situationen bewirkt dies, dass Kurven

bzw. Kreuzungen in der Rekonstruktion *geschnitten* werden und diese somit in Summe kürzer sind, wodurch weniger „Murmeln“ als „Löcher“ erzeugt werden.

In Abbildung 6.18 sind die Ergebnisse des neuen Ansatzes dargestellt. Wie bereits in den anderen Auswertungen ersichtlich, sind die Ergebnisse der ersten Stufen mit einem F-Score Grenzwert von $d_F = 96\,\%$ bereits sehr gut und können mit steigender Datenqualität kaum verbessert werden. Auch in den anderen Stufen konvergiert der F-Score gegen $d_F = 96\,\%$ bei einem durchschnittlichen Precision-Recall-Korridor von 1.6 %, 1.3 % bzw. 1.2 %. Der sehr schmale Korridor impliziert eine große Ähnlichkeit zwischen den Graphen, da kein relativer Mangel an „Löchern“ oder „Murmeln“ auftritt. Die fehlenden Bewertungspunkte sind somit auf leichte geometrische Fehler zurückzuführen.

Insgesamt ist das neue Verfahren, belegt durch die drei Auswertungen, in der Lage mit den Ergebnissen der Verfahren aus der Literatur mitzuhalten und diese sogar zu übertreffen. Hinsichtlich der Pfadlängen-Differenz d_{PL} erzielt das Verfahren im Vergleich sehr gute und konstant niedrige absolute Werte. Die dadurch implizierte topologische Qualität wird durch die Hausdorff-Distanz d_{HD} bestätigt, wobei in jeder Stufe ein maximaler Abstand von $d_{HD} < 10\,\mathrm{m}$ erreicht wird, ein sehr guter Wert bei einer durchschnittlichen Trajektorienlänge von 1.05 km. Abschließend wird nicht nur die topologische Qualität durch den F-Score bestätigt, sondern durch die konstant schmalen Precision-Recall-Korridore in allen Stufen auch die geometrische Qualität.

Objekt-Trajektorien

Bisher wurden die für die Auswertungen die zugrundeliegenden Rekonstruktionen auf Basis der Fahrzeug Ego-Trajektorien erzeugt. Als nächstes werden die, ebenfalls in Abschnitt 6.1 beschriebenen Objekt-Trajektorien zusätzlich verwendet, um die Straßengraphen zu rekonstruieren. Diese Trajektorien stammen dabei von Verkehrsteilnehmern, welche vom Egofahrzeug mit Hilfe einer Kamera beobachtet wurden. Die Gesamtanzahl an Trajektorien nimmt damit stark zu, allerdings sind viele der Objekt-Trajektorien sehr kurz und beinhalten wenige Informationen. Da die Objekt-Trajektorien immer relativ zum Egofahrzeug aufgenommen wurden, sind durch sie keine bisher unerschlossenen Gebiete vermessen. Die zu erwartenden Änderungen beziehen sich daher im besten Fall auf die Steigerung der geometrischen Genauigkeit und die Ausbesserung von topologischen Mängeln. Aus diesem Grund werden nicht alle Bewertungen neu durchgeführt, sondern nur die absoluten Pfadlängen-Differenzen und die Hausdorff-Distanz betrachtet, da in beide etwaige Fehler erkennbar waren und somit Unterschiede identifiziert werden können.

In den Ergebnissen der Pfadlängen-Differenz in Abbildung 6.19 nach Biagioni und Eriksson fällt zunächst auf, dass der fehlende Straßenabschnitt in der zweiten Stufen, welcher zuvor für Ausreißer im Bereich von $d_{PL,a} > 100\,\mathrm{m}$ gesorgt hat, durch die Hinzunahme der Objekt-Trajektorien erfolgreich rekonstruiert werden konnte. Außerdem wird ersichtlich, dass in allen Stufen der Median leicht gestiegen, die Ergebnisplots der ersten und dritten Stufe allerdings kompakter geworden sind. In den Ergebnissen der Hausdorff-Distanz in Abbildung 6.20 wird ebenfalls die erfolgreiche Rekonstruktion in der zweiten Stufe ersichtlich. In der dritten Stufe haben die Objekt-Trajektorien einen großen Einfluss, der Median ist im Vergleich stark gesunken und die Plots sind kompakter.

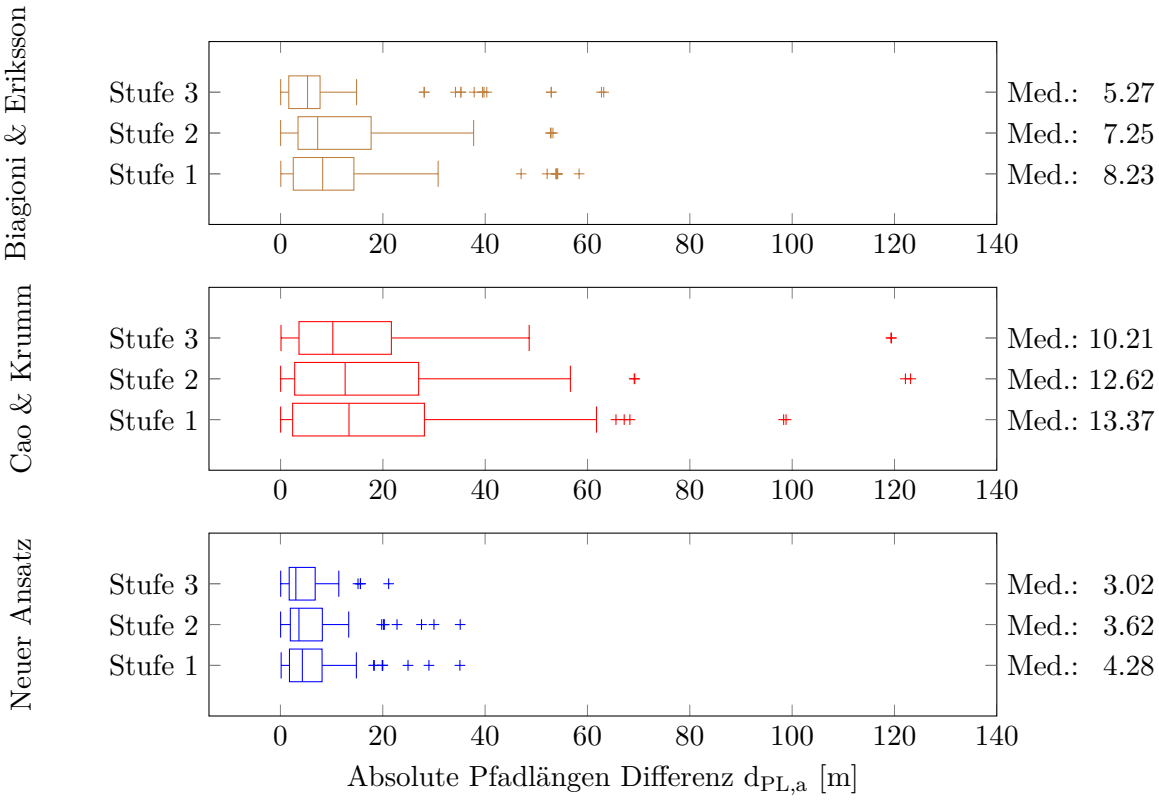


Abbildung 6.19.: Absolute Längenunterschiede zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen. Die rechte Beschriftung beschreibt den Wert des jeweiligen Medians.

Das Verfahren nach Cao und Krumm kann die zusätzlichen Eingabedaten gut verarbeiten. Durch das Vorverarbeiten der Daten können die redundanten Informationen gut vereinigt und der finale Graph geometrisch besser und topologisch zuverlässiger rekonstruiert werden. Bezüglich der absoluten Pfadlängen-Differenzen werden in allen Stufen die Werte im Median gesenkt, die Plots sind kompakter und die Ausreißer in Menge und Betrag dezimiert. Bei der Auswertung der Hausdorff-Distanz wird ersichtlich, dass die Werte im Median ebenfalls gesenkt, allerdings die Ergebnisse breiter gestreut werden.

Beim neuen Ansatz konnte der Median durch die Hinzunahme der Objekt-Trajektorien in der ersten und zweiten Qualitätsstufe ein wenig gesenkt werden, allerdings auf Kosten einiger Ausreißer. Diese sind zurückzuführen auf einige, vornehmlich in kurvigen Bereichen auftretende, fehlerbehaftete Beobachtungen. Im Verfahren nach Biagioni und Eriksson werden diese durch die Kerndichteschätzung verschmolzen, bei Cao und Krumm durch das Kräftemodell (meistens) auf die Straße zurück projiziert. Das neue Verfahren berücksichtigt diese als vollwertige Messung und trainiert das Modell entsprechend. Im Verhältnis zur Datenmenge wenige fehlerhafte Messungen haben dabei keinen expliziten Einfluss auf das Model. Häufig entstehen Messfehler aufgrund bestimmter geometrischer Konstellationen und somit mehrfach bei erneuter Durchfahrung. Da die Anzahl an Ausreißern nur einen kleinen Bruchteil aller Messungen ausmacht, sind diese dabei auf sehr wenige konstruktive Fehler zurückzuführen. Diese Schlussfolgerungen sind auch in der Hausdorff-Distanz erkennbar,

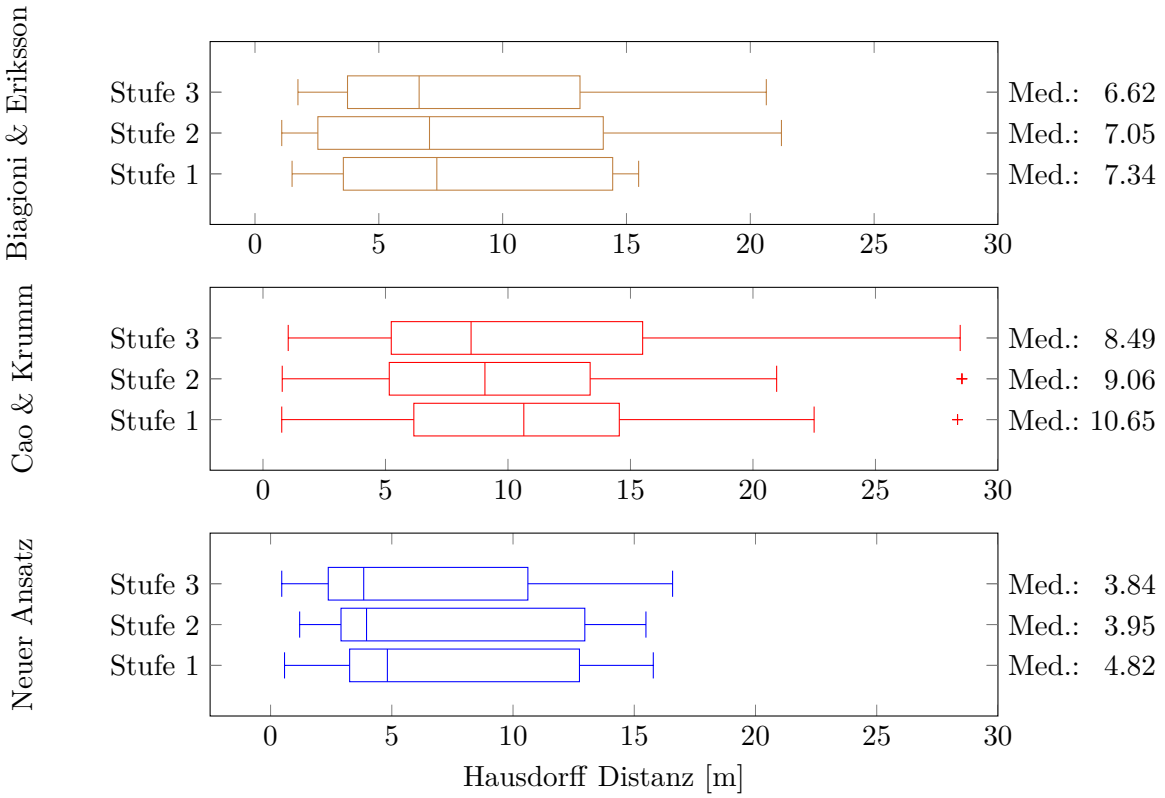


Abbildung 6.20.: Hausdorff-Distanz zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen.

in der zweiten und dritten Stufe konnte der Median leicht gesenkt werden, allerdings streuen die Ergebnisse weiter auseinander.

Insgesamt können die Objekt-Trajektorien von den Verfahren gut verarbeitet werden. Die redundanten Informationen erzeugen beim Verfahren nach Cao und Krumm die größte Verbesserung, allerdings können alle Verfahren in einigen Bereichen profitieren. Der neue Ansatz kann die Ergebnisse im Median häufig verbessern, allerdings immer auf Kosten einiger schlechterer Ergebnisse. Dabei ist erwähnenswert, dass die Ergebnisse aus den Ego-Trajektorien bereits qualitativ so hochwertig sind, dass eine Verbesserung in den meisten Situation kaum möglich ist. Dennoch kann das Verfahren insgesamt auch mit einer großen Anzahl an Trajektorien umgehen und erzeugt im Verhältnis die qualitativ besten Ergebnisse auf allen Eingabedatensätzen.

6.5. Fahrspurgenaue Rekonstruktion

In diesem Abschnitt wird der Algorithmus zur Erzeugung fahrspurgenauer Straßenkarten aus Abschnitt 5.4 auf die Eingabedaten aus Abschnitt 6.1 angewandt. Als Grundlage zur Initialisierung der Modelle wird, unter Verwendung der in Abschnitt 6.4 durchgeführten Parameterstudien, eine *fahrbahngenaue* Straßenkarte gemäß Algorithmus 5 erzeugt. Zur Erzeugung des bestmöglichen fahrspurgenauen Ergebnisses werden zunächst mehrere Parameterstudien durchgeführt. Unter Verwendung der ermittelten Parameter werden auf den verschiedenen Datensätzen fahrspurge-

naue Karten erzeugt. Zur Evaluierung der Ergebnisse werden diese mit der Referenzkarte aus Abschnitt 6.2 verglichen. Dabei werden zunächst die rekonstruierten Eigenschaften der Kreuzungs- und Straßenmodelle direkt verglichen und mehrdimensional visualisiert. Um den Mehrwert des Verfahrens zu beschreiben, werden die Ergebnisse anschließend mit jenen eines naiven Ansatzes, unter Verwendung der in Abschnitt 6.3 beschriebenen Bewertungen verglichen. Dabei werden als erstes die Ego-Trajektorien zur Erzeugung der Straßenkarten herangezogen. Anschließend werden die, ebenfalls in Abschnitt 6.1 beschriebenen Objekt-Trajektorien hinzugenommen, um die Auswirkungen zu bewerten. Zusätzlich wird eine Hauptverkehrsstraße entkoppelt betrachtet, um Experimente bezüglich der benötigten Anzahl an Trajektorien durchzuführen.

6.5.1. Analyse des Verfahrens

Gemäß der Beschreibung des Verfahrens in Abschnitt 5.4 wird die Entwicklung der Modelle durch sieben Parameter bestimmt. Bezüglich der Eigenschaften eines Blockes beeinflussen $\mu_{\text{add lane}}$ und $\sigma_{\text{add lane}}$ die initiale Spurbreite einer neuen Spur und $\sigma_{\text{middle gap}}$, $\sigma_{\text{lane width}}$, sowie $\sigma_{\text{curvature}}$ die Schrittweiten der jeweiligen Parameterentwicklungen. Als modellunabhängige Parameter werden die beiden Laufzeiten des Simulated Annealing der verschiedenen Phasen untersucht. Um in der Zielfunktion (5.9) keinen der Anteile zu bevorzugen, wird der Verhältnisparameter zu $\delta = 0.5$ gesetzt.

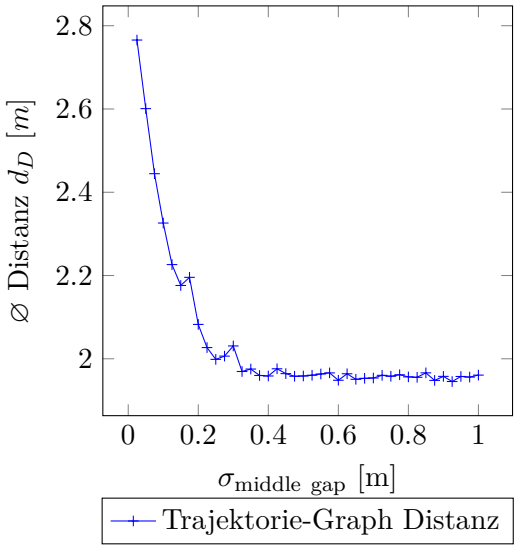
Die Parameter $\mu_{\text{add lane}}$ und $\sigma_{\text{add lane}}$ definieren eine Normalverteilung, welche verwendet wird, um die initiale Spurbreite einer hinzugefügten Fahrspur zu wählen. Diese werden nicht als Teil der Parameterstudien ermittelt, da sie, wie in Abschnitt 5.4.3 beschrieben, im Kontext der Szenarios gewählt werden müssen. Festgelegt sind derartige Werte in der *Richtlinie für die Anlage von Stadtstraßen (RASt)* (Baier, 2006). Da das Verfahren auf einer straßengenauen Karte basiert, wäre es möglich, diese um Informationen bezüglich der Stadtgebiete anzureichern[f]. Auf Basis einer derartigen Karte können Geltungsbereiche für verschiedene Parametersätze festgelegt werden. Somit könnte ein Wohngebiet mit anderen Parametern als ein Industriegebiet behandelt werden. Das im Rahmen dieser Arbeit behandelte Szenario erstreckt sich nicht über verschiedene Stadtgebiete und setzt sich aus den, in der RASt 06 festgelegten, Situationen „Verbindungsstraße“ und „örtliche Geschäftsstraße“ zusammen. Da sich dabei die relevanten Parameter nicht unterscheiden, ist die Funktionalität verschiedener Stadtgebiete nicht berücksichtigt. Die geltenden Parameter werden im Folgenden ermittelt.

Da eine echte Verkehrssituation nicht immer aus genau einer Situation gemäß der RASt besteht, muss für die Spurbreite ein Mittelwert $\mu_{\text{add lane}}$ gefunden werden, welcher sowohl für die „Verbindungsstraße“, als auch für die „örtliche Geschäftsstraße“ anwendbar ist. Basierend auf diesem Mittelwert wird eine Normalverteilung $\text{Norm}^{\text{g}}(\mu_{\text{add lane}},\sigma^2_{\text{add lane}})$ definiert, sodass eine zufällige Realisierung dieser Verteilung eine Spurbreite der in Frage kommenden Situationen repräsentiert. Allgemein gilt zunächst, dass auf dem gesamten Szenario Mischverkehr aus PKW, LKW und

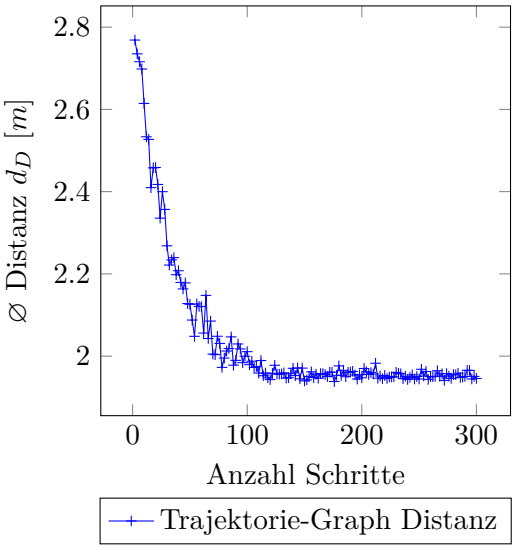
[f] bspw. durch die Hinzunahme von http://www.openstreetmap.org
[g] vgl. Anhang B.2

Linienbusverkehr besteht. Aus einer Verkehrszählung der Stadt Hildesheim aus dem Jahr 2006 (vgl. Anhang D.1) geht hervor, dass auf dem betrachteten Szenario pro Tag im Schnitt 30.000 Fahrzeuge fahren. Das bedeutet ein Verkehrsaufkommen von durchschnittlich 1250 Fahrzeugen pro Stunde, wobei es Schwankungen zwischen den Hauptverkehrs- und Nachtzeiten gibt. Außerdem gibt es in dieser Durchschnittsberechnung hoch und niedrig frequentiert befahrene Straßen. Somit wird es Verkehrswege mit Spitzenbelastungen von mehr und mit weniger als 1250 Fahrzeugen pro Stunde geben. In der RASt 06 ist die „örtliche Geschäftstraße“ in Abschnitt 5.2.7 definiert, wobei sich die Bemaßungen nach dem Verkehrsaufkommen unterscheiden. Somit fallen die Straßen des Szenario einerseits in die Kategorie $400 - 1000$ und andererseits in $800 - 1800$ Fahrzeuge pro Stunde. Die „Verbindungsstraße“ ist in Abschnitt 5.2.11 definiert. Entsprechende Straßen gehören zu den mehr befahrenen Verkehrswegen des Szenarios, daher wird hier nur die Kategorie $800 - 1800$ Fahrzeuge pro Stunde betrachtet. Zusätzliche Faktoren sind in beiden Klassifizierungen die vorhandene gesamte Straßenraumbreite, wie auch das Vorhandensein von Fuß- und Radwegen, sowie Parkflächen. Für die betrachteten Verkehrswege ergibt sich somit eine Spurbreite von entweder 2.25 m oder 3.25 m. Die Parameter der Normalverteilung werden so gewählt, dass 50 % der Werte einer Stichprobe zwischen den beiden Werten liegen:

$$\mu_{\text{add lane}} = 2.75\,\text{m}$$
$$\sigma^2_{\text{add lane}} = (0.55\,\text{m})^2$$



(a) Mehrfache Auswertung des Algorithmus auf dem Parameterbereich $\sigma_{\text{middle gap}} \in [0.025,\ 1.0]$ m.



(b) Mehrfache Auswertung des Algorithmus auf dem Parameterbereich $s_{\text{warumup}} \in [1{,}300]$.

Abbildung 6.21.: Parameterstudie von $\sigma_{\text{middle gap}}$ und SA_{warmup} in der Aufwärmphase.

Wie in Algorithmus 5 beschrieben, ist das Verfahren in eine Aufwärm- und eine Hauptphase gegliedert. Da in der Aufwärmphase nur die *Adjust Middlegap* Operation erlaubt ist und ein eigener Laufzeitparameter für das Simulated Annealing verwendet wird, können diese beiden Parameter innerhalb dieser Phase untersucht werden. Der Parameter $\sigma_{\text{middle gap}}$ beschreibt die

maximale Schrittweite, um welche die bauliche Trennung zwischen den Fahrtrichtungen verändert werden kann. Da noch kein optimaler Wert für das Simulated Annealing ermittelt wurde, wird für die Untersuchung in Gleichung (5.5) eine Schrittanzahl von $s = 300$ verwendet, da dieser Wert nach empirischen Versuchen den Parameter überschätzt. Zur Auswertung wird $\sigma_{\text{middle gap}}$ über den Werten $[0.025, 1.0]\,\text{m}$ variiert und in Abbildung 6.21a die Entwicklung der einfachen Distanz Bewertung d_D aufgetragen. Es wird eine Verbesserung bis zum Wert $\sigma_{\text{middle gap}} = 0.4\,\text{m}$ festgestellt.

Unter Verwendung dieses Wertes kann eine Studie bezüglich der Anzahl an Schritten im Simulated Annealing durchgeführt werden. Dabei wird λ_{warmup} gemäß Gleichung (5.5) in Abhängigkeit der Schrittanzahl $s_{\text{warmup}} \in [1{,}300]$ bestimmt. Abbildung 6.21b stellt die Schrittzahl über der einfachen Distanz Bewertung d_D dar, wobei eine Verbesserung bis $s_{\text{warmup}} = 120$, respektive $\lambda_{\text{warmup}} = 0.77$ ermittelt wird.

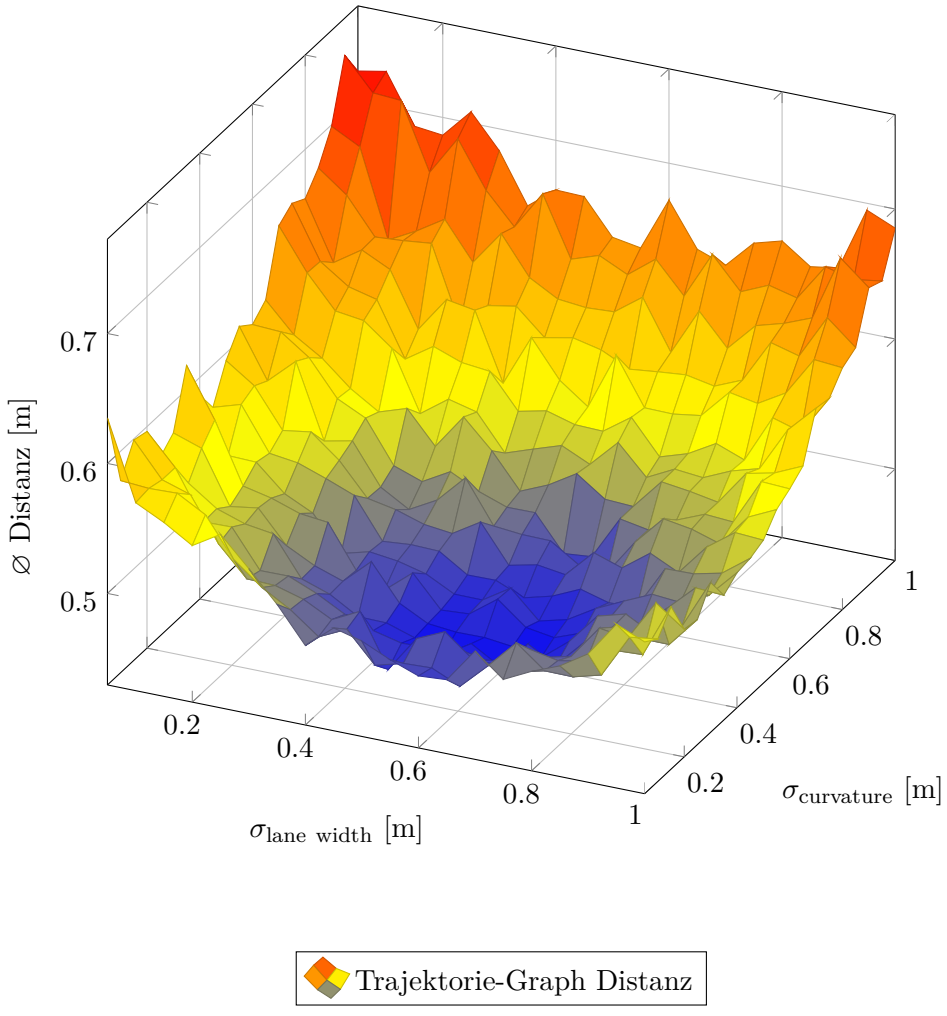


Abbildung 6.22.: Parameterstudie von $\sigma_{\text{lane width}} \in [0.05, 1.0]\,\text{m}$ und $\sigma_{\text{curvature}} \in [0.05, 1.0]\,\text{m}$ in der Hauptphase.

In der Hauptphase wird ein eigener zu studierender Laufzeitparameter des Simulated Annealing verwendet. Zusätzlich sind alle Operationen zulässig, wodurch die Parameter $\sigma_{\text{lane width}}$ und $\sigma_{\text{curvature}}$ relevant werden. Die beiden Variablen beschreiben die Veränderungsschrittweite der Parameter bezüglich der individuellen Spurbreiten und der Krümmung eines Blocks. In diesem Zusammenhang

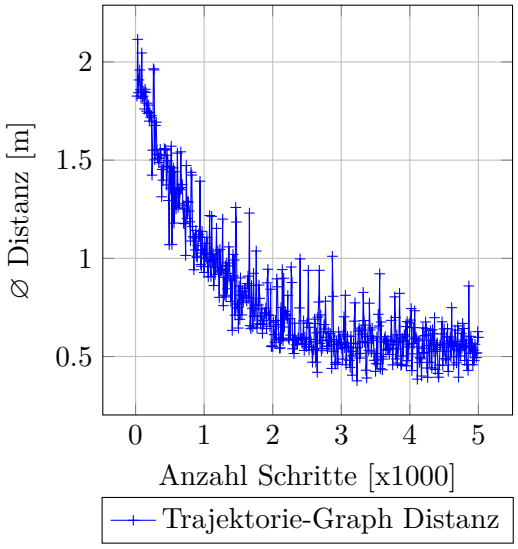


Abbildung 6.23.: Parameterstudie von $s_{\text{main}} \in [1; 5000]$ in der Hauptphase.

wird deutlich, dass ein nicht gekrümmter Block beispielsweise eine Kurve anders, mutmaßlich mit sehr breiten Spuren, abbilden würde als ein gekrümmter. Daher ist nicht möglich die beiden Parameter unabhängig voneinander zu untersuchen. Da noch kein Laufzeitparameter des Simulated Annealing ermittelt wurde, wird für die Auswertung eine Schrittanzahl von $s_{\text{main}} = 5000$ gewählt, da dieser Wert nach empirischen Versuchen den Parameter überschätzt. Zur Auswertung werden beide Parameter über den Werten $[0.05, 1.0]\,\text{m}$ variiert. Die Auswertung der einfachen Distanz Bewertung d_D über den beiden Variablen ist in Abbildung 6.22 dargestellt. Das Bewertungsminimum befindet sich bei $\sigma_{\text{lane width}} = 0.6\,\text{m}$ und $\sigma_{\text{curvature}} = 0.3\,\text{m}$.

Abschließend wird für die Hauptphase die optimale Laufzeit des Simulated Annealing bestimmt. Dazu werden, unter Berücksichtigung aller bisher studierten Parameter, mehrere Durchläufe des Algorithmus mit verschiedenen Laufzeiten mit der einfachen Distanz Bewertung d_D ausgewertet. In Abbildung 6.23 ist diese Auswertung dargestellt, wobei eine Verbesserung bis zu $s_{\text{main}} = 3200$, respektive $\lambda_{\text{main}} = 0.029$ festgestellt wird.

6.5.2. Bewertung der Ergebnisse

Mit den ermittelten (vgl. Tabelle 6.4) und gewählten (vgl. Tabelle C.4) Parametern kann, basierend auf den Eingabedaten der Ego-Trajektorien aus Abschnitt 6.1 eine fahrspurgenaue Straßenkarte erzeugt werden.

Direkter Vergleich

Im Folgenden werden die Eigenschaften der Straßen und Kreuzungen direkt mit der Referenzkarte verglichen, um etwaige Fehlerquellen untersuchen zu können. Dazu werden zunächst eine beispielhafte Straße und eine Kreuzung separat untersucht und bewertet. Anschließend werden diese Eigenschaften

Parameter	Wert
$\mu_{\text{add lane}}$	3.25 m
$\sigma_{\text{add lane}}$	0.55 m
$\sigma_{\text{middle gap}}$	0.4 m
$\sigma_{\text{lane width}}$	0.6 m
$\sigma_{\text{curvature}}$	0.3 m
s_{warmup}	120
s_{main}	3200

Tabelle 6.4.: Parameter des Algorithmus zur Rekonstruktion einer fahrspurgenauer Straßenkarten.

auf dem gesamten Szenario betrachtet. Die Ergebnisse wurden basierend auf den Trajektoriendaten der dritten Stufe erzeugt.

In Abbildung 6.24 werden auf einer exemplarischen Straße die Fehler in den Eigenschaften bauliche Trennung, Fahrspuranzahl und Fahrspurbreite (Summe) der Rekonstruktion im Vergleich zur Referenz dargestellt. In Abbildung 6.24a ist die entsprechende Straße als Auszug aus der Referenzkarte abgebildet, während in Abbildung 6.24b - 6.24d die Straße der Rekonstruktion mit der Visualisierung jeweils eines Fehlermaßes in Rot dargestellt ist. Zusätzlich sind die Blockgrenzen der Referenz gestrichelt markiert und mit den Bezeichnungen Ax nummeriert, während die Grenzen in der Rekonstruktion durchgezogen markiert und mit den Bezeichnungen Bx nummeriert sind. Diese Straße wird zur Analyse ausgewählt, da jeder Block eine individuell andere Verkehrssituation abbildet. Der Block A1 stellt einen vierspurigen Abschnitt dar, mit der Besonderheit, dass in der Referenz die äußere Spur nicht verschwindet, sondern in eine im weiteren Verlauf nicht mehr verzeichnete Busspur übergeht und somit abrupt endet. Block A2 beinhaltet einen normalen dreispurigen Straßenabschnitt, während A3 das Verschwinden einer Spur in einem Verbindungsblock darstellt. Dabei ist besonders, dass die innen liegende Spur verschwindet und in eine bauliche Trennung übergeht. Block A4 und A5 beschreiben einen zweispurigen Abschnitt mit sich ändernder baulicher Trennung.

Der Block B1 korrespondiert mit A1 und weist in der Rekonstruktion keinen Fehler in der baulichen Trennung und in der Anzahl der Spuren auf. Bezüglich der Spurbreiten wird ein auffällig großer Fehler verzeichnet. Dies liegt daran, dass die äußere der drei nach links führenden Fahrspuren als verschwindende Fahrspur abgebildet wird. In der Referenz bleibt die Spur aufgrund des Übergangs in eine Busspur konstant, wodurch der linear verlaufende Fehler als Differenz der Spurbreiten entsteht. Diese Informationen sind nicht in den Trajektorien abgebildet und können somit vom Verfahren nicht extrahiert werden. Somit beschreibt dieser Fehler keine Schwachstelle des Verfahrens, sondern eine Grenze.

Im Block B2 ist die bauliche Trennung korrekt konstruiert worden. Bezüglich der Spuranzahl ist ein häufig vorkommendes Artefakt ersichtlich: da in den Daten der longitudinale Übergang zwischen zwei Blöcken nicht eindeutig identifizierbar ist, können Übergangsbereiche entstehen, in denen die absolute Spuranzahl kurzweilig verschieden ist. Aufgrund der mangelnden longitudinalen Informationen können derartige Bereiche im besten Fall minimiert, allerdings selten vermieden werden. Dieser Fehler ist somit nicht auf eine schlechte Extraktion der Spuranzahl, sondern auf die

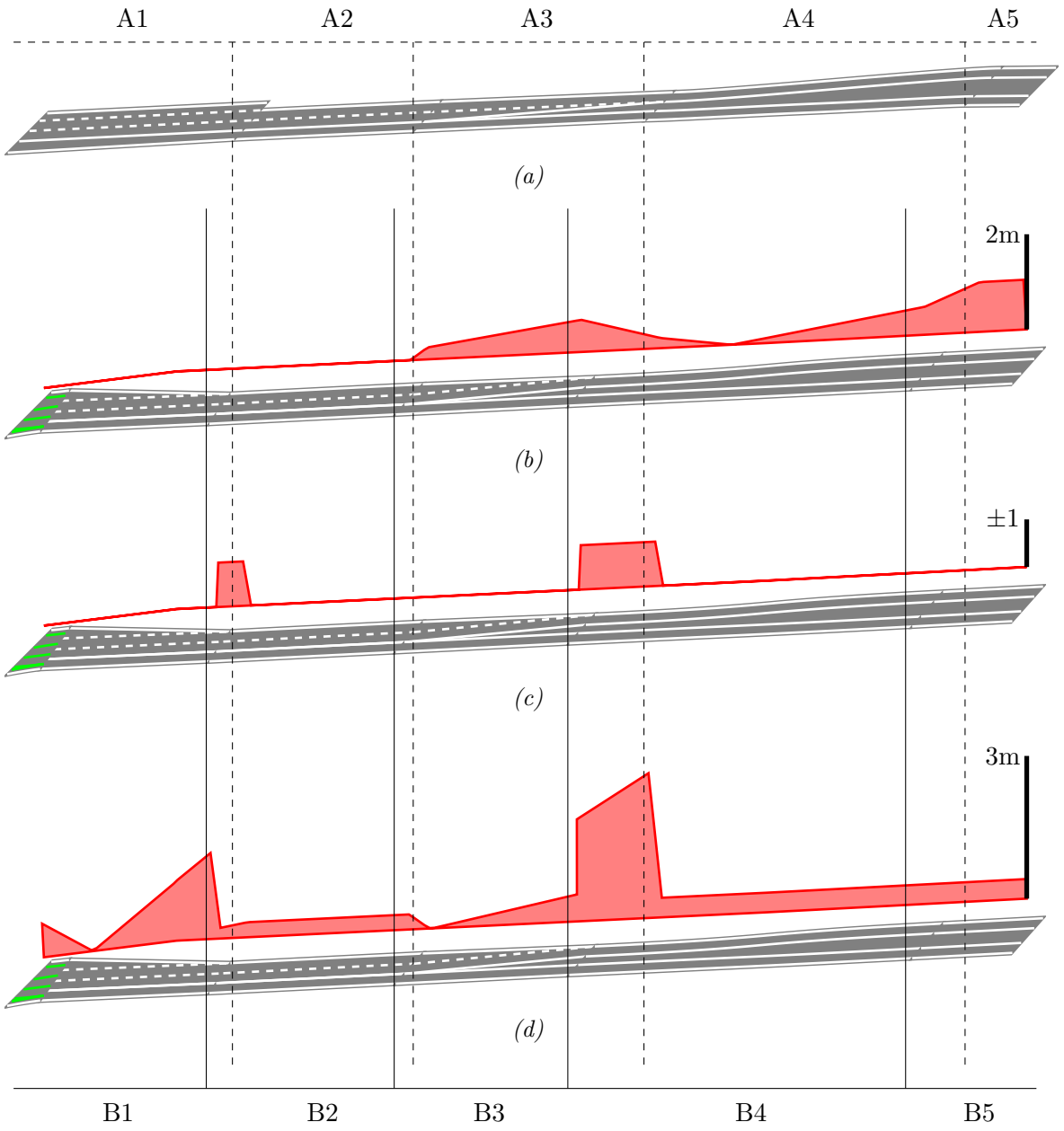


Abbildung 6.24.: Darstellung des Fehlers (rot) bezüglich der Eigenschaften bauliche Trennung (b), Spuranzahl (c) und Spurbreite (d) einer Straße der Rekonstruktion im Vergleich zur Referenzkarte (a). Zur Verdeutlichung sind die Blockgrenzen der Referenzkarte durch gestrichelte und jene der Rekonstruktion durch durchgezogene Linien gekennzeichnet. Zusätzlich sind die Blöcke der Referenz mit Ax und die der Rekonstruktion mit Bx benannt.

Bestimmung der Blockgrenze zurückzuführen. Hinsichtlich der Summe der Fehler in der Spurbreite treten Werte im Zentimeterbereich auf.

Im Block B3 ist das Verschwinden einer Spur und die damit einhergehende Überführung in eine bauliche Trennung korrekt ermittelt worden. Leichte Unterschiede in der baulichen Trennung und den Spurbreiten treten auf, wobei das bereits beschriebene Artefakt der longitudinalen Verschiebung

der Blockgrenzen ebenfalls Einfluss besitzt. Bezüglich der Trennung liegen die Fehlerwerte zu Beginn und Ende des Referenz Blocks A3 im niedrigen Zentimeterbereich. Aufgrund der verschobenen Grenze verschwindet die Fahrspur in der Rekonstruktion zu frühzeitig, wodurch ein Fehler im hohen Zentimeterbereich entsteht, welcher zum Ende von A3 absinkt. Dieser Übergangsbereich schlägt sich zwischen B3 und B4 bzw. A4 auch in den anderen Eigenschaften nieder. Eine kurzweilige Diskrepanz in der Spuranzahl und ein sehr großer Fehler den Spurbreiten sind ersichtlich. Letzteres entsteht durch die Zuordnung der Spuren. In der Rekonstruktion ist in diesem Bereich eine äußere nach links führende Fahrspur verzeichnet, während in A3 an dieser Stelle noch eine von zwei Fahrspuren verschwindet. Da die verschwindende Spur als innere Spur auf jene von B4 abgebildet wird, entsteht ein sehr großer Fehler, welcher mit der Fahrspur verschwindet. Somit ist dieser Fehler doppelt erkannt: einerseits als fehlende Fahrspur und andererseits als fehlende Spurbreite aufgrund der resultierenden Zuordnung. Daher kann der Spurbreitenfehler in diesem Zusammenhang vernachlässigt werden.

Weiterhin treten in B4 und B5 keine Fehler in der Spuranzahl, geringe Fehler in den Spurbreiten und in B5 Fehler im hohen Zentimeterbereich in der baulichen Trennung auf.

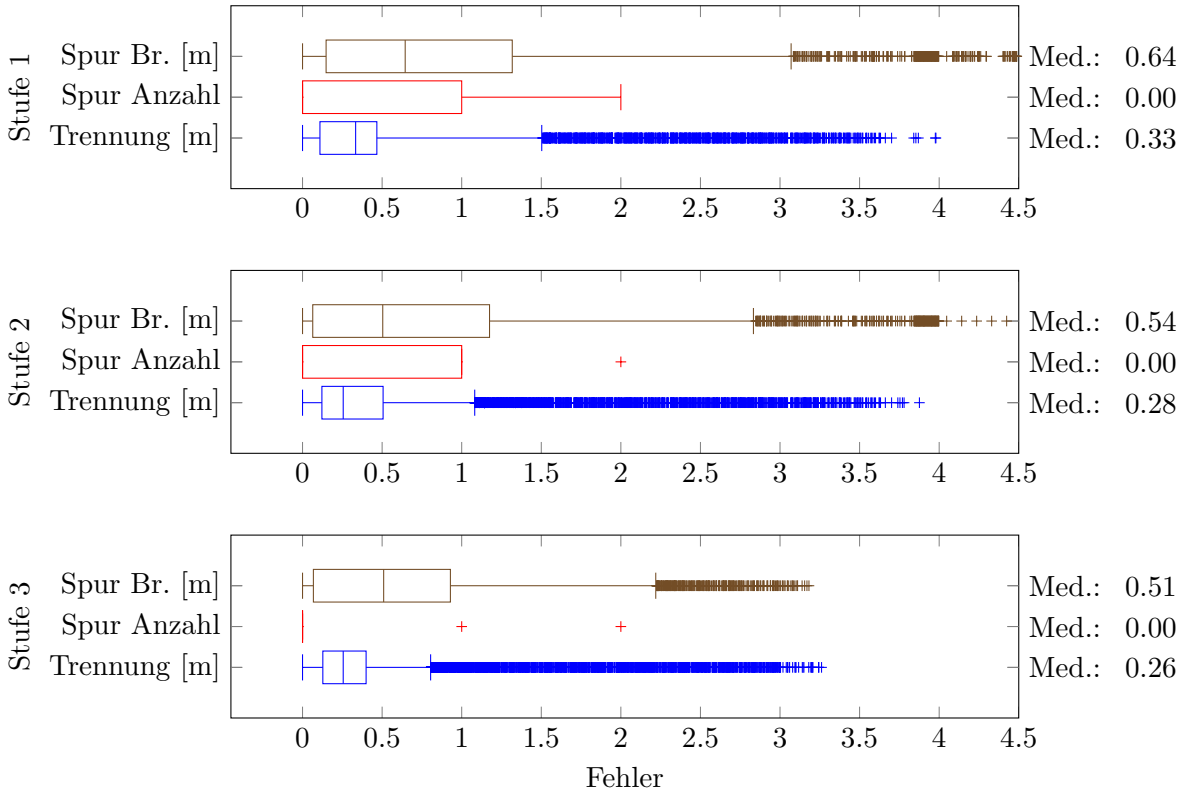


Abbildung 6.25.: Auswertung der in Abbildung 6.24 exemplarisch untersuchten Eigenschaften der Modelle auf dem gesamten Szenario in allen Qualitätsstufen der Eingabedaten.

Als nächstes werden die analysierten Fehlermaße der drei Eigenschaften auf dem gesamten Szenario und in allen Genauigkeitsstufen ausgewertet. Hierzu wird jede Straße in Abständen von einem Meter diskretisiert und die Eigenschaften im Querschnitt analysiert. Abbildung 6.25 stellt die Ergebnisse

als Boxplots[h] dar. Die Auswertungen der dritten Stufe weisen erwartungsgemäß die Dimensionen des in Abbildung 6.24 analysierten Beispiels auf. Bezüglich der baulichen Trennung liegen 75 % der Fehlerwerte unter 40 cm und bereits Werte über 80 cm werden relativ zur Gesamtheit als Ausreißer angesehen. Hinsichtlich der Fehlschätzungen in der Fahrspuranzahl wird ersichtlich, dass alle Kenngrößen des Boxplots auf 0 fallen. Dies bedeutet, dass jeder auftretende Fehler in Relation zur Menge der Messungen als Ausreißer gewertet wird und somit nahezu alle Rekonstruktionen korrekt sind. In der Rekonstruktion der Fahrspurbreiten liegen 75 % der Fehler unter 92 cm bei einem Median von 51 cm. Hierbei ist zu beachten, dass, wie in der Betrachtung der exemplarischen Straßen beschrieben, dieses Fehlermaß zur Überschätzung neigt, da es Fehler enthält, welche nicht aus einer falschen Schätzung, sondern einer falschen Spuranzahl resultieren.

In den Ergebnissen der ersten und zweiten Genauigkeitsstufe skalieren die Werte nach oben, da die in den Trajektorien inhärenten Fehler skalieren. Auffällig sind in beiden Auswertungen die Fehlschätzungen der Spuranzahl. Während in der dritten Stufe alle Fehler als Ausreißer gelten, sind es in der zweiten Stufe nur noch die groben Fehler und in der ersten Stufe keine mehr. Dabei liegen in beiden Stufen immer noch mindestens 50 % (Median) der Werte bei 0 und beschreiben somit eine korrekte Rekonstruktion.

Als nächstes wird eine exemplarische Straßenkreuzung genauer betrachtet. In Abbildung 6.26 ist eine Kreuzung mit allen topologischen Verknüpfungen und deren geometrischen Ausprägungen dargestellt. Als Bewertungsmaß ausgewertet ist dabei die Hausdorff-Distanz d_{HD} zwischen der Rekonstruktion und der Referenzkarte. Die Verbindungen sind gemäß der Bewertung nach dem dargestellten Farbverlauf eingefärbt. Zusätzlich ist in der Farbskala die maximale Bewertung durch einen schwarzen Strich markiert. Die abgebildete Kreuzung beinhaltet einige besondere Verkehrsführungen, welche entsprechende Herausforderungen für die Rekonstruktion darstellen. So existieren im oberen und unteren linken Bereich zwei Fahrspuren, welche nicht einfache Durchfahrten über die Fläche repräsentieren, sondern abseits der eigentlichen Kreuzung vorbeiführen. Trotzdem müssen diese Verbindungen topologisch erfasst und geometrisch abgebildet werden. Des Weiteren muss berücksichtigt werden, dass mehrere einspurig abbiegende Verbindungen existieren, welche mit mehreren ausführenden Spuren verbunden sind.

In der dargestellten Rekonstruktion wird ersichtlich, dass diese Herausforderungen topologisch korrekt und meistens geometrisch gut gelöst wurden und sowohl die besonderen Verkehrsführungen, als auch die Spurauswahl bei den Abbiegevorgängen abgebildet wurden. Die Kreuzung ist topologisch vollständig rekonstruiert, das heißt es wurde keine Verbindung fälschlicherweise erzeugt und es fehlt keine. Hinsichtlich der Hausdorff-Distanz liegen die meisten Auswertungen im Bereich $d_{HD} \leq 1\,\mathrm{m}$. Als sehr unproblematisch gestaltet sich die Rekonstruktion der geraden Verbindungen über die Kreuzungsfläche, daher weisen diese eine Bewertung im niedrigen Zentimeterbereich auf. Komplexer hingegen ist das Erzeugen des Fahrspurmodells einer Abbiegung. Das Verfahren ist in der Lage sehr individuelle Verläufe zu erzeugen. Beispielsweise beschreibt die Verbindung Nord nach West einen glatten und weiten Bogen von Beginn bis Ende, wobei die umgekehrte Verbindung zunächst

[h] Datenreferenz: Median: 50 %, untere Box: 25 %, obere Box: 75 %, Interquartilsabstand (IQR): Länge der Box als Wert, Abstand Whisker: maximal 1.5·IQR

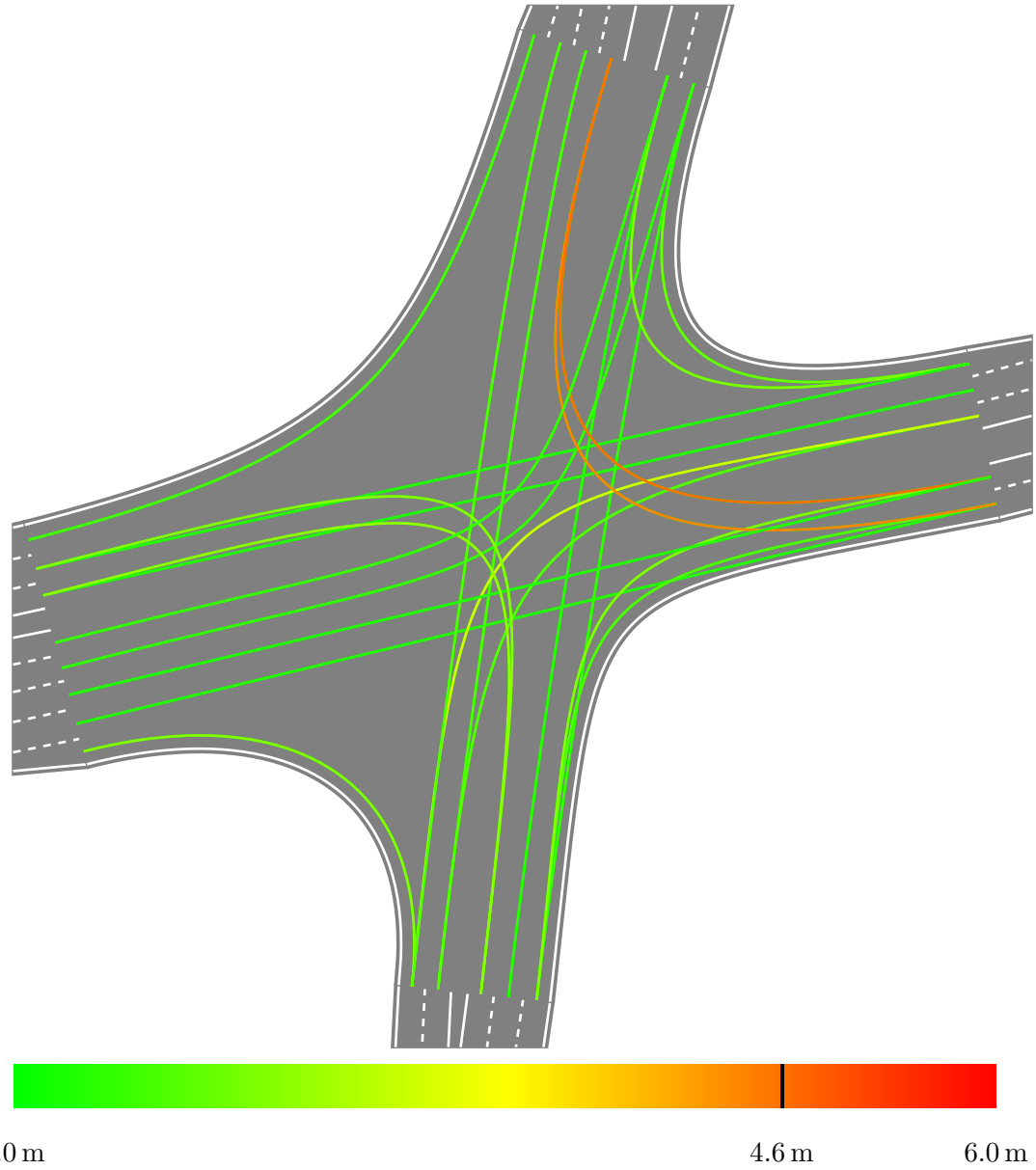


Abbildung 6.26.: Farbliche Visualisierung der Hausdorff-Distanz auf einer exemplarischen Kreuzung. Die schlechteste Bewertung ist auf der Farbskala mit einem schwarzen Strich markiert.

ein gerades Einfahren in den Kreuzungsbereich und anschließend einen engen kurzen Kurvenverlauf beschreibt. Ersichtlich wird ebenfalls, dass es in Abbiegeverläufen zu modellbedingten Grenzen kommt. Durch die Lage des Anschlussquerschnitts Nord müsste die Verbindung von Nord nach Ost zu Beginn eine leichte entgegengesetzte Krümmung aufweisen und somit eine Art S-Kurve beschreiben, um den Verlauf korrekt abzubilden und weist daher entsprechende Fehler auf.

Auf das gesamte Szenario bezogen gibt es 136 topologische Verbindungen zu rekonstruieren. In Tabelle 6.5 ist eine Übersicht über die korrekt rekonstruierten topologischen Verbindungen in allen Qualitätsstufen dargestellt. Zusätzlich ist in Abbildung 6.27 die Auswertung der Hausdorff-Distanz, wie bereits für eine exemplarische Kreuzung betrachtet, auf dem gesamten Szenario in allen Qualitätsstufen als Boxplots dargestellt.

Trajektorien	Rekonstruktionen	
Stufe 3	133	98 %
Stufe 2	127	93 %
Stufe 1	120	88 %

Tabelle 6.5.: Anzahl korrekt konstruierter topologischer Verbindungen aller Kreuzungen.

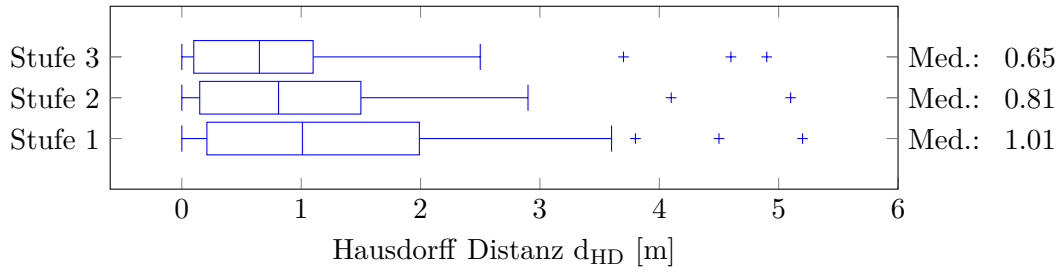


Abbildung 6.27.: Hausdorff-Distanz zwischen den geometrischen Ausprägungen der Verbindungen auf allen Kreuzungen in der Rekonstruktion und der Referenzkarte.

Naiver Ansatz

Im Folgenden werden die Ergebnisse des neuen Verfahrens mit jenen nach einem naiven Ansatz verglichen. Zunächst wird der Ablauf des naiven Ansatzes dargelegt und anschließend die Bewertungen aus Abschnitt 6.3 zur Evaluierung herangezogen.

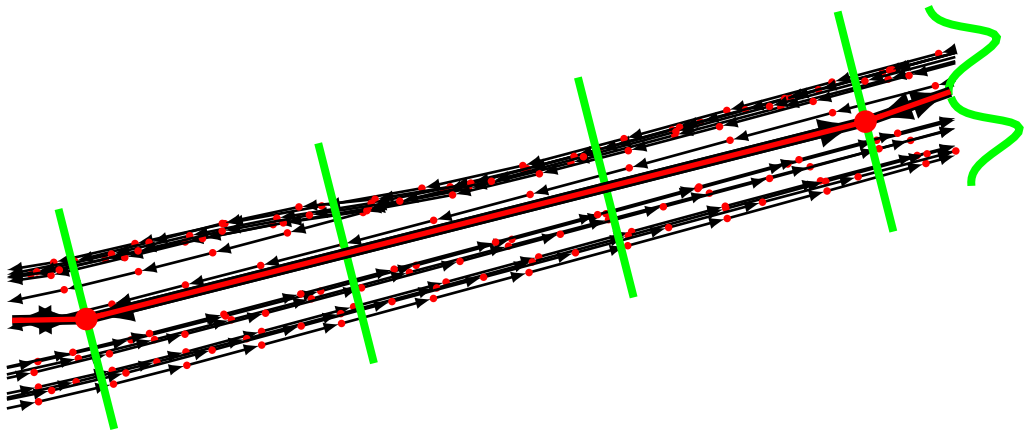


Abbildung 6.28.: Skizze zum Ablauf des naiven Verfahrens (grün), konstruiert auf einem Straßengraph (rot) und Fahrzeugtrajektorien (schwarze Pfeile, rote Punkte).

Zur Erzeugung einer fahrspurgenauen Straßenkarte nach dem naiven Ansatz dient das in (Uduwaragoda u. a., 2013) beschriebene Verfahren als Grundlage. Zunächst wird eine straßengenaue Repräsentation des Szenario in Straßen und Kreuzungen aufgeteilt. Anschließend wird jede Straße äquidistant senkrecht geschnitten und die Schnittpunkte dieser Lotrechten mit allen Trajektorien berechnet. Der beschriebene Vorgang ist in Abbildung 6.28 in grün skizziert. Auf jedem Schnitt wird nun eine Kerndichteschätzung mit Gaußkern durchgeführt, um die Spurmittellinien zu identifizieren. Eine beispielhafte Kerndichteschätzung für vier Fahrspuren ist in Abbildung 6.29 in Blau und die Schnittpunkte zwischen den Trajektorien und dem Lot in Rot dargestellt. Abschließend werden die

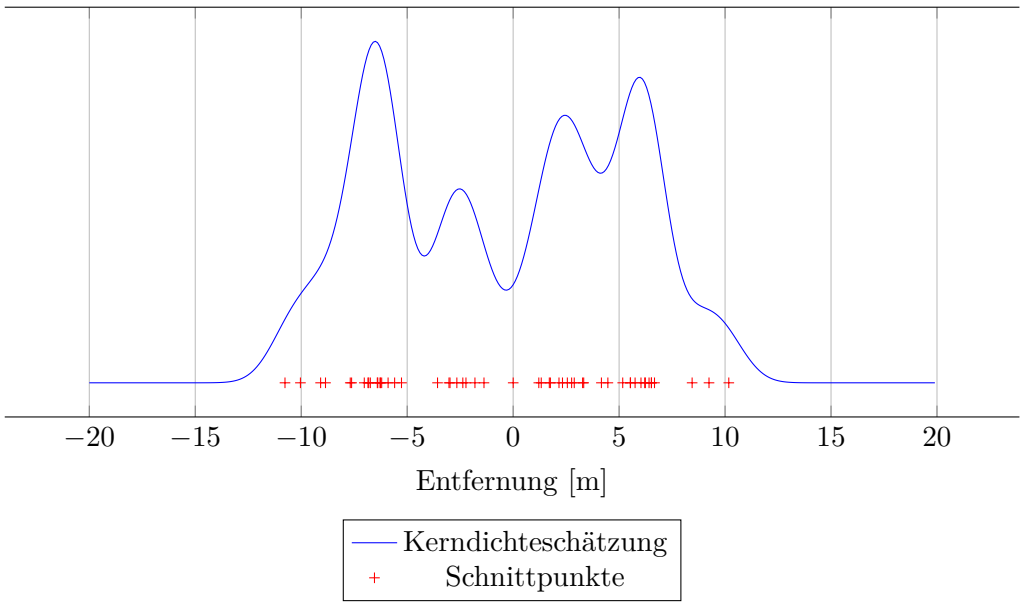


Abbildung 6.29.: Beispielhafte Kerndichteschätzung auf einem Querschnitt im naiven Ansatz.

benachbarten Schnitte entsprechend der Spurmittellinien verbunden, um eine spurgenaue Repräsentation zu erhalten. Ausgehend von jeder Kreuzung in der straßengenauen Karte wird auf jeder verbundenen Straße der Punkt bestimmt, bis zu welchem die Trajektorien parallel verlaufen und somit noch nicht auf der Kreuzungsfläche liegen. Innerhalb der Kreuzung werden die Trajektorien analysiert, um zu bestimmen welche Spuren über die Kreuzung direkt verbunden sind. Die Qualität der Ergebnisse dieser naiven Schätzung sind unmittelbar mit der Datenqualität korreliert. Sind die Spurmittellinien in den Daten eindeutig separiert liefert der naive Ansatz gute Ergebnisse. Allerdings ist das Verfahren nicht robust gegen Messfehler.

Ego-Trajektorien

Auf Grundlage der nachstehenden Ergebnisse sollen geometrische und topologische Schwachstellen bzw. Stärken identifiziert werden. Im Gegensatz zur Auswertung der straßengenauen Ergebnisse in Abschnitt 6.4.2, in welchen die Fehler auf fehlende oder falsch erzeugte Straßen bzw. deren Verläufe zurückzuführen waren, bedingen derartige Fehler im spurgenauen Kontext fehlende oder falsch erzeugte Spuren und deren geometrische Ausprägungen. Zur Auswertung werden die Trajektorien auf die Modelle abgebildet, um die entsprechenden Pfade durch die Graphen zu erhalten.

Die Abbildungen 6.30 und 6.31 beziehen sich auf den Längenvergleich der Pfade und stellen die Ergebnisse als Boxplots dar. Bezüglich der absoluten Längenunterschiede wird ersichtlich, dass bereits die Ergebnisse der ersten Stufe des neuen Verfahrens hinsichtlich des Medians alle Auswertungen des naiven Ansatzes übertreffen. In der dritten Stufe liegen 75 % der Vergleiche unter $d_{PL,a} \leq 1\,\mathrm{m}$. Bei beiden Verfahren treten in allen Qualitätsstufen (teilweise sehr große) Ausreißer auf. Dieses Verhalten ist beim naiven Ansatz auf ein häufiges Wechseln der Spuranzahl zurückzuführen, wenn die Messdaten keine eindeutige Kerndichteschätzung zulassen. Dies ist auch in den relativen Längenunterschieden ersichtlich, da dort beim naiven Verfahren ebenfalls große

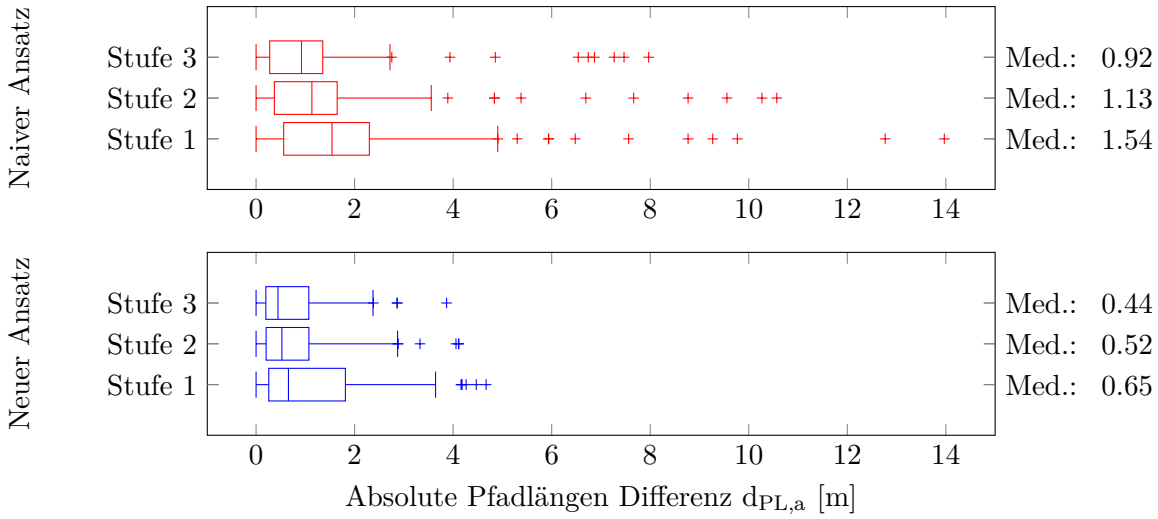


Abbildung 6.30.: Absolute Längenunterschiede zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen, basierend auf den Ego-Trajektorien. Die rechte Beschriftung beschreibt den Wert des jeweiligen Medians.

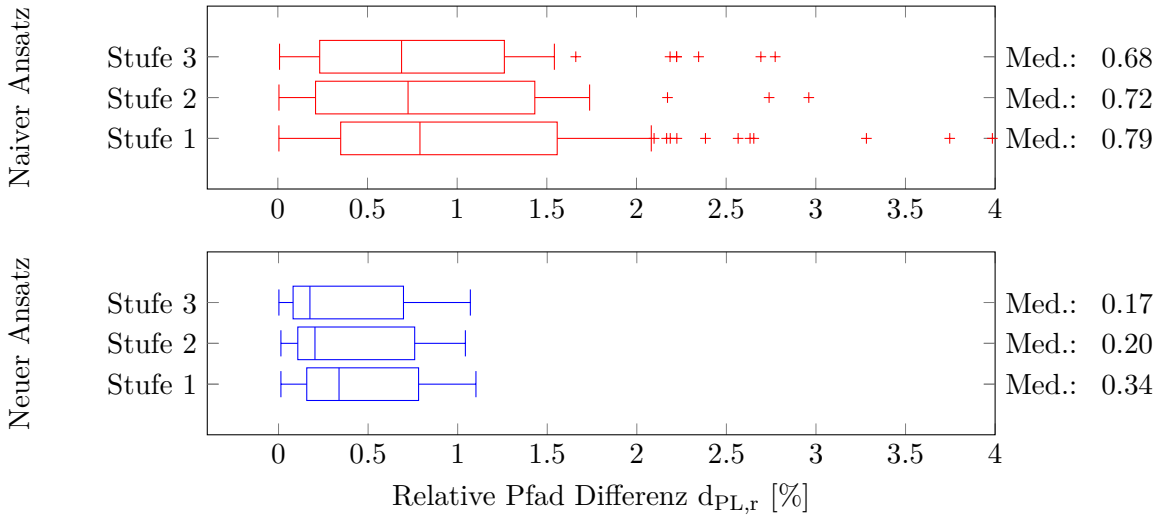


Abbildung 6.31.: Relative Längenunterschiede zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen, basierend auf den Ego-Trajektorien. Die rechte Beschriftung beschreibt den Wert des jeweiligen Medians.

Ausreißer auftreten, während beim neuen Ansatz keine ausgegeben sind. Daher sind die Ausreißer des absoluten Vergleichs beim neuen Ansatz nicht schwerwiegend, sondern skalieren mit der Länge der Trajektorie. In anderen Worten, je länger die Trajektorie, desto größer der akkumulierte Fehler, während beim naiven Ansatz größere punktuelle Mängel auftreten.

In Abbildung 6.32 ist die Auswertung der Hausdorff-Distanz dargestellt. Im Kontext des spurgenauen Modells lässt sich die Hausdorff-Distanz gut interpretieren, da sie den Punkt des größten Abstands zwischen den Pfaden beschreibt. Dieser Abstand kann entweder durch eine örtliche Diskrepanz zwischen korrekt erzeugten Spuren sein oder aufgrund einer falschen Schätzung der Spuranzahl entstanden sein. Wie in der Parameterstudie beschrieben ist eine Spurbreite zwischen

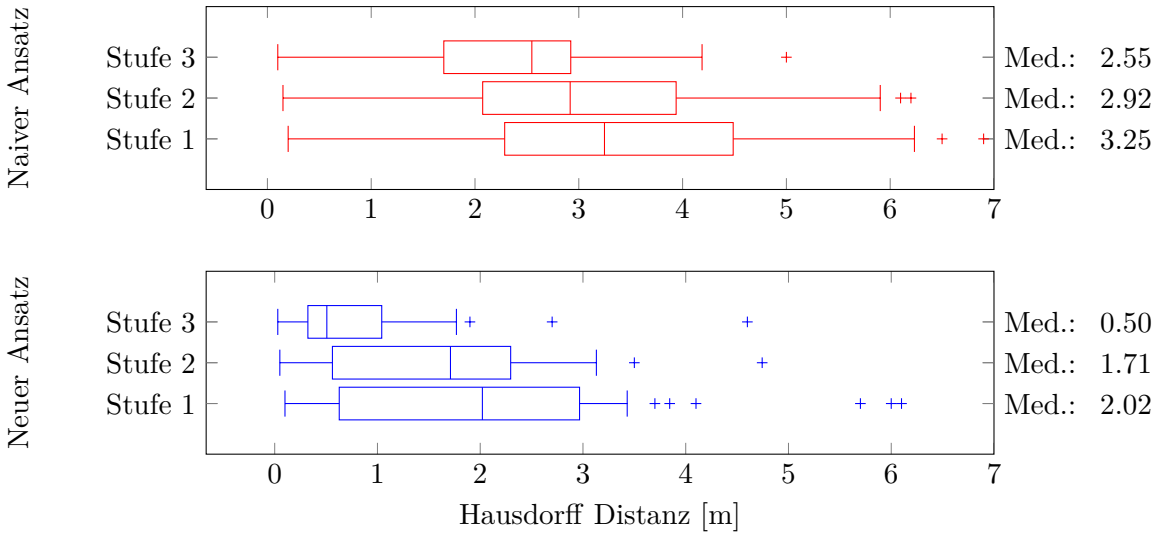


Abbildung 6.32.: Hausdorf-Distanz zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen.

2.25 m und 3.25 m zu erwarten. Somit deuten Ergebnisse in den Bereichen der Vielfachen dieser Werte auf falsche Spurschätzungen und sonst auf lokale Abweichungen hin. Wie bereits in Abbildung 6.25 ersichtlich kommt es beim neuen Verfahren in der ersten Stufe zu einfachen und vereinzelt zu zweifachen Fehlern in der Spurschätzung, daher sind entsprechende Dimensionen in dieser Auswertung der Hausdorff-Distanz erkennbar. Analoge Parallelen sind auch für die Stufen zwei und drei erkennbar. Andererseits liegen viele der Ergebnisse unterhalb dessen und stellen eine Abweichung der Rekonstruktion im teilweise niedrigen Zentimeterbereich und somit gute Ergebnisse dar. Bezüglich des naiven Ansatzes wird deutlich, dass in der ersten und zweiten Stufe sehr viele Auswertungen in Dimensionen einer einfachen und zweifachen Fehlschätzung der Spuranzahl in Kombination mit falschen Positionierung vorkommen. In der dritten Stufe konzentrieren sich die Fehler in der Dimension einer einfachen Fehlschätzung. Diese Ungenauigkeiten sind dabei auf die nicht eindeutig zu interpretierenden Kerndichteschätzungen zurückzuführen, denn diese rühren nicht stets von einer schlechten Positionierung her, sondern werden bereits durch das abgebildete normale Fahrverhalten beeinflusst.

Abschließend wird der F-Score betrachtet. Bei der Erzeugung der Ergebnisse wurden pro Auswertung 1500 Durchläufe mit einem Distanzwert von $\delta_F = 100\,\text{m}$ und einem longitudinalen Abstand von $\delta_D = 10\,\text{m}$ durchgeführt. Die Zuordnungsdistanz δ_M wird in jeder Auswertung variiert.

Die Ergebnisse des naiven Ansatzes sind in Abbildung 6.33 dargestellt. In den drei Qualitätsstufen konvergieren die Werte gegen 61 %, 64 % bzw. 65 %, bei einem durchschnittlichen Korridor von 11.4 %, 10.1 % bzw. 10.8 %. Die Dimensionen des F-Score implizieren bereits, dass in der Rekonstruktion Abweichungen von der Referenzkarte auftreten müssen. In allen drei Auswertungen liegt der Wert der Precision über dem Recall, daher sind mehr „Löcher“ als „Murmeln“ nicht zugeordnet. Dies bedeutet, dass es häufig zu wenig „Murmeln“ und somit einen Mangel in Form von fehlenden Fahrspuren in der Rekonstruktion gibt. Dieser Umstand wurde bereits durch die Hausdorff-Distanz

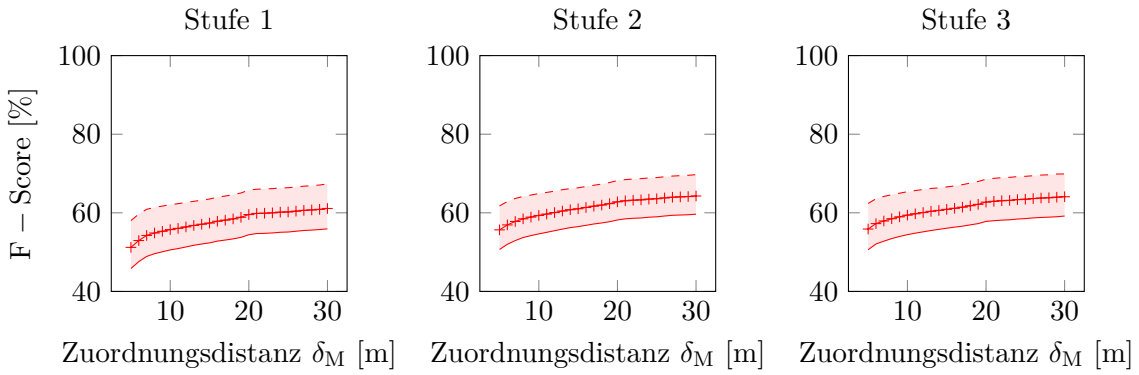


Abbildung 6.33.: Auswertung des F-Score auf den Ergebnissen nach dem naiven Ansatz in allen Trajektorien Qualitätsstufen. Legende: Der F-Score wird mit + Markierungen, der Recall das durchgezogene und die Precision als gestrichelte Linie dargestellt.

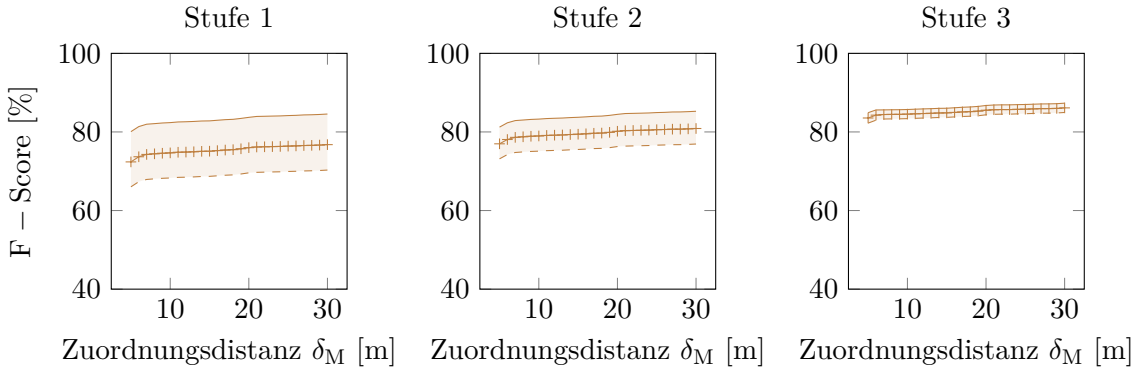


Abbildung 6.34.: Auswertung des F-Score auf den Ergebnissen nach dem neuen Ansatz in allen Trajektorien Qualitätsstufen. Legende: Der F-Score wird mit + Markierungen, der Recall das durchgezogene und die Precision als gestrichelte Linie dargestellt.

in allen Qualitätsstufen beschrieben. Die große Diskrepanz in den Korridoren impliziert dazu, dass, wenn Fehler auftreten diese häufig über eine große Spanne gültig sind.

Die Ergebnisse nach dem neuen Ansatz sind in Abbildung 6.34 dargestellt, wobei die Auswertungen in die drei Qualitätsstufen gegen die Werte 76 %, 81 % bzw. 86 % konvergieren. Die durchschnittlichen Korridore betragen dabei 14.1 %, 8.2 % bzw. 2.3 %. Im Vergleich zum naiven Ansatz werden deutlich höhere F-Score Werte erzielt, das bedeutet, dass eine insgesamt höhere Deckung zwischen der Referenzkarte und der Rekonstruktion vorliegt. Zusätzlich legt der Korridor in der ersten Qualitätsstufe ein ähnliches Verhalten wie beim naiven Ansatz nahe. Existierende Fehler sind nicht punktuell gravierend, sondern häufig über eine lange Spanne gültig. Da in diesen Auswertungen der Recall über der Precision liegt, neigt das neue Verfahren in der ersten Stufe zur Überschätzung der Realität. Dieses Verhalten wird durch eine höhere Eingabedatenqualität allerdings stark reduziert, sodass in der dritten Stufe ein hoher F-Score mit einem sehr schmalen Korridor vorliegt. Somit wird in der dritte Stufe eine Deckung von 86 % erreicht, wobei die Fehler zu gleichen Teilen aus der Über- und Unterschätzung der Realität stammen.

Insgesamt ist das neue Verfahren in der Lage eine präzise fahrspurgenaue Repräsentation eines Straßennetzes basierend auf Ego-Trajektorien zu erzeugen. Die Auswertungen belegen dabei ho-

he qualitative Ansprüche und einen deutlichen Mehrwert gegenüber einem naiven Ansatz. Die Auswertung und der Vergleich der Pfadlängen impliziert durch Median-Werte im Zentimeterbereich sowohl eine gute geometrische Darstellung, als auch eine topologische Vollständigkeit. Die Hausdorff-Distanz legt durch die Bewertung des naiven Verfahrens im Bereich von mehreren Metern dar, dass die Datensätze komplexe Situationen beinhalten, welche nicht auf einfach Weise bewältigt werden können. Somit untermauert diese Bewertung die geometrische Güte der Ergebnisse nach dem neuen Verfahren, da viele der Bewertungen zum Zentimeterbereich tendieren. Durch hohe absolute Werte in der Auswertung des F-Score wird die topologische Qualität betont, welche bezüglich einer spurgenauen digitalen Straßenkarte von großer Bedeutung ist.

Objekt-Trajektorien

Im Folgenden werden zur Erzeugung der fahrspurgenauen Straßenkarten die Objekt-Trajektorien hinzugezogen. Wie bereits bei der Generierung fahrbahngenauer Karten beschrieben, enthalten diese Trajektorien ausschließlich zu den Ego-Trajektorien redundante Informationen und können daher zur Steigerung der geometrischen Genauigkeit und topologischen Vollständigkeit beitragen. Um den Einfluss zu verdeutlichen, werden die absolute Pfadlängen-Differenz und die Hausdorff-Distanz ausgewertet und mit den vorherigen Ergebnissen verglichen.

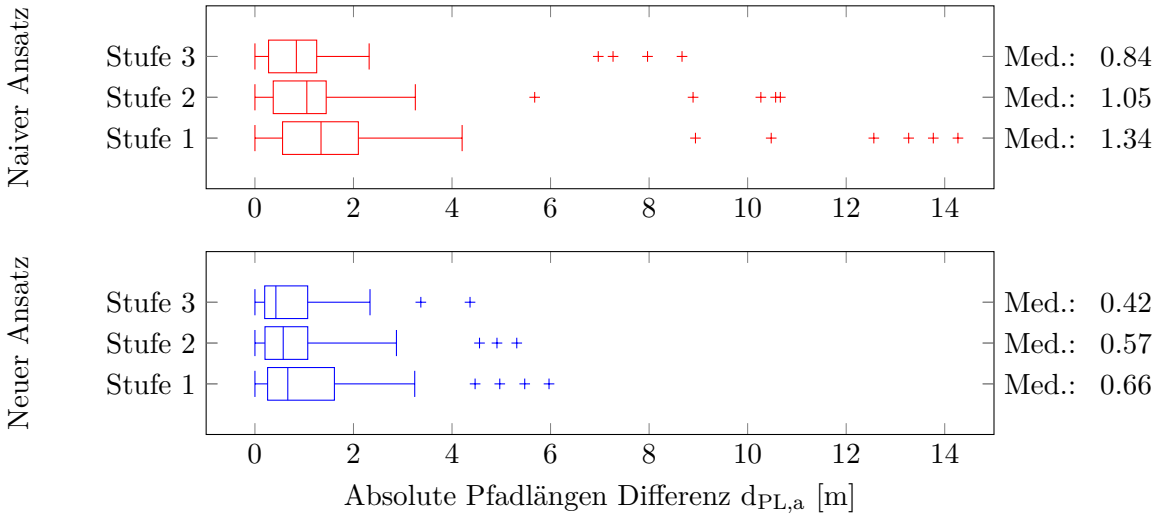


Abbildung 6.35.: Absolute Längenunterschiede zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen, basierend auf den Ego- und Objekt-Trajektorien. Die rechte Beschriftung beschreibt den Wert des jeweiligen Medians.

In Abbildung 6.35 sind die Auswertungen der absoluten Pfadlängen-Differenz dargestellt. Bezüglich des naiven Ansatzes wird ersichtlich, dass in allen Stufen die Werte leicht gesunken und gleichzeitig kompakter geworden sind. Dies deutet darauf hin, dass durch die Hinzunahme der redundanten Informationen die Kerndichteschätzungen eindeutiger geworden sind und sich die Schätzung der Spuranzahl weniger häufig ändert. Zusätzlich fällt auf, dass es weniger, dafür allerdings größere Ausreißer gibt, welche teilweise höhere Werte aufweisen als zuvor. Diese Ausreißer sind auf die in Abschnitt 6.1 beschriebenen groben Fehler in den Objekt-Trajektorien zurückzuführen. Derar-

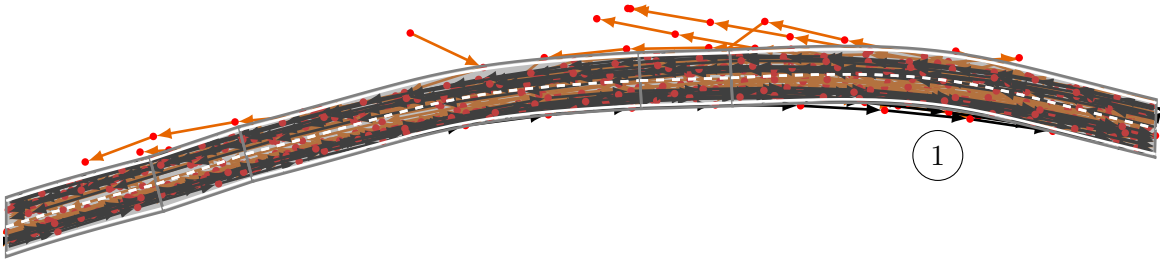


Abbildung 6.36.: Darstellung eines transparenten Straßenverlaufes (Rekonstruktion) überlagert mit den entsprechenden Ego- (schwarz) und Objekt-Trajektorien (orange).

tige Messungenauigkeiten können von dem naiven Ansatz nicht korrekt verarbeitet werden. Die Ergebnisse des neuen Ansatzes weisen ebenfalls leicht verbesserte Werte durch die redundanten Informationen, allerdings auch erheblich größere Ausreißer als vorher auf. Die beschriebenen Ungenauigkeiten, welche in den Objekt-Trajektorien vereinzelt verzeichnet sind, können vom Verfahren in den meisten Fällen korrekt gehandhabt, ein negativer Einfluss allerdings nicht immer vermieden werden. Dieser Einfluss wird in Abbildung 6.36 deutlich, welche einen transparenten Straßenverlauf, sowie die Ego-Trajektorien in Schwarz und die Objekt-Trajektorien in Orange zeigt. Es wird ersichtlich, dass das Verfahren in der Lage ist die Rekonstruktion auf den Trajektorienstrang zu konzentrieren und es vermeidet zusätzliche Fahrspuren auf den fehlerhaften Messungen auszubilden. Dennoch wird der Straßenverlauf verzerrt, solange die Messfehler nicht „eindeutig“ sind, wie bei der Markierung „1“ ersichtlich wird.

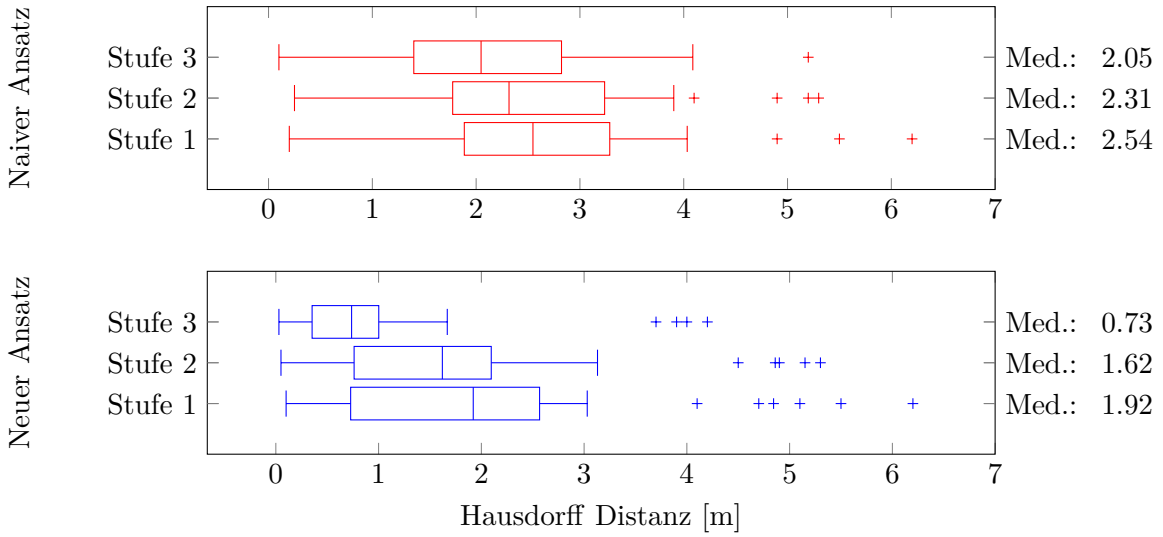


Abbildung 6.37.: Hausdorff-Distanz zwischen den Pfaden in den Rekonstruktionen und der Referenzkarte in allen Trajektorien Qualitätsstufen, basierend auf den Ego- und Objekt-Trajektorien. Die rechte Beschriftung beschreibt den Wert des jeweiligen Medians.

Die Auswertungen der Hausdorff-Distanz sind in Abbildung 6.37 dargestellt. Bezüglich des naiven Ansatzes sind erhebliche Verbesserungen in den Werten ersichtlich. Während in den Ergebnissen auf Basis der Ego-Trajektorien Messfehler im Bereich ein- bis zweifacher Spurbreite auftraten, reduzieren sich die neuen Auswertungen bereits in der ersten Qualitätsstufe auf die Dimension einer einfachen Spurfehlschätzung und lokalen Abweichungen. Hinsichtlich des neuen Ansatzes sind

geringe Verbesserungen in den Kennwerten des Boxplots erkennbar. Außerdem wurden, wie bereits in der Pfadlängen-Differenz beobachtet, die alten Ausreißer durch die redundanten Informationen korrigiert, jedoch neue in höheren Dimensionen durch starke Messfehler erzeugt.

Insgesamt profitiert das naive Verfahren stark von den redundanten Informationen, da sie eine eindeutigere Kerndichteschätzung zulassen. Der neue Ansatz kann durch die Objekt-Trajektorien ebenfalls einen Mehrwert erarbeiten und dabei in den meisten Fällen die inhärenten großen Messfehler detektieren und verarbeiten. Die kompakteren und niedrigeren Werte in den Auswertungen implizieren dabei eine gestiegene geometrische und topologische Qualität.

Versuche bezüglich der Trajektorienanzahl

Als nächstes wird untersucht, in wie weit die bisher analysierten Ergebnisse von der Menge an Eingabedaten abhängen. Dazu wird eine Straße des Szenario gesondert betrachtet. Diese hat eine Länge von 828 m, ist vierspurig und insgesamt durch 32 Ego-Trajektorien und 140 Objekt-Trajektorien abgedeckt. Jede Fahrspur ist dabei durch mindestens 3 Ego-Trajektorien durchgehend, das heißt ohne ein Spurwechselmanöver, befahren. Eine derartige Zuordnung der Objekt-Trajektorien ist nicht möglich, da die meisten Beobachtungen nicht durchgehend sind. Zur Untersuchung, wie viele Trajektorien nötig sind, um die bisher dokumentierten Ergebnisse zu erzielen, wird die betrachtete Straße im Folgenden mit eingeschränkten Datensätzen erzeugt und bewertet. Zunächst werden dabei ausschließlich die Ego-Trajektorien und im anschließenden Experiment zusätzlich die Objekt-Trajektorien berücksichtigt.

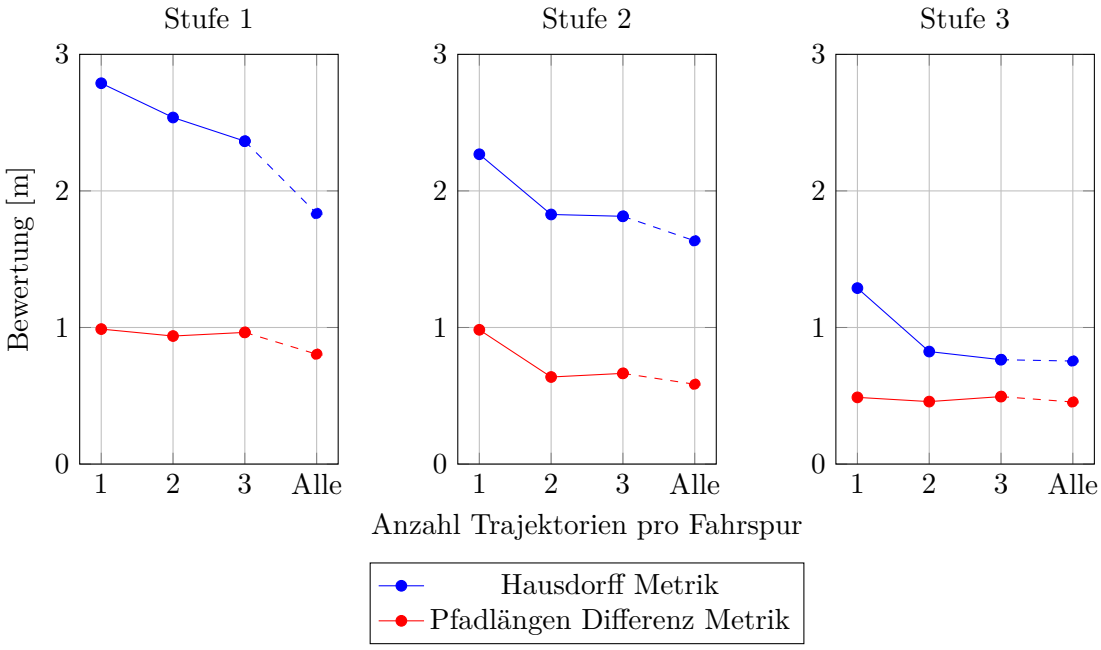


Abbildung 6.38.: Untersuchung der Hausdorff-Distanz (blau) und der Pfadlängen-Differenz (rot) in Abhängigkeit der Anzahl an Trajektorien pro Spur.

Die ersten drei Datensätze beinhalten jeweils eine, zwei bzw. drei Ego-Trajektorien pro Spur ohne Spurwechsel und der vierte Datensatz beinhaltet alle Daten. Ausgewertet wurden in jedem

Experiment die durchschnittliche Hausdorff-Distanz sowie die durchschnittliche absolute Pfadlängen-Differenz. Die Ergebnisse sind in Abbildung 6.38 dargestellt.

In der ersten Stufe wird ersichtlich, dass mit jeder hinzugekommenen Trajektorie die Bewertung bezüglich der Hausdorff-Distanz sinkt. Bedingt durch den größten Positionierungsfehler der Datensätze verbessert jede neue Information die Gesamtgüte. In dieser Auswertung reicht die begrenzte Anzahl von Trajektorien allerdings nicht aus, um die Ergebnisqualität wie basierend auf dem gesamten Datensatz zu erreichen. Da das betrachtete Szenario lediglich eine gerade Straße beschreibt, ist die Pfadlängen-Differenz weniger stark aussagekräftig. Dennoch lässt sie Rückschlüsse über die Schätzung der konstruierten Spuranzahl zu, da fehlerhafte Schätzungen längere Pfade bedingen. Da die Auswertung über alle Teildatensätze konstant bleibt, gibt es zumindest keine Veränderung in der Spuranzahl.

In der zweiten Stufe wird die Hausdorff-Distanz durch die Hinzunahme der zweiten Trajektorie verbessert, durch die dritte allerdings nicht. Zusätzlich liegt die Bewertung des gesamten Datensatzes nur knapp darunter. Dieses Verhalten kann durch den Datensatz bedingt sein und liefert nicht die notwendige Sicherheit, um eine Aussage über die benötigte Anzahl zu treffen, allerdings zeigt sich entsprechend dem geringeren Positionierungsfehler eine geringere Steigerung der Qualität. Bei Betrachtung der Pfadlängen-Differenz ist ein Sprung vom ersten zum zweiten Experiment ersichtlich, dies impliziert die Vermutung, dass durch die Hinzunahme einer zweiten Trajektorie die Schätzung der Spuranzahl positiv korrigiert werden konnte. Entsprechendes schlägt sich auch in der Hausdorff-Distanz nieder.

In der dritten Stufe zeigt sich durch die Hinzunahme der zweiten Trajektorie eine Verbesserung der Bewertung durch die Hausdorff-Distanz. Die Dimension bleibt bei der weiteren Vergrößerung des Datensatzes erhalten, somit kann die Aussage getroffen werden, dass bezüglich dieser Positionsgenauigkeit für ein derartiges Szenario eine zweifache Durchfahrung ausreichend für ein gutes Ergebnis ist. Zusätzlich impliziert die Pfadlängen-Differenz durch den konstanten Verlauf und die geringen Werte eine gute Schätzung der Spuranzahl.

Abschließend wird der Einfluss der Objekt-Trajektorien auf die benötigte Anzahl an Trajektorien untersucht. Dazu wird das vorherige Experiment erneut durchgeführt, jedoch werden zu jeder Ego-Trajektorie die entsprechenden Beobachtungen berücksichtigt. Diese sind nicht notwendiger Weise von der selben Fahrspur aufgezeichnet. Die Ergebnisse sind in Abbildung 6.39 dargestellt. Zusätzlich sind die Auswertungen basierend auf den Ego-Trajektorien verzeichnet.

Bezüglich der ersten Qualitätsstufe der Eingabedaten ist festzustellen, dass sich die Hinzunahme der Beobachtungen positiv auf die Hausdorff-Distanz auswirkt. Jedoch ändert sich nicht die Erkenntnis, dass die Qualität der Ergebnisse mit steigender Anzahl an Daten zunimmt. Hinsichtlich der zweiten Stufe ist ebenfalls ein positiver Einfluss auf die Hausdorff-Distanz erkennbar. Einen deutlichen Effekt hat die Hinzunahme der Objekt-Trajektorien auf die Pfadlängen-Differenz bei der Verwendung einer Ego-Trajektorie. Dies bedeutet, dass das Verfahren an dieser Stelle durch die Beobachtungen besser in der Lage ist die korrekte Spuranzahl abzuleiten. In der dritte Stufe zeigt sich keine Veränderung in der Pfadlängen-Differenz, jedoch eine starke Abweichung in der Hausdorff-Distanz bei der Verwendung einer Ego-Trajektorie. In diesem Fall kann die Aussage getroffen werden, dass

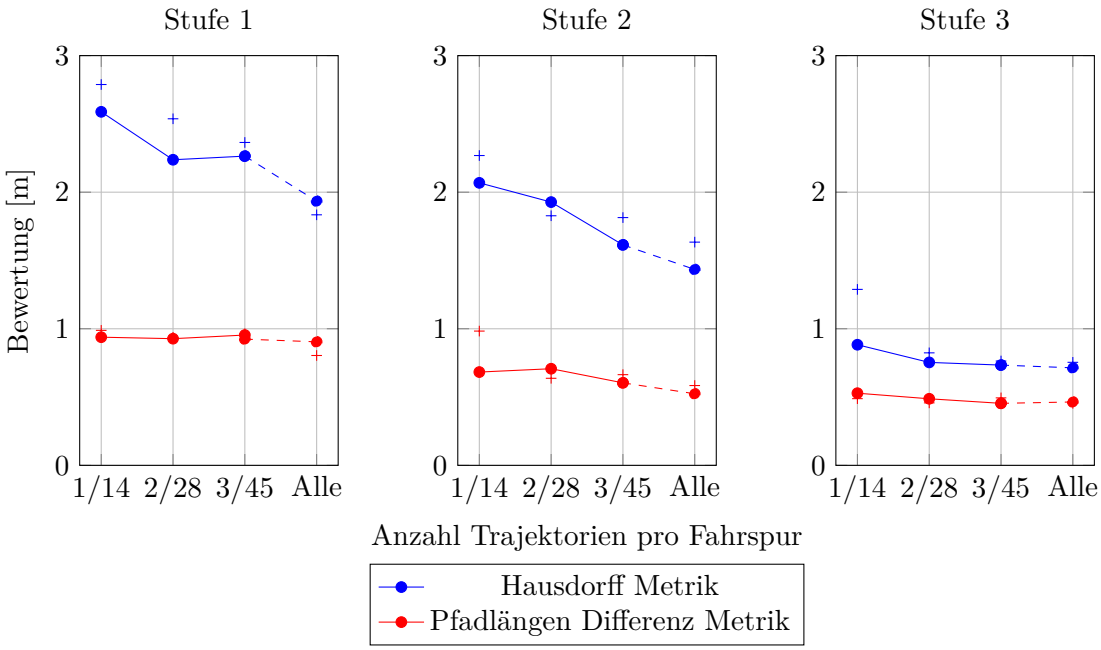


Abbildung 6.39.: Untersuchung der Hausdorff-Distanz (blaue Punkte) und der Pfadlängen-Differenz (rote Punkte) in Abhängigkeit der Anzahl an Ego- und Objekt-Trajektorien. Zusätzlich sind als Kreuze die Ergebnisse der Versuche ohne Objekt-Trajektorien (vgl. Abbildung 6.38) verzeichnet. Die Angaben der X-Achse beschreiben „Ego-Trajektorien pro Spur / deren Beobachtungen“.

ein qualitativ ebenbürtiges Ergebnis basierend auf vier Ego-Trajektorien (eine pro Spur), respektive 14 Objekt-Trajektorien erzielt werden kann wie basierend auf 32 Ego- und 140 Objekt-Trajektorien (ganzer Datensatz).

7. Zusammenfassung und Ausblick

Diese Arbeit befasst sich mit der Rekonstruktion hochgenauer Straßenkarten, welche im Kontext aktueller und zukünftiger Fahrerassistenzsysteme, sowie des automatisierten Fahrens immer mehr an Bedeutung gewinnt. Im Gegensatz zu heutzutage verfügbaren fahrbahngenauen Straßenkarten umfassen hochgenaue Karten einerseits präzisere Informationen über die Lage und den Verlauf eines Verkehrsnetzes und andererseits sehr viel weitreichendere Angaben über die Topologie und zentimetergenaue Geometrie. Eine große Herausforderung stellt dabei die Akquise derartigen Kartenmaterials dar. Derzeit werden speziell ausgerüstete Versuchsfahrzeuge verwendet, um Laser-, Radar-, Kamera- und Positionsdaten aufzuzeichnen, um daraus teil-automatisch und mit viel händischer Nachbearbeitung eine hochgenaue Straßenkarte zu extrahieren.

Zusammenfassung

Diese Arbeit beschreibt einen Schritt in Richtung der vollautomatischen Rekonstruktion derartiger Karten aus *Crowd Sourcing* Daten. Diese Daten können mit einem herkömmlichen, und heutzutage bereits in den meisten Fahrzeugen verfügbaren, Positionierungssystem aus einer beliebigen Fahrzeugflotte erzeugt werden und sind somit nahezu ohne Erfassungskosten zugänglich. Die dargelegten Algorithmen sind in der Lage die inhärenten Messungenauigkeiten aus der Menge an Daten zu extrahieren und eine spurgenaue Straßenkarte mit geringem Fehler zu generieren. In Kapitel 5 wurden verschiedene Modelle eingeführt, deren Parameter mit Hilfe eines Reversible Jump Markov Chain Monte Carlo Verfahrens auf Basis von Trainingsdaten angelernt werden. Die Eingabedaten liegen dabei einerseits in Form von Fahrzeug Ego-Trajektorien und andererseits in Form von beobachteten Objekt-Trajektorien vor. Wie in Abschnitt 6.1 beschrieben sind die Ego-Trajektorien dabei in drei verschiedenen Genauigkeitsstufen vermessen. Die Objekt-Trajektorien stammen von per Kamera detektierten und verfolgten anderen Verkehrsteilnehmern und weisen u.U. große Positionsfehler auf. Der Algorithmus besteht aus zwei Schritten: zunächst werden Modelle zur fahrbahngenauen Beschreibung genutzt, um von den Eingabedaten eine herkömmliche digitale Straßenkarte abzuleiten. Im zweiten Schritt werden, basierend auf dieser Darstellung fahrspurgenaue Modelle eingeführt, welche anschließend an die Eingabedaten angelernt werden. In Kapitel 6 wurden vier verschiedene Bewertungen eingeführt, welche sowohl zur Beurteilung der fahrbahngenauen als auch der fahrspurgenauen Ergebnisse verwendet wurden. Die Bewertungen berücksichtigen dabei sowohl die longitudinale als auch laterale geometrische Präzision, sowie den topologischen Zusammenhang der generierten Karte im Vergleich zu einer Referenzkarte, welche aufgrund von Laserscandaten erzeugt wurde. Die Ergebnisse des ersten Schrittes des Algorithmus wurden unter Verwendung der Bewertungen mit den Verfahren von Biagioni und Eriksson (2012) und Cao und Krumm (2009) verglichen. Untersucht wurden zunächst die Ergebnisse, welche basierend auf den

Ego-Trajektorien erzeugt werden konnten. Anschließend wurden die Objekt-Trajektorien hinzugenommen, um die Auswirkungen zu studieren. Bezüglich der Ego-Trajektorien konnte der neue Algorithmus die Qualität im Vergleich zum Stand der Forschung in allen Belangen übertreffen und sowohl hinsichtlich der geometrischen Präzision, als auch der topologischen Vollständigkeit neue Maßstäbe setzen. Die Hinzunahme der Objekt-Trajektorien bewirkte durch die Redundanz der Informationen an vielen Stellen eine zusätzliche Verbesserung, durch die teilweise großen inhärenten Fehler allerdings auch weniger eindeutige Lösungen. Die Ergebnisse des zweiten Schrittes des Algorithmus und Fokus der Arbeit wurden ebenfalls unter Verwendung der gleichen Bewertungen gegen einen naiven Ansatz zur Erzeugung einer spurgenauen Straßenkarte, basierend auf der Arbeit von Uduwaragoda u. a. (2013), verglichen. Auf diese Weise wurde der Mehrwert des Ansatzes gegenüber einer einfach und schnell zu erzeugenden Lösung dargelegt. Dabei konnte herausgestellt werden, dass die geometrische Präzision bezüglich der Spurbreiten und anderen baulichen Eigenschaften einer Straße bereits unter Verwendung des ungenausten Eingabedatensatzes alle Auswertungen des naiven Ansatzes übertreffen konnte. Gleiches gilt für die abgeleiteten topologischen Informationen, vor allem in Kreuzungsbereichen. Durch die Hinzunahme der Objekt-Trajektorien konnte der naive Ansatz durch die redundanten Informationen sehr viel besser arbeiten. Allerdings stellten sich auch neue Schwachstellen an jenen Stellen ein, wo die Objekt-Trajektorien ungenaue Messungen eingebracht haben. Der neue Ansatz konnte an vielen Stellen ebenfalls bei der geometrischen Genauigkeit profitieren und zusätzlich die meisten der schlechten Messungen ignorieren, da sie im Hinblick auf die Masse an Messungen wenig Gewicht hatten. Anschließend wurden Versuche bezüglich der benötigten Anzahl an Trajektorien durchgeführt. Dabei wurde ausgewertet, welche Qualität auf einer vierspurigen Straße erzielt werden konnte, indem zur Erzeugung pro Fahrspur eine bis drei Ego-Trajektorien ausgewählt wurden. Abschließend wurden zu diesen Fahrten die entsprechenden Objekt-Trajektorien hinzugenommen. Zusammenfassend lässt sich sagen, dass bei der Verwendung herkömmlicher GNSS Messungen die Qualität der Ergebnisse von der Menge an Eingabedaten abhängt. Mit jeder neuen Messung und redundanten Information konnte die Präzision gesteigert werden. Zusätzlich ergab sich die Erkenntnis, dass bei der Verwendung von hochgenauen Positionierungsdaten zwei Ego-Trajektorien bzw. eine Ego- und die zugehörigen Objekt-Trajektorien ausreichend waren, um die gleiche Qualität wie unter Verwendung des gesamten Datensatzes zu erzeugen.

Insgesamt sind die beschriebenen Algorithmen in der Lage geometrisch und topologisch sehr präzise und im Vergleich zum Stand der Forschung hervorragende Ergebnisse zu erzielen. Die Verfahren können Eingabedaten verschiedenen Ursprungs verwenden und bereits basierend auf Daten von heutzutage standardmäßig verwendeten GNSS Geräten arbeiten. Dadurch wird die Möglichkeit eröffnet, aus kostengünstigen Positionsdaten von Fahrzeugflotten Kartenmaterial für zukünftige teil- und vollautomatisch arbeitende Systeme zu erzeugen.

Ausblick

Für zukünftige Arbeiten sind diverse Erweiterungen denkbar. Zum einen können die vorhandenen Modelle flexibler gestaltet werden, um z. B. das in Abbildung 6.26 dargestellte Problem der S-Kurve behandeln zu können. Zum anderen kann der Modellkatalog erweitert werden, um dem System die einfachere Handhabung von Sonderfällen wie temporären Spurverengungen oder die Behandlung von gängigen, bisher nicht betrachtete Szenarien wie Kreisverkehren zu ermöglichen. Außerdem können mehr Sensordaten in die Modellbildung einbezogen werden. In dieser Arbeit war eine wiederkehrende, in Abbildung 6.24 beschriebene Schwachstelle die Schätzung der longitudinalen Blockgrenzen. Diese und darüber hinaus die Schätzung der Spurbreiten könnte verbessert werden, indem die verwendete Kamera zusätzlich die Fahrbahnmarkierungen detektieren würde. Auf diese Weise könnten jeder Spur direkt die geometrischen Begrenzungen und darüber hinaus die Art der Fahrbahnmarkierungen zugeordnet werden. Basierend darauf ließen sich die longitudinalen Begrenzungen der Blöcke eindeutiger bestimmen.

Im Rahmen dieser Arbeit lag der Fokus auf der Entwicklung der dargelegten Methodiken, während die Optimierung der Laufzeit nicht betrachtet wurde. Diesbezüglich von Bedeutung sind Faktoren wie die Komplexität und die Skalierbarkeit des Systems. Da die individuellen Schritte der Verfahren bezüglich der Komplexität gering sind, allerdings sehr häufig ausgeführt werden, liegt das größte Optimierungspotential in der Parallelisierung der Schritte. Dazu müsste es ermöglicht werden beispielsweise die Auswertung der Bewertungsfunktion parallel auf verschiedenen Unterszenarien durchzuführen. Sowohl in der fahrbahn- als auch in der fahrspurgenauen Rekonstruktion könnten die Eingabedaten auf die Modelle verteilt und die Bewertungen pro Modell unabhängig durchgeführt werden. Die Anzahl an parallelen Prozessen würde somit mit der Anzahl der Modelle, respektive mit der Größe des Szenarios skalieren. Entsprechende Maßnahmen sind für alle Schritte der Algorithmen umsetzbar.

Literaturverzeichnis

Aarts, E. H., Korst, J. H., 1989. Boltzmann machines for travelling salesman problems. *European Journal of Operational Research* 39 (1), S. 79–95.

Ahmed, M., Karagiorgou, S., Pfoser, D., Wenk, C., 2015. A comparison and evaluation of map construction algorithms using vehicle tracking data. *GeoInformatica* 19 (3), S. 601–632.

Ahmed, M., Wenk, C., 2012. Constructing Street Networks from GPS Trajectories. In: *Proceedings of the 20th Annual European Symposium on Algorithms*. Ljubljana, Slovenia, S. 60–71.

Andrieu, C., de Freitas, N., Doucet, A., 2000. Reversible Jump MCMC Simulated Annealing for Neural Networks. In: *Uncertainty in Artificial Intelligence (UAI)*. Morgan Kaufmann, Burlington, S. 11–18.

Andrieu, C., De Freitas, N., Doucet, A., Jordan, M. I., 2003. An introduction to MCMC for machine learning. *Machine Learning* 50 (1-2), S. 5–43.

Baier, R., 2006. Richtlinien für die Anlage von Stadtstraßen (RASt 06). Forschungsgesellschaft für Straßen- und Verkehrswesen e.V., Köln.

Beniger, J. R., Barnett, V., Lewis, T., 1980. Outliers in Statistical Data. *Contemporary Sociology* 9 (4), S. 560–561.

Bentley, Ottmann, 1979. Algorithms for Reporting and Counting Geometric Intersections. *IEEE Transactions on Computers* C-28 (9), S. 643–647.

Biagioni, J., Eriksson, J., 2012. Map Inference in the Face of Noise and Disparity. In: *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. ACM, Redondo Beach, California, S. 79–88.

Bishop, C. M., 2006. Pattern Recognition and Machine Learning. Springer, New York.

Brüntrup, R., Edelkamp, S., Jabbar, S., Scholz, B., 2005. Incremental Map Generation with GPS Traces. In: *Proceedings of the 8th Conference on Intelligent Transportation Systems*. IEEE, Vienna, Austria, S. 413–418.

Buchin, K., Buchin, M., Wang, Y., 2009. Exact Algorithms for Partial Curve Matching via the Frechet Distance. In: *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms*. ACM, New York, S. 645–654.

Cao, L., Krumm, J., 2009. From GPS Traces to a Routable Road Map. In: *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, Seattle, Washington, S. 3–12.

Chen, A., Ramanandan, A., Farrell, J. A., 2010. High-precision lane-level road map building for vehicle navigation. In: *Proceedings of the Position Location and Navigation Symposium*. IEEE, S. 1035–1042.

Chen, C., Cheng, Y., 2008. Roads Digital Map Generation with Multi-track GPS Data. In: *Proceedings of the International Workshop on Education Technology and Training & International Workshop on Geoscience and Remote Sensing*. IEEE, Shanghai, China, S. 508–511.

Chen, Y., Krumm, J., 2010. Probabilistic Modeling of Traffic Lanes from GPS Traces. In: *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, San Jose, California, S. 81–88.

Chib, S., Greenberg, E., 1995. Understanding the Metropolis-Hastings Algorithm. *The American Statistician* 49 (4), S. 327–335.

Cramer, H., Scheunert, U., Wanielik, G., 2004. A New Approach for Tracking Lanes by Fusing Image Measurements with Map Data. In: *Proceedings of the Intelligent Vehicles Symposium*. IEEE, Parma, Italy, S. 607–612.

Davies, J., Beresford, A., Hopper, A., 2006. Scalable, Distributed, Real-Time Map Generation. *Pervasive Computing* 5 (4), S. 47–54.

Defense Advanced Research Projects Agency, 2018. Transit Satellite: Space-based Navigation.

Dempster, A. P., Laird, N. M., Rubin, D. B., 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society* 39 (1), S. 1–38.

Devroye, L., 1987. A Course in Density Estimation. Birkhauser, Boston.

Dickmanns, E. D., Mysliwetz, B. D., 1992. Recursive 3-D Road and Relative Ego-State Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (2), S. 199–213.

Dickmanns, E. D., Zapp, A., 1987. A Curvature-based Scheme for Improving Road Vehicle Guidance by Computer Vision.

Dijkstra, E. W., 1959. A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik* 1 (1), S. 269–271.

Douglas, D., Peucker, T., 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization* 10, S. 112–122.

Duda, R. O., Hart, P. E., 1973. Pattern Classification and Scene Analysis. Wiley, Hoboken.

Edelkamp, S., Schrödl, S., 2003. Route Planning and Map Inference with Global Positioning Traces. In: *Computer Science in Perspective: Essays Dedicated to Thomas Ottmann*. Springer, Berlin Heidelberg, S. 128–151.

Ester, M., Kriegel, H. P., Sander, J., Xu, X., 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*. AAAI, Menlo Park, S. 226–231.

Fathi, A., Krumm, J., 2010. Detecting Road Intersections from GPS Traces. In: *Proceedings of the 6th International Conference on Geographic Information Science*. Springer, S. 56–69.

Feller, W., 1968. An Introduction to Probability Theory and Its Applications, 3. Auflage. Bd. 1. Wiley, Hoboken.

Gilks, W. R., Richardson, S., Spiegelhalter, D. J., 1996. Markov Chain Monte Carlo in Practice. Chapman & Hall, London.

Goldbeck, J., Huertgen, B., 1999. Lane Detection and Tracking by Video Sensors. In: *Proceedings of the 2nd Conference on Intelligent Transportation Systems*. IEEE, Tokyo, Japan, S. 74–79.

Green, P. J., 1995. Reversible jump Markov chain monte carlo computation and Bayesian model determination. *Biometrika* 82 (4), S. 711–732.

Guo, T., Iwamura, K., Koga, M., 2007. Towards High Accuracy Road Maps Generation from Massive GPS Traces Data. In: *Proceedings of the International Geoscience and Remote Sensing Symposium*. IEEE, Barcelona, Spain, S. 667–670.

Hand, D. J., McLachlan, G. J., Basford, K. E., 1989. Mixture Models: Inference and Applications to Clustering. *Applied Statistics* 38 (2), S. 384.

Hastings, W. K., 1970. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* 57 (1), S. 97–109.

Heuser, H., 2004. Lehrbuch der Analysis, Bd. 2. Vieweg+Teubner, Wiesbaden.

Hofmann-Wellenhof, B., Lichtenegger, H., Wasle, E., 2007. GNSS – Global Navigation Satellite Systems - GPS, GLONASS, Galileo, and more.

Huang, A. S., Moore, D., Antone, M., Olson, E., Teller, S., 2009. Finding Multiple Lanes in Urban Road Networks with Vision and Lidar. *Autonomous Robots* 26 (2-3), S. 103–122.

Huertgen, B., Poechmueller, W., Stiller, C., Heiner, A., Roessig, C., Goldbeck, J., 1999. Vehicle Environment Sensing by Video Sensors. In: *Proceedings of the International Congress and Exposition*. SAE International, Detroit, S. 1–6.

Jang, S., Kim, T., Lee, E., 2010. Map Generation System with Lightweight GPS Trace Data. In: *Proceedings of the 12th International Conference on Advanced Communication Technology*. IEEE, Phoenix Park, South Korea, S. 1489–1493.

Kaliyaperumal, K., Lakshmanan, S., Kluge, K., 2001. An Algorithm for Detecting Roads and Obstacles in Radar Images. *IEEE Transactions on Vehicular Technology* 50 (1), S. 170–182.

Kang, D. J., Jung, M. H., 2003. Road lane segmentation using dynamic programming for active safety vehicles. *Pattern Recognition Letters* 24 (16), S. 3177–3185.

Karagiorgou, S., Pfoser, D., 2012. On Vehicle Tracking Data-Based Road Network Generation. In: *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*. ACM, Redondo Beach, California, S. 89.

Kaufman, L., Rousseeuw, P. J., 1990. Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, Hoboken.

Kelly, F. P., 2011. Reversibility and Stochastic Networks. Cambridge University Press, New York.

Kim, Z., 2008. Robust Lane Detection and Tracking in Challenging Scenarios. *IEEE Transactions on Intelligent Transportation Systems* 9 (1), S. 16–26.

Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., 1983. Optimization by Simulated Annealing. *Science* 220 (4598), S. 671–680.

Klotz, A., Sparbert, J., Hotzer, D., 2004. Lane data fusion for driver assistance systems. In: *Proceedings of the 7th International Conference on Information Fusion*. S. 657–663.

Lauer, M., 2004. Entwicklung eines Monte-Carlo-Verfahrens zum selbständigen Lernen von Gauss-Mischverteilungen. Phd-thesis, Universität Osnabrück.

Lee, D. T., Schachter, B. J., 1980. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer & Information Sciences* 9 (3), S. 219–242.

Liu, X., Biagioni, J., Eriksson, J., Wang, Y., Forman, G., Zhu, Y., 2012. Mining Large-Scale, Sparse GPS Traces for Map Inference: Comparison of Approaches. In: *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Beijing, China, S. 669–677.

McLachlan, G., Peel, D., 2000. Finite Mixture Models. Wiley, Hoboken.

Mélo, J., Naftel, A., Bernardino, A., Santos-Victor, J., 2006. Detection and Classification of Highway Lanes Using Vehicle Motion Trajectories. *IEEE Transactions on Intelligent Transportation Systems* 7 (2), S. 188–199.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., Teller, E., 1953. Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics* 21 (6), S. 1087–1092.

Mueller-Gronbach, T., Novak, E., Ritter, K., 2012. Monte Carlo-Algorithmen. Springer, Berlin, Heidelberg.

Naccache, N. J., Shinghal, R., 1984. SPTA: A Proposed Algorithm for Thinning Binary Patterns. *IEEE Transactions on Systems, Man and Cybernetics* SMC-14 (3), S. 409–418.

Nadler, S., 1978. Hyperspaces of sets: a text with research questions. M. Dekker, New York.

Newson, P., Krumm, J., 2009. Hidden Markov Map Matching Through Noise and Sparseness. In: *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, Seattle, Washington, S. 336–343.

Pahl, P. J., Damrath, R., 2000. Mathematische Grundlagen der Ingenieurinformatik, 1. Auflage. Springer, Berlin, Heidelberg.

Parzen, E., 1962. On Estimation of a Probability Density Function and Mode. *Annals of Mathematical Statistics* 33 (3), S. 1065–1076.

Piegl, L., Tiller, W., 1996. The NURBS Book, 2. Auflage. Springer, Berlin, Heidelberg.

Plehn, T., 2014. Markov Chain Monte Carlo - Methoden: Herleitung, Beweis und Implementierung. Bachelor + Master Publishing.

Robert, C. P., Casella, G., 2004. Monte Carlo Statistical Methods. Springer, New York.

Roeth, O., Zaum, D., Brenner, C., 2015. Improving GPS-based Road Network Constructions in a Post-Processing Step of Global Optimization. In: *Proceedings of the 14th International Conference on ITS Telecommunications*. IEEE, Copenhagen, Denmark, S. 70–74.

Roeth, O., Zaum, D., Brenner, C., 2016. Road Network Reconstruction using Reversible Jump MCMC Simulated Annealing based on Vehicle Trajectories from Fleet Measurements. In: *Proceedings of the Intelligent Vehicles Symposium*. IEEE, Gothenburg, Sweden, S. 194–201.

Roeth, O., Zaum, D., Brenner, C., 2017. Extracting Lane Geometry and Topology Information from Vehicle Fleet Trajectories in Complex Urban Szenario using a Reversible Jump MCMC Method. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, S. 51–58.

Rogers, S., Langley, P., Wilson, C., 1999. Mining GPS Data to Augment Road Models. In: *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, San Diego, California, S. 104–113.

Schmeiser, B., Devroye, L., 1988. Non-Uniform Random Variate Generation. Springer, New York.

Schrödl, S., Wagstaff, K., Rogers, S., Langley, P., Wilson, C., 2004. Mining GPS Traces for Map Refinement. *Data Mining and Knowledge Discovery* 9 (1), S. 59–87.

Shakarji, C., 1998. Least-Squares Fitting Algorithms of the NIST Algorithm Testing System. *Journal of Research of the National Institute of Standards and Technology* 103 (6), S. 633.

Shi, W., Shen, S., Liu, Y., 2009. Automatic Generation of Road Network Map from Massive GPS Vehicle Trajectories. In: *Proceedings of the 12th International Conference on Intelligent Transportation Systems*. IEEE, St. Louis, Missouri, S. 48–53.

Silverman, B., 1986. Density Estimation for Statistics and Data Analysis. Chapman & Hall, London.

Sparbert, J., Dietmayer, K., Streller, D., 2001. Lane Detection and Street Type Classification Using Laser Range Images. In: *Proceedings of the Intelligent Transportation Systems*. IEEE, Oakland, California, S. 454–459.

Stahel, W. A., 2002. Statistische Datenanalyse: Eine Einführung für Naturwissenschaftler, 4. Auflage. Vieweg+Teubner, Wiesbaden.

Steiner, A., Leonhardt, A., 2011. A Map Generation Algorithm using Low Frequency Vehicle Position Data. In: *Proceedings of the Transportation Research Board 90th Annual Meeting*. Washington DC, S. 1–17.

Titterington, D. M., Smith, A. F. M., Makov, U. E., 1985. Statistical Analysis of Finite Mixture Distributions. Wiley, Hoboken.

Tournaire, O., Brédif, M., Boldo, D., Durupt, M., 2010. An efficient stochastic approach for building footprint extraction from digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing* 65 (4), S. 317–327.

Uduwaragoda, E. R. I. A. C. M., Perera, A. S., Dias, S. A. D., 2013. Generating Lane Level Road Data from Vehicle Trajectories Using Kernel Density Estimation. In: *Proceedings of the 16th Conference on Intelligent Transportation Systems*. IEEE, The Hague, Netherlands, S. 384–391.

Vapnik, V. N., 1995. The Nature of Statistical Learning Theory, 2. Auflage. Springer, New York.

Viterbi, A. J., 1967. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory* 13 (2), S. 260–269.

Wang, Y., Liu, X., Wei, H., Forman, G., Chen, C., Zhu, Y., 2013. CrowdAtlas: Self-Updating Maps for Cloud and Personal Use. In: *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, Taipei, Taiwan, S. 27–40.

Weigel, H., Cramer, H., Wanielik, G., Polychronopoulos, A., Saroldi, A., 2006. Accurate Road Geometry Estimation for a Safe Speed Application. In: *Proceedings of the Intelligent Vehicles Symposium*. IEEE, Tokyo, Japan, S. 516–521.

Wilson, C., Rogers, S., Weisenburger, S., 1998. The Potential of Precision Maps in Intelligent Vehicles. In: *Proceedings of the International Conference on Intelligent Vehicles*. IEEE, Piscataway, New Jersey, S. 419–422.

Worrall, S., Nebot, E., 2007. Automated Process for Generating Digitised Maps through GPS Data Compression. In: *Proceedings of the Australasian Conference on Robotics & Automation*. ACRA, Brisbane.

Yokoi, S., Toriwaki, J.-i., Fukumura, T., 1975. An Analysis of Topological Properties of Digitized Binary Pictures Using Local Features. *Computer Graphics and Image Processing* 4 (1), S. 63–73.

Zhang, L., Thiemann, F., Sester, M., 2010. Integration of GPS Traces with Road Map. In: *Proceedings of the 2nd International Workshop on Computational Transportation Science*. ACM, San Jose, California, S. 17–22.

Zhang, T. Y., Suen, C. Y., 1984. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM* 27 (3), S. 236–239.

B. Mathematische Definitionen

B.1. Notationen

Konventionen

Folgende Konventionen sind im Rahmen dieser Arbeit gültig, sofern sie bei der Verwendung nicht anders beschrieben sind.

Bezeichner:

d	Dimension eines Datensatzes
n	Größe eines Datensatzes
i,j	Indexvariablen
p,q	Dichten von Verteilungen

Mathematische Konventionen:

A,B,C,...	Großbuchstaben → Mengen, Matrizen (Kontextbezogen)
a,b,c,...	Kleinbuchstaben → Skalare, Vektoren (Kontextbezogen)
a_i	Indizierte Kleinbuchstaben → Mengeneintrag
a_{ij}	Doppelt indizierte Kleinbuchstaben → Matrixeintrag

Schreibweisen

Die folgende Auflistung beinhaltet verwendete Formelsymbole und beispielhafte Schreibweisen, sofern sie bei der Verwendung nicht anders beschrieben sind.

$X \sim V$	Die Zufallsvariable X ist gemäß der Verteilung V verteilt
$\mathbb{R}$	reelle Zahlen
$\mathbb{R}_{>0}$	positive reelle Zahlen
$\mathbb{R}_{\geq 0}$	nicht-negative reelle Zahlen
$\mathbb{N}$	natürliche Zahlen
I_d	Einheitsmatrix der Dimension d
$\|A\|$	Determinante einer Matrix, Größe einer Menge (Kontextbezogen)
A^T	Transponierte einer Matrix
$\emptyset$	Leere Menge

B.2. Wahrscheinlichkeitsverteilungen

Univariate Gleichverteilung $Unif(a,b)$

Parameter	$a,b \in \mathbb{R}, a < b$
Träger	$x \in \mathbb{R}$
Dichtefunktion	$\begin{cases} \frac{1}{b-a} & \text{falls } a < x < b \\ 0 & \text{sonst} \end{cases}$
Erwartungswert	$\frac{a+b}{2}$
Varianz	$\frac{(b-a)^2}{12}$

Univariate Normalverteilung $Norm(\mu,\sigma^2)$

Parameter	Erwartungswert $\mu \in \mathbb{R}$, Varianz $\sigma^2 \in \mathbb{R}_{>0}$
Träger	$x \in \mathbb{R}$
Dichtefunktion	$\frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{(x-\mu)^2}{2\sigma^2}\Big)$
Erwartungswert	μ
Varianz	σ^2

Multivariate Normalverteilung $Norm(\mu,\Sigma)$

Parameter	Dimension d, Erwartungswert $\mu \in \mathbb{R}^d$, (Ko-)Varianz $\Sigma \in \mathbb{R}^{d\times d}$
Träger	$x \in \mathbb{R}^d$
Dichtefunktion	$\frac{1}{\sqrt{(2\pi)^d \cdot \lvert\Sigma\rvert}} \exp\Big(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\Big)$
Erwartungswert	μ
Varianz/Kovarianz	Σ

Gammaverteilung $\Gamma(\alpha,\beta)$

Parameter	$\alpha \in \mathbb{R}_{>0}$, $\beta \in \mathbb{R}_{>0}$
Träger	$x \in \mathbb{R}_{>0}$
Dichtefunktion	$\begin{cases} \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} \mathrm{e}^{-\beta x} & x > 0 \\ 0 & x \leq 0 \end{cases}$
Sonstiges	Gammafunktion $\Gamma(n+1) = n!$
Erwartungswert	$\frac{\alpha}{\beta}$
Varianz	$\frac{\alpha}{\beta^2}$

Inverse Gammaverteilung $\Gamma^{-1}(\alpha,\beta)$

Parameter	$\alpha \in \mathbb{R}_{>0}$, $\beta \in \mathbb{R}_{>0}$
Träger	$x \in \mathbb{R}_{>0}$
Dichtefunktion	$\begin{cases} \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} \mathrm{e}^{-\frac{\beta}{x}} & x > 0 \\ 0 & x \leq 0 \end{cases}$
Sonstiges	Gammafunktion $\Gamma(n+1) = n!$
Erwartungswert	$\frac{\beta}{\alpha-1}$
Varianz	$\frac{\beta^2}{(\alpha-1)^2(\alpha-2)}$

χ^2_n - Verteilung

Parameter	$n \in \mathbb{R}_{>0}$
Träger	$x \in \mathbb{R}_{>0}$
Dichtefunktion	$\begin{cases} \frac{x^{\frac{n}{2}-1} \mathrm{e}^{-\frac{x}{2}}}{2^{\frac{n}{2}} \Gamma(\frac{n}{2})} & x > 0 \\ 0 & x \leq 0 \end{cases}$
Sonstiges	Gammafunktion $\Gamma(n+1) = n!$
Erwartungswert	n
Varianz	$2n$

C. Eingabedaten

C.1. Zusammenfassung der Parameter

C.1.1. Segmentierung der Eingabedaten

Parameter	Erklärung	Wert
	Allgemeine Parameter	
α	Fahrtwinkeländerung zur Klassifikation als Abbiegepunkt	15 °
s_{a}	Durchschnittsgeschwindigkeit zur Klassifikation als Abbiegepunkt	30 km/h
d_{cl}	Maximale Distanz zwischen einer Trajektorie und einem Cluster	45 m
d_{vc}	Distanz zwischen zwei virtuellen Clustern	30 m
w_{vp}	Größe der Dilatation um die Trajektorien	10 m
ϵ_{seg}	DBSCAN Algorithmus: Mindestabstand zwischen zwei Clustern. Verwendet zur Detektion von Kurven ud Kreuzungen in Abbiegepunkten.	35 m
$\mathrm{minPts}_{\mathrm{seg}}$	DBSCAN Algorithmus: Mindestgröße eines Clusters. Verwendet zur Detektion von Kurven ud Kreuzungen in Abbiegepunkten.	5
	Unterscheidung von Brücken und Kreuzungen	
ϵ_{br}	DBSCAN Algorithmus: Mindestabstand zwischen zwei Clustern. Verwendet zum Bündeln von Trajektorien auf Straßen.	10 m
$\mathrm{minPts}_{\mathrm{br}}$	DBSCAN Algorithmus: Mindestgröße eines Clusters. Verwendet zum Bündeln von Trajektorien auf Straßen.	3

Tabelle C.1.: Zusammenfassung aller verwendeter Parameter zur Segmentierung der Eingabedaten.

C.1.2. Bewertung von Ergebnissen

Parameter	Erklärung	Wert
	F-Score	
δ_I	Anzahl der Iterationen zur Bestimmung des F-Score	1500
δ_F	Maximale Distanz zum Startknoten bei der Erzeugung einer Baumstruktur zur Auswertung des F-Score	100 m
δ_D	Entfernung zwischen zwei „Murmeln“ bzw. „Löchern“.	10 m

Tabelle C.2.: Zusammenfassung aller verwendeter Parameter zur Bewertung von Ergebnissen.

C.1.3. Fahrbahngenaue Rekonstruktion

Parameter	Erklärung	Wert
	Allgemeine Parameter	
ϵ_r	DBSCAN Algorithmus: Mindestabstand zwischen zwei Clustern. Verwendet zum Bündeln von Trajektorien auf Straßen.	5 m
$\mathrm{minPts}_\mathrm{r}$	DBSCAN Algorithmus: Mindestgröße eines Clusters. Verwendet zum Bündeln von Trajektorien auf Straßen.	3
β	Verhältnis zwischen Modellannahme und a-priori Wissen	0.5
λ	Formparameter der Abkühlfunktion im Simulated Annealing.	0.92
s	Schrittanzahl zum Abkühlen im Simulated Annealing.	100
	RJMCMC Parameter	
σ_move	Maximale Bewegungsweite der „Move“ Operation	2 m
σ_birth	Maximaler Abstand zum Graphen beim Erzeugen eines neuen Knotens durch die „Birth“ Operation	5.5 m

Tabelle C.3.: Zusammenfassung aller verwendeter Parameter zur Rekonstruktion fahrbahngenauer Straßenkarten.

C.1.4. Fahrspurgenaue Rekonstruktion

Parameter	Erklärung	Wert
	Allgemeine Parameter	
d_{cr}	Cluster Grenzwert zum Vereinen von Knoten der fahrbahngenauen Karte zu einer Kreuzung	12 m
δ_{init}	Minimale Fahrtwinkeländerung zur Bestimmung des initialen Armabstandes auf einer Kreuzung	10 °
$d_{\text{init,min}}$	Minimaler initialer Armabstand auf einer Kreuzung	20 m
$d_{\text{init,max}}$	Maximaler initialer Armabstand auf einer Kreuzung	50 m
δ	Verhältnis zwischen Modellannahme und a-priori Wissen	0.5
η	Schwellwert der Sprungfunktion bezüglich der benötigten Durchfahrten einer Spur	1
κ	Minimale Blocklänge	10 m
λ_{warmup}	Formparameter der Abkühlfunktion im Simulated Annealing.	0.77
s_{warmup}	Schrittanzahl zum Abkühlen im Simulated Annealing.	120
λ_{main}	Formparameter der Abkühlfunktion im Simulated Annealing.	0.029
s_{main}	Schrittanzahl zum Abkühlen im Simulated Annealing.	3200
	RJMCMC Parameter	
$\sigma_{\text{add lane}}$	Erwartungswert zur Bestimmung der initialen Spurbreite der „Add Lane“ Operation	2.75 m
$\sigma_{\text{add lane}}$	Varianz zur Bestimmung der initialen Spurbreite der „Add Lane“ Operation	0.55 m^2
$\sigma_{\text{middle gap}}$	Random Walk Parameter der „Adjust Middle Gap“ Operation	0.4 m
$\sigma_{\text{lane width}}$	Random Walk Parameter der „Adjust Lane Width“ Operation	0.6 m
$\sigma_{\text{curvature}}$	Random Walk Parameter der „Adjust Curvature“ Operation	0.3 m
σ_{distance}	Random Walk Parameter der „Adjust Distance“ Operation	1 m
σ_{angle}	Random Walk Parameter der „Adjust Angle“ Operation	5 °

Tabelle C.4.: Zusammenfassung aller verwendeter Parameter zur Rekonstruktion fahrspurgenauer Straßenkarten.

C.2. Egotrajektorien

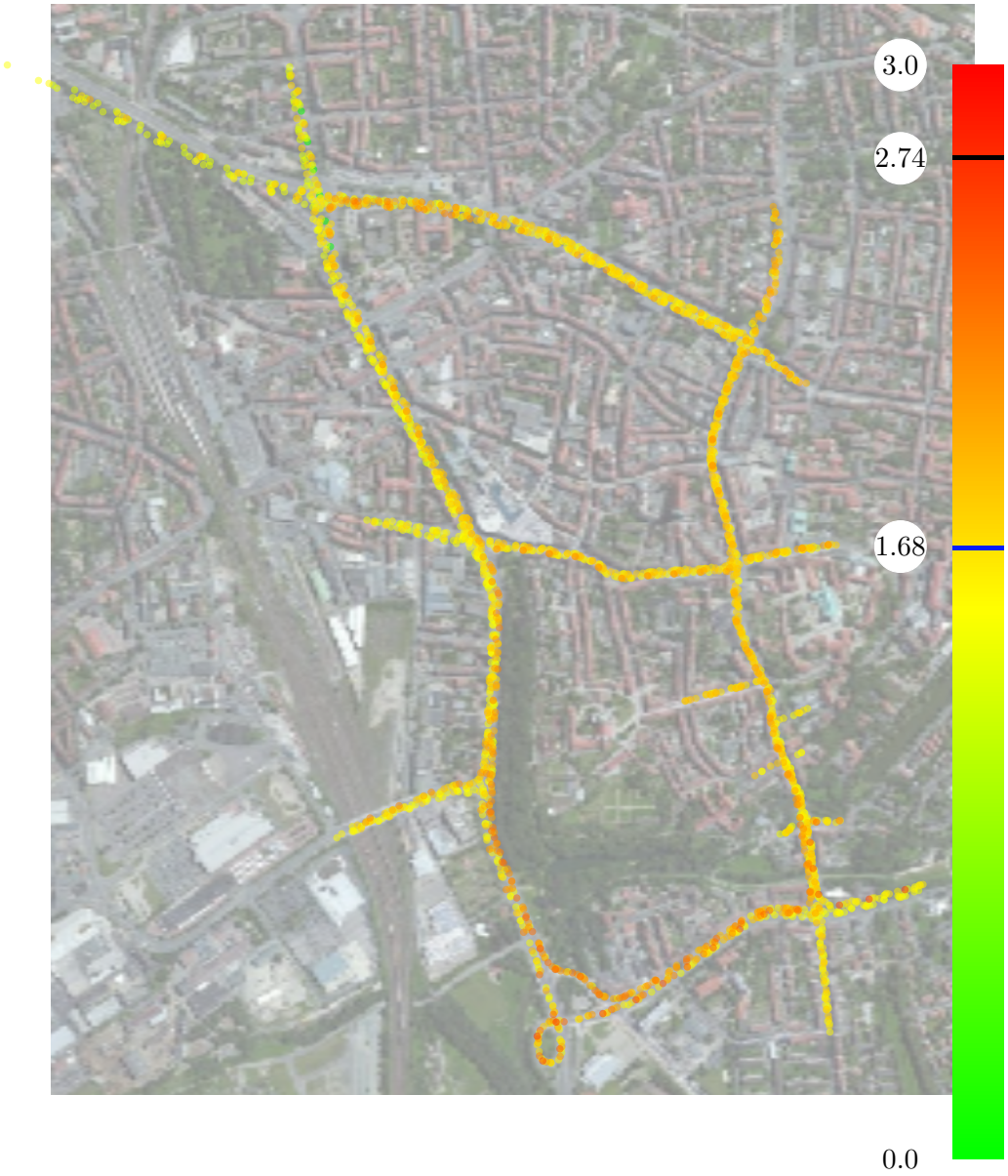


Abbildung C.1.: RMS Positionierungsfehler (Ausgabe des Systems von Applanix) in den Egotrajektorien der 1. Stufe: GNSS (GAMS). Die Farbskala beschreibt den Fehler in Metern. Die blaue Markierung markiert den durchschnittlichen Fehler, die schwarze den maximalen.

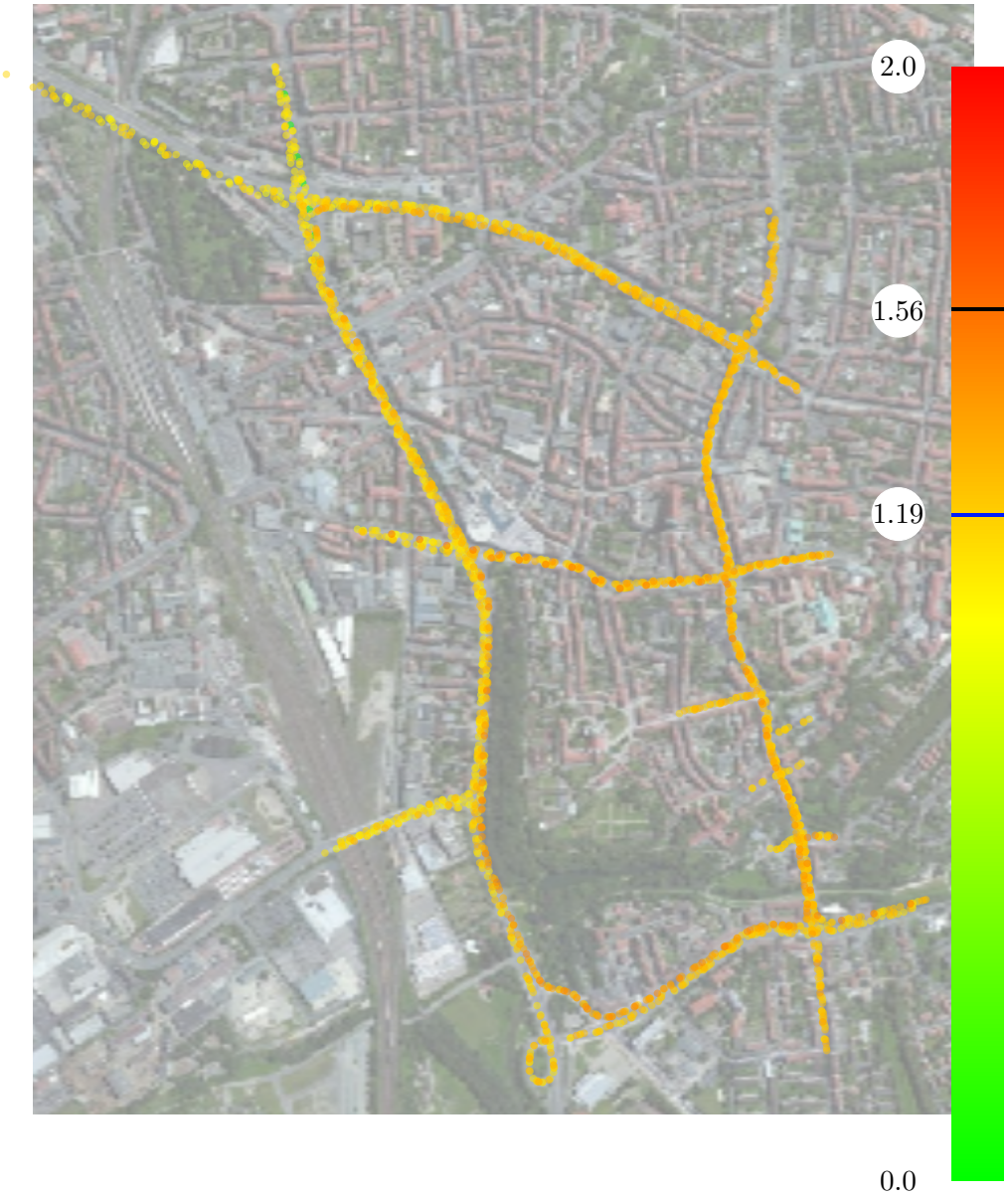


Abbildung C.2.: RMS Positionierungsfehler (Ausgabe des Systems von Applanix) in den Egotrajektorien der 2. Stufe: GNSS (GAMS) + IMU + DMI. Die Farbskala beschreibt den Fehler in Metern. Die blaue Markierung markiert den durchschnittlichen Fehler, die schwarze den maximalen.

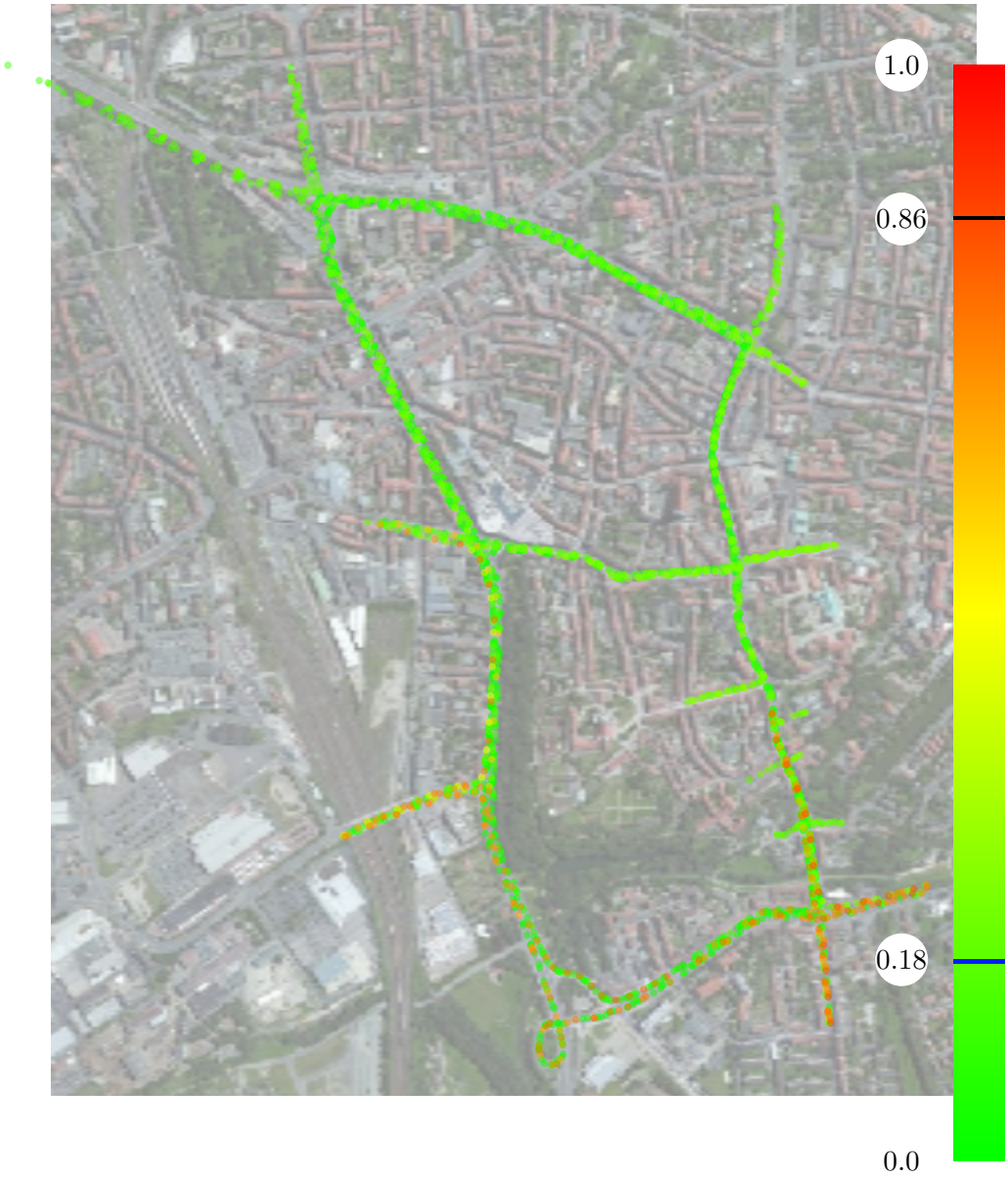


Abbildung C.3.: RMS Positionierungsfehler (Ausgabe des Systems von Applanix) in den Egotrajektorien der 3. Stufe: GNSS (GAMS) + IMU + DMI + SAPOS. Die Farbskala beschreibt den Fehler in Metern. Die blaue Markierung markiert den durchschnittlichen Fehler, die schwarze den maximalen.

C.3. Objekttrajektorien

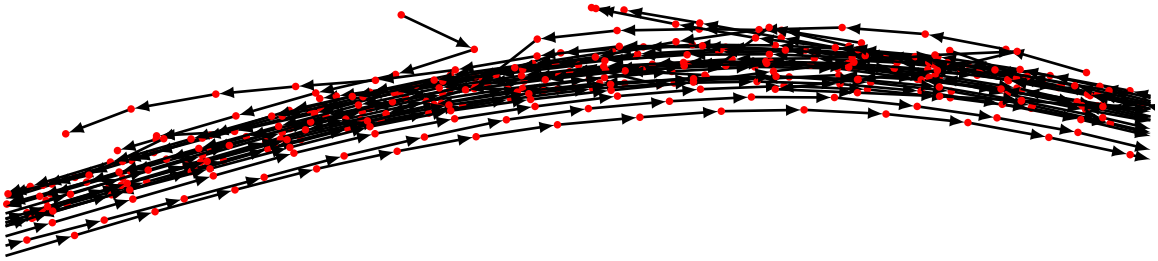


Abbildung C.4.: Objekttrajektorien auf einer Straße basierend auf GNSS (GAMS)

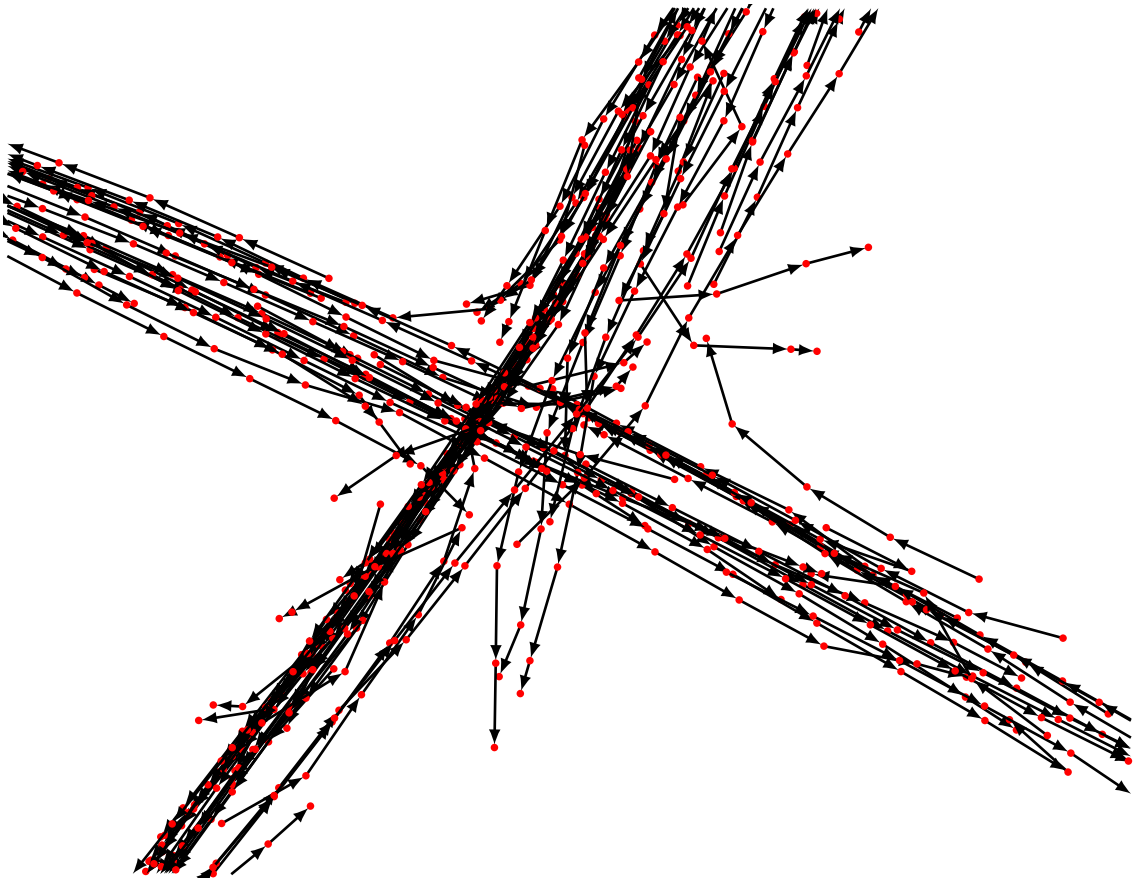


Abbildung C.5.: Objekttrajektorien an einer Kreuzung basierend auf GNSS (GAMS) + IMU + DMI

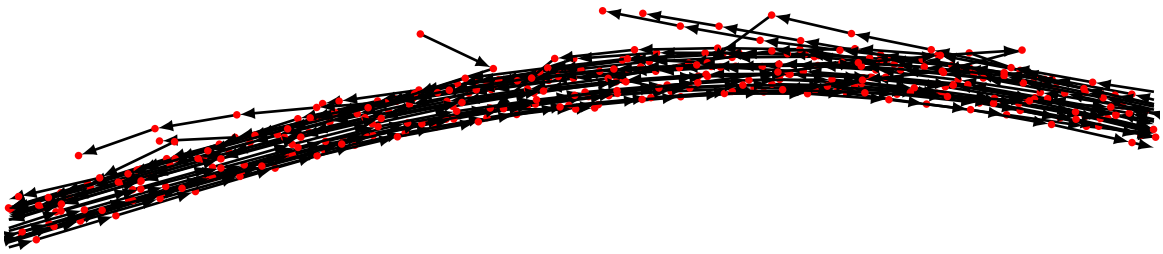


Abbildung C.6.: Objekttrajektorien auf einer Straße basierend auf GNSS (GAMS) + IMU + DMI

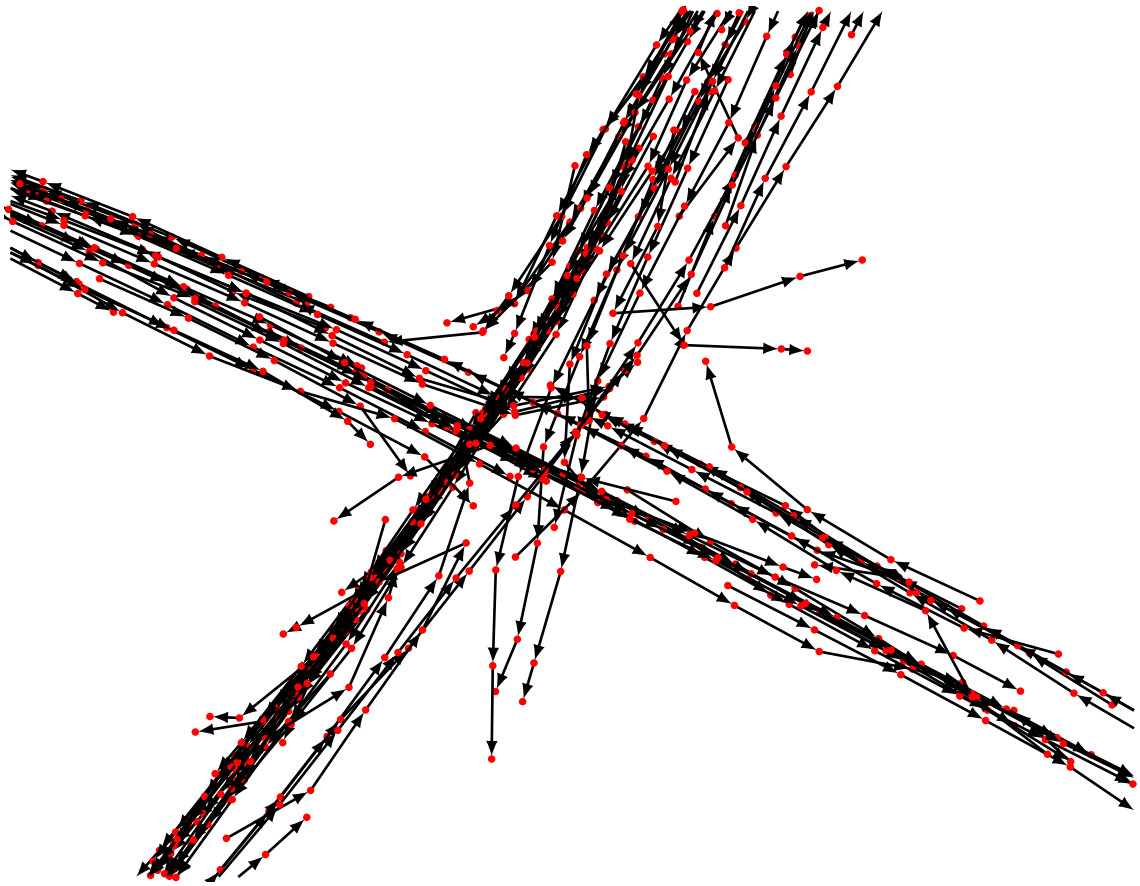


Abbildung C.7.: Objekttrajektorien an einer Kreuzung basierend auf GNSS (GAMS) + IMU + DMI + SAPOS

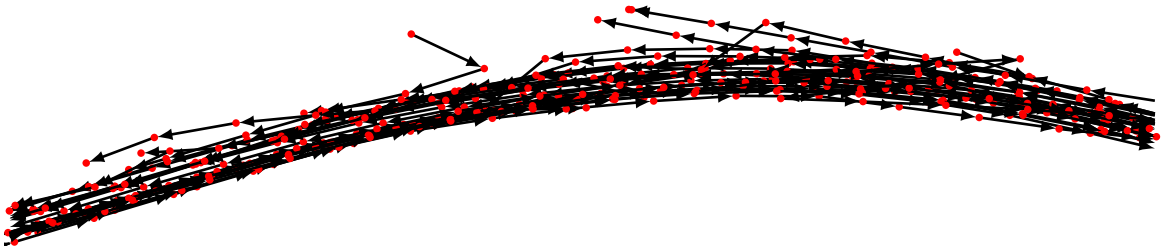


Abbildung C.8.: Objekttrajektorien auf einer Straße basierend auf GNSS (GAMS) + IMU + DMI + SAPOS

D. Dokumente

D.1. Verkehrszählung Hildesheim

Das Dokument beschreibt die Verkehrszählung auf Teilen des Hildesheimer Stadtgebietes und wurde von der Stadt Hildesheim im Jahr 2006 im Rahmen einer Klage zur Radwegbenutzungspflicht eingebracht. Diese Zählung wird in einem Blog Beitrag[a] erwähnt und wurde auf Anfrage vom Autor ausgehändigt:

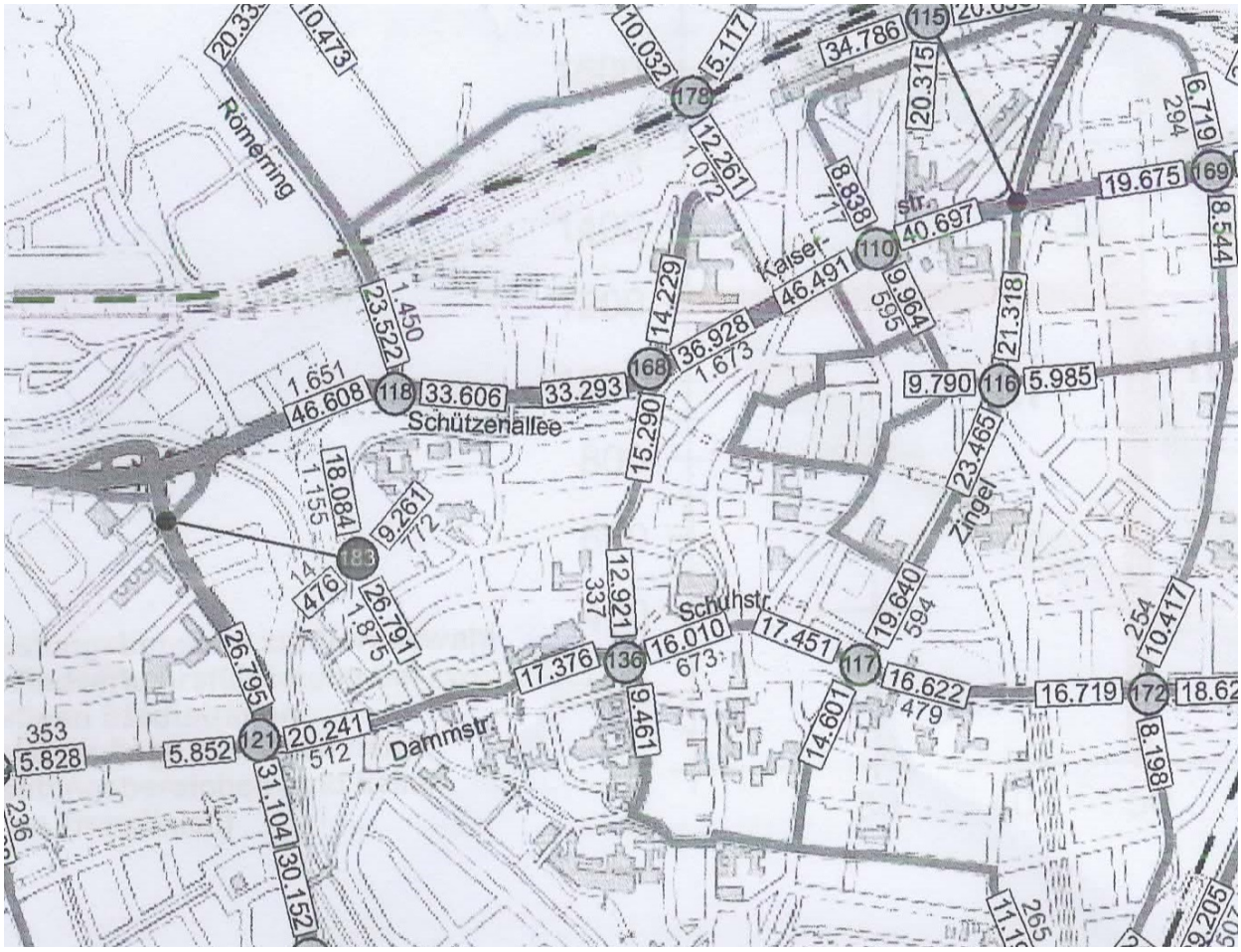


Abbildung D.1.: Verkehrszählung Hildesheim 2006

[a] http://blog.flaccus-hildesheim.de/tag/verkehr/

E. Danksagung

Die vorliegende Arbeit entstand in einem dreijährigen Engagement in der Abteilung für Forschung und Vorausentwicklung der Robert Bosch GmbH am Standort Hildesheim im Rahmen eines Forschungsprojektes in dem Bereich des autonomen Führens von Fahrzeugen.

Mein herzlichster Dank geht an Prof. Dr.-Ing. Claus Brenner vom Institut für Kartographie und Geoinformatik der Leibniz Universität Hannover für die Übernahme der Position des Doktorvaters und Betreuers dieser Arbeit. Durch die vielen eingebrachten Anregungen, Ideen und Vorschläge, sowie ein großes Maß an Geduld und Freundlichkeit wurde die Durchführung dieser Arbeit erst ermöglicht.

Bei Prof. Dr.-Ing. Christoph Stiller vom Institut für Mess- und Regelungstechnik des Karlsruher Instituts für Technologie, sowie bei Prof. Dr.-Ing. Christian Heipke vom Institut für Photogrammetrie und Geoinformation der Leibniz Universität Hannover bedanke ich mich herzlichst für die Teilnahme am Prüfungsverfahren als Referenten.

Eine ausgesprochene Freude war mir die Zusammenarbeit mit Herrn Dr.-Ing. Daniel Zaum von der Robert Bosch GmbH. Ihm gilt mein herzlichster Dank für die Betreuung der Arbeit vor Ort und für die entstandene Freundschaft. Durch seinen enormen persönlichen Einsatz, zahllose produktive Ideen und stets motivierende Art hat er entscheidenden Anteil am Gelingen dieser Arbeit. An dieser Stelle möchte ich mich auch nochmals für die ermöglichten vorhergegangenen Arbeiten unter seiner Betreuung bedanken.

Mein Dank geht auch an alle meine Kollegen für die hervorragende Unterstützung, für viele eingebrachte Ideen und für die stets angenehme Arbeitsatmosphäre.

Besonders dankbar bin ich meiner Familie, meinen Eltern Astrid und Hans-Werner und meinen Geschwistern Greta und Marius für die moralische Unterstützung in schweren und guten Zeiten und für den großartigen Rückhalt.

Mein tiefster Dank für einen liebevollen und unschätzbaren Beistand, für die richtigen Worte zur richtigen Zeit und für den nie endende Strom an Motivation geht eine meine wundervolle Ehefrau Miriam.

F. Widmung

Reich sind nur die, die wahre Freunde haben.

Thomas Fuller

In tiefster Dankbarkeit und enger Verbundenheit widme ich diese Arbeit meinem Freund und Wegbegleiter

Tom Heinrich Seeska

Durch seine jahrelange, selbstlose Unterstützung hat er maßgeblichen Anteil am Gelingen dieser, sowie aller dafür nötigen vorhergegangenen Arbeiten. Seine Güte und Treue werden mich mein Leben lang begleiten und mich immer an den unermesslichen Reichtum einer Freundschaft erinnern.

G. Lebenslauf

Persönliche Daten

Name	Oliver Bertin Röth
Geburtsdatum	28.12.1988
Geburtsort	Essen
Anschrift	Thiebachstraße 1
	31188 Holle

Berufliche Erfahrung

09/2017 – Heute	Entwicklungsingenieur, Robert Bosch GmbH Hildesheim, Geschäftsbereich: Chassis System Control
11/2014 – 09/2017	Wissenschaftliche Arbeit mit Promotionsziel, Robert Bosch GmbH Hildesheim, Geschäftsbereich: Corporate Research
11/2009 – 03/2014	Hilfswissenschaftlicher Mitarbeiter, Institut für Bauinformatik, Leibniz Universität Hannover
10/2010 – 02/2011	Hilfswissenschaftlicher Mitarbeiter, Institut für Baumechanik, Leibniz Universität Hannover

Ausbildung

2012 – 2014	Studium: Computergestützte Ingenieurwissenschaften, Leibniz Universität Hannover, Abschluss: Master of Science
2009 – 2012	Studium: Computergestützte Ingenieurwissenschaften, Leibniz Universität Hannover, Abschluss: Bachelor of Science
2008 – 2009	Freiwilliges Soziales Jahr Kultur, Calenberger Musikschule Gehrden
1995 – 2008	Schulische Ausbildung, Matthias-Claudius-Gynmnasium Gehrden, Abschluss: Abitur

Sonstige Erfahrung

09/2012	Teilnahme an einem Seminar zum Thema Zeitmanagement
09/2011	Teilnahme am Forum Bauinformatik, University College, Cork, Irland. Beitrag: Bewertung von Fußgängersimulationen im Rahmen eines kollektiven Lebensdauermanagement von Gebäuden
05/2011	Teilnahme am Seminar Rhetorik und Argumentation, Leibniz Universität Hannover
10/2010	Teilnahme an einer Schulung zur Leitung von Gruppen, Leibniz Universität Hannover

Wissenschaftliche Veröffentlichungen

Juni 2017	O. Roeth, D. Zaum, C. Brenner: Extracting Lane Geometry and Topology Information from Vehicle Fleet Trajectories in Complex Urban Scenarios using a Reversible Jump MCMC Method, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences (CMRT), Seite 51-58
Juni 2016	O. Roeth, D. Zaum, C. Brenner: Road Network Reconstruction using Reversible Jump MCMC Simulated Annealing based on Vehicle Trajectories from Fleet Measurements, IEEE Intelligent Vehicles Symposium (IV), Seite 194-201
Dezember 2015	O. Roeth, D. Zaum, C. Brenner: Improving GPS-based Road Network Constructions in a Post-Processing Step of Global Optimization, 14th International Conference on ITS Telecommunications (ITST), Seite 70-74

Praktika

10/2013 – 03/2014	Robert Bosch GmbH, 31139 Hildesheim
08/2011 – 09/2011	Frobese Informatikservices, 30169 Hannover
08/2008 – 09/2008	Holzbau Berens, 34471 Volkmarsen