

Virtueller Zug als mobile Entwicklungs- und Validierungsumgebung für den Galileo Online Navigationsempfänger

Virtual Train as Mobile Development and Validation Environment for the Galileo Online Navigation Receiver

Von der Fakultät für Maschinenwesen der Rheinisch-Westfälischen Technischen
Hochschule Aachen zur Erlangung des akademischen Grades eines Doktors der
Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von

Andreas Trzuskowsky

Berichter: Universitätsprofessor Dr.-Ing. Dirk Abel
Universitätsprofessor Dr. rer. nat. Bernhard Rumpe

Tag der mündlichen Prüfung: 7. September 2021

Diese Dissertation ist auf den Internetseiten der Universitätsbibliothek online verfügbar.

Vorwort

Diese Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Regelungstechnik an der Rheinisch-Westfälischen Technischen Hochschule Aachen. Mein besonderer Dank gilt dem Leiter des Instituts für Regelungstechnik, meinem Doktorvater, Herrn Univ.-Prof. Dr.-Ing. Dirk Abel, für sein mir entgegengebrachtes Vertrauen, seine Unterstützung und die Betreuung dieser Arbeit.

Ebenso gilt mein Dank Herrn Prof. Dr. rer. nat. Bernhard Rumpe, Leiter des Lehrstuhls für Software Engineering an der Rheinisch-Westfälischen Technischen Hochschule Aachen, für das Interesse an meiner Dissertation und die Übernahme des Koreferats. Univ.-Prof. Dr.-Ing. Christian Schindler danke ich für den Vorsitz der Prüfungskommission und Herrn Prof. em. Dr.-Ing. Heinrich Rake für den Beisitz in der Prüfung sowie für seine Anmerkungen zu meiner Arbeit.

Allen Mitarbeitern des Instituts für Regelungstechnik bin ich sehr dankbar für die konstruktive und angenehme Zusammenarbeit während meiner Zeit am Institut und darüber hinaus. In dieser Zeit sind viele andauernde Freundschaften entstanden. Besonders möchte ich Thomas Konrad, Sebastian Stemmler, Maike Stemmler, Michael Breuer und Jiaying Lin für die Unterstützung gerade in den letzten Phasen meiner Promotion danken.

An dieser Stelle möchte ich meiner Familie danken, dafür dass sie immer für mich da ist. Für meine Familie war von Anfang an klar, dass ich diese Arbeit zu einem erfolgreichen Ende bringen werde.

Aachen, im November 2021

Trutzzy

Kurzfassung

Bei der Entwicklung neuer Anwendungen, z. B. in der Automobilindustrie, gibt es zwei verbreitete Herangehensweisen. Auf der einen Seite kann mit Simulationen gearbeitet werden. So kann je nach Güte der Simulation das zu erwartende Ergebnis gut abgeschätzt werden. Auf der anderen Seite steht das vollwertige Experiment mit einem Prototypen. Hiermit können die besten Ergebnisse erzielt werden. Der Prototypenbau ist zeitaufwendig und kostspielig. Daher wird in der Arbeit gezeigt, dass Experimente im Modellmaßstab genutzt werden können, um schnell an nützliche Ergebnisse zu kommen.

Als Beispiel hierfür dient in dieser Arbeit die Entwicklung des Satellitennavigationsempfängers für Bahnanwendungen im Rahmen des Projektes Galileo Online. Die Entwicklung wird weitestgehend mit Testfahrzeugen im Modellmaßstab durchgeführt. Als Testfahrzeuge kommen die am Institut für Regelungstechnik der Rheinisch Westfälische Technische Hochschule (RWTH) Aachen University entwickelten sogenannten Buggys zum Einsatz. Sie sind etwa 1 m lang, 90 cm breit sowie 30 cm hoch und in der Lage, durch vorhandene Echtzeithardware automatisiert zu werden.

Damit der Galileo Online Navigationsempfänger keine Unterschiede zwischen einem Zug und den Buggys feststellen kann, müssen sich die Buggys wie ein Zug verhalten. Das bedeutet die Fahrzeuge dürfen sich nur auf fest vorgegebenen Pfaden bewegen, um so Schienen nachzubilden (virtuelle Schiene). Außerdem sollen sich mehrere zu einem „Zug“ zusammengeschlossene Fahrzeuge so verhalten, als wären sie mechanisch miteinander verbunden. Dazu soll, basierend auf Positionsmessungen und Kommunikation der Buggys untereinander, ein für alle Geschwindigkeiten gleichbleibender Abstand zwischen den Fahrzeugen eingeregelt werden (virtuelle Kupplung). Der Zusammenschluss von virtueller Schiene und virtueller Kupplung wird als virtueller Zug bezeichnet und im Rahmen dieser Arbeit entwickelt sowie vorgestellt.

Zur Realisierung des virtuellen Zuges werden zwei Verfahren untersucht: die Pfadregelung und eine Flachheitsbasierte Folgeregung. Bei der Pfadregelung handelt es sich um ein robustes aber rechenintensives Verfahren, das mit einer Rate von 10 Hz arbeitet. Die Flachheitsbasierte Folgeregung lässt sich vergleichsweise leicht berechnen, stellt aber höhere Anforderungen an die Sollwerte. Deshalb muss ihr ein Trajektoriengenerator vorangestellt werden. Trotz des Trajektoriengenerators kann die Flachheitsbasierte Folgeregung mit einer Rate von 100 Hz ausgeführt werden.

Es wird gezeigt, dass beide Verfahren den virtuellen Zug umsetzen können. Aufgrund der höheren Rate sind die Ergebnisse der Flachheitsbasierten Folgeregung leicht besser. Außerdem wird mittels Simulationen und Testfahrten gezeigt, dass das Konzept des virtuellen Zuges an sich zur Entwicklung des Galileo Online Empfängers geeignet ist.

Abstract

When developing new applications, for example in the automotive industry, there are two common approaches. On the one hand, simulations can be carried out. Depending on the quality of the simulation, the expected result can be well estimated. On the other hand, there is the full-scale experimentation with a prototype. Hereby the best results can be achieved. However, prototyping is time-consuming and costly. Therefore, it is shown in this thesis, that also scaled experiments can be used to obtain useful results quickly.

An example for this is the development of the satellite navigation receiver for railroad applications within the project Galileo Online. The development is carried out as far as possible with model-scale test vehicles. The used test vehicles were developed at the Institute of Automatic Control at RWTH Aachen University. They are called Buggys and are about 1 m long, 90 cm wide and 30 cm high and suitable to be automated by existing real-time hardware.

The Buggys have to behave like a train, so that the Galileo Online navigation receiver cannot detect a difference between a train and the Buggys. The vehicles are only allowed to move along predefined paths in order to simulate railway tracks (virtual track). In addition, several vehicles combined to form a “train” should behave as if they were mechanically coupled to each other. For this purpose, based on position measurements and inter Buggy communication, a constant distance in between the vehicles shall be maintained at all velocities (virtual coupling). The combination of virtual track and virtual coupling is called virtual train: It is developed and presented in this thesis.

Two methods are investigated in order to realize the virtual train: The so-called path control and a flatness-based control. The path control, which operates at a rate of 10 Hz, is robust but computationally intensive. The flatness-based control is comparatively easy to calculate but makes higher demands on the setpoints. Therefore, a trajectory generator must be included. Despite the trajectory generator, the flatness-based control can operate at a rate of 100 Hz.

It is shown that both methods can implement the virtual train. Due to the higher rate, the results of the flatness-based control are slightly better. Furthermore, it is shown by means of simulations and test drives that the concept of the virtual train itself, is suitable for the development of the Galileo Online navigation receiver.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Einordnung der Thematik	2
1.2	Gliederung der Arbeit	3
2	Stand der Technik	5
2.1	Nomenklatur	5
2.2	Koordinatensysteme	6
2.2.1	Geozentrisches Koordinatensystem	6
2.2.2	World Geodetic System 1984	6
2.2.3	North-East-Down und East-North-Up Koordinatensystem	8
2.2.4	Gleiskoordinaten	9
2.3	Globale Satellitennavigationssysteme	10
2.3.1	Der NMEA-0183 Standard	12
2.3.2	Echtzeitkinematik (RTK)	14
2.4	Der Galileo Online Empfänger	14
2.5	Kartendaten und Map-Matching	18
2.6	Regelungstechnische Grundlagen	22
2.6.1	Zustandsraum	23
2.6.2	Flachheit	24
2.6.3	Linear Quadratischer Regler	25
2.6.4	Modellbasierte Prädiktive Regelung	26
2.7	Rapid Control Prototyping	30
2.8	Der IRT-Buggy	31
2.9	Pfadregelung	36
2.10	Mann-in-der-Mitte-Technik	42
3	Aufgabenstellung	45
3.1	Virtuelle Schiene	46
3.2	Virtuelle Kupplung	47
3.3	Szenarioleitstelle und Virtueller Zug	48
3.4	Lösungsansätze und Zielsetzung	49
4	Regelungsverfahren für den virtuellen Zug	53
4.1	Zielbremsung	53
4.2	Kolonnenregelung	57
4.3	Flachheitsbasierte Folgeregelung	60
4.3.1	Geometrische Figuren	61

4.3.2	Modellierung	68
4.3.3	Regler Synthese	70
4.3.4	Trajektorien-generator	76
5	Testumgebungen	87
5.1	IRT-Buggy Modifikationen	87
5.1.1	Buggykommunikation mittels GO!-Navigationsempfänger	87
5.1.2	CAN Erweiterungen	89
5.1.3	Sicherheitskonzept	90
5.2	Testfahrzeug <i>E-MAXI L</i>	92
5.3	Testfelder	95
5.3.1	Parkhaus	95
5.3.2	Aldenhoven Testing Center	97
5.3.3	Rail und Logistic Center Wustermark	100
5.4	Simulationsumgebung	101
6	Numerische und experimentelle Validierung	105
6.1	Validierung der Simulation	106
6.2	Auslegung der Flacheitsbasierten Folgeregelung	119
6.3	Untersuchung der virtuellen Kupplung	122
6.4	Untersuchung der Rechenkomplexität	124
6.5	GO!-Abschluss mit den IRT-Buggys	125
6.6	GO!-Abschluss mit dem <i>E-Maxi L</i>	127
6.7	Diskussion der Ergebnisse	131
7	Zusammenfassung	133
8	Ausblick	135
	Literaturverzeichnis	139

Akronyme

ATC Aldenhoven Testing Center
BeiDou Chinesisches Satellitennavigationssystem
CAN Controller Area Network
DGL Differenzialgleichung
DOP Dilution of Precision
ECEF Geozentrisches Koordinatensystem (engl. Earth-Centered, Earth-Fixed)
EECS Lehrstuhl für Allgemeine Elektrotechnik und Datenverarbeitungssysteme der RWTH Aachen University
E-MAXI L Modellbezeichnung eines Elektrorangiergeräts der ZAGRO Group
ENU East-North-Up Koordinatensystem
ESA Europäische Weltraumorganisation (engl. European Space Agency)
FBFR Flachheitsbasierte Folgeregelung
Galileo Europäisches Satellitennavigationssystem
GGA Global Positioning System Fix Data
GGQ Real-time Position with Coordinate Quality
GLONASS Russisches Satellitennavigationssystem
GNSS Globales Navigationssatellitensystem
GO! Galileo Online
GOOSE Entwicklungsplattform für Satellitennavigation des IIS
GPS Global Positioning System
GSA GNSS DOP and Active Satellites
GST GNSS Pseudorange Error Statistics
GUI Grafische Nutzer-Schnittstelle (engl. Graphical User Interface)
HB Herzschlag (engl. Heart Beat)
HiL Hardware in the Loop
HiL-Modus Hardware in the Loop Modus des Buggy
ID Identifikation
IIS Fraunhofer-Institut für Integrierte Schaltungen
IMST Institut für Mobil- und Satellitenfunktechnik GmbH
IMU Inertiale Messeinheit (engl. Inertial Measurement Unit)
InnoZ Innovationszentrum für Mobilität und gesellschaftlichen Wandel GmbH
IRT Institut für Regelungstechnik der RWTH Aachen University
LLH Gebräuchliche bezeichnung für WGS-84-Koordinaten (engl. latitude, longitude, height)
LQR Lienar Quadratischer Regler

LTE	Long Time Evolution
Map Matching	Kartenabgleich (engl. Map Matching)
MiL	Model in the Loop
MPR	Modelbasierte Prädiktive Regelung
NAVSTAR	Navigational Satellite Timing and Ranging
NED	North-East-Down Koordinatensystem
NMEA	Nationale Vereinigung für Marineelektronik (engl. National Marine Electronics Association)
Ntrip	Vernetzter Transport von RTCM über das Internetprotokoll (engl. Networked Transport of RTCM via Internet Protocol)
OSM	OpenStreetMap
PKW	Personenkraftwagen
POI	Punkte von Interesse (engl. Point of Interest)
PPS	Puls pro Sekunde
PVT	Position, Geschwindigkeit und Zeit (engl. Position, Velocity and Time)
QZSS	Quasi-Zenit-Satelliten-System
RC	Fernsteuerung (engl. Remote Control)
RCP	Rapid Control Prototyping
RLCW	Rail und Logistic Center Wustermark GmbH & Co. KG
RMC	Recommended Minimum Sentence C
RTCM	Funktechnische Kommission für Seeverkehrsdienste (engl. Radio Technical Commission for Maritime Services)
RTK	Echtzeitkinematik (engl. Real Time Kinematic)
RWTH	Rheinisch Westfälische Technische Hochschule
SAPOS®	Satellitenpositionierungsdienst der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland
SFCD	Smart Feature Cellular Device
SiL	Software in the Loop
SLS	Szenarioleitstelle
SNR	Signal-zu-Rausch-Verhältnis (engl. signal-to-noise ratio)
TET	Gesamtausführungszeit (engl. total execution time)
TTM	Time to Market
UDP	User Datagram Protocol
USB	Universal Serial Bus
V2V	Fahrzeug zu Fahrzeug (engl. Vehicle to Vehicle)
VDI	Verein Deutscher Ingenieure
VPN	Virtuelles Privates Netzwerk
WGS-84	World Geodetic System aus dem Jahr 1984
WLAN	drahtloses lokales Netzwerk (engl. Wireless Local Area Network)
WPAN	drahtloses privates Netzwerk (engl. Wireless Personal Area Network)
XML	Erweiterbare Auszeichnungssprache (engl. Extensible Markup Language)
XPC	Auf dem Buggy als externer Regler verbauter Industrie-PC
ZigBee	auf WPAN aufgesetzte Spezifikation

Symbole

a	Große Halbachse des WGS-84-Ellipsoids
a	Beschleunigung des Kolonnenfahrzeugs
a	Variable zur Bestimmung der Kreisparameter
\mathbf{A}	. . .	Kontinuierliche Systemmatrix
A	. . .	Parameter der Klothoide
A	. . .	Linke Strecke auf dem Parkdeck
a_0	. . .	Beschleunigung des Führungsfahrzeugs der Kolonne
\mathbf{A}_D	. .	Zeitdiskrete Systemmatrix
a_{lat}	. .	Querbeschleunigung
b	Kleine Halbachse des WGS-84-Ellipsoids
b	Variable zur Bestimmung der Kreisparameter
\mathbf{B}	. . .	Kontinuierliche Eingangsmatrix
B	. . .	Rechte Strecke auf dem Parkdeck
\mathbf{B}_D	. .	Zeitdiskrete Eingangsmatrix
c	Variable zur Bestimmung der Kreisparameter
\mathbf{C}	. . .	Kontinuierliche Ausgangsmatrix
\mathbf{C}_D	. .	Zeitdiskrete Ausgangsmatrix
CP_1	. .	1. Karotten-Punkt der Pfadregelung
$CP_{1.5}$. .	1½. Karotten-Punkt der Pfadregelung
CP_2	. .	2. Karotten-Punkt der Pfadregelung
d	Abstand der Zielbremsung
D	. . .	Sollabstand der Kolonnenregelung
D	. . .	Abrückmaß einer Klothoide
\mathbf{D}	. . .	Kontinuierliche Durchgriffsmatrix
d_1	. . .	Abstand des ersten Karottenpunkts von der Referenzstrecke
$d_{1.5}$. .	Abstand des 1½ Karottenpunkts von der Referenzstrecke
d_2	. . .	Abstand des zweiten Karottenpunkts von der Referenzstrecke
δ	Einspurmodelllenkwinkel
δ	Dämpfung eines Verzögerungsgliedes 2. Ordnung
δ_{max}	. .	Maximaler Lenkwinkel
d_l	. . .	Zielabstand der Zielbremsung, ab der die Geschwindigkeit nicht weiter verringert wird, sondern konstant bei v_l bleibt
d	Abwärts- (engl. down) Komponente des NED-Koordinatensystems
d_s	. . .	Zielabstand der Zielbremsung, ab der die Geschwindigkeit verringert wird
d_T	. . .	Klothoidenversatz einer Scheitel-Klothoide
e	Exzentrizität des WGS-84-Ellipsoids
\mathbf{e}	Fehlervektor

e	Abstandsfehler
e	Ost- (engl. east) Komponente des NED- bzw. ENU-Koordinatensystems
\dot{e}	Geschwindigkeitsfehler
\ddot{e}	Beschleunigungsfehler
f	Abflachung des WGS-84-Ellipsoids
Ψ	Gierwinkel des MPR-Fahrzeugmodells
h	Höhe im WGS-84-Koordinatensystem
h	Höhe über dem WGS-84-Ellipsoids
i	Antriebsstrom
\mathbf{I}	Quadratische Einheitsmatrix
id	Strecken ID
J	Wert der Kostenfunktion
J_{CP}	Anteil der Kostenfunktion des 2½ Karotten-Reglers, der die Prädiktion berücksichtigt
J_{opt}	Optimale Lösung der Kostenfunktion
J_{Path}	Gesamtwert der Kostenfunktion des 2½ Karotten-Reglers
J_R	Anteil der Kostenfunktion des 2½ Karotten-Reglers, der die Stellenergie berücksichtigt
J_{WD}	Anteil der Kostenfunktion des 2½ Karotten-Reglers, der die Fahrtrichtung berücksichtigt
κ	Krümmung
k	Diskrete Zeitvariable
k	Parameter der Zielbremsung, der angibt, ob kuppelbereit angehalten werden soll ($k = 1$) oder nicht ($k = 0$)
\mathbf{K}	Zustandsrückführungsmatrix
K	Element der Zustandsrückführungsmatrix \mathbf{K}
K_1	Parameter der Flachen Folgeregelung
K_0	Parameter der Flachen Folgeregelung
K_{step}	Anzahl der Zeitschritte T_{sample} , für die der 2½ Karotten-Regler einen Lenkwinkel in der Prädiktion anlegt
l_{Lenk}	Weg, der beim Volleinschlag der Lenkung bei Fahrt mit der Geschwindigkeit v_{max} zurückgelegt wird
L	Radstand der Buggys
L	Länge am Endpunkt einer Klothoide
λ	Längengrad im WGS-84-Koordinatensystem
λ	Wichtungsfaktor für die Stellenergie des 2½ Karotten-Reglers
LP	Lenkposition
l	Laufparameter einer Geraden
l	Laufparameter einer Klothoide
l	Laufparameter eines Kreises
l	Allgemeiner Laufparameter
n	Drehzahl
N	Höhe des WGS-84-Ellipsoids beim Winkel φ
\mathbf{n}	Normalenvektor, der von \mathbf{p}_s zu \mathbf{p}_z zeigt
N_1	Beginn des Prädiktionshorizonts der MPR

N_2	. . .	Ende des Prädiktionshorizonts der MPR
N	. . .	Anzahl der Folgefahrzeuge der Kolonnenregelung
n_{delta}	. .	Anzahl der äquidistant verteilten Werte pro Iteration bei der Lösung des 2½ Karotten-Regler Optimierungsproblems
n_{iter}	. .	Anzahl der Iterationen zur Lösung des 2½ Karotten-Regler Optimierungsproblems
n	. . .	Nord- (engl. north) Komponente des NED- bzw. ENU-Koordinatensystems
N_u	. . .	Stellhorizont der MPR
N_u	. . .	Stellhorizont des 2½ Karotten-Reglers
ω_0	. . .	Kreisfrequenz eines Verzögerungsgliedes 2. Ordnung
\mathbf{P}	. . .	Lösung der Matrix Riccati-Gleichung
\mathbf{p}_A	. . .	Geradenaufpunkt
\mathbf{p}_0	. . .	Kreismittelpunkt
P_1	. . .	Map Matching-Punkt des 1. Karotten-Punkt der Pfadregelung
\mathbf{p}_R	. . .	Punkt auf der Geraden
$P_{1.5}$. . .	Map Matching-Punkt des 1½. Karotten-Punkt der Pfadregelung
\mathbf{p}_1	. . .	Kreispunkt 1
P_2	. . .	Map Matching-Punkt des 2. Karotten-Punkt der Pfadregelung
\mathbf{p}_2	. . .	Kreispunkt 2
\mathbf{p}_3	. . .	Kreispunkt 3
\mathbf{p}_a	. . .	Aktuelle Position
$\mathbf{p}_ $. . .	Punkt der Geraden
\mathbf{p}_\sim	. . .	Punkt der Klothoide
φ	. . .	Breitengrad im WGS-84-Koordinatensystem
φ_0	. . .	Winkel des Anfangspunktes einer Bewegung auf einem Kreis
φ	. . .	Lenkwinkel des Flachen Modells
φ_o	. . .	Vom Kreisbogen abgedeckter Winkel
φ_{Dreh}	. .	Winkel zwischen zwei Geraden im Drehpunkt
φ_{Kreis}	. .	Vom Kreisabschnitt abgedeckter Winkel
\mathbf{p}_{φ_0}	. . .	Startpunkt des Kreisebogens entlang des Kreises
\mathbf{p}_o	. . .	Punkt des Kreises
\mathbf{P}_{opt}	. . .	Lösung der stationären Riccati-Gleichung
\mathbf{p}_s	. . .	Startpunkt
p_{Ax}	. . .	X-Koordinate des Geradenaufpunktes
p_{Rx}	. . .	X-Koordinate eines Punktes auf der Geraden
p_{Ay}	. . .	Y-Koordinate des Geradenaufpunktes
p_{Ry}	. . .	Y-Koordinate eines Punktes auf der Geraden
\mathbf{p}_z	. . .	Zielpunkt
\mathbf{Q}	. . .	Ausgangswichtungsmatrix
\mathbf{r}	. . .	Referenz der MPR
r	. . .	Radius eines Kreises
\mathbf{R}	. . .	Stellsignalwichtungsmatrix
R	. . .	Radius des MPR-Fahrzeugmodells
R	. . .	Krümmungsradius am Endpunkt einer Klothoide
R_{min}	. .	Minimaler Krümmungsradius

R	... Rotationsmatrix
r_{ZB}	... Radius des Zielgebietes bei der Zielbremsung bei Definition des Zielgebietes als Kreis
s	... Weglänge
s_1	... Weglänge des ersten Karottenpunkts
s_2	... Weglänge des zweiten Karottenpunkts
s	... Laufparameter einer Klothoide, der nicht der zurückgelegten Strecke entspricht
t	... Kontinuierliche Zeitvariable
T	... Beliebiger Zeitpunkt bei der Prädiktion des MPR-Fahrzeugmodells
T	... Schnittwinkel der Tangenten im Anfangs- und Endpunkt einer Klothoide
T_{ab}	... Abtastzeit
τ	... Integrationsvariable
t_{C1}	... Zeitpunkt, an dem der 1. Karotten-Punkt der Pfadregelung bestimmt wird
t_{C2}	... Zeitpunkt, an dem der 2. Karotten-Punkt der Pfadregelung bestimmt wird
θ	... Orientierungswinkel des Flachen Modells
$\hat{\theta}$... Schätzwert des Orientierungswinkels des Flachen Modells
T_{iter}	... Dauer eines Zeitschritts der 2½ Karotten Prädiktion
t_{Lenk}	... Zeit die benötigt wird um die Lenkung von der neutralen Position voll einzuschlagen
$T_{\frac{\pi}{2}}$... Zeit, die benötigt wird, um bei der aktuellen Geschwindigkeit mit maximalem Lenkeinschlag einen viertel Kreis zu fahren
T_{sample}	Abtastzeit mit der die Pfadregelung arbeitet
T_{step}	... Dauer, für die der 2½ Karotten-Regler einen Lenkwinkel in der Prädiktion anlegt
\mathbf{u}	... Eingangsvektor
\mathbf{u}_{opt}	... Optimaler Eingangsvektor
u	... Eingangskomponente des LQR-Modells (Beschleunigung)
u	... Komponente eines Kreis des Eingangsvektors
u	... Aufwärts- (engl. up) Komponente des ENU-Koordinatensystems
v	... Geschwindigkeit der Zielbremsung
v	... Fahrzeuggeschwindigkeit
V	... Fahrzeuggeschwindigkeit
\hat{v}	... Schätzwert der Fahrzeuggeschwindigkeit
v_l	... Langsame Geschwindigkeit der Zielbremsung
v_{Lenk}	... Lenkgeschwindigkeit
v_{max}	... Höchstgeschwindigkeit des Streckenabschnittes
v_s	... Schnelle Geschwindigkeit der Zielbremsung
\mathbf{w}	... Sollwert der MPR
\mathbf{x}	... Zustandsvektor
x	... Positionskomponente des LQR-Modells
x_o	... X-Koordinate eines Kreises
\mathbf{x}	... 2-Dimensionale Ortsvariable des Flachen Modells
\mathbf{y}	... 2-Dimensionaler Ausgang des Flachen Modells
X	... X-Koordinate des Referenzpunkts des MPR-Fahrzeugmodells
\mathbf{x}_0	... Anfangszustand zum Zeitpunkt $t = 0$
x_0	... X-Koordinate des Kreismittelpunkts

X_0	. . .	X-Koordinate des Kreismittelpunkts des MPR-Fahrzeugmodells
x_1	. . .	1. Ortsvariable des Flachen Modells
\dot{x}_1	. . .	Erste Ableitung der 1. Ortsvariable des Flachen Modells
\hat{x}_1	. . .	Schätzwert der 1. Ortsvariable des Flachen Modells
\ddot{x}_1	. . .	Zweite Ableitung der 1. Ortsvariable des Flachen Modells
x_1	. . .	X-Koordinate des 1. Kreispunktes
x_2	. . .	2. Ortsvariable des Flachen Modells
\dot{x}_2	. . .	Erste Ableitung der 2. Ortsvariable des Flachen Modells
\hat{x}_2	. . .	Schätzwert der 2. Ortsvariable des Flachen Modells
\ddot{x}_2	. . .	Zweite Ableitung der 2. Ortsvariable des Flachen Modells
x_2	. . .	X-Koordinate des 2. Kreispunktes
x_3	. . .	X-Koordinate des 3. Kreispunktes
x_{φ_0}	. .	X-Koordinate des Punktes \mathbf{p}_{φ_0}
x_a	. . .	ECEF X-Komponente des aktuellen Messpunkts \mathbf{p}_a
\dot{x}	Geschwindigkeitskomponente des LQR-Modells
\ddot{x}	Beschleunigungskomponente des LQR-Modells
x	X-Achse des ECEF-Koordinatensystems
$x $. . .	X-Koordinate der Geraden
y	Ausgangskomponente des LQR-Modells
x	Komponente des Zustandsvektors
x_{\sim}	. . .	X-Koordinate der Klothoide
x_z	. . .	ECEF X-Komponente des Zielpunkts \mathbf{p}_z
x_s	. . .	ECEF X-Komponente des Startpunkts \mathbf{p}_s
X_{VA}	. .	X-Koordinate des Vorderachsmittelpunkts des MPR-Fahrzeugmodells
\mathbf{y}	Ausgangsvektor
y_o	. . .	Y-Koordinate eines Kreises
Y	. . .	Y-Koordinate des Referenzpunkts des MPR-Fahrzeugmodells
y_0	. . .	Y-Koordinate des Kreismittelpunkts
Y_0	. . .	Y-Koordinate des Kreismittelpunkts des MPR-Fahrzeugmodells
y_1	. . .	1. Komponente des Ausgangs des Flachen Modells
\dot{y}_1	. . .	Erste Ableitung der 1. Komponente des Ausgangs des Flachen Modells
\ddot{y}_1	. . .	Zweite Ableitung der 1. Komponente des Ausgangs des Flachen Modells
y_1	. . .	Y-Koordinate des 1. Kreispunktes
y_2	. . .	2. Komponente des Ausgangs des Flachen Modells
\dot{y}_2	. . .	Erste Ableitung der 2. Komponente des Ausgangs des Flachen Modells
\ddot{y}_2	. . .	Zweite Ableitung der 2. Komponente des Ausgangs des Flachen Modells
y_2	. . .	Y-Koordinate des 2. Kreispunktes
y_3	. . .	Y-Koordinate des 3. Kreispunktes
y_{φ_0}	. . .	Y-Koordinate des Punktes \mathbf{p}_{φ_0}
y_a	. . .	ECEF Y-Komponente des aktuellen Messpunkts \mathbf{p}_a
\mathbf{y}_d	. . .	Soll Ausgangsvektor (vom engl. desired)
y	Y-Achse des ECEF-Koordinatensystems
$y $. . .	Y-Koordinate der Geraden
y	Komponente des Ausgangsvektors
y_{\sim}	. . .	Y-Koordinate der Klothoide

y_z	. . .	ECEF Y-Komponente des Zielpunkts \mathbf{p}_z
y_s	. . .	ECEF Y-Komponente des Startpunkts \mathbf{p}_s
Y_{VA}	. .	Y-Koordinate des Vorderachsmittelpunkts des MPR-Fahrzeugmodells
z	Eine beliebige Variable
z_a	. . .	ECEF Z-Komponente des aktuellen Messpunkts \mathbf{p}_a
z	Z-Achse des ECEF-Koordinatensystems
z_z	. . .	ECEF Z-Komponente des Zielpunkts \mathbf{p}_z
z_s	. . .	ECEF Z-Komponente des Startpunkts \mathbf{p}_s

1 Einleitung

In allen Branchen spielt die Time to Market (TTM) eine zunehmend wichtigere Rolle [15]. Um die TTM klein zu halten, muss die Entwicklungsgeschwindigkeit eines neuen Systems/Produktes entsprechend steigen. Es gibt mehrere Möglichkeiten, um im Rahmen eines Entwicklungsprozesses schnell an valide Ergebnisse zu gelangen. Einerseits steht eine immer größer werdende Anzahl an Simulationsumgebungen zur Verfügung. Durch immer mehr Rechenleistung werden auch komplexe Simulationen immer schneller. Andererseits sind die Ergebnisse eines Realversuchs viel aussagekräftiger, da alle in einer Simulation vernachlässigten Einflüsse mit in das Ergebnis eingehen. Im Allgemeinen sind solche Experimente aber auch sehr kostenintensiv. Betrachtet man beispielsweise die Automobilindustrie, so muss bei der Entwicklung neuer Technologien meist ein Testfahrzeug aufgebaut werden. Dieses weicht durch nachträgliche Anpassungen von einem Serienfahrzeug ab, was unter Umständen zum Erlöschen der Straßenzulassung führen kann. Dies macht es dann nötig, die Tests in abgesperrten Bereichen durchzuführen. Selbst wenn die Straßenzulassung erhalten bleibt, ist es vorteilhaft, Experimente in einer definierten Umgebung durchzuführen, z. B. einem Fahrzeug-Testgebiet. Dies kann neben den Fahrzeugkosten weitere Kosten mit sich bringen.

Es muss also ein Kompromiss zwischen Verlässlichkeit von Ergebnissen und deren Kosten gefunden werden. Ein Mittelweg, um in die Lücke zwischen Simulation und vollwertigem Experiment zu treten, kann die Verwendung von skalierten Testfahrzeugen sein. In [41] wird ein solches am Institut für Regelungstechnik der RWTH Aachen University (IRT) entwickeltes Testfahrzeug, der IRT-Buggy, vorgestellt. Es handelt sich dabei um einen fernsteuerbaren Versuchsträger, der etwa 1 m lang und 90 cm breit ist. Das Fahrzeug verfügt über zwei unabhängig voneinander lenkbare Vorderräder und unabhängig voneinander angetriebene Hinterräder. Der Fahrzeugaufbau ist durch ein Aluminium-Profil realisiert, sodass das Fahrzeug mit zusätzlichen Mess-/Steuer-/Regelungs-Aufbauten erweitert werden kann. Das System wird elektrisch angetrieben und verfügt daher über Batterien, von denen auch die zusätzlichen Aufbauten mit Strom versorgt werden können.

Die Verwendung eines skalierten Testfahrzeuges, wie dem IRT-Buggy, bringt einige Vorteile mit sich. Das Fahrzeug ermöglicht es, unter Verwendung der später tatsächlich genutzten Hardware-Ergebnisse zu erzielen. Im Vergleich zu einer Simulation bietet dies die Möglichkeit, Wechselwirkungen der Hardware direkt zu erkennen und zu berücksichtigen. Aus den gewonnenen Erkenntnissen kann sogar eine parallel eingesetzte Simulation verbessert werden. So kann die Simulation um die beobachteten Wechselwirkungen erweitert werden, falls diese bisher nicht berücksichtigt wurden. Anders als bei der Verwendung

eines vollwertigen Testfahrzeuges, beispielsweise eines Personenkraftwagen (PKW), sind die nötigen Anpassungen an dem Fahrzeug gering. Der Aufbau ist auf das Aufnehmen von Testhardware ausgelegt. Außerdem wird durch die Skalierung des Testfahrzeugs weniger Platz zum Durchführen der Experimente benötigt. Es kann also flexibler und agiler getestet werden. Ein weiterer Vorteil ist die Ausrichtung des Fahrzeugs als Entwicklungsplattform im Sinne von Rapid Control Prototyping (RCP).

Die Vorteile lassen sich auch ausspielen, wenn andere Verkehrsträger als der vorher genannte PKW untersucht werden sollen. So ist beispielsweise auch die Hardwareentwicklung im Bahnsektor ein möglicher Anwendungsfall, um die Vorteile des IRT-Buggy als Entwicklungsplattform auszuspielen. Dies wird im Rahmen des Projektes Galileo Online (GO!) ausgenutzt. Dort repräsentieren mehrere IRT-Buggys Züge bestehend aus Lokomotiven und Wagen. Zur Realisierung wird das Konzept des *virtuellen Zuges* verwendet, mit dessen Umsetzung sich in dieser Arbeit auseinandergesetzt wird.

1.1 Einordnung der Thematik

Das Thema dieser Arbeit besteht darin, das Konzept des virtuellen Zuges als mobile Entwicklungs- und Validierungsumgebung für den Galileo Online Navigationsempfänger zu verwenden. Dabei sieht es das Konzept des virtuellen Zuges vor, dass das Verhalten von Zügen durch Testfahrzeuge nachgebildet wird. Die Testfahrzeuge sind etwa 1 m lang und 90 cm breit. Sie können mobil nahezu überall eingesetzt werden. Im Rahmen dieser Arbeit werden sie verwendet, um den Galileo Online Navigationsempfänger zu entwickeln und zu validieren. Der Galileo Online Navigationsempfänger ist ein auf Bahnanwendungen zugeschnittener Satellitenempfänger, der im Rahmen des Projektes Galileo Online (GO!) entwickelt wird. Die Verwendung der Testfahrzeuge als virtueller Zug ist kostengünstiger als die Entwicklung mit realen Zügen. Zur späteren Validierung werden auch Experimente mit Zügen durchgeführt. Es soll jedoch gezeigt werden, dass die IRT-Buggys einen praktikablen Mittelweg zwischen reiner Simulation und vollwertigem Experiment darstellen und im Anwendungsfall des Projekts GO! von Nutzen sind.

Das Konzept des virtuellen Zuges sieht es vor, dass die Testfahrzeuge so geregelt werden, dass sie sich bewegen, als würden sie auf Schienen fahren. Sie sollen also automatisch immer dem selben Pfad folgen. Außerdem soll es möglich sein, mehrere Testfahrzeuge virtuell zu kuppeln. Das bedeutet, die Testfahrzeuge sollen einen festen Abstand zueinander einhalten und sich wie eine Einheit auf dem hinterlegten Pfad bewegen.

Können die IRT-Buggys das Verhalten des virtuellen Zuges abbilden, eignen sie sich als Entwicklungswerkzeug. Diese stellen dann einen Mittelweg zwischen Simulation und vollwertigem Experiment dar. So können realistischere Ergebnisse als die einer idealen Simulation erzielt werden, ohne dabei aber den Aufwand eines vollwertigen Experiments inkl. Prototyp betreiben zu müssen. Dies soll mit dieser Arbeit gezeigt werden.

1.2 Gliederung der Arbeit

Zunächst wird in Kapitel 2 der Stand der Technik in den für diese Arbeit relevanten Bereichen vorgestellt. Dazu gehören eine allgemeine Vorstellungen der Satellitennavigation und des Projekts GO! sowie Kartendaten und Map Matching. Aus methodischer Sicht wird das Verfahren des RCP vorgestellt sowie die Mann-in-der-Mitte-Technik. Außerdem werden das Testfahrzeug sowie bereits dafür entwickelte Regler präsentiert. Damit sind die Grundlagen dieser Arbeit gelegt.

Das Ziel der Arbeit ist es, die IRT-Buggys als Entwicklungsplattform für die Entwicklung des GO!-Empfängers zu validieren. Die sich aus diesem Ziel ergebenden Aufgabenstellungen werden in Kapitel 3 definiert. Dabei werden die Konzepte virtuelle Schiene, virtuelle Kupplung und virtueller Zug hergeleitet. Um diese Konzepte umzusetzen, werden weitere Notwendigkeiten ermittelt. So wird beispielsweise die sogenannte Szenarioleitstelle eingeführt. Basierend auf diesen Anforderungen werden zwei Lösungsvorschläge vorgestellt. Diese werden im weiteren Rahmen der Arbeit umgesetzt und validiert.

Basierend auf den in den Grundlagen vorgestellten Regelungsverfahren werden in Kapitel 4 Regelungskonzepte angeführt, mit denen sich das Konzept des virtuellen Zuges umsetzen lässt. Dabei gibt es zwei unterschiedliche Ansätze. Beide nutzen die sogenannte Zielbremsung und die Kolonnenregelung, weshalb diese Verfahren zunächst diskutiert werden. Anschließend wird der Ansatz der Pfadregelung vorgestellt. Er gründet auf einer Modelbasierten Prädiktiven Regelung und kommt im Rahmen des Projektes GO! zum Einsatz. In den letzten beiden Abschnitten des Kapitels zu den Regelungskonzepten wird sich mit dem zweiten Ansatz, einer Flachheitsbasierten Folgeregung, befasst. Zunächst wird diese hergeleitet und anschließend wird detailliert auf die für diese Methode notwendige Trajektoriengenerierung eingegangen.

Um die gefundenen Regelungen zu untersuchen, müssen diese auf den Testfahrzeugen implementiert und geeignete Testumgebungen gefunden werden. Die dazu nötigen Anpassungen der Testfahrzeuge und geeignete Testfelder werden in Kapitel 5 beschrieben. Zusätzlich wird die verwendete Simulationsumgebung vorgestellt. Diese wird für Voruntersuchungen genutzt.

In Kapitel 6 wird die Umsetzung des virtuellen Zuges validiert. Dazu werden Simulationen und reale Experimente mit den IRT-Buggys durchgeführt. Zusätzlich wird die Abschlussdemonstration aus dem Projekt GO! vorgestellt. So wird es möglich, die mit den IRT-Buggys erzielten Entwicklungsergebnisse mit der tatsächlichen Anwendung zu vergleichen.

Nachdem die Experimente vorgestellt worden sind, folgten in Kapitel 7 eine Zusammenfassung und Bewertung der Ergebnisse. Die Arbeit schließt mit einem Ausblick auf mögliche weiterführende Arbeiten in Kapitel 8.

2 Stand der Technik

Im Verlauf dieses Kapitels wird aufgezeigt, welche Vorarbeiten zu Beginn der Arbeit vorhanden sind. Dabei werden die Grundlagen und der Stand der Technik aufeinander aufbauend vorgestellt. Zunächst ist eine einheitliche Nomenklatur hilfreich. Daher werden in Abschnitt 2.1 die in dieser Arbeit verwendeten Notationen vorgestellt. Um ein System regeln zu können, werden in der Regel Istwerte benötigt. Für die Anwendung in dieser Arbeit ist es unerlässlich, die Position und Orientierung der Fahrzeuge bestimmen zu können. In Abschnitt 2.2 werden daher gängige Koordinatensysteme präsentiert. Damit können Positionen beschrieben werden. Zur Lokalisierung der Testfahrzeuge kann Satellitennavigation verwendet werden. Daher werden in Abschnitt 2.3 globale Satellitennavigationssysteme aufgeführt. Um mit den Daten, die von den globalen Satellitennavigationssystemen bereitgestellt werden, eine Position zu bestimmen, ist ein Empfänger nötig. In Abschnitt 2.4 wird der Galileo Online Navigationsempfänger als ein Beispiel vorgestellt. Außerdem wird dort gezeigt, was den Empfänger von anderen Empfängern unterscheidet. Mit Kartendaten kann aus der gemessenen Position eine Position in einer Karte bestimmt werden. Diese Koordinatentransformation wird in Abschnitt 2.5 als sogenanntes Map Matching vorgestellt. Damit ist grundlegend beschrieben, wie die Daten für eine Regelung zur Verfügung gestellt werden können. Die in dieser Arbeit verwendeten Grundlagen der Regelungstechnik werden in Abschnitt 2.6 kurz vorgestellt. In Abschnitt 2.7 wird als nächstes erläutert, wie mit den Methoden des Rapid Control Prototyping (RCP) ein Regler auf Echtzeithardware ausgelegt werden kann. Anschließend folgt die Vorstellung der in dieser Arbeit verwendeten Testfahrzeuge in Abschnitt 2.8. Die Testfahrzeuge sind so aufgebaut, dass die Methoden des RCP mit ihnen angewendet werden können. Ein Beispiel hierfür ist der in Abschnitt 2.9 vorgestellte Regler zur Querverführung der Testfahrzeuge. Im letzten Abschnitt, Abschnitt 2.10, wird eine Methode zur Kommunikationsmanipulation vorgestellt. Sie findet bei der Automatisierung des Schienenfahrzeugs Anwendung, die bei der Validierung zum Einsatz kommt.

2.1 Nomenklatur

Da im weiteren Verlauf dieser Arbeit immer wieder Skalare, Vektoren und Matrizen verwendet werden, folgen an dieser Stelle einige Notationen. Vektoren werden mit dick gedruckten Kleinbuchstaben und Matrizen mit dick gedruckten Großbuchstaben geschrieben. Um Matrizen und Vektoren in Formeln besser von Klammern abzugrenzen, werden sie mit eckigen Klammern eingefasst. Skalare verbleiben in der normalen Schriftstärke. Des Weiteren wird für das Skalarprodukt der Vektoren ***a*** und ***b*** die Schreibweise $\langle \mathbf{a}, \mathbf{b} \rangle$

verwendet. Für das Kreuzprodukt gilt die Schreibweise $\mathbf{a} \times \mathbf{b}$. Die Matrixmultiplikation zweier Matrizen \mathbf{A} und \mathbf{B} wird in der Notation $\mathbf{A} \cdot \mathbf{B}$ beschrieben. Die Einheitsmatrix wird als \mathbf{I} und die Nullmatrix als $\mathbf{0}$ bezeichnet. Bei der Einheitsmatrix \mathbf{I} handelt es sich, im Rahmen dieser Arbeit, immer um eine quadratische Matrix, anders als die Nullmatrix $\mathbf{0}$, die zwar quadratisch sein kann, aber kontextabhängig auch für eine nicht quadratische Matrix oder den Nullvektor stehen kann. Die Dimension kann aus dem Kontext bestimmt werden.

2.2 Koordinatensysteme

Positionen werden in unterschiedlichen Koordinatensystemen angegeben. Im Umfeld der Satellitennavigation haben sich die in [51] beschriebenen Koordinatensysteme etabliert. Die in dieser Arbeit verwendeten Koordinatensysteme werden im Folgenden, basierend auf [51], vorgestellt. Das in Abschnitt 2.2.4 angeführte System ist für das Map Matching bedeutsam und wird basierend auf [38] vorgestellt.

2.2.1 Geozentrisches Koordinatensystem

Als Geozentrisches Koordinatensystem (engl. Earth-Centered, Earth-Fixed) (ECEF) ist ein kartesisches Koordinatensystem bezeichnet, das seinen Ursprung im Erdmittelpunkt hat. Wie in der englischen Bezeichnung des Systems angedeutet, ist es erdfest. Es dreht sich also mit der Erde, wobei die z -Achse entlang der Erd-Rotationsachse ausgerichtet ist und für positive Werte in Richtung Nordpol zeigt. Die x -Achse zeigt vom Ursprung zum Schnittpunkt des Äquators und des Nullmeridians. Gemeinsam mit der y -Achse wird die Äquatorebene durch die beiden Achsen aufgespannt. [51]

2.2.2 World Geodetic System 1984

Da die Erde einem Ellipsoid ähnelt, gibt es mehrere Koordinatensysteme, die über Ellipsoide definiert sind. Eines davon ist das World Geodetic System aus dem Jahr 1984 (WGS-84). In ihm werden Positionen als Vektor mit den drei Koordinaten Breitengrad φ , Längengrad λ und Höhe h angegeben. Dabei wird die Höhe über dem Ellipsoid gemessen. Eine weitere gebräuchliche Bezeichnung der drei Werte ist LLH, was aus dem englischen kommt und für *latitude*, *longitude* und *height* steht. In Abbildung 2.1 ist der Ellipsoid abgebildet und wie die Winkel in Bezug auf das ECEF-Koordinatensystem gemessen werden. Der Breitengrad φ und der Längengrad λ werden vom Nullmeridian bzw. dem Äquator gemessen. Die Winkel werden in Richtung Osten bzw. Norden positiv gezählt. In Tabelle 2.1 sind die Parameter des WGS-84-Ellipsoids angegeben.

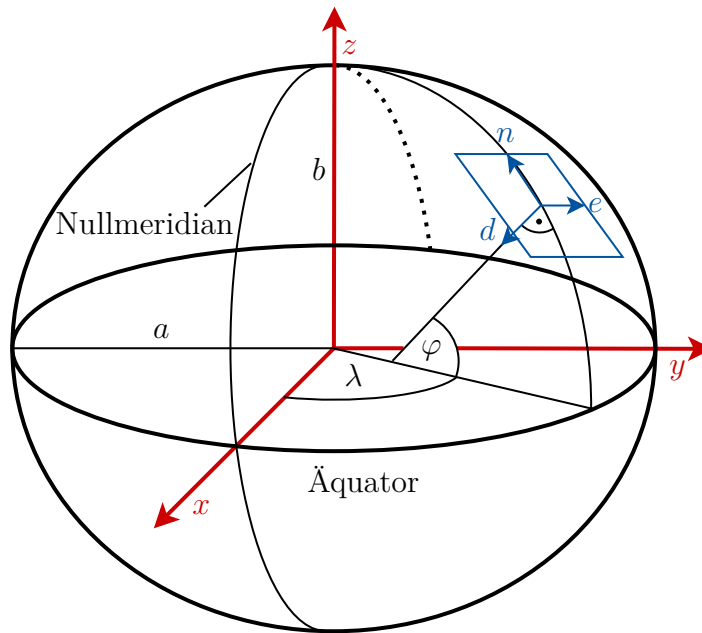


Abb. 2.1: WGS-84-Erdellipsoid nach [51] mit WGS-84-Koordinaten, ECEF-Koordinaten (rot) und lokalem NED-Koordinatensystem (blau)

Die WGS-84-Koordinaten $(\varphi, \lambda, h)^T$ können in die ECEF-Koordinaten $(x, y, z)^T$ umgerechnet werden. Hierzu wird zunächst

$$N(\varphi) = \frac{a}{\sqrt{1 - e^2 \cdot \sin^2(\varphi)}} \quad (2.1)$$

gebildet. Damit kann die Umrechnung gemäß

$$x = (N(\varphi) + h) \cdot \cos(\varphi) \cdot \cos(\lambda) \quad (2.2a)$$

$$y = (N(\varphi) + h) \cdot \cos(\varphi) \cdot \sin(\lambda) \quad (2.2b)$$

$$z = (N(\varphi) \cdot (1 - e^2) + h) \cdot \sin(\varphi) \quad (2.2c)$$

erfolgen. [51]

Bezeichnung	Symbol	Wert
Große Halbachse des WGS-84-Ellipsoids	a	6 378 137 m
Kleine Halbachse des WGS-84-Ellipsoids	$b = a(1 - f)$	6 356 752,3142 m
Abflachung des WGS-84-Ellipsoids	$f = \frac{a-b}{a}$	$\frac{1}{298,257\,223\,563}$
Exzentrizität des WGS-84-Ellipsoids	$e = \sqrt{f(2 - f)}$	0,081 819 190 842 6

Tabelle 2.1: Kennwerte des WGS-84-Ellipsoids [51]

2.2.3 North-East-Down und East-North-Up Koordinatensystem

Zwei weitere Koordinatensysteme sind das North-East-Down Koordinatensystem (NED) und das East-North-Up Koordinatensystem (ENU). Bei beiden handelt es sich um lokale kartesische Koordinatensysteme, deren Nord-Achse (engl. north) n gemeinsam mit der Ost-Achse (engl. east) e die Tangentialebene an den Erdellipsoid aufspannt. Beide Achsen sind in beiden Systemen identisch, wobei sie ihre Positionen tauschen. Bei NED entspricht n der x-Achse und e der y-Achse. Wird ENU verwendet, ist dies genau umgekehrt, sodass n der y-Achse und e der x-Achse entspricht. Beide Koordinatensysteme sind Rechtssysteme. Daher wird die z-Achse, die senkrecht auf der Tangentialebene steht, in beiden Fällen unterschiedlich bezeichnet. Im Fall von ENU wird sie nach oben (engl. up) positiv gezählt und als u genannt. Wird NED verwendet, wird die z-Achse positiv nach unten (engl. down) gezählt und als d bezeichnet. Im weiteren Verlauf der Arbeit wird das NED-Koordinatensystem verwendet. In Abbildung 2.1 ist es das blau eingezeichnete Koordinatensystem.

Um aus WGS-84-Koordinaten NED-Koordinaten zu gewinnen, ist es sinnvoll, zunächst aus LLH die zugehörigen ECEF-Koordinaten gemäß den Gleichungen (2.1)-(2.2c) zu bestimmen. Liegen diese vor, können daraus, durch Verschiebung des Koordinatenursprungs gefolgt von einer Rotation, die gewünschten NED-Koordinaten ermittelt werden. Da beide Systeme kartesische Koordinatensysteme sind, können Rotation und Verschiebung durch einfache mathematische Berechnungen durchgeführt werden. Der neue Koordinatenursprung wird als Referenzpunkt \mathbf{P}_{Ref}^{ECEF} bezeichnet und die Verschiebung kann durch einfache Subtraktion erreicht werden. Für die Rotation muss der verschobene Vektor mit der Rotationsmatrix R multipliziert werden. Um einen beliebigen Punkt \mathbf{P}^{ECEF} in das NED-Koordinatensystem zu transformieren, was durch den Referenzpunkt \mathbf{P}_{Ref}^{ECEF} definiert ist, kann gemäß [14] die Formel

$$\mathbf{P}^{NED} = R \left(\mathbf{P}^{ECEF} - \mathbf{P}_{Ref}^{ECEF} \right) \quad (2.3)$$

verwendet werden. Dabei ist die Rotationsmatrix

$$R = \begin{bmatrix} -\sin(\varphi_{Ref}) \cos(\lambda_{Ref}) & -\sin(\varphi_{Ref}) \sin(\lambda_{Ref}) & \cos(\varphi_{Ref}) \\ -\sin(\lambda_{Ref}) & \cos(\lambda_{Ref}) & 0 \\ -\cos(\varphi_{Ref}) \cos(\lambda_{Ref}) & -\cos(\varphi_{Ref}) \sin(\lambda_{Ref}) & -\sin(\varphi_{Ref}) \end{bmatrix} \quad (2.4)$$

durch die Referenzkoordinaten φ_{Ref} und λ_{Ref} definiert.

Mit den Gleichungen (2.1) bis (2.4) lässt sich eine exakte Koordinatentransformation durchführen, jedoch sind deren Berechnungen vergleichsweise aufwendig. Für die spätere Anwendung müssen neue WGS-84-Koordinaten online in ein lokales NED-Koordinatensystem überführt werden. Damit dies in einem Echtzeitsystem geschehen kann, dürfen die Berechnungen nicht zu komplex sein. Daher werden im Folgenden einige Vereinfachungen getroffen. Zunächst wird auf die Down-Komponente d verzichtet. Die späteren Fahrversuche werden auf nahezu ebenen Teststrecken durchgeführt, was eine Berücksichtigung der Höhe nicht unbedingt erforderlich macht. Die Höhe wird zu Null gesetzt, was zu einem geringen Fehler in den n - und e -Koordinaten führt. Als weitere Vereinfachung

werden Entfernungen nicht in der Nord-Ost-Ebene bestimmt, sondern näherungsweise als Bogenlängen auf dem Ellipsoid. Die exakte Berechnung der Bogenlängen auf dem WGS-84-Ellipsoiden und weitere Eigenschaften werden in [36] ausführlicher beschrieben. Hier wird mit den Berechnungsvorschriften

$$n = b \cdot (\varphi - \varphi_{Ref}) \cdot \frac{\pi}{180^\circ} \quad (2.5a)$$

$$e = a \cdot \cos(\varphi_{Ref}) \cdot (\lambda - \lambda_{Ref}) \cdot \frac{\pi}{180^\circ}, \quad (2.5b)$$

gearbeitet. Bei den Gleichungen (2.5) ist berücksichtigt, dass die Winkel in Grad angegeben sind und entsprechend in Radiant umgerechnet werden müssen. Die Entfernungen in Nord- und Ost-Richtung ergeben sich aus dem Winkelunterschied vom Referenzpunkt zum betrachteten Punkt multipliziert mit dem entsprechenden Radius. Als Radius für die Nord-Komponente n wird der Polradius b verwendet. Der Radius bezüglich der Ost-Richtung nimmt, startend vom Äquator, immer weiter ab, je dichter sich den Polen genähert wird. Am Äquator gilt der Äquator-Radius a und an den Polen beträgt er Null. Das beschriebene Verhalten kann durch Multiplikation von a mit dem Kosinus der Breite φ_{Ref} des Referenzpunktes abgebildet werden, vergleiche (2.5b). Hierbei gilt die Annahme, dass der Referenzpunkt $(\varphi_{Ref}, \lambda_{Ref})$ und der betrachtete Punkt (φ, λ) so dicht beieinander liegen, dass der Radius des Referenzpunktes für beide Punkte verwendet werden kann. Diese Annahme kann ohne weiteres getroffen werden, da bei den im späteren Verlauf durchgeführten Experimenten die maximalen Entfernungen in Bereichen weniger hundert Meter liegen, was im Vergleich zum Äquator-Radius einem Unterschied von vier Größenordnungen entspricht.

Die Berechnungsvorschriften aus (2.5a) und (2.5b) sind deutlich weniger komplex als die zuvor hergeleiteten exakten Umrechnungsvorschriften aus (2.3) unter Verwendung von (2.4), (2.1) und (2.2). Es ergeben sich zwar Ungenauigkeiten, diese sind aber vernachlässigbar. Für das im Rahmen dieser Arbeit verwendete Testgebiet liegt der Fehler im einstelligen cm-Bereich, was bezogen auf die betrachtete Ausdehnung des Testgebiets einer Abweichung von 2,22 ‰ entspricht.

2.2.4 Gleiskoordinaten

Eine weitere Möglichkeit, Positionen anzugeben, ist die Verwendung eines Gleiskoordinatensystems. Dies ist besonders im Schienenverkehr gebräuchlich. Dabei gibt es eindeutig unterscheidbare Gleise. Die Eindeutigkeit kann beispielsweise durch die Nummerierung der Gleisabschnitte erfolgen, sodass die Gleise durch eine Identifikation (ID) identifiziert werden können. Außerdem besitzt jeder Gleisabschnitt einen Anfang und ein Ende. Eine Ortung auf dem Gleis kann dann durch das Tupel (id, s) erfolgen, wobei id die Nummer des Gleises angibt und s der Entfernung vom Gleisanfang zum definierten Punkt in Metern entspricht. Der Vorteil bei der Verwendung von Gleiskoordinaten liegt darin, dass sehr einfach Abstände zwischen zwei Positionen berechnet werden können. Hierfür werden im Wesentlichen die zugehörigen Entfernungswerte s_1 und s_2 voneinander abgezogen.

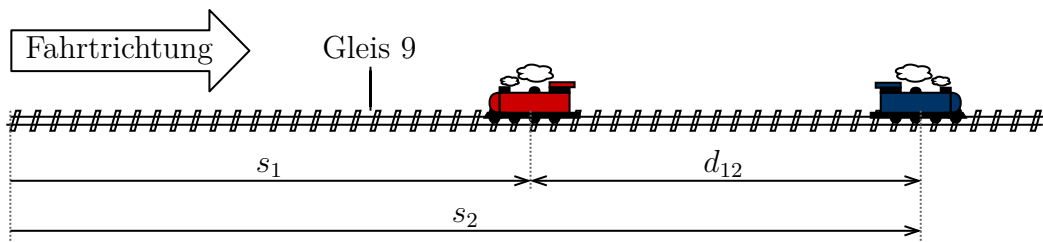


Abb. 2.2: Abstand zweier Lokomotiven zueinander und zum Gleisanfang

In Abbildung 2.2 kann der Abstand der beiden Lokomotiven beispielsweise als

$$d_{12} = s_2 - s_1 \quad (2.6)$$

berechnet werden.

Bei den im späteren Verlauf der Arbeit verwendeten Gleiskoordinaten wird die Fahrtrichtung im Gleiskoordinatensystem mit angegeben. Dazu bekommt die *id* ein Vorzeichen. Eine negative *id* bedeutet, dass das Fahrzeug entgegen der Gleis-Fahrtrichtung orientiert ist. Die Gleisrichtung ist als die Richtung von Gleisanfang zu Gleisende definiert. In Abbildung 2.2 ergeben sich entsprechend $\mathbf{p}_{rot} = (-9, s_1)$ für die rote Lokomotive und $\mathbf{p}_{blau} = (9, s_2)$ für die blaue.

2.3 Globale Satellitennavigationssysteme

Als Globales Navigationssatellitensystem (GNSS) oder auch als Satellitennavigationssystem werden Systeme bezeichnet, die Satelliten zur Ortung verwenden. Üblicherweise umkreisen Navigationssatelliten eines Systems die Erde auf mehreren Orbits und senden von dort Signale auf die Erde. Diese können von GNSS-Empfängern empfangen werden. In den übermittelten Nachrichten ist die Zeit des Absendens enthalten. Die Satelliten besitzen Atomuhren und arbeiten zeitsynchron. Anhand der Ephemeriden, den Bahndaten der Satelliten, die ebenfalls übermittelt werden, sind GNSS-Empfänger in der Lage, aus den Messungen der Satellitensignale die Laufzeit und damit die Entfernung zu den Satelliten zu messen. Basierend auf den Abständen zu den Satelliten kann die eigene Position durch Trilateration bestimmt werden. Da in den Empfängern keine Atomuhren verbaut sind, wird auch die Zeit im Empfänger basierend auf den Messungen korrigiert. Dies führt dazu, dass für die Bestimmung einer dreidimensionalen Position mindestens vier Satelliten benötigt werden. Die Zeit kann als vierte Dimension betrachtet werden, weshalb vier Gleichungen, also vier Satelliten, nötig sind. Die Messungen sind allgemein mit Fehlern behaftet, was zu Ungenauigkeiten in den Gleichungen führt. Werden mehr Nachrichten von mehr Satelliten empfangen, können diese zusätzlichen Informationen, in Form weiterer Gleichungen, die Genauigkeit weiter erhöhen. Neben der Bestimmung der Position können die Satellitenmessungen auch verwendet werden, um Geschwindigkeiten und die genaue Uhrzeit zu bestimmen. Daher ist der englische Ausdruck „Position, Velocity, and Time“ (PVT) für die Positionslösung mittels GNSS gebräuchlich.

Über die Zeit haben sich mehrere weltweit verfügbare Systeme etabliert. Das wohl bekannteste GNSS ist das amerikanische Global Positioning System (GPS), genauer das Navigational Satellite Timing and Ranging (NAVSTAR) - GPS. Die Entwicklung wurde durch das US-Verteidigungsministerium im Jahr 1971 initiiert [51]. Das russische Globale Satellitennavigationssystem (russisch *Globalnaja nawigazionnaja sputnikowaja sistema*) (GLONASS) wurde ebenfalls durch die militärische Nutzung vorangetrieben. Seit einem Erlass der russischen Regierung vom 7. März 1995 wird GLONASS sowohl für die militärische als auch die zivile Nutzung bereitgestellt [24]. Auch die Europäische Union hat ein GNSS aufgebaut. Es trägt den Namen Galileo. Im Jahr 2003 wurde eine Einigung zur Finanzierung durch die Mitgliedsstaaten der Europäische Weltraumorganisation (engl. European Space Agency) (ESA) erzielt und es wurde begonnen, ein eigenständiges europäisches Satellitennavigationssystem unter ziviler Kontrolle aufzubauen. Seit dem Jahr 2016 kann das System teilweise genutzt werden. Mittlerweile ist es voll einsatzfähig. Das chinesische Satellitennavigationssystem trägt den Namen BeiDou, was der chinesische Ausdruck für *großer Bär* ist. Auch BeiDou soll 2020 nach Abschluss der dritten Phase voll einsatzfähig sein [33]. Anders als bei GPS und GLONASS wurde BeiDou zunächst als lokales Satellitennavigationssystem betrieben. Dabei umkreisen die Satelliten die Erde nicht, sondern bewegen sich auf Bahnen, die nur einen bestimmten Teil der Erde abdecken. Um diese Abdeckung zu erreichen, kommen Satelliten auf geostationären Orbits, geneigten geosynchronen Orbits und mittleren Erdumlaufbahnen zum Einsatz [49]. Die geostationären und geneigten geosynchronen Orbits liegen deutlich höher als die mittleren Erdumlaufbahnen, auf denen sich auch die Satelliten von GPS, GLONASS und Galileo bewegen. Auf einem geostationären Orbit befindet sich der Satellit von der Erde aus betrachtet immer an derselben Stelle, da sich der Orbit mit der Erde dreht. Ein Satellit auf einem geosynchronen Orbit folgt, von der Erde aus betrachtet, einem wie die Ziffer 8 geformten Pfad am Himmel. Neben den globalen Satellitennavigationssystemen GPS, GLONASS, Galileo und BeiDou gibt es weitere lokale Systeme, die nur an gewissen Teilen der Erde verwendet werden können. Diese nutzen ebenfalls geostationäre oder geneigte geosynchrone Orbits. Ein Vertreter dieser Variante ist das japanische Quasi-Zenit-Satelliten-System (QZSS), das zur Unterstützung anderer GNSS im Asien-Pazifik-Gebiet verwendet werden kann [53].

Prinzipiell werden Satellitennavigationssysteme in drei Segmente unterteilt. Dabei handelt es sich um das Raumsegment, das Boden- oder auch Kontrollsegment und das Nutzersegment. Alle Einheiten, die sich im Weltraum befinden, sind im Raumsegment vereint. Dies sind im Wesentlichen die Satelliten. Als Komponenten des Kontrollsegments werden alle Einrichtungen, die am Boden zum Betrieb des Navigationssystems benötigt werden, bezeichnet. Dazu zählt das Kontrollzentrum genauso wie Messstationen, die der Überwachung der von den Satelliten ausgesendeten Signale dienen. Das letzte Segment ist das Nutzersegment, in dem alle Empfangsgeräte zusammengefasst werden.

2.3.1 Der NMEA-0183 Standard

Im maritimen Bereich hat es sich die US-amerikanische Nationale Vereinigung für Marineelektronik (engl. National Marine Electronics Association) (NMEA) zur Aufgabe gemacht, Standards für die Schifffahrt zu fördern [2]. Ein weit verbreiteter Standard ist der NMEA-0183-Standard. Dieser empfiehlt die Nutzung der RS-422-Schnittstelle [2], lässt aber auch die RS-232-Schnittstelle zu. Viele GNSS-Empfänger besitzen eine RS-232-Schnittstelle und übertragen ihre Messwerte im NMEA-0183-Format.

Der NMEA-0183-Standard sieht es vor, dass die Daten in Datensätzen (englisch *sentences*) übertragen werden. Die NMEA-Datensätze werden aus menschenlesbaren ASCII-Zeichen aufgebaut. Es folgt ein Auszug einer GPS-Messung vom 5. September 2018 um 13:05:10 Uhr:

```
$GPRMC,130510.00,A,5054.55032,N,00613.81654,E,0.0,,050918,1.7,W,D*06
$GPGGA,130510.00,5054.55032,N,00613.81654,E,2,11,0.9,123.12,M,47.57,M,2.0,0123*41
$GPGSA,A,3,10,03,11,27,01,28,32,08,18,22,14,,1.6,0.9,1.3*37
$GPGST,130510.00,1.439,1.614,1.182,14.437,1.590,1.213,2.625*6C.
```

Der Aufbau der Datensätze folgt immer dem gleichen Schema. Ein neuer Datensatz beginnt mit einem \$-Zeichen, ist maximal 80 Zeichen lang und endet mit einem Zeilenumbruch in Windows-Encodierung. Der Zeilenumbruch in Windows-Encodierung besteht aus einem *carriage return*-Zeichen gefolgt vom *line feed*-Zeichen. Die Daten werden in durch Kommas voneinander getrennten Feldern eingetragen. Das erste Feld beschreibt, um welche Nachricht es sich handelt. Dazu werden üblicherweise fünf Buchstaben verwendet, wobei die ersten beiden als Talker-ID bezeichnet werden und definieren, welcher Typ von Sensor bzw. welches Messverfahren eingesetzt wird. Die restlichen drei Buchstaben geben an, welcher Nachrichtentyp verwendet wird. Daraus ergibt sich, wie alle folgenden durch Kommas getrennten Daten zu interpretieren sind. In vielen Nachrichten wird eine Checksumme zur Validierung der Daten angefügt. Hierzu wird an das letzte Datum vor dem Zeilenumbruch ein *-Zeichen angefügt, gefolgt von zwei Hexadezimalzeichen, die der Byte-Repräsentation des Ergebnisses einer Exklusiv-Oder-Operation aller Zeichen zwischen \$ und * entsprechen. Proprietäre Nachrichten sind auch vorgesehen. Diese besitzen eine vier Zeichen lange Talker-ID, die mit einem P (für proprietär) beginnt, gefolgt von einer drei Zeichen langen Abkürzung der Herstellerbezeichnung. Die weiteren Felder können dann frei vom Hersteller belegt werden. [16]

Für einzelne Satellitennavigationssysteme sowie für die Kombination mehrerer Systeme existieren die folgenden Talker-IDs:

- GA $\hat{=}$ Galileo,
- GL $\hat{=}$ GLONASS,
- GP $\hat{=}$ GPS.
- GB $\hat{=}$ BeiDou,
- GN $\hat{=}$ System Mix,

Im folgenden Abschnitt werden einige für diese Arbeit relevante NMEA-Datensätze vorgestellt. Da der NMEA-Standard englischsprachig ist, werden im Folgenden die englischen Datensatzbezeichnungen verwendet. Die Datensätze sind Recommended Minimum

Sentence C (RMC), Global Positioning System Fix Data (GGA), GNSS DOP and Active Satellites (GSA), GNSS Pseudorange Error Statistics (GST) und Real-time Position with Coordinate Quality (GGQ). Dies entspricht den Datensätzen aus dem Auszug der GPS-Messung, wobei kein GGQ-Datensatz vorhanden ist. Nach den Beobachtungen des Autors wird nur einer der beiden Datensätze GST oder GGQ auf einem Empfänger umgesetzt.

Alle hier betrachteten Datensätze werden vom IRT-Buggy, siehe Abschnitt 2.8, decodiert und stehen zumindest in Teilen an einer Fahrzeugschnittstelle zur Verfügung. In der Beschreibung der Datensätze wird nur auf die von den IRT-Buggys dekodierten Werte eingegangen. Eine ausführlichere Beschreibung mit allen enthaltenen Daten kann in [13] sowie [16] gefunden werden.

Recommended Minimum Sentence C (RMC)

Der RMC-Datensatz enthält die Messwerte zu einer Positionsbestimmung des GNSS-Empfängers. Diese beinhalten den Breitengrad φ und den Längengrad λ basierend auf dem WGS-84-Ellipsoid, den Zeitpunkt der Messung, die aktuelle Geschwindigkeit aufgeteilt in Betrag in Knoten und Winkel in Grad sowie weitere Werte.

Global Positioning System Fix Data (GGA)

In dem GGA Datensatz sind weitere Messdaten der Positionsbestimmung enthalten. So kann hier neben Längen- und Breitengrad auch die Höhe h über dem WGS-84-Ellipsoid entnommen werden. Außerdem gibt es eine Angabe darüber, wie viele Satelliten der Empfänger empfängt und welche Methode zur Positionsbestimmung herangezogen wird. Daraus lassen sich Rückschlüsse auf die erreichbare Positionsgenauigkeit ziehen.

GNSS DOP and Active Satellites (GSA)

Der GSA Datensatz gibt an, welche Satelliten zur Positionsbestimmung verwendet wurden. Außerdem wird mit mehreren Dilution of Precision (DOP)-Werten angegeben wie gut sich die aktuelle Satellitenkonstellation überhaupt zur Positionsbestimmung eignet. Verfügbar sind HDOP (horizontal dilution of precision), VDOP (vertical dilution of precision) und PDOP (position dilution of precision).

GNSS Pseudorange Error Statistics (GST)

In dem GST-Datensatz sind die 1 Sigma Fehler der Positionsbestimmung in Metern entlang des Längen- und Breitengrades enthalten.

Real-time Position with Coordinate Quality (GGQ)

Der GGQ-Datensatz enthält eine Genauigkeitsangabe über den Fehler in der 2D- sowie 3D-Positionsbestimmung in Metern.



Abb. 2.3: GO! Logo links (a) und Projektkonsortium rechts (b) [56]

2.3.2 Echtzeitkinematik (RTK)

Unter dem Begriff Echtzeitkinematik (engl. Real Time Kinematic) (RTK) wird die genaue Positionsbestimmung mithilfe eines Rovers und einer oder mehrerer Referenzstationen verstanden. Der Rover ist ein beweglicher GNSS-Empfänger, der eine Kommunikationsverbindung zur Referenzstation benötigt. Über diesen Kommunikationskanal sendet die Referenzstation Daten, anhand derer der Rover seine Positionsmessung korrigieren und somit verbessern kann. Das Verfahren erreicht Positionsgenauigkeiten im Zentimeterbereich. Ähnlich der NMEA gibt es die amerikanische Funktechnische Kommission für Seeverkehrsdienste (engl. Radio Technical Commission for Maritime Services) (RTCM), die in dem Sonderausschuss (engl. Special Committee) Nummer 104 (SC-104), einen Standard zur Korrekturdatenübertragung, empfiehlt. Die Bezeichnung des Standards lautet RTCM SC-104, wobei im Folgenden die Bezeichnung RTCM verwendet wird. Um Korrekturdaten im RTCM-Format zu übertragen, kann ein vernetzter Transport von RTCM über das Internetprotokoll (engl. Networked Transport of RTCM via Internet Protocol) (Ntrip) verwendet werden. Ein Ntrip-Anbieter in Deutschland ist SAPOS®, der bei den Messungen im Rahmen dieser Arbeit Anwendung findet. [48]

2.4 Der Galileo Online Empfänger

Galileo Online (GO!) ist ein vom Bundesministerium für Wirtschaft und Energie aufgrund eines Beschlusses des Deutschen Bundestages gefördertes Projekt, in dem ein Navigationsempfänger für Bahn-Anwendungen entwickelt wird. Es wird ein Empfänger speziell für Bahnanwendungen erstellt, der die Anforderungen aus dem Bahnsektor erfüllen soll. Die Abbildung 2.3a zeigt das Projektlogo. Das Projekt wird von dem

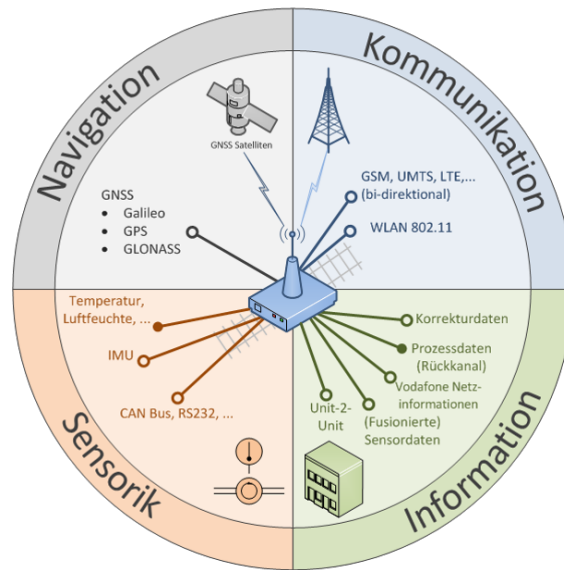


Abb. 2.4: GO!-Empfänger Konzept [56]

Konsortium bestehend aus dem IRT, dem Institut für Mobil- und Satellitenfunktechnik GmbH (IMST), dem Fraunhofer-Institut für Integrierte Schaltungen (IIS), der SCISYS Deutschland GmbH, der Vodafone GmbH, dem Lehrstuhl für Allgemeine Elektrotechnik und Datenverarbeitungssysteme der RWTH Aachen University (EECS) und der Innovationszentrum für Mobilität und gesellschaftlichen Wandel GmbH (InnoZ) in der Zeit von April 2015 bis Juli 2018 durchgeführt. In Abbildung 2.3b sind unter den einzelnen Partnern deren Kernaufgaben im Projekt angegeben.

Der zu entwickelnde GO!-Empfänger soll seine Stärken nicht nur im Bereich der Navigation ausspielen. Abbildung 2.4 zeigt die vier Bereiche, die durch den GO!-Empfänger abgedeckt werden. Im Bereich der Navigation wird die Genauigkeit des Positionssignals durch die Kombination mehrerer GNSS-Systeme und die Nutzung von Augmentierungsdiensten erhöht. Hierzu ist der Empfänger in der Lage, mehrere, GNSS-Signale unterschiedlicher Systeme zu empfangen und zu verarbeiten. Ebenso ist er fähig, Augmentierungsdaten zu verwenden, die von Diensteanbietern, wie z. B. SAPOS® (Satellitenpositionierungsdienst der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland), bereitgestellt werden. Der zweite Bereich aus Abbildung 2.4 ist die Kommunikation. Dazu gehört die starke Vernetzungsfähigkeit, die durch optimierte Kommunikationslösungen erreicht wird. Diese werden über die eingebauten Kommunikationskanäle, wie drahtloses lokales Netzwerk (engl. Wireless Local Area Network) (WLAN) und Long Time Evolution (LTE), bereitgestellt. So ist der Empfänger in der Lage, die sogenannten *zentralen Dienste* direkt in die Anwendung einzubeziehen. Diese sind als Serviceplattform Teil des dritten Bereichs, Information, aus Abbildung 2.4. Die zentralen Dienste sind nicht nur mit den Empfängern vernetzt, sondern können auch an diverse öffentliche und/oder private Dienste angebunden werden. So sind die zentralen Dienste in der Lage, Mehrwertdienste zu realisieren. D. h. systemeigene und externen Daten können fusioniert und prozessiert werden, um Mehrwerte zu schaffen. Dabei kann auch der letzte Bereich, Sen-

sorik, aus Abbildung 2.4 mit einbezogen werden. Der Empfänger kann die Daten von integrierter oder extern angeschlossener Sensorik an die zentralen Dienste übertragen. Die Messwerte kann er aber auch selbst verwenden, um beispielsweise die Verfügbarkeit des Positionssignals zu verbessern oder eine Aussage über die Integrität des erzeugten Positionssignals zu erstellen. Weitere Informationen zur Sensorfusion, die im Empfänger durchgeführt wird, und der verwendeten Softwarearchitektur können in [12, 28] und [56] gefunden werden.

Einige Anwendungsbeispiele der zentralen Dienste im Projekt GO! sind die folgenden:

- Zugvollständigkeitskontrolle mit mehreren Empfängern,
- Detektion von Schadstellen im Gleis bzw. Schäden am Fahrzeug,
- Automatisierung von Rangiervorgängen sowie
- vorausschauende Wartung.

Zur Realisierung der vorgenannten Beispiele werden in den zentralen Diensten an sie übermittelte Daten entsprechend verarbeitet und Auswertungen vorgenommen. Beispielsweise können bei der Schadstellenerkennung die Messwerte eines Beschleunigungssensors über das Internet von den zentralen Diensten angezeigt und abgerufen werden. Als Ergebnisse sind hier die Positionen beschädigter Trassen denkbar. Im Fall der Rangiervorgänge werden von den zentralen Diensten Rangiermanöver bestimmt, die an den Empfänger zurück gespielt werden, um dort von einer lokalen Recheneinheit auf dem Fahrzeug umgesetzt zu werden. Eine ausführlichere Beschreibung des Projektes GO! mit allen Zielen und Ergebnissen findet sich im Projektabschlussbericht in [56].

An dieser Stelle wird detaillierter auf den internen Hardwareaufbau eingegangen, da dieser eine wesentliche Rolle im Rahmen dieser Arbeit einnimmt. Er hebt die in Abschnitt 2.8 beschriebenen Einschränkungen der vorhandenen Kommunikationskanäle der IRT-Buggys auf und ermöglicht dadurch die Vernetzung aller acht IRT-Buggys. Hier wird zunächst der Hardwareaufbau des Empfängers beschrieben, der nötig ist, damit der Empfänger seine Funktionalität im Rahmen des Projektes GO! erfüllt.

Abbildung 2.5 zeigt den internen Aufbau des GO!-Empfängers. Dieser setzt sich neben peripheren Komponenten zur Stromversorgung und der internen Verkabelung aus zwei Hauptkomponenten zusammen. Diese sind ein Smart Feature Cellular Device (SFCD) und eine Entwicklungsplattform für Satellitennavigation des IIS (GOOSE).

Die GOOSE bietet eine offene Softwareschnittstelle zur Satellitennavigationsempfängerentwicklung. Zusammen mit dem auf ihr montierten IMST-Frontend stellt sie den eigentlichen Navigations-Empfänger dar. Das Frontend empfängt die analogen Satellitensignale, digitalisiert diese und stellt sie der GOOSE zur Verfügung. Diese verwendet die enthaltenen Code- und Trägerphasen sowie die übertragenen Daten zur Bestimmung der PVT-Lösung. Mittels Korrekturdaten wird die Position weiter verbessert. Diese Korrekturdaten werden wie bereits erwähnt von SAPOS[®] bezogen. Um auf der GOOSE eine Sensorfusion durchführen zu können, werden Daten einer Inertialen Messeinheit (engl. Inertial Measurement Unit) (IMU) benötigt. Diese können entweder mittels Controller

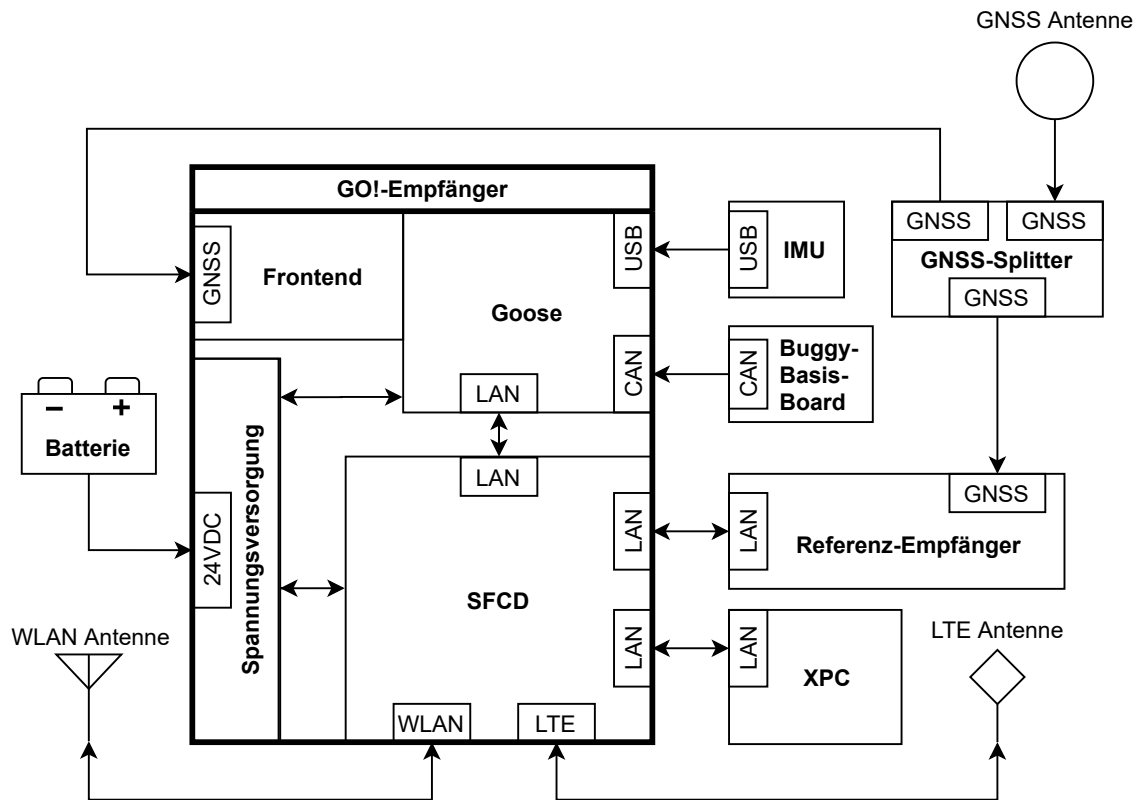


Abb. 2.5: Aufbau des GO!-Empfängers mit seiner Peripherie

Area Network (CAN)-Bus oder per Universal Serial Bus (USB) an die GOOSE übertragen werden. [56]

Das SFCD stellt der GOOSE diese Korrekturdaten bereit. Es ist ein PC-System mit verbauter LTE-Empfangstechnik. Diese ermöglicht einerseits einen Datenaustausch mittels LTE, andererseits werden aber auch detailliertere Netzwerkstatusinformationen bereitgestellt. Dazu gehören beispielsweise die Kennung der verbundenen Sendemasten und deren empfangene Signalstärken. Dank drei verbauter Netzwerkadapter sowie eines WLAN-Adapters ist das SFCD in der Lage, auf unterschiedlichen Wegen Verbindungen aufzubauen. Wie in Abbildung 2.5 zu erkennen ist, wird eine der Netzwerkverbindungen verwendet, um darüber die GOOSE mit dem SFCD zu verbinden. Die beiden anderen Netzwerkverbindungen sind nach außen geführt und können von externen Komponenten verwendet werden. Im Rahmen des Projektes GO! wird eine dieser Verbindungen an den XPC der IRT-Buggys verbunden, vergleiche Abschnitt 2.8. So ist die Empfängersoftware, die auf dem SFCD läuft, in der Lage, Fahrbefehle an die IRT-Buggys zu kommandieren. Die letzte Schnittstelle steht für den Aufbau einer Internetverbindung von Validationssensorik, wie beispielsweise dem RTK-Navigationsempfänger der Firma Septentrio, bereit. Diese Internetverbindung wird über die LTE-Verbindung des SFCD aufgebaut und kann unabhängig von den Daten des GO!-Empfängers betrieben werden. Der Referenzempfänger kann also seine eigenen Korrekturdaten beziehen oder er wird mit den

Daten von den zentralen Diensten beliefert. In Abbildung 2.5 ist dargestellt, dass nicht zwei GNSS-Antennen verbaut sind, sondern nur eine, deren Signal durch einen Splitter an den GO!- und den Referenz-Empfänger weitergeleitet wird. So können die Messwerte besser miteinander verglichen werden, da sie auf denselben Signalen beruhen.

Sowohl auf der GOOSE als auch auf dem SFCD kommt das Linux-Betriebssystem Ubuntu zum Einsatz, sodass für Linux eine verfügbare Netzwerksoftware zur Verfügung steht. Mit dieser ist es möglich, alle GO!-Empfänger respektive die SFCDs in ein Virtuelles Privates Netzwerk (VPN) zu verbinden. Die Verbindung zum VPN-Server, der aus dem Internet erreichbar ist, kann dabei entweder mittels WLAN oder LTE hergestellt werden. Im VPN können die Empfänger dann untereinander und mit den zentralen Diensten kommunizieren. In Abschnitt 5.1.1 wird detaillierter auf diese Kommunikationsflüsse eingegangen.

2.5 Kartendaten und Map-Matching

Das Map Matching wird verwendet, um aus den Satellitenmessungen die Position eines Fahrzeuges in Gleiskoordinaten abzuleiten. Hierzu werden im Rahmen dieser Arbeit zwei Verfahren gebraucht. Das erste Verfahren wird bei dem im Abschnitt 2.9 vorgestellten Regler eingesetzt und stammt aus [37]. Es ist robust, bringt aber auch Probleme mit sich, auf die später, bei der Vorstellung des Verfahrens in diesem Abschnitt, eingegangen wird. Die Probleme machen das Verfahren für den in Abschnitt 4.3 vorgestellten Regler unbrauchbar. Daher wird ein zweites Verfahren implementiert und für diesen Regler verwendet. Im Folgenden werden beide Verfahren vorgestellt. Zunächst wird aber beschrieben, in welcher Form die Karten vorliegen und wie die Kartendaten aufgearbeitet werden, um sie online verwenden zu können. Dieser Schritt ist für beide Verfahren notwendig und daher identisch.

Die Kartendaten liegen im OpenStreetMap (OSM)¹-Format vor. Das OSM-Format basiert auf der Erweiterbaren Auszeichnungssprache (engl. Extensible Markup Language) (XML), in der Objekte mit Attributen, den sogenannten *Tags*, gespeichert werden. Die in diesem Kontext wichtigsten Objekte sind *Knoten* (engl. Nodes) und *Wege* (engl. Ways). Knoten enthalten unter anderem eine ID und einen WGS-84-Koordinatensatz, der die Position des Knotens angibt. Wege setzen sich aus einer Auflistung von Knoten(-IDs) zusammen und werden aus der Abfolge dieser Knoten definiert, wobei sich die Richtung des Weges vom ersten Knoten zum letzten ergibt. Zusätzlich können Tags hinzugefügt werden, die weitere Informationen enthalten. Eine solche Information ist der Wegtyp. Mögliche Wegtypen sind beispielsweise Landstraße, Fahrstreifen einer Autobahn, Gleis und weitere.

Das OSM-Format ermöglicht keine sinnvolle echtzeitfähige Bereitstellung der enthaltenen Informationen. Daher werden die Daten konvertiert und in Matrizen abgelegt. Die

¹OpenStreetMap ist ein freies Projekt, das frei nutzbare Kartendaten in einer Datenbank vorhält (<https://openstreetmap.org>).

in einer OSM-Datei enthaltenen Wege werden hierzu von eins beginnend der Reihe nach durchnummeriert. Anschließend werden Matrizen, deren erste Dimension genau der Anzahl der vorhandenen Wege entspricht, erstellt. So kann die Wegnummer als Index für den entsprechenden Weg verwendet werden. In diesen Matrizen werden unterschiedliche Informationen gespeichert. Diese sind die Anzahl Knoten entlang eines Weges, Verzweigungen zu Beginn und Ende des Weges, Weglänge und weitere. Die Informationen, die über Knoten bekannt sind, werden in sortierten Vektoren gespeichert. Die Sortierung gibt vor, dass zunächst nach Wegen und dann nach der Position innerhalb des Weges sortiert wird. Außerdem werden Vektoren erstellt, in denen die Anzahl der Knoten jedes Weges sowie die kumulierte Summe aller Knoten der Wege vor dem aktuellen Weg enthalten sind. Daraus können also die Indizes, unter denen die Informationen der Knoten eines Weges zu finden sind, bestimmt werden. Die wichtigsten Informationen zu den Knoten sind die WGS-84-Koordinaten (ohne Höhe $\hat{=}$ $h = 0$) und die Länge von Beginn des Weges zum aktuellen Knoten. Um die Weglänge zu bestimmen, werden die WGS-84-Koordinaten gemäß den Gleichungen (2.5a) und (2.5b) in NED-Koordinaten überführt, wobei der Startpunkt des Weges als Referenzpunkt gewählt wird. Dort wird anschließend die euklidische Distanz zwischen aufeinander folgenden Knoten bestimmt. Durch kumulative Summierung dieser Entfernungen entlang des Weges, vom Start zum Ende, ergibt sich für jeden Knoten die Entfernung zum Start des Weges. Diese ist eine wichtige Information bei der Bestimmung von Gleiskoordinaten.

Am Anfang und Ende eines Weges ist eine Verzweigung möglich, beispielsweise an einer Kreuzung. Werden die Kartendaten, wie im Rahmen dieser Arbeit, auch zur Trajektoriengenerierung verwendet, ist eine fixe Zuordnung, wie sich an Kreuzungen verhalten werden soll, nötig. Diese Zuordnung wird in weiteren, den Wegen zugehörigen Vektoren gespeichert. Die Information, welche Wege sich an einer Kreuzung treffen, ist ebenfalls in Matrizen abgelegt. Im Rahmen dieser Arbeit wird für die Initialwerte der Vektoren immer der Weg gewählt, der die geringste Winkeländerung mit sich bringt. Bei einer normalen Straßenkreuzung würde demnach geradeaus gefahren werden. Die Zuordnung kann zur Laufzeit angepasst werden, um so ein Abbiegen an einer Kreuzung zu realisieren oder im Rahmen dieser Arbeit das Stellen von Weichen abzubilden.

Neben den Kartendaten dienen dem Map Matching-Verfahren aus [37] auch Informationen über die zurückgelegte Strecke als Eingang. Sie werden mit den Kartendaten korreliert. So werden nur Strecken ausgewählt, deren Verlauf eine gewisse Ähnlichkeit mit den letzten Fahrmanövern aufweisen. Dabei werden auch durchfahrene Krümmung und Orientierung berücksichtigt. Es kommt nur zu einem *Matching* (Abbildung auf die Karte), wenn Fahrdaten und Karte zusammenpassen. Diese Betrachtungen machen die Robustheit und Komplexität des Verfahrens aus. Aufgrund der Robustheit ist das Verfahren gut geeignet, wenn ein herkömmlicher Empfänger verwendet wird, der Genauigkeiten im ein- bis zweistelligen Meter-Bereich aufweist. Um echtzeitfähig zu bleiben, wird nicht zu jedem Zeitschritt der komplette Abgleich durchgeführt. Er wird mit einer Rate von 5 Hz erneut gestartet. Dies entspricht auch der Rate, mit der im Rahmen dieser Arbeit GNSS-Messdaten durch einen GNSS-Empfänger bereitgestellt werden. Das bedeutet alle 0,2s wird eine Position bestimmt. Zu den anderen Zeiten wird die Position durch Integration über die Geschwindigkeit geschätzt. Dabei wird die Annahme getroffen, dass

sich das Fahrzeug exakt entlang der Strecke bewegt. D. h. es wird die zurückgelegte Strecke gemessen und entlang der Karte aufgetragen. Wird aber beispielsweise mit einem Winkel von der Strecke aus der Karte weg gefahren, müsste die neue Position dichter an der ursprünglichen Position liegen. Dies ist aber systembedingt nicht der Fall, was dazu führt, dass die Position während der fortgeführten Integration abdriften kann. Sie wird zwar immer wieder zurückgesetzt, kann aber in der Zwischenzeit Abweichungen aufweisen und bei neuen Messwerten zurück springen. Dies ist für den Regler aus Abschnitt 4.3 problematisch und macht das Verfahren damit ungeeignet für den Regler.

Um das Problem der driftenden Position zu vermeiden, wird ein eigenes Vorgehen zum Map Matching implementiert, das nun vorgestellt wird. Bei dem Verfahren wird das Map Matching zu jedem Zeitpunkt neu durchgeführt, wobei für die Zeiten zwischen zwei Satellitenmessungen die integrierte Position aus einem Navigationsfilter verwendet wird. Der Navigationsfilter berücksichtigt bei der Positionsbestimmung zusätzlich die Orientierung, die Drehrate sowie weitere Fahrzeugmessdaten und stellt eine WGS-84-Position und eine Orientierung mit einer Rate von 100 Hz zur Verfügung. Dies ist eine deutlich höhere Rate als die 5 Hz der GNSS-Positionsmessung. Es wird davon ausgegangen, dass die GNSS-Messungen mit RTK durchgeführt werden. So kann auch die Genauigkeit der geschätzten Position im Bereich weniger Zentimeter liegen. Details zu dem Navigationsfilter können in [5] und [38] gefunden werden.

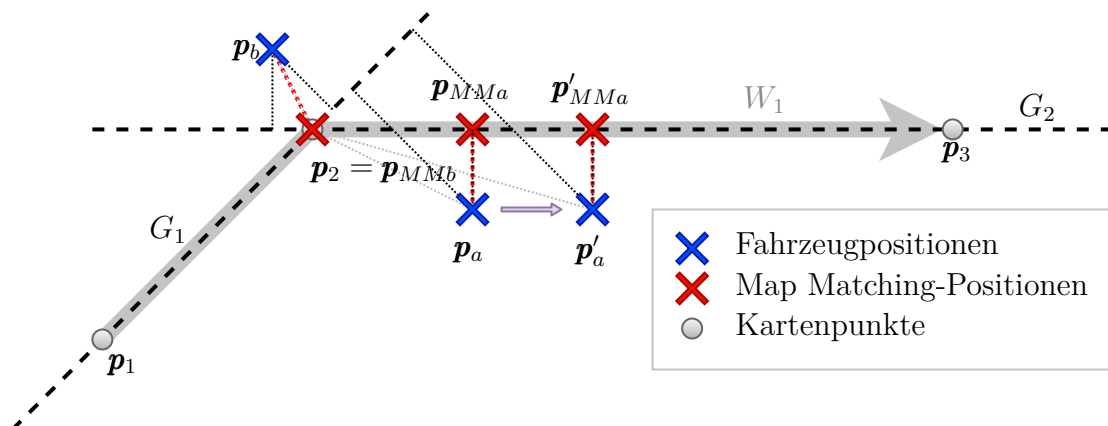


Abb. 2.6: Positionsbestimmung mittels Map Matching

Abbildung 2.6 zeigt wie aus den Kartendaten und Positionen die Gleiskoordinaten bestimmt werden. Die Punkte \mathbf{p}_1 - \mathbf{p}_3 stellen dabei drei Knoten des Weges W_1 (grauer Pfeil) dar, der von Punkt \mathbf{p}_1 zu Punkt \mathbf{p}_3 verläuft. Die Positionen \mathbf{p}_a und \mathbf{p}_b sollen durch das Map Matching in Gleiskoordinaten überführt werden. Alle Koordinaten liegen zunächst als WGS-84-Koordinaten vor und werden entsprechend mittels (2.5a) und (2.5b) unter Verwendung des selben Referenzpunktes in NED-Koordinaten überführt. Anschließend wird der Punkt des Weges bestimmt, der den geringsten euklidischen Abstand (rote Linien) zur Ausgangsposition aufweist. Dazu wird jeweils das Lot (schwarze gepunktete Linien in Abbildung 2.6) auf die Geraden G_1 bzw. G_2 (schwarze gestrichelte Linien), die sich aus je zwei aufeinander folgenden Punkten ergeben, bestimmt. Zunächst werden die

NED-Koordinaten \mathbf{p}_1^n , \mathbf{p}_2^n und \mathbf{p}_3^n bestimmt. Ausgehend von diesen Koordinaten werden anschließend die Vektoren

$$\mathbf{v}_{i,a} = \mathbf{p}_a^n - \mathbf{p}_i^n \quad \text{für } i \in \{1, 2, 3\}. \quad (2.7)$$

zum Punkt a berechnet. Dabei ist i die Laufvariable für die drei Punkte. Der Pfad, der durch die drei Punkte gegeben ist, kann durch die zwei Karten-Vektoren

$$\mathbf{v}_{j,j+1} = \mathbf{p}_{j+1}^n - \mathbf{p}_j^n \quad \text{für } j \in \{1, 2\} \quad (2.8)$$

mit der Laufvariable j über die Anzahl der zwei Geraden dargestellt werden. Anschließend wird das normierte Skalarprodukt

$$\lambda_j = \frac{\langle \mathbf{v}_{j,a}, \mathbf{v}_{j,j+1} \rangle}{\|\mathbf{v}_{j,j+1}\|} = \frac{\mathbf{v}_{j,a}^T \cdot \mathbf{v}_{j,j+1}}{\|\mathbf{v}_{j,j+1}\|} \quad \text{für } j \in \{1, 2\} \quad (2.9)$$

zwischen den gefundenen Vektoren $\mathbf{v}_{i,a}$ für a und den beiden Karten-Vektoren $\mathbf{v}_{j,j+1}$ berechnet. Fällt ein Lot dabei auf den Teil einer Geraden, der nicht Teil des Weges ist, muss dieses verworfen werden. Durch die Normierung in (2.9) kann dies daran erkannt werden, dass das gefundene λ_j einen Wert außerhalb des geschlossenen Intervalls $[0, 1]$ annimmt. Dieser ist kleiner 0 für Punkte vor dem Anfang und größer 1 für Punkte hinter dem Ende. Um diese Punkte zu verwerfen, wird λ_j im Anschluss an die Berechnung mittels

$$\tilde{\lambda}_j = \begin{cases} 0 & \text{für } \lambda_j < 0 \\ 1 & \text{für } \lambda_j > 1 \\ \lambda_j & \text{sonst} \end{cases} \quad \text{mit } j \in \{1, 2\} \quad (2.10)$$

beschränkt. So werden anstelle der ungültigen Punkte entweder der Start- oder Endpunkt in der Suche berücksichtigt. In Abbildung 2.6 fällt das Lot von Position a auf den gestrichelten Teil der Geraden von Punkt \mathbf{p}_1 zu \mathbf{p}_2 ($\mathbf{v}_{1,2}$), daher wird Punkt \mathbf{p}_2 durch das Setzen von $\tilde{\lambda}_1 = 1$ mit in die Suche einbezogen (graue gepunktete Linien Abbildung 2.6). Der Abstand für die gefundenen Lotfußpunkte wird mit dem Kreuzprodukt bestimmt. Dabei ist zu berücksichtigen, dass dies für Punkte, bei denen λ_j beschränkt wurde, nicht gilt. In die Berechnung würden Punkte außerhalb des Weges einfließen. Der so ermittelte Abstand wäre zu kurz, vergleiche grau und schwarz gepunktete Linien in der Abbildung. Für diese Punkte wird der Abstand des jeweiligen Start- oder Endpunktes der Geraden explizit berechnet. Es ergeben sich für die Entfernungen

$$d_j = \begin{cases} \|\mathbf{v}_{j,a}\| & \text{für } \lambda_j < 0 \\ \|\mathbf{v}_{j+1,a}\| & \text{für } \lambda_j > 1 \\ \frac{\|\mathbf{v}_{j,a} \times \mathbf{v}_{j,j+1}\|}{\|\mathbf{v}_{j,j+1}\|} & \text{sonst} \end{cases} \quad \text{für } j \in \{1, 2\} \quad (2.11)$$

zur Karte. Im nächsten Schritt wird

$$j_{min} = \arg \min_{j \in \{1, 2\}} d_j \quad (2.12)$$

bestimmt, welches die geringste Entfernung zum gesamten Weg repräsentiert. Das so gefundene j_{min} bzw. das zugehörige $d_{j_{min}}$ entspricht der kürzesten Entfernung zu den Kartendaten (rote Linien) für den untersuchten Punkt. Aus j_{min} ergibt sich die Zuordnung zu einem Wegabschnitt und damit die Koordinate id der Gleiskoordinaten. Um die Entfernung s zum Startpunkt des Weges zu bestimmen, wird der Abstand s_λ vom Geradenstartpunkt \mathbf{p}_j zum Lotfußpunkt \mathbf{p}_{MM} benötigt. Unter Verwendung von $\tilde{\lambda}_{j_{min}}$ kann der Abstand als

$$s_\lambda = \tilde{\lambda}_{j_{min}} \cdot \|\mathbf{v}_{j_{min}, j_{min}+1}\| \quad (2.13)$$

bestimmt werden. Damit ergibt sich die zweite Koordinate der Gleiskoordinaten s als

$$s = s_{p, j_{min}} + s_\lambda, \quad (2.14)$$

wobei $s_{p, j}$ den in den Vektoren gespeicherten Abständen zum Startpunkt des Weges für die Punkte \mathbf{p}_j entspricht. Somit lässt sich jede Eingangsposition durch das Map Matching in Gleiskoordinaten der Form (id, s) überführen, wobei die Orientierung noch unbekannt ist.

Zur Bestimmung der Orientierung werden die Zeit und das Vorzeichen der Geschwindigkeit mit einbezogen. In Abbildung 2.6 wird mit der Zeit aus Position a die Position a' . Die Position wird durch eine Vorwärtsbewegung (positives Vorzeichen der Geschwindigkeit) erreicht. Dies wird durch den Pfeil in der Abbildung angedeutet. In den Gleiskoordinaten der Map Matching-Position von $\mathbf{p}_{a'}$ wächst s' an, was einer Vorwärtsbewegung in Gleiskoordinaten entspricht. Da reale Bewegung und Bewegung in Gleiskoordinaten übereinstimmen, wird das Vorzeichen von id positiv gewählt. Würden sich die Bewegungen widersprechen, würde das Vorzeichen von id negativ gewählt werden. Ist die Orientierung, das Vorzeichen von id , einmal gefunden, wird sie beibehalten.

Es ist auch möglich, dass das Map Matching keine Position zurückliefert. Bei dem vorgestellten Verfahren wird, wie zuvor beschrieben, unter anderem der Abstand zwischen den Wegen der Kartendaten und dem untersuchten Eingangspunkt berechnet. Wird dieser Abstand zu groß, im Rahmen der Arbeit größer als 10 m, wird keine gültige Map Matching-Position zurückgegeben. Konnte mindestens einmal eine Position ermittelt werden, wird der Suchraum eingeschränkt, um die Rechenzeit zu reduzieren. Dadurch werden nur noch die Wege betrachtet, die vom aktuellen Weg zu erreichen und weniger als 10 m vom aktuell gefundenen Map Matching-Punkt entfernt sind.

In der Literatur gibt es noch andere Ansätze. Für den Anwendungsbereich dieser Arbeit ist das vorgestellte simple Verfahren aber ausreichend. Daher wird nicht weiter auf andere Verfahren eingegangen.

2.6 Regelungstechnische Grundlagen

In diesem Abschnitt werden die für diese Arbeit wichtigen Aspekte der Regelungstechnik kurz beleuchtet. Dabei liegt der Fokus auf den in dieser Arbeit verwendeten bzw. entwickelten Reglern. Zunächst wird aber eine geeignete Modellierung eines zu steuernden bzw.

regelnden Systems benötigt. Daher wird in Abschnitt 2.6.1 auf den Zustandsraum als eine mögliche Modellform eingegangen. Sowohl der Lianar Quadratische Regler (LQR) als auch die Modelbasierte Prädiktive Regelung (MPR) werden in diesem Abschnitt ausgehend vom Zustandsraummodell beschrieben. Nach der Vorstellung des Zustandsraumes wird in Abschnitt 2.6.2 auf die Systemeigenschaft der Flachheit eingegangen. Sie wird beim Regler-Entwurf der Flachheitsbasierte Folgeregelung (FBFR) in Abschnitt 4.3 ausgenutzt. Die in Abschnitt 4.2 vorgestellte Kolonnenregelung wird wie ein LQR ausgelegt. Daher wird der LQR in Abschnitt 2.6.3 vorgestellt. Schließlich wird in Abschnitt 2.6.4 der Aufbau einer MPR präsentiert. Die in Abschnitt 2.9 vorgestellte Pfadregelung ist als MPR realisiert.

2.6.1 Zustandsraum

Die Modellierung eines zu steuernden bzw. regelnden Systems stellt einen wesentlichen Aspekt der Regelungstechnik dar. Daher wird an dieser Stelle kurz auf den Zustandsraum als eine mögliche Modellform eingegangen. Dieser Abschnitt basiert auf [4].

Üblicherweise lassen sich (physikalische) Systeme durch Differenzialgleichungen (DGL) darstellen. Diese wiederum lassen sich auf Differenzialgleichungssysteme mit n Gleichungen erster Ordnung zurückführen und können gemäß

$$\begin{aligned}\dot{x}_1 &= f_1(x_1, \dots, x_n, u_1, \dots, u_p, t) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, \dots, x_n, u_1, \dots, u_p, t)\end{aligned}\tag{2.15}$$

mit den Zuständen x_i angegeben werden. Dabei bezeichnen x_i die n Zustandsgrößen und u_i die p Eingangsgrößen des Systems. Aus diesen Größen können durch die weiteren q Gleichungen

$$\begin{aligned}y_1 &= h_1(x_1, \dots, x_n, u_1, \dots, u_p, t) \\ &\vdots \\ y_q &= h_q(x_1, \dots, x_n, u_1, \dots, u_p, t),\end{aligned}\tag{2.16}$$

q Ausgänge y_i des Systems definiert werden. Üblicherweise werden die Größen der Gleichungen (2.15) und (2.16) in Vektorschreibweise angegeben. Es ergibt sich

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}, \mathbf{u}, t), & \mathbf{x} &\in \mathbb{R}^n, & \mathbf{u} &\in \mathbb{R}^p \\ \mathbf{y} &= \mathbf{h}(\mathbf{x}, \mathbf{u}, t), & \mathbf{y} &\in \mathbb{R}^q\end{aligned}\tag{2.17}$$

mit dem Zustandsvektor \mathbf{x} , dem Eingangsvektor \mathbf{u} und dem Ausgangsvektor \mathbf{y} . Im Fall von linearen, zeitinvarianten Systemen lässt sich das Zustandsraummodell als

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A} \cdot \mathbf{x}(t) + \mathbf{B} \cdot \mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C} \cdot \mathbf{x}(t) + \mathbf{D} \cdot \mathbf{u}(t)\end{aligned}\tag{2.18}$$

mit der Systemmatrix \mathbf{A} , der Eingangsmatrix \mathbf{B} , der Ausgangsmatrix \mathbf{C} und der Durchgriffsmatrix \mathbf{D} angeben. Hierbei bezeichnen $\mathbf{x}(t)$, $\mathbf{y}(t)$ und $\mathbf{u}(t)$ die zeitlichen Verläufe von Zustands-, Ausgangs- bzw. Stellsignal der Regelstrecke. Technische Systeme sind für gewöhnlich nicht sprungfähig. Es gibt folglich keinen direkten Durchgriff vom Eingang \mathbf{u} auf den Ausgang \mathbf{y} , sodass $\mathbf{D} \equiv \mathbf{0}$ gilt. Damit folgt das Zustandsraum Modell

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A} \cdot \mathbf{x}(t) + \mathbf{B} \cdot \mathbf{u}(t), & \mathbf{x}(0) &= \mathbf{x}_0 \\ \mathbf{y}(t) &= \mathbf{C} \cdot \mathbf{x}(t)\end{aligned}\tag{2.19}$$

mit dem Anfangszustand \mathbf{x}_0 .

Kommen rechnergestützte Regelungsverfahren zum Einsatz, ist eine kontinuierliche Betrachtung des Systems meistens nicht mehr möglich. Es wird Zeit benötigt, um die aktuellen Zustände zu messen bzw. zu schätzen und dann mit den bestimmten Werten weiter zu rechnen. Um dieser benötigten Zeit Rechnung zu tragen, wird eine zeitdiskrete Betrachtung der Systeme durchgeführt. Das bedeutet, dass zu fest definierten Abtastzeitpunkten die Systemzustände ermittelt werden. Danach hat die Regelung Zeit, diese bis zum nächsten Abtastzeitpunkt zu verarbeiten und beispielsweise ein Steuerungssignal daraus zu ermitteln und an das System zu kommandieren. Werden die Abtastzeitpunkte äquidistant verteilt, bildet die Abtastzeit T_{ab} , die zwischen zwei aufeinanderfolgenden Abtastungen vergeht, einen charakteristischen Wert der zeitdiskreten Betrachtung. Für die Betrachtung zeitdiskreter Systeme wird die ganzzahlige Zeitvariable $k \in \mathbb{Z}$ eingeführt. Sie gibt die einzelnen betrachteten Abtastzeitpunkte wieder und steht mit der Abtastzeit T_{ab} und der kontinuierlichen Zeitvariable t in der folgenden Beziehung:

$$t = k \cdot T_{ab}.\tag{2.20}$$

Das kontinuierliche Modell (2.19) kann entsprechend der in [4] hergeleiteten Umrechnungsvorschriften

$$\begin{aligned}\mathbf{A}_D &= e^{\mathbf{A} \cdot T_{ab}} \\ \mathbf{B}_D &= \left(e^{\mathbf{A} \cdot T_{ab}} - \mathbf{I} \right) \mathbf{A}^{-1} \cdot \mathbf{B} \\ \mathbf{C}_D &= \mathbf{C}\end{aligned}\tag{2.21}$$

als zeitdiskretes Zustandsraummodell

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}_D \cdot \mathbf{x}(k) + \mathbf{B}_D \cdot \mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}_D \cdot \mathbf{x}(k)\end{aligned}\tag{2.22}$$

angegeben werden. Dabei sind $\mathbf{x}(k)$, $\mathbf{u}(k)$ und $\mathbf{y}(k)$ der zeitdiskrete Zustands-, Eingangs- bzw. Ausgangsvektor zum Zeitpunkt k .

2.6.2 Flachheit

Flachheit ist eine Systemeigenschaft, die insbesondere für Trajektorienfolge nützlich ist. Sowohl lineare als auch nichtlineare Systeme können die Flachheitseigenschaft besitzen.

Dementsprechend werden allgemein Systeme betrachtet, die in der Form (2.17) vorliegen. Anders als in Abschnitt 2.6 wird zur Erhöhung der Übersichtlichkeit im Folgenden die Zeitabhängigkeit der Größen nicht mehr explizit mit angegeben. Erfüllt ein System die in [4] vorgestellte Definition, wird es als flaches System bezeichnet. Sie lautet:

„Das System (2.17) heißt genau dann flach, wenn ein (fiktiver) Ausgang \mathbf{y} existiert, so dass mit differentiell unabhängigen Komponenten von \mathbf{y} und $\dim \mathbf{y} = \dim \mathbf{u}$ gilt:

$$\mathbf{y} = \Psi_0(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \dots, \mathbf{u}^{(\alpha)}) \quad (2.23)$$

$$\mathbf{x} = \Psi_1(\mathbf{y}, \dot{\mathbf{y}}, \dots, \mathbf{y}^{(\beta)}) \quad (2.24)$$

$$\mathbf{u} = \Psi_2(\mathbf{y}, \dot{\mathbf{y}}, \dots, \mathbf{y}^{(\beta+1)}) . \text{“} \quad (2.25)$$

Die Anzahlen α und β der Ableitungen in der Definition sind als vektorwertig notiert, was darstellen soll, dass für jede Komponente y_i von \mathbf{y} eine andere Anzahl Ableitungen angegeben werden kann. Die Definition fordert also einerseits, dass der Ausgang \mathbf{y} aus den Systemzuständen \mathbf{x} sowie dem Systemeingang \mathbf{u} und einer endlichen Anzahl zeitlicher Ableitungen von \mathbf{u} angegeben werden kann. Andererseits sollen sich der Zustand \mathbf{x} und der Eingang \mathbf{u} als Funktionen des Ausgangs \mathbf{y} und endlich vielen Zeitableitungen des Ausgangs \mathbf{y} darstellen lassen. Kann dies für ein System nachgewiesen werden, ist es steuerbar [4]. Ein System heißt vollständig steuerbar, wenn es in endlicher Zeit von jedem beliebigen Anfangszustand durch eine geeignet gewählte Eingangsgröße in einen beliebig vorgegebenen Endzustand überführt werden kann [29]. Die Steuerbarkeit folgt aus einem in [4] vorgestellten Satz. Er lautet:

„Jedes flache System ist steuerbar bzw. die Flachheit ist hinreichend für die Steuerbarkeit nichtlinearer Systeme.“

Für den Beweis des Satzes sei an dieser Stelle auf [4] verwiesen. Detailliertere Betrachtungen der Flachheit sind in [21–23, 42] und [43] zu finden.

Aus (2.25) folgt, dass der Eingang des Systems in Abhängigkeit seines Ausgangs und dessen Ableitungen angegeben werden kann. Soll ein System so gesteuert werden, dass einer bestimmten Ausgangstrajektorie gefolgt wird, muss diese Trajektorie die folgende Bedingung erfüllen. Die Trajektorie muss so oft stetig differenzierbar sein, dass alle zur Darstellung des Eingangs und der Zustände nötigen Ableitungen gebildet werden können und existieren. Ist dies erfüllt, kann das System durch Vorgabe der Ausgangstrajektorie und Bestimmung der benötigten Ableitungen so gesteuert werden, dass es der Ausgangstrajektorie exakt folgt.

2.6.3 Linear Quadratischer Regler

Unter anderem wird der Linear Quadratische Regler (LQR) in [4, 9, 29] eingeführt. Basierend auf diesen Arbeiten soll das Entwurfskonzept hier kurz vorgestellt werden. Sein Name ergibt sich, da ein lineares System mittels einer quadratischen Gleichung optimiert

wird. Liegt ein Zustandsraummodell in Form von (2.19) vor, kann ein quadratisches Gütefunktional J definiert werden als

$$J = \int_0^{\infty} \left(\mathbf{x}^T(t) \cdot \mathbf{Q} \cdot \mathbf{x}(t) + \mathbf{u}^T(t) \cdot \mathbf{R} \cdot \mathbf{u}(t) \right) dt \quad (2.26)$$

[4, 29]. Dabei sind \mathbf{Q} und \mathbf{R} zugehörige Wichtungsmatrizen. Das Eingangssignal $\mathbf{u}_{opt}(t)$, das das Gütefunktional J minimiert, stellt die optimale Lösung im Sinne des LQR dar.

Das Gütefunktional J setzt sich additiv aus zwei Termen zusammen. Im ersten Term wird der Einfluss der Zustände berücksichtigt. Ihm ist die Zustandswichtungsmatrix \mathbf{Q} zugeordnet. Eine hohe Wichtung von \mathbf{Q} führt zu einem agileren geschlossenen Regelkreis. Wenn die Zustände des Systems in die Umgebung der Ruhelage $\mathbf{x} = \mathbf{0}$ gesteuert werden, wird dieser Term besonders klein. Der zweite Term spiegelt den Stellaufwand wider und ihm ist die Eingangswichtungsmatrix \mathbf{R} zugeordnet. Wird \mathbf{R} hoch gewichtet, wird der Stellaufwand kleiner gehalten, was das System träger macht. Durch die Wahl von \mathbf{Q} und \mathbf{R} wird festgelegt, welches Systemverhalten der geschlossene Regelkreis abbildet. [4, 29]

Das Minimum des Gütefunktional J wird als J_{opt} bezeichnet. Es kann gemäß

$$J_{opt} = \int_0^{\infty} \left(\mathbf{x}^T(t) \cdot \mathbf{Q} \cdot \mathbf{x}(t) + \mathbf{u}_{opt}^T(t) \cdot \mathbf{R} \cdot \mathbf{u}_{opt}(t) \right) dt = \mathbf{x}_0^T \cdot \mathbf{P}_{opt} \cdot \mathbf{x}_0 \quad (2.27)$$

berechnet werden [29]. Dabei bezeichnet \mathbf{P} die positiv definite Lösung der stationären Matrix-Riccati-Gleichung [9]

$$\mathbf{P} \cdot \mathbf{A} + \mathbf{A}^T \cdot \mathbf{P} - \mathbf{P} \cdot \mathbf{B} \cdot \mathbf{R}^{-1} \cdot \mathbf{B}^T \cdot \mathbf{P} + \mathbf{Q} = \mathbf{0}. \quad (2.28)$$

Zusätzlich gilt mit der Matrix \mathbf{P}_{opt} für das im Sinne der Kostenfunktion (2.26) optimale Stellgesetz

$$\mathbf{u}_{opt} = -\mathbf{R}^{-1} \cdot \mathbf{B}^T \cdot \mathbf{P}_{opt} \cdot \mathbf{x} = -\mathbf{K} \cdot \mathbf{x}. \quad (2.29)$$

Der optimale Stelleingang \mathbf{u}_{opt} kann also aus den Zuständen \mathbf{x} berechnet werden [4]. Durch eine Zustandsrückführung mit der Matrix $\mathbf{K} = \mathbf{R}^{-1} \cdot \mathbf{B}^T \cdot \mathbf{P}_{opt}$ kann der LQR realisiert werden. Die Matrix \mathbf{P}_{opt} kann numerisch aus (2.28) bestimmt werden. In die Gleichungen (2.28) und (2.29) gehen nur die Systemmatrizen aus (2.19) sowie die Wichtungsmatrizen \mathbf{Q} und \mathbf{R} ein, die als zeitinvariant angenommen werden. Die Zustandsrückführungsmatrix \mathbf{K} kann also bereits während der Reglersynthese bestimmt werden und ändert sich dann nicht mehr. Der LQR ist also eine Zustandsrückführung. [4, 9, 29]

2.6.4 Modellbasierte Prädiktive Regelung

Das Regelungsverfahren der Modellbasierten Prädiktiven Regelung (MPR) wird in diesem Abschnitt basierend auf [4] vorgestellt. Um die Formeln zu vereinfachen, wird die Notation aus [4] eingeführt. Mit $a(k+i|k)$ für $i \in \mathbb{N}_0$ sei der zum Zeitpunkt k für den

Zeitschritt $k + i$ prognostizierte Wert von a beschrieben. Ferner sei $a(\cdot|k)$ die prädizierte Folge

$$a(\cdot|k) = (a(k|k), a(k+1|k), a(k+2|k), \dots, a(k+N_2|k)) \quad (2.30)$$

des Elements a zum Zeitschritt k . Durch $N_2 \in \mathbb{N}$ wird das Ende des Prädiktionshorizonts beschrieben. Dieser wird durch die beiden Werte N_1 und N_2 definiert und erstreckt sich über den Teil der Folge $a(\cdot|k)$ von $a(k+N_1|k)$ hin zu $a(k+N_2|k)$, vgl. Abbildung 2.7. Er gibt an, welche prädizierten Werte in der MPR berücksichtigt werden. Neben dem Prädiktionshorizont gibt es den sogenannten Stellhorizont N_u . Dieser erstreckt sich von $a(k|k)$ bis zu $a(k+N_u-1|k)$. Innerhalb des Stellhorizontes können die Stellgrößen angepasst werden. Für den Zeitraum von $a(k+N_u|k)$ zu $a(k+N_2|k)$ werden die Stellgrößen üblicherweise auf dem letzten Wert $a(k+N_u-1|k)$ konstant gehalten.

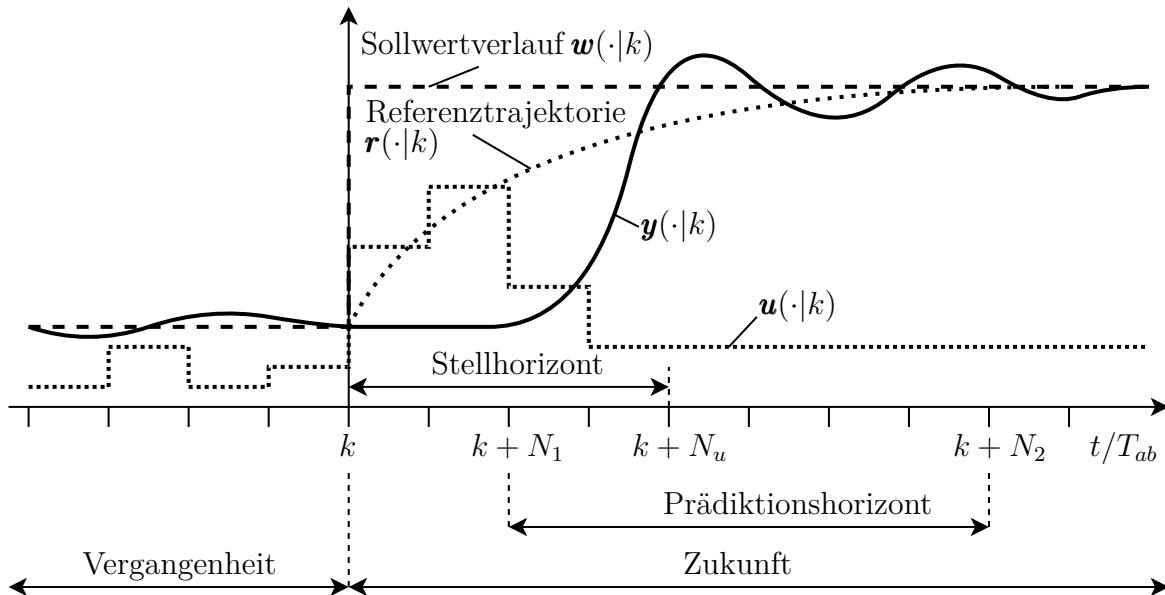


Abb. 2.7: Prinzip der MPR nach [4]

Abbildung 2.7 zeigt eine Momentaufnahme verschiedener Signale einer MPR zum Zeitpunkt k . Alle Verläufe vor diesem Zeitpunkt liegen in der Vergangenheit. Zum Zeitpunkt k ändert sich der Sollwert w . Damit das System der Sollwertänderung besser folgen kann, ist es möglich, eine Referenztrajektorie r zu definieren. Diese gibt an, wie vom bisherigen Sollwert auf den neuen übergegangen werden soll. Dazu wird versucht, eine Eingangsfolge $u(\cdot|k)$ zu finden, für die sich y am dichtesten an r anschmiegt. Der Eingang u darf dabei nur während des Stellhorizonts angepasst werden. Mit einem hinterlegten Systemmodell wird daraus der Ausgang y prädiziert. Die Referenztrajektorie r und der Ausgang y werden anschließend während des Prädiktionshorizonts verglichen. Auf diese Weise kann die Totzeit, die Zeit, die das System benötigt, um auf eine Änderung am Eingang zu reagieren, und in der sich kaum etwas am Ausgang ändert, berücksichtigt werden.

Die MPR basiert darauf, die optimale Eingangsfolge u durch das Lösen eines Optimierungsproblems zu finden. Das Optimierungsproblem besteht darin, eine Kostenfunktion J

unter Beachtung von Nebenbedingungen zu minimieren. Unter Verwendung der gewichteten quadratischen euklidischen Vektornorm $\|\mathbf{a}\|_{\mathbf{A}}^2 = \mathbf{a}^T \mathbf{A} \mathbf{a}$ mit der Wichtungsmatrix \mathbf{A} und mit den Definitionen

$$\Delta \mathbf{u}(i|k) = \mathbf{u}(i|k) - \mathbf{u}(i-1|k) \quad \forall i \in \mathbb{N}_{\geq k} \quad (2.31a)$$

$$\mathbf{u}(k-i|k) = \mathbf{u}(k|k-i) \quad \forall i \in \mathbb{N} \quad (2.31b)$$

kann beispielsweise die Kostenfunktion

$$J = \left\| \begin{bmatrix} \mathbf{y}(k+N_1|k) \\ \vdots \\ \mathbf{y}(k+N_2|k) \end{bmatrix} - \begin{bmatrix} \mathbf{r}(k+N_1|k) \\ \vdots \\ \mathbf{r}(k+N_2|k) \end{bmatrix} \right\|_{\mathbf{Q}}^2 + \left\| \begin{bmatrix} \Delta \mathbf{u}(k|k) \\ \vdots \\ \Delta \mathbf{u}(k+N_u|k) \end{bmatrix} \right\|_{\mathbf{R}}^2, \quad (2.32)$$

mit Wichtungsmatrizen \mathbf{Q} und \mathbf{R} angegeben werden. Bei \mathbf{Q} und \mathbf{R} handelt es sich um positiv definite bzw. positiv semidefinite Diagonalmatrizen, die nur positive Elemente bzw. positive Elemente und Nullen auf der Hauptdiagonalen besitzen. Mit der Wichtungsmatrix \mathbf{Q} wird die Regelabweichung $\mathbf{e} = \mathbf{y} - \mathbf{r}$ gewichtet. Die Wichtungsmatrix \mathbf{R} dagegen gewichtet die nötige Stellsignaländerung. In Gleichung (2.32) gibt $\mathbf{r}(\cdot|k)$ einen Referenzvektorverlauf an. Dieser kann aus dem Sollwertverlauf $\mathbf{w}(\cdot|k)$ bestimmt werden. Mittels des Referenzvektors \mathbf{r} kann die Dynamik vorgegeben werden, mit der das System vom Istwert zum Sollwert überführt werden soll. Soll keine besondere Dynamik vorgegeben werden, sondern dem Sollwert \mathbf{w} direkt gefolgt werden, so kann $\mathbf{r} = \mathbf{w}$ gesetzt werden.

Die zu berücksichtigenden Nebenbedingungen können sich zum Beispiel aus Eingangs- oder anderen Systembeschränkungen ergeben. Ein Beispiel für eine Nebenbedingung ist der mechanisch begrenzte Lenkwinkel der IRT-Buggys. Die Lenkung kann nur bis zu einem bestimmten Winkel eingeschlagen werden. Dürfen die Eingänge, wie in dem Beispiel, definierte Werte nicht unter- bzw. überschreiten, lässt sich dies durch eine Nebenbedingung der Form

$$\mathbf{u}_{min} \leq \mathbf{u}(k) \leq \mathbf{u}_{max} \quad \forall k \quad (2.33)$$

berücksichtigen.

Mit (2.32) lässt sich nun das Optimierungsproblem aufstellen. Gesucht wird diejenige optimale Steuersequenz $\mathbf{u}_{opt}(\cdot|k)$, die J minimiert

$$\mathbf{u}_{opt}(\cdot|k) = \arg \min_{\mathbf{u}(\cdot|k)} J(\mathbf{u}(\cdot|k))$$

mit den Nebenbedingungen

$$\begin{aligned} \mathbf{u}(k+j|k) &= \mathbf{u}(k+j-1|k) + \Delta \mathbf{u}(k+j|k) \\ \mathbf{x}(k+j+1|k) &= \mathbf{f}(\mathbf{x}(k+j|k), \mathbf{u}(k+j|k)) \\ \mathbf{y}(k+j|k) &= \mathbf{h}(\mathbf{x}(k+j|k), \mathbf{u}(k+j|k)) \\ \mathbf{0} &\geq \mathbf{g}_{un}(\mathbf{x}(k+j+1|k), \mathbf{x}(k+j|k), \mathbf{u}(k+j|k), \Delta \mathbf{u}(k+j|k)) \\ \mathbf{0} &= \mathbf{g}_{gl}(\mathbf{x}(k+j+1|k), \mathbf{x}(k+j|k), \mathbf{u}(k+j|k), \Delta \mathbf{u}(k+j|k)). \end{aligned} \quad (2.34)$$

Die Funktionen \mathbf{f} und \mathbf{h} stellen das verwendete zeitdiskrete Prozessmodell dar, vgl. (2.17) (dort kontinuierlich), während mit \mathbf{g}_{un} und \mathbf{g}_{gl} harte Ungleichungs- bzw. Gleichungsnebenbedingungen berücksichtigt werden können. Die Nebenbedingung aus (2.33) würde sich also in \mathbf{g}_{un} wieder finden. Üblicherweise wird das Prinzip des zurückweichenden Horizontes angewendet. Dieses besagt, dass nicht die gesamte optimale Folge $\mathbf{u}_{opt}(\cdot|k)$ als Steuereingang in den nächsten Zeitschritten verwendet wird, sondern nur der erste Wert der Folge $\mathbf{u}_{opt}(k|k)$. Für den nächsten Zeitschritt wird die gesamte Berechnung erneut durchgeführt, um so den Wert für das nächste Eingangssignal \mathbf{u} zu bestimmen. Durch dieses Vorgehen ist es möglich, nicht gemessene Störgrößen besser zu berücksichtigen [19].

Bei der Auslegung einer MPR stehen als Freiheitsgrade das Modell, gegeben durch die Funktionen \mathbf{f} und \mathbf{h} , eine Rechenvorschrift zur Bestimmung der Referenztrajektorie $\mathbf{r}(\cdot|k)$, der Stellhorizont N_u , der Prädiktionshorizont, bestimmt durch N_1 und N_2 , die Wichtungsmatrizen \mathbf{Q} und \mathbf{R} sowie die Abtastzeit T_{ab} zur Parametrierung der Regelung zur Verfügung. Wie auch beim LQR kann durch die Gewichtung von \mathbf{Q} und \mathbf{R} die Geschwindigkeit und Robustheit der MPR beeinflusst werden. N_1 sollte in der Größe der Systemtotzeit gewählt werden, da die Änderung des Eingangssignals $\mathbf{u}(k)$ erst nach der Totzeit am Ausgang wirkt. Mit der Wahl von N_u sollte es möglich sein, der Dynamik, die durch die Referenz $\mathbf{r}(\cdot|k)$ vorgegeben ist, zu folgen. Um das Folgeverhalten zu bewerten, muss N_2 entsprechend gewählt werden. Die Abtastzeit T_{ab} sollte so gewählt werden, dass der Dynamik des Systems gefolgt werden kann, es aber zu keiner Unterabtastung kommt. Zusätzlich zu den gerade beschriebenen Parametern sind andere Definitionen der Kostenfunktion J denkbar, um weitere Faktoren zu berücksichtigen. Für ein System bei dem das Modell (2.22) ohne Berücksichtigung von Störgrößen verwendet werden soll, lautet eine mögliche und übliche Definition von \mathbf{f} und \mathbf{h} wie folgt:

$$\mathbf{f}(\mathbf{x}(k+j|k), \mathbf{u}(k+j|k)) = \mathbf{A}_D \cdot \mathbf{x}(k+j|k) + \mathbf{B}_D \cdot \mathbf{u}(k+j|k) \quad (2.35)$$

$$\mathbf{h}(\mathbf{x}(k+j|k), \mathbf{u}(k+j|k)) = \mathbf{C}_D \cdot \mathbf{x}(k+j|k). \quad (2.36)$$

Einige vorteilhafte Eigenschaften der MPR sind laut [4]:

- einfache Behandlung von gekoppelten Mehrgrößensystemen mit inhärenter Entkopplung,
- explizite Berücksichtigung von Begrenzungen der Stell-, Zustands- und Ausgangsgrößen,
- inhärente Kompensation von Totzeiten,
- inhärente Störgrößenaufschaltung durch Berücksichtigung mess- oder schätzbarer Störgrößen im Prozessmodell sowie
- Berücksichtigung zukünftiger Sollwerttrajektorien.

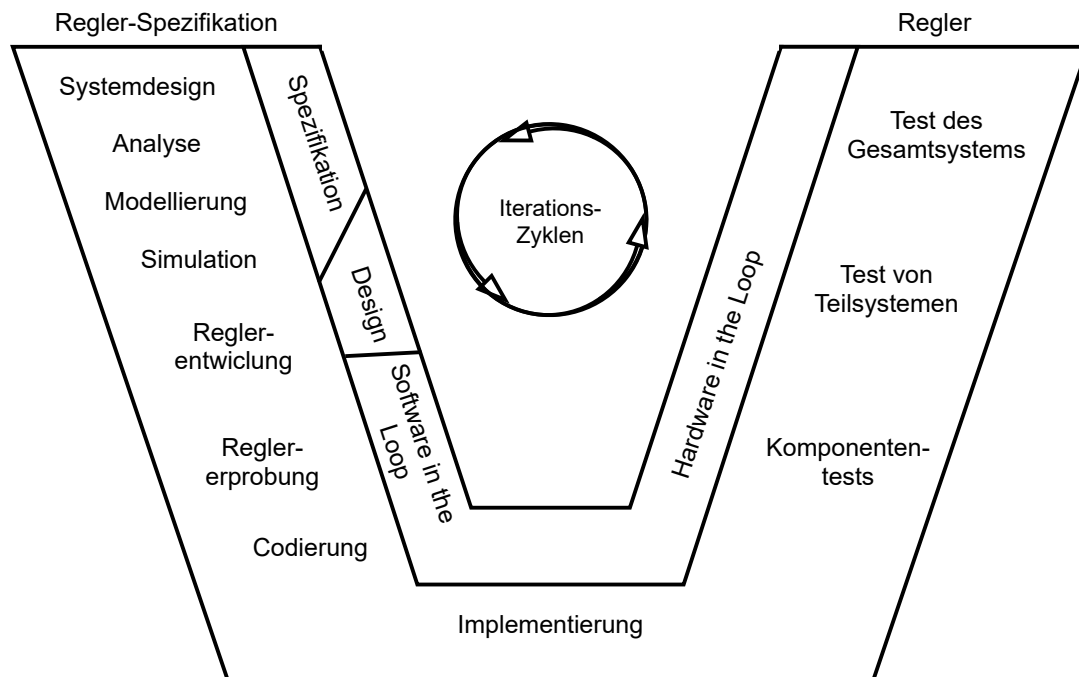


Abb. 2.8: V-Modell nach [6]

2.7 Rapid Control Prototyping

Die im Rahmen dieser Arbeit durchgeführten Entwicklungen von Regelungen bzw. Steuerungen folgen dem Konzept des Rapid Control Prototyping (RCP). Diesem liegt das in Abbildung 2.8 gezeigte V-Modell zugrunde. Üblicherweise wird dem V von links oben nach unten und dann wieder über den rechten Zweig nach oben gefolgt. Der Detaillierungsgrad nimmt auf dem Weg nach unten immer weiter zu. Auf dem Weg nach oben nimmt er wieder ab. Nach der vom Verein Deutscher Ingenieure (VDI) aufgestellten Richtlinie VDI 2206 wird der linke Ast als Systementwurf und der rechte Ast als Systemintegration bezeichnet. Entwicklungen werden iterativ durchgeführt, es kann also vom rechten Ast zurück auf den linken Ast gesprungen werden, wenn die gewünschten Eigenschaften nicht erreicht sind. In diesem Fall findet eine waagerechte Bewegung auf den linken Zweig in gleicher Höhe statt. Vom erreichten Punkt wird anschließend wieder dem V in die Tiefe und dann erneut auf den rechten Zweig nach oben gefolgt. Das Zurückspringen kann mehrfach und in unterschiedlichen Detaillierungsgraden erfolgen.

Dabei ist es wichtig, möglichst früh die Modelle zu erproben. Die hierfür verwendete RCP-Struktur sieht es vor, dass es Prozess und Regler sowohl in der Simulation gibt als auch in der Realität. Idealerweise werden dabei die gleichen Modelle verwendet, so dass die Simulations- und Testergebnisse direkt umgesetzt werden können und auch schon im jeweils anderen Bereich verfügbar sind. Um möglichst schnell voranzukommen, gibt es Mischformen, bei denen eine Komponente nur in Form einer Simulation vorhanden ist, andere Komponenten aber in der Realität existieren. Zwei Vertreter sind die im V-Modell in Abbildung 2.8 aufgeführten Software in the Loop (SiL)- und Hardware in the Loop

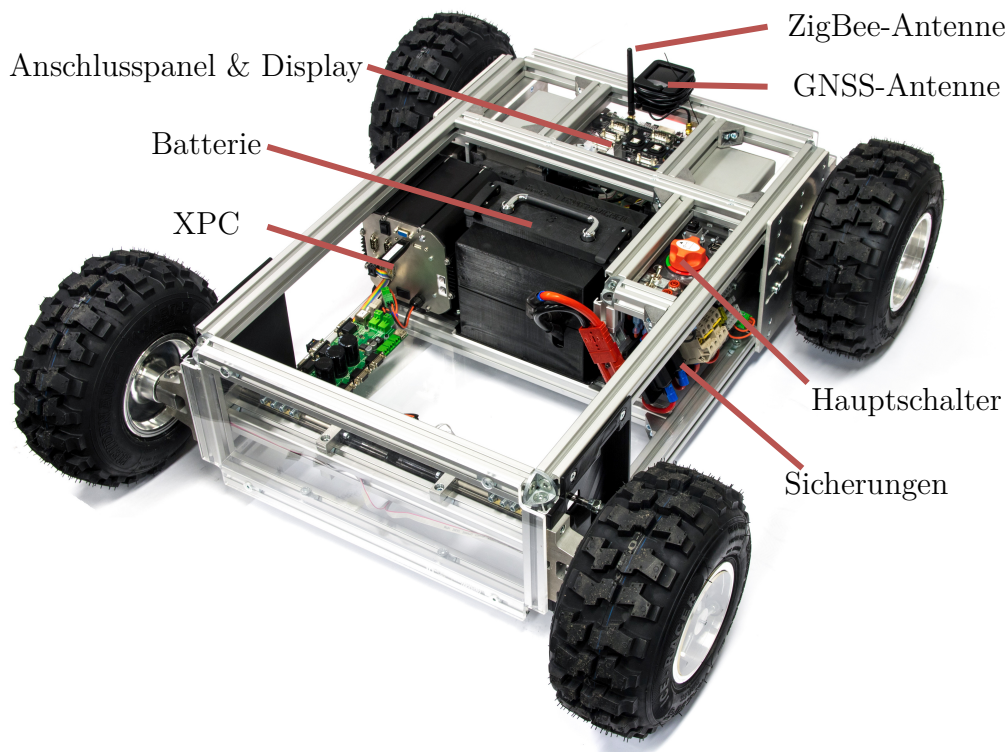


Abb. 2.9: IRT-Buggy, am IRT entwickeltes Testfahrzeug für Forschung und Lehre

(HiL)-Simulationen. Dabei gilt nach [6]:

HiL Von HiL wird gesprochen, wenn der entwickelte Regelungs- und Steuerungsalgorithmus auf einem Entwicklungsrechner ausgeführt wird und mit dem realen Prozess oder Teilen davon verbunden ist.

SiL Von SiL ist die Rede, wenn der auf der Zielhardware implementierte Regelungs- und Steuerungsalgorithmus mithilfe eines Simulationsmodells des realen Prozesses erprobt wird, das auf einem Entwicklungsrechner läuft.

2.8 Der IRT-Buggy

Abbildung 2.9 zeigt den sogenannten IRT-Buggy, im Folgenden einfach Buggy genannt. Er wird als Entwicklungsplattform in Forschung und Lehre eingesetzt und im Folgenden basierend auf [38, 40, 41, 50] und [56] vorgestellt. Sein Rahmen ist aus einem Aluminium-Profil gefertigt, sodass weitere Versuchsaufbauten schnell und effizient am Versuchsträger angebracht werden können. In Tabelle 2.2 sind die wesentlichen Kennwerte der Buggys dargestellt.

Die Hinterräder werden von zwei unabhängigen Permanentmagnet-Gleichstrom-Motoren angetrieben. Auf diese Weise ist es möglich, über den Antrieb durch entsprechendes Ansteuern der Motoren ein Giermoment auf das Fahrzeug aufzuprägen. Jeder Antriebsmotor

Kennwert	Wert	Einheit
Gesamtgewicht	62	kg
Versorgungsspannung	24	V
Spitzenleistung Motoren (2×)	1,5	kW
Fahrzeuglänge	98	cm
Fahrzeugbreite (vorne)	88	cm
Fahrzeugbreite (hinten)	81	cm
Radstand L	69	cm
Raddurchmesser	14,2	cm
Wenderadius (Fahrzeugmitte) etwa	1,3	m
Getriebeübersetzungsverhältnis	4,8	–

Tabelle 2.2: Kennwerte der IRT-Buggys [1, 7, 38]

verfügt über einen eigenen Drehzahlregler. Damit diese sich im normalen Betrieb nicht gegenseitig beeinflussen, wird zusätzlich ein virtuelles Differenzial umgesetzt. Als normaler Betrieb wird hier eine reine Nutzung als ferngesteuerter Versuchsträger verstanden. Für die Umsetzung des Differenzials und den ferngesteuerten Betriebsmodus wird die sogenannte Basisregelung der Buggys verwendet. Durch die Basisregelung werden gewisse Basisfunktionalitäten bereitgestellt. Ein Beispiel ist die bereits erwähnte Nutzung der Buggys als rein ferngesteuertes Fahrzeug. Dabei kann über einen Joystick die gewünschte Geschwindigkeit und über einen anderen der Lenkwinkel vorgegeben werden. Auf die anderen Funktionalitäten wird im weiteren Verlauf eingegangen.

Die Basisregelung ist in MATLAB[®] bzw. SIMULINK^{®2} modelliert. Aus dem Modell kann der Programmcode mittels Codegenerierung erzeugt werden. Die Basisregelung läuft auf einem Texas Instruments C2000[™] Mikrocontroller. Der Mikrocontroller befindet sich auf einem EZDSP F28335 Entwicklungsboard von Spectrum Digital. Zusammen mit einer Hilfsplatine bildet es das sogenannte Basisboard. Neben der Basisregelung stellt es Schnittstellen zwischen Sensoren, Aktoren und höheren Regelungssystemen zur Verfügung. Ein solcher Sensor ist eine ADIS16405 BMLZ IMU von Analog Devices, die auf der Hilfsplatine verbaut ist. Mit der IMU können unter anderem Beschleunigungen und Drehraten in allen drei Raumrichtungen gemessen werden.

Aufgrund der voneinander unabhängig angetriebenen Hinterräder bzw. der getrennten Lenkung müssen für beide Seiten grundsätzlich unterschiedliche Sollwertvorgaben an die Aktoren kommandiert werden. Nur so wird eine Kurvenfahrt ermöglicht. Die Vorgaben an den Buggy können sich auf ein Einspur-Fahrzeugmodell oder individuell auf jedes Rad einzeln beziehen. Beim Einspur-Fahrzeugmodell wird die vereinfachende Annahme getroffen, dass der Buggy nur eine Spur besitzt, also nur ein Vorder- und ein Hinterrad. Diese imaginären Räder werden in der Fahrzeugmitte, auf Höhe der Vorder- bzw. Hinterachse angenommen, sodass der Radstand des Modells dem der Realität entspricht. Das

²MATLAB[®] und SIMULINK[®] sind registrierte Marken der The MathWorks, Inc.
(<https://www.mathworks.com>).

Basisboard bestimmt aus den Sollwerten des Einspur-Fahrzeugmodells die individuellen Sollwerte für jedes Rad.

Dem Basisboard können über den CAN-Bus Sollwerte für die folgenden Fahrzeugparameter vorgegeben werden:

- Lenkwinkel δ (Einspurmodell),
- Drehzahl n (Einspurmodell),
- Lenkposition LP_L und LP_R ,
- Lenkgeschwindigkeit v_{LenkL} und v_{LenkR} ,
- Drehzahl n_L und n_R ,
- Strom i_L und i_R .

Hierbei bedeuten die Indizes L bzw. R, dass der entsprechende Sollwert explizit für die linke bzw. rechte Fahrzeugseite vorgegeben werden kann. Für das Einspurmodell müssen lediglich ein Lenkwinkel δ für das imaginäre Vorderrad und/oder eine Drehzahl n für das imaginäre Hinterrad bezogen auf die Fahrzeugmitte vorgegeben werden. Intern werden diese Werte nach Ackermann auf die einzelnen Räder umgerechnet bzw. durch a priori bestimmte Lookup-Tabellen bestimmt. So wird eine Ackermann-Lenkung (siehe z. B. [32]) künstlich nachgebildet. In die Berechnungen gehen Fahrzeugdaten wie der Radstand L und die Spurweiten des Buggys ein. Es ergeben sich LP_L und LP_R aus δ , sowie n_L und n_R aus δ als auch n [32]. Diese Umwandlungen stehen nicht nur für die Fernsteuerung des Buggys zur Verfügung, sondern auch höheren externen Regelungen. Eine höhere Steuerung bzw. Regelung kann also entweder jedes Rad einzeln kommandieren oder Vorgaben für das in der Basisregelung hinterlegte Einspurmodell machen.

Überdies ist es dem Nutzer freigestellt, ob er einen in der Basisregelung implementierten Geschwindigkeitsregler verwendet oder ob die Regelung selbst implementiert wird. Die hierzu nötigen Messwerte werden vom Basisboard bzw. anderen Hilfsplatinen auf dem CAN-Bus bereitgestellt. Es sind die folgenden Messwerte vorhanden:

- Raddrehzahlen und -umdrehungen hinten,
- GST / GGQ (Coordinate Quality Latitude/Longitude oder 3D),
- Raddrehzahlen und -umdrehungen vorne,
- RMC (Latitude, Longitude, Velocity, Angle, Time, Quality),
- Batteriespannung,
- Puls pro Sekunde (PPS),
- Tastgrad der Antriebsmotoren,
- Fahrzeugbeschleunigung (dreidimensional),
- Strom der Antriebsmotoren,
- Fahrzeugdrehrate (dreidimensional),
- GGA (Altitude, numSat, Quality),
- Magnetfeld (dreidimensional),
- GSA (H/V/P-DOP),
- Temperatur.

Die GNSS-Daten können von zwei unterschiedlichen Quellen stammen. Einerseits ist ein GNSS-Empfängermodul der Baureihe LEA-M8T der u-blox AG fest auf jedem Buggyver-

baut, der seine Messungen mit 5 Hz in Form von NMEA-Datensätzen an das Basisboard überträgt. Dieses wiederum extrahiert die wichtigsten Daten der einzelnen Datensätze und stellt diese über den CAN-Bus zur Verfügung. Andererseits verfügt der Buggy respektive das Basisboard über eine weitere RS-232-Schnittstelle, die für den Anschluss eines externen Empfängers vorgesehen ist. Auch über diese Schnittstelle empfangene NMEA-Datensätzen stellt das Basisboard auf dem CAN-Bus bereit. Es sind also beide Messungen gleichzeitig verfügbar, wobei die Datensätze einer der Empfänger intern als Referenz bezeichnet werden. Die Wahl der Referenz kann im laufenden Betrieb umgeschaltet werden. Daher werden üblicherweise die nicht als Referenz gekennzeichneten Messwerte in den höheren Reglern verwendet, da später entschieden werden kann, von welchem Empfänger diese nun stammen. Das PPS-Signal hingegen kommt immer vom u-blox-Empfänger, da es über eine separate Leitung digital an das Basisboard übertragen wird.

Neben den Messwerten gibt es auch Parameter der Buggys, die über den CAN-Bus auslesbar sind. Auf einem Display können die Parameter und Messwerte zusätzlich angezeigt werden. Die Parameter werden bei der Programmierung des Basisboards initialisiert und sind im laufenden Betrieb anpassbar. Hierzu wird die im Display integrierte Tastatur verwendet. Einer der Parameter bestimmt beispielsweise, welcher GNSS-Empfänger als Referenz betrachtet wird. Neben fahrzeugspezifischen Parametern, wie beispielsweise dem Radumfang, den Drehzahl- und Stromhöchstgrenzen und anderen Grenzwerten, gibt es auch Parameter, die für angeschlossene Geräte gedacht sind. Das bedeutet sie werden nicht vom Basisboard ausgewertet. Vielmehr wird so eine Schnittstelle zu angeschlossenen Geräten bereitgestellt, wenn diese über eine entsprechende Implementierung verfügen. Üblicherweise handelt es sich bei einem angeschlossenen Gerät um ein Rechner-System, das im Betrieb nur eingeschränkt zugänglich ist und meist ohne Monitor oder Tastatur betrieben wird. Die frei definierbaren Parameter erlauben es so beispielsweise, je nach Implementierung den Betriebsmodus über die Tasten am Buggy zu ändern.

In dieser Arbeit werden die untersuchten Fahrzeugführungssysteme auf einem Industrie-PC in der PC-104 Bauform implementiert. Die Bauform wird verwendet, da sie die Bauform einer von SIMULINK[®] unterstützten CAN-Schnittstelle ist. Der Industrie-PC in Verbindung mit der CAN-Schnittstelle kann also ohne weiteres als SIMULINK[®] Real-Time[™] Target³ verwendet werden. Das bedeutet, es ist möglich, direkt aus dem SIMULINK[®]-Modell Code generieren zu lassen. Dieser Code liefert sein eigenes Betriebssystem und kann so direkt auf der Hardware ausgeführt werden. In Anlehnung an frühere MATLAB[®] Releases, in denen das Target noch XPC Target hieß, wird der Industrie-PC im weiteren Verlauf der Arbeit als XPC bezeichnet. Der XPC verfügt über eine Taktrate von 2,4 GHz und 2 GB Arbeitsspeicher. Neben der CAN-Schnittstelle, die zwei getrennte CAN-Busse hat, besitzt der XPC zwei serielle RS-232-Schnittstellen, einen Gbit LAN-Port und zwei USB 2.0-Anschlüsse.

Das Basisboard besitzt ein WLAN-Modul, über das die selben Vorgaben wie über die Fernsteuerung möglich sind. Dies ermöglicht das Steuern des Buggys beispielsweise aus

³Beginnend mit MATLAB[®] Version 2017A ist die Unterstützung des Targets eingeschränkt worden.

einer App eines Smartphones. Außerdem ist im Gehäuse der XPC ein WLAN-Router verbaut, der über ein Ethernet-Hub mit dem XPC verbunden ist. Das Hub besitzt zwei weitere nach außen geführte Ethernet-Ports, an die weitere Netzwerkkomponenten angeschlossen werden können. Durch den WLAN-Router kann der XPC kabellos via WLAN programmiert werden. Der Router kann theoretisch auch verwendet werden, um die XPCs mehrerer Buggys und somit auch die Buggys untereinander zu verbinden. Praktisch ist diese Verbindung aber eher ungeeignet, da die Reichweite der Router sehr begrenzt ist und es so ständig zu Verbindungsabbrüchen kommt. Dahingegen besitzt jeder Buggy ein ZigBee-Modul, das mit den ZigBee-Modulen der anderen Buggys über ein Mesh-Netzwerk verbunden ist. ZigBee bezeichnet dabei eine auf WPAN aufgesetzte Spezifikation. Die Module sind an den CAN-Bus angeschlossen und übertragen definierte CAN-Nachrichten an die anderen Buggys bzw. duplizieren die Nachrichten der anderen Buggys auf dem eigenen CAN-Bus. Über ein an einen PC angeschlossenes ZigBee-Modul kann dieser die Daten ebenfalls mitlesen und in einer am IRT eigens dafür programmierten Grafischen Nutzer-Schnittstelle (engl. Graphical User Interface) (GUI) darstellen. Da die Datenrate im ZigBee-Protokoll auf 250 kbit/s begrenzt ist, funktioniert dieser Kommunikationsweg für zwei Buggys ausreichend gut. Bei drei und mehr Buggys kommt es aber zu Einschränkungen im praktischen Betrieb. Daher muss für die verlässliche Kommunikation mehrerer Buggys untereinander ein besserer Kommunikationskanal gefunden werden. In Abschnitt 5.1.1 wird die LTE-Verbindung des GO!-Empfängers als ein solcher Kommunikationskanal vorgestellt.

Neben den oben angeführten Funktionen, als Schnittstelle zwischen Sensoren, Aktoren, Display, Fernsteuerung und Regler, realisiert das Basisboard zusätzlich den sogenannten Hardware in the Loop Modus des Buggy (HiL-Modus). Ist ein Buggy in den HiL-Modus versetzt, werden die Messwerte der Sensoren durch simulierte Werte ersetzt. Dazu verfügt der Buggy intern über ein kinematisches Modell von sich selbst. Die Sollwerte von Fernsteuerung und/oder Regler werden in dieses Modell geführt, was daraus die simulierten Sensormesswerte berechnet. Sensorrauschen und andere Einflüsse werden dabei nicht betrachtet. Lediglich einige Zustände werden gespeichert. Die GNSS-Position beispielsweise wird aus dem alten Zustand und den neuen Sollwertvorgaben bestimmt. So ist es möglich, dass die Position vom Einschalten des HiL-Modus an immer weiter angepasst wird. Dies gilt analog für Geschwindigkeit und Orientierung des Buggys. Die Ergebnisse der Berechnungen werden direkt und ohne weitere Datenverarbeitung als Sensormesswerte auf dem CAN-Bus ausgegeben. Anhand der Werte und der Übertragungsart kann ein Regler nicht erkennen, ob sich ein Buggy im HiL-Modus befindet oder nicht. Der Regler arbeitet normal mit den gelieferten Werten.

Der HiL-Modus spielt seine Vorteile in der Vorentwicklung aus. Bevor ein Experiment auf einer Teststrecke durchgeführt wird, kann der Hardwareaufbau bereits vorab im HiL-Modus getestet werden, ohne überhaupt zur Teststrecke fahren zu müssen. Werden die Startkoordinaten des HiL-Modus so gesetzt, dass sie sich im Testgebiet befinden, arbeitet ein an den Buggy angeschlossener Regler so, als befände er sich im Testgebiet. Die Information, ob der HiL-Modus aktiviert ist oder nicht, liegt als Parameter auf dem CAN-Bus vor und kann über das Display angezeigt werden. Über die Tasten am Buggy kann der HiL-Modus aktiviert und deaktiviert werden. Zusätzlich ist es möglich, im HiL-Modus

eine Visualisierung einzuschalten. Ist die Visualisierung eingeschaltet, werden die Sollwerte nicht nur an das Buggy interne Modell geschickt, sondern zusätzlich auch an die Lenkung und den Antrieb kommandiert. Hierbei wird der Buggy nur gesteuert, da der eigentliche Regelkreis über das interne Modell geschlossen wird.

In jedem Buggy ist ein CANcaseXL Log der Vector Informatik GmbH verbaut und an den CAN-Bus angeschlossen. Damit ist es möglich, per USB an einem PC, beispielsweise einem auf dem Buggy montierten Laptop, die CAN-Nachrichten live mitzulesen. Zusätzlich ist das CANcaseXL Log in der Lage, auch ohne angeschlossenen PC alle Daten, die auf dem CAN-Bus übertragen werden, aufzuzeichnen. Mittels USB können sie später zur Auswertung auf dem PC ausgelesen werden.

2.9 Pfadregelung

In [39] wird ein Regler vorgestellt, mit dem die Buggys entlang von vorgegebenen Pfaden geregelt werden können. Als Pfad wird dabei eine Abfolge von Punkten betrachtet, die durch Geraden miteinander verbunden werden. Der Regler wird in dieser Arbeit als Pfadregelung bezeichnet und in diesem Abschnitt, basierend auf [39] und [38], vorgestellt. Die Pfadregelung zeichnet sich dadurch aus, dass sie mit Pfaden umgehen kann, die ihr als Liste von Positions-Punkten vorgegeben werden. Außerdem ist sie sehr robust, da sie, auch wenn die Punktfolge keine fahrbare Trajektorie bildet, eine Lösung findet. Da der Pfadregelung direkt eine Liste von Positionen vorgegeben werden kann, eignet sie sich sehr gut für das Arbeiten mit Kartendaten. So lassen sich Pfade beispielsweise im OSM-Format sehr einfach abbilden. Die Fähigkeit, die Kartendaten direkt verarbeiten zu können, setzt die Pfadregelung von vielen anderen Reglern ab. Meist benötigen diese Regler eine fahrbare Trajektorie als Vorgabe. Dies wiederum macht es nötig, aus den in einer Karte hinterlegten Pfaden zunächst eine fahrbare Trajektorie zu generieren. Üblicherweise kommt dafür ein dem Regler vorgeschalteter Trajektoriengenerator zum Einsatz.

Die Pfadregelung verwendet das in [39] eingeführte Fahrzeugmodell. Es wird an dieser Stelle kurz vorgestellt. Zunächst werden einige Annahmen und Definitionen getroffen. Als betrachteter Referenzpunkt (X, Y) für die Fahrzeugposition wird der Mittelpunkt der Hinterachse definiert. Des weiteren wird davon ausgegangen, dass es keinen Schlupf und damit auch keinen Seitenschlupfwinkel gibt. Außerdem werden ausschließlich Bewegungen in der Ebene betrachtet. Daher ist es ausreichend, die Nord- und Ost-Koordinaten eines lokalen NED-Koordinatensystems zu berücksichtigen. Im Folgenden wird die Ostkomponente als X und die Nordkomponenten als Y bezeichnet.

Abbildung 2.10 zeigt das verwendete Modell. Weiterhin wird davon ausgegangen, dass der Referenzpunkt eine kreisförmige Bewegung um den Kreismittelpunkt (X_0, Y_0) durchführt. Der Radius R der Bewegung kann mit dem Radstand L und dem aktuell vorliegenden

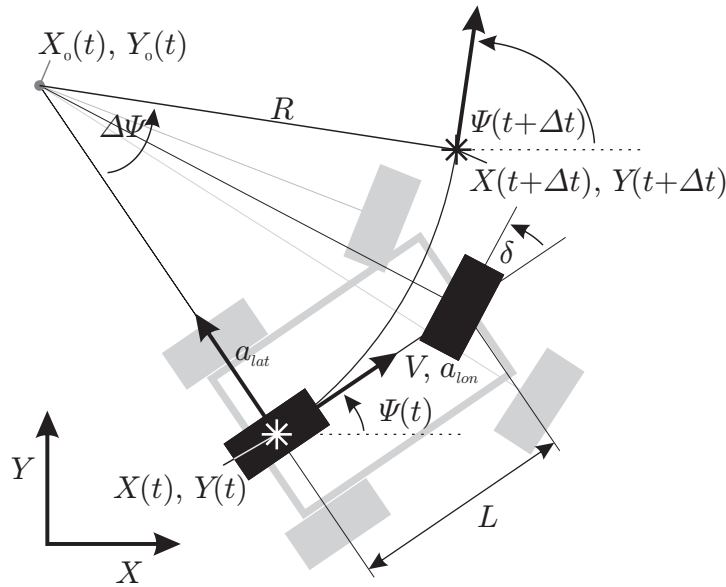


Abb. 2.10: Einspurmodell zur Vorhersage der Bewegung des Referenzpunktes [38]

Lenkwinkel δ gemäß

$$R = \frac{L}{\tan(\delta)} \quad (2.37)$$

bestimmt werden. Als Referenz für den Lenkwinkel δ kann δ_{REF} vorgegeben werden, die dann von den Lenkung des Buggys eingeregelt wird. Dabei wird angenommen, dass das Einstellen des Lenkwinkels dem Zeitverlauf eines Haltegliedes erster Ordnung entspricht und sich gemäß der DGL

$$\dot{\delta} = \frac{1}{0,1 \text{ s}} (\delta_{REF} - \delta) \quad (2.38)$$

darstellen lässt. Mit dem sich aus dem Lenkwinkel δ ergebenden Radius R und dem Gierwinkel Ψ lassen sich die Koordinaten des Kreismittelpunkts (X_0, Y_0) als

$$X_0 = X - R \cdot \sin(\Psi) \quad (2.39)$$

$$Y_0 = Y + R \cdot \cos(\Psi) \quad (2.40)$$

ermitteln. Zur Berechnung zukünftiger Punkte ergeben sich die folgenden Vorschriften:

$$\begin{aligned} \Delta\Psi &= \frac{V \cdot \Delta t}{R} \\ X(t + \Delta t) &= X_0(t) + R \cdot \sin(\Psi + \Delta\Psi) \\ Y(t + \Delta t) &= Y_0(t) - R \cdot \cos(\Psi + \Delta\Psi) \\ \Psi(t + \Delta t) &= \Psi(t) + \Delta\Psi. \end{aligned} \quad (2.41)$$

Dabei bezeichnet V die aktuelle Fahrzeuggeschwindigkeit und Δt die zeitliche Schrittweite des Modells. [39]

Der Vorderachsmittelpunkt (X_{VA}, Y_{VA}) kann aus dem Hinterachsmittelpunkt (X, Y) zum beliebigen Zeitpunkt T berechnet werden. Es gilt

$$\begin{aligned} X_{VA}(T) &= X(T) + L \cdot \cos(\Psi(T)) \\ Y_{VA}(T) &= Y(T) + L \cdot \sin(\Psi(T)). \end{aligned} \quad (2.42)$$

Der in dem Modell betrachtete Lenkwinkel δ ist durch den maximalen Lenkwinkel δ_{max} beschränkt. Dieser soll zwei Bedingungen erfüllen. Einerseits darf der mechanisch maximale Lenkwinkel $\delta_{max, mech}$, der bei etwa 28° liegt, nicht überschritten werden. Andererseits soll die maximale Querbeschleunigung a_{lat} während der Kurvenfahrt beschränkt sein. Mit der Querbeschleunigung a_{lat} kann gemäß

$$r_{min, a_{lat}}(V) = \frac{V^2}{a_{lat}} \quad (2.43)$$

zurückgerechnet werden, welcher minimale Radius $r_{min, a_{lat}}$ geschwindigkeitsabhängig angenommen werden kann, ohne die Beschränkung der Querbeschleunigung zu verletzen [39]. Daraus lässt sich mit dem Radstand L der Buggys bestimmen, welcher maximale Lenkwinkel $\delta_{max, a_{lat}}$ nicht überschritten werden darf. Es ergibt sich nach dem Umstellen von (2.37)

$$\delta_{max, a_{lat}} = \arctan\left(\frac{L}{r_{min, a_{lat}}}\right) = \arctan\left(\frac{L \cdot a_{lat}}{V^2}\right). \quad (2.44)$$

Unter Einhaltung der beiden Beschränkungen muss δ_{max} als das Minimum von $\delta_{max, a_{lat}}$ und $\delta_{max, mech}$ gewählt werden [39]. Im Rahmen dieser Arbeit wird für eine maximale Querbeschleunigung von $a_{lat} = 7 \text{ m/s}^2$ und den Radstand der Buggys $L = 0,69 \text{ m}$ ab einer Geschwindigkeit von mehr als $V = \sqrt{\frac{4,83 \text{ m}^2/\text{s}^2}{\tan(28^\circ)}} \approx 3,014 \text{ m/s}$ der maximale Lenkwinkel durch $\delta_{max, a_{lat}}$ bestimmt. Es ergibt sich also für δ_{max} die Funktion

$$\delta_{max}(V) = \begin{cases} \arctan\left(\frac{4,83 \text{ m}^2/\text{s}^2}{V^2}\right) & \text{für } V \geq \sqrt{\frac{4,83 \text{ m}^2/\text{s}^2}{\tan(28^\circ)}} \\ 28^\circ & \text{sonst.} \end{cases} \quad (2.45)$$

Die Pfadregelung entspricht dem in [39] und [38] vorgestellten sogenannten „Two and a half Carrots“-Regelungsschema, das im Folgenden basierend auf [39] und [38] erörtert wird. Bei dem Schema handelt es sich um eine nichtlineare MPR, deren Kostenfunktion

$$J_{Path} = J_R + J_{CP} + J_{WD} \quad (2.46)$$

sich aus drei Teilen zusammensetzt. Hierbei entspricht der erster Term J_R einer Bestrafung hoher Stellenergien. Es gilt

$$J_R = \lambda \cdot \sum_{i=0}^{N_u-1} (\Delta u(k + i \cdot K_{step}|k))^2. \quad (2.47)$$

Damit ähnelt J_R dem zweiten Term von (2.32), wobei die Wichtungsmatrix \mathbf{R} als Diagonalmatrix mit dem Eintrag λ auf der Hauptdiagonalen gewählt wurde. Es gilt also $\mathbf{R} = \lambda \mathbf{I}$, woraus folgt, dass alle Stellaufwände gleich gewichtet und keine gemischten Terme berücksichtigt werden. Anders im Vergleich zu (2.32) ist, dass hier nicht die N_u nächsten Zeitschritte betrachtet werden, sondern ein ganzzahliger Faktor $K_{step} \in \mathbb{N}$ eingeführt wird, dessen Vielfaches betrachtet wird. Dies kommt daher, dass die Prädiktion iterativ in sehr kleinen Zeitschritten von $T_{iter} = 0,05 \text{ s} = \frac{T_{sample}}{2}$ durchgerechnet wird, ein Lenkwinkel aber für einen längeren Zeitraum gleich gehalten werden soll. Durch dieses Vorgehen werden hochfrequente Änderungen vermieden, was eine sanftere Bewegung zur Folge hat. Die Abtastzeit des Reglers T_{sample} beträgt 0,1 s. Es ist die Zeit, die vergeht, bis ein neues Eingangssignal ausgerechnet und kommandiert wird. Die Dauer, für die ein Lenkwinkel δ innerhalb der Prädiktion konstant vorgegeben wird, ergibt sich als

$$T_{step} = K_{step} \cdot T_{sample}. \quad (2.48)$$

K_{step} wird dabei abhängig von N_u und der aktuellen Geschwindigkeit V bestimmt. Es soll gelten, dass $T_{\frac{\pi}{2}} = N_u \cdot T_{step}$ der Zeit entspricht, die benötigt wird, um mit der aktuellen Geschwindigkeit V und voll eingeschlagener Lenkung eine 90°-Kurve zu durchfahren. N_u ist der Stellhorizont und gibt an, wie viele unterschiedliche Lenkstellungen später bei der Prädiktion angenommen werden können. Jeder Winkel liegt dabei für die Zeit T_{step} an. Um $T_{\frac{\pi}{2}}$ zu bestimmen, kann mit (2.37) und (2.45) der minimal fahrbare, geschwindigkeitsabhängige Radius bestimmt werden, der multipliziert mit $\frac{\pi}{2}$ die zu fahrende Strecke für die Kurve ergibt. Wird diese Strecke durch den Betrag der Geschwindigkeit V dividiert, ergibt sich die Zeit, die nötig wäre, um die 90°-Kurve zu fahren. Es gilt folglich

$$T_{\frac{\pi}{2}} = \frac{\frac{L}{\tan(\delta_{max}(V))} \cdot \frac{\pi}{2}}{|V|}. \quad (2.49)$$

Daraus lässt sich mit dem Stellhorizont N_u der Wert für K_{step} als

$$K_{step} = \left\lceil \frac{T_{\frac{\pi}{2}}}{N_u \cdot T_{sample}} \right\rceil \cdot N_u \quad (2.50)$$

berechnen.

Der zweite Term der Kostenfunktion J_{Path} , J_{CP} , berücksichtigt die prädizierte Abweichung von der vorgegebenen Strecke. Hierzu werden die zukünftigen Fahrzeugpositionen mithilfe des Modells aus Abbildung 2.10 prädiziert. Zwei als Karotten-Punkte (engl. carrot points) CP_1 und CP_2 bezeichnete Positionen werden ausgewählt und gehen in J_{CP} mit ein. Sie entsprechen der Position des Hinterachsmittelpunktes zu bestimmten Zeiten, vgl. Abbildung 2.11 unten. Die schwarze Linie in der Abbildung zeigt den sich aus drei Punkten ergebenden Pfad, der vom Map Matching verwendet wird. Wie zuvor beschrieben gilt, dass während der Prädiktion N_u unterschiedliche Lenkwinkel jeweils für die Zeit T_{step} angelegt werden. Der Punkt CP_1 wird dabei während des ersten gestellten Winkels berechnet und zwar nachdem die Zeit $t_{C1} = T_{sample} + \frac{T_{step}}{2}$ prädiziert wurde, vgl. Abbildung 2.11 oben. Am Ende der Prädiktion, also nach der Zeit $t_{C2} = T_{sample} + N_u \cdot T_{step}$, wird der Punkt CP_2 berechnet. Die Punkte CP_1 und CP_2 befinden sich frei im Raum und

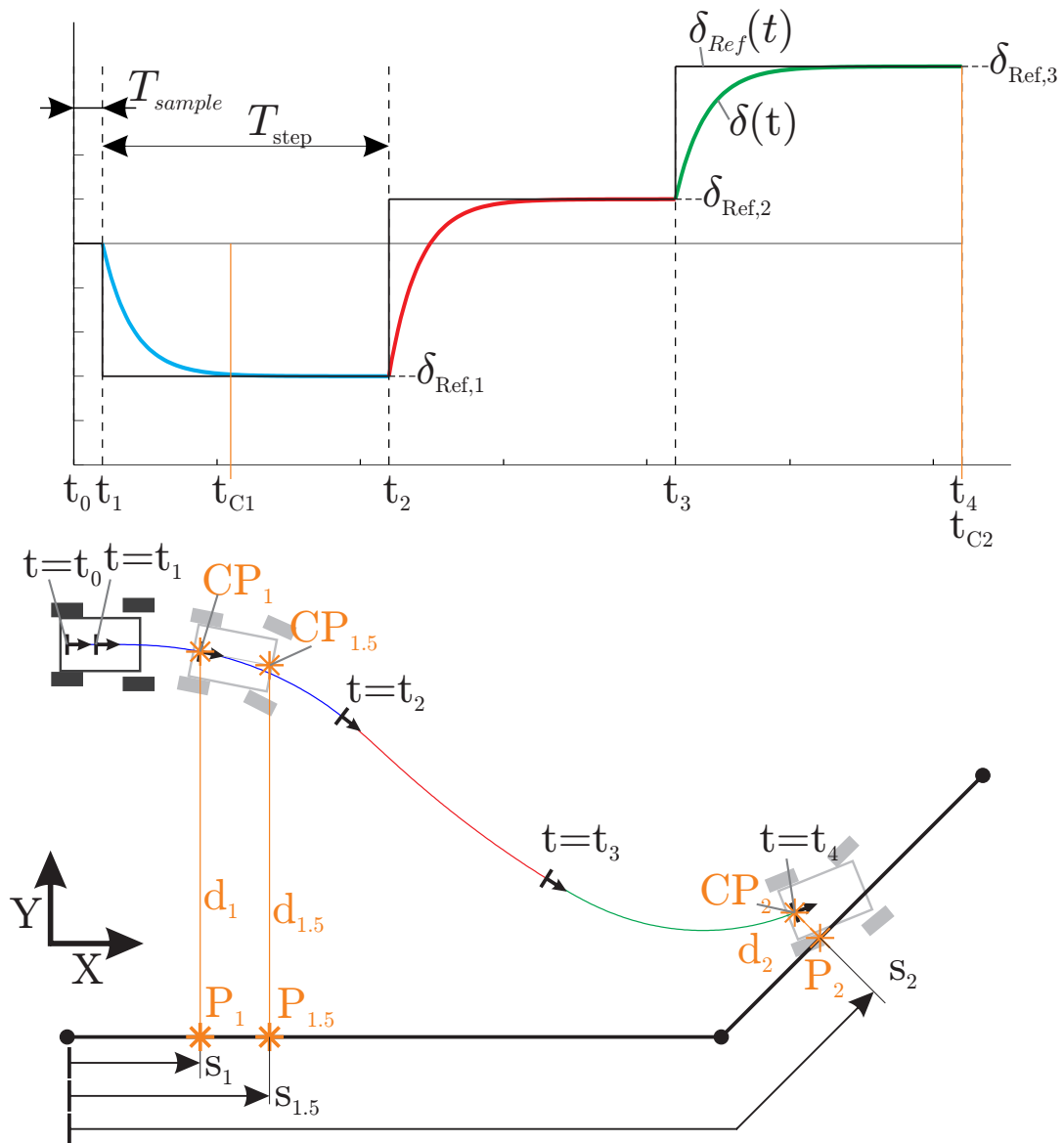


Abb. 2.11: Vorhergesagter Verlauf des Lenkwinkels (oben) und der Fahrzeugposition (unten) [38]

es muss noch ein Bezug zu dem Pfad hergestellt werden. Dazu wird für die Punkte ein Map Matching durchgeführt, mit dem zu jedem Punkt sein zugehöriger Map Matching-Punkt in Gleis- und NED-Koordinaten bestimmt wird. Zwischen einem Karotten-Punkt und seinem zugehörigen Map Matching-Punkt P_1 bzw. P_2 kann der euklidische Abstand d aus den NED-Koordinaten bestimmt werden. Dieser Abstand wird d_1 für CP_1 genannt und analog d_2 für CP_2 . Zusätzlich gibt es einen *halben* Karotten-Punkt. Dieser beschreibt die Position des Vorderachsmittelpunktes zum Prädiktionszeitpunkt t_{C1} , zu dem auch CP_1 ermittelt wird. Er wird als $CP_{1.5}$ bezeichnet und auch für ihn wird wie beschrieben der Abstand $d_{1.5}$ zu seinem Map Matching-Punkt $P_{1.5}$ auf der Referenzstrecke berechnet. Durch Einführen von $CP_{1.5}$ kann die Robustheit der Regelung gesteigert werden. Er bestraft Trajektorien, bei denen versucht wird, möglichst schnell mit dem Mittelpunkt der Hinterachse an den Pfad zu gelangen, ohne dabei die Orientierung zu berücksichtigen. Der Beitrag der Karotten-Punkte zur Kostenfunktion ergibt sich als

$$J_{CP} = d_1^2 + d_{1.5}^2 + d_2^2. \quad (2.51)$$

Der Term J_{WD} der Kostenfunktion J_{Path} sorgt für eine eindeutige Lösung, indem er das Fahren entgegen der Fahrtrichtung bestraft. Fahrzeug und Strecke besitzen jeweils eine definierte Fahrtrichtung. Bewegt sich das Fahrzeug in die falsche Richtung (engl. wrong direction), ist die Fahrtrichtung nicht im Einklang mit der Strecke. Dies wird durch

$$J_{WD} = \begin{cases} 0 & \text{für } V \geq 0 \wedge s_2 > s_1 \\ 10\,000 & \text{für } V \geq 0 \wedge s_2 < s_1 \\ 0 & \text{für } V < 0 \wedge s_2 < s_1 \\ 10\,000 & \text{für } V < 0 \wedge s_2 > s_1 \end{cases} \quad (2.52)$$

bestraft. Hierbei bezeichnen s_1 und s_2 die Streckenlänge der den Karotten-Punkten zugehörigen Gleiskoordinaten aus den zuvor beschriebenen Map Matching-Punkten. Sie stehen also für die prädizierte Bewegungsrichtung entlang der Kartendaten. Der Punkt $CP_{1.5}$ bzw. dessen zugehörige Streckenlänge geht nicht mit in J_{WD} ein. Mit der Fahrzeuggeschwindigkeit V , die für die Prädiktion als konstant angenommen wird, kann aus dem Vorzeichen von V die Fahrzeugbewegungsrichtung bestimmt werden. Stimmt die Bewegungsrichtung mit der Streckenorientierung überein, so gibt es keinen Beitrag durch J_{WD} . Andernfalls folgt einen Strafwert von 10 000. Der Wert ist so hoch gewählt, dass er im Vergleich zu den anderen Termen immer überwiegt und daher in jedem Fall vermieden wird.

Um die Pfadregelung einzustellen, kann λ angepasst werden. Im Rahmen dieser Arbeit gilt wie in [38], $\lambda = 0,0001$. Zusätzlich kann über den Stellhorizont N_u eingestellt werden, wie weit in die Zukunft prädiziert wird. Der Stellhorizont stellt gleichzeitig auch den Prädiktionshorizont dar. Außerdem gibt er vor, wo die Karotten-Punkte CP_1 und CP_2 liegen, vgl. Abbildung 2.11. Es wird bis zum Erreichen des Punktes CP_2 prädiziert, um die Kostenfunktion bestimmen zu können. Im Rahmen dieser Arbeit werden dieselben Anforderungen an diese Karottenpunkte gestellt wie in [38]. Das bedeutet, dass wie bereits beschrieben die Punkte CP_1 , $CP_{1.5}$ und CP_2 zu den sich aus $T_{\frac{\pi}{2}}$ ergebenden Zeitpunkten t_{C1} bzw. t_{C2} bestimmt werden.

Mit der beschriebenen Kostenfunktion J_{Path} kann das Optimierungsproblem der Pfadregelung gemäß

$$\begin{aligned} \mathbf{u}_{opt}(\cdot|k) &= \arg \min_{\mathbf{u}(\cdot|k)} J(\mathbf{u}(\cdot|k)) \\ \text{mit der Stellgrößenbeschränkung} \\ \mathbf{u}(\cdot|k) &\in \left\{ \mathbb{R}^{N_u} \mid -\delta_{max}(V) \leq u(k+i|k) \leq \delta_{max}(V) \quad \forall i \in \{1, 2, \dots, N_u\} \right\} \end{aligned} \quad (2.53)$$

aufgestellt und gelöst werden.

Zur Lösung des Optimierungsproblems (2.53) wird der ebenfalls in [38] vorgestellte iterative Algorithmus mit den Parametern Stellhorizont $N_u = 3$, Anzahl Iterationen $n_{iter} = 3$ und Anzahl untersuchte Werte pro Iteration $n_{delta} = 5$ verwendet. Zunächst wird das Intervall $[-\delta_{max}, \delta_{max}]$ äquidistant in n_{delta} Werte unterteilt. Diese Werte werden als Stellwerte über den Stellhorizont N_u verwendet. Es ergeben sich also $n_{delta}^{N_u} = 125$ Kombinationen, für die der Wert der Kostenfunktion J in der ersten Iteration bestimmt wird. Diejenige Stellwertfolge u_1, u_2, u_3 für die J das Minimum annimmt, wird als Startwertfolge für den nächsten Iterationsschritt verwendet. Die Intervalle, in denen nun erneut mit 5 äquidistant verteilten Werten gesucht wird, werden verkleinert auf $\left[u_i - \frac{\delta_{max}}{5}, u_i + \frac{\delta_{max}}{5} \right]$ für $i \in \{1, 2, 3\}$. Es ergibt sich eine neue Stellwertfolge, die J minimiert. Auch diese dient als Startwerte für den letzten Iterationsschritt. Ebenso wie zuvor werden die Suchintervalle erneut verkleinert und äquidistant in $n_{delta} = 5$ Werte aufgeteilt. Die Intervalle lauten für die neu gefundenen u_i $\left[u_i - \frac{\delta_{max}}{125}, u_i + \frac{\delta_{max}}{125} \right]$ für $i \in \{1, 2, 3\}$. Es wird erneut die Folge gefunden, die J minimiert. Der erste Stellwert u_1 dieser Folge ist derjenige Stellwert, der an den Buggy kommandiert wird. Für jeden Zeitschritt erfolgt eine erneute Berechnung von u_1 nach diesem iterativen Verfahren.

Die Pfadregelung zeichnet sich durch die hohe Robustheit bezogen auf ihre Anforderungen an den Pfad aus. In Kombination mit dem Map Matching reichen ihr die Abfolge beliebiger Punkte als Pfadvorgabe aus. Der Regler wird eine optimale Trajektorie finden. Die Robustheit wird durch ein rechenintensives iteratives Optimierungsverfahren erkauft. In der Anwendung der virtuellen Schiene ist diese Robustheit aber nicht unbedingt nötig. Der virtuelle Zug soll das Verhalten von Zügen abbilden. Züge bewegen sich auf Schienen, die so verlegt sind, dass sie von den Zügen problemlos befahren werden können. Daher werden auch die Pfade in der Anwendung der virtuellen Schiene in aller Regel fahrbare Pfade sein. In Kapitel 3 wird daher ein weiteres Verfahren motiviert, bei dem die Anforderung an die Pfadvorgaben in Bezug auf die Fahrbarkeit höher sind, dafür aber der Rechenaufwand geringer ist. Der Regler wird in Abschnitt 4.3 im Detail vorgestellt.

2.10 Mann-in-der-Mitte-Technik

Bei der Mann-in-der-Mitte-Technik oder auch dem Mann-in-der-Mitte-Angriff wird die Kommunikation von mindestens zwei Kommunikationsteilnehmern durch einen weiteren

überwacht oder manipuliert. Um dies zu erreichen, gibt der Angreifer sich gegenüber den anderen als deren jeweiliger Kommunikationspartner aus [30].

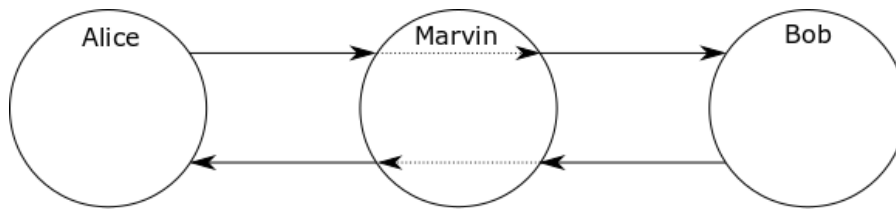


Abb. 2.12: Mann in der Mitte Prinzip. Marvin (in der Mitte) belauscht die Unterhaltung von Alice und Bob.

Beispiel 2.1 *Alice und Bob wollen sich unterhalten. Marvin möchte die Unterhaltung der beiden unbemerkt belauschen. Um sein Ziel zu erreichen, stellt sich Marvin gegenüber Alice als Bob vor und als Bob gegenüber Alice. Auf diese Weise läuft die Kommunikation zwischen Alice und Bob immer über Marvin, wie in Abbildung 2.12 dargestellt. Alles, was Marvin übermittelt bekommt, gibt er an den richtigen Adressaten weiter, besitzt die Informationen aber auch selbst.*

Ohne größere Probleme lässt sich vorstellen, dass Marvin in Beispiel 2.1 nicht nur in der Lage ist, die Kommunikation von Alice und Bob abzuhören, sondern auch zu manipulieren. Dabei bestehen für Marvin die Möglichkeiten, eine Nachricht nicht weiter zuleiten, sie zu verändern oder durch eine eigene Nachricht zu ersetzen. So ist er in der Lage, das Verhalten von Alice und Bob entsprechend seinen Wünschen zu manipulieren.

Im Rahmen dieser Arbeit wird das Verfahren genutzt, um das in Abschnitt 5.2 vorgestellte Testfahrzeug zu automatisieren.

3 Aufgabenstellung

Die Buggys können als Entwicklungswerkzeug verwendet werden und stehen bei der Funktionsentwicklung zwischen reiner Simulation und vollwertigem Experiment. Durch ihren modularen Aufbau und die vorhandenen Montagemöglichkeiten können sie sehr schnell und einfach mit zusätzlicher Hardware für die zu entwickelnde Anwendung ausgerüstet werden. So können die unterschiedlichen Hardwarekomponenten bereits früh zusammen betrieben und etwaige Wechselwirkungen früh erkannt werden. Auch die Kommunikation mit Sensoren und Aktoren kann getestet werden, ohne ein vollwertiges Testfahrzeug aufbauen zu müssen. Dank der verwendeten Echtzeithardware der Buggys kann schnell zwischen reiner Simulation und realen Experimenten mit den Buggys gewechselt werden. Dabei ist die Verwendung unterschiedlicher RCP-Methoden, wie HiL, SiL und Model in the Loop (MiL), möglich.

Im Rahmen dieser Arbeit sollen die Buggys verwendet werden, um die Entwicklung und Validierung des GO!-Satellitennavigationsempfängers voran zu treiben. Der GO!-Empfänger ist auf Bahnanwendungen zugeschnitten. Er ist sowohl für die Anbringung an der Lokomotive als auch an allen Wagen eines Zuges gedacht. Weitere Details zum Empfänger sind in Abschnitt 2.4 bzw. in [56] zu finden.

Um die Buggys sinnvoll zur Entwicklung und Validierung des Empfängers nutzen zu können, darf es für die Empfänger keinen Unterschied machen, ob sie auf den Buggys montiert sind oder auf einem Zug. Daher müssen sich die Buggys wie ein Zug verhalten und bewegen. Im Vergleich zu den Buggys ist ein Zug in seinem Bewegungspfad durch die Gleise, auf denen er sich bewegt, eingeschränkt. Diese Einschränkung muss folglich durch die Buggys nachgebildet werden. Im Folgenden wird die Nachbildung der Bewegung als virtuelle Schiene bezeichnet.

Die zweite wesentliche Eigenschaft eines Zuges ist die starre Verbindung der Wagen und Lokomotive. Es kommen zwar Puffer zwischen den Wagen zum Einsatz, diese erlauben aber trotzdem nur wenig Änderung des Abstandes zweier Wagen. Auch dies muss durch die Buggys nachgebildet werden. Hierbei soll weiterhin eine hohe Flexibilität möglich sein. Daher wird eine Abstandsregelung gegenüber einer mechanischen Kupplung der Buggys bevorzugt. Die von dieser Abstandsregelung umzusetzende Funktionalität wird im Folgenden als virtuelle Kupplung bezeichnet. Bei der virtuellen Kupplung ist darauf zu achten, dass die Fahrzeuge bei unterschiedlichen Geschwindigkeiten immer den selben relativen Abstand zueinander einhalten. Dies stellt einen Unterschied zu den meisten Anwendungen aus der Automobilindustrie dar. Dort wird, beispielsweise bei Abstandstempomaten, der Abstand zwischen den Fahrzeugen bei zunehmender Geschwindigkeit vergrößert.

Werden die virtuelle Schiene und die virtuelle Kupplung gleichzeitig realisiert, wird dies im Folgenden als virtueller Zug bezeichnet. Setzen die Buggys das Verhalten des virtuellen Zuges um, so können sie als Zwischenschritt im Entwicklungsprozess des GO!-Satellitennavigationsempfängers verwendet werden.

Das Ziel dieser Arbeit besteht darin, den virtuellen Zug umzusetzen und als geeignetes Werkzeug bei der Navigationsempfängerentwicklung zu validieren. Dazu erfolgt in den nächsten drei Abschnitten die Definition der Anforderungen an die virtuelle Schiene, virtuelle Kupplung und den virtuellen Zug. Anschließend wird, im letzten Absatz dieses Kapitels, aufgeführt, mit welchen Lösungsansätzen die Anforderungen erfüllt werden.

3.1 Virtuelle Schiene

Damit sich die Buggys wie auf Schienen bewegen, wird zunächst eine Vorgabe benötigt. Diese Vorgabe soll definieren, entlang welcher Pfade sich die Buggys bewegen. Eine digitale Karte eignet sich sehr gut für diesen Zweck. In ihr können Strecken abgelegt werden. Liegen diese Strecken vor, so kann mit der gemessenen Position der Buggys und durch Map Matching geschätzt werden, wo sich der Buggy auf der Strecke befindet. Es kann ein Soll-Ist-Vergleich stattfinden und so eine Abweichung bestimmt werden. Durch den Einsatz eines Reglers wird diese Abweichung minimiert, sodass die Buggys der in der Karte vorgegebenen Strecke folgen. Neben der Position spielt auch die Orientierung eine bedeutsame Rolle. Sollen die Buggys einer Strecke folgen, so ist es wichtig, dass sie die richtige Orientierung besitzen. In den in Abschnitt 2.2.4 vorgestellten Gleiskoordinaten findet eine Berücksichtigung der Orientierung statt. Auch diese kann an den Regler gegeben werden, um so die virtuelle Schiene zu realisieren.

Züge können an Weichen das Gleis wechseln. Dieser Mechanismus soll auch durch die virtuelle Schiene abgebildet werden. In den Kartendaten ist bereits die Information enthalten an welchen Punkten (Weichen) von einer Strecke auf mehrere andere Strecken gewechselt werden kann. Durch ein internes Abspeichern der Weichenstellung kann dem Regler vorgegeben werden, wo, an welcher Weiche und auf welche Strecke abgebogen werden soll.

Es wird also ein Regler benötigt, der dafür sorgt, dass nur Positionen auf der Strecke eingenommen werden, die Orientierung entlang der Strecke eingehalten wird und die Weichenstellungen berücksichtigt werden. Hierzu müssen dem Regler die Kartendaten bekannt sein und Geschwindigkeit respektive Beschleunigung sowie Weichenstellung von außen vorgegeben werden. Wird wie zuvor beschrieben intern ein Map Matching verwendet, sind die berechneten Gleiskoordinaten auch bei der Realisierung der virtuellen Kupplung sehr nützlich.

3.2 Virtuelle Kupplung

Bei der virtuellen Kupplung sollen die Buggys wie die Wagen eines Zuges zusammen gekuppelt werden. Im Unterschied zu dem Vorbild soll die Kopplung nicht mechanisch erfolgen. Die Buggys sollen selbstständig einen konstanten Abstand zueinander einregeln. Da bei den Buggys jeder für sich individuell agiert und eben keine mechanische Verbindung zwischen den Buggys besteht, muss der Abstand entsprechend groß genug gewählt werden, sodass im Falle einer Störung genug Abstand zum Reagieren verbleibt. So kann verhindert werden, dass die Buggys kollidieren, wenn bei einem der Fahrzeuge ein Fehler auftritt. Außerdem soll durch die virtuelle Kupplung der Buggys ein Zug nachgebildet werden, der größere Ausmaße als die Buggys besitzt. Ein Abstand zwischen den Buggys verbessert also die Nachbildung eines realen Zuges und erhöht gleichzeitig die Sicherheit.

Um den Abstand zwischen den Buggys bestimmen zu können, eignen sich die zuvor erwähnten Gleiskordinaten, die durch ein Map Matching mit der Karte gebildet werden können. Wie in Kapitel 2.2.4 vorgestellt, bestehen die Koordinaten aus einer Streckeninformation id und der auf der Strecke zurückgelegten Entfernung s . Durch simple Subtraktion der Entfernungswerte zweier Buggys kann der Abstand der Fahrzeuge untereinander entlang der hinterlegten Strecke berechnet werden. Die Berücksichtigung der Strecke bei der Entfernungsbestimmung führt zu einem besseren Ergebnis als die Bestimmung der euklidischen Entfernung der beiden Fahrzeuge. Dies kommt daher, dass beide Fahrzeuge die selbe Strecke abfahren und so auch in Kurvenfahrten der kartenbasierte Abstand ermittelt werden kann.

Zur Realisierung wird folglich ein Regler benötigt, der den Abstand zweier gekuppelter Buggys konstant hält. Dazu muss zunächst eine Unterteilung in Führungs- und Folgefahrzeug vorgenommen werden. In jedem Verband darf es nur ein Führungsfahrzeug geben, das die Sollwerte vorgibt, die von den Folgefahrzeugen eingehalten werden müssen. Da das Führungsfahrzeug ebenfalls Vorgaben für seine Geschwindigkeit bzw. Beschleunigung von außen vorgegeben bekommt, bietet es sich an, diese Vorgaben als Vorsteuerung an die Folgefahrzeuge zu kommandieren. Würden alle Fahrzeuge exakt die selbe Beschleunigung realisieren, würden sich daraus die selben Geschwindigkeiten ergeben und ein konstanter Abstand eingehalten werden. In der Realität lässt sich dies durch eine Vorsteuerung allein nicht realisieren, da Kommunikationsverzögerungen und Unterschiede der in den Buggys verbauten Komponenten schon zu Abweichungen führen würden. Daher soll ein Regler auf die Vorsteuerung aufgesetzt werden. Dieser soll die entstehenden Abweichungen in Beschleunigung, Geschwindigkeit und Position ausregeln. Dabei liegt die Sollposition in einem definierten Abstand hinter der Position des Führungsfahrzeugs.

Es werden also zwei Regler benötigt, um sowohl der Rolle als Führungsfahrzeug als auch der Rolle als Folgefahrzeug gerecht zu werden. Zur Realisierung der virtuellen Kupplung ist es also nötig, zu wissen welcher Regler gerade aktiv sein soll. Hierzu ist ein Datenaustausch der Buggys untereinander notwendig. So kann ermittelt werden, welcher Buggy aus einem Verband das Führungsfahrzeug ist. Außerdem können so die für die Vorsteuerung nötigen Informationen ausgetauscht werden. Als Eingangsgrößen der

Regelung werden die Beschleunigung, Geschwindigkeit und aktuelle Position des Führungsfahrzeuges benötigt sowie die entsprechenden eigenen Messwerte jedes Fahrzeugs. Damit können die Abweichungen bestimmt und ausgeregelt werden.

3.3 Szenarioleitstelle und Virtueller Zug

Um den virtuellen Zug realisieren zu können, ist es von Nöten, dass alle beteiligten Fahrzeuge koordiniert werden. Die sogenannte Szenarioleitstelle (SLS) übernimmt diese Aufgabe. Sie dient als virtuelles Stellwerk, Zugbildungssystem und als Überwachungseinheit. Sie wird als eigenständige Komponente vorgesehen, die in die Buggy-Kommunikation eingebunden ist. Die Buggys empfangen und senden Nachrichten an die SLS.

Die Aufgabe als Überwachungseinheit hat zwei wesentliche Aspekte. Einerseits soll es möglich sein, alle Buggys im Fehlerfall abzuschalten. Hierzu kann die SLS eine entsprechende Nachricht an alle Buggys senden. Andererseits wird der aktuelle Status der Buggys von der SLS empfangen und angezeigt. So ist der Bediener in der Lage, z. B. bei niedrigem Akkustand zu reagieren und das Fahrzeug zu laden bzw. die Batterie auszutauschen. Außerdem können die fahrzeuginternen Reglerstatus überwacht werden, was eine Analyse von Fehlverhalten erleichtert. Die Daten können bei Bedarf auch aufgezeichnet werden, um im Postprocessing zur Auswertung der Messfahrten genutzt zu werden.

Um die Aufgabe als virtuelles Stellwerk zu realisieren, wird in der SLS die selbe Karte hinterlegt wie in allen Buggys. Es werden also die in Kapitel 2.5 vorgestellten Matrizen und Vektoren mit den gleichen Initialwerten hinterlegt. Das bedeutet, alle Buggys besitzen den selben Gleisplan und verhalten sich an den Wegenden gleich. Alle biegen auf denselben Weg ab. Soll nun an einer Weggabelung anders abgebogen werden, müssen dafür die entsprechenden Datenvektoren in allen Buggys angepasst werden. Hierzu sendet die SLS entsprechende Änderungsnachrichten an die Buggys. Im Kontext des virtuellen Zuges wird dies als *Weichen stellen* definiert, was erklärt, warum die SLS als virtuelles Stellwerk agiert.

Zum Stellen einer Weiche kann die SLS in der entsprechenden Matrix überprüfen, welche Wege überhaupt zulässig sind, weil sie an das entsprechende Wegende grenzen. Wenn die Wege zueinander passen, wird eine entsprechende Nachricht an alle Fahrzeuge gesendet. In der Nachricht steht, welche Einträge die Fahrzeuge bei sich ändern sollen. Nachdem ein Fahrzeug seine Werte angepasst hat, wird eine Bestätigung zur SLS gesendet. Die SLS verwaltet intern den Status aller Weichen. Wird eine Weiche umgelegt, setzt sie den Status dieser Weiche intern auf undefiniert und schickt die entsprechende Nachricht zum Umlegen der Weiche an alle Buggys. Sobald alle Fahrzeuge die Änderung bestätigt haben, wird die Weiche als umgestellt betrachtet.

Als Zugbildungssystem muss die SLS in der Lage sein, zu verwalten, welche Fahrzeuge miteinander gekuppelt sind. Bevor Fahrzeuge virtuell gekuppelt werden, muss zunächst geprüft werden, ob dies gefahrlos möglich ist. Anhand der Positionen in Gleiskoordinaten kann die SLS prüfen, ob die zu kuppelnden Fahrzeuge nah genug beieinander stehen

und dass kein anderes Fahrzeug zwischen den zu kuppelnden Fahrzeugen steht. Wenn dies gegeben ist, wird die Konfiguration entsprechend angepasst und eine entsprechende Nachricht zur Aktivierung der virtuellen Kupplung an die Buggys gesendet. Da es vorkommen kann, dass die Nachricht nicht ordnungsgemäß übertragen wurde, ist auch hier ein Rückkanal vorgesehen. Wie auch beim Stellen der Weichen werden solange Nachrichten an die Buggys gesendet, bis die neue Konfiguration bestätigt wird.

Das Gesamtkonzept des virtuellen Zuges kann also, wie in diesem Abschnitt geschehen, als das Zusammenspiel von virtueller Schiene, virtueller Kupplung und der SLS betrachtet werden. Dabei stellt die SLS die Schnittstelle des virtuellen Zuges nach außen dar. Das Stellen von Weichen und das Kuppeln von Fahrzeugen kann über sie vorgenommen werden - entweder durch einen Bediener, der die Änderungen manuell in der SLS einträgt, oder automatisiert über definierte Nachrichten an die SLS.

3.4 Lösungsansätze und Zielsetzung

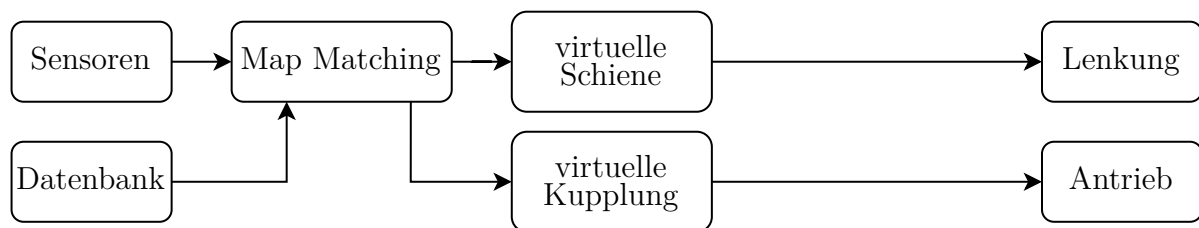


Abb. 3.1: Prinzipieller Aufbau des virtuellen Zuges

Die Zielsetzung dieser Arbeit ist die Umsetzung der zuvor vorgestellten Ansätze der virtuellen Schiene, der virtuellen Kupplung und des virtuellen Zuges, zu denen auch die SLS gehört. Dazu gilt es, mehrere Aufgaben, die teilweise getrennt voneinander betrachtet werden können, zu lösen. Abbildung 3.1 zeigt, wie der virtuelle Zug prinzipiell innerhalb eines Buggys aufgebaut sein kann. Mit den (Karten-)Daten aus einer Datenbank und den Positionsmessungen erfolgt ein Map Matching. Dies liefert die eigene Position in Gleiskoordinaten. Unter Einhaltung des Konzepts der virtuellen Schiene werden daraus Anforderungen an die Lenkung definiert. Die virtuelle Kupplung erzeugt entsprechende Anforderungen an den Antrieb der Buggys. Zu den Sensoren in Abbildung 3.1 wird auch die Kommunikation mit den anderen Buggys und der SLS gezählt.

Im Folgenden werden zwei Ansätze zur Realisierung des virtuellen Zuges betrachtet. Abbildung 3.2 zeigt den prinzipiellen Aufbau der beiden Ansätze. Im Rahmen dieser Arbeit wird der erste Ansatz ebenfalls als Pfadregelung bezeichnet, wie auch schon der Regler, auf dem er basiert. Der bei diesem Ansatz verwendete Regler ist der in Abschnitt 2.9 bzw. in [39] vorgestellte Regler. Er setzt die virtuelle Schiene um. Abbildung 3.2a zeigt den Aufbau des Ansatzes der Pfadregelung. Wie in Abschnitt 3.2 ausgeführt und in Abbildung 3.2 abgebildet, gibt es zwei Szenarien bei der virtuellen Kupplung. Ein Buggy kann als Lokomotive oder Wagen agieren. Dementsprechend werden zwei Konzepte benötigt,

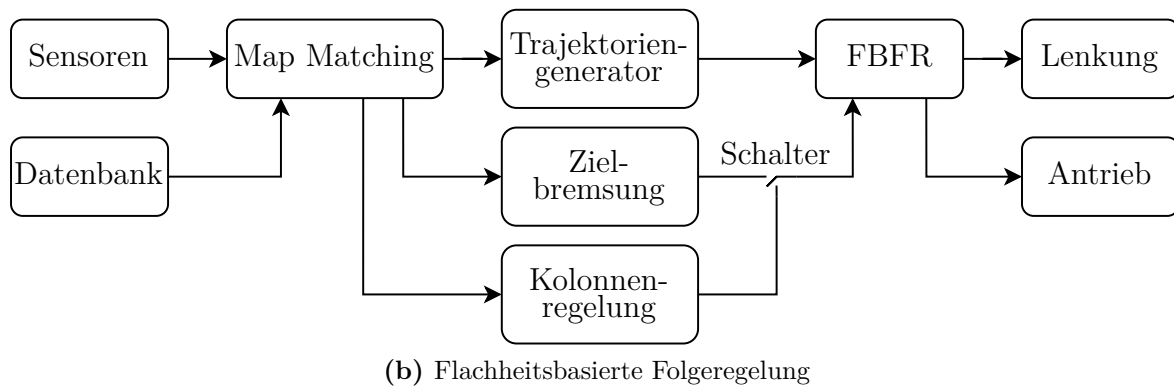
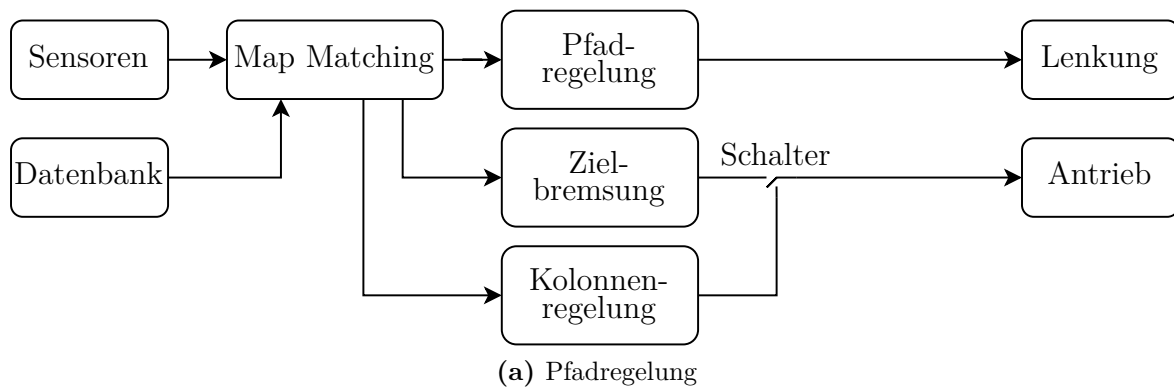


Abb. 3.2: Aufbau der Regelungsansätze Pfadregelung und Flachheitsbasierte Folgeregelung

zwischen denen ausgewählt wird. Die sogenannte Zielbremsung stellt dabei die Regelung der Buggys als Lokomotive dar. Wird ein Buggy als Wagen verwendet, so wird der im Rahmen dieser Arbeit als Kolonnenregelung bezeichnete Regler verwendet. Sowohl Zielbremsung als auch Kolonnenregelung werden im Rahmen dieser Arbeit vorgestellt.

Der Ansatz der Pfadregelung wird verwendet, da er sehr robust ist und direkt auf die Kartendaten angewendet werden kann, auch wenn in diesen nicht-fahrbare Pfade hinterlegt sind. Ein Beispiel für nichtfahrbare Pfade sind Abzweigungen, an denen auf der Stelle im 90°-Winkel abgebogen werden soll. Die Pfadregelung kann mit solchen Pfaden umgehen und wird inhärent dafür sorgen, dass der Buggy dem Pfad weitestgehend folgt und nur in der Umgebung der nichtfahrbaren Stelle des Pfades von diesem abweicht. Bei anderen Regelungsverfahren wird ein Trajektoriengenerator benötigt, der aus den Pfaden in der Karte Trajektorien als Regelvorgabe erzeugt. Die Pfadregelung kann also direkt - wie in Abschnitt 2.9 vorgestellt - verwendet werden.

Da mit der virtuellen Schiene Strecken realisiert werden sollen, die von Zügen gefahren werden können, kann davon ausgegangen werden, dass die in den Kartendaten hinterlegten Pfade fahrbar sind. Daher ist es nicht nötig, einen Ansatz zu verfolgen, der auch mit nicht fahrbaren Pfaden umgehen kann. Zudem ist diese Fähigkeit bei der Pfadregelung durch ein rechenintensives iteratives Optimierungsverfahren realisiert. Daher wird ein

weiterer Ansatz zur Realisierung des virtuellen Zuges vorgestellt, die Flachheitsbasierte Folgeregelung (FBFR). Der Aufbau der Flachheitsbasierte Folgeregelung (FBFR) ist in Abbildung 3.2b abgebildet. Wie aus dem Namen abgelesen werden kann, nutzt die FBFR die Flachheitseigenschaft der Buggys aus. Dieser Ansatz ist deutlich weniger rechenintensiv. Er stellt aber die Anforderung, dass dem Regler stetige Trajektorien vorgegeben werden müssen. Hierzu müssen die Pfade aus den Kartendaten durch einen Trajektoriengenerator als Trajektorien bereitgestellt werden. Der nötige Trajektoriengenerator wird als Teil des Regelungsansatzes der FBFR betrachtet. Gemeinsam mit der FBFR selbst wird er im Rahmen dieser Arbeit entwickelt.

Ein wesentlicher Unterschied der Ansätze aus Abbildung 3.2 ist die strikte Trennung von Längs- und Querregelung bei der Pfadregelung, wohingegen durch die FBFR gleichzeitig Einfluss auf Längs- und Querregelung genommen wird. Außerdem zeigt Abbildung 3.2, dass die FBFR nicht direkt verwendet wird. Der Regelung muss der erwähnte Trajektoriengenerator vorangestellt werden. Ferner ist zu erkennen, dass Zielbremsung und Kolonnenregelung in beiden Ansätzen vorhanden sind. Sie sind für beide Ansätze identisch. Auf die Pfadregelung wird nicht weiter eingegangen, da sie aus dem Stand der Technik bereits bekannt ist, siehe Abschnitt 2.9. Zielbremsung, Kolonnenregelung und FBFR inkl. Trajektoriengenerator werden in Kapitel 4 vorgestellt.

Mit dieser Arbeit soll gezeigt werden, dass der virtuelle Zug allgemein als Entwicklungswerkzeug verwendet werden kann. Dabei kann er als Bindeglied zwischen Simulation und vollwertigem Experiment betrachtet werden. So schafft das Konzept einen Mehrwert, in dem es ermöglicht, regelungstechnische Ansätze im Modellmaßstab kostengünstig zu entwickeln und zu evaluieren. In Kapitel 6 wird am Beispiel des Projektes GO! untersucht, ob dieses Ziel durch den virtuellen Zug erreicht werden kann. Durch die Vorstellung und den Vergleich der beiden Verfahren Pfadregelung und FBFR soll zusätzlich gezeigt werden, dass mehrere Ansätze zielführend sind. Jedes Verfahren bringt dabei seine eigenen Vor- und Nachteile mit sich. Ein weiteres Ziel besteht darin, zu zeigen, dass die Vorteile der im Rahmen dieser Arbeit realisierten FBFR überwiegen, sodass dieses Verfahren bessere Ergebnisse erzielt. Auch dieses Ziel wird in Kapitel 6 untersucht.

4 Regelungsverfahren für den virtuellen Zug

In diesem Kapitel werden die im Rahmen dieser Arbeit entwickelten Regelungskonzepte vorgestellt. Begonnen wird in Abschnitt 4.1 mit der sogenannten Zielbremsung. Bei der Zielbremsung handelt es sich um eine Längsführung, die zum Einsatz kommt, wenn ein Buggy als Lokomotive im Kontext des virtuellen Zuges verwendet wird.

Das zweite Konzept ist die Kolonnenregelung, die in Abschnitt 4.2 ausgeführt wird. Sie realisiert im Wesentlichen die virtuelle Kupplung. Mehrere zu einem Verband zusammengeschlossene Buggys kooperieren und regeln einen festen Abstand zueinander ein. Buggys, die die Regelung verwenden, werden im Kontext des virtuellen Zuges als Wagen betrachtet.

Im letzten Abschnitt 4.3 des Kapitels wird die FBFR vorgestellt. Die FBFR stellt eine Alternative zur Pfadregelung dar. Beide Regler realisieren die virtuelle Schiene. Da anderes als bei der Pfadregelung zusätzlich zum Regler auch ein Trajektoriengenerator nötig ist, wird dieser ebenfalls in Abschnitt 4.3 vorgestellt.

4.1 Zielbremsung

Ein häufig durchgeführtes Fahrmanöver im Bahnsektor ist das Anfahren einer Zielposition. Dabei wird eine Position als Zielpunkt \mathbf{p}_z definiert, an dem angehalten werden soll. Um zu der Zielposition zu gelangen, muss der Zug in die entsprechende Fahrtrichtung beschleunigt werden. Üblicherweise gibt es eine Geschwindigkeitsbeschränkung, sodass bei der Anfahrt des Ziels eine Höchstgeschwindigkeit nicht überschritten werden darf. Mit ausreichend Abstand vor dem Ziel wird die Geschwindigkeit verringert, um sich dem definierten Zielpunkt vorsichtig zu nähern und dort zum Stillstand zu kommen. Diese Geschwindigkeitsverringern kann je nach Zugzusammenstellung variieren. Ein schwerer Güterzug wird beispielsweise früher eine Bremsung einleiten als eine einzelne Lokomotive ohne gekuppelte Wagen.

Um das Fahrmanöver automatisiert durchführen zu können, wird die sogenannte Zielbremsung verwendet. Diese gibt basierend auf dem aktuellen Abstand zum Ziel d die Geschwindigkeit v des Fahrmanövers vor. Dazu wird ein parametrierbares Geschwindigkeitsprofil verwendet. Abbildung 4.1 zeigt das im Rahmen dieser Arbeit verwendete Profil. Das Profil setzt sich aus den beiden Geschwindigkeiten $v_s = 1,5 \text{ m/s}$ und $v_l = 0,2 \text{ m/s}$,

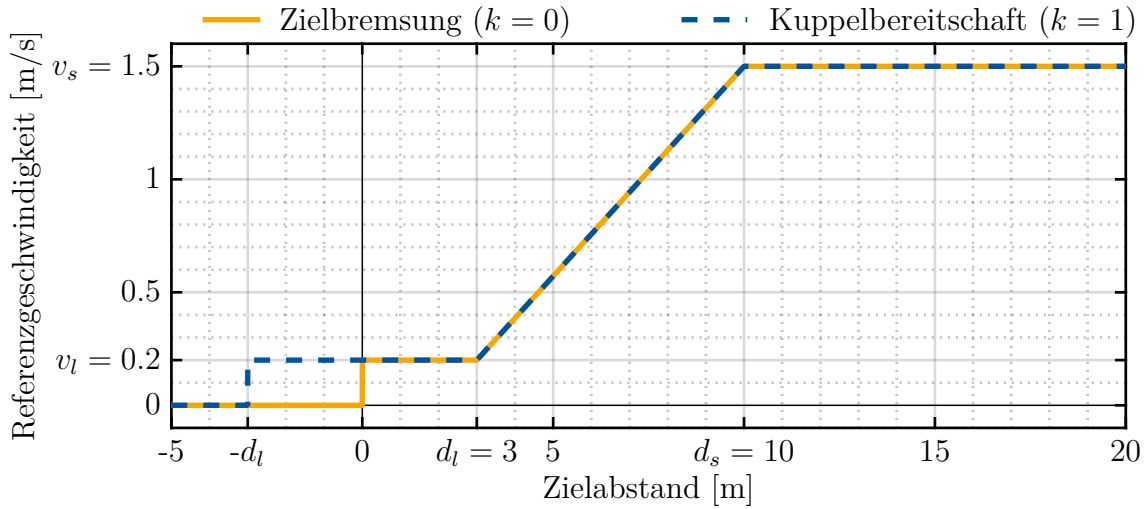


Abb. 4.1: Geschwindigkeitsprofil der Zielbremsung mit den Parametern d_s , d_l , v_s , v_l und k

den beiden Abständen $d_s = 10$ m und $d_l = 3$ m sowie dem Kuppelparameter $k \in \{0, 1\}$ zusammen. Der Index s steht dabei für schnell und l für langsam. Durch k wird angegeben ob am Ende kuppelbereit angehalten werden soll. Die resultierende Geschwindigkeit v lässt sich wie folgt berechnen:

$$v(d) = \begin{cases} v_s & \text{für } d > d_s \\ v_l + \frac{d-d_l}{d_s-d_l} \cdot (v_s - v_l) & \text{für } d_s \geq d > d_l \\ v_l & \text{für } d_l \geq d > 0 \\ k \cdot v_l & \text{für } 0 \geq d > -d_l \\ 0 & \text{sonst.} \end{cases} \quad (4.1)$$

Im Fall der Buggys wird der im Basisbord integrierte Geschwindigkeitsregler verwendet, um die vorgegebene Geschwindigkeit v einzuregeln.

In (4.1) ist zu erkennen, dass für $k = 1$ auch für negative Abstände d eine Fahrgeschwindigkeit definiert ist. Dies ermöglicht das kuppelbereite Positionieren an einem Wagen, dessen Position nicht exakt bekannt ist. Es wird solange weiter gefahren, bis auf den stehenden Wagen aufgefahren wird. Dabei ist zu beachten, dass v_l so niedrig zu wählen ist, dass kein Schaden durch das Auffahren entsteht. Experimente des Autors mit realen Zügen zeigen, dass $v_l \leq 0,5$ m/s sinnvoll ist. Das Auffahren muss detektiert werden, um die Fahrt zu stoppen. Dies kann durch eine an der Lokomotive angebrachte IMU erfolgen. Sie liefert Beschleunigungsmessungen, aus denen durch Ableitung der Ruck bestimmt werden kann. Wenn dieser im Bereich $-d_l < d < d_l$ den gegebenen Schwellwert überschreitet, wird dies als Auffahren gewertet. Das Zielbremsungsmanöver wird als abgeschlossen betrachtet, was eine Geschwindigkeitsvorgabe von $v = 0$ m/s bedeutet. Im Rahmen dieser Arbeit gilt ein empirisch gefundener Schwellwert von 20 m/s³. Der Parameter k gibt also vor, ob das Ziel genau angefahren wird und bei einer Zielentfernung von $d = 0$ gestoppt wird ($k = 0$) oder ob in der Umgebung $d \in (-d_l, d_l)$ um den Zielpunkt

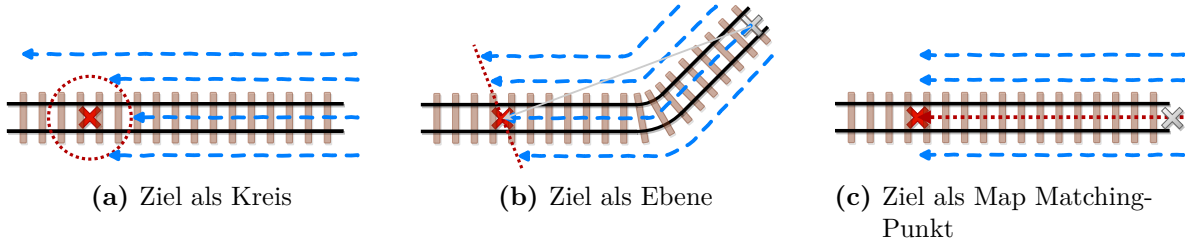


Abb. 4.2: Unterschiedliche Zieldefinitionen der Zielbremsung, mit

\times Startpunkt \mathbf{p}_s , \times Zielpunkt \mathbf{p}_z , \leftarrow Vektor von Start zu Ziel $\overline{\mathbf{p}_s \mathbf{p}_z}$,
 \leftarrow mögliche aktuelle Messwerte \mathbf{p}_a , \leftarrow Map Matching Verlauf von \mathbf{p}_a

auf einen stehenden Wagen aufgefahren wird ($k = 1$), was ebenfalls zum Stoppen der Fahrt führt.

Die Buggys sind nicht für das Auffahren aufeinander ausgelegt. Daher wird bei Verwendung der Buggys der Zielpunkt 1,5 m vor Fahrzeugbeginn definiert und intern immer $k = 0$ gesetzt. So ergibt sich ein Sicherheitsabstand von $D = 1,5$ m zwischen den Buggys und Kollisionen werden verhindert.

Der Vorteil einer IMU gegenüber anderen Sensoren liegt in der Verwendbarkeit auch mit bereits gekuppelten Wagen. Ein an der Lokomotive angebrachter Laser-basierter Abstandssensor beispielsweise würde immer nur den bereits gekuppelten Wagen detektieren. Der Ruck hingegen kann auch mit bereits gekuppeltem Wagen von einer auf der Lokomotive montierten IMU gemessen werden. Experimente des Autors zeigen, dass auch bei drei gekuppelten Wagen eine problemlose Detektion möglich ist. Größere Anzahlen von Wagen wurden nicht untersucht. Es konnte jedoch beobachtet werden, dass das Signal-zu-Rausch-Verhältnis (engl. signal-to-noise ratio) (SNR) mit steigender Anzahl bereits gekuppelter Wagen abnimmt. Ab einer gewissen Anzahl von bereits gekuppelten Wagen wird es folglich nicht mehr möglich sein, das Auffahren mit einer auf der Lokomotive montierten IMU zu detektieren.

Zur Bestimmung des Zielabstandes d sind mehrere Varianten möglich. Abbildung 4.2 zeigt drei unterschiedliche Ansätze zur Abstandsbestimmung. Der erste, dargestellt in Abbildung 4.2a, basiert darauf, dass der Abstand zwischen dem Zielpunkt \mathbf{p}_z und dem aktuellen Messpunkt \mathbf{p}_a als euklidischer Abstand bestimmt wird. Dabei wird davon ausgegangen, dass die Punkt \mathbf{p}_z und \mathbf{p}_a mit einem GNSS-Empfänger eingemessen werden und die Messwerte damit im WGS-84-Koordinatensystem vorliegen. Daher werden sie zunächst in das ECEF-Koordinatensystem transformiert. Dort kann der Anstand d einfach als

$$\begin{aligned}
 d_o &= \|\overline{\mathbf{p}_a \mathbf{p}_z}\| - r_{ZB} = \|\mathbf{p}_z - \mathbf{p}_a\| - r_{ZB} = \left\| \begin{bmatrix} x_z \\ y_z \\ z_z \end{bmatrix} - \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} \right\| - r_{ZB} \\
 &= \sqrt{(x_z - x_a)^2 + (y_z - y_a)^2 + (z_z - z_a)^2} - r_{ZB}
 \end{aligned} \tag{4.2}$$

berechnet werden. Dabei gibt r_{ZB} den Radius an, der ein Zielgebiet um den Zielpunkt \mathbf{p}_z definiert, abgebildet durch den roten Kreis in Abbildung 4.2a. Die Fahrt wird beendet, sobald \mathbf{p}_a innerhalb des Kreises liegt. Bei dieser Methode ist es möglich, bei ungenauer Positionsbestimmung von \mathbf{p}_a (blaue Pfeile), die beispielsweise einen konstanten Versatz besitzen, an dem Zielgebiet vorbeizufahren und so nie anzuhalten.

Mit dem in Abbildung 4.2b vorgestellten Ansatz, der auch in [44] beschrieben ist, kann sichergestellt werden, dass in diesen Fällen ebenfalls angehalten wird - selbst dann, wenn es einen Versatz der Messwerte von \mathbf{p}_a gibt. Auch diese Variante arbeitet basierend auf dem ECEF-Koordinatensystem. Soll die Fahrt begonnen werden wird die aktuell gemessene Position \mathbf{p}_a der Lokomotive als Startpunkt \mathbf{p}_s gesetzt. Aus \mathbf{p}_z und \mathbf{p}_s wird der ECEF-Normalenvektor \mathbf{n} gemäß

$$\mathbf{n} = \frac{\overline{\mathbf{p}_s \mathbf{p}_z}}{\|\overline{\mathbf{p}_s \mathbf{p}_z}\|} = \frac{\begin{bmatrix} x_z \\ y_z \\ z_z \end{bmatrix} - \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix}}{\left\| \begin{bmatrix} x_z \\ y_z \\ z_z \end{bmatrix} - \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \right\|} \quad (4.3)$$

berechnet. Er definiert eine Ebene, auf der er senkrecht im Zielpunkt \mathbf{p}_z steht. Der Abstand d von der aktuellen Position \mathbf{p}_a zu dieser Ebene kann als

$$d = \langle \mathbf{p}_a, \mathbf{n} \rangle - \langle \mathbf{p}_z, \mathbf{n} \rangle \quad (4.4)$$

berechnet werden. Die Werte von \mathbf{n} und \mathbf{p}_z werden gespeichert und so kann d fortlaufend aus \mathbf{p}_a berechnet werden.

Durch die Verwendung der Ebene zur Definition des Zielabstandes ist sichergestellt, dass auch bei ungenaueren Messungen das Ziel erreicht wird. Vergleiche hierzu Abbildung 4.2b mit Abbildung 4.2a. Im Fall der Ebene würde auch eine Trajektorie mit Versatz das Ziel erreichen. Es ergibt sich aber das Problem, dass die Lage der Ebene allein auf den Punkten \mathbf{p}_s (graues Kreuz) und \mathbf{p}_z (rotes Kreuz) basiert. Der Vektor \mathbf{n} (grauer Pfeil) zwischen den beiden Punkten steht senkrecht auf der Ebene und legt so ihre Orientierung fest. Eine Berücksichtigung des Gleisnetzes findet wie in Abbildung 4.2b dargestellt nicht statt. Außerdem kann je nach vorliegender Topologie des Gleisnetzes der Extremfall auftreten, dass die Ziel-Ebene entlang der Gleise verläuft.

Um diese Problematik zu umgehen, wird im Rahmen dieser Arbeit die Abstandsbestimmung weiter verbessert. Wie in [8] vorgestellt, kann basierend auf Map Matching-Positionen der kartenbasierte Abstand zwischen zwei Fahrzeugen berechnet werden. Mit demselben Verfahren kann auch der Abstand zwischen der aktuellen Position \mathbf{p}_a und dem Zielpunkt \mathbf{p}_z berechnet werden. Dazu werden die Punkte \mathbf{p}_a und \mathbf{p}_z durch ein Map Matching in Gleiskoordinaten umgerechnet (dargestellt durch den roten Pfeil in Abbildung 4.2c). Auf Basis einer zugehörigen Karte kann dann der Karten-basierte Abstand von der aktuellen Map Matching-Position zur Map Matching-Position des Zielpunktes \mathbf{p}_z berechnet werden. Werden die in Kapitel 2.2.4 vorgestellten Gleiskoordinaten verwendet

und sind beide Punkte auf dem selben Streckenabschnitt, kann der Abstand d zum Ziel durch Subtraktion der Weglängen s_a und s_z beider Map Matching-Positionen bestimmt werden. Es gilt also

$$d_{MM} = s_z - s_a \quad (4.5)$$

für die Abbildung 4.2c. Dieser Abstand d_{MM} berücksichtigt durch die Berechnung in Gleiskoordinaten auch die Topologie des Gleisnetzes. Im Rahmen dieser Arbeit wird diese Variante zur Abstandsbestimmung verwendet - sowohl für die Zielbremsung als auch bei der Kolonnenregelung, die im folgenden Abschnitt vorgestellt wird.

Die Ausgabe der Zielbremsung ist die Sollgeschwindigkeit $v(d)$ aus (4.1). Wie diese vom Fahrzeug umgesetzt wird, kann variieren. Bei Verwendung der Buggys wird die Geschwindigkeit v an den im Basisboard implementierten Regler vorgegeben. Dieser regelt die vorgegebene Geschwindigkeit ein. Bei dem in Abschnitt 5.2 vorgestellten Testfahrzeug hingegen gibt es keinen Geschwindigkeitsregler. Dem Fahrzeug kann eine Geschwindigkeit vorgegeben werden, aber diese ist nur vorgesteuert.

4.2 Kolonnenregelung

Die Kolonnenregelung setzt sich aus einer Vorsteuerung und einem darauf aufgesetzten Regler zusammen. Dieser Regler wird nach den Methoden zur Auslegung eines LQR entworfen, vgl. Abschnitt 2.6.3. Das Gesamtkonzept der Kolonnenregelung wird an dieser Stelle, basierend auf [50] und [46], vorgestellt. In den angegebenen Quellen kann eine detaillierte Diskussion des Reglers gefunden werden. Im Rahmen dieser Arbeit, wie auch in [46], findet er Anwendung, um die virtuelle Kupplung zu realisieren. Bei der hier verwendeten Variante gibt es ein Führungsfahrzeug (Lokomotive) und N Folgefahrzeuge (Wagen). Zunächst wird hierfür ein Modell hergeleitet.

Wie auch in [25, 31] und [34] wird ideales Beschleunigungsverhalten angenommen, das als Systemeingang vorgegeben werden kann. Als kinematisches Modell folgt

$$\ddot{x}_i = u_i \quad \forall i \in \{1, 2, \dots, N\} \quad (4.6)$$

für die Folgefahrzeuge. Dabei bezeichnet x_i die kartenbasierte Position des i -ten Folgefahrzeugs, vgl. Abbildung 4.3. Das Fahrzeug, das als Lokomotive agiert, bekommt im Folgenden den Index 0. Seine Position wird dementsprechend als x_0 bezeichnet. Die dem Fahrzeug zugehörige Beschleunigung \ddot{x}_0 ist kein durch die Regelung vorgegebener Eingang. Sie ist vielmehr die Sollwertvorgabe für die Kolonnenregelung. Alle Fahrzeuge weisen im Idealfall die selbe Beschleunigung wie das Führungsfahrzeug auf. Sie wird fortan auch als a_0 bezeichnet und kann durch die Fernsteuerung des Führungsfahrzeuges oder beispielsweise aus der Geschwindigkeitsanforderung der Zielbremsung definiert werden.

Aus den Fahrzeugpositionen können Abstandsfehler zwischen den Buggys berechnet werden. Dazu wird zunächst ein Sollabstand D definiert. Im Rahmen dieser Arbeit gilt

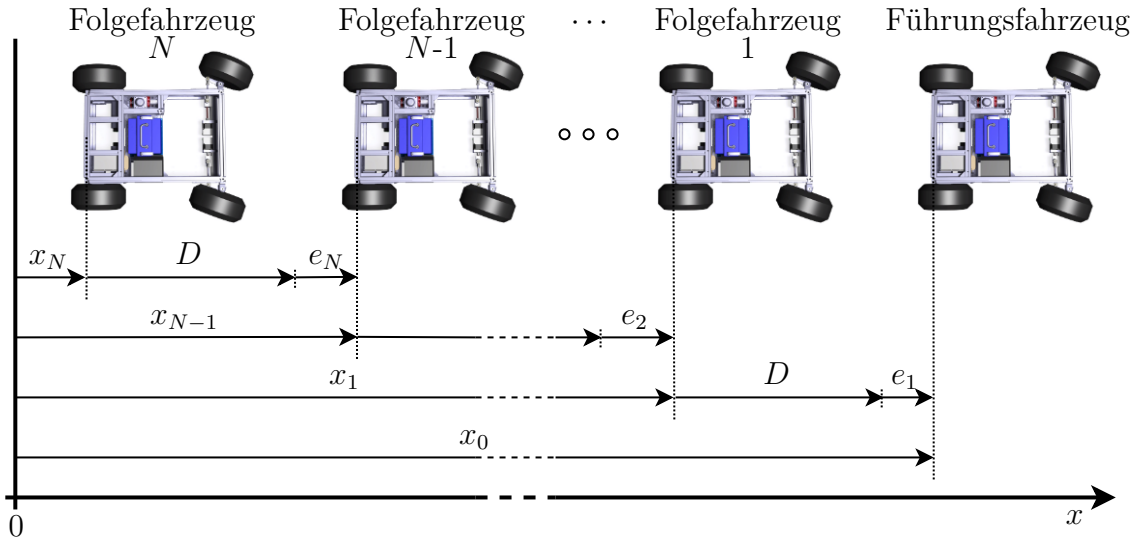


Abb. 4.3: Bestimmung der Fehlerzustände e_i einer Kolonne nach [50]

$D = 1,5\text{ m}$. Der Abstandsfehler e ergibt sich aus der Differenz der aktuellen Position und der Sollposition. Die Sollposition des vorherigen Fahrzeugs wiederum ergibt sich aus der Summe der eigenen Position und dem Sollabstand D . Es gilt also, wie in Abbildung 4.3 abgebildet,

$$e_i = x_{i-1} - (x_i + D) = x_{i-1} - x_i - D \quad \forall i \in \{1, 2, \dots, N\}. \quad (4.7)$$

Durch die Bildung der ersten beiden zeitlichen Ableitungen

$$\dot{e}_i = \dot{x}_{i-1} - \dot{x}_i \quad (4.8)$$

$$\ddot{e}_i = \ddot{x}_{i-1} - \ddot{x}_i \quad (4.9)$$

und unter Verwendung von (4.6) lässt sich der Fehler in Bezug zu den Eingängen u_i setzen. Es ergibt sich

$$\ddot{e}_i = u_{i-1} - u_i \quad \forall i \in \{1, 2, \dots, N\} \quad (4.10)$$

mit $u_0 = 0$, da das Führungsfahrzeug unabhängig ist und die Regelung keinen Einfluss

auf es nehmen kann. Daraus kann der Fehlerzustandsraum gemäß

$$\begin{aligned}
 \begin{bmatrix} \dot{e}_1 \\ \ddot{e}_1 \\ \dot{e}_2 \\ \ddot{e}_2 \\ \vdots \\ \dot{e}_N \\ \ddot{e}_N \end{bmatrix} &= \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \end{bmatrix}}_{=:A} \cdot \begin{bmatrix} e_1 \\ \dot{e}_1 \\ e_2 \\ \dot{e}_2 \\ \vdots \\ e_N \\ \dot{e}_N \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ -1 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 1 & -1 \end{bmatrix}}_{=:B} \cdot \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_N \end{bmatrix} \\
 \begin{bmatrix} y_{P_1} \\ y_{V_1} \\ y_{P_2} \\ y_{V_2} \\ \vdots \\ y_{P_N} \\ y_{V_N} \end{bmatrix} &= \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}}_{=:C} \cdot \begin{bmatrix} e_1 \\ \dot{e}_1 \\ e_2 \\ \dot{e}_2 \\ \vdots \\ e_N \\ \dot{e}_N \end{bmatrix}
 \end{aligned} \tag{4.11}$$

aufgestellt werden. Die Indizes P und V an den Ausgangskomponenten stehen jeweils für den Positions- bzw. Geschwindigkeitsfehler der einzelnen Fahrzeuge.

Mit dem Modell (4.11) und dem im Abschnitt 2.6.3 vorgestellten Verfahren zur Auslegung eines LQR kann gemäß den Gleichungen (2.28) und (2.29) die Rückführungsmatrix \mathbf{K} berechnet werden. Die Wichtungsmatrizen werden als $\mathbf{Q} = 25 \cdot \mathbf{I}$ und $\mathbf{R} = \mathbf{I}$ festgelegt, wobei \mathbf{I} die Einheitsmatrix ist. Durch diese Wahl liegt der Regler eher auf der dynamischen als der konservativen Seite, vgl. Abschnitt 2.6.3. Bei dem Verfahren gilt es zu beachten, dass für jeden Wert von N eine eigene Rückführungsmatrix \mathbf{K}_N mit N Zeilen und $2 \cdot N$ Spalten berechnet werden muss. Damit folgt

$$\mathbf{K}_N = \begin{bmatrix} K_{P_{1,1}} & K_{V_{1,1}} & K_{P_{1,2}} & K_{V_{1,2}} & \dots & K_{P_{1,N}} & K_{V_{1,N}} \\ K_{P_{2,1}} & K_{V_{2,1}} & K_{P_{2,2}} & K_{V_{2,2}} & \dots & K_{P_{2,N}} & K_{V_{2,N}} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ K_{P_{N,1}} & K_{V_{N,1}} & K_{P_{N,2}} & K_{V_{N,2}} & \dots & K_{P_{N,N}} & K_{V_{N,N}} \end{bmatrix}, \tag{4.12}$$

wobei die Berechnung a priori erfolgen kann. Zeilenweise notiert ergibt sich für den Kolonnenregler das Stellgesetz

$$\begin{aligned}
 u_1 &= -K_{P_{1,1}} \cdot e_1 - K_{V_{1,1}} \cdot \dot{e}_1 - K_{P_{1,2}} \cdot e_2 - K_{V_{1,2}} \cdot \dot{e}_2 - \dots - K_{P_{1,N}} \cdot e_N - K_{V_{1,N}} \cdot \dot{e}_N \\
 u_2 &= -K_{P_{2,1}} \cdot e_1 - K_{V_{2,1}} \cdot \dot{e}_1 - K_{P_{2,2}} \cdot e_2 - K_{V_{2,2}} \cdot \dot{e}_2 - \dots - K_{P_{2,N}} \cdot e_N - K_{V_{2,N}} \cdot \dot{e}_N \\
 &\vdots \\
 u_N &= -K_{P_{N,1}} \cdot e_1 - K_{V_{N,1}} \cdot \dot{e}_1 - K_{P_{N,2}} \cdot e_2 - K_{V_{N,2}} \cdot \dot{e}_2 - \dots - K_{P_{N,N}} \cdot e_N - K_{V_{N,N}} \cdot \dot{e}_N
 \end{aligned} \tag{4.13}$$

Jedes Folgefahrzeug setzt entsprechend seiner Position in der Kolonne die ihm zugeordnete Zeile des Stellgesetzes um.

Der Regler wurde in diesem Abschnitt für den kontinuierlichen Fall hergeleitet. In der Realität wird er aber auf Echtzeithardware ausgeführt, die zeitdiskret arbeitet. Daher wird das Modell (4.11) vor der Berechnung der Rückführungsmatrizen $\mathbf{K}_i \forall i \in \{1, 2, \dots, N\}$ entsprechend den Umrechnungsvorschriften (2.21) diskretisiert.

Wie eingangs beschrieben, gibt es neben dem LQR auch noch eine Vorsteuerung. Diese ist so realisiert, dass die gemessene Gesamtbeschleunigung a_0 des Führungsfahrzeugs von allen Folgefahrzeugen mit der Beschleunigungskomponente des LQR superponiert wird. Für die einzelnen Fahrzeuge ergibt sich also die Gesamtbeschleunigung gemäß

$$a_i = a_0 + u_i \quad \forall i \in \{1, 2, \dots, N\}. \quad (4.14)$$

Die Vorsteuerung sorgt dafür, dass die Fahrzeuge gleichmäßig beschleunigen, und der LQR gleicht dabei entstehende Positions- und Geschwindigkeitsfehler aus.

Damit jedes Fahrzeug seine Beschleunigung a_i berechnen kann, benötigt es neben der entsprechenden Zeile von \mathbf{K}_N die gemessenen Positionen x_i und Geschwindigkeiten \dot{x}_i aller Kolonnenfahrzeuge inkl. der des Führungsfahrzeugs sowie dessen gemessene Beschleunigung a_0 . Daher ist eine sichere und schnelle Kommunikation der Buggys untereinander erforderlich. Durch Verwendung des GO!-Empfängers kann diese Kommunikation auch mit bis zu acht Buggys gewährleistet werden. Wie in Abschnitt 5.1.1 gezeigt wird, bietet die LTE-Verbindung des GO!-Empfängers die benötigte Datenverbindung.

4.3 Flachheitsbasierte Folgeregelung

In diesem Abschnitt wird die im Rahmen dieser Arbeit entwickelte Flachheitsbasierte Folgeregelung (FBFR) vorgestellt. Die FBFR kann verwendet werden, um die virtuelle Schiene zu realisieren. Dabei werden die in Abschnitt 2.5 vorgestellten Kartendaten als Pfadvorgabe dienen. Die Kartendaten sind nicht direkt als Eingang für die FBFR geeignet und müssen in ein geeignetes Format überführt werden. Dies geschieht durch einen im Rahmen der Arbeit entwickelten Trajektoriengenerator. Hierzu wird die Map Matching-Position zusammen mit den Kartendaten genutzt, um daraus eine Parameterdarstellung einer der drei geometrischen Figuren Gerade, Kreis oder Klothoide abzuleiten. Diese drei geometrischen Figuren sind auch im Straßen- und Trassenbau gebräuchlich. Mit ihnen lassen sich Gleise abbilden. Die Parameterdarstellung der Figuren gemeinsam mit der Soll-Geschwindigkeit bzw. Beschleunigung dienen der FBFR als Eingänge.

Zunächst werden in Abschnitt 4.3.1 die benötigten Parameterdarstellungen von Gerade, Kreis und Klothoide hergeleitet. Sie werden zuerst vorgestellt, da die drei Geometrien über eine Eigenschaft verfügen, die bei der späteren Reglerauslegung ausgenutzt werden wird. Anschließend wird in Abschnitt 4.3.2, basierend auf den Überlegungen aus [43], ein Modell für den Buggy aufgestellt. Mit dem Modell wird gezeigt, dass es sich bei den Buggys um flache Systeme handelt. In Abschnitt 4.3.3 wird eine mögliche flache Vorsteuerung für die Buggys vorgestellt. Diese weist einige Probleme auf, weshalb sie zu einer Regelung erweitert wird. Der Regelung können eine Soll-Geschwindigkeit und eine

Soll-Trajektorie vorgegeben werden. In Abschnitt 4.3.4 ist aufgeführt, wie der Trajektoriengenerator diese Soll-Trajektorie aus den Kartendaten und der Map Matching-Position herleitet.

4.3.1 Geometrische Figuren

Wege setzen sich im OSM-Format aus der Abfolge von Punkten zusammen. Zur Beschreibung von Pfadabschnitten können Analogien zu geometrischen Figuren, wie z. B. Gerade, Kreis oder Klothoide, verwendet werden. Hierfür wird zunächst eine mathematische Beschreibung der Geometrien definiert. Dabei wird darauf geachtet, dass es sich um eine zweidimensionale Parameterdarstellung handelt. In diesem Abschnitt werden die beiden Dimensionen als x und y bezeichnet. Außerdem soll die Parameterdarstellung so gewählt sein, dass der verwendete Laufparameter l angibt, welche Weglänge entlang der Figur zurückgelegt wird. Er soll also dem Bogenmaß entlang der zurückgelegten Kurve entsprechen. Die Parameterdarstellung legt einen Startpunkt und Formcharakteristika der entsprechenden Figur fest. Diese Charakteristika sind zum Beispiel die Orientierung einer Geraden oder der Radius eines Kreises. Über den Laufparameter l kann angegeben werden, welche Bogenlänge entlang der Figur zurückgelegt werden soll, um den gewünschten Punkt zu erreichen. Diese Bedingung erleichtert später in den Abschnitten 4.3.3 und 4.3.4 die Handhabung der geometrischen Figuren zur Regelung bzw. Trajektoriengenerierung. Für die Regelung werden neben der Parameterdarstellung auch deren Ortsableitungen bis hin zur zweiten Ableitung benötigt. Diese werden durch partielle Ableitung der Parameterdarstellung nach l bestimmt.

Ein in dem Zusammenhang mit den geometrischen Figuren interessantes Maß ist das Krümmungsmaß, im Folgenden kurz als Krümmung κ bezeichnet. Sie gibt den Kehrwert des Radius an, der in der aktuell betrachteten Pfadposition vorherrscht. Die Krümmung kann für zweidimensionale Pfade in Parameterdarstellung allgemein mit der Formel

$$\kappa(l) = \frac{x'(l)y''(l) - x''(l)y'(l)}{(x'(l)^2 + y'(l)^2)^{\frac{3}{2}}} \quad (4.15)$$

berechnet werden [3]. Sie ist abhängig von den ersten Ableitungen x' und y' sowie den zweiten Ableitungen x'' und y'' der Koordinaten x und y . Im Folgenden werden die Figuren Gerade, Kreis und Klothoide vorgestellt.

4.3.1.1 Gerade

Eine Gerade kann durch zwei Punkte bestimmt werden. Der erste Punkt \mathbf{p}_A wird als Aufpunkt bezeichnet und gibt an, wo die Gerade beginnt. Die Richtung wird durch den

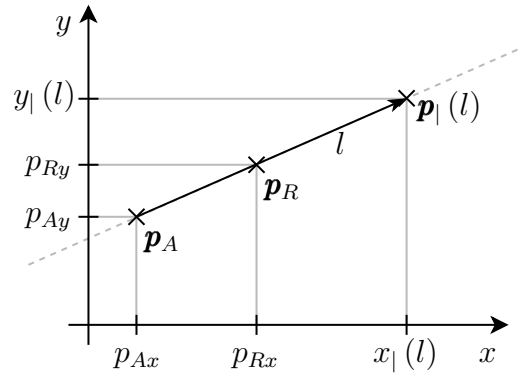


Abb. 4.4: Konstruktion einer Geraden mittels Aufpunkt \mathbf{p}_A , Richtungspunkt \mathbf{p}_R und Laufparameter l

zweiten Punkt, den Richtungspunkt \mathbf{p}_R vorgegeben. In Parameterdarstellung gilt

$$\mathbf{p}_l(l) = \begin{bmatrix} x_l(l) \\ y_l(l) \end{bmatrix} = \mathbf{p}_A + l \cdot \frac{\mathbf{p}_R - \mathbf{p}_A}{\|\mathbf{p}_R - \mathbf{p}_A\|} = \begin{bmatrix} p_{Ax} + l \cdot \frac{p_{Rx} - p_{Ax}}{\sqrt{(p_{Rx} - p_{Ax})^2 + (p_{Ry} - p_{Ay})^2}} \\ p_{Ay} + l \cdot \frac{p_{Ry} - p_{Ay}}{\sqrt{(p_{Rx} - p_{Ax})^2 + (p_{Ry} - p_{Ay})^2}} \end{bmatrix}. \quad (4.16)$$

Durch die Normierung in (4.16) wird dafür gesorgt, dass der Parameter l der Entfernung zwischen Aufpunkt \mathbf{p}_A und dem betrachteten Punkt auf der Geraden \mathbf{p}_l entspricht. Als Ableitungen von (4.16) nach l ergeben sich die ersten Ableitungen

$$x'_l(l) = \frac{p_{Rx} - p_{Ax}}{\sqrt{(p_{Rx} - p_{Ax})^2 + (p_{Ry} - p_{Ay})^2}} \quad (4.17a)$$

$$y'_l(l) = \frac{p_{Ry} - p_{Ay}}{\sqrt{(p_{Rx} - p_{Ax})^2 + (p_{Ry} - p_{Ay})^2}} \quad (4.17b)$$

und die zweiten Ableitungen

$$x''_l(l) = 0 \quad (4.18a)$$

$$y''_l(l) = 0. \quad (4.18b)$$

Wird die Gleichung (4.15) betrachtet, wird mit den zweiten Ableitungen gleich null sofort klar, dass die Krümmung einer Geraden immer null ist. Dies entspricht auch der Erwartung bei der Annahme, dass Geraden Kreise mit unendlichem Radius bzw. mit Krümmung $\kappa = 0$ sind.

4.3.1.2 Kreis

Ein Kreis kann mit der Kreisgleichung

$$(x_o - x_0)^2 + (y_o - y_0)^2 = r^2 \quad (4.19)$$

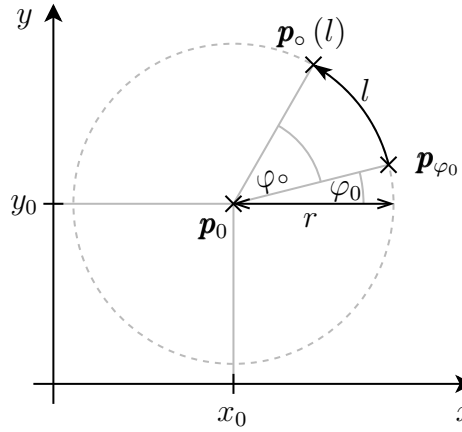


Abb. 4.5: Kreisbogen der Länge l um Mittelpunkt p_0

beschrieben werden. Der Mittelpunkt p_0 des beschriebenen Kreises liegt an den Koordinaten $(x_0, y_0)^T$ und der Kreis besitzt den Radius r . [35]

Alternativ kann ein Kreis auch durch drei beliebige verschiedene Punkte auf dem Kreis beschrieben werden. Aus den drei Kreispunkten p_1 , p_2 und p_3 können der Mittelpunkt p_0 und der Radius r bestimmt werden. Hierzu wird im Folgenden das in [47] vorgestellte Vorgehen beschrieben.

Um aus den drei Punkten p_1 , p_2 und p_3 die gesuchten Kreisparameter x_0 , y_0 und r zu bestimmen, wird die Gleichung (4.19) so aufgelöst, dass

$$\underbrace{2x_0 x_o}_{=:a} + \underbrace{2y_0 y_o}_{=:b} + \underbrace{r^2 - (x_0^2 + y_0^2)}_{=:c} = x_o^2 + y_o^2 \quad (4.20)$$

folgt. Werden die Koordinaten der drei Punkte $p_1 = [x_1, y_1]^T$, $p_2 = [x_2, y_2]^T$ und $p_3 = [x_3, y_3]^T$ in die Gleichung (4.20) eingesetzt, ergibt sich das lineare Gleichungssystem

$$\begin{aligned} x_1 \cdot a + y_1 \cdot b + c &= x_1^2 + y_1^2 \\ x_2 \cdot a + y_2 \cdot b + c &= x_2^2 + y_2^2 \\ x_3 \cdot a + y_3 \cdot b + c &= x_3^2 + y_3^2 \end{aligned} \quad (4.21)$$

mit den drei Unbekannten a , b und c . Die Lösung dieses Gleichungssystems liefert den Kreismittelpunkt p_0 und den Radius r als

$$p_0 = \left[\frac{a}{2}, \frac{b}{2} \right]^T \quad (4.22)$$

$$r = \sqrt{\left(\frac{a}{2} \right)^2 + \left(\frac{b}{2} \right)^2 + c}. \quad (4.23)$$

Um die Parameterdarstellung eines Kreises aufzustellen, sei der Punkt p_{φ_0} der Schnittpunkt mit dem Kreis, der erreicht wird, wenn der Kreismittelpunkt p_0 entlang einer

Geraden in der Richtung des Winkels φ_0 verlassen wird. Dabei wird der Winkel φ_0 bezogen auf die x-Achse gemessen, vgl. Abbildung 4.5. Der Punkt \mathbf{p}_{φ_0} ist der Startpunkt der Bewegung entlang des Kreises und besitzt die Koordinaten

$$\mathbf{p}_{\varphi_0} = \begin{bmatrix} x_{\varphi_0} \\ y_{\varphi_0} \end{bmatrix} = \mathbf{p}_0 + r \cdot \begin{bmatrix} \cos(\varphi_0) \\ \sin(\varphi_0) \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + r \cdot \begin{bmatrix} \cos(\varphi_0) \\ \sin(\varphi_0) \end{bmatrix}. \quad (4.24)$$

Durch die Bewegung wird der Punkt \mathbf{p}_o erreicht. Bei der Bewegung wird die Strecke

$$l = r \cdot \varphi_o \quad (4.25)$$

zurückgelegt. Der Winkel φ_o gibt den Winkel zwischen den Punkten \mathbf{p}_{φ_0} und \mathbf{p}_o , im Kreismittelpunkt \mathbf{p}_0 in Radian an. Die Koordinaten des Punkts \mathbf{p}_o können folglich mittels der Parameterdarstellung

$$\mathbf{p}_o(l) = \begin{bmatrix} x_o(l) \\ y_o(l) \end{bmatrix} = \mathbf{p}_0 + r \cdot \begin{bmatrix} \cos(\varphi_0 + \varphi_o) \\ \sin(\varphi_0 + \varphi_o) \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + r \cdot \begin{bmatrix} \cos\left(\varphi_0 + \frac{l}{r}\right) \\ \sin\left(\varphi_0 + \frac{l}{r}\right) \end{bmatrix} \quad (4.26)$$

berechnet werden. Zur Definition der Kreisbahn werden also Kreismittelpunkt \mathbf{p}_0 , Kreisradius r und Winkel φ_0 benötigt.

Alternativ kann der Kreis durch die Punkte \mathbf{p}_0 und \mathbf{p}_{φ_0} definiert werden, da Kreisradius r und Winkel φ_0 daraus berechnet werden können. Zur Ermittlung des Winkels φ_0 kann die arctan2-Funktion verwendet werden. Die in der Programmierung verbreitete Variation der Arkustangens-Funktion sei im Rahmen dieser Arbeit definiert als

$$\arctan2(x, y) := \begin{cases} \arctan\left(\frac{y}{x}\right) & \text{für } x > 0 \\ +\frac{\pi}{2} & \text{für } x = 0 \wedge y > 0 \\ 0 & \text{für } x = 0 \wedge y = 0 \\ -\frac{\pi}{2} & \text{für } x = 0 \wedge y < 0 \\ \arctan\left(\frac{y}{x}\right) + \pi & \text{für } x < 0 \wedge y > 0 \\ +\pi & \text{für } x < 0 \wedge y = 0 \\ \arctan\left(\frac{y}{x}\right) - \pi & \text{für } x < 0 \wedge y < 0. \end{cases} \quad (4.27)$$

Damit gilt

$$\varphi_0 = \arctan2(x_{\varphi_0} - x_0, y_{\varphi_0} - y_0). \quad (4.28)$$

Der Kreisradius r ergibt sich aus dem euklidischen Abstand der beiden Punkte als

$$r = \sqrt{(x_{\varphi_0} - x_0)^2 + (y_{\varphi_0} - y_0)^2}. \quad (4.29)$$

Die ersten und zweiten Ableitungen der Parameterdarstellung lassen sich bestimmen zu:

$$x_o'(l) = -\sin\left(\varphi_0 + \frac{l}{r}\right) \quad (4.30a)$$

$$y_o'(l) = \cos\left(\varphi_0 + \frac{l}{r}\right) \quad (4.30b)$$

und

$$x_o''(l) = -\frac{\cos\left(\varphi_0 + \frac{l}{r}\right)}{r} \quad (4.31a)$$

$$y_o''(l) = -\frac{\sin\left(\varphi_0 + \frac{l}{r}\right)}{r}. \quad (4.31b)$$

Werden die Ableitungen in Gleichung (4.15) eingesetzt, ergibt sich die Krümmung

$$\kappa(l) = \frac{\sin^2\left(\varphi_0 + \frac{l}{r}\right) + \cos^2\left(\varphi_0 + \frac{l}{r}\right)}{r\left(\sin^2\left(\varphi_0 + \frac{l}{r}\right) + \cos^2\left(\varphi_0 + \frac{l}{r}\right)\right)^{\frac{3}{2}}} = \frac{1}{r}. \quad (4.32)$$

Sie ist konstant und entspricht definitionsgemäß dem Kehrwert des Kreisradius r .

4.3.1.3 Klothoiden

Klothoiden sind geometrische Figuren, bei denen sich die Krümmung linear über die zurückgelegte Strecke ändert. Sie können als Übergang zwischen zwei Kreisen mit unterschiedlichen Radien bzw. zwischen einer Geraden und einem Kreis verwendet werden. Zur Berechnung der Positionen entlang einer Klothoide werden die Fresnelschen Integrale

$$S_\lambda(s) = \int_0^s \sin(\lambda\tau^2) d\tau \quad (4.33a)$$

$$C_\lambda(s) = \int_0^s \cos(\lambda\tau^2) d\tau \quad (4.33b)$$

mit $\lambda = \frac{\pi}{2}$ genutzt [26]. Zur Lösung dieser Integrale gibt es verschiedene numerische Ansätze [26]. Im Rahmen dieser Arbeit wird die in [17] vorgestellte analytische Approximation verwendet. Mit den Integralen $S_{\frac{\pi}{2}}(s)$ und $C_{\frac{\pi}{2}}(s)$ lässt sich die Parameterdarstellung einer Klothoide als

$$\mathbf{p}_\sim(s) = \begin{bmatrix} x_\sim(s) \\ y_\sim(s) \end{bmatrix} = A\sqrt{\pi} \begin{bmatrix} C_{\frac{\pi}{2}}(s) \\ S_{\frac{\pi}{2}}(s) \end{bmatrix} = A\sqrt{\pi} \int_0^s \begin{bmatrix} \cos\left(\frac{\pi}{2}\tau^2\right) \\ \sin\left(\frac{\pi}{2}\tau^2\right) \end{bmatrix} d\tau \quad (4.34)$$

finden. In (4.34) bezeichnet A einen Parameter der Klothoide, auf den später detaillierter eingegangen wird. Im Rahmen dieser Arbeit wird A immer als konstant angenommen. Außerdem ist zu beachten, dass die Laufvariable s verwendet wird. Diese entspricht entgegen der Forderung in diesem Kapitel nicht der zurückgelegten Strecke entlang der Klothoide. Für die zurückgelegte Strecke l entlang einer Klothoide gilt

$$l = A\sqrt{\pi} \cdot s. \quad (4.35)$$

Durch das Umstellen von (4.35) nach s kann (4.34) so umgeformt werden, dass der Laufparameter l verwendet wird. Dieser entspricht der entlang der Figur zurückgelegten Strecke. Es folgt

$$\mathbf{p}_{\sim}(l) = \begin{bmatrix} x_{\sim}(l) \\ y_{\sim}(l) \end{bmatrix} = A\sqrt{\pi} \int_0^{\frac{l}{A\sqrt{\pi}}} \begin{bmatrix} \cos\left(\frac{\pi}{2}\tau^2\right) \\ \sin\left(\frac{\pi}{2}\tau^2\right) \end{bmatrix} d\tau = \int_0^l \begin{bmatrix} \cos\left(\frac{\tau^2}{2A^2}\right) \\ \sin\left(\frac{\tau^2}{2A^2}\right) \end{bmatrix} d\tau. \quad (4.36)$$

Die Ableitungen dieser Klothoidengleichung lassen sich bestimmen zu:

$$x_{\sim}'(l) = \cos\left(\frac{l^2}{2A^2}\right) \quad (4.37a)$$

$$y_{\sim}'(l) = \sin\left(\frac{l^2}{2A^2}\right) \quad (4.37b)$$

und

$$x_{\sim}''(l) = -\frac{l}{A^2} \sin\left(\frac{l^2}{2A^2}\right) \quad (4.38a)$$

$$y_{\sim}''(l) = \frac{l}{A^2} \cos\left(\frac{l^2}{2A^2}\right) \quad (4.38b)$$

Werden die Gleichungen (4.37a), (4.37b), (4.38a) sowie (4.38b) in Gleichung (4.15) eingesetzt, ergibt sich die Krümmung

$$\kappa(l) = \frac{\frac{l}{A^2} \left(\cos^2\left(\frac{l^2}{2A^2}\right) + \sin^2\left(\frac{l^2}{2A^2}\right) \right)}{\left(\cos^2\left(\frac{l^2}{2A^2}\right) + \sin^2\left(\frac{l^2}{2A^2}\right) \right)^{\frac{3}{2}}} = \frac{l}{A^2}. \quad (4.39)$$

Damit ist die Krümmung κ proportional zur Strecke l . Dies stellt eine wesentliche Eigenschaft von Klothoiden dar. Der Klothoidenparameter A bestimmt folglich die Krümmungsänderung der Klothoide, die konstant ist, wenn A wie hier konstant ist. Neben A gibt es weitere Eigenschaften bzw. charakteristische Werte von Klothoiden. Dazu wird die Annahme getroffen, dass $l \in [0, L]$ gilt, wobei hier $l = L$ dem Parameter für den Endpunkt der betrachteten Klothoide entspricht. Damit können die folgenden Bezeichnungen eingeführt werden:

A Parameter der Klothoide, über den sich die Krümmungsänderung definiert,

L Gesamtlänge der Klothoide im Endpunkt ($l = L$),

R Krümmungsradius am Endpunkt ($l = L$) der Klothoide,

T Schnittwinkel der Tangenten im Anfangs- ($l = 0$) und Endpunkt ($l = L$),

D Abrückmaß der Klothoide, das angibt, wie weit von einem Kreis mit dem Radius R abgerückt wird, um in eine Gerade überzugehen.

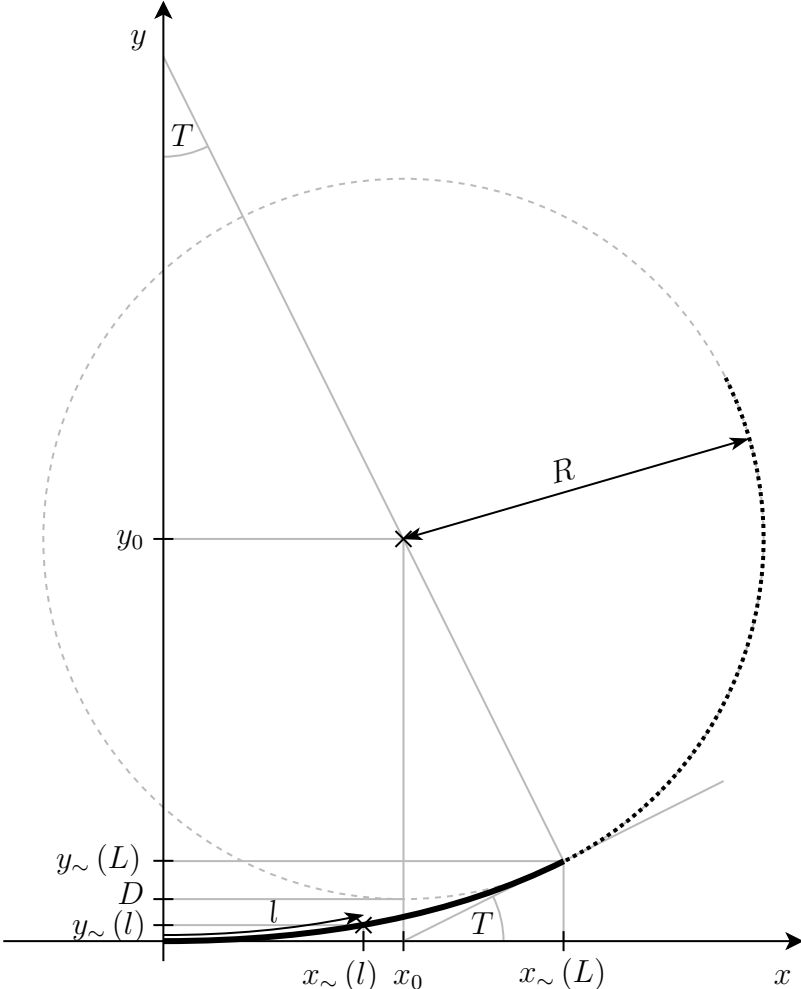


Abb. 4.6: Klothoide als Übergang von einer Geraden in einen Kreis mit dem Radius R

In Abbildung 4.6 ist der Übergang von einer Geraden in einen Kreis mit dem Radius R abgebildet. Die aufgeführten Parameter können in der Abbildung gefunden werden. Es gelten die folgenden Beziehungen:

$$R = \frac{A^2}{L} \quad (4.40)$$

$$T = \frac{L}{2 \cdot R} = \frac{L^2}{2 \cdot A^2} = \frac{A^2}{2 \cdot R^2} \quad (4.41)$$

$$D = y_{\sim}(L) - (R - R \cdot \cos(T)) \approx \frac{L^2}{24 \cdot R} \quad (4.42)$$

[10, 20, 27]

Klothoiden finden unter anderem neben Kreisen und Geraden als Entwurfselement Verwendung im Straßenbau. Dabei werden verschiedene Namen für die unterschiedlichen

Elementfolgen verwendet. Im Rahmen dieser Arbeit kommen davon Scheitel-Klothoide und Verbundkurve zum Einsatz. Nach [52] werden diese Bezeichnungen wie folgt eingesetzt:

- Scheitel-Klothoide** Eine Scheitel-Klothoide wird verwendet, um von einer Geraden in eine andere überzugehen. Dabei wird die erste Gerade mittels einer Klothoide verlassen und im Scheitelpunkt wird auf eine zweite Klothoide mit gleichem A gewechselt, mit der in die zweite Gerade übergegangen wird. Die zweite Klothoide baut die Krümmung ab, die von der ersten aufgebaut wurde.
- Verbundkurve** Ähnlich wie eine Scheitel-Klothoide findet eine Verbundkurve Anwendung, um von einer Geraden in eine weitere Gerade überzugehen. Anders als zuvor wird die Krümmung, die durch die erste Klothoide aufgebaut wird, zunächst beibehalten und dann von der zweiten Klothoide wieder abgebaut. Es entsteht folglich die Elementfolge:
Gerade - Klothoide - Kreis - Klothoide - Gerade.

4.3.2 Modellierung

Nachdem nun die geometrischen Figuren vorgestellt worden sind, wird nun das dynamische Modell des Buggys zur Reglerauslegung vorgestellt. Abbildung 4.7 zeigt den Buggy in einem lokalen NED-Koordinatensystem mit seinen wichtigsten Größen zur Herleitung des kinematischen Modells. Das im Folgenden in Anlehnung an [43] hergeleitete Modell unterscheidet sich von dem der Pfadregelung aus Abschnitt 2.9, auch wenn hier ebenfalls der Mittelpunkt der Fahrzeughinterachse als Bezugspunkt verwendet wird. Dieser kann in zweidimensionalen lokalen kartesischen NED-Koordinaten durch den Punkt \mathbf{p} mit

$$\mathbf{p} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (4.43)$$

dargestellt werden, vgl. Abbildung 4.7. Neben der Position wird dem Fahrzeug ein Winkel θ als Orientierung zugeordnet. Die Orientierung θ beschreibt die Drehung der Fahrzeug-Längsachse im mathematisch positiven Sinn bezogen auf die e -Richtung. Als Eingänge des Modells werden der mittlere Lenkwinkel φ und die Fahrzeuggeschwindigkeit v gewählt. Da die Lenkung der Buggys nach Ackermann ausgelegt ist [45], kann der Lenkwinkel φ wie bei einem Einspurmodell betrachtet werden.

Unter der Annahme, dass kein Gleiten stattfindet, lassen sich die folgenden Modellgleichungen entsprechend den Bezeichnungen aus Abbildung 4.7 aufstellen:

$$\dot{x}_1 = v \cos \theta \quad (4.44a)$$

$$\dot{x}_2 = v \sin \theta \quad (4.44b)$$

$$\dot{\theta} = v \tan \frac{\varphi}{L} \quad (4.44c)$$

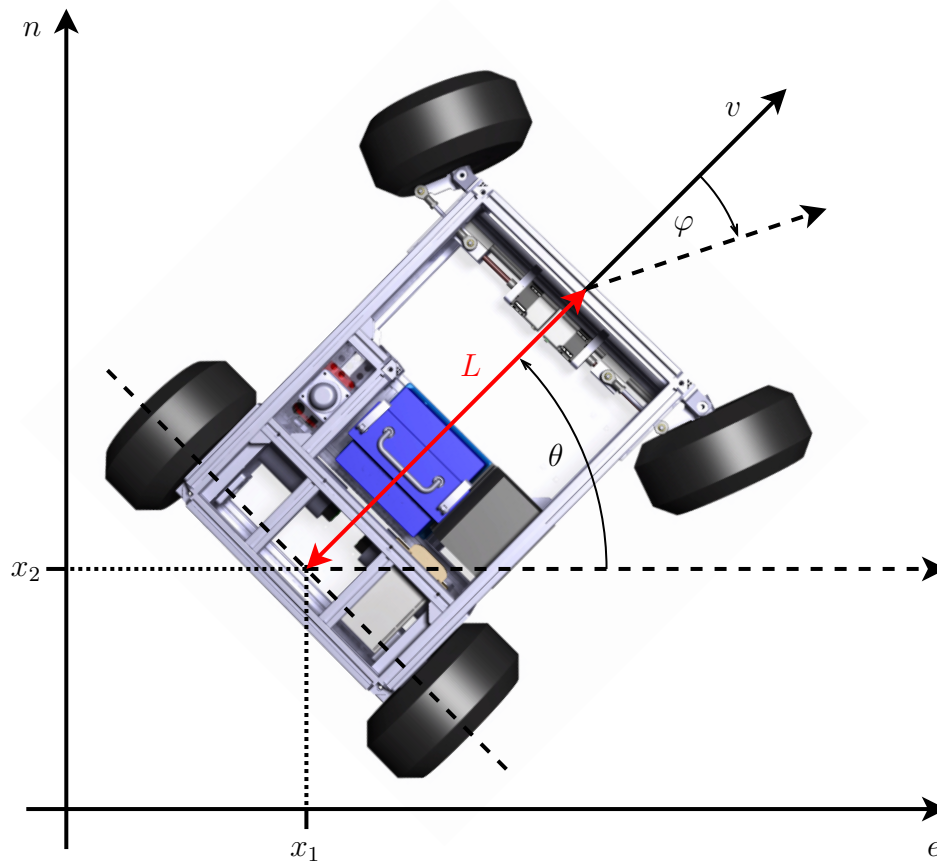


Abb. 4.7: Der Buggy als flaches System

Hierbei ist L der Radstand des Fahrzeugs. Das Fahrzeugmodell besitzt den Zustandsvektor

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \theta \end{bmatrix}. \quad (4.45)$$

Als Ausgang \mathbf{y} wird der Hinterachsmittelpunkt definiert, sodass

$$\mathbf{y} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (4.46)$$

gilt. Der Eingang \mathbf{u} setzt sich aus der Geschwindigkeit v und dem Lenkwinkel φ zusammen:

$$\mathbf{u} = \begin{bmatrix} v \\ \varphi \end{bmatrix}. \quad (4.47)$$

Damit ist das Modell vollständig beschrieben. Es erfüllt die Voraussetzungen der Flachheitsdefinition aus Abschnitt 2.6.2 womit der Buggy ein flaches System ist, was es im Folgenden zu beweisen gilt:

Die Definition aus Abschnitt 2.6.2 besagt, dass der gewählte Ausgang \mathbf{y} aus differentiell unabhängigen Komponenten bestehen und $\dim \mathbf{y} = \dim \mathbf{u}$ gelten muss. Dies ist für das

vorgestellte Modell erfüllt. Zusätzlich ist die Bedingung (2.23) bereits durch die Wahl von \mathbf{y} erfüllt, da der Ausgang \mathbf{y} nur aus Komponenten des Zustandes \mathbf{x} besteht. Um nun zu zeigen, dass \mathbf{y} ein flacher Ausgang gemäß der Flachheitsdefinition ist, müssen sich \mathbf{x} und \mathbf{u} als Funktion ausschließlich in Abhängigkeit von \mathbf{y} und Ableitungen von \mathbf{y} beschreiben lassen, vgl. Bedingungen (2.24) und (2.25). Dazu wird zunächst durch Division von (4.44a) und (4.44b) die Fahrzeugorientierung θ in Abhängigkeit des Zustandes \mathbf{x} als

$$\theta = \arctan\left(\frac{\dot{x}_2}{\dot{x}_1}\right) \quad (4.48)$$

definiert. Anschließend kann aus der Summe der Quadrate von (4.44a) und (4.44b) die Fahrzeuggeschwindigkeit

$$v = \pm\sqrt{\dot{x}_1^2 + \dot{x}_2^2} \quad (4.49)$$

bestimmt werden. Der Lenkwinkel φ in Abhängigkeit von \mathbf{x} ergibt sich durch Umstellen von (4.44c) gefolgt durch Einsetzen von (4.49) sowie der Ableitung von (4.48). Daraus folgt für den Lenkwinkel

$$\varphi = \arctan\left(\frac{L \cdot \dot{\theta}}{v}\right) = \pm \arctan\left(\frac{L \cdot (\ddot{x}_2 \dot{x}_1 - \ddot{x}_1 \dot{x}_2)}{(\dot{x}_1^2 + \dot{x}_2^2)^{\frac{3}{2}}}\right). \quad (4.50)$$

In (4.48), (4.49) und (4.50) gibt es nur Ableitungen erster und zweiter Ordnung von x_1 bzw. x_2 . Diese bilden nach Gleichung (4.46) den Ausgang \mathbf{y} . Damit sind alle Bedingungen der Flachheitsdefinition erfüllt, vgl. Abschnitt 2.6.2. Demnach ist das System flach. Außerdem ist der Ausgang \mathbf{y} ein flacher Ausgang des Systems. □

4.3.3 Regler Synthese

Durch Vorgabe von

$$\mathbf{y}_d(t) = \begin{bmatrix} x_{1d} \\ x_{2d} \end{bmatrix} \quad (4.51)$$

als eine zweifach stetig differenzierbare Funktion kann eine Trajektorie für den Hinterachsmittelpunkt des Fahrzeuges vorgegeben werden. Der Index d steht für desired (engl. für gewollt), also für die Sollwertvorgabe. Um zu bestimmen, welcher Eingang sich aus \mathbf{y}_d ergibt, kann (4.51) in (4.49) und (4.50) eingesetzt werden. Es ergeben sich

$$v_d(t) = \pm\sqrt{\dot{x}_{1d}^2(t) + \dot{x}_{2d}^2(t)} \quad (4.52a)$$

$$\varphi_d(t) = \pm \arctan\left(\frac{L \cdot (\ddot{x}_{2d}(t) \dot{x}_{1d}(t) - \ddot{x}_{1d}(t) \dot{x}_{2d}(t))}{(\dot{x}_{1d}^2(t) + \dot{x}_{2d}^2(t))^{\frac{3}{2}}}\right) \quad (4.52b)$$

als Sollwertvorgaben für die Geschwindigkeit v und den Lenkwinkel φ [43]. Die Gleichungen 4.52 realisieren eine Steuerung. Der zugehörige Systemaufbau ist in Abbildung 4.8 abgebildet.

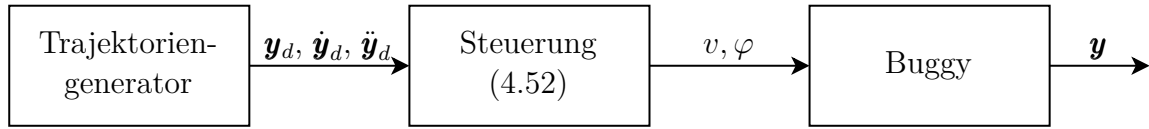


Abb. 4.8: Aufbau der Fahrzeugsteuerung

Ein noch nicht näher spezifizierter Trajektoriengenerator erzeugt den Sollausgangsverlauf \mathbf{y}_d und gibt diesen an die Steuerung (4.52). Diese erzeugt daraus die Vorgaben v_d und φ_d für die Eingänge v und φ des Buggys. Der Buggy setzt die Vorgaben um und es ergibt sich der tatsächliche Ausgang \mathbf{y} .

Dieser wird von seinem Sollwertverlauf \mathbf{y}_d abweichen, da bei der Modellierung der Fahrzeuge vereinfachende Annahmen getroffen werden. Außerdem findet keinerlei Rückführung statt, sodass weder auf die modellbedingten Abweichungen noch Störungen von außen reagiert werden kann. Um diese Modellierungsfehler bzw. nicht modellierte Störgrößen auszugleichen, wird der bisherige Ansatz im Folgenden zu einer Zwei-Freiheitsgrad-Regelung erweitert. Dazu wird die modellbasierte Vorsteuerung um einen Ausgangsfolgeregler ergänzt [18].

Die Ausgangsfolgeregelung wird in Analogie zu [43] hergeleitet. Zunächst wird der Folgefehler \mathbf{e} definiert als

$$\mathbf{e}(t) = \begin{bmatrix} x_1(t) - y_{1d}(t) \\ x_2(t) - y_{2d}(t) \end{bmatrix}. \quad (4.53)$$

In der ersten Zeile von (4.53) ist die Abweichung des Hinterachsmittelpunktes in Ost-Richtung und in der zweiten Zeile in Nord-Richtung beschrieben. Das Ziel des Reglers ist es, den Fehler \mathbf{e} zu minimieren, was gleichbedeutend mit der Forderung

$$\lim_{t \rightarrow \infty} \mathbf{e}(t) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (4.54)$$

ist. Um dieses Verhalten zu erreichen, wird gefordert, dass sich der Fehler in jeder Komponente von \mathbf{e} wie ein stabiler Oszillator verhält. Das bedeutet der Fehler kann um Null herum schwingen, konvergiert aber in endlicher Zeit gegen Null. Um diese Forderung zu erfüllen, muss die Differentialgleichung

$$\ddot{\mathbf{e}} + K_1 \cdot \dot{\mathbf{e}} + K_0 \cdot \mathbf{e} = \mathbf{0} \quad (4.55)$$

mit $K_0 > 0$ und $K_1 > 0$ erfüllt sein. Da für alle Fahrtrichtungen dasselbe Verhalten gewünscht ist, werden die Parameter K_0 und K_1 als Skalare definiert. Die Gleichung (4.55) lässt sich unter der Verwendung der Gleichungen (4.44a), (4.44b) und (4.53) auf die Form

$$\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} = \begin{bmatrix} \ddot{y}_{1d} \\ \ddot{y}_{2d} \end{bmatrix} - K_1 \begin{bmatrix} v \cos \theta - \dot{y}_{1d} \\ v \sin \theta - \dot{y}_{2d} \end{bmatrix} - K_0 \begin{bmatrix} x_1 - y_{1d} \\ x_2 - y_{2d} \end{bmatrix} \quad (4.56)$$

bringen.

Als nächstes wird betrachtet, wie die Stellgrößen berechnet werden müssen, damit auch sie dem Ansatz aus Gleichung (4.55) genügen. Durch Ableiten der Gleichungen (4.44a) und (4.44b), gefolgt von Einsetzen der Gleichung (4.44c) ergibt sich

$$\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} = \begin{bmatrix} \dot{v} \cos \theta - v \dot{\theta} \sin \theta \\ \dot{v} \sin \theta + v \dot{\theta} \cos \theta \end{bmatrix} = \underbrace{\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}}_{=:R(\theta)} \begin{bmatrix} \dot{v} \\ \frac{v^2}{L} \tan \varphi \end{bmatrix}. \quad (4.57)$$

Für die verwendete Rotationsmatrix $R(\theta)$ gilt $R^{-1}(\theta) = R(-\theta) = R^T(\theta)$. Durch Gleichsetzen von (4.56) mit (4.57) und Umformen ergibt sich

$$\begin{bmatrix} \dot{v} \\ \frac{v^2}{L} \tan \varphi \end{bmatrix} = R^{-1}(\theta) \left(\begin{bmatrix} \ddot{y}_{1d} \\ \ddot{y}_{2d} \end{bmatrix} - K_1 \begin{bmatrix} v \cos \theta - \dot{y}_{1d} \\ v \sin \theta - \dot{y}_{2d} \end{bmatrix} - K_0 \begin{bmatrix} x_1 - y_{1d} \\ x_2 - y_{2d} \end{bmatrix} \right). \quad (4.58)$$

Die erste Zeile dieser Gleichung stellt eine DGL in v dar. Sind die Zustandswerte x_1 , x_2 sowie θ bekannt, beispielsweise durch Messung, so kann v aus der DGL bestimmt werden. Wird die Gleichung (4.58) zeilenweise betrachtet, umgestellt und λ zur Vereinfachung der Schreibweise eingeführt, so ergibt sich der Regelkreis als

$$\dot{v} = \begin{bmatrix} \cos \theta & \sin \theta \end{bmatrix} \lambda(v, \mathbf{x}, \theta, \mathbf{y}_d, \dot{\mathbf{y}}_d, \ddot{\mathbf{y}}_d) \quad (4.59a)$$

$$v(0) = \sqrt{\dot{y}_{1d}^2(0) + \dot{y}_{2d}^2(0)} \quad (4.59b)$$

$$\varphi = \arctan \left(\frac{L}{v^2} \begin{bmatrix} -\sin \theta & \cos \theta \end{bmatrix} \lambda(v, \mathbf{x}, \theta, \mathbf{y}_d, \dot{\mathbf{y}}_d, \ddot{\mathbf{y}}_d) \right) \quad (4.59c)$$

$$\lambda = \begin{bmatrix} \ddot{y}_{1d} \\ \ddot{y}_{2d} \end{bmatrix} - K_1 \begin{bmatrix} v \cos \theta - \dot{y}_{1d} \\ v \sin \theta - \dot{y}_{2d} \end{bmatrix} - K_0 \begin{bmatrix} x_1 - y_{1d} \\ x_2 - y_{2d} \end{bmatrix} \quad (4.59d)$$

[43]. Er kann, wie in Abbildung 4.9 dargestellt, aufgebaut werden.

In der Mitte der Abbildung 4.9 ist der Regler abgebildet. Dieser bekommt weiterhin die Solltrajektorie \mathbf{y}_d und deren Ableitungen $\dot{\mathbf{y}}_d$ und $\ddot{\mathbf{y}}_d$ von einem Trajektoriengenerator vorgegeben, vgl. Abbildung 4.9. Außerdem kann der Anfangswert für $v(t=0)$ mittels (4.59b) aus $\dot{\mathbf{y}}_d$ vom Trajektoriengenerator ermittelt und dem Regler zur Verfügung gestellt werden. Die Trajektorie bildet den ersten Freiheitsgrad der Zwei-Freiheitsgrade-Regelung. Den zweiten Freiheitsgrad bildet der Fehler \mathbf{e} zwischen \mathbf{y}_d und \mathbf{y} , der in Gleichung (4.59d) enthalten ist. Dies ist in Abbildung 4.9 durch den linken Kasten innerhalb des Reglers dargestellt. Bei der Bestimmung von λ werden nicht die tatsächlichen Zustände des Fahrzeugs x_1 , x_2 und θ verwendet. Diese sind nicht direkt zugänglich und werden daher durch einen nicht linearen Beobachter mittels Sensorfusion geschätzt. Daher gehen die geschätzten Zustände \hat{x}_1 , \hat{x}_2 und $\hat{\theta}$ in (4.59d), vgl. Abbildung 4.9. Aus λ werden die Signale \dot{v} gemäß (4.59a) und φ gemäß (4.59c) bestimmt. Auch hierbei ist in Abbildung 4.9 dargestellt, dass für die Komponenten der Rotationsmatrix $R(\theta)$ der Schätzwert $\hat{\theta}$ verwendet wird. Da die Eingangssignale v und φ für den Buggy bestimmt werden sollen, muss \dot{v} noch integriert werden. Auch dies geschieht innerhalb des rechten Blocks im Regler in Abbildung 4.9. Die ermittelte Geschwindigkeit v wird zum einen gemeinsam mit φ an den Buggy gegeben, woraus sich \mathbf{y} ergibt. Zum anderen wird sie bei der Berechnung von λ berücksichtigt.

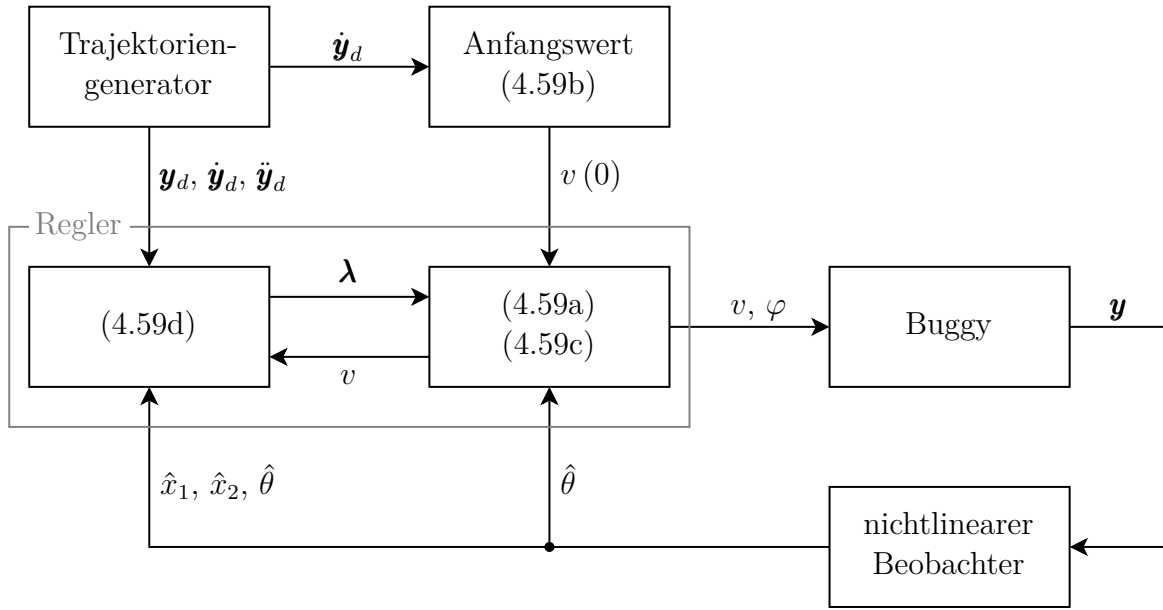


Abb. 4.9: Der Regelkreis zur Folgeregelung des Fahrzeugs nach [43]

Bei den bisher durchgeführten Betrachtungen wurden nicht alle Sonderfälle eingehend beachtet. Im Sonderfall des Stillstandes gibt es ein Problem, das im Folgenden herausgearbeitet und anschließend umgangen wird. Dazu wird das verwendete Modell (4.44) mit den Beziehungen (4.49) und (4.50) erneut betrachtet. Damit lassen sich die Systemgrößen wie folgt darstellen [43]:

$$\cos \theta = \frac{\dot{x}_1}{\sqrt{\dot{x}_1^2 + \dot{x}_2^2}} \quad (4.60a)$$

$$\sin \theta = \frac{\dot{x}_2}{\sqrt{\dot{x}_1^2 + \dot{x}_2^2}} \quad (4.60b)$$

$$v = \sqrt{\dot{x}_1^2 + \dot{x}_2^2} \quad (4.60c)$$

$$\tan \varphi = \frac{L \cdot (\ddot{x}_2 \dot{x}_1 - \ddot{x}_1 \dot{x}_2)}{(\dot{x}_1^2 + \dot{x}_2^2)^{\frac{3}{2}}}. \quad (4.60d)$$

Es ist zu erkennen, dass für $v = 0$ eine Singularität vorliegt, also Definitionslücken in den Gleichungen (4.60a), (4.60b) und (4.60d) existieren. Die Gleichungen besitzen zwar links- und rechtsseitige Grenzwerte, diese sind aber nicht identisch, was die Funktionen undefiniert für $v = 0$ macht. Dies führt dazu, dass die bisher erarbeitete Steuerung und der Regler (4.59) so nicht für den Stillstand geeignet sind.

Um dieses Problem zu lösen, wird von einer Parametrierung durch die Zeit t zu einer Parametrierung über die Bogenlänge l übergegangen. Für eine Variable z besteht mit der Kettenregel der Differentiation der folgende Zusammenhang zwischen der Zeitableitung \dot{z} und der Ortsableitung z' bzgl. der in Abschnitt 4.3.1 eingeführten Bogenlänge l :

$$\dot{z} = \frac{dz}{dt} = \frac{dz}{dl} \frac{dl}{dt} = z' \dot{l} \quad (4.61)$$

Mit dieser Substitution wird aus (4.60)

$$\cos \theta = \frac{\dot{x}_1'}{\sqrt{(\dot{x}_1')^2 + (\dot{x}_2')^2}} = \frac{x_1'}{\sqrt{x_1'^2 + x_2'^2}} \quad (4.62a)$$

$$\sin \theta = \frac{\dot{x}_2'}{\sqrt{(\dot{x}_1')^2 + (\dot{x}_2')^2}} = \frac{x_2'}{\sqrt{x_1'^2 + x_2'^2}} \quad (4.62b)$$

$$v = \sqrt{(\dot{x}_1')^2 + (\dot{x}_2')^2} = \dot{l} \cdot \sqrt{x_1'^2 + x_2'^2} \quad (4.62c)$$

$$\tan \varphi = L \cdot \frac{\dot{x}_1' (\ddot{x}_2' + \dot{l}^2 x_2'') - \dot{x}_2' (\ddot{x}_1' + \dot{l}^2 x_1'')}{\left((\dot{x}_1')^2 + (\dot{x}_2')^2 \right)^{\frac{3}{2}}} = L \cdot \frac{x_1' x_2'' - x_2' x_1''}{\left(x_1'^2 + x_2'^2 \right)^{\frac{3}{2}}} \quad (4.62d)$$

[43]. Es bleibt also lediglich in der Geschwindigkeitskomponente v eine Zeitabhängigkeit in Form des Terms \dot{l} erhalten. Alle anderen Komponenten enthalten ausschließlich Ortsableitungen. Für die Lenkung ist folglich nur wichtig, an welcher Stelle sich das Fahrzeug befindet. Daraus ergibt sich der zu wählende Lenkwinkel φ . Bei genauerer Betrachtung von (4.62d) zeigt sich, dass der hintere Teil der in Gleichung (4.15) definierten Krümmung κ entspricht, also

$$\tan \varphi = L \cdot \kappa(l) \quad (4.63)$$

gilt.

Zur Erweiterung des Ansatzes aus [43] wird die Gleichung (4.62c) weiter untersucht. Sie setzt sich multiplikativ aus einer zeit- und einer ortsabhängigen Komponente zusammen. Die Ortskomponente wird im Folgenden als

$$v_{geo}(l) = \sqrt{x_1'^2 + x_2'^2} \quad (4.64)$$

bezeichnet und die zeitabhängige Komponente als

$$v_{ref}(t) = \dot{l}. \quad (4.65)$$

Damit folgt aus Gleichung (4.62c) für die Geschwindigkeit

$$v = v_{ref}(t) \cdot v_{geo}(l). \quad (4.66)$$

Die Terme werden im Folgenden getrennt voneinander betrachtet.

Zunächst wird der ortsabhängige Geschwindigkeitsanteil v_{geo} betrachtet. Für die in Abschnitt 4.3.1 vorgestellten Figuren (Gerade, Kreis und Klothoide) lässt sich dieser Geschwindigkeitsanteil explizit berechnen. Mit den Gleichungen (4.17a) und (4.17b), (4.30a) und (4.30b) sowie (4.37a) und (4.37b) ergeben sich die folgenden Zusammenhänge für eine Gerade

$$v_{geo}(l) = \sqrt{\frac{(p_{Rx} - p_{Ax})^2}{(p_{Rx} - p_{Ax})^2 + (p_{Ry} - p_{Ay})^2} + \frac{(p_{Ry} - p_{Ay})^2}{(p_{Rx} - p_{Ax})^2 + (p_{Ry} - p_{Ay})^2}} = 1, \quad (4.67)$$

für einen Kreis

$$v_{geo}(l) = \sqrt{\sin^2\left(\varphi_0 + \frac{l}{r}\right) + \cos^2\left(\varphi_0 + \frac{l}{r}\right)} = 1 \quad (4.68)$$

und für eine Klothoide

$$v_{geo}(l) = \sqrt{\cos^2\left(\frac{l^2}{2A^2}\right) + \sin^2\left(\frac{l^2}{2A^2}\right)} = 1. \quad (4.69)$$

Der ortsabhängige Teil v_{geo} nimmt also für die betrachteten Figuren immer den Wert 1 an, womit die Geschwindigkeit nur von v_{ref} abhängt. Dies entspricht den Anforderungen der virtuellen Schiene. Die Lenkung ist ortsabhängig und es gibt einen frei wählbaren Parameter, hier v_{ref} , für die Geschwindigkeit. Unter dieser Betrachtung kann das ortsabhängige Regelgesetz wie folgt definiert werden:

$$v'_{geo}(l) = \begin{bmatrix} \cos \theta & \sin \theta \end{bmatrix} \boldsymbol{\lambda}(v_{geo}, \mathbf{x}, \theta, \mathbf{y}_d, \mathbf{y}'_d, \mathbf{y}''_d) \quad (4.70a)$$

$$v_{geo}(0) = \sqrt{y_1''^2(0) + y_2''^2(0)} = 1 \quad (4.70b)$$

$$\varphi(l) = \arctan\left(\frac{L}{v_{geo}^2} \begin{bmatrix} -\sin \theta & \cos \theta \end{bmatrix} \boldsymbol{\lambda}(v_{geo}, \mathbf{x}, \theta, \mathbf{y}_d, \mathbf{y}'_d, \mathbf{y}''_d)\right) \quad (4.70c)$$

$$\boldsymbol{\lambda}(l) = \begin{bmatrix} y_1''_d \\ y_2''_d \end{bmatrix} - K_1 \begin{bmatrix} v_{geo} \cos \theta - y_1'_d \\ v_{geo} \sin \theta - y_2'_d \end{bmatrix} - K_0 \begin{bmatrix} x_1 - y_{1d} \\ x_2 - y_{2d} \end{bmatrix}. \quad (4.70d)$$

In (4.70b) geht die Annahme ein, dass sich initial nur entlang von Geraden, Kreisen und Klothoiden bewegt wird. Für die gilt, gemäß (4.67), (4.68) sowie (4.69), $v_{geo} \equiv 1$ und somit auch $v_{geo}(0) = 1$. Durch Integration von Gleichung (4.70a) kann v_{geo} bestimmt werden. Dabei nehmen Abweichungen von den Sollwerten in Gleichung (4.70d) Einfluss auf die Integration und damit auf v_{geo} . Über v_{geo} wird der Regler folglich mittels (4.66) Einfluss auf die Geschwindigkeit v nehmen, um vorhandene Abweichungen auszuregeln.

Für den nun betrachteten zeitabhängigen Geschwindigkeitsterm v_{ref} aus Gleichung (4.66) gibt es keine besonderen Anforderungen. Gemäß (4.65) muss lediglich

$$l(t) = \int_0^t v_{ref}(\tau) d\tau \quad (4.71)$$

gelten. Damit kann dem Regler in Form von v_{ref} ein beliebiger stetiger Referenzgeschwindigkeitsverlauf vorgegeben werden.

Mit diesen Vorüberlegungen und unter Einbeziehung des Map Matching ergibt sich die in Abbildung 4.10 dargestellte Reglerstruktur. Sie ähnelt der Struktur aus Abbildung 4.9. Ein wesentlicher Unterschied besteht darin, dass es mit v_{ref} einen dedizierten frei wählbaren Eingang gibt. In Abbildung 4.9 stammen die Vorgaben allesamt vom Trajektoriengenerator. Außerdem ist in Abbildung 4.10 zu erkennen, dass die Schätzwerte des

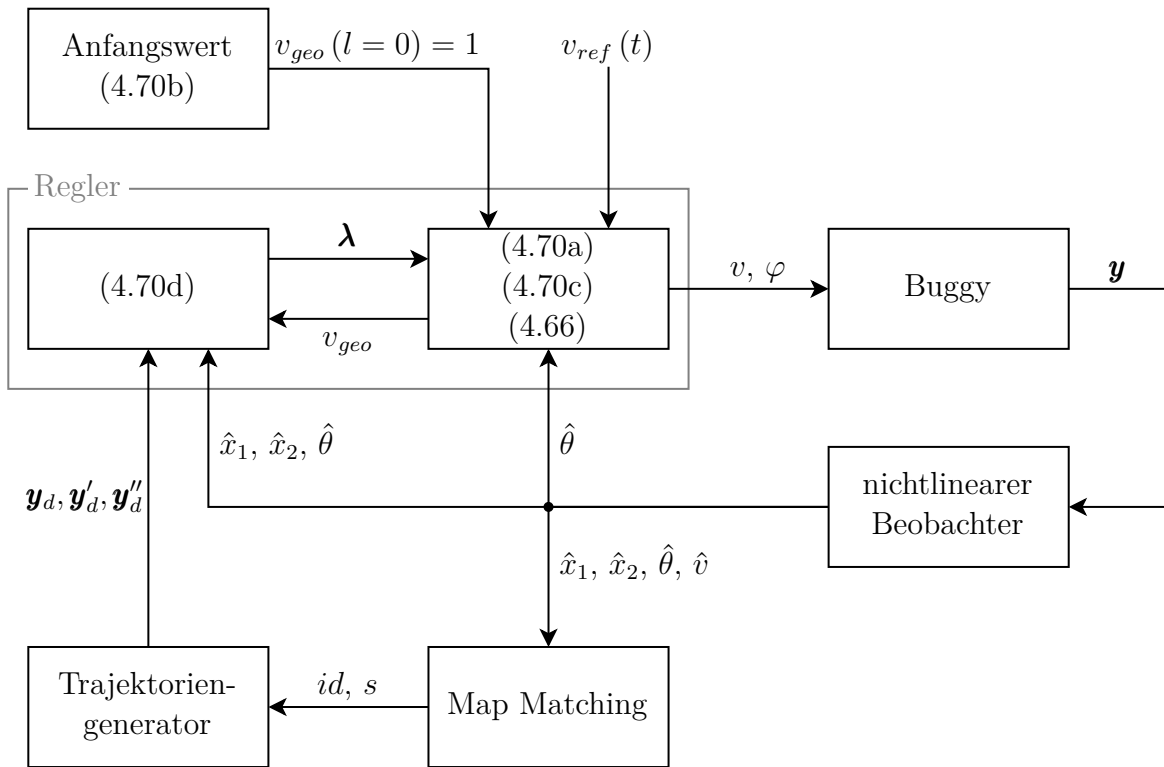


Abb. 4.10: Regelkreis zur Realisierung der virtuellen Schiene mittels Folgeregelung

Beobachters auch für das Map Matching verwendet werden und dafür auch eine Schätzung der Geschwindigkeit \hat{v} stattfindet. Die vom Map Matching bestimmte Position in Gleiskoordinaten dient dem Trajektoriengenerator als Eingang. Zusätzlich wird ihm auch das Vorzeichen von v_{ref} bereitgestellt, was der Übersicht halber nicht in Abbildung 4.10 eingezeichnet ist. Wie der Trajektoriengenerator aus diesen Informationen und den Kartendaten die Trajektorien generiert, wird im folgenden Abschnitt vorgestellt.

Als Beobachter wird ein Navigationsfilter verwendet, der sowohl Zugriff auf die Raddrehzahlen als auch die GNSS- sowie IMU-Messwerte auf dem CAN-Bus der Buggys hat. Da diese Messwerte mit unterschiedlichen Raten verfügbar sind, schätzt ein Kalman-Filter mit Strap-Down-Algorithmus die Position, Orientierung und Geschwindigkeit. In [5] und [38] können Details, zur Implementierung des Navigationsfilters gefunden werden. Die GNSS-Messung liegt mit der niedrigsten Rate von 5 Hz vor. Trotzdem stellt der Filter, dank der anderen Messwerte, eine Zustandsschätzung mit einer Rate von 100 Hz zur Verfügung. Die Rate der FBFR wird daher ebenfalls zu 100 Hz gewählt.

4.3.4 Trajektoriengenerator

Die vorgestellte FBFR benötigt als Eingang eine Trajektorie y_d entlang derer die Buggys geführt werden sollen. Diese muss entsprechend der vorgestellten Methode zweifach stetig differenzierbar sein. Die Ableitungen y'_d und y''_d werden vom Trajektoriengenerator mit

berechnet. Als Eingang dienen dieselben Kartendaten, die auch dem Map Matching zur Verfügung stehen, sowie das Vorzeichen der Soll-Geschwindigkeit v_{ref} , das angibt, ob vor- oder rückwärts gefahren werden soll. Im folgenden Beispiel wird davon ausgegangen, dass das Vorzeichen positiv ist und der Buggy damit vorwärts fahren soll. Außerdem liefert das Map Matching die Position des Buggys \mathbf{p}_{MM} in Gleiskoordinaten (id, s) . Die Position \mathbf{p}_{MM} wird dabei vom Map Matching aus der aktuell vom Beobachter geschätzten Fahrzeugposition \mathbf{p}_a gewonnen.

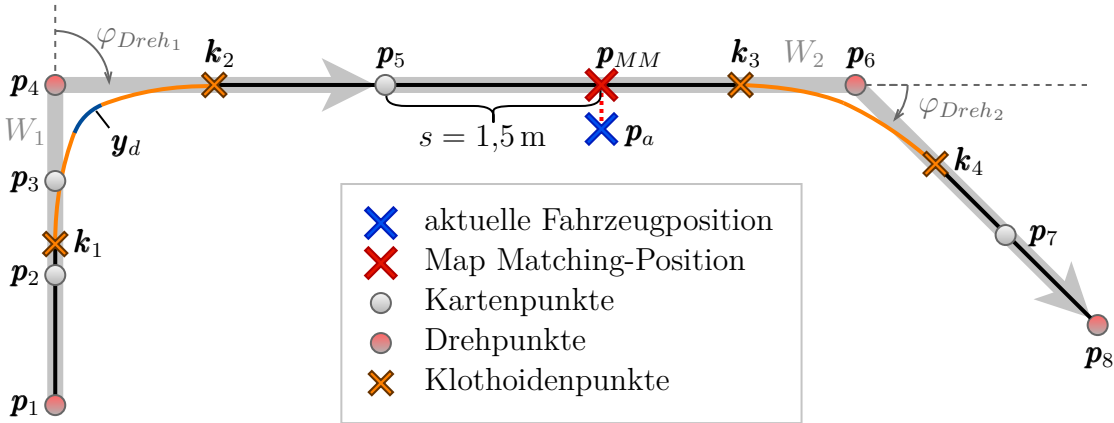


Abb. 4.11: Ermittlung der Solltrajektorie \mathbf{y}_d (dünne durchgezogene Linie; Geraden schwarz, Kreise blau, Klothoiden gelb) anhand von Kartendaten und der Map Matching-Position \mathbf{p}_{MM}

Abbildung 4.11 zeigt beispielhafte Eingangsdaten und wie daraus die Soll-Trajektorie (dünne durchgezogene Linie) generiert wird. Die Kartendaten liegen in Form der Koordinaten der Kartenpunkte \mathbf{p}_1 bis \mathbf{p}_8 (graue bzw. rote Punkte in Abbildung 4.11) sowie des Wissens, wie daraus Wege gebildet werden, vor. In dem Beispiel aus Abbildung 4.11 bildet sich der Weg W_1 durch die Abfolge der Punkte \mathbf{p}_1 , \mathbf{p}_2 , \mathbf{p}_3 , \mathbf{p}_4 und \mathbf{p}_5 , was dazu führt, dass der Weg von \mathbf{p}_1 zu \mathbf{p}_5 zeigt, angedeutet durch einen dicken grauen Pfeil. Analog bildet sich der Weg W_2 von \mathbf{p}_5 zu \mathbf{p}_8 , vgl. den zweiten grauen Pfeil in Abbildung 4.11. In der Abbildung ist zu erkennen, dass die Pfeile an den Punkten \mathbf{p}_4 und \mathbf{p}_6 geknickt sind. Dies macht diese Punkte zu sogenannten Drehpunkten, rote Punkte in Abbildung 4.11. An Drehpunkten ändert sich die Richtung eines Weges, weshalb nur an ihnen Klothoiden oder Kreise zum Einsatz kommen. Alle anderen Stellen werden durch Geraden abgebildet. Die Information, ob ein Punkt ein Drehpunkt ist, wird beim Laden der Karte bestimmt und für die FBFR im SIMULINK®-Modell hinterlegt, ebenso wie eine wählbare Höchstgeschwindigkeit v_{max} für jeden Weg. Im Rahmen dieser Arbeit gilt $v_{max} = 3 \text{ m/s}$, wenn nicht anders angegeben. Mit der eingezeichneten aktuellen Position \mathbf{p}_a ergibt sich die Map Matching-Position in Gleiskoordinaten

$$\mathbf{p}_{MM} = (2, 1,5 \text{ m}), \quad (4.72)$$

da die Position \mathbf{p}_{MM} auf Weg W_2 fällt ($id_{MM} = 2$) und 1,5 m von dessen Startpunkt \mathbf{p}_5 entfernt ist ($s_{MM} = 1,5 \text{ m}$). In dem Beispiel bewegt sich der Buggy entlang der Pfeilrichtung, weshalb der Streckenabschnitt id positiv ist. Aus den soeben beschriebenen

Eingangsdaten bildet der Trajektoriengenerator die Ausgangstrajektorie \mathbf{y}_d , die in der Abbildung 4.11 als dünne durchgezogene farbige Linie dargestellt ist. Die Farben geben an, um welche geometrische Figur es sich bei dem vorliegenden Abschnitt handelt. Geraden sind schwarz, Kreisabschnitte blau und Klothoiden gelb eingezeichnet. Zusätzlich sind vier sogenannte Klothoidenpunkte \mathbf{k}_1 bis \mathbf{k}_4 in der Abbildung eingezeichnet (gelbe Kreuze). Diese geben an, an welcher Stelle von einer Geraden in eine Klothoide übergegangen wird. Um die Klothoidenpunkte berechnen zu können, müssen zunächst die Parameter der Klothoiden bestimmt werden.

4.3.4.1 Bestimmung der Klothoidenparameter

Um die Klothoiden aus Abbildung 4.11 berechnen zu können, muss der Winkel zwischen den Geraden in ihrem Drehpunkt bestimmt werden. Aus der Map Matching-Position wird dazu mittels der Strecken-ID id bestimmt, auf welchem Streckenabschnitt sich das Fahrzeug befindet. Entsprechend dem aktuell in der Karte hinterlegten Pfad werden die nächsten zwei Drehpunkte in beiden Fahrtrichtungen aus der Karte extrahiert. In Abbildung 4.11 sind das entsprechend \mathbf{p}_4 und \mathbf{p}_1 entgegen der Fahrtrichtung sowie \mathbf{p}_6 und \mathbf{p}_8 in Fahrtrichtung. Mit diesen Punkten kann der Pfad um die Punkte \mathbf{p}_4 und \mathbf{p}_6 ermittelt werden. Dazu wird aus den Strecken $\overline{\mathbf{p}_1\mathbf{p}_4}$ und $\overline{\mathbf{p}_4\mathbf{p}_6}$ der Drehwinkel φ_{Dreh_1} im Punkt \mathbf{p}_4 bzw. aus den Strecken $\overline{\mathbf{p}_4\mathbf{p}_6}$ und $\overline{\mathbf{p}_6\mathbf{p}_9}$ der Drehwinkel φ_{Dreh_2} im Punkt \mathbf{p}_6 bestimmt, vgl. Abbildung 4.11.

Für die Berechnung der Klothoiden wird auch der Klothoidenparameter A benötigt. Wie in Abschnitt 4.3.1.3 beschrieben, ist er ein Maß für die Krümmungsänderung einer Klothoide. Die maximale Krümmungsänderung hängt sowohl von der Lenk- als auch Fahrge-
schwindigkeit ab. Dazu wird zunächst bestimmt, welche Strecke l_{Lenk} vom Fahrzeug bei einer Fahrt mit der maximalen Höchstgeschwindigkeit v_{max} in der Zeit t_{Lenk} zurückgelegt wird. Die Lenkzeit t_{Lenk} bezeichnet dabei die Zeit, die benötigt wird, um die Lenkung von der neutralen Position voll zu einer Seite einzuschlagen, wenn das Fahrzeug aufgebockt ist. Es gilt folglich

$$l_{Lenk} = v_{max} \cdot t_{Lenk}. \quad (4.73)$$

Anschließend kann A gemäß (4.40) und unter der Annahme, dass kein Gleiten stattfindet, zu

$$A = \sqrt{l_{Lenk} \cdot R_{min}} = \sqrt{v_{max} \cdot t_{Lenk} \cdot R_{min}} \quad (4.74)$$

bestimmt werden. Hierbei entspricht R_{min} dem minimal fahrbaren Radius, der wie t_{Lenk} eine Fahrzeugkonstante ist und sich aus dem bauartbedingten, maximalen Lenkwinkel φ_{max} und dem Radstand L ergibt. Auch für das in Abschnitt 4.3 aufgestellte Fahrzeugmodell für die FBFR gilt der Zusammenhang (2.37), mit dem

$$R_{min} = \frac{L}{\tan(\varphi_{max})} \quad (4.75)$$

bestimmt werden kann. Der Klothoidenparameter A ist also eine Funktion der in den Kartendaten hinterlegten Höchstgeschwindigkeit v_{max} , da die anderen Abhängigkeiten, t_{Lenk} , φ_{max} und L , Fahrzeugkonstanten sind.

4.3.4.2 Scheitel-Klothoide oder Verbundkurve

Sind φ_{Dreh} und A bestimmt, wird ermittelt, ob es ausreichend ist, eine Scheitel-Klothoide zu verwenden, um von der einen auf die andere Gerade zu gelangen. Andernfalls muss zwischen den Klothoiden eine Kreisfahrt eingefügt werden, um φ_{Dreh} mit einer Verbundkurve gänzlich abzudecken. Dazu wird zunächst der Schnittwinkel

$$T = \frac{l_{Lenk}^2}{2 \cdot A^2} = \frac{v_{max} \cdot t_{Lenk}}{2 \cdot R_{min}} \quad (4.76)$$

gemäß (4.41) ermittelt. Er gibt an, welcher Winkel mit einer Klothoiden erreicht werden kann. Am Ende der Klothoiden ist die Lenkung voll eingeschlagen und das Fahrzeug hat seinen minimalen Radius R_{min} erreicht. Da für beide Klothoiden der Scheitel-Klothoide dieselben Parameter verwendet werden, wird anschließend geprüft, ob beide Klothoiden zusammen den nötigen Winkel φ_{Dreh} erreichen können. Es wird geprüft, ob

$$\varphi_{Scheitel,max} = 2 \cdot T = \frac{v_{max} \cdot t_{Lenk}}{R_{min}} \geq \varphi_{Dreh} \quad (4.77)$$

gilt. Dabei ist $\varphi_{Scheitel,max}$ der maximal durch eine Scheitel-Klothoide abdeckbare Winkel. Ist Gleichung (4.77) erfüllt, reicht eine Scheitel-Klothoide aus, falls nicht, wird eine Verbundkurve benötigt.

Bei Gleichheit in (4.77) müssen die Klothoiden eine Länge $L = l_{Lenk}$ besitzen. Ist $\varphi_{Scheitel,max}$ größer als φ_{Dreh} muss die Länge L der Klothoiden entsprechend verkürzt werden. Wird eine Verbundkurve benötigt, kommt zu $\varphi_{Scheitel,max}$ noch additiv ein Anteil durch das zusätzliche Kreisstück hinzu. Der vom Kreisstück abzudeckende Winkel φ_{Kreis} ergibt sich als

$$\varphi_{Kreis} = \varphi_{Dreh} - \varphi_{Scheitel,max}, \quad (4.78)$$

sodass φ_{Dreh} vollständig abgedeckt wird. Um so lange wie möglich auf dem Weg zu bleiben, wird versucht, die Änderung der Fahrtrichtung so nah wie möglich am Wendepunkt durchzuführen. Dazu muss die Lenkung komplett eingeschlagen werden und der Kreisabschnitt mit dem sich aus (4.75) ergebenden Radius R_{min} durchfahren werden. Konkret liegt der maximale Lenkwinkel der Buggys bei $\varphi_{max} = 28^\circ$, er wird aber im Rahmen dieser Arbeit für den Trajektoriengenerator auf $\varphi_{max} = 25^\circ$ gesetzt. Damit ergibt sich der minimale Radius R_{min} zu gerundet 1,48 m. Die Differenz zwischen dem tatsächlichen φ_{max} und den 25° wird als Reserve für die FBFR definiert. Der Regler hat damit einen Spielraum von etwa 3° , um auch bei „voll“ eingeschlagener Lenkung Abweichungen auszugleichen.



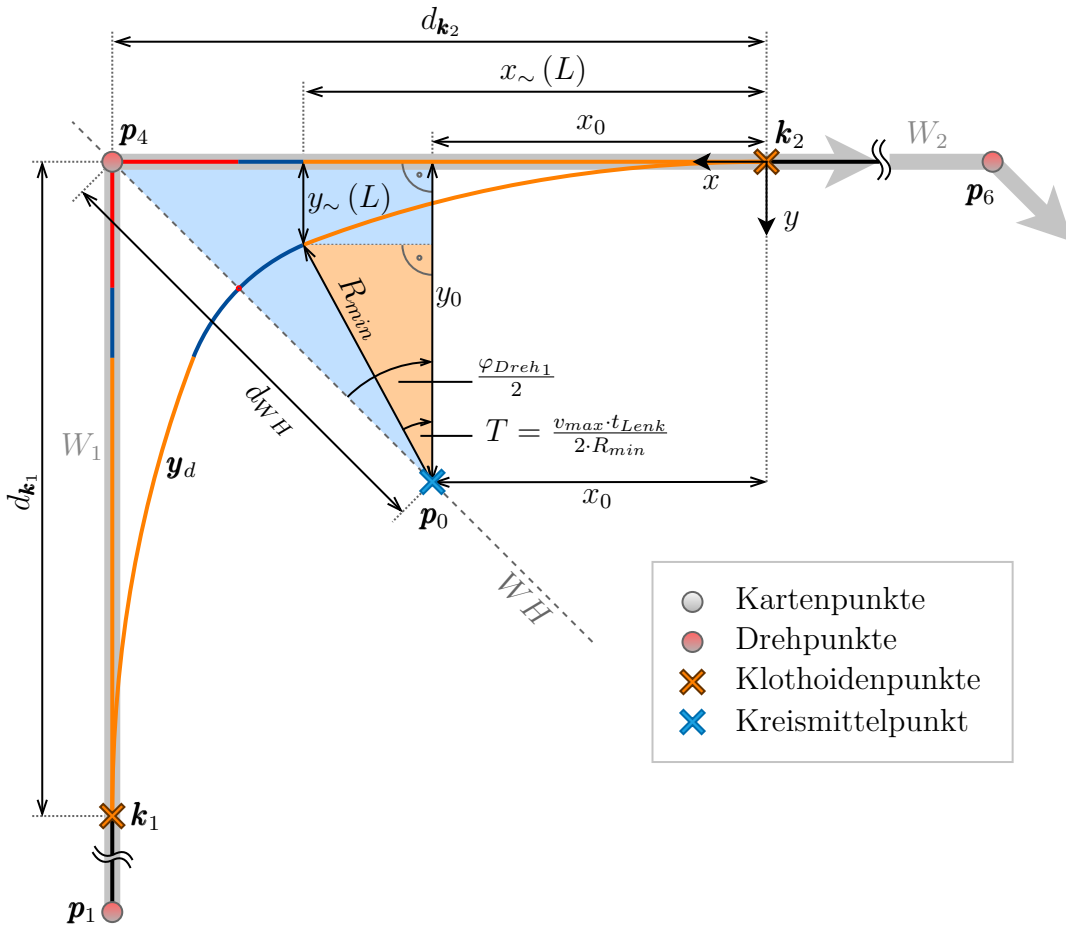


Abb. 4.13: Skizze zur Parameterbestimmung einer Verbundkurve

Der nötige Abstand d_{k_3} , den k_3 zu p_6 einnehmen muss, kann als Summe von $x_{\sim}(L)$ und d_T berechnet werden. Damit gilt:

$$d_{k_3} = x_{\sim}(L) + d_T = x_{\sim}(L) + y_{\sim}(L) \cdot \tan\left(\frac{|\varphi_{Dreh2}|}{2}\right). \quad (4.81)$$

Der Klothoidenpunkt k_3 kann also bestimmt werden, indem von p_6 die Entfernung d_{k_3} in Richtung von p_4 gegangen wird. Entsprechend der Symmetrien ist mit (4.81) auch der Abstand $d_{k_4} = d_{k_3}$ bestimmt, sodass der Punkt k_4 analog zu k_3 berechnet werden kann.

In Abbildung 4.12 sind die Wege farbig markiert, um zu zeigen, wohin die Map Matching-Positionen projiziert werden. Dabei ist zu erkennen, dass alle möglichen roten Positionen auf den Punkt zusammenfallen, an dem sich die beiden Klothoiden treffen.

4.3.4.4 Bestimmung der Klothoidenpunkte bei einer Verbundkurve

Abbildung 4.13 zeigt eine detailliertere Ansicht der Verbundkurve um p_4 aus Abbildung 4.11. Bei einer Verbundkurve ist die Bestimmung des Abstandes d_{k_2} vom Dreh-

punkt \mathbf{p}_4 zum Klothoidenpunkt \mathbf{k}_2 komplexer als bei einer Scheitel-Klothoide, da zusätzlich zu den Klothoidenparametern auch die Kreisparameter berücksichtigt werden müssen. Für den Kreisabschnitt gilt die Parameterdarstellung aus Gleichung (4.26). Für die Bestimmung des Punktes \mathbf{k}_2 muss der Kreisanteil der Verbundkurve berücksichtigt werden. Außerdem müssen die nötigen Werte für die Parameterdarstellung des Kreises (4.26) ermittelt werden, um für den Kreisabschnitt entsprechend die Sollwerte vorgeben zu können. Die Berechnungen werden, wie auch bei der Scheitel-Klothoide, in einem verschobenen und rotierten Koordinatensystem durchgeführt. Hier hat das Koordinatensystem den Ursprung \mathbf{k}_2 . Im konkreten Fall bedeute dies hier, dass der Ursprung in Abbildung 4.13 in den Punkt \mathbf{k}_2 verschoben und das Koordinatensystem um 180° gedreht ist, vgl. Achsenbeschriftung des Koordinatensystems im Punkt \mathbf{k}_2 . Zunächst werden auch hier $x_\sim(L)$ und $y_\sim(L)$ gemäß Gleichung (4.36) berechnet, wobei hier $L = l_{Lenk}$ gilt, da die Klothoide von der Geraden auf den Radius R_{min} führt. Anschließend kann der Kreismittelpunkt \mathbf{p}_0 ermittelt werden, der sich entsprechend dem orangefarbenen Dreieck in Abbildung 4.13 zu

$$\mathbf{p}_0 = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} x_\sim(L) \\ y_\sim(L) \end{bmatrix} + R_{min} \cdot \begin{bmatrix} -\sin(T) \\ \cos(T) \end{bmatrix} \quad (4.82)$$

ergibt. Von \mathbf{p}_0 ist aufgrund der Auslegung der Verbundkurve bekannt, dass er auf der Winkelhalbierenden WH , die zum Punkt \mathbf{p}_4 gehört, liegt. Damit kann \mathbf{p}_4 erreicht werden, wenn sich entlang von WH bewegt wird. Dabei muss die Entfernung d_{WH} zurückgelegt werden, die gemäß dem blauen Dreieck in der Abbildung 4.13 zu

$$d_{WH} = \frac{y_0}{\cos\left(\frac{\varphi_{Dreh1}}{2}\right)} \quad (4.83)$$

berechnet werden kann. Der Normalenvektor der Winkelhalbierenden \mathbf{n}_{WH} , der von \mathbf{p}_0 zu \mathbf{p}_4 zeigt, kann gemäß Abbildung zu

$$\mathbf{n}_{WH} = \begin{bmatrix} \sin\left(\frac{\varphi_{Dreh1}}{2}\right) \\ -\cos\left(\frac{\varphi_{Dreh1}}{2}\right) \end{bmatrix} \quad (4.84)$$

bestimmt werden. Mit ihm kann

$$\mathbf{p}_4 = \mathbf{p}_0 + d_{WH} \cdot \mathbf{n}_{WH} = \mathbf{p}_0 + y_0 \cdot \begin{bmatrix} \tan\left(\frac{\varphi_{Dreh1}}{2}\right) \\ -1 \end{bmatrix} = \begin{bmatrix} x_0 + y_0 \cdot \tan\left(\frac{\varphi_{Dreh1}}{2}\right) \\ 0 \end{bmatrix} \quad (4.85)$$

ermittelt werden. Da nun \mathbf{p}_4 bekannt ist, kann daraus auch der Abstand $d_{\mathbf{k}_2}$ zwischen \mathbf{p}_4 und \mathbf{k}_2 ermittelt werden. Er entspricht dem x -Anteil von \mathbf{p}_4 , da es keinen y -Anteil gibt. Dies liegt an der Wahl des verwendeten Koordinatensystems mit dem Punkt \mathbf{k}_2 im Ursprung und der x -Richtung hin zu \mathbf{p}_4 weisend. Mit (4.82) ergibt sich der Abstand $d_{\mathbf{k}_2}$ aus (4.85) als

$$d_{\mathbf{k}_2} = x_\sim(L) - R_{min} \cdot \sin(T) + (y_\sim(L) + R_{min} \cdot \cos(T)) \cdot \tan\left(\frac{\varphi_{Dreh1}}{2}\right). \quad (4.86)$$

Analog zu $d_{\mathbf{k}_2}$ kann $d_{\mathbf{k}_1}$ bestimmt werden. Da in dieser Arbeit ausschließlich symmetrische Verbundkurven zum Einsatz kommen, kann die Symmetrie ausgenutzt und ohne weitere Berechnungen $d_{\mathbf{k}_1} = d_{\mathbf{k}_2}$ verwendet werden. Somit sind fast alle Werte, die für die Abbildung der Map Matching-Position auf die Verbundkurve nötig sind, bekannt. Es fehlt lediglich φ_0 für den Kreisabschnitt. Da der Endpunkt der Klothoide berechnet werden kann und dieser gleichzeitig der Anfangspunkt des Kreisabschnittes ist, kann φ_0 mit $\mathbf{p}_{\varphi_0} = \mathbf{p}_{\sim}(L)$ und \mathbf{p}_0 nach Gleichung (4.28) mit der $\arctan2$ -Funktion berechnet werden. Damit sind alle Parameter bestimmt.

Auch in Abbildung 4.13 sind die Abschnitte der Verbundkurve und ihre zugehörigen Geradenabschnitte in derselben Farbe eingezeichnet. Dabei werden die rot markierten Geradenabschnitte auf einen einzelnen Punkt abgebildet, nämlich den Schnittpunkt von der Winkelhalbierenden WH mit dem Kreisstück.

4.3.4.5 Bestimmung der Trajektorie für Geraden

Zwischen den Punkten \mathbf{k}_2 und \mathbf{k}_3 handelt es sich um eine Gerade von \mathbf{p}_4 zu \mathbf{p}_6 , vgl. Abbildung 4.11. Diese Punkte können direkt in die Gleichung (4.16) für Geraden eingesetzt werden, wobei $\mathbf{p}_A = \mathbf{p}_4$ und $\mathbf{p}_R = \mathbf{p}_6$ gewählt wird, da die Fahrtrichtung der Wege W_1 und W_2 von \mathbf{p}_4 zu \mathbf{p}_6 zeigt. Es bleibt lediglich der Laufparameter l aus \mathbf{p}_{MM} zu bestimmen. Für das Beispiel hier kann l zwischen \mathbf{k}_2 und \mathbf{k}_3 als

$$l = \begin{cases} s_{W_1}(\mathbf{p}_{MM}) - s_{W_1}(\mathbf{p}_4) & \text{für } \mathbf{p}_{MM} \in W_1 \wedge s_{W_1}(\mathbf{p}_{MM}) \geq s_{W_1}(\mathbf{k}_2) \\ s_{W_2}(\mathbf{p}_{MM}) + s_{W_1}(\mathbf{p}_5) - s_{W_1}(\mathbf{p}_4) & \text{für } \mathbf{p}_{MM} \in W_2 \wedge s_{W_2}(\mathbf{p}_{MM}) \leq s_{W_2}(\mathbf{k}_3) \end{cases} \quad (4.87)$$

berechnet werden. In der Gleichung (4.87) steht $s_{W_i}(\mathbf{p})$ mit $i \in \{1, 2\}$ für die Entfernung gemessen vom Anfangspunkt des Weges W_i entlang des Weges W_i hin zum Punkt \mathbf{p} auf dem Weg W_i .

4.3.4.6 Bestimmung der Trajektorie für Kreise und Klothoiden

Neben Geraden können auch andere Pfade in den Kartendaten enthalten sein, wie beispielsweise eine Kreisfahrt. Abbildung 4.15 zeigt beispielhaft fünf Drehpunkte, mit denen ein kreisförmiger Pfad vorgegeben werden soll. In der Realität würden viel mehr Punkte für einen Halbkreis verwendet werden. Die Abbildung 4.15 soll nur prinzipiell die Verfahren skizzieren. Mit dem bisher vorgestellten Verfahren zur Verarbeitung von Kartendaten würden Scheitel-Klothoiden bzw. Verbundkurven in die den Kreis definierenden Punkte gelegt werden. Der so generierte Pfad würde das Fahrzeug dazu bringen, ständig zu lenken, statt mit einem konstanten Lenkeinschlag den Kreis zu durchfahren, vgl. gelber Verlauf in Abbildung 4.15.

Um das Problem des ständigen Lenkens zu lösen, wird in einem neuen Verfahren der Abstand zwischen zwei aufeinanderfolgenden Drehpunkten betrachtet. Ist dieser kleiner als der Buggy-Radstand $L = 0,69 \text{ m}$, wird nicht wie zuvor beschrieben versucht, mittels

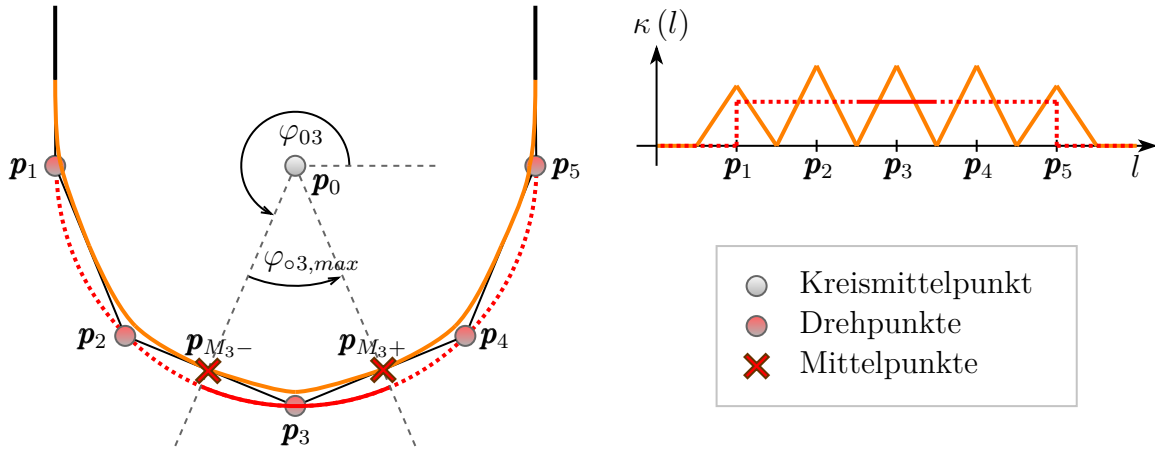


Abb. 4.14: Verläufe der Kreisfahrt; angenähert mit Klothoiden (gelb) versus Kreisabschnitte (rot gestrichelt) gemäß (4.88) - (4.92) beispielhaft für $i = 3$ (rot durchgezogen)

Klothoiden eine Trajektorie zu bestimmen, sondern es werden Kreisabschnitte in die Punkte gelegt, vgl. rot gestrichelter Verlauf in Abbildung 4.15. Hierzu werden für jeden dieser Drehpunkte \mathbf{p}_i seine beiden Nachbardrehpunkte \mathbf{p}_{i-1} und \mathbf{p}_{i+1} benötigt. Aus diesen drei Punkten können der zugehörige Kreismittelpunkt \mathbf{p}_0 und der Radius r gemäß (4.21), (4.22) und (4.23) berechnet werden. Um den Startwinkel φ_{0i} zu ermitteln, wird der Mittelpunkt \mathbf{p}_{M_i-} zwischen \mathbf{p}_i und \mathbf{p}_{i-1} als

$$\mathbf{p}_{M_i-} = \begin{bmatrix} x_{M_i-} \\ y_{M_i-} \end{bmatrix} = \frac{\mathbf{p}_{i-1} + \mathbf{p}_i}{2}, \quad (4.88)$$

bestimmt und in (4.28) eingesetzt. Es ergibt sich

$$\varphi_{0i} = \arctan2(y_{M_i-} - y_0, x_{M_i-} - x_0). \quad (4.89)$$

Analog gilt, dass der Kreisbogen nur bis zum Mittelpunkt \mathbf{p}_{M_i+} zwischen \mathbf{p}_i und \mathbf{p}_{i+1} aufgespannt wird. Mit

$$\mathbf{p}_{M_i+} = \begin{bmatrix} x_{M_i+} \\ y_{M_i+} \end{bmatrix} = \frac{\mathbf{p}_i + \mathbf{p}_{i+1}}{2} \quad (4.90)$$

gilt für den maximalen Kreisbogen Winkel

$$\varphi_{0i,max} = \arctan2(y_{M_i+} - y_0, x_{M_i+} - x_0) - \arctan2(y_{M_i-} - y_0, x_{M_i-} - x_0). \quad (4.91)$$

Zur Bestimmung des konkreten Wertes von φ_o zwischen \mathbf{p}_{M_i-} und \mathbf{p}_{M_i+} wird der Map Matching-Punkt \mathbf{p}_{MM} herangezogen, welcher der Übersicht halber nicht in Abbildung 4.15 eingezeichnet ist. Es gilt

$$\varphi_{oi} = \arctan2(y_{MM} - y_0, x_{MM} - x_0) - \arctan2(y_{M_i-} - y_0, x_{M_i-} - x_0). \quad (4.92)$$

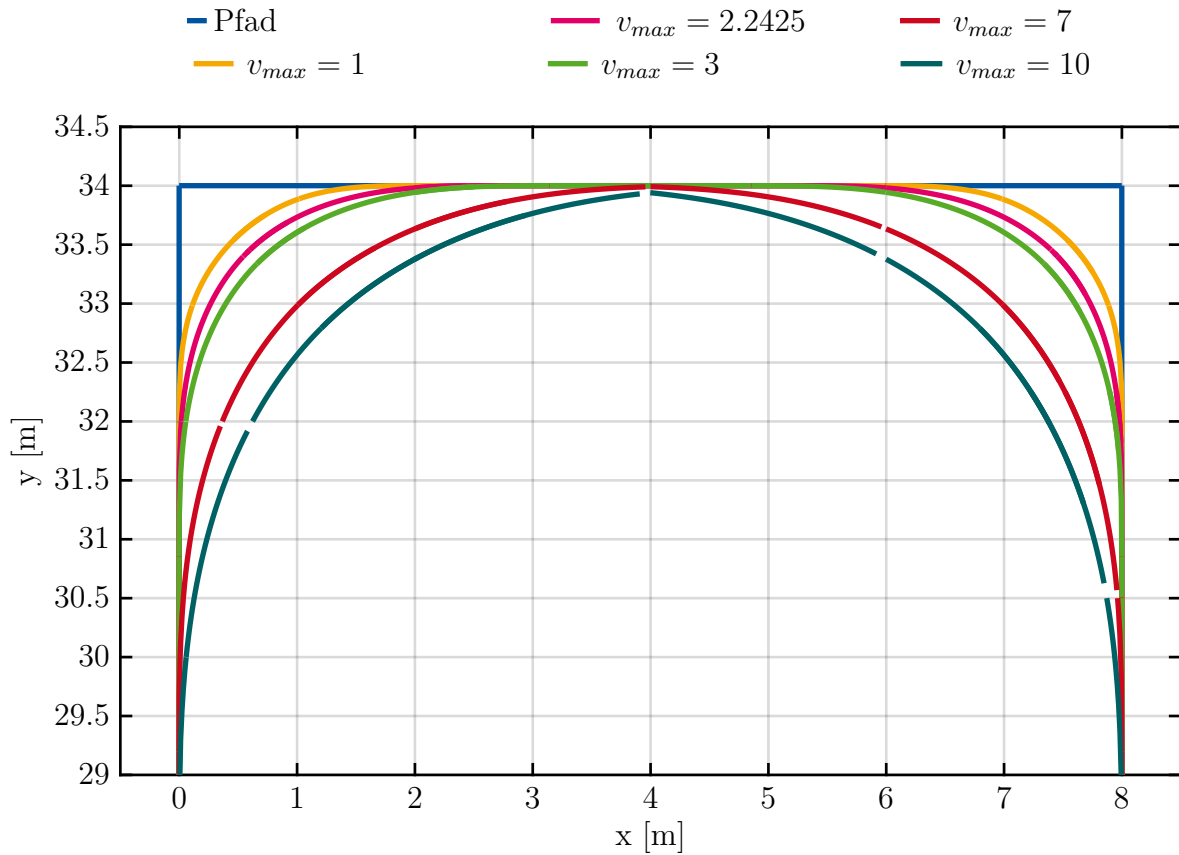
Somit sind alle für die Parameterdarstellung des Kreises nötigen Parameter bekannt bzw. können mit den Gleichungen aus Abschnitt 4.3.1.2 bestimmt werden. In Abbildung 4.15 sind die entsprechenden Winkel und der Kreisabschnitt (rot durchgezogen) für den Fall $i = 3$ eingezeichnet.

Neben den Pfadverläufen skizziert Abbildung 4.15 auch die zugehörigen qualitativen Verläufe der Krümmung κ . Dabei ist zu erkennen, dass im Fall der Klothoide (gelb) die Krümmung κ ständig auf- und abgebaut wird, folglich permanent gelenkt werden muss. Beim Kreis (rot) ist dies nicht so. Hier ändert sich die Krümmung κ sprungförmig beim Übergang von der Geraden in den Kreis, bleibt dann aber immer gleich, bis zur nächsten Gerade gewechselt wird. Dies führt zu einem ruhigeren Fahrverhalten, da nicht ständig gelenkt werden muss.

Durch numerische Abweichungen und die Abweichung der Punkte vom Kreis kann es in der Realität an den Übergängen von einem Kreisabschnitt zum nächsten zu Unstetigkeiten kommen. Diese können nicht exakt nachgefahren werden, was zu Abweichungen führt die von der Folgeregelung ausgeglichen werden müssen. Gleiches gilt, wenn der Pfad in der Karte eine Klothoide durch mehrere dicht aufeinander folgende Punkte repräsentiert. Dabei wird die Krümmung κ von einem Abschnitt zum nächsten immer weiter ansteigen respektive abnehmen, je nach Fahrtrichtung. Die Krümmung wird folglich einen stufenförmigen Verlauf annehmen mit Sprüngen in der Mitte benachbarter Punkte. Dennoch führt auch dies zu einem ruhigeren Fahrtverlauf als die Approximation mittel Klothoiden.

4.3.4.7 Einfluss von v_{max} auf die Trajektorie

Abbildung 4.15 zeigt am Beispiel von zwei aufeinanderfolgenden Verbundkurven, welchen Einfluss v_{max} auf die Trajektorie hat. Bei den verwendeten Kartendaten handelt es sich um einen in Kapitel 5.3.1 vorgestellten Streckenabschnitt. Es ist zu erkennen, dass mit zunehmender Geschwindigkeit immer früher begonnen werden muss, die Geraden zu verlassen. Dies liegt an der begrenzten Lenkgeschwindigkeit. Da nicht schneller gelenkt werden kann, muss bei höheren Geschwindigkeiten früher begonnen werden, zu lenken. Wird die Auslegungs-Geschwindigkeit v_{max} über Werte größer als 7,1 m/s erhöht, wird die entstehende Trajektorie unfahrbar. In Abbildung 4.15 ist dies für $v_{max} = 7,1$ m/s der Fall. Die obere Gerade wird nicht mehr erreicht, bevor sie bereits für die nächste Kurve verlassen werden muss. Es entsteht ein unstetiger Übergang, der die Anforderung der zweifach stetigen Differenzierbarkeit der Trajektorie verletzt. Übertragen auf das Beispiel aus Abbildung 4.11 würde dies bedeuten, dass \mathbf{k}_2 und \mathbf{k}_3 immer dichter zusammenrücken, bis ihre Reihenfolge vertauscht wird. Die Trajektorie ist genau bis zum Fall $\mathbf{k}_2 = \mathbf{k}_3$ einschließlich stetig und damit fahrbar. Für den Fall, dass \mathbf{k}_2 und \mathbf{k}_3 ihre Reihenfolge vertauschen, gibt der Trajektoriengenerator von beiden Seiten bis zum Mittelpunkt zwischen \mathbf{k}_2 und \mathbf{k}_3 stetige Trajektorien aus. Am Mittelpunkt selbst liegt die unstetige Stelle. Da die FBFR immer nur die aktuelle Sollwertvorgabe kennt, wird es an dieser Stelle einen Sprung der Sollwerte geben. Dieser wird dann von der Folgeregelung ausgeregelt.

Abb. 4.15: Referenzstrecke in Abhängigkeit von v_{max}

4.3.4.8 Ausgabe des Trajektoriengenerators

Anders als in den Abbildungen 4.11, 4.12 sowie 4.13 dargestellt, wird nicht der gesamte Sollwertverlauf \mathbf{y}_d (durchgezogene Linie in den Abbildungen) berechnet und ausgegeben, sondern immer nur die Werte, die zu der gegebenen Map Matching-Position \mathbf{p}_{MM} gehören. Das heißt \mathbf{y}_d , \mathbf{y}'_d und \mathbf{y}''_d werden in jedem Durchlauf nur an der Position \mathbf{p}_{MM} berechnet. Sind die zugehörigen Parameter der Parameterdarstellungen aus Abschnitt 4.3.1 der geometrischen Figuren, wie zuvor beschrieben, bestimmt, können mit den Gleichungen für Geraden - (4.16) bis (4.18) -, für Kreise - (4.26), (4.30) und (4.31) - sowie für Klothoiden - (4.36) bis (4.38) - die Werte für \mathbf{y}_d , \mathbf{y}'_d und \mathbf{y}''_d berechnet werden. Dabei ist zu beachten, dass die Berechnungen teilweise in verschobenen und rotierten bzw. gespiegelten Koordinatensystemen durchgeführt werden. Dies muss für allen Komponenten \mathbf{y}_d , \mathbf{y}'_d und \mathbf{y}''_d wieder korrigiert werden. Anschließend können die ermittelten Werte an die FBFR übergeben werden, vgl. Abbildung 4.10 auf Seite 76.

5 Testumgebungen

Im Rahmen der vorliegenden Arbeit werden sowohl simulative als auch reale Experimente durchgeführt. In den ersten beiden Abschnitten dieses Kapitels wird sich daher mit den verwendeten Testfahrzeugen auseinandergesetzt. Der in Abschnitt 2.8 vorgestellte IRT-Buggy ist eines der Testfahrzeuge. Für die Experimente müssen die in Abschnitt 5.1 vorgestellten Anpassungen durchgeführt werden. Als weiteres Testfahrzeug wird ein *E-MAXI L* verwendet. Mit ihm werden die mittels virtuellem Zug und Buggys erprobten Konzepte in realen Bahnanwendungen überprüft. Der *E-MAXI L* wird in Abschnitt 5.2 vorgestellt. Jedes der beiden Testfahrzeuge hat andere Ansprüche an die Testfelder, in denen die Experimente durchgeführt werden. So muss für Experimente mit den Buggys idealerweise eine asphaltierte Freifläche vorhanden sein, während für die Experimente mit dem *E-MAXI L* Gleise benötigt werden. In Abschnitt 5.3 werden die verwendeten Testfelder vorgestellt. Da die Simulationen so aufgebaut sind, als würden sie in den entsprechenden Testfeldern stattfinden, wird die Simulationsumgebung zu guter Letzt in Abschnitt 5.4 präsentiert.

5.1 IRT-Buggy Modifikationen

Damit bis zu acht Buggys in der Lage sind, den virtuelle Zug sicher zu realisieren, wurden einige Modifikationen und Erweiterungen für die Buggys entwickelt. Diese sollen hier vorgestellt werden. Dazu wird zunächst aufgezeigt, wie die Kommunikation der Buggys untereinander durch den GO!-Navigationsempfänger verbessert wird. Anschließend wird beschrieben, wie die existierenden CAN-Nachrichten erweitert bzw. neue Nachrichten hinzugefügt werden: zum einen um die für den virtuellen Zug nötige SLS zu berücksichtigen und mehrere unabhängige Fahrzeugverbände gleichzeitig zu unterstützen und zum anderen um die sichere Handhabbarkeit von bis zu acht Buggys gleichzeitig zu gewährleisten. Dazu wird am Ende dieses Abschnittes ein eigens entwickeltes Sicherheitskonzept vorgestellt.

5.1.1 Buggykommunikation mittels GO!-Navigationsempfänger

Wie bereits in Kapitel 2.4 beschrieben, bietet der GO!-Navigationsempfänger und insbesondere das SFCD dank seines Linux Betriebssystems die Möglichkeit, auf diverse Netzwerksoftwarekomponenten zuzugreifen. Dank dieser Softwarekomponenten kann

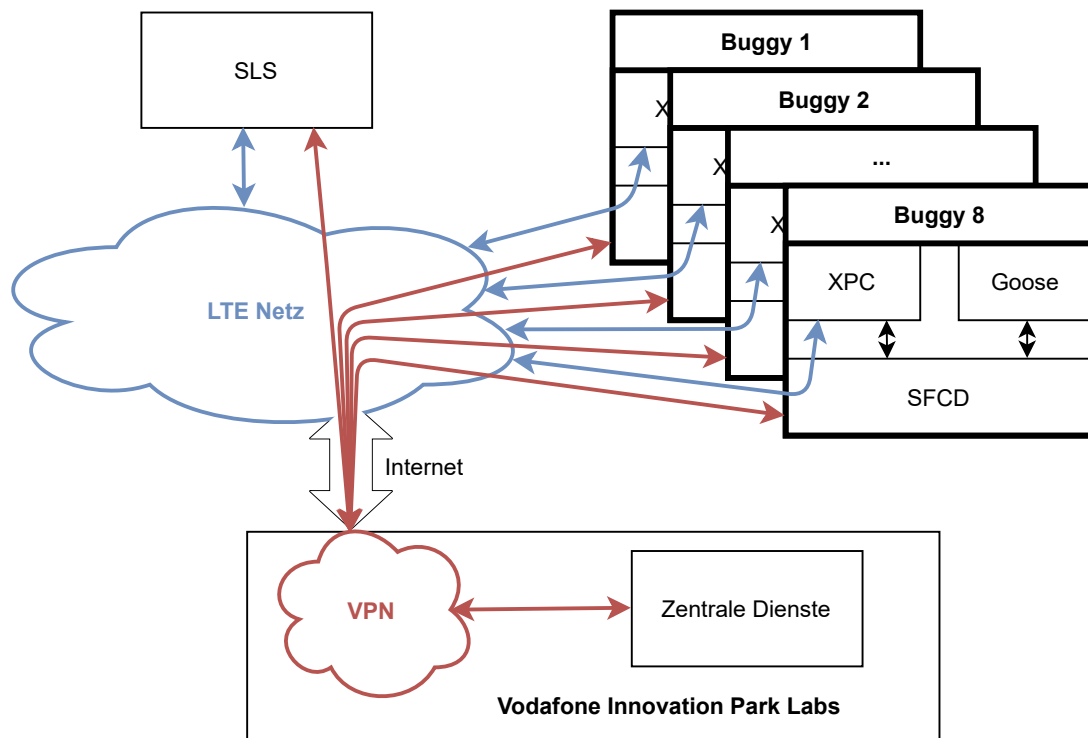


Abb. 5.1: Kommunikationsaufbau im Projekt GO!

problemlos via LTE-Netz ein VPN-Server, der vom Projektpartner Vodafone bereitgestellt wird, erreicht werden. Der Server befindet sich in den sogenannten *Vodafone Innovation Park Labs*, einer Testeinrichtung von Vodafone. Dort sind auch die zentralen Dienste, die mit den GO!-Empfängern via VPN kommunizieren, anzutreffen. Abbildung 5.1 zeigt den Aufbau des VPN, das von den Empfängern mittels Internetverbindung aus dem LTE-Netz erreicht wird (rote Pfeile in Abbildung 5.1). Wie die Empfänger kann sich auch die SLS in das VPN einwählen. So können über diesen Kommunikationskanal Weichenstellbefehle von den zentralen Diensten an die SLS gesendet werden. Um ihre Aufgabe als Stellwerk zu erledigen muss die SLS entsprechende Befehle an die Buggys senden. Dies erfolgt zur Realisierung des virtuellen Zuges und ist im realen Anwendungsszenario nicht vorgesehen. Daher wird eine vom VPN unabhängige Kommunikationsverbindung zwischen den Buggys und der SLS benötigt. Ursprünglich sollte dies über die ZigBee Verbindung der Buggys realisiert werden. Die Verbindung ist aber, wie bereits in Abschnitt 2.8 vorgestellt, nicht performant genug, um eine verlustfreie Kommunikation von bis zu acht Fahrzeugen und der SLS zu gewährleisten. Um das Problem zu umgehen, kann die, für die Realisierung des virtuellen Zuges, nötige Kommunikation über das SFCD und das LTE-Netz stattfinden. Dazu kann das SFCD mittels Routingtabellen eine Verbindung vom XPC durch das SFCD hindurch in das LTE-Netz aufbauen. Dank der Verwendung eines speziellen LTE-Testnetzes ist es so möglich, direkt im LTE-Testnetz miteinander zu kommunizieren. In Abbildung 5.1 ist dieser von der GO!-VPN-Verbindung unabhängige Kommunikationskanal in blau eingezeichnet. Die Verbindung ist mit einer Datenrate von über 50 Mbit/s schnell und stabil genug, sodass acht Buggys und die SLS gleichzeitig un-

tereinander kommunizieren können. Damit ist sie ideal für die Realisierung des virtuellen Zuges geeignet.

Bei der Verwendung der XPCs als SIMULINK® Real-Time™-Target, wie es im Rahmen dieser Arbeit geschieht, können User Datagram Protocol (UDP) Nachrichten gesendet und empfangen werden. Da das Basisboard der Buggys weiterhin CAN-Nachrichten erstellt und erwartet, bietet es sich an, das Format beizubehalten. Die CAN-Nachrichten können einfach byteweise in UDP-Nachrichten überführt werden und im LTE-Netz unter den Buggys ausgetauscht werden. Dies ermöglicht eine Weiterverwendung der bereits implementierten CAN-Parser, auch für den Datenaustausch via UDP. Dank der hohen Datenrate des LTE-Netzes können die CAN-Nachrichten weiterhin mit 100 Hz bzw. 10 Hz je nach Nachrichtentyp übertragen werden.

5.1.2 CAN Erweiterungen

In [50] wurde bereits untersucht, wie sich mehrere Fahrzeuge synchronisieren können, um die Längsregelaufgabe zu erfüllen. Basierend auf diesen Erfahrungen wird das CAN-Protokoll um weitere Nachrichten erweitert. Zur Fahrzeugkoordination wird die in Kapitel 3.3 vorgestellte SLS eingeführt. Diese hat unter anderem die Aufgabe, zu bestimmen, welche Fahrzeuge zu einem sogenannten Platoon vereint werden und welche Karte die Buggys intern verwenden. Als Platoon wird im Folgenden der Zusammenschluss mehrerer Fahrzeuge bezeichnet, die die Aufgabe haben, eine Einheit zu bilden. Um die Fahrzeuge einem Platoon zuzuordnen, wird die CAN-Nachricht Platoon-Config entwickelt. Die Nachricht ist für acht Buggys ausgelegt und ordnet jedem eine Platoon-ID und Platoon-Size zu. Die Platoon-ID nummeriert die entsprechenden Platoons. Die Platoon-Size gibt an, wie viele Fahrzeuge dem eigenen Platoon angehören. Die Platoon Konfiguration wird solange verwendet, bis eine neue Nachricht die alte überschreibt. Aus der Konfiguration können die Fahrzeuge bestimmen, wessen Fahrzeugdaten für die eigenen Berechnungen verwendet werden. Vergleiche hierzu die Kolonnenregelung aus Abschnitt 4.2.

Die wesentlichen Fahrzeugdaten werden zwischen den Fahrzeugen mit der Platoon-Data Fahrzeug zu Fahrzeug (engl. Vehicle to Vehicle) (V2V)-Nachricht ausgetauscht. Wie in [38] vorgestellt, liegen die Nachricht-Adressen in dem Adressbereich, der unter den Fahrzeugen ausgetauscht wird. Die Platoon-Data-Daten beinhalten die Fahrzeug- Geschwindigkeit, Beschleunigung und die Map Matching-Position in Gleiskoordinaten.

Als weitere Nachricht wird die Vehicle-Order-Nachricht hinzugefügt. Auch diese liegt im V2V-Kommunikationsbereich. In der Nachricht wird nochmals die eigene empfangene Platoon-ID und Platoon-Size übertragen. Dies bildet den Rückkanal der Platoon-Config-Nachricht. Diese Nachricht wird durch die SLS von jedem Buggy empfangen und ausgewertet. So ist die SLS in der Lage, festzustellen, dass alle Buggys dieselben Werte verwenden. Zusätzlich teilt jedes Fahrzeug mit, welche Reihenfolge der Fahrzeuge im eigenen Platoon detektiert wird. Dazu werden die Buggy-IDs der Buggys, die laut der Platoon-Config-Nachricht zum eigenen Platoon gehören, als geordnete Liste übertragen.

Die Länge der Liste ist durch die Platoon-Size definiert. Zur Bestimmung der Reihenfolge verwendet jeder Buggy die Daten aus der Platoon-Data-Nachricht und die geladene Karte. Eine Regelung ist nur sinnvoll, wenn alle Fahrzeuge aus einem Platoon die selbe Reihenfolge bestimmen. Befinden sich beispielsweise die Fahrzeuge mit der ID 3 und 7 auf dem gleichen Streckenabschnitt und die Fahrzeuge 8 und 2 befänden sich zwischen ihnen, dann wären die beiden Fahrzeugreihenfolgen $\{3, 8, 2, 7\}$ sowie $\{7, 2, 8, 3\}$ möglich und valide. Um ein eindeutiges Resultat zu erhalten, ist definiert, dass die Reihenfolge immer von dem Randfahrzeug mit der niedrigeren Fahrzeug ID zu dem mit der größeren angegeben wird. In diesem Beispiel ist also die Fahrzeugreihenfolge $\{3, 8, 2, 7\}$ zu verwenden.

Um die Buggys bei den Experimenten zentral überwachen zu können, werden zwei weitere Nachrichten definiert: zum einen eine Herzschlag (engl. Heart Beat) (HB)-Nachricht, die im Rahmen des Sicherheitskonzepts, siehe Abschnitt 5.1.3, eine zentrale Rolle spielt. Sie gibt im Wesentlichen an, ob mit dem entsprechenden Fahrzeug alles in Ordnung ist. Zum Anderen wird eine Statusnachricht definiert. Diese wird von der SLS empfangen und dient zur Fahrzeugüberwachung. Es wird übertragen, von welchen Fahrzeugen das betreffende Fahrzeug einen HB empfängt und ob es selbst einen HB sendet. Zusätzlich werden in der Status-Nachricht Informationen des virtuellen Zuges mitgeteilt. Dies sind die Informationen, ob eine Zielbremsung aktiv ist oder ob aktuell eine Notbremsung durchgeführt wird. Als weiterer wichtiger Datensatz wird auch die Batteriespannung übertragen. Sinkt diese zu stark ab, ist eine problemfreie Nutzung der Buggys nicht mehr garantiert und es kann zu einer Schädigung der Batterien kommen. Durch die zentrale Überwachung an der SLS ist es möglich, entsprechend zu reagieren.

5.1.3 Sicherheitskonzept

Für den Betrieb autonomer Systeme ist die Entwicklung geeigneter Sicherheitskonzepte zwingend erforderlich, um bei Fehlverhalten des Systems Schäden abwenden zu können. Ein Notaus bietet die Möglichkeit, das System im Fehlerfall abschalten zu können. Im Falle der Buggys wird diese Möglichkeit durch die Fernsteuerung (engl. Remote Control) (RC)¹ und das Basisboard bereitgestellt. Die Signale der Fernsteuerung RC_1 bis RC_3 liegen sowohl dem Basisboard als auch dem übergeordneten Reglersystem (via CAN) vor. Ein Kippschalter an der Fernbedienung eines jeden Buggys überführt den zugehörigen Buggy in definierte Zustände. Der Schalter hat drei mögliche Positionen.

Befindet sich der Schalter in der ersten Stellung ($RC_3 = 0$), werden vom Basisboard alle Steuerungsbefehle, die per CAN empfangen werden, verworfen. Nur die Stellung der Joysticks der Fernsteuerung wird berücksichtigt. Der Wert RC_1 bzw. RC_2 multipliziert mit der im Buggybasisboard aktuell eingestellten Maximalgeschwindigkeit bzw. dem maximalen Lenkwinkel $\varphi_{max} = 28^\circ$ ergibt die mittels Fernsteuerung vorgegebene

¹Die Signale RC_1 bis RC_3 bezeichnen die drei verwendeten Kanäle der Fernsteuerung:
Antrieb $\hat{=}$ $RC_1 \in [-1, 1]$; Lenkung $\hat{=}$ $RC_2 \in [-1, 1]$; Betriebsmodus $\hat{=}$ $RC_3 \in \{0, 0.5, 1\}$

Geschwindigkeit bzw. den vorgegebenen Lenkwinkel. Da die Joysticks der Fernbedienung sich, wenn sie nicht genutzt werden, automatisch in die mittlere Stellung zurück stellen, bedeutet dies, dass die Geschwindigkeits- und Lenkwinkelvorgabe beim Umlegen des Schalters gleich null werden. Der Buggy würde folglich die Lenkung gerade stellen und versuchen, zum Stillstand zu kommen.

In der mittleren Stellung ($RC_3 = 0,5$) werden die Steuerbefehle, die mittels CAN an das Basisboard gesendet werden, umgesetzt. Dabei besteht die Möglichkeit, nicht kommandierte Werte durch die Fernsteuerung vorzugeben. Das heißt, werden in dem Modus keine Signale über den CAN-Bus empfangen, wird das Fahrzeug wie zuvor über die Joysticks gesteuert. Wird ein Lenkwinkel per CAN vorgegeben, kann immer noch die Fahrzeuggeschwindigkeit durch die Fernsteuerung vorgegeben werden, solange keine CAN-Nachricht Strom oder Geschwindigkeit vorgibt. Da die Werte von RC_1 und RC_2 auch dem übergeordneten System zur Verfügung gestellt werden, ist es möglich, diese in der Regelung beliebig anders zu nutzen, solange der Regler Lenk- und Geschwindigkeitsvorgaben übermittelt.

Die dritte Schalterstellung ($RC_3 = 1$) bewirkt dasselbe Verhalten, als wäre die Fernsteuerung außerhalb ihrer Reichweite bzw. ausgeschaltet. In diesem Fall werden die Vorgaben vom übergeordneten Reglersystem beschränkt umgesetzt. Dieser Modus ist wie der vorher beschriebene, mit der Ausnahme, dass Strom und Geschwindigkeit sehr stark beschränkt sind. Gibt es in dem Modus keine Vorgaben, werden die Vorderräder entgegengesetzt nach außen ausgelenkt, sodass der Buggy gegen Wegrollen gesichert ist.

Wird ein Buggy genutzt, um einen Regler zu entwickeln, wird üblicherweise die mittlere Stellung verwendet. Sollte dabei ein unerwartetes Verhalten auftreten, besteht die Möglichkeit, den Schalter an der Fernsteuerung umzulegen. Damit kann entweder die Fernsteuerung aktiviert ($RC_3 = 0$) oder der Regler stark beschränkt werden ($RC_3 = 1$). Mit der Beschränkung ist es möglich, das Fehlverhalten sicherer beobachten zu können.

Wenn vernetzte Systeme untersucht werden sollen und dafür mehrere Buggys zum Einsatz kommen, müsste im Fehlerfall der Modus an der zum entsprechenden Buggy passenden Fernsteuerung umgestellt werden. Ggf. müssen auch andere Buggys angehalten werden, da durch das Abschalten des ersten Buggys dieser zu einem unerwarteten Hindernis werden kann. Idealerweise wird daher jedes Fahrzeug von einer Person beobachtet. Jede dieser Personen hat die entsprechende Fernsteuerung zu ihrem Buggy und kann so jederzeit die Kontrolle über den Buggy übernehmen. Dies stellt aber einen enormen Personalaufwand dar und ist nicht immer gerechtfertigt. Daher werden die Vorüberlegungen zum Sicherheitskonzept aus [50] aufgegriffen und wie im Folgenden beschrieben umgesetzt.

Die V2V-Kommunikation wird um eine HB-Nachricht für jeden Buggy und die SLS erweitert. Jede dieser HB-Nachrichten enthält nur ein Bit. Ist der HB aktiv, alterniert das Bit mit einer Rate von 10 Hz. Alle Kommunikationspartner senden ihre Nachricht an alle anderen. Wird eine HB-Nachricht empfangen, wird sie mit der vorher empfangenen HB-Nachricht desselben Senders verglichen und bei einer Änderung ein Timer für den entsprechenden Kommunikationspartner zurückgesetzt. In dieser Arbeit ist der Wert des

Timers zu 0,75 s gewählt. Die Buggys und die SLS haben also interne Timer für jeden Teilnehmer. Läuft einer dieser Timer ab, deaktiviert der Teilnehmer, bei dem der Timer abgelaufen ist, seine eigene HB-Nachricht. Dies hat zur Folge, dass nach weiteren 0,75 s in allen anderen Teilnehmern ein Timer abläuft und diese ihrerseits ihren HB abstellen. So wird erreicht, dass innerhalb von 0,75 s bis maximal 1,5 s ($= 2 \times 0,75$ s) alle HB abgestellt werden. Das System ist so umgesetzt, dass an die Buggys nur Kommandos geschickt werden, wenn der eigene HB aktiv ist. Wird das Schicken der Kommandos unterbrochen, so setzt der Buggy wie zuvor beschrieben die Vorgaben seiner Fernsteuerung um. Dies sind üblicherweise eine gerade Lenkstellung und eine Geschwindigkeit von 0 m/s, wenn sich die Joysticks in ihrer neutralen Position befinden. Der HB eines Fahrzeuges wird nur dann gestartet, wenn der Schalter der eigenen Fernsteuerung im mittleren Modus ist, genauer wenn $0,4 < RC_3 < 0,6$ gilt. Die SLS spielt eine gesonderte Rolle. Ihr HB wird durch das Drücken der Leertaste durch den Benutzer de-/aktiviert. Somit ist es möglich, alle Buggys durch eine einzelne Aktion nahezu zeitgleich anzuhalten. Dazu kann entweder an einer beliebigen Fernsteuerung der Modus umgestellt oder die Leertaste an der SLS gedrückt werden. Beides führt dazu, dass ein entsprechender HB abgeschaltet wird. Dadurch laufen in allen Fahrzeugen und der SLS die Timer ab und das System wird so in einen definierten Zustand überführt. Bei der Abschaltung werden alle Timer gelöscht und auf einen neuen HB der SLS gewartet. Wird dieser erneut gesendet, starten die Buggys ihren eigenen HB wieder, wenn die Voraussetzung $0,4 < RC_3 < 0,6$ erfüllt ist.

Neben den HB-Nachrichten gibt es Statusnachrichten, in denen jeder Buggy anzeigt, für welche anderen Buggys aktuell ein Timer läuft. Durch ein Zusammenführen dieser Information in der SLS kann analysiert werden, ob es Kommunikationsprobleme zwischen einzelnen Komponenten gibt. Der HB wird über den selben Kommunikationspfad wie die anderen V2V-Nachrichten übertragen. Wird der HB nicht empfangen, ist es wahrscheinlich, dass andere Nachrichten ebenfalls betroffen sind.

5.2 Testfahrzeug *E-MAXI L*

Die Abbildung 5.2 zeigt ein Foto eines Elektrorangierers der ZAGRO Group in der Modellvariante *E-MAXI L*. Es handelt sich um ein Zweiradfahrzeug, das über vier um 360° drehbare angetriebene Gummiräder verfügt. Damit kann sich das Fahrzeug im Straßenmodus in jede beliebige Richtung bewegen oder auf der Stelle drehen. Die Spurweite entspricht der von Gleisen, so dass das Fahrzeug beispielsweise an einem Bahnübergang eingeleist werden kann. Hierzu werden die Räder auf den Gleisen positioniert und anschließend Führungsrollen an der Vorder- und Rückseite des Fahrzeugs auf die Gleise abgelassen. Die Führungsrollen sind im unteren Teil von Abbildung 5.2 zu erkennen. Die Gummiräder sorgen auch im aufgleisten Zustand für den Vortrieb. Durch das hohe Eigengewicht von etwa 6,5 t kann genügend Traktion aufgebaut werden, um bis zu 350 t in der Ebene zu bewegen und auch zu bremsen. Der *E-MAXI L* ist etwa 3 m lang, 2 m breit und 1,2 m hoch.

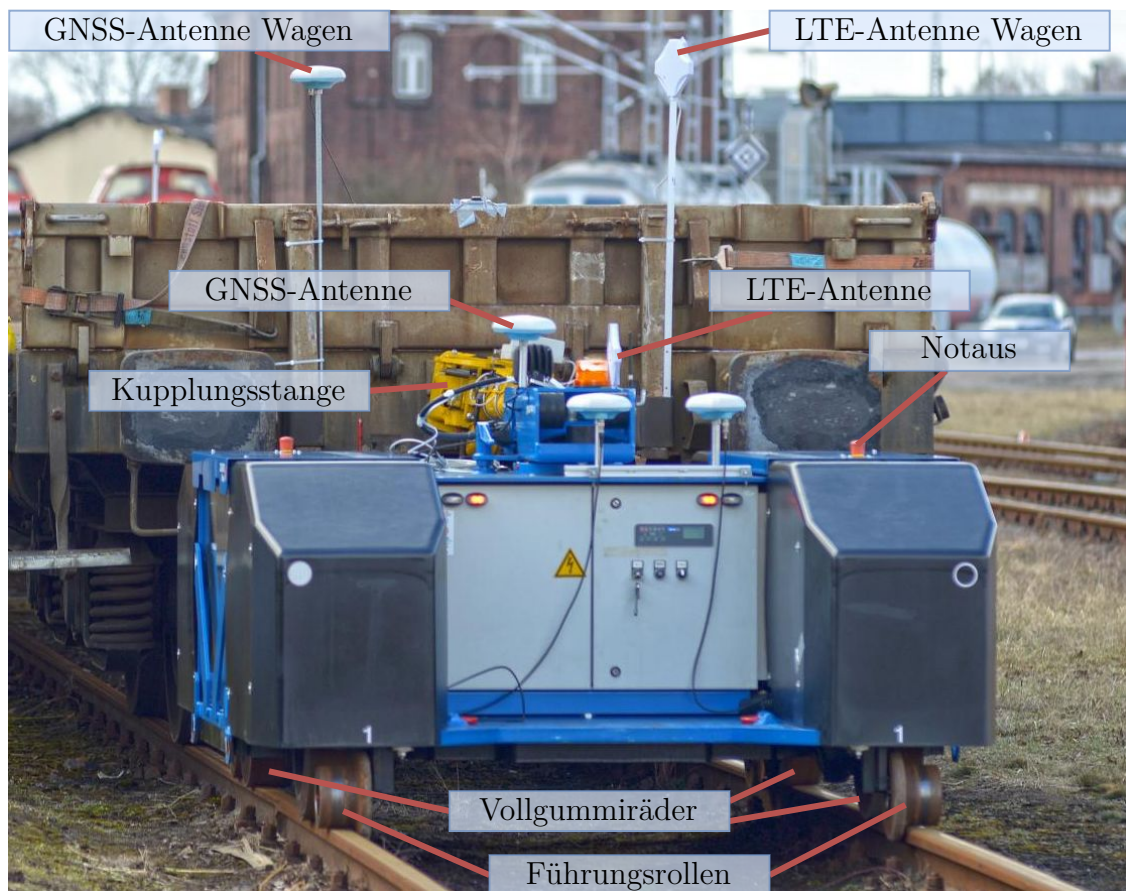


Abb. 5.2: Elektrorangierer *E-MAXI L* der ZAGRO Group [54]

Um Wagen an den *E-MAXI L* anhängen zu können, verfügt dieser über eine Kuppelstange, an deren Ende eine automatische Schnappkupplung montiert ist. Diese ist zu den im Bahnbetrieb üblichen Kupplungen kompatibel. Zum Entriegeln gibt es einen Antrieb, der wie die sämtlichen anderen Funktionen des *E-MAXI L* über dessen Funkfernsteuerung bedient werden kann.

Der *E-MAXI L* der ZAGRO Group soll verwendet werden, um die mit der virtuellen Schiene erzielten Ergebnisse mit den Ergebnissen aus einem realen Anwendungsfall zu vergleichen. Dazu dienen die Daten, die beim Projektabschluss des Projektes GO! aufgezeichnet werden. Um den Anwendungsfall durchführen zu können, muss der als Testfahrzeug dienende *E-MAXI L* automatisiert werden. Dies wird mit dem in Kapitel 2.10 vorgestellten Mann-in-der-Mitte-Ansatz erreicht. Normalerweise wird der Elektrorangierer mit einer Fernsteuerung bedient. Das Fahrzeug besitzt einen Funkempfänger, der die Steuersignale der Fernsteuerung empfängt und der zentralen Fahrzeugsteuerung bereitstellt. Die Abbildung 5.3 zeigt, wie diese Kommunikation manipuliert wird, um den Elektrorangierer zu automatisieren. Hierzu wird ein Steuerrechner zwischen Funkempfänger und Fahrzeugsteuerung geschaltet. Für den Funkempfänger verhält sich der Steuerrechner wie die Fahrzeugsteuerung und umgekehrt. So wird erreicht, dass die Komponenten normal weiter arbeiten und nicht in einen Fehlermodus schalten. Der Steuerrechner kann

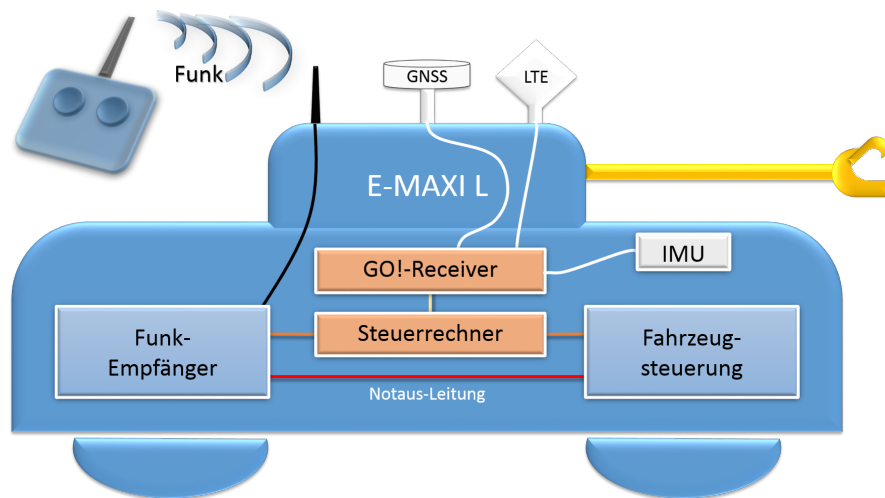


Abb. 5.3: Hardware-Aufbau des GO!-Systems im *E-MAXI L* [56]

die Befehle der Fernsteuerung beliebig manipulieren und so das Fahrzeug entsprechend hinterlegten Abläufen oder Regelungen kontrollieren. Eine solche hinterlegte Regelung ist die in Kapitel 4.1 vorgestellte Zielbremsung. Da als Steuerrechner ein XPC zum Einsatz kommt, kann das selbe SIMULINK®-Modell wie für die Buggys verwendet werden. Hierbei ist zu beachten, dass der Kuppelparameter k der Zielbremsung nun intern nicht überschrieben wird. So ist auch das Anfahren eines nicht exakt eingemessenen Wagens möglich. Die benötigten Messwerte wie beispielsweise GNSS-Position und -Geschwindigkeit werden dem Steuerrechner vom GO!-Empfänger bereitgestellt. Auch die Beschleunigung, die zur Detektion eines Auffahrens auf einen stehenden Wagen verwendet wird, steht durch eine an den GO!-Empfänger angeschlossene IMU zur Verfügung.

Um das Szenario abzubilden, in dem mit dem *E-MAXI L* Wagen rangiert werden soll, kann die Steuerung auch durch die zentralen Dienste erfolgen. Dazu erhält der Steuerrechner über die LTE-Verbindung des GO!-Empfängers die Anweisungen, wie sich der Elektrorangierer verhalten soll. Ein weiteres Beispiel eines hinterlegten Ablaufes neben der Zielbremsung ist das automatische Abkuppeln. Dazu wird der Kuppelhaken des *E-MAXI L* für 12 Sekunden geöffnet. Nach der Hälfte der Zeit wird der *E-MAXI L* für 3 Sekunden von dem angehängten Wagen weggefahren. Weitere 3 Sekunden später sind die 12 Sekunden abgelaufen und der Kuppelhaken schließt wieder. Damit ist das Abkuppelmanöver beendet.

Die Fernsteuerung bleibt weiterhin als Sicherheitskomponente erhalten. In der Standardvariante des *E-MAXI L* ist ein Kippschalter der Fernsteuerung ungenutzt. Von ihm gesendete Signale werden von der Fahrzeugsteuerung ignoriert. Dieser Kippschalter wird nun genutzt, um zwischen Normalbetrieb und Automatisierung umzuschalten. Im Normalbetrieb gibt der Steuerrechner alle Nachrichten der Fernsteuerung unverändert weiter. Wird der Kippschalter umgelegt, werden alle Nachrichten der Fernsteuerung ignoriert und der Steuerrechner gibt seine selbst berechneten Signale an die Fahrzeugsteuerung. Sollte es zu einem Fehler im automatisierten Modus kommen, kann durch Umlegen des Kipp-

schalters die Kontrolle an die Fernsteuerung zurück gegeben werden. Zusätzlich kann der Notausschalter als weitere Rückfallebene verwendet werden. Fällt beispielsweise der Steuerrechner aus und gibt die Befehle der Fernsteuerung bei entsprechender Kippschalterstellung nicht mehr weiter, kann der Notausschalter an der Fernsteuerung gedrückt werden. Es wird ein entsprechendes Signal an den Funkempfänger im *E-MAXI L* gesendet. Dieser ist zusätzlich zu der digitalen Verbindung zum Steuerrechner nochmals separat mit der Fahrzeugsteuerung verbunden. Über diese analoge Verbindung wird nur das Notausignal zusätzlich zur digitalen Verbindung übertragen. In Abbildung 5.3 ist die analoge Verbindung durch eine rote Leitung dargestellt. Das Notaus-Signal kann also vom Steuerrechner nicht unterbrochen und so in jedem Fall durch die Fahrzeugsteuerung umgesetzt werden. Es bewirkt ein sofortiges Anziehen der Bremsen.

5.3 Testfelder

In diesem Abschnitt wird auf die unterschiedlichen Testfelder eingegangen, die im Rahmen des Projektes GO! und bei der Erstellung dieser Arbeit genutzt werden. Jedes der Testfelder besitzt individuelle Vor- und Nachteile. Daher werden an unterschiedlichen Orten verschiedene Tests durchgeführt. In den folgenden Abschnitten werden die Testfelder kurz vorgestellt. Dabei wird ebenfalls aufgezeigt, welche Tests dort durchgeführt werden und warum sich das entsprechende Gebiet für diese Tests eignet.

5.3.1 Parkhaus

Im Bereich Campus Melaten gibt es ein Parkhaus der RWTH, das nicht weit vom IRT entfernt ist. Es eignet sich gut für weniger aufwendige Testfahrten mit den Buggys. Auf der obersten Etage können die Experimente unter freiem Himmel durchgeführt werden, sodass der Empfang von GNSS-Signalen gewährleistet ist.

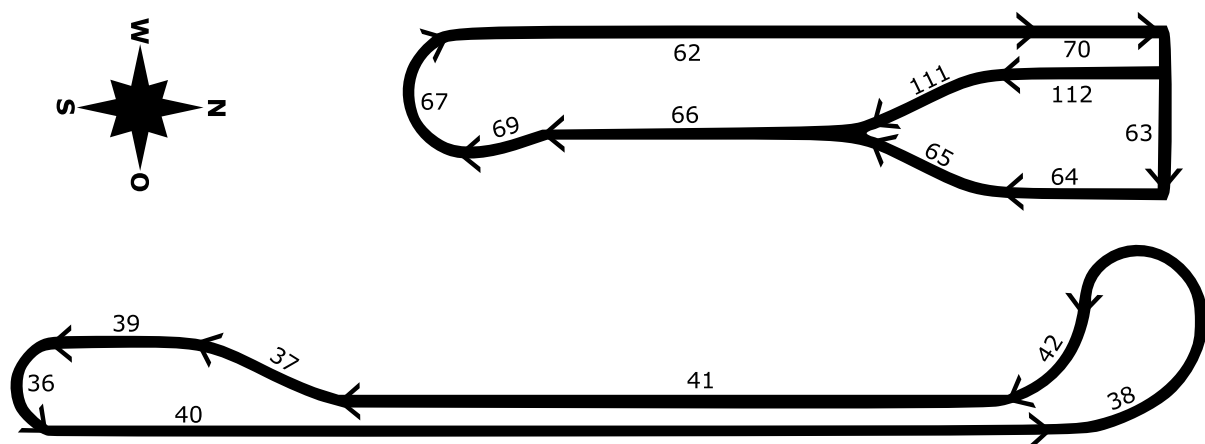


Abb. 5.4: Gedrehte Gleiskarte der Strecken A (oben) und B (unten)



zen. Dabei wird die kürzere der beiden Strecken als A und die längere als B bezeichnet. Die Strecken werden auch in den Arbeiten [38, 39, 41, 50] genutzt. In Abbildung 5.5 sind die Kartendaten auf eine Luftaufnahme des Parkdecks projiziert. Die hinterlegten Strecken zeichnen sich dadurch aus, dass wesentliche Streckenelemente vorhanden sind. Diese beinhalten insbesondere die in Kapitel 4.3.1 vorgestellten geometrischen Figuren Gerade, Kreis und Klothoide. Zusätzlich wurde versucht, ausreichend Abstand zu den begrenzenden Wänden zu halten, damit auch im Fehlerfall ein Anhalten der Buggys noch möglich ist. Im oberen Teil der Strecke B in Abbildung 5.5 ist eine Klothoide, die von einer Geraden in ein Kreisstück mit entgegengesetzter Drehrichtung übergeht. Diese harten Übergänge in der Streckenführung eignen sich gut, um die Regelgüte zu untersuchen. Ein weiteres Beispiel hierfür sind die beiden Übergänge von einer Gerade in eine andere um 90° gedrehte Gerade im Streckenverlauf von Strecke A . Sie werden durch die Abfolge der Wege 70, 63 und 64 aus erzeugt, vgl. Abbildung 5.4.

5.3.2 Aldenhoven Testing Center

Das Aldenhoven Testing Center (ATC) ist ein Joint Venture der RWTH Aachen University und des Kreises Düren, das von der Europäischen Union und dem Ministerium für Wirtschaft, Innovation, Digitalisierung und Energie des Landes Nordrhein-Westfalen gefördert wurde. Es ist ein Testgebiet für Automotiv-Anwendungen. Auf unterschiedlichen Streckenabschnitten können diverse Mobilitätsszenarien abgebildet werden. Das ATC verfügt über einen Oval-Kurs, der aus zwei Steilkurven und zwei verbindenden Geraden besteht. Außerdem gibt es Streckenanschnitte mit unterschiedlichen Fahrbahnbelägen, einen Steigungshügel, einen Handlung-Kurs inklusive 1 g Senke, eine bewässerbare Bremsenstrecke, einen Kreuzungsbereich und eine Fahrdynamikfläche.

Abbildung 5.6 zeigt eine Luftaufnahme des ATC. In dieser ist die kreisrunde Fahrdynamikfläche gut zu erkennen. Sie verfügt über eine etwa 400 m lange gerade Anfahrstrecken und besitzt einen Durchmesser von 210 m.

Für die Experimente dieser Arbeit stellt diese große asphaltierte Fläche, die nur über 3 % Neigung verfügt, ein ideales Testfeld dar. Durch die Größe der Fahrdynamikfläche lassen sich sehr komplexe Szenarien darstellen, die auch mit höheren Fahrgeschwindigkeiten absolviert werden können, da es genug Ausrollfläche gibt. So ist die Gefahr, dass ein Testfahrzeug bei den Versuchen Schaden nimmt, geringer als beispielsweise auf dem Parkdeck aus Abschnitt 5.3.1. Des Weiteren stehen im näheren Umfeld keine Hindernisse, wie zum Beispiel Häuser oder Bäume, die die Satellitensignale ablenken oder reflektieren könnten. Folglich können Navigations-Tests ohne Störungen durch Abschattungen oder Mehrwegeausbreitung durchgeführt werden.

Aufgrund der zuvor genannten Eigenschaften werden in diesem Testgebiet komplexe Szenarien mit den Buggys durchgeführt. Abbildung 5.7 zeigt den Gleisplan, der für die Tests im ATC hinterlegt ist. Die Pfeile an den Gleisen geben die Vorwärtsrichtung des entsprechenden Gleises an und die Nummern entsprechen den intern vergebenen Gleisnummern.

²JOSM ist ein erweiterbarer Opensource Editor für OSM-Karten (<https://josm.openstreetmap.de>).



Abb. 5.6: Luftaufnahme inklusive Kartendaten des Aldenhoven Testing Center; erstellt mit JOSM mit Bilddaten von Geobasis NRW

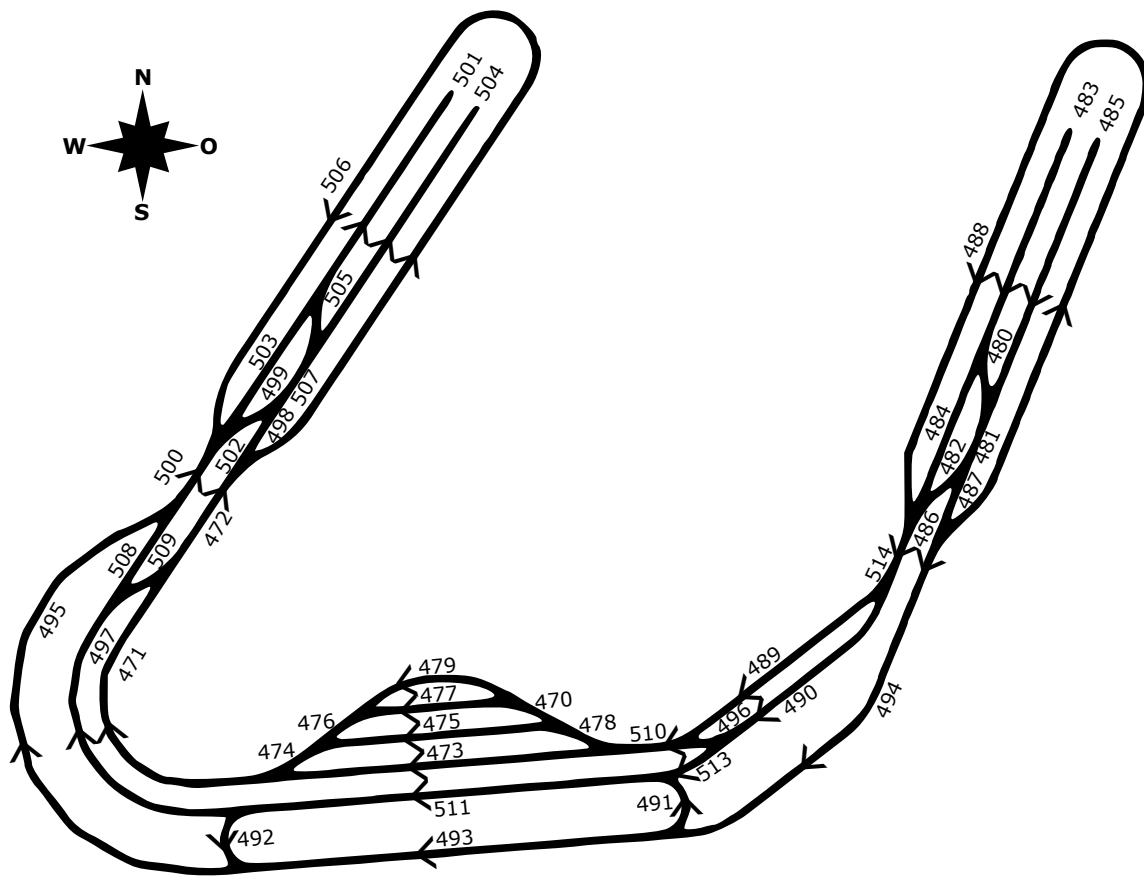


Abb. 5.7: Gleiskarte des virtuellen Rangierbahnhofs auf der Fahrdynamikfläche

Das abgebildete Gebiet ist etwa $100\text{ m} \times 100\text{ m}$ groß und nimmt damit etwas mehr als ein Viertel der Fahrdynamikfläche ein. Es liegt auf der rechten Seite, wenn über die Zuführung auf die Fahrdynamikfläche gefahren wird, vgl. die in gelb auf die Fahrdynamikfläche projizierte Gleiskarte in Abbildung 5.6.

Die Karte bildet einen imaginären Rangierbahnhof nach. Dafür bildet die Gleisharfe in der Mitte die Rangier-Gruppe eines Rangierbahnhofs ab. Eine Einfahr-Gruppe des Bahnhofs ist aus den Gleisen 489 und 490 gebildet. Auf eine Ausfahr-Gruppe wird verzichtet. Außerdem gibt es kein Verbindungsnetz zu anderen Bahnhöfen. Stattdessen werden die Wendegleise 506 bzw. 488 mit den eingeschlossenen Gleisen angeschlossen. Über diese Gleise ist es möglich, einen Zug zu wenden, um in anderer Fahrtrichtung in den Bahnhof einzufahren. Unterhalb der Rangier-Gruppe gibt es ein Oval, das sich aus den Gleisen 491, 492, 493 und 511 zusammensetzt. In dem Oval können Fahrten im Kreis durchgeführt werden, ohne die gesamte Strecke abfahren zu müssen. Bei einer Fahrt durch das Oval wird eine Strecke von rund 101 m zurückgelegt.

Im Rahmen des Projektes GO! werden hier die Szenarien Wagenverlust, Data-Analytics und Rangieren getestet, siehe [56] für Details. Das aufwendige Szenario des Rangierens stellt das Kernexperiment dieser Arbeit dar. Beim Rangieren muss alles zusammenspie-

len: Längs- und Querführung der Buggys einzeln, aber auch im Verbund. Nur so kann gezeigt werden, dass das Konzept des virtuellen Zuges geeignet ist. In Abschnitt 6.5 wird das Rangierexperiment im Detail vorgestellt.

5.3.3 Rail und Logistic Center Wustermark



Abb. 5.8: Luftaufnahme des Rail und Logistic Center Wustermark GmbH & Co. KG inkl. GO!-Teststrecke (blauer Kasten)
Quelle: Google © 2020 GeoBasis-DE/BKG, GeoContent, Maxar Technologies

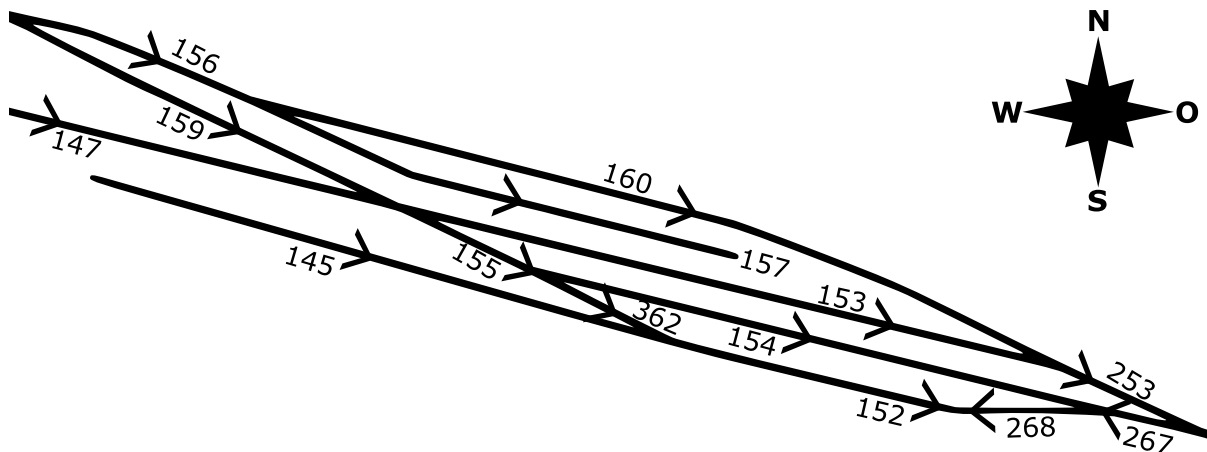


Abb. 5.9: Gleiskarte im Rail und Logistic Center Wustermark GmbH & Co. KG
(entspricht den Gleisen im blauen Kasten in Abbildung 5.8)

Die Rail und Logistic Center Wustermark GmbH & Co. KG (RLCW) ist Betreiber des größten privat geführten öffentlichen Bahnhofs für den Güterverkehr in Deutschland. Im Jahr 2008 übernahm sie den Rangierbahnhof Wustermark von der DB Netz AG. Der Bahnhof liegt in Brandenburg westlich von Berlin und dient im Projekt GO! zur

Validierung der Anwendungsszenarien des GO!-Empfängers. Diese Szenarien werden im Rahmen dieser Arbeit herangezogen, um zu bestätigen, dass die Buggys mit dem Konzept des virtuellen Zuges ein geeignetes Mittel zur Entwicklung darstellen.

Alle Tests im Rangierbahnhof werden mit dem GO!-Empfänger und dem in Abschnitt 5.2 vorgestellten Elektrorangierer durchgeführt. Zusätzlich kommen zwei Flachwagen, die vom *E-MAXI L* rangiert werden, zum Einsatz. Die Experimente werden in dem in Abbildung 5.8 blau eingezeichneten Bereich durchgeführt. Abbildung 5.9 zeigt die Lage der Gleise in diesem Bereich. Dabei sind Gleisnummern und Fahrtrichtung aus der verwendeten OSM-Karte eingezeichnet. Um die Darstellung zu verbessern, ist der Gleisplan nicht maßstabsgetreu. Er wurde in *y*-Richtung mit dem Faktor 2 skaliert. Die Breite des betrachteten Ausschnitts (lange Seite des blauen Kastens in Abbildung 5.8) beträgt etwa 250 m. Für die Demonstration werden die Gleise 156, 157 sowie 160 verwendet. In der Realität haben die Gleise andere Nummern. Die Gleise 156 und 160 sind in der Realität Gleis 48, von dem Gleis 49 (157 in der Karte) an Weiche 197 abgeht.

5.4 Simulationsumgebung

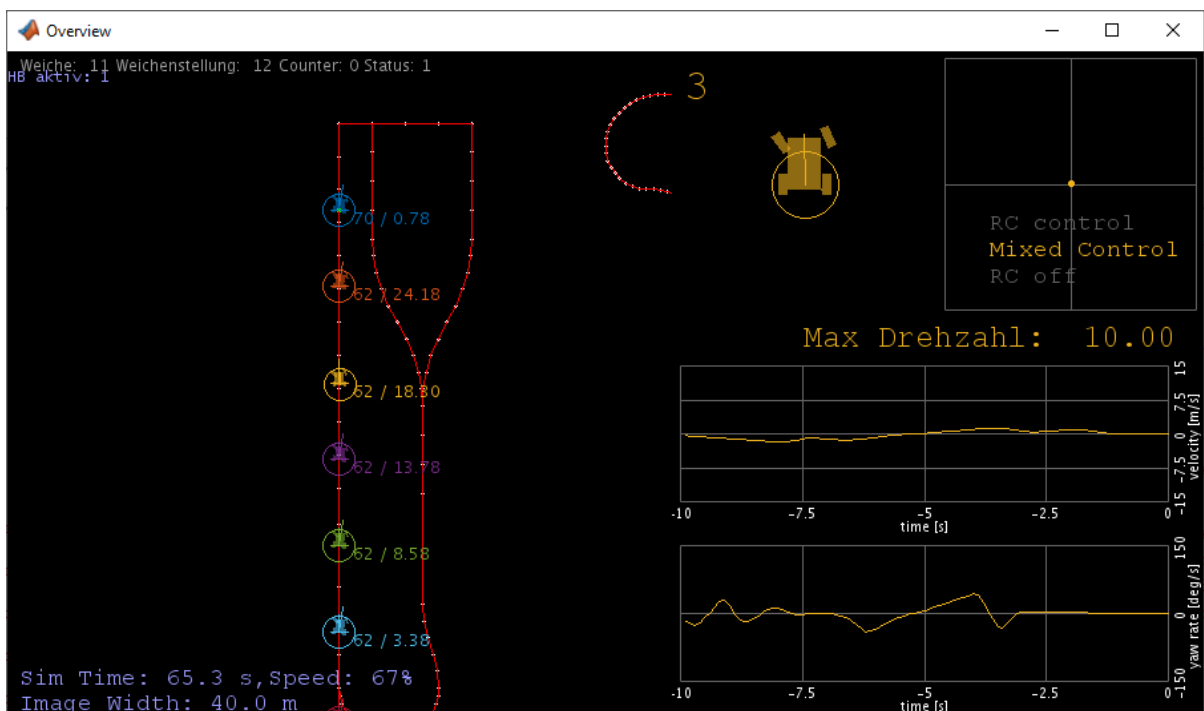


Abb. 5.10: Screenshot der Simulationsumgebung mit sechs Buggys auf der Strecke A

Am IRT wurde eine Simulationsumgebung geschaffen, die genutzt wird, um bis zu acht Buggys gleichzeitig zu simulieren. Abbildung 5.10 zeigt einen Screenshot der Umgebung. In der Simulationsumgebung wird, zur Fahrzeugsimulation, eine angepasste Version des bereits in Abschnitt 2.8 vorgestellten HiL-Modus-Fahrzeugmodells verwendet, vgl. [40].

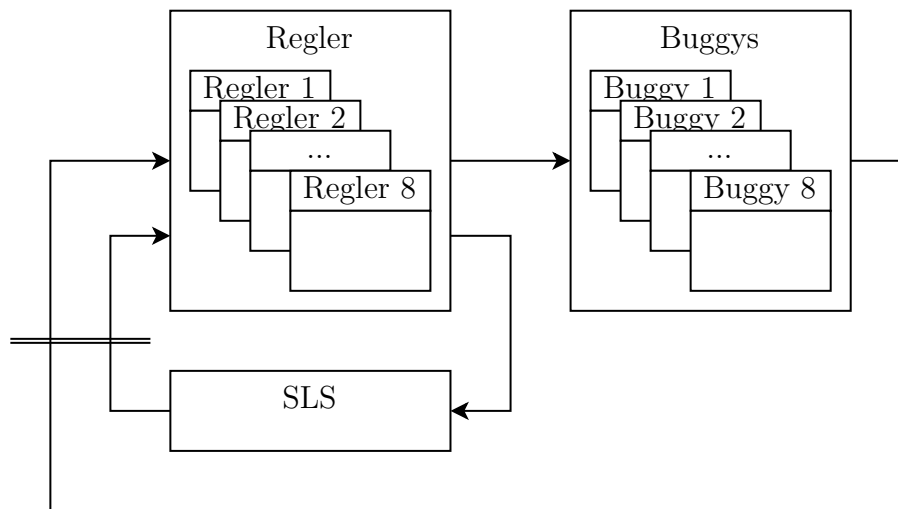


Abb. 5.11: Aufbau des Regelkreises innerhalb der Simulationsumgebung

Der Unterschied besteht im Wesentlichen in der Verwendung von einer 64 bit-Architektur anstelle der auf dem Basisboard zur Verfügung stehenden 32 bit-Architektur. Da die Simulationsumgebung ebenfalls in MATLAB[®]/SIMULINK[®] realisiert ist, können die SIMULINK[®]-Modelle für die XPCs sehr einfach in die Simulation eingebunden und verwendet werden. Sowohl für die Simulation als auch für die real durchgeführten Experimente werden dieselben SIMULINK[®]-Modelle verwendet. Dies gilt insbesondere für die Zielbremsung, die Kolonnenregelung, die Pfadregelung, die FBFR sowie die Szenarioleitstelle.

Intern wird die Simulation in drei Teile aufgeteilt - Buggys, Szenarioleitstelle und Regler. Mit diesen drei Komponenten ergibt sich der in Abbildung 5.11 abgebildete Regelkreis, wobei die Komponenten Buggys und Regler jeweils acht Instanzen enthalten. Der Doppelstrich in der Abbildung zeigt an, wo algebraische Schleifen durchbrochen werden. Dazu werden ausgehend von dem Doppelstrich alle Werte berechnet. Die Werte, die in den Doppelstrich einlaufen, werden vorgehalten und im nächsten Simulationsschritt als Startwerte verwendet. Die Buggys bekommen initial eine Position, von der aus sie sich entsprechend der Steuereingaben bewegen können. Dabei wird die Bewegung gemäß des hinterlegten Modells berechnet und als Messwert der Sensoren ausgegeben. Steuereingabe und Messwertausgabe geschehen im selben CAN-Nachrichtenformat wie beim richtigen Buggy. Sie werden aber nicht über CAN übermittelt, sondern stehen den anderen Komponenten verzögerungsfrei zur Verfügung.

Die Simulationen werden auf einem Laptop durchgeführt. Dieser ist ausgestattet mit einer Intel[®] Core[™] i7-8850H CPU mit sechs Kernen und 32 GB Arbeitsspeicher. In den Simulationen werden unterschiedliche Szenarien betrachtet.

Auf der einen Seite wird das allgemeine Verhalten beim Abfahren von vorgegebenen Strecken betrachtet. Hierzu werden zwei Teststrecken verwendet, die möglichst viele unterschiedliche geometrische Elemente enthalten. Die Strecken sind die im Abschnitt 5.3.1 vorgestellten Strecken A und B auf dem Parkhaus, vgl. Abbildung 5.4.

Auf der anderen Seite wird betrachtet, wie sich die Buggys beim Durchlaufen des Abschlusszenarios des GO!-Projektes verhalten. In Abbildung 5.7 ist die hierbei verwendete Gleiskarte abgebildet. Die soeben skizzierten Simulationen werden in Kapitel 6 durchgeführt, im Detail beschrieben und diskutiert.

6 Numerische und experimentelle Validierung

Im Rahmen dieses Kapitels werden die erarbeiteten Regelungskonzepte auf die Probe gestellt. Dazu wird zunächst in Abschnitt 6.1 gezeigt, dass die vorgestellte Simulationsumgebung ein geeignetes Entwicklungswerkzeug darstellt. Dies geschieht durch den Vergleich von Simulationsdaten mit den Messdaten eines real mit den Buggys durchgeführten Experiments. Das Szenario ist dabei in beiden Fällen dasselbe. Simulation und Experiment werden sowohl mit der Pfadregelung als auch der FBFR durchgeführt.

Nachdem gezeigt wurde, dass in der Simulationsumgebung valide Ergebnisse erzielt werden, wird sie in Abschnitt 6.2 dazu verwendet, um die Parameter der FBFR zu untersuchen. Dies stellt ein Beispiel dar, wie die Simulationsumgebung als Entwicklungswerkzeug genutzt werden kann. Bei der Untersuchung steht der Einfluss der Reglerparameter K_0 und K_1 im Fokus.

Neben der virtuellen Schiene wird auch das Konzept der virtuellen Kupplung benötigt, um den virtuellen Zug zu realisieren. Daher wird in Abschnitt 6.3 zunächst eine simulative Voruntersuchung durchgeführt. Dies geschieht im HiL-Modus der Buggys. Anschließend wird ein Experiment durchgeführt und mit der Simulation verglichen.

Pfadregelung und FBFR sind unterschiedlich aufgebaut und arbeiten mit verschiedenen Raten. In Abschnitt 6.4 wird untersucht, welchen Einfluss das auf die Performanz beider Regler hat. Hierzu werden sowohl Simulation als auch Experiment betrachtet.

In Abschnitt 6.5 wird ein Teil des Abschlusszenarios aus dem Projekt GO! vorgestellt. Es handelt sich um das Szenario des Rangierens. Bei dem Szenario wird das Zusammenspiel der einzelnen Komponenten auf die Probe gestellt. Dies betrifft die virtuelle Kupplung und die virtuelle Schiene. Dazu gehört auch die SLS, die zur Weichenstellung benötigt wird. Das Szenario prüft folglich alle Komponenten des virtuellen Zuges ab.

Schlussendlich wird in Abschnitt 6.6 die mit dem *E-MAXI L* durchgeführte Abschlussdemonstration im Projekt GO! untersucht. In der Demonstration wird das Rangieren von Güterwagen an einer Gleisharfe gezeigt. Die hierfür nötigen Entwicklungen sind zuvor unter Verwendung des virtuellen Zuges durchgeführt worden. Aus dem Ablauf der Demonstration und den aufgezeichneten Messwerten werden Rückschlüsse auf die Entwicklungsphase gezogen. So wird das Konzept des virtuellen Zuges als Entwicklungswerkzeug bewertet.

Im letzten Abschnitt dieses Kapitels werden die in den anderen Abschnitten vorgestellten Ergebnisse kurz zusammengefasst und bewertet.

6.1 Validierung der Simulation

Zur Validierung der Simulationsumgebung wird gezeigt, dass die mit ihr erzielten Simulationsergebnisse plausibel und realitätsnah sind. Dazu werden dieselben Experimente sowohl in der Realität als auch in der Simulation durchgeführt sowie miteinander verglichen. Als Testgebiet dient das in Abschnitt 5.3.1 vorgestellte Parkhaus. Die Teststrecken *A* und *B* zeichnen sich durch harte Übergänge zwischen den unterschiedlichen geometrischen Figuren Gerade, Kreis und Klothoide aus. Durch diese anspruchsvollen Strecken können die Regler aussagekräftig untersucht werden. In der Anwendung des virtuellen Zuges werden deutlich sanftere Übergänge verwendet. Beide Strecken auf dem Parkhaus werden von einem Buggy mit einer konstanten Geschwindigkeit von 2 Radumdrehungen pro Sekunde ($\cong 2,2425 \text{ m/s}$) durchfahren.

Bei der realen Testfahrt wird ein AsteRX 3 GNSS-Empfänger der Firma Septentrio verwendet. Der verwendete Empfänger ist RTK-fähig und wird über eine LTE-Verbindung mit Ntrip-Daten versorgt. Mit einer Updaterate von 5 Hz liefert dieser die in Kapitel 2.3.1 vorgestellten NMEA-Datensätze an das Basisboard, das diese dem Regler mittels CAN bereitstellt. Dieser wiederum kommandiert dem Basisbord die Sollwerte ebenfalls über CAN an die Aktoren. Auf dem CAN-Bus können also alle wesentlichen Daten gefunden werden, wenn auch in serialisierter Form, da nur immer eine Nachricht gleichzeitig übertragen werden kann. Durch die gewählte maximale CAN-Busübertragungsrate von 1 MBd spielt dies aber keine Rolle, da die Übertragungsdauer der Nachrichten wesentlich kleiner ist als die Taktrate des Reglers. Während der Experimente wird die gesamte CAN-Kommunikation aufgezeichnet. Die in diesen Aufzeichnungen gesammelten Daten werden für die folgenden Auswertungen verwendet. In der Datennachbearbeitung müssen die seriell aufgezeichneten Daten wieder auf die selbe Zeitbasis bezogen werden. So kann der Reglerzustand rekonstruiert werden.

Wie zuvor in Abschnitt 5.4 beschrieben, werden in der Simulation keine CAN-Nachrichten ausgetauscht. Die Daten stehen verzögerungsfrei in jedem Simulationsschritt zur Verfügung und können synchron gespeichert werden. Für die Simulation werden die selben Daten aufgezeichnet. Sie werden im SIMULINK®-Modell ausgelesen und gespeichert, bevor sie im echten Experiment in die CAN-Nachrichten geschrieben werden würden.

In Abbildung 6.1 ist ein Überblick über die Strecken *A* und *B* auf dem Parkdeck gegeben. Die Abbildung enthält den sich aus den Kartendaten ergebenden Pfad. Außerdem sind sogenannte Punkte von Interesse (engl. Point of Interest) (POI) entlang der beiden Strecken eingezeichnet. In Tabelle 6.1 ist eine Liste der POI aufgeführt, die auch die Entfernungen jedes POI vom Startpunkt der entsprechenden Strecke in Gleiskoordinaten enthält. Die Strecke *A* enthält zwei 90°-Kurven, zwei einfache Versätze und einen Kreisbogen mit einem Radius von etwa 3 m. Der Koordinatenursprung für beide Strecken liegt am Übergang vom linken Kreisbogen in die linke Gerade. Er definiert auch den Startpunkt für die Strecke *A* (POI *a*). Die Strecke *A* wird in den folgenden Untersuchungen rechtsherum durchfahren. Bei der Strecke *B* liegt der Startpunkt (POI *h*) am Übergang vom rechten Kreisbogen in die rechte Gerade. Sie wird in den folgenden Untersuchungen

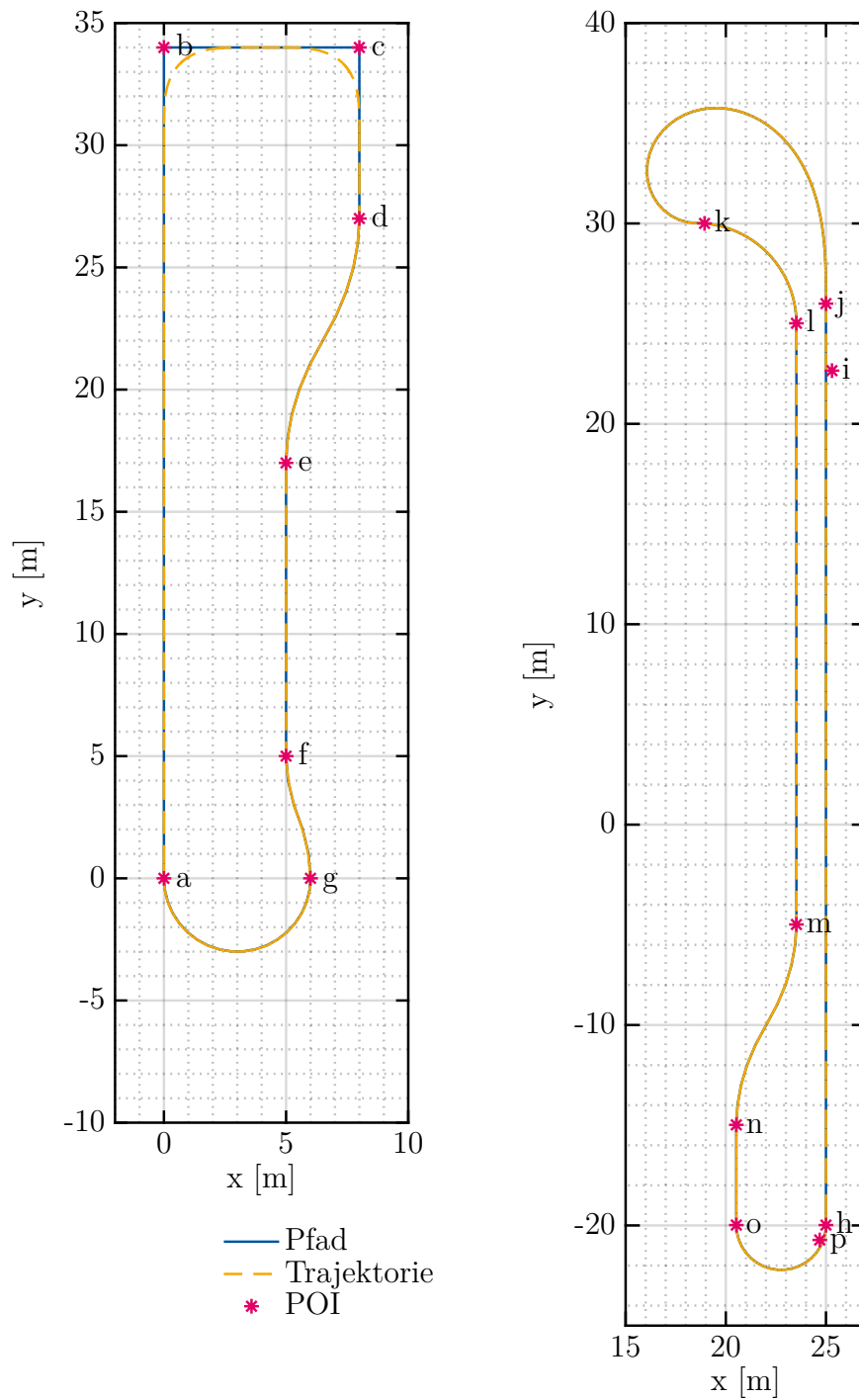


Abb. 6.1: Übersicht der beiden Teststrecken auf dem Parkdeck in lokalen Koordinaten mit POI; Strecke A links und Strecke B rechts

POI von Strecke A			POI von Strecke B		
	<i>Distanz</i>	<i>Beschreibung</i>		<i>Distanz</i>	<i>Beschreibung</i>
<i>a</i>	0,00 m	Startpunkt, Ende Kreis	<i>h</i>	0,00 m	Startpunkt, Ende Kreis
<i>b</i>	34,00 m	1. 90°-Biegung	<i>i</i>	42,64 m	Ende Ungenauigkeit
<i>c</i>	42,00 m	2. 90°-Biegung	<i>j</i>	45,98 m	Anfang Klothoide
<i>d</i>	49,00 m	Anfang Versatz	<i>k</i>	68,06 m	Übergang Klothoide, Kreis
<i>e</i>	59,57 m	Ende Versatz	<i>l</i>	75,50 m	Ende Kreis
<i>f</i>	71,57 m	Anfang Versatz	<i>m</i>	105,50 m	Anfang Versatz
<i>g</i>	76,69 m	Übergang Versatz, Kreis	<i>n</i>	116,07 m	Ende Versatz
<i>a</i>	86,11 m	Ende Kreis, Startpunkt	<i>o</i>	121,07 m	Anfang Kreis
			<i>p</i>	127,25 m	Anfang Ungenauigkeit
			<i>h</i>	128,09 m	Ende Kreis, Startpunkt

Tabelle 6.1: POI der Strecken A und B mit ihren kartenbasierten Distanzen zum jeweiligen Streckenstartpunkt

linksherum durchfahren. Die Strecke B enthält eine Klothoide, die von einer Geraden, beginnend im Startpunkt (POI *h*), zu einem Radius von knapp 4,80 m übergeht. Am Ende der Klothoide folgt ein viertel Kreis mit gleichem Radius in entgegengesetzter Richtung. Um die Strecke B zu schließen, folgen nach einer weiteren Geraden ein Versatz und ein Halbkreis mit einem Radius von etwa 2,25 m.

In Abbildung 6.1 ist neben den Kartendaten und dem daraus entstehenden Pfad (blau) ist zusätzlich die sich für die FBFR ergebende Referenztrajektorie (gelb) abgebildet. Um an diese Referenz zu gelangen, werden die Parameter für das Experiment berücksichtigt. Diese lauten für den FBFR bzw. dessen Trajektoriengenerator:

$$\begin{aligned}
 K_0 &= 1 \\
 K_1 &= 1 \\
 v_{soll} &= 2,2425 \text{ m/s} \hat{=} 2 \text{ U/s} \\
 v_{max} &= 3 \text{ m/s}
 \end{aligned}$$

Dabei bezeichnet v_{max} die Geschwindigkeit, die zur Auslegung der Referenztrajektorie verwendet wird. Sie stellt also diejenige Geschwindigkeit dar, der ein Buggy mit der FBFR noch folgen kann, da hierbei die verwendeten Designgrenzen der Regelung ausgenutzt werden, vgl. Abschnitt 4.3.4. Die Parameter für die Pfadregelung werden wie in Kapitel 2.9 angegeben gewählt, wobei die Sollgeschwindigkeit v_{soll} für beide Regelungskonzepte gleich ist.

Zum Zeitpunkt der Durchführung der Experimente ist die Trajektoriengenerierung der FBFR noch fehlerhaft. Die vorgegebene Position liegt aufgrund eines systematischen Fehlers immer etwa 7 cm vor der eigentlichen Position. Daher gibt es einen Positionsfehler, der permanent im Folgeregler auftritt. Der Fehler sorgt dafür, dass der Buggy beispielsweise zu früh in eine Kurve einlenkt. In den aufgezeichneten Daten sind nur die GNSS-Positionen, die mit einer Rate von 5 Hz vorliegen, enthalten. Gemeinsam mit

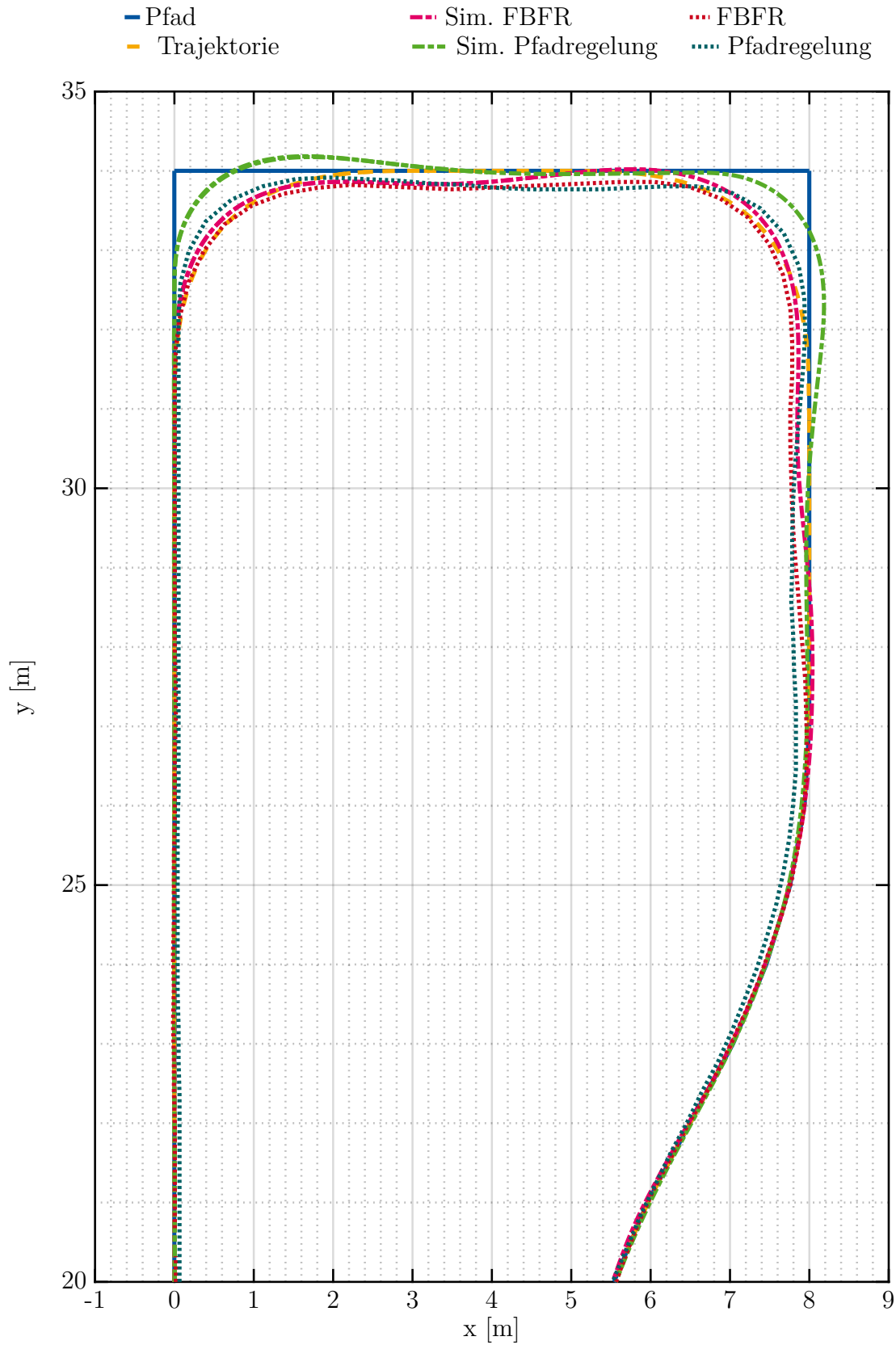


Abb. 6.2: Simulations- und Messdaten bei Abfahrt des oberen Teils der Strecke A

der gewählten Geschwindigkeit von 2,2425 m/s ergibt sich daraus, dass im Fall der realen Versuche etwa alle 45 cm einen Messwert für die Position vorliegt. Aufgrund dieser Auflösung kann es schwer fallen, den zuvor beschriebenen systematischen Fehler in den Abbildungen in diesem Abschnitt zu beobachten. Im Vergleich hierzu liegen die Daten aus den Simulationen deutlich präziser vor. In den Abbildungen 6.2, 6.3, 6.4 und 6.7 sind die Positionsverläufe beider Regelungsverfahren jeweils für die Simulation und für das reale Experiment in den interessantesten Regionen aus Abbildung 6.1 dargestellt.

Um eine bessere Vergleichbarkeit zwischen den einzelnen Untersuchungen zu ermöglichen, werden in den weiteren Abbildungen in diesem Abschnitt alle Werte über dem Ort, genauer der Strecke, aufgetragen. Die Strecke gibt an, wie weit der aktuell betrachtete Map Matching-Punkt kartenbasiert vom Startpunkt der entsprechenden Strecke entfernt ist. Sie kann also sehr leicht aus den Gleiskoordinatendaten, die zu den aufgezeichneten Daten gehören, bestimmt werden.

Abbildung 6.2 zeigt den oberen Teil der Strecke A. Er enthält die POI b und c , die jeweils einen 90° Übergang im Pfad markieren. Der Trajektoriengenerator erzeugt daraus Verbundkurven, mit denen die Übergänge für die FBFR realisiert werden. Der Vergleich von Simulation und Experiment der FBFR zeigt, dass in beiden Fällen etwa am gleichen Ort von der Geraden in die Verbundkurve um POI b übergegangen wird. Dabei ist zu beobachten, dass in der Simulation später und weniger stark eingelenkt wird. Dies könnte an dem vorher beschriebenen systematischen Fehler bei der Implementierung der FBFR liegen. Dagegen spricht, dass auch bei der Pfadregelung dasselbe Phänomen auftritt. Auch hier wird die Gerade in der Simulation später verlassen als im Experiment. Der Unterschied von Simulation zu Experiment für beide Verfahren verhält sich ähnlich. In beiden Fällen reagiert das Fahrzeug in der Realität früher als in der Simulation. Dabei ist der Unterschied bei der Pfadregelung besonders auffällig. Hier wird in der Simulation im niedrigen zweistelligen cm-Bereich später mit der Lenkbewegung begonnen als im Experiment. Da die Pfadregelung mit 10 Hz Steuerbefehle generiert, folgt daraus, dass bei der gewählten Geschwindigkeit von $v = 2,2425$ m/s etwa alle 22,4 cm eine neue Lenkvorgabe zustande kommt. Erfolgt daher die Abtastung zu einem ungünstigen Zeitpunkt, könnte dies die verzögerte Reaktion erklären. Im Fall des zweiten 90° Übergangs an POI c tritt dieser Versatz erneut auf.

Nur bei der Simulation der Pfadregelung kommt es dazu, dass der Buggy über die nächste Gerade hinaus fährt. Bei allen anderen Messwerten lenkt der Buggy stärker ein, sodass er parallel zu der nächsten Geraden auskommt. In der Simulation wird der Versatz nach dem POI b von beiden Verfahren noch vor POI c wieder ausgeglichen. Im realen Experiment geschieht dies nicht mehr rechtzeitig, wobei eine leichte Tendenz zum Ausgleich zu erkennen ist und sich diese bei der FBFR ausgeprägter darstellt. Hinter dem POI c ergibt sich ein ähnliches Bild. Da die Gerade zum nächsten Streckenabschnitt länger ist, kann auch das Ausregeln der Abweichung beobachtet werden. In der Simulation regelt die Pfadregelung schneller als die FBFR, wobei diese als einzige auf der östlichen Seite der Geraden beginnt. Bei den realen Messungen erreicht die FBFR die Gerade früher auf die Pfadregelung. Dabei ist sie jedoch später auf der Geraden als die entsprechende Simulation.

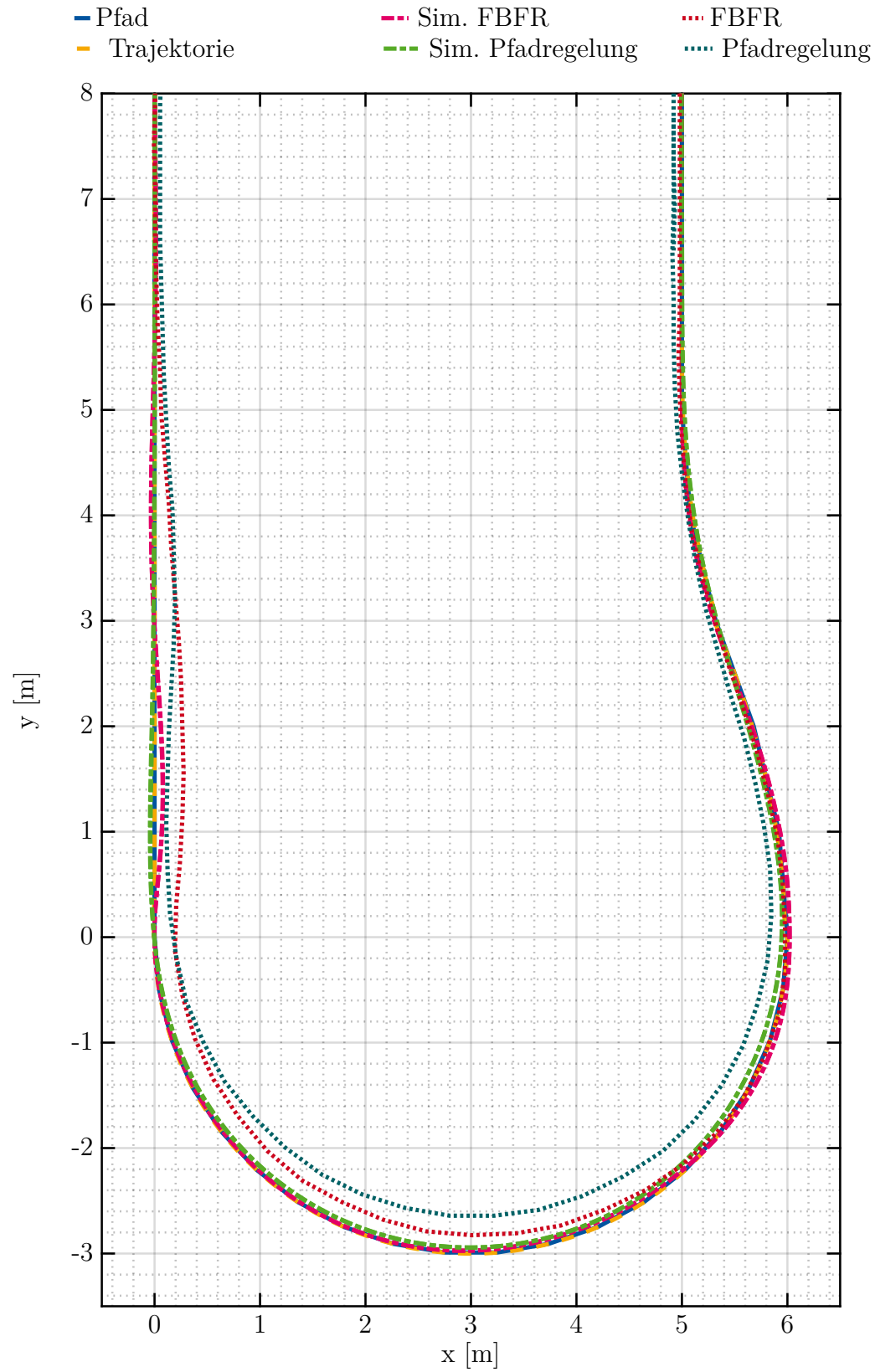


Abb. 6.3: Simulations- und Messdaten bei Abfahrt des unteren Teils der Strecke A

Aus den beschriebenen Abweichungen zwischen Simulation und Realität kann abgelesen werden, dass es bei der Modellierung der Lenkung Ungenauigkeiten gibt. Bei der langsameren Pfadregelung scheinen diese Ungenauigkeiten deutlicher auszufallen als bei der FBFR. Die betrachteten POI b und c sind Extremsituationen, in denen der Buggy mit nahezu vollem Lenkeinschlag durch die Kurve fährt. Dies entspricht einem Kurvenradius von etwa 1,3 m. Im Vergleich hierzu kann in Abbildung 6.3 bei der Kreisfahrt zwischen den POI g und a beobachtet werden, dass die Simulationen der Solltrajektorie deutlich besser folgen können. Der Halbkreis besitzt einen Radius von 3 m. Es ist aber dennoch zu beobachten, dass in den Experimenten erneut engere Kreise gefahren werden als in der Simulation. Eine mögliche Erklärung könnte die Montageposition der GNSS-Antenne auf dem Fahrzeug sein. Beide Regler sind so modelliert, dass der Messpunkt am Hinterachsmittelpunkt liegt. In der Realität befindet sich an dieser Stelle aber das Display des Buggys, sodass die Antenne am nächsten Punkt auf dem Rahmen montiert ist. Dadurch liegt der Messpunkt zwar immer noch in der Fahrzeugmitte bezogen auf die Querachse, aber etwa 5 cm hinter der Hinterachse.

Am POI a ist zu erkennen, dass sich die beiden Regler unterschiedlich verhalten. Die Pfadregelung kann durch ihre „Sicht“ auf den kommenden Streckenabschnitt besser auf den direkten Übergang von einem Kreis in eine Gerade reagieren. Der Buggy beginnt schon vor dem Übergang, die Lenkung gerade zu stellen. Dadurch bleibt er dichter an der Geraden und weist nur einen geringen Versatz in westlicher Richtung zu der Geraden auf. Dieser wird anschließend ausgeregelt. Auch bei dem realen Experiment mit der Pfadregelung ist dieses Ausregeln zu erkennen, auch wenn östlich von der Geraden ausgekommen wird. Die FBFR verhält sich wie zu erwarten anders an dieser Stelle. Vom Trajektoriengenerator wird bis zum POI a eine Kreisfahrt vorgegeben. An dem POI a gibt es dann einen unstetigen Übergang in eine Gerade. Der damit einhergehenden Lenkwinkeländerung kann nicht instantan gefolgt werden. Daher fährt der Buggy weiterhin eine Rechtskurve. Diese wird zunehmend weiter, sodass der Buggy bei diesem Regler mit einem östlichen Versatz parallel zur Geraden fährt. Dieser Versatz wird durch die FBFR nachträglich ausgeregelt, sowohl in der Simulation als auch im Experiment. Wobei auch hier im Experiment ein größerer Versatz aufgetreten ist als in der Simulation.

In Abbildung 6.4 ist der obere Teil der Strecke B abgebildet. Er enthält die POI i bis l . Dabei nimmt der POI i eine gesonderte Rolle ein. Anders als bei den anderen POI liegt dort keine Änderung der Topologie der Strecke vor. Er gibt das Ende von Messungenauigkeiten im Experiment der FBFR an. Während der Experimente werden Korrekturdaten von SAPOS[®] verwendet, um mittels RTK eine möglichst genaue Messung zu erreichen. Die hohe Positionsgenauigkeit wird im Fall der FBFR nicht durchgehend erreicht. Der POI i markiert die Stelle, ab der die Positionsgenauigkeit wieder im niedrigen cm-Bereich liegt.

Die Abbildungen 6.5 und 6.6 zeigen die Positionsgenauigkeiten für die Strecken A und B während der realen Experimente. Diese werden aus der GST-Nachricht des Septentrio GNSS-Empfängers gewonnen. Dazu wird die Wurzel aus der Summe der quadrierten longitudinalen und lateralen 1 Sigma Fehler gezogen. Für die Strecke A werden für beide Regelungskonzepte dank der Verwendung von RTK sehr gute Genauigkeiten im Bereich

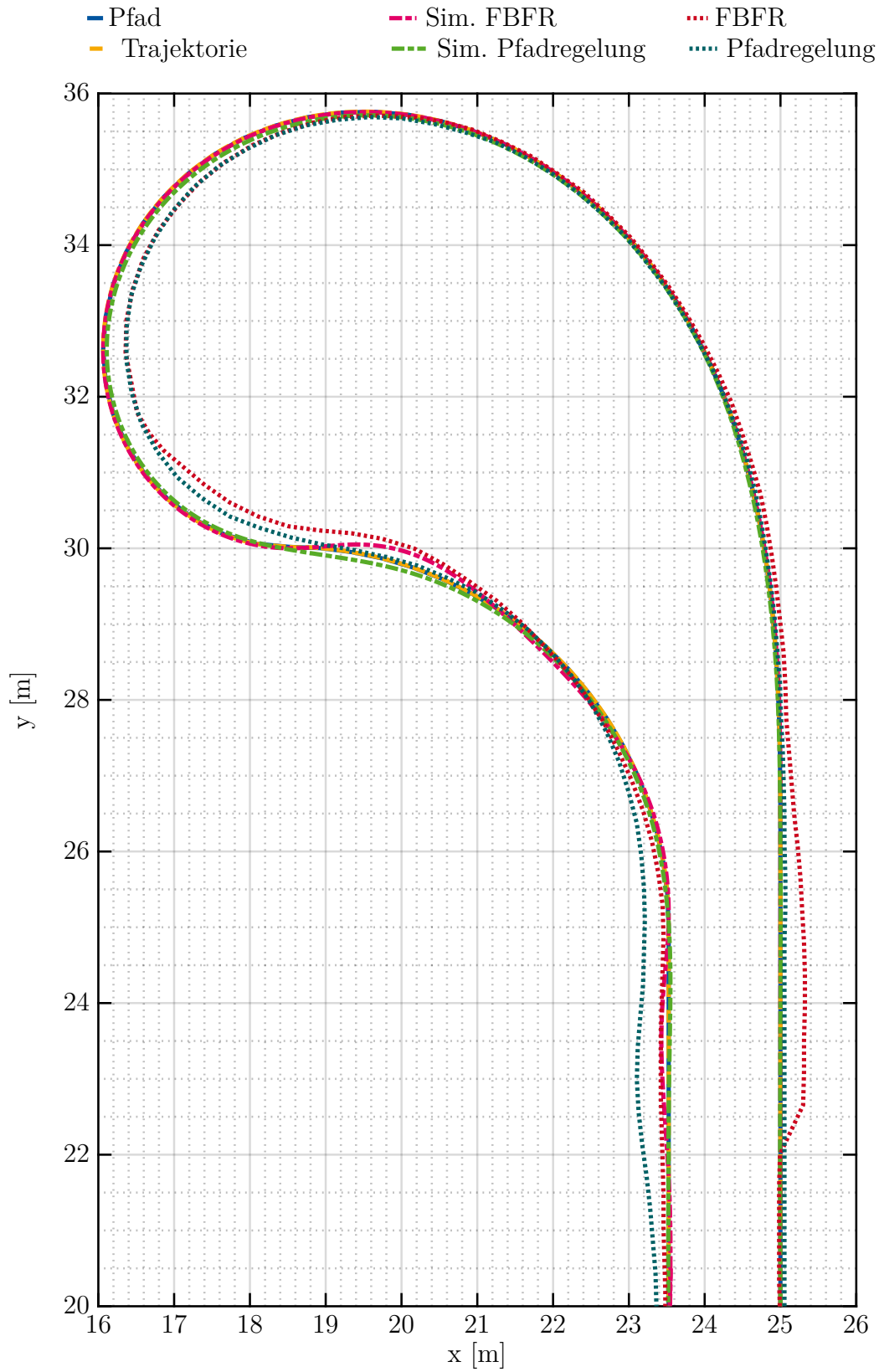


Abb. 6.4: Simulations- und Messdaten bei Abfahrt des oberen Teils der Strecke B

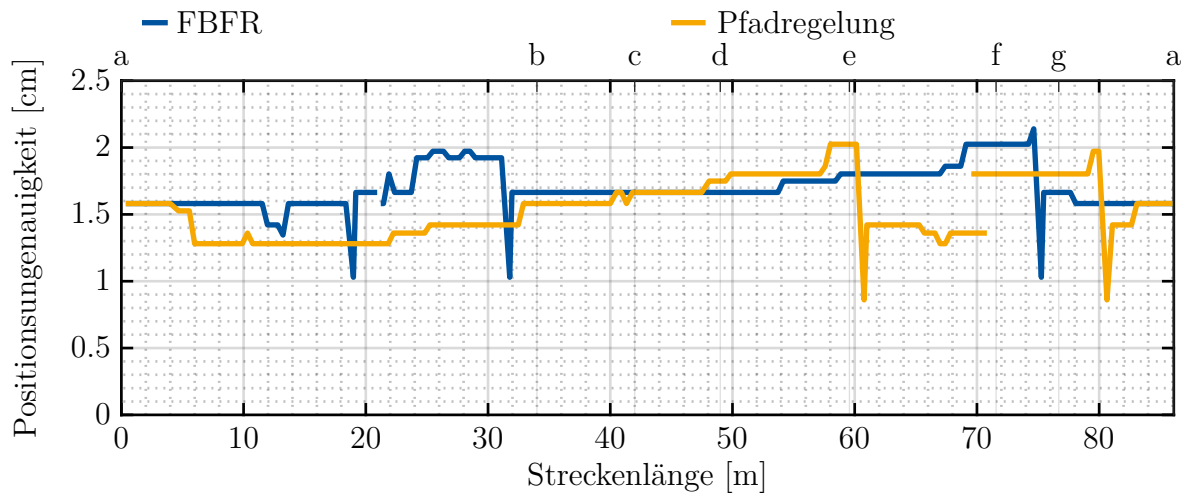


Abb. 6.5: GNSS-Positionsgenauigkeit während der Fahrt auf Strecke A

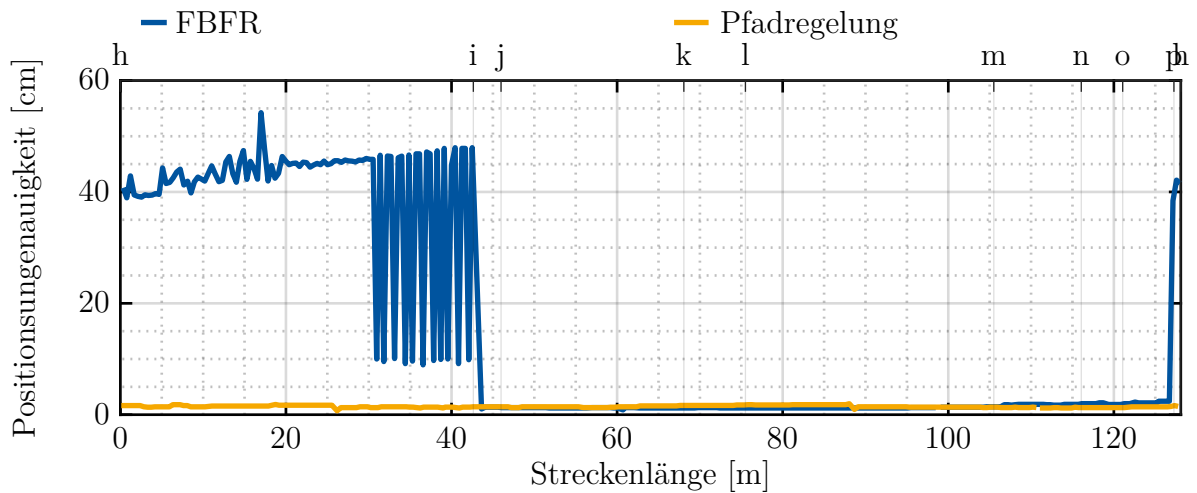


Abb. 6.6: GNSS-Positionsgenauigkeit während der Fahrt auf Strecke B

um 2 cm erreicht, siehe Abbildung 6.5. Beim Durchfahren der Strecke B gibt es, wie Abbildung 6.6 zeigt, im Fall der FBFR Messwerte, die deutlich ungenauer sind und Abweichungen von bis zu etwa 55 cm annehmen können. Diese Ungenauigkeiten beginnen in dem Punkt p , erstrecken sich über den Startpunkt h hinweg und enden im Punkt i . Die niedrigen Messwerte in Abbildung 6.6 haben dasselbe Niveau wie die Messwerte in Abbildung 6.5. Aufgrund der y-Skalierung ist dies in Abbildung 6.6 schwer ersichtlich. Die Messung der FBFR erreicht dieses Niveau von etwa 2 cm nur im Bereich zwischen den POI i und p .

Da keine andere Messmethode für die Positionsbestimmung zur Verfügung steht, kann der Fehler weder von den Regelungssystemen noch in der Nachbearbeitung erkannt oder gar kompensiert werden. Für die Systeme gelten die gemessenen Positionen als Absolutwerte, auch wenn diese durch Messungenauigkeiten von den tatsächlichen Positionen abweichen. Bei der Betrachtung der Abbildung 6.7 im Bereich um den Punkt p , der etwa

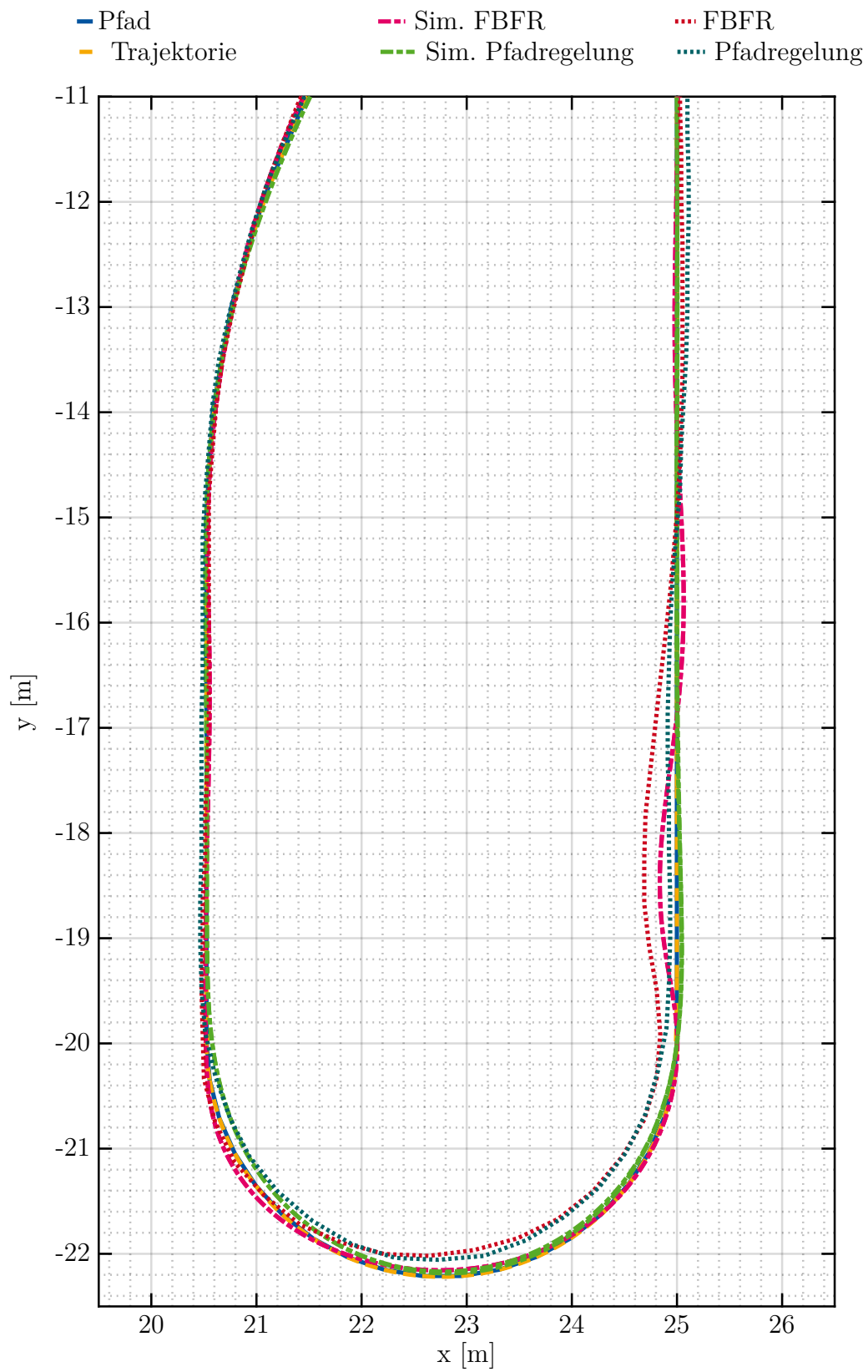


Abb. 6.7: Simulations- und Messdaten bei Abfahrt des unteren Teils der Strecke *B*

bei den Koordinaten (25 m, -20 m) liegt, fällt keine sprunghafte Veränderung bei der FBFR auf. Es gibt zwar einen Knick im Positionsverlauf, dieser liegt aber im Bereich der Erwartungen der Simulation an dieser Stelle und lässt sich durch den unstetigen Übergang von dem Kreisbogen in die Gerade erklären. Um von der Kreisfahrt in die Geradeausfahrt zu wechseln, wird etwas Zeit bzw. Weg benötigt, was zu einer Abweichung führt, die danach ausgeregelt werden muss. Dies scheint für den in Abbildung 6.7 dargestellten Fall auch zu gelingen. Im weiteren Verlauf der Geraden, der weder in Abbildung 6.7 noch Abbildung 6.4 gezeigt ist, kommt es zu Versätzen der Position um bis zu 10 cm. Am Punkt i , wenn die Positionsgenauigkeit wieder zunimmt, gibt es einen Sprung in der Position, der in Abbildung 6.4 durch einen Versatz um nahezu 40 cm nach rechts zu erkennen ist. Es scheint also, als driftet die Position im Bereich der schlechteren Empfangsbedingungen immer weiter ab. Der Positionsversatz wird wieder ausgeregelt, was aber nicht gänzlich vor dem Übergang in die Klothoide an POI j gelingt.

In der Klothoide selbst kann, ähnlich wie schon zuvor im Kreis, beobachtet werden, dass sowohl Simulation und Experiment dem Pfad folgen können. Außerdem kann erneut beobachtet werden, dass die Messwerte im Experiment wieder engere Kreise fahren als von der Simulation vorhergesagt. Am POI k , wo von einer Linkskurve direkt in eine Rechtskurve übergegangen wird, können erneut die Charakteristika der beiden Regelungen beobachtet werden. Dank der Prädiktion der Pfadregelung wird durch diese, schon vor dem POI k , die Lenkbewegung in entgegengesetzter Richtung einleiten. Dadurch folgt die Pfadregelung, in Simulation und Experiment, der Streckenführung besser. Bei der FBFR hingegen wird erst ab dem Übergang im Punkt k festgestellt, dass sich die Kurvenrichtung geändert hat. Daher beginnt die Gegenlenkbewegung erst ab diesem Punkt. Dasselbe Verhalten kann auch im POI l beobachtet werden. Dabei liegen Simulation und Experiment der FBFR deutlich dichter übereinander als an den bisher betrachteten Stellen. Dies kommt von dem Versatz, der am POI k zwischen Simulation und Experiment vorliegt und im Experiment nicht so schnell abgebaut wird. Die Pfadregelung zeigt auch hier, im Experiment, das bisher beobachtete, stärkere Einlenken, was zu einer Abweichung am Kreisende führt. Im weiteren Verlauf wird auch diese ausgeregelt.

In Abbildung 6.7 ist der untere Teil der Strecke B abgebildet. Sie enthält den POI p , ab dem die Messung der FBFR im Experiment ungenau wird. Außerdem ist ein Kreisabschnitt mit einem Radius von etwa 2,25 m zwischen den POI o und h abgebildet. Die Regler verhalten sich, auch in diesem Abschnitt, wie bereits in den anderen betrachteten Abschnitten. Es ist zu beobachten, dass hinter dem Ende des Kreisabschnittes eine größere Abweichung vom Pfad vorliegt als bei dem Kreisabschnitt in Strecke A zwischen POI g und a . In den Abbildungen 6.8 und 6.9 kann dies besser abgelesen werden, da dort der Abstand zum nächsten Punkt auf dem Pfad aufgetragen ist. Die höheren Werte für Strecke B lassen sich aber mit dem kleineren Kreisradius erklären. Um die Lenkung ab dem POI h wieder gerade zu stellen wird mehr Zeit bzw. Weg benötigt, was zu einer größeren Abweichung führt.

Bei weiterer Betrachtung der Abbildungen 6.8 und 6.9 ist zu erkennen, dass die Pfadregelung sowohl in der Simulation als auch im Experiment größere Abweichungen vom

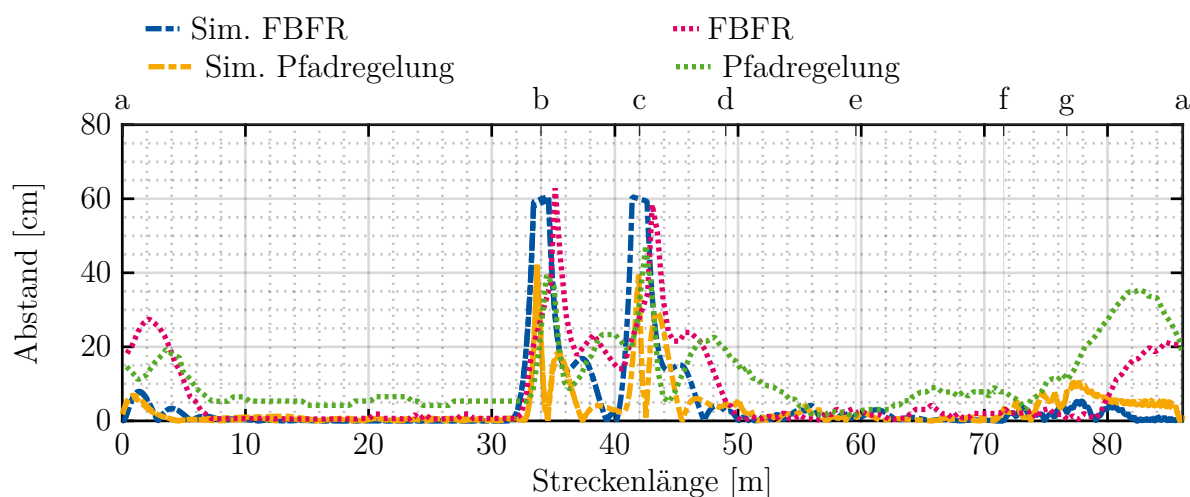


Abb. 6.8: Abstand zum Pfad auf der Strecke A

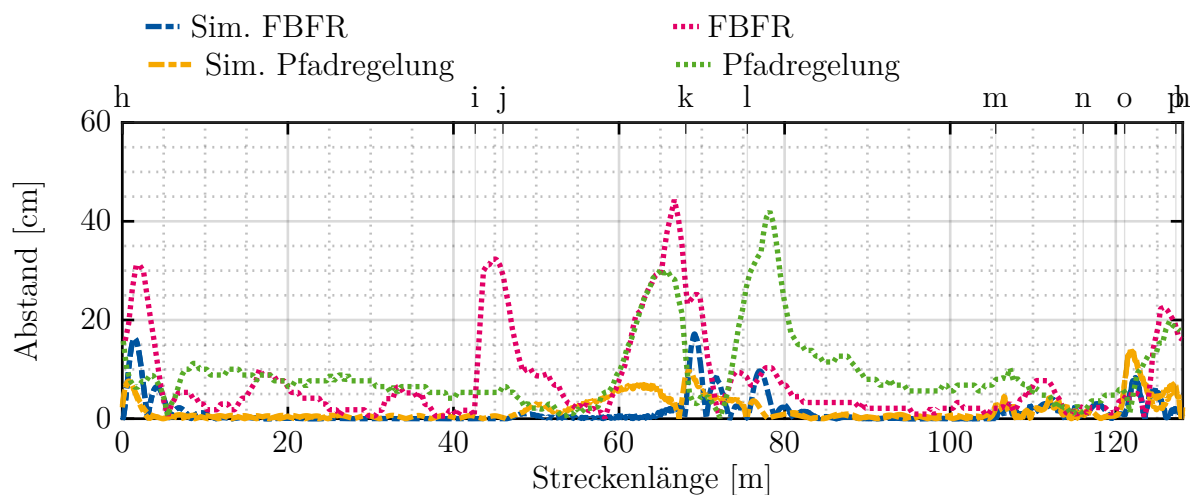


Abb. 6.9: Abstand zum Pfad auf der Strecke B

Pfad aufweist. Auszunehmen sind dabei die zuvor diskutierten Stellen, an denen der Aufbau der FBFR zu designbedingten, größeren Abweichungen führt. Dies sind die Punkte hinter unstetigen Übergängen der Referenztrajektorie (die POI a , b , c , h und k) sowie, im Falle des Experiments, der POI i , an dem die Messgenauigkeit wieder angestiegen ist. Die Abweichung dort tritt, wie in Abbildung 6.9 zu erkennen ist, sehr plötzlich ein. Ferner kann in Abbildung 6.8 beobachtet werden, dass um die 90° Übergänge herum (POI b und c) Sprünge in den Verläufen auftauchen. Dies ist besonders gut im Verlauf der FBFR-Simulation zu erkennen. Die Erklärung hierfür ist die Wahl der Streckenlänge als x -Achse. An den POI b und c wird der Map Matching-Punkt gewisse Abschnitte des Pfades nie erreichen. Dies kann anhand von Abbildung 4.13 auf Seite 81 deutlich gemacht werden. Die dort in rot eingezeichneten Abschnitte des Weges W_1 fallen auf den roten Punkt in der Referenztrajektorie y_d . In dem hier vorliegenden Fall wird durch das Map Matching umgekehrt projiziert. Von der tatsächlichen Position wird auf den

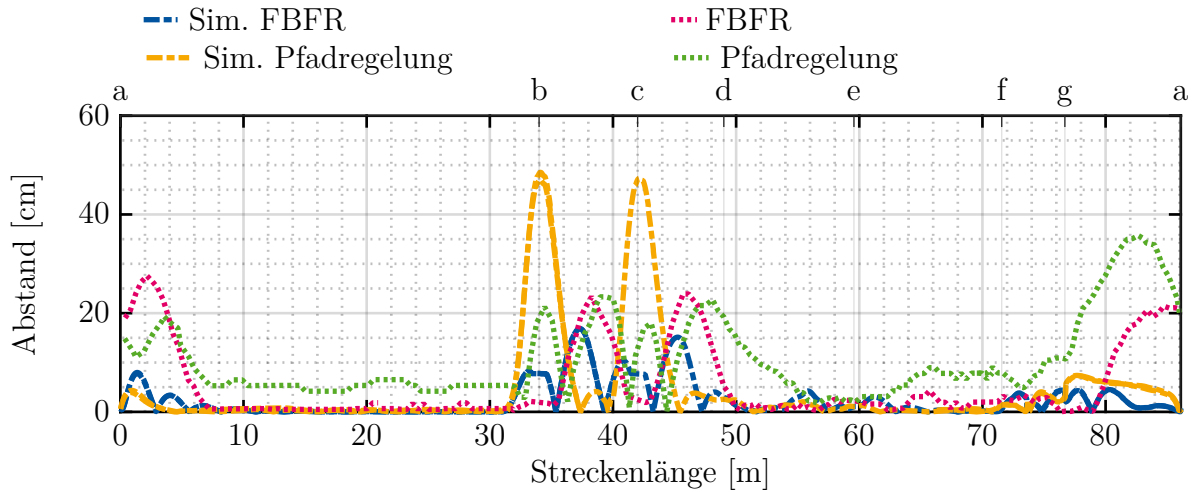


Abb. 6.10: Abstand zur FBFR-Referenz auf der Strecke A

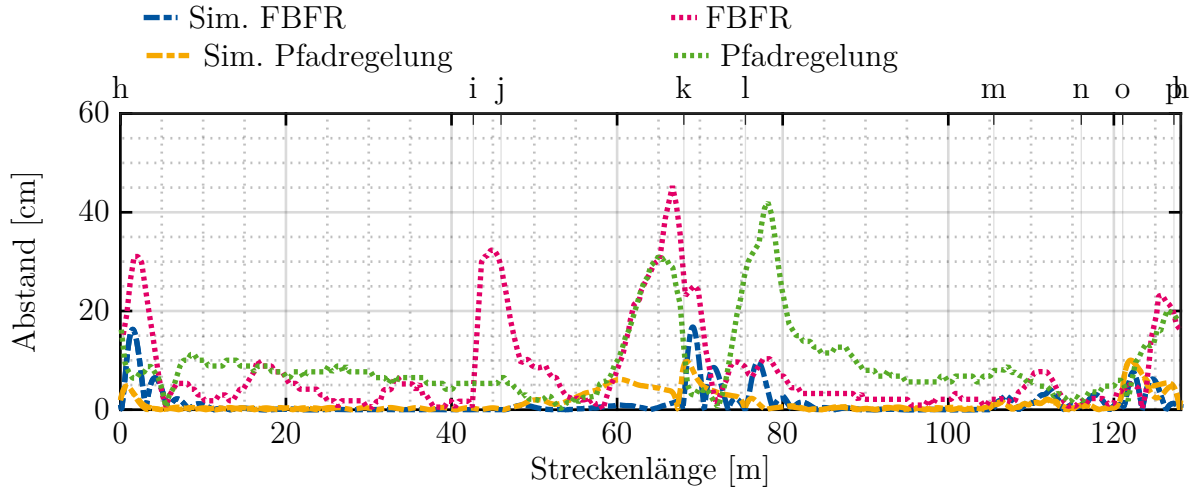


Abb. 6.11: Abstand zur FBFR-Referenz auf der Strecke B

Weg abgebildet. Dabei enden Positionen links unterhalb der Winkelhalbierenden WH , in Abbildung 4.13, auf dem senkrechten Teil des Wegs W_1 . Positionen rechts oberhalb der Winkelhalbierenden gelangen auf den waagerechten Teil von W_1 . Beim Überqueren der Winkelhalbierenden WH findet also ein Sprung der Map Matching-Position statt, der sich in Abbildung 6.8 an den POI b und c erkennen lässt.

In den Abbildungen 6.10 und 6.11 ist der Abstand zu der vom Trajektoriengenerator generierten Referenz abgebildet. Diese stellt besonders im Verlauf um die POI b und c eine realistischere Referenz dar, wenngleich der Vergleich zwischen den Verläufen der Pfadregelung und dieser Referenz nicht ganz fair ist. Der Pfadregelung liegen zu keiner Zeit Informationen über die Trajektorie vor. Im Fall der FBFR sind die Abbildungen 6.10 und 6.11 aber aufschlussreich, da sie die Abweichung von der tatsächlich in dem Regler verwendeten Referenz darstellen. Für die Strecke A ist zu erkennen, dass die Abweichungen immer wesentlich unter 5 cm betragen. Dabei sind die kritischen Stellen (90° Übergänge

und Kreisbogen) außen vorgelassen. An diesen Stellen treten größere Abweichungen auf. Ferner ist zu erkennen, dass in der Simulation größere Abweichungen bei den 90° Übergängen auftreten als am Kreisbogen. Im Experiment zeigt sich dies genau andersherum. In 6.11 kann im Bereich der Empfangsbeeinträchtigung zwischen POI p und i beobachtet werden, wie es auf der Geraden immer wieder zu Abweichungen kommt. Der schlechtere Empfang spiegelt sich also in einer schlechteren Querführung wider.

Die in diesem Abschnitt durchgeführten Untersuchungen zeigen, dass die Messwerte von Simulation und Experiment leicht voneinander abweichen. Qualitativ sind sie jedoch miteinander vergleichbar. Für beide Regelungen zeigen die entsprechenden Simulationen und Experimente die selben Charakteristika. Ein Beispiel hierfür ist die zunächst weiter anwachsende Abweichung vom Pfad nach einem unstetigen Übergang bei Verwendung der FBFR. Das Verhalten kann in Simulation und Experiment gleichermaßen beobachtet werden. Damit ist die vorgestellte Simulationsumgebung als Entwicklungswerkzeug geeignet. Die mit ihr bestimmten Ergebnisse eignen sich als Abschätzung der zu erwartenden Verläufe im realen Experiment.

Fazit: Mit den erzielten Ergebnissen kann einerseits gezeigt werden, dass die Simulation eine sinnvolle Voruntersuchung der Regelkonzepte ermöglicht. Andererseits konnte mit den durchgeführten Experimenten gezeigt werden, dass die vorgestellten Regelungsverfahren zur Realisierung der virtuellen Schiene geeignet sind. Beide Verfahren erzielten gute Ergebnisse.

6.2 Auslegung der Flacheitsbasierten Folgeregelung

Im Folgenden wird die Simulationsumgebung verwendet, um den Einfluss der Parameter K_0 und K_1 auf die FBFR zu untersuchen. Dazu wird die Strecke A auf dem Parkhaus genutzt. Dem Trajektoriengenerator wird eine Geschwindigkeit von $v_{max} = 3 \text{ m/s}$ vorgegeben. Anschließend fährt der Buggy die Strecke in der Simulation mit einer Geschwindigkeit $v_{soll} = 2,2425 \text{ m/s}$ mehrfach ab. Bei jeder Fahrt werden andere Parameterkombinationen K_0 und K_1 verwendet. Dabei gilt $K_0 \in \{0.1, 0.2, 0.5, 1\}$ und $K_1 \in \{0.5, 1\}$. Die Abbildungen 6.12 und 6.13 zeigen die Abstände zum Referenzpfad und die Abweichung von der Sollgeschwindigkeit. Dabei werden die Streckenabschnitte 62 und 70 betrachtet, vgl. Abbildung 5.4 auf Seite 95, die die Gerade zwischen den POI a und b bilden. Der Buggy vollzieht also eine Kreisfahrt und biegt zu Beginn auf die Gerade ein. Dabei ändert sich die Referenz unstetig und die FBFR muss die Lenkung gerade stellen sowie die dabei entstehende Abweichung ausregeln.

In den Abbildungen 6.12 und 6.13 ist zu erkennen, dass sich der Positionsfehler wie ein Verzögerungsglied 2. Ordnung verhält. Genau dies entspricht auch der Erwartung. In Abschnitt 4.3.3 wurde die Folgeregelung genau auf dieses Verhalten ausgelegt, vgl.

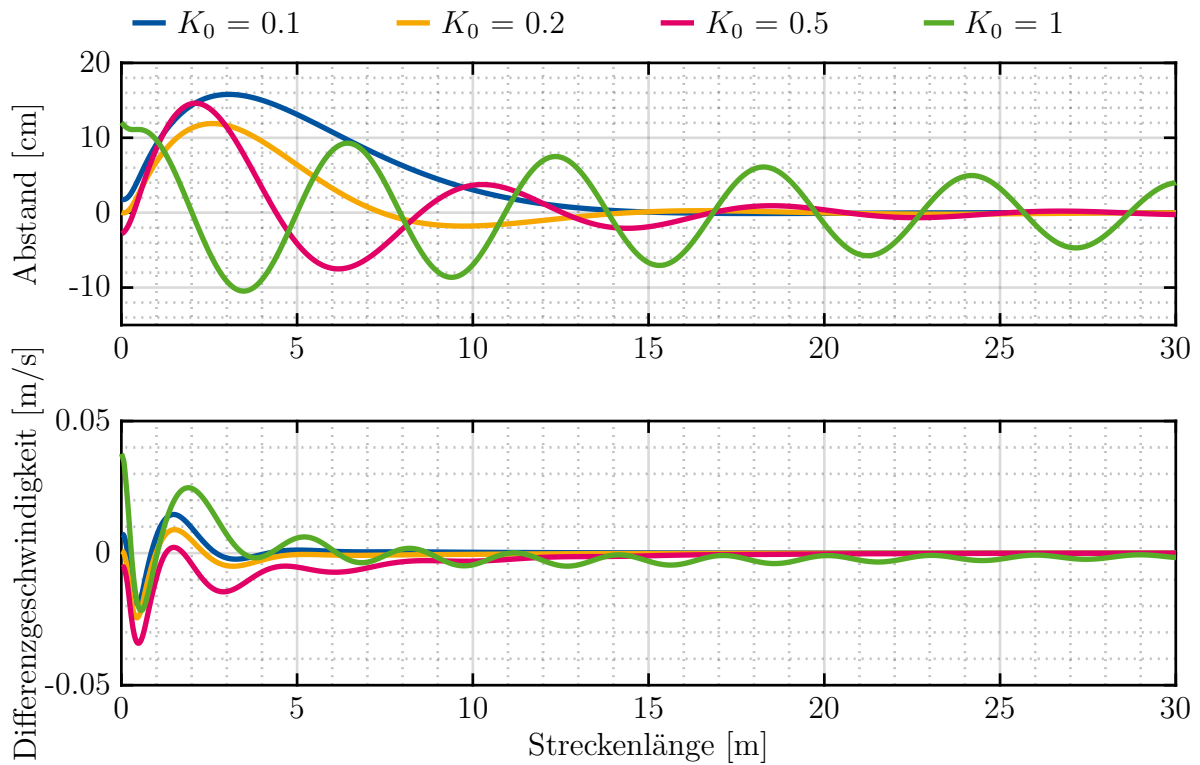


Abb. 6.12: Einfluss der Parameter K_0 und K_1 auf das Regelverhalten mit $K_1 = 0,5$

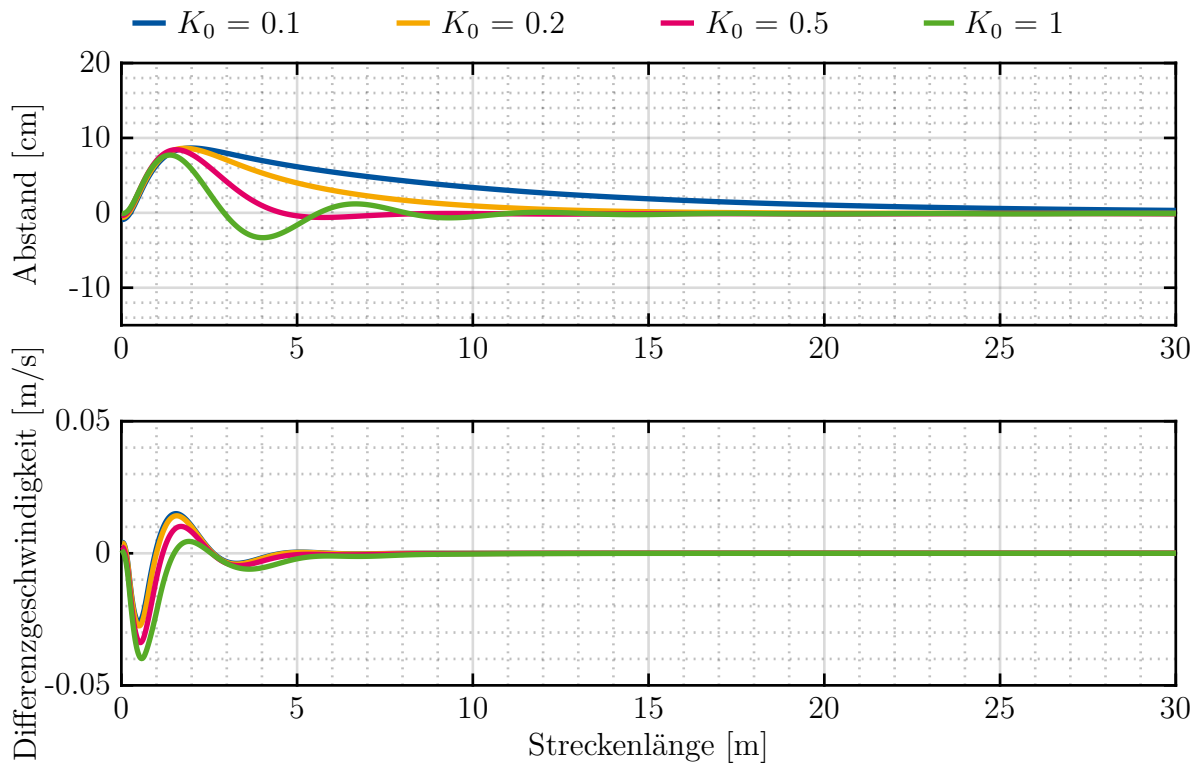


Abb. 6.13: Einfluss der Parameter K_0 und K_1 auf das Regelverhalten mit $K_1 = 1$

DGL (4.55). Durch einen Koeffizienten Vergleich der DGL eines Verzögerungsgliedes 2. Ordnung

$$\ddot{\mathbf{y}} + 2 \cdot \delta \cdot \omega_0 \cdot \dot{\mathbf{y}} + \omega_0^2 \cdot \mathbf{y} = \mathbf{u} \quad (6.1)$$

mit der Gleichung (4.55) können die folgenden Zusammenhänge zwischen den Parametern K_0 sowie K_1 und der Kreisfrequenz ω_0 bzw. der Dämpfung δ aufgestellt werden:

$$\omega_0 = \sqrt{K_0} \quad (6.2)$$

$$\delta = \frac{K_1}{2 \cdot \omega_0} = \frac{K_1}{2 \cdot \sqrt{K_0}}. \quad (6.3)$$

Die Tabelle 6.2 listet die Kreisfrequenzen und Dämpfungen der in den Abbildungen 6.12 und 6.13 betrachteten Konstellationen auf.

K_0 [1/s ²]	0,1	0,2	0,5	1	0,1	0,2	0,5	1
K_1 [1/s]	0,5	0,5	0,5	0,5	1	1	1	1
ω_0 [Hz]	0,3162	0,4472	0,7071	1	0,3162	0,4472	0,7071	1
δ	0,7906	0,5590	0,3536	0,25	1,5811	1,1180	0,7071	0,5

Tabelle 6.2: Kreisfrequenzen ω_0 und Dämpfungen δ in Abhängigkeit von K_0 und K_1

Tabelle 6.2 kann entnommen werden, dass für die Parameter in Abbildung 6.12 die Dämpfung δ immer zwischen 0 und 1 liegt. Dies ist auch in den Verläufen der Abstände zur Referenz zu erkennen. Alle Verläufe schwingen. Mit wachsendem K_0 nimmt die Dämpfung δ weiter ab. Daher schwingt die Kurve für $K_0 = 1$ am wenigsten gedämpft. Die Dämpfung für diese Kurve ist sogar so gering, dass sie anders als die anderen Kurven schon zu Beginn eine Abweichung von mehr als 10 cm aufweist. Dies kommt daher, dass der Buggy vor dem betrachteten Abschnitt einen Halbkreis durchfährt und bereits dabei um die Referenz schwingt. In der Abbildung kann weiter betrachtet werden, dass auch um die Sollgeschwindigkeit geschwungen wird.

In Abbildung 6.13 kann beobachtet werden, dass bei zwei Parameterkonstellationen eine Dämpfung größer als 1 erreicht wird. Wie zu erwarten, findet für die beiden entsprechenden Kurven kein Überspringen statt, sondern der Abstand zur Referenz wird asymptotisch kleiner. Bei der Geschwindigkeit hingegen kommt es sehr wohl zu Schwingungen um die Sollgeschwindigkeit.

Fazit: Die FBFR weist das im Design festgelegte Regelverhalten eines Verzögerungsgliedes 2. Ordnung auf. Über ihre Parameter K_0 und K_1 kann das Verhalten beeinflusst werden. Es ist ein geeignetes Verfahren, da die Abweichung mit der Zeit bzw. dem Ort ausgeregelt wird. Dabei kann der Regler behäbiges Verhalten einer überkritischen Dämpfung aufweisen, aber auch so eingestellt werden, dass er sehr agil arbeitet, was aber Schwingungen mit sich bringt.

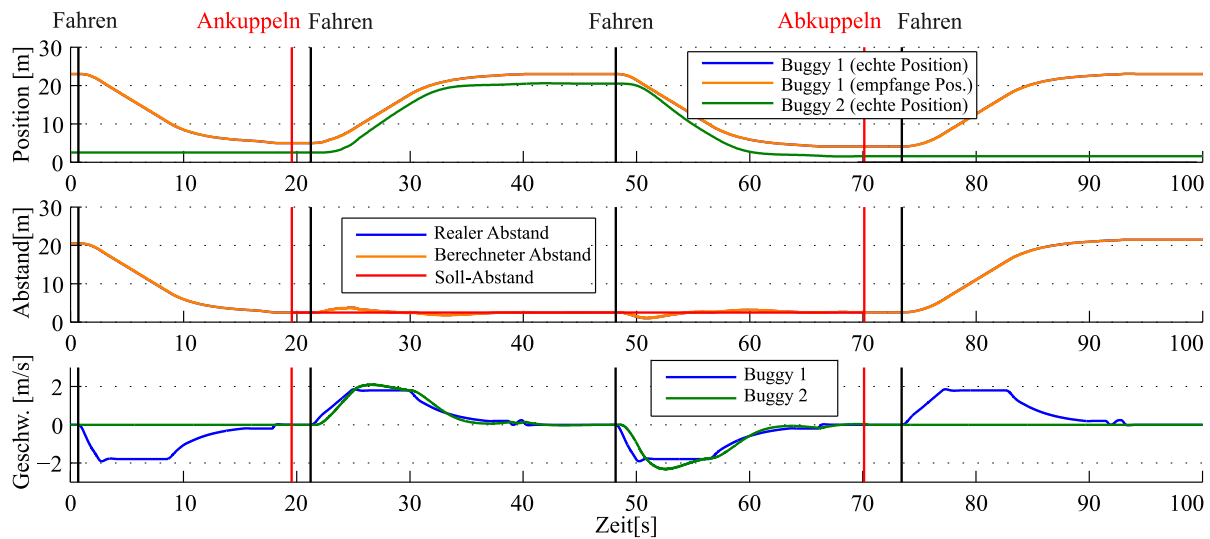


Abb. 6.14: HiL-Simulationsergebnisse der virtuellen Schiene (Position oben, Abstand Mitte und Geschwindigkeit unten) [46]

6.3 Untersuchung der virtuellen Kupplung

Bei der bisherigen Untersuchung lag der Fokus auf den beiden vorgestellten Querreglern. An dieser Stelle wird nun die Längsregelung untersucht. Während der Experimente wurde die Buggy interne Kommunikation mittels ZigBee-Modulen und nicht wie in Abschnitt 6.5 via LTE durchgeführt.

Als Teststrecke dient auch hier wieder die Strecke A. Wie in der Gleiskarte in Abbildung 5.4 auf Seite 95 dargestellt ist, gibt es eine Weiche in Strecke A. Von dem Weg 66 kann entweder auf den Weg 65 oder 111 abgebogen werden. An dieser Weiche wird ein minimales Rangierbeispiel durchgeführt. Buggy 1 dient als Lokomotive und startet auf Weg 66. Von dort aus soll der Buggy eine kuppelbereite Zielbremsung zum, durch Buggy 2 dargestellten, Wagen durchführen. Anschließend wird der Wagen gekuppelt und beide Buggys begeben sich, durch eine weitere Zielbremsung durchgeführt von Buggy 1, vor die Weiche. Diese wird virtuell umgestellt und es folgt eine dritte Zielbremsung, bei der das Folgefahrzeug dieses Mal „geschoben“ wird. Am Ziel auf Weg 112 angekommen wird der Wagen dort abgekuppelt und die Lok fährt zurück in ihre Ausgangsposition. Die im Folgenden vorgestellten Resultate stammen aus [46].

Auch die Untersuchung wird sowohl simulativ als auch im Experiment durchgeführt. Abbildung 6.14 zeigt dabei zunächst die Simulationsergebnisse, die mit beiden Buggys im HiL-Modus erzielt werden. Durch schwarze bzw. rote senkrechte Striche ist in der Abbildung eingezeichnet, zu welchem Zeitpunkt ein bestimmtes Manöver gestartet wurde. Die schwarzen Striche geben dabei den Beginn einer Zielbremsung an. Zwischen den roten Strichen sind die Buggys gekuppelt. Nur zu dieser Zeit ist der Kolonnenregler aktiv und besitzt eine Sollwertvorgabe, vgl. Abbildung 6.14 in der Mitte. Neben dem Sollabstand ist auch der errechnete Abstand, der in die Fehlerbestimmung im Regler eingeht, aufgetra-

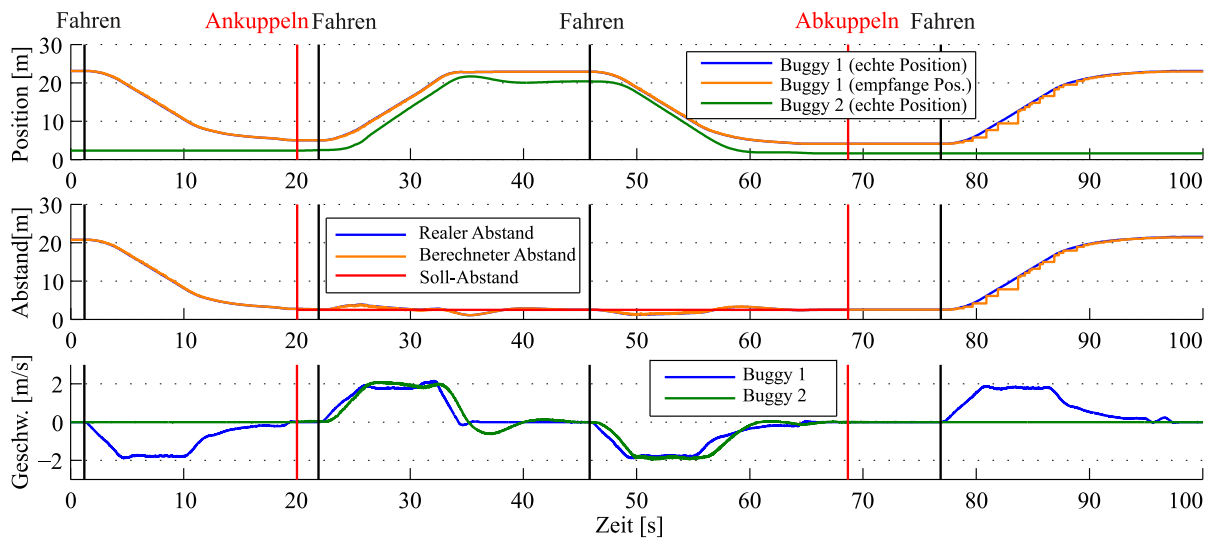


Abb. 6.15: Experimentell erzielte Ergebnisse der virtuellen Schiene (Position oben, Abstand Mitte und Geschwindigkeit unten) [46]

gen und zusätzlich der tatsächliche Abstand. Aufgrund von Kommunikationsstörungen können diese voneinander abweichen.

Im oberen Teil der Abbildung 6.14 kann der Positionsverlauf beider Buggys betrachtet werden. Er wird in Gleiskoordinaten bezogen auf den Abstand zum POI c bzw. zum Startpunkt des Weges 112 berechnet. Dieser liegt zwar auf dem Weg 63, ist aber kein Punkt des Weges und auch sonst nicht verknüpft. In Abbildung 6.14 ist auch die von Buggy 1 an Buggy 2 mittels Kommunikation übermittelte Position aufgeführt. Im unteren Teil sind die Geschwindigkeiten beider Buggys abgebildet. In ihren Verläufen kann das sich durch die Zielbremsungen ergebende Geschwindigkeitsprofil beim Anfahren einer Position erkannt werden.

Die HiL-Simulation zeigt gute Ergebnisse. Es ist zu erkennen, dass beim Start einer Zielbremsung im gekuppelten Zustand zu den größten Abweichungen des Kolonnenreglers kommt. Dies ist dadurch zu erklären, dass in diesem Fall der Sollwert der Zielbremsung einen vergleichsweise große sprunghafte Änderung vollzieht. Beim Annähern an eine Zielposition verändert sich der Sollwert langsamer, was auch für die Kolonnenregelung zu geringeren Sollwertänderungen führt, welchen besser gefolgt werden kann.

In Abbildung 6.15 sind die Ergebnisse der Testfahrt mit den Buggys abgebildet. Anders als in der HiL-Simulation, bei der die beiden Buggys die gesamte Zeit in einem festen Abstand zueinander gestanden haben, gibt es im Experiment Kommunikationsprobleme. Diese zeigen sich nach etwa 80 s im Positions- und Abstandsverlauf. Empfängt der Buggy 2 keine Position, wird die letzte bekannte Position bis zu einem neuen Update gehalten. Die treppenförmigen Verläufe deuten also auf Kommunikationsprobleme der verwendeten ZigBee-Module hin, wenn sich die Buggys voneinander entfernen. Zu den anderen Zeiten kann dies nicht beobachtet werden. Die weiteren Ergebnisse liegen in dem durch die HiL-Simulation vorhergesagten Bereich.

Fazit: Das Beispiel zeigt, welchen Nutzen der HiL-Modus der Buggys mit sich bringt. Die zuvor im Labor durchgeführte HiL-Simulation hat das Reglerkonzept schon vor dem tatsächlichen Experiment validiert. Wäre es bei dieser Untersuchung zu Abweichungen gekommen, hätte entsprechend des RCP-Konzepts reagiert werden und eine weitere Iteration des Regler-Entwurfs stattfinden können. Diese hätte geschehen können, ohne dass das reale Experiment hätte stattfinden müssen. So können Zeit und Geld gespart werden.

6.4 Untersuchung der Rechenkomplexität

Eine spannende Frage bei der Bewertung der beiden Regelungsverfahren ist die Frage nach der Performanz der Regler. Um diese bewerten zu können, wird die Rechnerauslastung der Regler im Betrieb herangezogen. Zunächst wird die Rechnerauslastung während der Simulation betrachtet und anschließend während des Experiments.

Bei der Ausführung eines Reglers auf Echtzeithardware wird üblicherweise eine Abtastzeit T_{ab} vorgegeben. Diese gibt an, in welchem Zeitabstand neue Berechnungen angestoßen werden. Können die Berechnungen in weniger Zeit als T_{ab} durchgeführt werden, wird trotzdem mit der Berechnung des nächsten Zyklus gewartet, bis die Zeit T_{ab} verstrichen ist. Bei der hier verwendeten Simulation ist das anders. Es werden alle Werte für einen Zeitschritt berechnet und anschließend werden direkt die Werte des nächsten Zeitschritts bestimmt. Wird also in unterschiedlichen Simulationen dieselbe Zeit simuliert, kann über die Zeit, die zur Simulation benötigt wurde, ein Rückschluss auf die Komplexität der Berechnungen gezogen werden. Dies gilt nur unter der Voraussetzung, dass zum Zeitpunkt der Simulation dieselben Ressourcen verfügbar sind und diese nicht von anderen Prozessen genutzt werden. Für die beiden Verfahren, Pfadregelung und FBFR, werden jeweils 10 Minuten simuliert. Die Simulation benötigt eine Rechenzeit von 460,0 s ($\cong 76,67\%$ bezogen auf die 10 Minuten) für die Pfadregelung und 427,1 s ($\cong 71,18\%$) für die FBFR. Die FBFR benötigt also etwa 5 % weniger Rechenzeit in der Simulation als die Pfadregelung. Der Aspekt, dass beide Verfahren Werte unter 100 % erreichen, ist ein guter Indikator für die Echtzeitfähigkeit der Verfahren. Der Rechner, auf dem die Simulation durchgeführt wird, ist leistungstärker als die verwendete Echtzeithardware und sollte die Berechnungen daher schneller als in Echtzeit durchführen können, auch wenn in der Simulation eine grafische Ausgabe mit berechnet wird, die es auf der Echtzeithardware so nicht gibt.

Im Experiment kann die Rechnerauslastung anders festgestellt werden. SIMULINK® stellt auf der Echtzeithardware, dem XPC im Fall der Buggys, ein Schnittstelle zur Verfügung, mit dem Regler-Informationen zur Laufzeit abgerufen werden können. Der interessanteste Wert, der auf diese Weise ausgelesen werden kann, ist der Mittelwert der sogenannten Gesamtausführungszeit (engl. total execution time) (TET). Er gibt an, wie lange im Mittel gerechnet wird, um alle Berechnungen, die innerhalb der Abtastzeit T_{ab} berechnet werden müssen, durchzuführen. Setzt man die TET und T_{ab} ins Verhältnis, erhält man einen Näherungswert dafür, wie stark die Hardware ausgelastet wird. Für beide Regler

gilt $T_{ab} = 10$ ms. Hierbei ist zu beachten, dass es in den Reglern selbst Prozesse mit unterschiedlichen Raten gibt. Es wird aber immer die niedrigste vorkommende Abtastzeit betrachtet. Diese ist auch bei der Pfadregelung $T_{ab} = 10$ ms, auch wenn die enthaltene MPR alle 100 ms ausgeführt wird. Die Rechenzeit der MPR verteilt sich entsprechend auf zehn Zeitfenster der Dauer T_{ab} . Bei der Bestimmung der TET wird dies entsprechend berücksichtigt. Beide Regler werden mehr als 8,5 min betrieben, bevor die TET-Werte ausgelesen werden. Für die Pfadregelung wird eine TET von 3,438 ms ($\cong 34,38\%$ Auslastung) beobachtet und für die FBFR eine leicht höhere TET von 3,777 ms ($\cong 37,77\%$ Auslastung). Es werden also etwa 3,39 % mehr der auf dem XPC verfügbaren Ressourcen für die Berechnung der FBFR benötigt als für die Pfadregelung.

Fazit: Der Rechenzeitunterschied beider Regler liegt im einstelligen Prozentbereich und fällt damit eher gering aus. Es muss aber auch berücksichtigt werden, was trotz nahezu identischer Ressourcenauslastung geleistet wird. Hier kann die FBFR ihren Vorteil ausspielen. Bei ihr wird alle 10 ms komplett neu gerechnet. Die Pfadregelung hingegen liefert nur alle 100 ms neue Werte. Die FBFR bringt also bei etwa gleicher Ressourcenauslastung 10-mal häufiger neue Werte an und kann so schneller auf Änderungen reagieren.

6.5 GO!-Abschluss mit den IRT-Buggys

Zum Abschluss des Projekts GO! gehört auch eine Demonstration mit den Buggys. Dabei wird an mehreren Beispielen das Potential des GO!-Empfängers gezeigt. Der aufwendigste Teil der Demonstration ist ein Rangierszenario, bei dem fünf Buggys kooperieren und gleichzeitig den virtuellen Zug umsetzen. Das Szenario besteht aus drei Phasen.

In der ersten Phase fährt ein Zug bestehend aus einer Streckenlok und drei Wagen in den Bahnhof ein und stellt die Wagen auf Gleis 473, siehe die Gleiskarte in Abbildung 5.7 auf Seite 99. Anschließend fährt die Streckenlok aus dem Bahnhof aus und wartet auf Gleis 471 auf den neu rangierten Zug.

Die zweite Phase besteht aus dem Rangieren der abgestellten Wagen. Dazu fährt die Rangierlok von dem Gleis 489 an die Wagen und rangiert diese abwechselnd auf die Gleise 475 und 477. Am Ende des Rangiervorgangs ist der mittlere Wagen auf ein anderes Gleis als die restlichen Wagen rangiert. Die Rangierlok fährt dann wieder auf das Gleis 489 zurück.

Abgeschlossen wird das Szenario in der dritten Phase. Die Streckenlok fährt zurück in den Bahnhof und kuppelt den Wagen auf Gleis 477 an. Bevor der neugebildete Zug den Bahnhof verlässt, wird noch eine Bremsprobe simuliert. Dabei fährt der Zug kurz an und bremst dann direkt wieder ab. Im realen Bahnbetrieb wird bei dieser Bremsprobe überprüft, ob die Bremsen der einzelnen Wagen funktionieren. Nach der Bremsprobe fährt der Zug erneut an und verlässt anschließend den Rangierbahnhof.

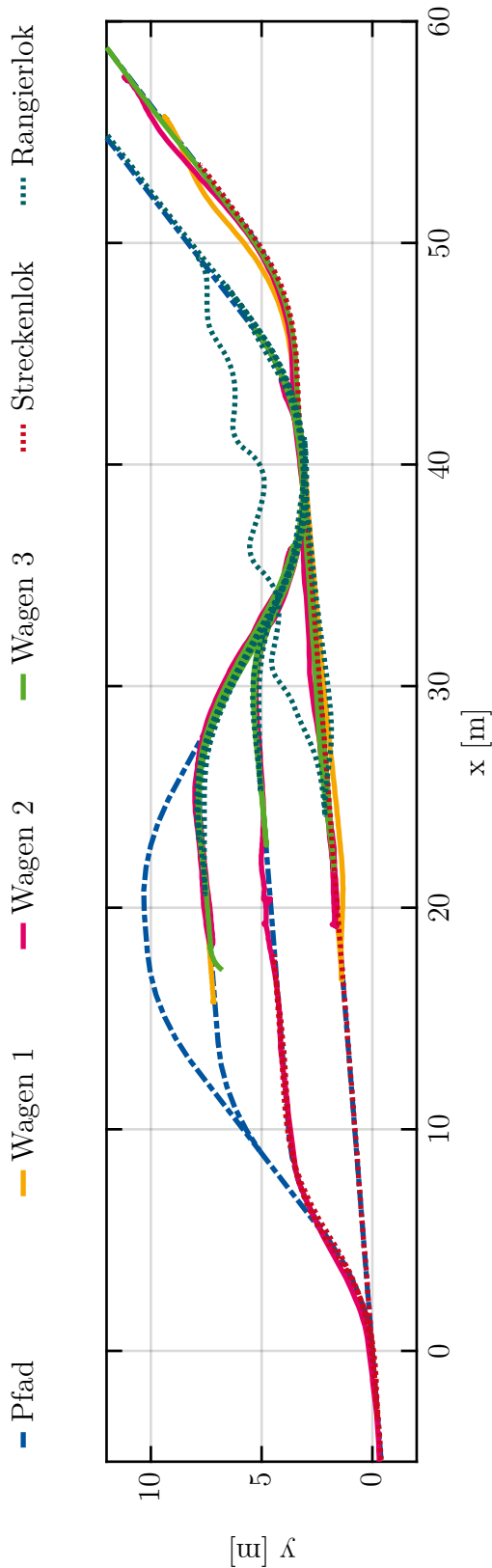


Abb. 6.16: Positionen der Buggys während des Rangierszenarios basierend auf [56]

Zum Zeitpunkt der GO!-Abschluss Testfahrten steht die FBFR noch nicht zur Verfügung. Das Szenario wird daher nur mit der Pfadregelung durchgeführt. Abbildung 6.16 zeigt die Positionsverläufe der Buggys während des Rangierszenarios. Es ist zu erkennen, dass es zu Beginn der Fahrt der Rangierlok zu einer Störung kommt. Das Navigationsfilter des entsprechenden Buggys gibt zu Beginn der Bewegung falsche Werte aus. Dies führt dazu, dass der Buggy Schlangenlinien fährt. Nach einiger Zeit kann der Navigationsfilter durch die bei der Fahrt aufgenommenen Messwerte immer besser die Position schätzen, was dazu führt, dass sich der Buggy der Strecke immer weiter annähert. Kurz vor dem Erreichen der anderen Wagen ist der Fehler nahezu ausgeglichen. Das Problem tritt auch bei anderen Buggys auf. Für Wagen 1 kommt es beim Losfahren auch zu Abweichungen, sowohl auf Gleis 490 als auch auf Gleis 473.

Abgesehen von den Positionsschwankungen einzelner Buggys wird das Szenario aber vollständig absolviert. Die Zielbremsung wird genutzt, um mit den Loks neue Positionen anzufahren. Mit und ohne gekuppelte Wagen gelingt dies. Sind Wagen virtuell gekuppelt, ist die Kolonnenregelung in der Lage, zwischen den Fahrzeugen nahezu einen konstanten Abstand einzuhalten. Auch hier treten die größten Schwankungen beim Losfahren auf, genau wie es auch schon in Abschnitt 6.3 beobachtet wurde.

Fazit: Mit den Buggys und der Pfadregelung konnte das Konzept des virtuellen Zuges erfolgreich umgesetzt werden. Alle Funktionalitäten des GO!-Empfängers für das Rangierszenario konnten erfolgreich getestet werden. Das Zusammenspiel von zentralen Diensten und Empfänger hat das Rangieren der Buggys möglich gemacht.

6.6 GO!-Abschluss mit dem *E*-Maxi L

Um die Fähigkeiten des GO!-Empfängers unter realistischen Bedingungen zu testen, wird mit dem *E*-MAXI L eine Demonstration im Rangierbahnhof Wustermark durchgeführt. Das Szenario ist eine vereinfachte Variante des Buggy-Abschlusszenarios. Zu Beginn des Szenarios steht der *E*-MAXI L vor zwei Wagen auf einem Gleis. Am Ende des Szenarios soll die Reihenfolge der Wagen getauscht sein. Um dies zu erreichen, wird zunächst der erste Wagen an den Elektrorangierer angekuppelt, anschließend über eine Weiche auf ein paralleles Gleis gefahren und dort abgestellt. Der *E*-MAXI L fährt dann ohne Wagen wieder zurück und kuppelt den verbliebenen Wagen an. Mit diesem wird auch auf das parallele Gleis gefahren, sodass nun beide Wagen in umgekehrter Reihenfolge an den *E*-MAXI L gekuppelt werden. Nach einer Bremsprobe fährt der Elektrorangierer mit beiden Wagen über die Weiche aus der Gleisharfe aus.

Abbildung 6.17 zeigt eine Luftaufnahme des Szenarios während der Abschlussdemonstration im GO!-Projekt. Die bei dem Szenario aufgenommenen Messdaten der Zielbremsung werden in Abbildung 6.18 visualisiert. Dazu gehören die Fahrzeuggeschwindigkeit (oben), die Beschleunigung in Fahrtrichtung (mitte) und der Abstand zur Zielposition (unten).

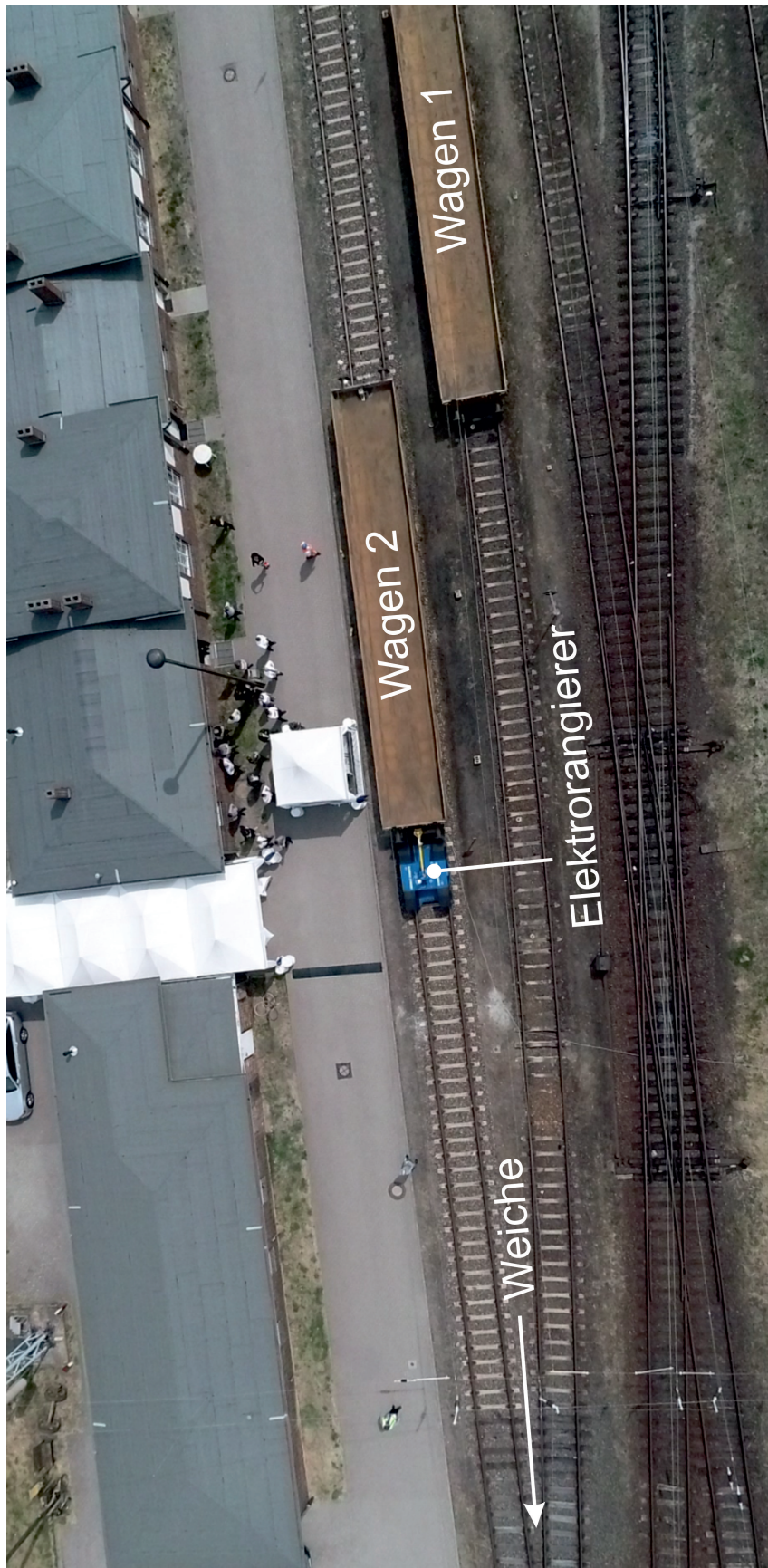


Abb. 6.17: Luftaufnahme vom Rangierszenario im RLCW [55]

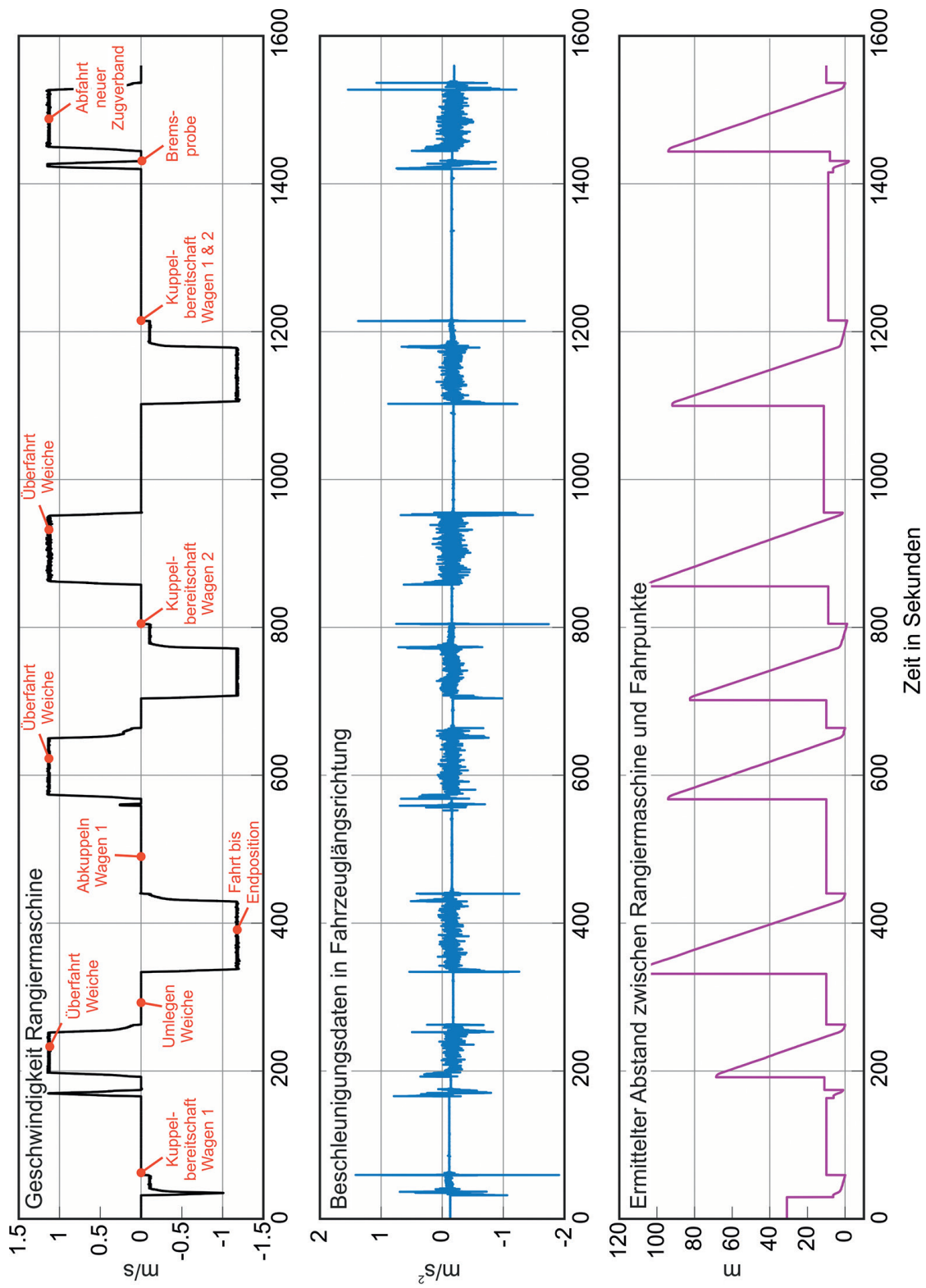


Abb. 6.18: Messdatenauswertung für das automatisierte Rangierszenario in GO! [54]

Die Zielbremsung ist das einzige Regelungsverfahren, das vom virtuellen Zug in die reale Anwendung übernommen wird.

Anders als bei den Buggys werden nun die IMU-Messwerte zum Stoppen des Zuges beim kuppelbereiten Positionieren verwendet. Aus dem oberen Graph der Abbildung 6.18 können die Phasen des Rangiervorgangs entnommen werden. Unter anderem ist zu erkennen, wann Kuppelbereitschaft hergestellt werden soll. In diesen Fällen kann beobachtet werden, dass die Geschwindigkeit auf die langsame Geschwindigkeit, entsprechend dem Geschwindigkeitsprofil, verringert wird und nach den zu kuppelnden Wagen „gesucht“ wird. Beim Auffahren auf den/die Wagen ist eine entsprechende Beschleunigungsspitze zu beobachten, der das Anhalten auslöst. Im unteren Teil von Abbildung 6.18 kann abgelesen werden, wann die Zielbremsung aktiv ist. Sobald das Ziel erreicht ist bzw. der Elektorangierer kuppelbereit positioniert ist, wird der Zielabstand auf einen niedrigen Wert um etwa 8 m gesetzt. Weitere Details zu der Demonstration können in [54] und [56] gefunden werden.

Fazit: Das Szenario wird unter Verwendung der GO!-Empfänger erfolgreich absolviert. Der Empfänger kann für die Positionsbestimmung und zur Steuerung des Elektrorangierers verwendet werden. Die mit dem virtuellen Zug erprobten Funktionalitäten können auch im vollwertigen Experiment Anwendung finden. Dies zeigt, dass der virtuelle Zug ein geeignetes Entwicklungswerkzeug darstellt.

6.7 Diskussion der Ergebnisse

Mit den Untersuchungen in diesem Kapitel wurde gezeigt, dass alle Komponenten des virtuellen Zuges sinnvoll zusammenspielen und es so ermöglichen, den virtuellen Zug mit den Buggys als mobile Entwicklungs- und Validierungsumgebung zu nutzen. Mit dem Konzept ist es möglich, mit den Buggys Entwicklungen für Bahnanwendungen voranzutreiben, wie es am Beispiel des GO!-Navigationsempfängers gezeigt wurde. Die Ergebnisse untermauern dies.

Im Abschnitt 6.1 konnte gezeigt werden, dass mit der Simulation ein weiteres mächtiges Tool zur Entwicklung der Anwendung vorhanden ist. Außerdem wurde dort demonstriert, dass sowohl Pfadregelung als auch FBFR in der Lage sind, die Funktionalität der virtuellen Schiene zu realisieren. Die zweite wesentliche Komponente, die virtuelle Kupplung, konnte in Abschnitt 6.3 validiert werden. Sowohl Zielbremsung als auch Kolonnenregelung versetzten die Buggys in die Lage, sich auf der virtuellen Schiene wie Züge zu verhalten. Zusätzlich wurde dabei die HiL-Simulation verwendet, ein weiteres Werkzeug zur Realisierung des Entwicklungsprozesses gemäß dem RCP-Gedanken. Später in Abschnitt 6.5 wurde dann gezeigt, dass die Komponenten auch zusammenspielen und das Konzept des virtuellen Zuges erfolgreich umgesetzt werden kann. Dazu wurde ein umfängliches Rangierszenario durchgeführt, bei dem fünf Buggys gleichzeitig involviert waren. Trotz dieser vielen Kommunikationsteilnehmer gab es neben den Problemen mit zwei Navigationsfiltern keine Probleme bei der Durchführung der Demonstration. Der virtuelle Zug hat also seine Aufgabe erfüllt. Dies wurde auch im Abschnitt 6.6 untermauert. Der GO!-Navigationsempfänger war dank der Vorentwicklung mit den Buggys in der Lage, auch den *E-MAXI L* zu automatisieren und durch die zentralen Dienste zu steuern.

Fazit: Virtuelle Schiene und virtuelle Kupplung realisieren im Zusammenspiel den virtuellen Zug. Dabei spielt es keine große Rolle, ob die Pfadregelung oder die FBFR eingesetzt wird. Die Buggys sind in der Lage, Züge nachzubilden, was als Basis weiterführender Tests und Entwicklungen genutzt werden kann.

7 Zusammenfassung

Im Rahmen dieser Arbeit wurde das Konzept des virtuellen Zuges vorgestellt und mit zwei unterschiedlichen Verfahren umgesetzt. Zunächst wurde die Pfadregelung basierend auf dem $2\frac{1}{2}$ Karotten-Regler ausgeführt. Mit diesem Ansatz konnte der virtuelle Zug realisiert werden. Der Ansatz zeichnet sich dadurch aus, dass er robust ist und direkt auf den Kartendaten angewendet werden kann. Um dies zu ermöglichen, optimiert die zugrundeliegende MPR die Lenkvorgaben. Dabei werden mehrere Iterationen durchlaufen, was rechenaufwendig ist. Daher arbeitet die Regelung mit einer Rate von 10 Hz.

Im Vergleich hierzu wurde die FBFR vorgestellt, die mit 100 Hz arbeitet. Dies kann erreicht werden, da bei ihr nicht iterativ optimiert werden muss. Vielmehr ist es so, dass durch die Ausnutzung der Flachheitseigenschaft der Buggys die Sollwerte direkt aus der Solltrajektorie bestimmt werden können. Anders als bei der Pfadregelung müssen hierfür die Kartendaten aufbereitet werden, um daraus die Solltrajektorie bestimmen zu können. Diese Aufgabe übernimmt der Trajektoriengenerator, der die nötigen Werte basierend auf geometrischen Zusammenhängen bestimmt. Durch diese gegebene Möglichkeit, vorwärts rechnen zu können, kann die Rate im Vergleich zur Pfadregelung gesteigert werden. Auf der Zielhardware zeigte sich, dass die FBFR etwa 3,4 % mehr Rechenleistung beansprucht. Mit einer Auslastung des Systems von etwa 37,8 %, im Fall der FBFR, ist die Hardware aber noch nicht an ihren Grenzen.

Es konnte gezeigt werden, dass die Abweichung von der Referenz im Allgemeinen bei der FBFR geringer gehalten werden konnte als bei der Pfadregelung. So liegt die Abweichung der FBFR im einstelligen Zentimeterbereich, während die der Pfadregelung gelegentlich in den zweistelligen Zentimeterbereich hineinragt. Dies gilt für gut definierte, fahrbare Strecken. An Stellen, bei denen unstetige Übergänge auftreten, kann diese Güte nicht gehalten werden. Dennoch konnten beide Verfahren auf die Problemstellen reagieren und die Abweichungen zur Referenz in diesen Bereichen unter 50 cm halten. Dies entspricht in etwa der halben Fahrzeugbreite.

Die geringere Abweichung der FBFR kann teilweise auf die höhere Rate des Verfahrens zurückgeführt werden. Es kann schneller auf Abweichungen reagiert werden, so dass diese auch früher ausgeglichen werden. Bezogen auf den Anwendungsfall, der im Rahmen dieser Arbeit betrachtet wurde, die Umsetzung des virtuellen Zuges, eignet sich die FBFR daher besser zur Umsetzung. Die Abweichungen fallen allgemein geringer aus. Außerdem bilden die Kartendaten von Gleisen fahrbare Strecken ab, was dazu führt, dass die Trajektorien allgemein so vorgegeben sind, dass es seltener zu Unstetigkeiten kommt als auf den Teststrecken, die zur Validierung herangezogen wurden.

8 Ausblick

Auch wenn die erzielten Resultate zufriedenstellend sind, gibt es weitere Verbesserungspotentiale für den vorgestellten Ansatz der FBFR. In diesem Abschnitt wird kurz auf diese Potentiale eingegangen und darauf, wie sie das Ergebnis der Regelaufgabe beeinflussen könnten.

Wie sich in Abschnitt 6.1 zeigt, scheint der Versatz der Antennenposition vom Hinterachsmittelpunkt Einfluss auf das im Experiment erzielte Ergebnis zu haben. Augenscheinlich führt dieser Einfluss zu schlechteren Ergebnissen als in der Simulation. Daher ist die zusätzliche Berücksichtigung des Versatzes der Antennenposition im Modell denkbar. So könnte der Einfluss dieser vorhandenen Modellabweichung minimiert werden. Dies wiederum könnte zu einer besseren Übereinstimmung vom Experiment mit der Simulation führen.

Bei dem vorgestellten Trajektoriengenerator werden Klothoiden durch aufeinanderfolgende Kreisabschnitte mit unterschiedlichen Radien approximiert. Klothoiden gehören aber zu den in der Trajektorienberechnung hinterlegten geometrischen Figuren. Daher könnte durch eine weiterführende Untersuchung der Kartendaten bestimmt werden, ob es sich bei dem Pfad um einen Kreis oder eine Klothoide handelt. Wird dabei eine Klothoide identifiziert, z. B. weil bei den aufeinanderfolgenden Kreisabschnitten der Radius kontinuierlich zu- oder abnimmt, könnte bestimmt werden, welchem Krümmungsverlauf dies entspricht. So könnten die Parameter der Klothoide bestimmt und diese direkt zur Trajektoriengenerierung verwendet werden. Dies würde die Unstetigkeiten entlang der Klothoide reduzieren.

Aktuell werden die in der Karte hinterlegten Punkte durch die geometrischen Figuren Gerade, Kreis und Klothoide verbunden. Dabei kommt es zum Beispiel beim Kreis zu un stetigen Übergängen in der Trajektorie. Dies folgt aus numerischen Ungenauigkeiten bei der Rekonstruktion. Würden die geometrischen Figuren stattdessen mit ihren Parametern in der Karte hinterlegt werden, so könnten die Unstetigkeiten deutlich reduziert werden. Das verwendete Kartenformat könnte hierzu beispielsweise in den Wegdaten um entsprechende Tags erweitert werden. So könnten die Parameter zusätzlich mitaufgenommen und eine Konstruktion der Trajektorie aus diesen Werten direkt vorgenommen werden. Dabei könnten die ursprünglichen Punkte zur Repräsentation erhalten bleiben. Die dadurch mögliche Verringerung der Unstetigkeiten sollte sich in geringeren Abweichungen von der Referenzstrecke widerspiegeln. Zusätzlich wird der Rechenaufwand reduziert, da die Parameter nicht erst aus den Kartendaten berechnet werden müssen, sondern direkt verwendet werden können. Das Hinterlegen der Parameter in den Kartendaten ist sowohl für Kreise als auch Klothoiden denkbar.

Neben der Erweiterung des Trajektoriengenerators um die direkte Umsetzung ist auch die Hinzunahme weiterer geometrischer Figuren denkbar. Ein Element aus dem Gleisbau wäre beispielsweise der Blossbogen, bei dem sich die Krümmung nicht linear wie bei den Klothoiden verändert, sondern kubisch [11]. So wären auch bei der Verbindung unterschiedlicher Elemente stetige Verläufe der Krümmung erreichbar.

Bei der Transformation der Kartendaten in das im Regler verwendete Format werden bereits einige Vorberechnungen angestellt und deren Ergebnisse im Speicher hinterlegt. Ein Beispiel ist die Bestimmung, ob ein Kartenpunkt ein Drehpunkt ist oder nicht. Dies könnte weiter vorangetrieben werden. Beispielsweise könnten Parameter wie der Kreisradius r für die einzelnen Kreisabschnitte im Speicher hinterlegt werden. Dadurch ließe sich die Rechenzeit bei der Online-Trajektoriengenerierung weiter reduzieren. Hierbei würde, anders als bei den vorher vorgestellten Verbesserungsvorschlägen, nicht auf die Berechnung verzichtet. Sie würde a priori erfolgen und so zur Laufzeit lediglich Speicheroperationen erfordern. Es ist auch denkbar, Teile oder die gesamte Referenztrajektorie in einer Lookup-Table zu hinterlegen. Dies sollte je nach gewählter Diskretisierung den Speicheraufwand enorm nach oben treiben, aber auch zu einer stark verringerten Laufzeit führen.

Die Parameter K_0 und K_1 nehmen bei der FBFR Einfluss darauf, ob Positions- oder Geschwindigkeitsabweichungen stärker gewichtet werden. Würde die Gewichtung beispielsweise in Abhängigkeit von der Geschwindigkeit erfolgen, könnte ggf. ein besseres Regelergebnis erreicht werden. Es erscheint jedenfalls plausibel, dass bei höheren Geschwindigkeiten zurückhaltender auf Positionsabweichungen reagiert werden sollte. Bereits bei kleinen Lenkbewegungen bei hoher Geschwindigkeit kommt es zu großen Positionsänderungen, wohingegen bei niedrigen Geschwindigkeiten deutlich mehr Lenkbewegungen erfolgen können. Als Beispiel sei hier der Vergleich zwischen dem Spurwechsel auf der Autobahn und dem Vorwärtseinparken in eine Parklücke längs zur Fahrbahn genannt. In beiden Fällen soll das Fahrzeug um eine Spurbreite zur Seite bewegt werden. Beim Parken wird die Lenkung im Vergleich zum Spurwechsel deutlich weiter eingeschlagen. Dieses Verhalten könnte in dem Regler durch geschwindigkeitsabhängige Parameter nachgebildet werden. So könnte die Möglichkeit bestehen, das Regelergebnis weiter zu verbessern.

In Abbildung 4.15 auf Seite 86 wird gezeigt, dass es eine Höchstgeschwindigkeit gibt, mit der die beiden aufeinanderfolgenden 90°-Kurven noch fahrbar sind. Dabei wird davon ausgegangen, dass jede der beiden Kurven durch eine Scheitel-Klothoide bzw. Verbundkurve angenähert werden soll. Daher wird versucht, auf die Gerade zwischen den beiden Kurven zurückzugelangen. Es ist aber auch denkbar, beide Kurven durch eine 180°-Verbundkurve anzunähern. Dadurch wird nur einmal die Krümmung auf und wieder abgebaut, was bisher für jede Kurve einzeln erfolgt, also insgesamt zwei mal. Das zwischenzeitliche Abbauen der Krümmung kostet „Strecke“, die durch einen Kreisbogen mit größerem Radius genutzt werden könnte. Das wiederum würde dazu führen, dass die Kurve mit einer höheren Geschwindigkeit durchfahren werden könnte. Ähnlich diesem Beispiel lassen sich sicherlich weitere Optimierungen bei der Trajektoriengenerierung finden.

Bei allen aufgegriffenen Verbesserungspotentialen wird es immer darauf ankommen, was das gewünschte Verhalten in der speziellen Anwendung ist: Ist es beispielsweise wichtiger, die Strecke möglichst schnell abfahren zu können, oder soll der Abstand zu den Kartenpositionen möglichst gering sein? Daher ist es wichtig, dass diese Ziele klar definiert sind, und so anschließend bewertet werden kann, ob die Umsetzung tatsächlich zu einer Verbesserung im Sinne der Anwendung führt.

Literaturverzeichnis

- [1] *AmpFlow.com Motor Performance Chart*. http://www.ampflow.com/E30-400_Chart.png. – besucht am 12.01.2019
- [2] *NMEA National Marine Electronics Association*. <https://www.nmea.org/>. – besucht am 10.02.2021
- [3] ABBENA, E. ; SALAMON, S. ; GRAY, A. : *Modern differential geometry of curves and surfaces with Mathematica*. CRC press, 2017
- [4] ABEL, D. : *Regelungstechnik und Ergänzungen (Höhere Regelungstechnik)*. 35. Verlagsgruppe Mainz GmbH Aachen, 2011
- [5] ABEL, D. ; ALRIFAE, B. ; HESSELER, F.-J. ; HOPPE, M. ; REITER, M. : Galileo-Based Advanced Driver Assistance Systems: Key Components and Development. In: *Automated Driving*. Springer, 2017, S. 183–200
- [6] ABEL, D. ; BOLLIG, A. : *Rapid control prototyping*. Springer, 2006
- [7] ALRIFAE, B. : *Networked model predictive control for vehicle collision avoidance*, Rheinisch-Westfälische Technische Hochschule Aachen, Dissertation, 2017. <http://dx.doi.org/10.18154/RWTH-2017-04199>
- [8] ALRIFAE, B. ; REITER, M. ; MASCHUW, J. P. ; CHRISTEN, F. ; ECKSTEIN, L. ; ABEL, D. : Satellite-and map-based long range cooperative adaptive cruise control system for road vehicles. In: *IFAC Proceedings Volumes 46* (2013), Nr. 21, S. 732–737
- [9] ÅSTRÖM, K. J. ; MURRAY, R. M.: *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010
- [10] BACHMANN, E. : Die Klothoide als Übergangskurve im Straßenbau. In: *Schweizerische Zeitschrift für Vermessung, Kulturtechnik und Photogrammetrie* 49 (1951), Nr. 6, 133–140. <http://doi.org/10.5169/seals-208340>
- [11] BLOSS, A. : Der Übergangsbogen mit geschwungener Überhöhungsrampe. In: *Organ für Fortschritte des Eisenbahnwesens* 91 (1936), Nr. 15, S. 319–320
- [12] BREUER, M. ; KONRAD, T. ; ABEL, D. : High precision localisation in customised gnss receiver for railway applications. In: *Proceedings of the 29th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2016)*, 2016, S. 779–787

- [13] BUSSE, M. : *Die Datensätze der NMEA Schnittstellen*. <http://www.nmea.de/nmea0183datensaetze.html>. Version: 2013. – besucht am 13.02.2019
- [14] CAI, G. ; CHEN, B. M. ; LEE, T. H.: *Unmanned rotorcraft systems*. Springer Science & Business Media, 2011
- [15] COHEN, M. A. ; ELIASBERG, J. ; HO, T.-H. : New product development: The performance and time-to-market tradeoff. In: *Management Science* 42 (1996), Nr. 2, S. 173–186
- [16] DEPRIEST, D. : *Dale DePriest's Navigation and GPS Articles - Explanation of NMEA sentences as used in GPS receivers*. <https://www.gpsinformation.org/dale/nmea.htm>. – besucht am 07.02.2019
- [17] D'ERRICO, J. R.: Efficient Computation of the Fresnel Sine and Cosine Integrals. Version: 05 2012. <https://de.mathworks.com/matlabcentral/mlc-downloads/downloads/submissions/28765/versions/2/download/zip>. – White Paper
- [18] DEUTSCHER, J. : Regelung mit mehreren Freiheitsgraden. In: *Zustandsregelung verteilt-parametrischer Systeme*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2012. – ISBN 978-3-642-19559-4, S. 67–74
- [19] DITTMAR, R. : Model Predictive Control mit MATLAB® und Simulink®. In: *Model Predictive Control mit MATLAB und Simulink-Model Predictive Control with MATLAB and Simulink*. IntechOpen, 2019
- [20] FENDRICH, L. : *Handbuch Eisenbahninfrastruktur*. Springer Berlin Heidelberg <https://books.google.de/books?id=5BgkBAAQBAJ>. – ISBN 9783540317074
- [21] FLIESS, M. ; LÉVINE, J. ; MARTIN, P. ; ROUCHON, P. : On Differentially Flat Nonlinear Systems. In: *IFAC Proceedings Volumes* 25 (1992), Nr. 13, S. 159–163
- [22] FLIESS, M. ; LÉVINE, J. ; MARTIN, P. ; ROUCHON, P. : Linéarisation par bouclage dynamique et transformations de Lie-Bäcklund. In: *Comptes rendus de l'Académie des sciences. Série 1, Mathématique* 317 (1993), Nr. 10, S. 981–986
- [23] FLIESS, M. ; LÉVINE, J. ; MARTIN, P. ; ROUCHON, P. : Flatness and defect of non-linear systems: introductory theory and examples. In: *International journal of control* 61 (1995), Nr. 6, S. 1327–1361
- [24] HABRICH, H. : *Geodetic applications of the global navigation satellite system (GLO-NASS) and of GLONASS/GPS combinations*. Verlag des Bundesamtes für Kartographie und Geodäsie, 2000
- [25] HAKENBERG, M. : *Reglerentwurf für die Längsführung von Fahrzeugkolonnen auf der Basis von Linearen Matrix-Ungleichungen*, Institut für Regelungstechnik, RWTH Aachen University, Diplomarbeit, 2007
- [26] HENRICI, P. : Zur numerischen Berechnung der Fresnelschen Integrale. In: *Zeitschrift für angewandte Mathematik und Physik ZAMP* 30 (1979), Nr. 2, S. 209–219

-
- [27] HINTERSEHER, M. : *Entwicklung von Konzepten, Algorithmen und Optimierungsverfahren zur Transformation von Knoten in einem Netzwerk unter Beachtung von Integritätsbedingungen*, Fachhochschule München, Fachbereich 07 Informatik, Diplomarbeit, 1999. <http://www.hinterseher.de/Diplomarbeit/>
- [28] LIN, J. ; GEHRT, J.-J. ; KONRAD, T. ; BREUER, M. ; ABEL, D. ; ZWEIGEL, R. : Embedded GNSS-receiver for Multi-Constellation Localization in Railway Applications. In: *Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018)*, 2018, S. 2007–2017
- [29] LUNZE, J. : *Regelungstechnik 2: Mehrgrößensysteme Digitale Regelung*. Bd. 9. Springer-Verlag, 2019
- [30] MALLIK, A. : Man-in-the-middle-attack: Understanding in simple words. In: *Cyberspace: Jurnal Pendidikan Teknologi Informatika* 2 (2019), Nr. 2, S. 109–134
- [31] MASCHUW, J. P. ; KESSLER, G. C. ; ABEL, D. : Kolonnenregelung auf der Basis linearer Matrixungleichungen bei kritischen längsdynamischen Manövern; Als Ms. gedr. In: *Steuerung und Regelung von Fahrzeugen und Motoren : AUTOREG 2008 ; 4. Fachtagung Baden-Baden, 12. und 13. Februar 2008 / VDI/VDE-Gesellschaft Mess- und Automatisierungstechnik* Bd. 2009. VDI-Verl. (VDI-Berichte), 175–186. – Beitr. überw. dt., teilw. engl.
- [32] MITSCHKE, M. ; WALLENTOWITZ, H. : *Dynamik der Kraftfahrzeuge. 5., überarb. u. erg. Aufl. 2014 Auflage*. Wiesbaden: Springer Vieweg, 2014
- [33] MONTENBRUCK, O. ; HAUSCHILD, A. ; STEIGENBERGER, P. ; HUGENTOBLE, U. ; TEUNISSEN, P. ; NAKAMURA, S. : Initial assessment of the COMPASS/BeiDou-2 regional navigation satellite system. In: *GPS solutions* 17 (2013), Nr. 2, S. 211–222
- [34] RAJAMANI, R. : *Vehicle dynamics and control*. Springer Science & Business Media, 2011
- [35] RAPP, H. : Kreisgleichungen. In: *Mathematik für die Fachschule Technik*. Springer, 1991, S. 225–231
- [36] RAPP, R. H. : *Geometric geodesy part I*. Ohio State University Department of Geodetic Science and Surveying, 1991
- [37] REITER, M. ; ABEL, D. ; GULYAS, D. ; SCHMIDT, C. : Concept implementation and test of a real-time capable map matching algorithm for use in GNSS-based driver assistance systems. In: *AUTOREG 5* (2011), S. 389–400
- [38] REITER, M. : *Environment and tools for the development of navigation- and map-based driver assistance and vehicle guidance systems*, Rheinisch-Westfälische Technische Hochschule Aachen, Dissertation, 2017. <http://dx.doi.org/10.18154/RWTH-2017-09313>
- [39] REITER, M. ; ABEL, D. : Two and a half carrots-a versatile and intuitive optimisation-based path-following approach for road vehicles. In: *Control and Automation (MED), 2015 23th Mediterranean Conference on IEEE*, 2015, S. 364–370

- [40] REITER, M. ; WEHR, M. ; ABEL, D. : Built-in HiL simulator: A concept for faster prototyping of navigation-and communication-based control systems. In: *Advanced Intelligent Mechatronics (AIM), 2016 IEEE International Conference on IEEE*, 2016, S. 1363–1369
- [41] REITER, M. ; WEHR, M. ; SEHR, F. ; TRZUSKOWSKY, A. ; TABORSKY, R. ; ABEL, D. : The IRT-Buggy–Vehicle Platform for Research and Education. In: *IFAC-PapersOnLine* 50 (2017), Nr. 1, S. 12588–12595
- [42] ROTHFUSS, R. : *Anwendung der flachheitsbasierten Analyse und Regelung nichtlinearer Mehrgrößensysteme*. 1997
- [43] RUDOLPH, J. : Flachheitsbasierte Folgeregelung / Lehrstuhl für Systemtheorie und Regelungstechnik Universität des Saarlandes. 2011. – Skript zur Vorlesung
- [44] SCHÄFER, B. ; TRZUSKOWSKY, A. ; ABEL, D. : GNSS-Based Assistance System for Automation of Shunting Operations of Freight Trains. In: *Proceedings of the 10th ITS European Congress (ITS)*, 2014, S. 1–12
- [45] TRZESNIEWSKI, M. : *Rennwagenteknik: Grundlagen, Konstruktion, Komponenten, Systeme*. Springer-Verlag, 2014
- [46] TRZUSKOWSKY, A. ; WEHR, M. ; ABEL, D. : Virtual Coupling: A Cooperative and Collaborative Autonomous Navigation Application. The Institute of Navigation, 1096-1102
- [47] WALTHER, A. : Der Kreis durch drei Punkte. In: *Mathematische Zeitschrift* 33 (1931), Nr. 1, S. 796–800
- [48] WANNINGER, L. : *Introduction to Network RTK*. <http://www.wasoft.de/e/iagwg451/intro/introduction.html>. Version: 2008. – besucht am 14.02.2019
- [49] WANNINGER, L. ; BEER, S. : BeiDou satellite-induced code pseudorange variations: diagnosis and therapy. In: *GPS solutions* 19 (2015), Nr. 4, S. 639–648
- [50] WEHR, M. : *Regelung, HiL-Simulation und experimentelle Untersuchung einer navigations- und kommunikationsbasierten Fahrzeugkette*, Institut für Regelungstechnik, RWTH Aachen University, Masterarbeit, 2015
- [51] WENDEL, J. : *Integrierte Navigationssysteme: Sensordatenfusion, GPS und Inertiale Navigation*. 2. Oldenbourg Verlag, 2011
- [52] WOLF, G. ; BRACHER, A. : *Straßenplanung*. 8. Auflage. 2013 (Werner-Ingenieur-Texte)
- [53] ZAMINPARDAZ, S. ; WANG, K. ; TEUNISSEN, P. J.: Australia-first high-precision positioning results with new Japanese QZSS regional satellite system. In: *GPS Solutions* 22 (2018), Nr. 4, S. 1–14
- [54] ZWEIGEL, R. ; TRZUSKOWSKY, A. ; LIN, J. ; ABEL, D. : Autonomes Rangieren im Förderprojekt Galileo Online: GO! In: *Der Eisenbahningenieur* 10/2018 (2018), 44-47. <https://publications.rwth-aachen.de/record/751693>. – ISSN 0013–2810

- [55] ZWEIGEL, R. : Gütertransport: Neue Möglichkeiten durch Automatisierung. In: *Technik in Bayern* 2019 (2019), Nr. 4, S. 16–17
- [56] ZWEIGEL, R. ; TRZUSKOWSKY, A. ; LIN, J. ; GEHRT, J.-J. ; TASDEMIR, E. ; NOLL, T. G. ; SIEMER, F. ; MÜLLER, H. ; RECHER, S. ; DRIESEN, J. ; BÖSINGER, M. ; DOHMANN, D. ; OVERBECK, M. ; STROBEL, C. ; DIETMAYER, K. ; GARZIA, F. : *Schlussbericht zum Vorhaben Galileo Online: GO! Deutscher Galileo-Empfänger für Bahnanwendungen*. 2019