

Karlsruher Institut für Technologie

Institut für Technik der Informationsverarbeitung (ITIV)

Institutsleitung

Prof. Dr.-Ing. Dr. h. c. Jürgen Becker (Sprecher)

Prof. Dr.-Ing. Eric Sax

Prof. Dr.rer.nat. Wilhelm Stork

Masterarbeit Nr. ID-3159

von Herrn cand. el. Jindong **Shi**

**Relative Positionierung mit LiDAR, Kamera,
IMU und Fahrzeugdaten**

**Fusion für Platooning; Relative Positioning
with LiDAR, Camera, IMU and Vehicle Date
Fusion for Platooning**

Beginn: 16.05.2022
Abgabe: 16.11.2022

Betreuer: Daniel Baumann
ITIV

Korreferent: -

Prof. Dr.-Ing. Eric Sax

Ich versichere hiermit, dass ich meine Masterarbeit selbständig und unter Beachtung der Regeln zur Sicherung guter wissenschaftlicher Praxis im Karlsruher Institut für Technologie (KIT) in der aktuellen Fassung angefertigt habe.

Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und wörtlich oder inhaltlich übernommene Stellen als solche kenntlich gemacht.

Karlsruhe, den 24/10/2022

Abstract

Platooning, a highly automated form of driving a group of vehicles together, in which several vehicles automatically drive in a group behind a lead vehicle, will have a significant impact on road traffic. A typical example is driving two buses in the context of urban platooning. A major challenge in bus platooning is the relative positioning of two buses. For this purpose, several external sensors, such as LiDAR and camera, are mounted on the rear bus to measure the relative distance and direction to the front bus.

In previous work in the institute ITIV, two different methods for the relative position estimation of buses are already designed. They are LiDAR and camera based approaches. However, continuous position detection by LiDAR or camera cannot be guaranteed at all times, because of their limited field of view, range and sampling rate. In addition, the relative position determined by LiDAR and camera contains a lot of noise, which can make trajectory planning and vehicle control more difficult. Since the front and the rear buses are also equipped with other internal sensors to estimate the internal vehicle driving dynamics, such as IMU, wheel encoder and steering angle sensor, and the rear bus can receive the measurement data of the front bus through a vehicle-to-vehicle (V2V) communication, a fusion of the position information determined by LiDAR and camera with the driving dynamics of two buses is possible, so that a more precise and more robust position estimation between two buses will be achieved. This master thesis has developed such an information fusion concept.

Zusammenfassung

Platooning, eine hochautomatisierte Form der Kolonnenfahrt, bei dem mehrere Fahrzeuge automatisiert hinter einem Führungsfahrzeug in Kolonne fahren, wird den Straßenverkehr maßgeblich beeinflussen. Ein typisches Beispiel ist das Fahren zweier Linienbusse in Kontext des urbanen Platoonings. Eine wesentliche Herausforderung beim Bus-Platooning ist die relative Positionierung zweier Busse. Dafür werden mehrere Umfeldsensoren, beispielsweise LiDAR und Kamera, auf dem verfolgende Bus montiert, um den relativen Abstand und die relative Richtung zum führenden Bus zu erfassen.

Bei früheren Arbeiten im Institut ITIV werden bereits zwei unterschiedlichen Methoden zur relativen Positionserkennung zweier Busse entworfen. Sie sind LiDAR- und Kamerabasierte Ansätze. Jedoch kann eine kontinuierliche Positionserkennung mit LiDAR oder Kamera in Bezug auf deren Blickfeld, Reichweite und Abtastrate nicht jederzeit garantiert sein. Außerdem enthalten die mit LiDAR und Kamera ermittelten Positionsinformationen viel Rauschen, welches die weitere Trajektorienplanung und Fahrzeugsteuerung erschweren kann. Da der führende und der verfolgende Bus zur Schätzung der internen Fahrdynamik auch mit anderen internen Sensoren, wie zum Beispiel IMU, Radencoder und Lenkwinkelsensor, ausgestattet sind, und der verfolgende Bus auf die Messdaten des führenden Busses mit einer Fahrzeug-zu-Fahrzeug (V2V-) Kommunikation zugreifen kann, ist eine Fusion der durch LiDAR und Kamera ermittelten Positionsinformationen mit der fahrdynamischen Kenngrößen zweier Busse möglich. Das Ergebnis der Fusion ist eine präzisere und robustere Positionsschätzung zweier Busse. Die vorliegende Masterarbeit beschäftigt sich mit solchem Informationsfusionskonzept.

Inhaltsverzeichnis

Abstract	3
Zusammenfassung	4
Inhaltsverzeichnis	5
Abbildungsverzeichnis	7
Tabellenverzeichnis	10
1 Einleitung	1
1.1 Umfeld	1
1.2 Motivation	1
1.2.1 Motivation 1: Reduzierung der Beobachtungsunsicherheit	2
1.2.2 Motivation 2: Erweiterung des „sichtbaren“ Bereichs	3
1.2.3 Motivation 3: Erhöhung der Robustheit	4
1.2.4 Motivation 4: Wiederherstellung des Lidar-Trackers	5
1.3 Aufgabenstellungen	6
1.4 Gliederung der Arbeit	6
2 Grundlagen und Stand der Technik beim Platooning	7
2.1 Einführung in Platooning	7
2.2 Sensorsysteme beim Bus-Platooning	8
2.2.1 Koordinatensysteme	8
2.2.2 IMU	9
2.2.3 Radencoder	11
2.2.4 Lenkwinkelsensor	12
2.3 Grundlagen der Fahrzeugdynamik	14
2.3.1 Seitenkraftfreie Kurvenfahrt	14
2.3.2 Kurvenfahrt mit Seitenkraft	15
2.3.3 Beschleunigung bei Kurvenfahrt	17
2.3.4 D'Alembert basierte Gleichungen	18
2.4 Kalman Filterung	20
2.4.1 Kalman Filter für lineare Systeme	20
2.4.2 Extended Kalman Filter	22
2.4.3 Unscented Kalman Filter	24
3 Konzeption der Informationsfusion für relative Positionierung beim Platooning	29
3.1 Kinematisches Modell der relativen Bewegung zweier Busse	30
3.2 Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs	34
3.2.1 Transformation der Radgeschwindigkeit	36
3.2.2 Detektion des Radschlups	38
3.2.3 Auswahl der besten Radgeschwindigkeit	40
3.2.4 Fusionierung mittels des Kalman Filters	41
3.3 Schätzung des Schwimmwinkels des Fahrzeugs	42
3.4 UKF basierte Fusionsalgorithmen für relative Positionierung	46
3.5 Zeitsynchronisierung der Sensoren	51

4	Implementierung in Simulation.....	53
4.1	Busmodell.....	53
4.2	Testszenario, Teststrecke und Fahrmanöver	54
4.3	Sensorik.....	55
4.4	Datenaustausch zwischen Fahrzeugen mittels des SimNet.....	57
5	Erprobung und Analyse der Ergebnisse	58
5.1	Methode zur Beurteilung der Schätzgüte	58
5.1.1	Häufigkeitsverteilung des Schätzfehlers	58
5.1.2	Wurzel der mittleren Fehlerquadratsumme.....	60
5.2	Auslegung des UKF für den Schätzer der relativen Pose in Simulation.....	60
5.2.1	Bestimmung der Kovarianz des Beobachtungsrauschens	60
5.2.2	Einstellung der Kovarianzmatrix des Prozessrauschens	63
5.2.3	Initialisierung des UKF im Schätzer der relativen Pose	65
5.3	Analyse der Ergebnisse in Simulation	66
5.3.1	Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs	66
5.3.2	Schätzung des Schwimmwinkels des Fahrzeugs	68
5.3.3	Schätzung der relativen Pose beim Platooning	69
6	Fazit.....	88
6.1	Zusammenfassung.....	88
6.2	Ausblick	88
7	Literaturverzeichnis.....	89

Abbildungsverzeichnis

Abbildung 1.1: Darstellung der relativen Pose zweier Busse	1
Abbildung 1.2: Häufigkeitsverteilung der Beobachtungsunsicherheiten der ermittelten relativen Pose	3
Abbildung 1.3: Szenarien des Misserfolgs der Detektion von Lidar-Tracker	4
Abbildung 1.4: Verarbeitung der Lidardaten aus [Mül22, p. 20]	5
Abbildung 2.1: Koordinatensysteme des verfolgenden Linienbusses	9
Abbildung 2.2: ein Feder-Masse-Dämpfer-System zur Messung der Beschleunigung	9
Abbildung 2.3: Darstellung der Coriolisbeschleunigung	10
Abbildung 2.4: Stimmgabelgyroskop [Trä14, p. 562]	11
Abbildung 2.5: InvenSense MPU-6050 MEMS-Inertialsensor	11
Abbildung 2.6: aktiver Raddrehzahlsensor aus [Reif17, p. 146]	12
Abbildung 2.7: AMR-Lenkwinkelsensor aus [Reif17, p. 154]	13
Abbildung 2.8: Fahrzeugmodell bei seitenkraftfreier Kurvenfahrt.....	15
Abbildung 2.9: Fahrzeugmodell bei einer Kurvenfahrt mit Seitenkraft	15
Abbildung 2.10: Einspurmodell bei einer Kurvenfahrt mit Seitenkraft nach [Bre15, p. 148].	16
Abbildung 2.11: Verfälschung einer Normalverteilung nach der Propagierung durch eine nichtlineare Funktion $g(\dots)$	24
Abbildung 2.12: Unscented Transform durch eine nichtlineare Funktion $f(\dots)$	26
Abbildung 3.1: Konzept der Informationsfusion zur besseren Schätzung der relativen Pose .	30
Abbildung 3.2: Darstellung der relativen Pose zweier Busse zum Zeitpunkt $tk - 1$	31
Abbildung 3.3: Darstellung der Bewegungen zweier Busse vom Zeitpunkt $tk - 1$ zum Zeitpunkt tk	32
Abbildung 3.4: Darstellung der Koordinatentransform vom Zeitpunkt $tk - 1$ zum Zeitpunkt tk	33
Abbildung 3.5: Rekonstruierung der longitudinalen Geschwindigkeit und des Schwimmwinkels des jeweiligen Busses	34
Abbildung 3.6: Blockschaltbild des Geschwindigkeit-Schätzers nach [Klo14, p. 2]	35
Abbildung 3.7: Transformation der Radgeschwindigkeit	37
Abbildung 3.8: Radschlupf beim Bremsen und Beschleunigen.....	39
Abbildung 3.9: Änderungen der Rotationsgeschwindigkeit beim Radschlupf	39
Abbildung 3.10: Abweichungen zwischen tatsächlichen und prädizierten Radgeschwindigkeit	39
Abbildung 3.11: Fahrt auf der geneigten Ebene für $\theta > 0$ und $\theta < 0$	41
Abbildung 3.12: Blockschaltbild des Schwimmwinkelschätzers	43
Abbildung 3.13: Blockschaltbild der zentralisierten Fusion zur Schätzung der relativen Pose	46
Abbildung 3.14: Darstellung der Zeitstempel des Fusionssystems.....	52
Abbildung 4.1: Busmodell des Mercedes-Benz Citaro 2005	53
Abbildung 4.2: Teststrecke im urbanen Testszenario	55

Abbildung 4.3: Änderung des longitudinalen Abstands, des lateralen Versatzes sowie der longitudinalen Geschwindigkeit des verfolgenden und führenden Busses	55
Abbildung 4.4: Blockschalbild des Camera RSI nach [Ref22, p. 830]	56
Abbildung 5.1: ein typisches Histogramm für die Häufigkeitsverteilung des Schätzfehlers e nach [Leó19]	59
Abbildung 5.2: OBTS Aruco-Marker-Konzept nach [Dim22]	61
Abbildung 5.3: Änderung des Beobachtungsrauschens	61
Abbildung 5.4: Darstellung des Schätzfehlers der relativen Pose, wenn Qk auf Nullmatrix gesetzt ist	64
Abbildung 5.5: Darstellung des Schätzfehlers der relativen Pose, wenn Qk auf sehr kleine Werte gesetzt ist	64
Abbildung 5.6: Darstellung des Schätzfehlers der relativen Pose, wenn Qk auf sehr große Werte gesetzt ist	65
Abbildung 5.7: Schätzfehler der longitudinalen Geschwindigkeit des verfolgenden Busses $\mathbf{vx1}, est$ und des führenden Busses $\mathbf{vx2}, est$	67
Abbildung 5.8: Verläufe für die Schätzwerte und die wahren Werte des Schwimmwinkels des verfolgenden Busses $\beta1$ und des führenden Busses $\beta2$	68
Abbildung 5.9: Schätzfehler des Schwimmwinkels des verfolgenden Busses $\beta1$ und des führenden Busses $\beta2$	69
Abbildung 5.10: Verläufe des Schätzfehlers der relativen Pose im Testfall „LKV“	71
Abbildung 5.11: Verläufe des Schätzfehlers der relativen Pose im Testfall „LKV“ (vertikal vergrößert)	72
Abbildung 5.12: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „LKV“	73
Abbildung 5.13: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „LKV“	73
Abbildung 5.14: Häufigkeitsverteilung des Schätzfehlers von $\Delta \varphi$ im Testfall „LKV“	74
Abbildung 5.15: Verläufe des Schätzfehlers der relativen Pose im Testfall „LV“	75
Abbildung 5.16: Verläufe des Schätzfehlers der relativen Pose im Testfall „LV“ (vertikal vergrößert)	75
Abbildung 5.17: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „LV“	76
Abbildung 5.18: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „LV“	76
Abbildung 5.19: Häufigkeitsverteilung des Schätzfehlers von $\Delta \varphi$ im Testfall „LV“	77
Abbildung 5.20: Verläufe des Schätzfehlers der relativen Pose im Testfall „KV“	78
Abbildung 5.21: Verläufe des Schätzfehlers der relativen Pose im Testfall „KV“ (vertikal vergrößert)	78
Abbildung 5.22: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „KV“	79
Abbildung 5.23: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „KV“	79
Abbildung 5.24: Häufigkeitsverteilung des Schätzfehlers von $\Delta \varphi$ im Testfall „KV“	80
Abbildung 5.25: Verläufe des Schätzfehlers der relativen Pose im Testfall „LK“	81
Abbildung 5.26: Verläufe des Schätzfehlers der relativen Pose im Testfall „LK“ (vertikal vergrößert)	82
Abbildung 5.27: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „LK“	82
Abbildung 5.28: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „LK“	83

Abbildung 5.29: Häufigkeitsverteilung des Schätzfehlers von $\Delta\varphi$ im Testfall „LK“	83
Abbildung 5.30: Wahrscheinlichkeit der Ausreißer der Schätzwerte Δx	84
Abbildung 5.31: Wahrscheinlichkeit der Ausreißer der Schätzwerte Δy	85
Abbildung 5.32: Wahrscheinlichkeit der Ausreißer der Schätzwerte $\Delta\varphi$	85
Abbildung 5.33: Verläufe des Schätzfehlers der relativen Pose im Testfall „V“	86
Abbildung 5.34: Verläufe des Schätzfehlers der relativen Pose im Testfall „Mode Switch“..	87
Abbildung 5.35: Verläufe des Schätzfehlers der relativen Pose im Testfall „Mode Switch“ (vertikal vergrößert)	87

Tabellenverzeichnis

Tabelle 2.1: Algorithmus des Kalman Filters	21
Tabelle 2.2: Algorithmus des Extended Kalman Filters	23
Tabelle 2.3: Algorithmus des Unscented Kalman Filters	28
Tabelle 3.1: Kriterien für die Auswahl der besten longitudinalen Geschwindigkeiten nach [Klo14, p. 4]	41
Tabelle 3.2: Frequenzen der Sensoren, CPU und V2V-Kommunikation	46
Tabelle 4.1: Wichtige Parameter des Busmodells (Mercedes-Benz Citaro 2005).....	54
Tabelle 5.1: Root Mean Square Error von $\mathbf{vx1}, \mathbf{est}$ und $\mathbf{vx2}, \mathbf{est}$	67
Tabelle 5.2: Root Mean Square Error von $\beta1, \mathbf{est}$ und $\beta2, \mathbf{est}$	69
Tabelle 5.3: Testfälle für den Schätzer der relativen Pose in der Simulation	70

1 Einleitung

1.1 Umfeld

Das Platooning ist keine neue Technologie und schon in viele Pilotprojekte für Lastkraftwagen-Güterverkehr auf Autobahnen getestet. Jedoch ist das Platooning selten für öffentlichen Personennahverkehr in städtischen Umgebungen untersucht. Ab Jahr 2021 wurde ein Pionierprojekt in der Stadt München gestartet, in dem die Forschende des KIT gemeinsam mit den Stadtwerken München (SWM) und Fahrzeughersteller Ebusco ein Stadtbus-Platoon für München entwickeln [01]. Beim sogenannten Bus-Plattonning werden zwei fahrerlose Busse mittels elektronischer Steuerung einem anderen geführten Bus automatisch folgen, und nur im vordersten Bus ein Fahrer sitzt. Diese Bus-Kolonnen kann sehr einfach an den jeweiligen Bedarf je nach verschiedenen Tagen oder verschiedenen Tageszeiten angepasst werden, weil die Busse in Kolonne nicht physikalisch sondern nur informationstechnisch gekoppelt sind. Sie können je nach Bedarf in kurzer Zeit sehr leicht geteilt oder verbunden werden [02]. Dabei bietet das Bus-Platooning eine hohe Flexibilität und eine Einsparung der Personalkosten. Ein weiterer Vorteil dieses Konzepts ist eine Einsparung der enormen Energie und eine Verringerung des Ausstoßes von Treibhausgasen, weil die verfolgende Busse nicht vom führenden Bus motorisiert sondern nur selbst gesteuert und selbst motorgetrieben sind, während die traditionelle Zugfahrzeuge viele Energie braucht, um die physikalisch gekoppelten Anhänger mitzubewegen [01].

1.2 Motivation

Um dem vorderen Bus automatisch zu folgen, muss der hintere Bus die relative Pose des vorderen Busses bezüglich des hinteren Busses kontinuierlich erfassen. Diese relativen Pose enthält den longitudinalen Abstand Δx , den lateralen Versatz Δy sowie die Differenz der Gierwinkel $\Delta\varphi$ der beiden Busses (siehe Abbildung 1.1). Mit dieser relativen Pose kann der hintere Bus die Trajektorie des vorderen Busses richtig schätzen und entlang dieser Trajektorie weiter fahren.

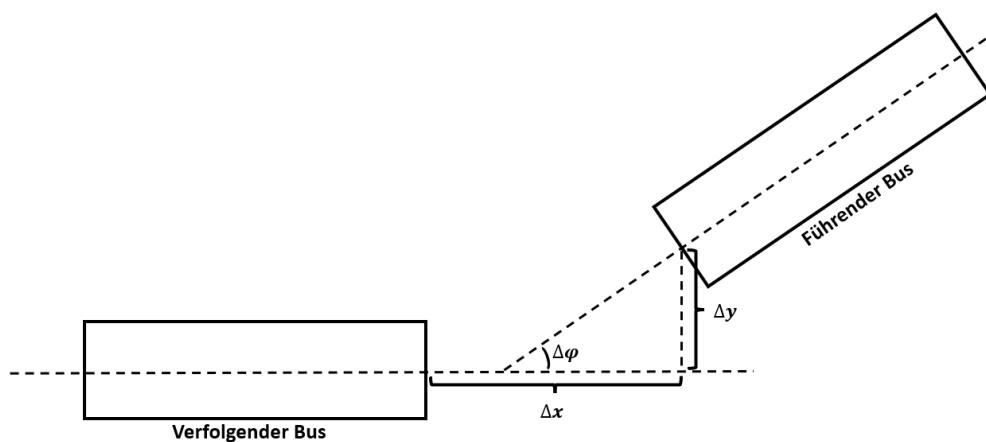


Abbildung 1.1: Darstellung der relativen Pose zweier Busse

Für eine eindeutige Detektion, Identifikation und Positionierung des führenden Busses sind der verfolgende Bus mit mehreren redundanten Umfeldsensoren ausgestattet. Eine Analyse der verschiedenen Umfeldsensoren für das Bus-Platooning findet sich in der Masterarbeit [16]. In diesem Projekt sind hauptsächlich zwei Lidarsensoren und eine Kamera genutzt. Bei früheren Arbeiten im Institut ITIV des KIT werden bereits auf Lidar und Kamera basierende Methoden zur relativen Positionserkennung zweier Busse entworfen. Die Masterarbeit [15] entwickelt einen Lidar-Tracker zum Detektieren und Tracken des vorausfahrenden Busses lediglich mittels der Punktewolken der zwei Lidarsensoren, während das Ergebnis die relative Pose der beiden Busse zueinander ist. In der Bachelorarbeit [13] werden mehrere Stereokamera benutzt, um die relative Pose zweier Busse mit ArUco-Markern zu erfassen. Im Vergleich dazu wird eine Monokamerabasierte Methode zur Erfassung der relativen Pose mit ArUco-Markern in [14] gefunden.

Zur besseren Schätzung der relativen Pose zweier Busse wird in dieser Arbeit ein Informationsfusionskonzept verwendet, in dem die Informationen aus LiDAR, Kamera, IMU und Fahrzeugdaten (Raddrehzahl, Lenkwinkel) zusammen verknüpft werden. Es gibt hauptsächlich **vier** Motivationen für eine Anwendung des Informationsfusionskonzepts, sie sind Reduzierung der Beobachtungsunsicherheit, Erweiterung des „sichtbaren“ Bereichs, Erhöhung der Robustheit sowie Wiederherstellung des Lidar-Trackers.

1.2.1 Motivation 1: Reduzierung der Beobachtungsunsicherheit

Der LiDAR basierte und der Kamerabasierte Schätzer für die relative Pose zweier Busse verwendet die Informationen aus den redundanten verschiedenen Sensorquellen, und arbeiten dabei unabhängig voneinander. Die geschätzte relative Pose vom jeweiligen Schätzer werden als Beobachtungen für tatsächliche relative Pose zweier Busse betrachtet. Jedoch enthalten diese Beobachtungen unbekannte Fehler, nämlich ein Beobachtungsrauschen oder eine Beobachtungsunsicherheit. Diese Fehler stammen von zufälligen Störungen im Messsignal der LiDAR und Kamera sowie von der Ungenauigkeit des Schätzalgorithmus der jeweiligen Schätzer. Die Beobachtungsunsicherheit lässt sich vollständig durch Verteilungsfunktion bzw. Wahrscheinlichkeitsdichte beschreiben. Falls der LiDAR- und der Kamerabasierte Pose-Schätzer richtig entworfen und erwartungstreu sind, können diese Beobachtungsunsicherheit als mittelwertfreie, weiße Gaußsche Rauschprozesse mit einer bestimmten Varianz modelliert werden. In Abbildung 1.2 sind die Häufigkeitsverteilung für die Beobachtungsunsicherheiten in der Simulation dargestellt. Diese Beobachtungsunsicherheiten sind dabei die Abweichungen zwischen den vom Lidar-Tracker und Kamera-Tracker ermittelten und den tatsächlichen Werten der relativen Pose.

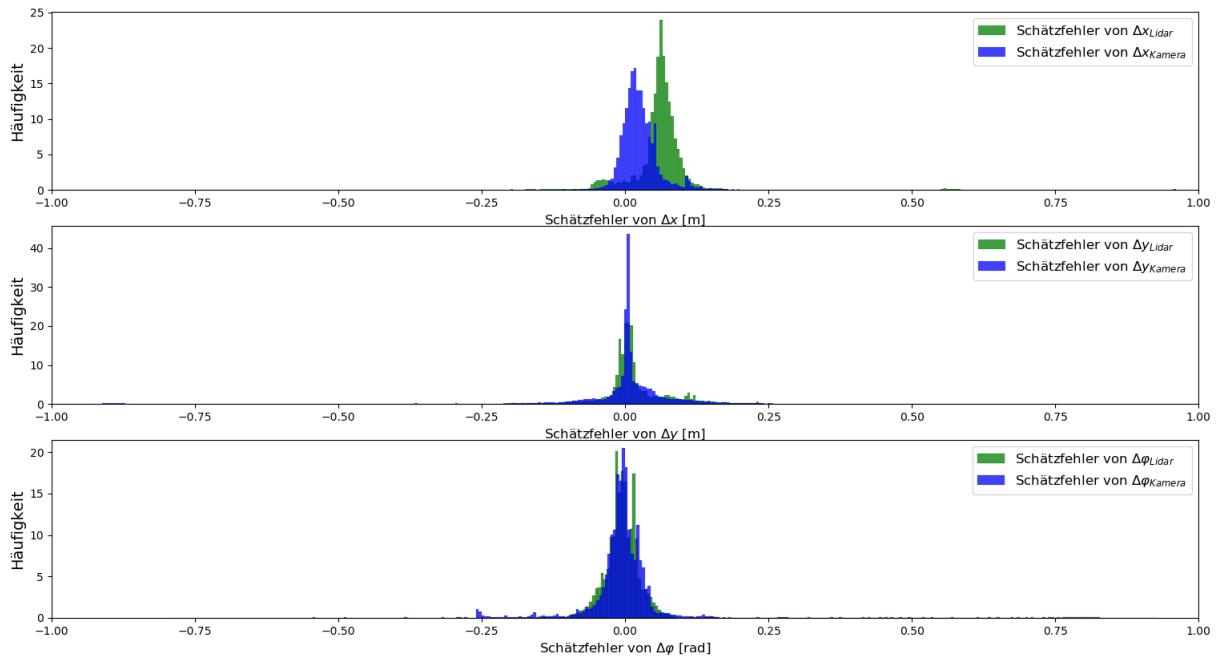


Abbildung 1.2: Häufigkeitsverteilung der Beobachtungsunsicherheiten der ermittelten relativen Pose

Man kann sehen, dass die vom Lidar-Tracker und Kamera-Tracker ermittelte relative Pose ein mittelwertfreies Gaußsches Rauschen enthält.

Da die Genauigkeit der relativen Pose zweier Busse die weitere Trajektorienplanung und Fahrzeugsteuerung maßgeblich beeinflusst, muss man die Beobachtungsunsicherheit der mit dem Kamera- und Lidar-Tracker ermittelten relativen Pose möglichst reduzieren, indem man sowohl die Beobachtungen von Lidar und Kamera, als auch die Messdaten der internen Sensoren (IMU, Radencoder, Lenkwinkelsensor), die auf dem jeweiligen Bus montiert sind, zusammen fusioniert.

1.2.2 Motivation 2: Erweiterung des „sichtbaren“ Bereichs

Eine kontinuierliche Positionserfassung durch den einzelnen Kamera- oder Lidarsensor kann nicht garantiert sein, weil der Kamera- und Lidarsensor aufgrund des begrenzten Blickfelds, der Reichweite und der niedrigen Abtastrate den führenden Bus nicht jederzeit detektieren können (besonders bei scharfer Abbiegung). Der führende Bus kann durch einen Umfeldsensor vom verfolgenden Bus detektiert werden, nur wenn dieser führende Bus in einem bestimmten Bereich relativ zu diesem Umfeldsensor liegt. Dieser Bereich wird als den „sichtbaren“ Bereich dieses Umfeldsensors genannt. In Abbildung 1.3 zeigt zwei Szenarien in der Simulation, in denen der Lidar-Tracker dem führenden Bus nicht richtig detektieren kann.

Ein Erweiterung des „sichtbaren“ Bereichs des einzelnen Umfeldsensors wird in zwei Fällen erzielt, wenn man das Informationsfusionskonzept für die Schätzung der relativen Pose verwendet:

Im ersten Fall ist ein Umfeldsensor (Lidar oder Kamera) gescheitert, den führenden Bus zu detektieren, während die anderen Umfeldsensoren aufgrund unterschiedlichem Blickfeld und Reichweite dem führenden Bus noch gut folgen. Durch eine komplementäre Fusion von verschiedenen Umfeldsensoren kann die „sichtbaren“ Bereiche aller einzelnen Umfeldsensoren zusammengesetzt. Eine Verfolgung des führenden Busses kann erzielt werden, solange mindestens ein Umfeldsensor den führenden Bus detektiert.

Im zweiten Fall können weder der Lidar- noch die Kamerasensoren den führenden Bus richtig detektieren. Im diesem Fall kann eine Schätzung der relativen Pose beim Informationsfusionskonzept mittels der Messdaten der internen Sensoren zweier Busse auch noch erzielt werden, bis der Kamera- oder Lidar-Tracker den führenden Bus wieder detektiert.

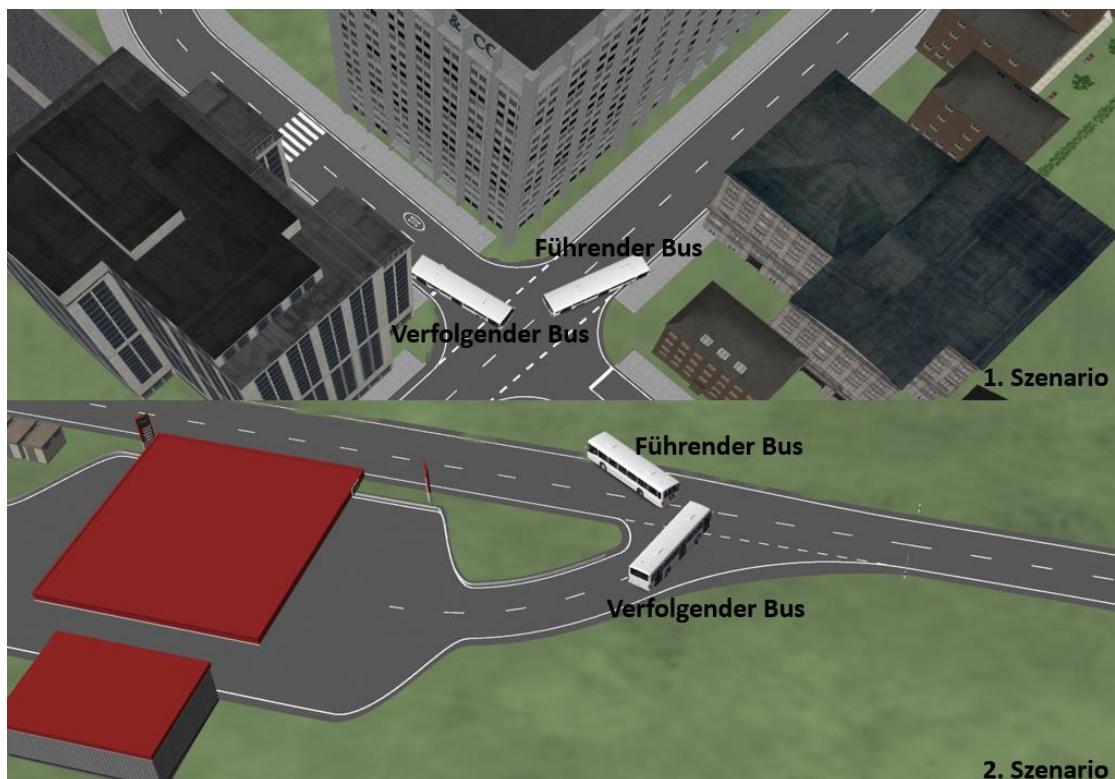


Abbildung 1.3: Szenarien des Misserfolgs der Detektion von Lidar-Tracker

1.2.3 Motivation 3: Erhöhung der Robustheit

Eine bessere Robustheit wird durch das Messen der redundanten Umfeldsensoren realisiert. Fällt ein Umfeldsensor aus, oder liefert ein Umfeldsensor komplett falsche Positionsinformationen, stehen noch die Beobachtungen anderer Umfeldsensoren beim Informationsfusionskonzept für die Positionierung zur Verfügung. Im Folgenden wird es auch dargestellt, dass eine bessere Schätzung der relativen Pose durch das Informationsfusionskonzept erzielt wird, auch wenn die V2V-Kommunikation zwischen Bussen abgebrochen ist.

1.2.4 Motivation 4: Wiederherstellung des Lidar-Trackers

Das Fusionssystem kann dem Lidar-Tracker helfen, den führenden Bus wieder zu detektieren, nachdem er irgendwann gescheitert ist, dem führenden Bus zu folgen. Um diesen Punkt genauer erläutern zu können, ist zunächst eine kurze Vorstellung des Lidar-Trackers notwendig.

In der Masterarbeit [15] ist die Vorgehensweise des Lidar-Trackers für die Verarbeitung der Lidardaten in mehrere Zwischenschritte aufgeteilt (siehe Abbildung 1.4). Die Punktwolke, welche von Objekten wie Fahrzeugen, Häuser, Menschen etc. reflektiert, aber nicht vom Boden reflektiert wurden, werden zunächst einzelnen Clustern zugeordnet. Danach wird ein geeignetes Buscluster aus der Liste aller gefundenen Cluster ausgewählt und die Busposition aus diesem Buscluster ermittelt. Da die Lidardaten keine semantischen Informationen über den vorausfahrenden Bus liefern, wird das Buscluster nach der Geometrie des vorausfahrenden Busses ausgewählt. Wenn das Cluster auf die geometrischen Eigenschaften, wie Höhe und Breite, des tatsächlichen vorausfahrenden Busses nicht passt, wird es direkt ausgeschlossen und nicht mehr als Kandidaten für das Buscluster berücksichtigt. Unter den restlichen Kandidaten für das Buscluster wählt der Algorithmus das Cluster mit dem niedrigsten euklidischen Abstand zur letzten Position des vorausfahrenden Busses aus. Das heißt, dass die Position $\mathbf{p}_{Buscluster,k}$ des ausgewählten Busclusters vom gegenwärtigen Zeitpunkt k möglichst nahe zur mit dem Lidar-Tracker ermittelten Busposition $\mathbf{p}_{Busposition,k-1}$ vom letzten Zeitpunkt $k-1$ sein muss. Im Subsystem „Ermittlung der Busposition“ wird auch geprüft, ob die ermittelte Busposition $\mathbf{p}_{Busposition,k}$ vom gegenwärtigen Zeitpunkt k nahe genug zur letzten Busposition $\mathbf{p}_{Busposition,k-1}$ liegt.

Der Lidar-Tracker kann gleich einen Fehler melden, sobald kein geeignetes Buscluster, das auf die geometrischen Eigenschaften des Busses passt und gleichzeitig nahe genug zur letzten Busposition liegt, gefunden wurde, oder die aus dem ausgewählten Buscluster ermittelte Busposition nicht nahe genug zur letzten Busposition liegt.

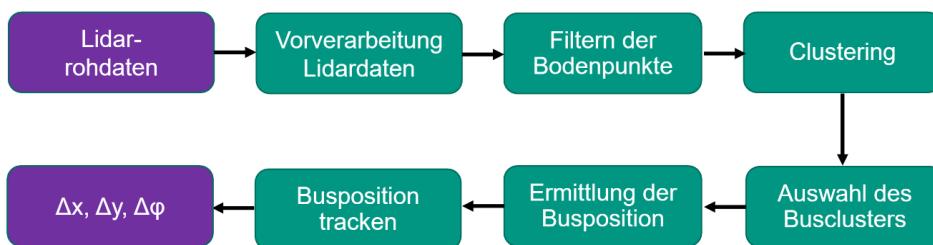


Abbildung 1.4: Verarbeitung der Lidardaten aus [15]

Falls der Lidar-Tracker irgendwann dem führenden Bus nicht mehr folgen kann, z.B. weil der führende Bus plötzlich eine scharfe Abbiegung hat, ist eine Wiederherstellung des Lidar-Trackers dann erforderlich, um den vorausfahrenden Bus wieder zu detektieren. Theoretisch ist eine Wiederherstellung des Lidar-Trackers möglich, sobald ein Buscluster mit richtigen geometrischen Eigenschaften in der Nähe der letzten Busposition gefunden wird. Aber eine Wiederherstellung des Lidar-Trackers ist tatsächlich schwierig, weil die Busposition vom letzten Zeitpunkt dem Lidar-Tracker unbekannt ist. Zum Beispiel, wenn das Tracking zum

Zeitpunkt k verloren wurde, und eine Schätzung der Busposition des führenden Busses ist danach nicht mehr erzielbar. Über eine Weile ist eine Wiederherstellung des Lidar-Trackers zum Zeitpunkt $k + n$ möglich, weil der Lidar den vorausfahrenden Bus zum Zeitpunkt $k + n$ gescannt hat, jedoch kann der Lidar-Tracker nicht feststellen, ob das gescannte Buscluster nahe genug zur Busposition vom letzten Zeitpunkt $k + n - 1$ ist, weil die Busposition vom letzten Zeitpunkt $k + n - 1$ dem Lidar-Tracker unbekannt ist.

Aber das Fusionssystem kann das Problem gut lösen, weil eine Schätzung der Busposition, wie in Abschnitt 1.2.2 erwähnt, beim Fusionssystem immer erzielbar ist, obwohl der Lidar-Tracker seit dem Zeitpunkt k den vorausfahren Bus nicht mehr folgen kann. Die vom Fusionssystem geschätzte Busposition $\hat{p}_{\text{Busposition}, k+n-1}$ wird in die Wiederherstellung des Lidar-Trackers zum Zeitpunkt $k + n$ eingesetzt, damit ein Buscluster in der Nähe der letzten Busposition detektiert wird. Dadurch erfolgt die Wiederherstellung des Lidar-Trackers.

1.3 Aufgabenstellungen

In Rahmen dieser Arbeit wird ein Fusionssystem mittels der Kalman-Filterung entworfen. Das Ziel des Fusionssystems ist eine sichere und robustere Schätzung der relativen Pose zweier Busse. Das entworfene Informationsfusionskonzept wird in der Simulationsumgebung CarMaker 11.0 implementiert. Die Performance des Fusionssystems wird je nach der Schätzgenauigkeit und der Robustheit evaluiert, in dem das Fusionssystem in verschiedenen Testfällen untersucht wird, z.B. wenn die Kamera- oder Lidarsensoren nicht angeschaltet sind, oder wenn die V2V-Kommunikation zwischen Bussen abgebrochen ist.

1.4 Gliederung der Arbeit

Im nächsten Kapitel werden alle technischen Grundlagen über Platooning, Sensorik, Fahrdynamik sowie die Methoden zur Informationsfusion erläutert. Das dritte Kapitel geht auf das Unscented Kalman Filter (UKF) basierende Informationsfusionskonzept für relative Positionierung, welches die Positionsbeobachtungen aus LiDAR und Kamera sowie das kinematisches Modell zweier Busse zusammen berücksichtigt und fusioniert. Voraussetzung für das Bilden des Kinematischen Modells ist zunächst eine Extraktion der wichtigen fahrdynamischen Kenngrößen zweier Busse aus Messdaten der internen Sensoren der jeweiligen Busse. Außerdem werden in diesem Kapitel noch das Verfahren zur Zeitsynchronisierung aller Sensoren vorgestellt. Danach erfolgt im vierten Kapitel eine Implementierung des entwickelten Konzepts innerhalb der Simulationsumgebung (CarMaker 11.0). Im fünften Kapitel wird das Verfahren der Einstellung des Filterparameters zuerst vorgestellt, danach werden die in der Simulation getesteten Ergebnisse evaluiert. Der Schwerpunkt der Evaluation liegt in der Genauigkeit und der Robustheit der Positionierungsverfahren, wie beispielsweise für die Szenarien, wenn LiDAR oder Kamera dem führenden Bus nicht mehr folgen kann, oder wenn die V2V-Kommunikation abgebrochen ist.

2 Grundlagen und Stand der Technik beim Platooning

2.1 Einführung in Platooning

Beim Platooning gibt der Fahrer des ersten Fahrzeugs Tempo und Richtung vor, alle anderen Fahrzeuge in der Kolonne folgen ihm automatisiert, aber lenken und bremsen selbsttätig im Takt des führenden Fahrzeugs. Die Fahrzeuge in der Kolonne kann sich über V2V-Kommunikation zu einer Einheit vernetzen. Die Fahrzeugdaten, die mit den lokalen Sensoren von jeweiligen Fahrzeugen am Bord gemessen sind, können zwischen Fahrzeugen ausgetauscht werden. Dabei bilden die Fahrzeuge beim Platooning ein kooperatives System, in welchem die nützlichen Information aller Fahrzeuge zusammen wirken, um die Umgebung zu erfassen, und das Fahrverhalten der Kolone zu steuern. Der Einsatz von Platooning kann viele Vorteile bringen, wie Beispielsweise eine Reduzierung des Kraftstoffverbrauchs, eine Erhöhung der Sicherheit, der Verkehrseffizienz und des Fahrkomforts sowie eine Einsparung der Personalkosten.

In vielen Projekten unterscheiden sich die Konzepte von Platooning aufgrund der verschiedenen Motivationen, Ziele und Technologie. In der Literatur [31] und [32] sind mehrere typischen Projekte von Platooning vorgestellt:

A. Energy ITS

Im japanischen Autobahnplatooning Projekt „Energy ITS“ sind drei schwere (25 Tonnen) Lastkraftwagen und ein leichter Lastkraftwagen beteiligt, wobei sowohl die longitudinale als auch die laterale Steuerung der Folgefahrzeuge automatisiert werden. Außerdem bietet das Platooning die Funktionen über Spurhaltung, Geschwindigkeitsregelung und Kollisionsvermeidung. Jeder Lkw ist mit einem 76 GHz Radar und einem 2 dimensionalen Lidar ausgestattet, damit der verfolgende Lkw den Abstand zum führenden Lkw messen, und einen spezifizierten Abstand zum führenden Fahrzeug einregeln kann. Das Testergebnis zeigt, dass Lkw-Platooning mit 10 Metern Abstand und der Geschwindigkeit von 80 km/h die Energie um ca. 15 % einsparen kann.

B. PATH

Das Projekt PATH der Universität von Kalifornien erforschte dabei, ob die Kapazität der Autobahn durch Platooning deutlich erhöht wird. Das Forschungsergebnis zeigt, dass es möglich sein könnte, die gegenwärtige Autobahnkapazität durch Platooning zwei bis drei mal zu erhöhen, falls von bis zu zehn Fahrzeuge als Kolonne auf der Autobahn fahren. Um die relative Pose des vorausfahrenden Lastkraftwagens zu bestimmen, hat PATH einen 24 GHz Radar und einen Lidar verwendet. Außerdem sind die Lastkraftwagen beim Platooning mit einer 50 Hz V2V-Kommunikation ausgestattet.

PATH hat die longitudinale Steuerung der Lastkraftwagen auf der Autobahn getestet. Der allgemeine Abstand zwischen Lkw beträgt 10 m, 8 m, 6 m und 4 m. Der Root Mean Square Error für die Regelung des longitudinalen Abstands zwischen Lkw ist ca. 22 ~ 25 cm, welche

den Fahrgästen eine ausreichende Fahrkomfort bringt. Die Lastkraftwagen beim Platooninn können auch viele Fahrmanöver, wie Beispielsweise Spurwechsel, realisieren. Während der Fahrt können die Lastkraftwagen auch sehr flexible mit der Kolonne an- oder abkoppeln.

2.2 Sensorsysteme beim Bus-Platooning

2.2.1 Koordinatensysteme

Der Linienbus beim Platooning besteht aus mehreren Teilsystemen, in welchen die orthogonalen Koordinatensysteme mit unterschiedlichen Ausrichtungen genutzt werden. In Abbildung 2.1 sind alle Koordinatensysteme des verfolgenden Linienbusses dargestellt:

1. Fahrzeugfestes Koordinatensystem $O_F X_F Y_F Z_F$

Der Ursprung O_F dieses Koordinatensystems liegt im Schwerpunkt des Busses. Die X_F und Y_F -Ebene liegen in der Horizontalebene. Die X_F -Achse ist mit der Längsachse des Busses überlagert, und die Z_F -Achse ist nach oben gerichtet.

2. Reifenfestes Koordinatensystem $O_R X_R Y_R Z_R$

Man sieht vier reifenfeste Koordinatensysteme für vier Räder. Der Ursprung O_R liegt im Reifenaufstandspunkt des jeweiligen Rades. Die X_R und Z_R -Ebene liegt in der Radmittelebene, und die X_R und Y_R -Ebene in der Horizontalebene.

3. Sensorkoordinatensysteme $O_K X_K Y_K Z_K$, $O_{LL} X_{LL} Y_{LL} Z_{LL}$ und $O_{LR} X_{LR} Y_{LR} Z_{LR}$

Für jeden Sensor wird ein Sensorkoordinatensystem definiert. $O_K X_K Y_K Z_K$ ist das Koordinatensystem der Frontkamera, in welchem die X_K -Achse mit der Längsachse des Busses überlagert nach vorne gerichtet ist. $O_{LL} X_{LL} Y_{LL} Z_{LL}$ und $O_{LR} X_{LR} Y_{LR} Z_{LR}$ sind die Koordinatensysteme der linken und rechten Lidarsensoren, welche X -Achse nicht nach vorne gerichtet sondern in Z -Achse um 45° und -45° verdreht sind. Da diese Lidarsensoren jeweils ein horizontales Sichtfeld von 270° und ein vertikales Sichtfeld von 90° aufweisen, können sie dadurch den kompletten Bereich vor der Busfront abdecken [15].

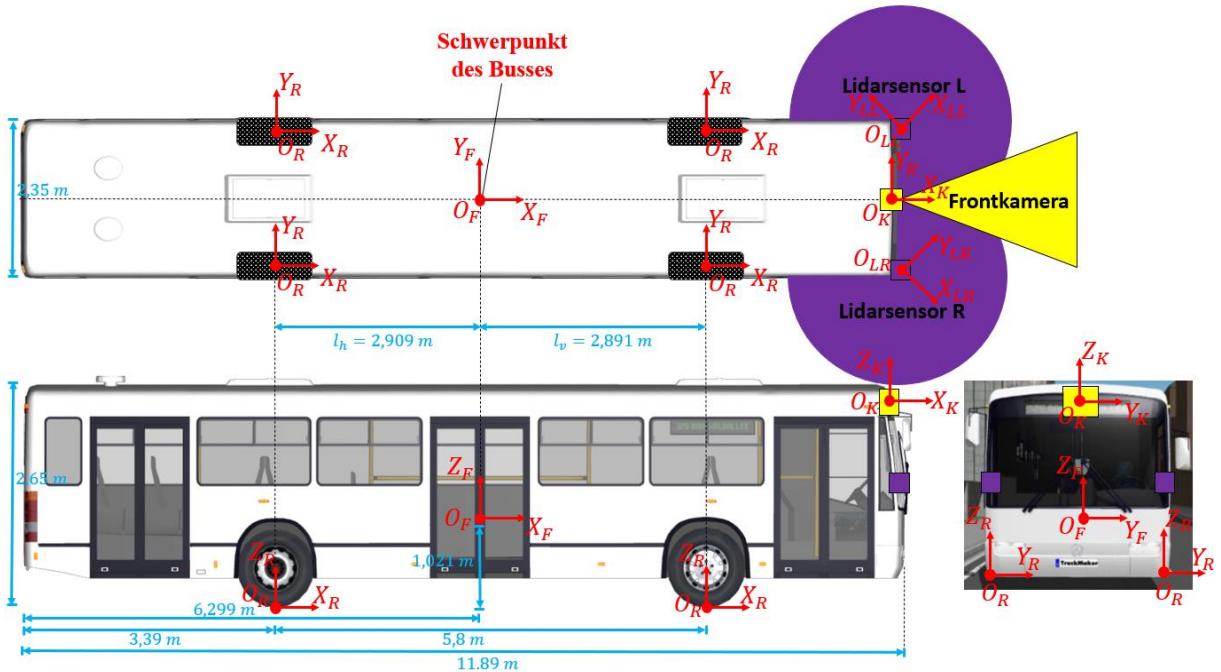


Abbildung 2.1: Koordinatensysteme des verfolgenden Linienbusses

2.2.2 IMU

Die Inertialsensoren (engl. inertial measurement unit, IMU) sind in viele Anwendungsgebiete, wie beispielsweise die Automobiltechnik, Robotik, Luft- und Raumfahrttechnik verbreitet. Sie führen eine Messung der Beschleunigung und Winkelgeschwindigkeit aus, indem die Beschleunigungssensoren und Drehratensensoren (engl. gyroscopes) verwendet werden. Die Beschleunigung und Winkelgeschwindigkeit sind zwei mechanische Größen, die in vielen Bereichen der Technik eine große Rolle spielen, weil sie die internen dynamischen Zustände des Systems beschreiben, und daraus andere Zustände wie beispielsweise Geschwindigkeit, Wegstrecke und Gierwinkel über zeitliche Integration auch zu herleiten sind. Die Messprinzipien der Beschleunigungssensoren und Drehratensensoren werden hier nach [08] kurz erläutert.

Beschleunigungssensoren

Ein Beschleunigungssensor misst die Beschleunigung α um eine definierte Achse in m/s^2 . Das Grundprinzip aller Beschleunigungssensoren nach verschiedenen Sensorarten besteht darin, die Wirkung der Beschleunigung auf ein Feder-Masse-Dämpfer-System [08] zu messen. (siehe Abbildung 2.2)

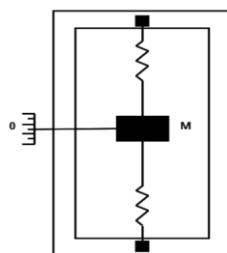


Abbildung 2.2: ein Feder-Masse-Dämpfer-System zur Messung der Beschleunigung

Dabei kann sich die elastisch an das Gehäuse angekoppelte Masse M bei Beschleunigung des Gehäuses gegenüber dem Gehäuse verschieben. Nach dem 2. Newton'schen Axiom

$$F = M \cdot a \quad (2.1)$$

ist die auf die Masse wirkende Kraft F proportional zur Beschleunigung a . Und die Kraft F führt zu einer bestimmten Verschiebung der Masse gegenüber dem Gehäuse. Dadurch erfolgt eine absolute Messung der Beschleunigung des Gehäuses durch eine Messung der Verschiebung der Masse. Es gibt verschiedene Beschleunigungssensoren nach verschiedener Sensorarten. Kapazitive Beschleunigungssensoren sind am häufigsten verbreitet, in denen die Änderung der Kapazität zur Messung der Verschiebung der Masse verwendet werden [08].

Drehratensensoren

Ein Drehratensensor, der die Winkelgeschwindigkeit ω um eine definierte Achse in rad/s misst, kann je nach der Anwendung verschiedene Messprinzipien verwenden. Bei hochgenauen Drehratensensoren, die häufig in der Luft- und Raumfahrt eingesetzt sind, werden die Winkelgeschwindigkeit durch den Sagnac-Effekt gemessen, indem zwei Lichtstrahlen durch Spiegel in einem Kreis in entgegengesetzte Richtungen geführt werden und je nach der Winkelgeschwindigkeit ein Lichtstrahl einen längeren Weg als der andere durchlaufen muss [08]. Die Differenz der Durchlaufzeit zweier Lichtstrahlen kann die Größe der Winkelgeschwindigkeit bestimmen. Die kostengünstigen Drehratensensoren messen die Winkelgeschwindigkeit über die Corioliskraft. Da solche Drehratensensoren häufig in Automobiltechnik eingesetzt sind, werden sie hier genauer vorgestellt.

Wie Abbildung 2.3 dargestellt, bewegt sich eine mit der Winkelgeschwindigkeit ω rotierende Masse translatorisch mit dem Geschwindigkeitsvektor v . Dadurch wird eine Beschleunigung a_c senkrecht zur der durch ω und v aufgespannten Ebene entstehen.

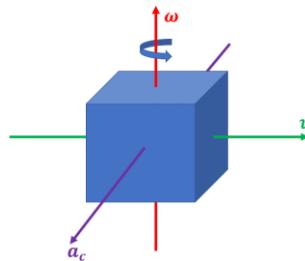


Abbildung 2.3: Darstellung der Coriolisbeschleunigung

Diese Beschleunigung a_c nennt man Coriolisbeschleunigung, die gemäß folgende Formel berechnet werden kann [08]:

$$a_c = 2v \times \omega \quad (2.2)$$

Mit diesem Prinzip kann eine Messung der Winkelgeschwindigkeit der Masse in eine Messung der Coriolisbeschleunigung gewandelt werden. Nach diesem Prinzip ist Stimmgabelgyroskop, das tatsächlich einen Beschleunigungssensor nutzt, zur Messung der Winkelgeschwindigkeit sehr weit verbreitet (siehe Abbildung 2.4).

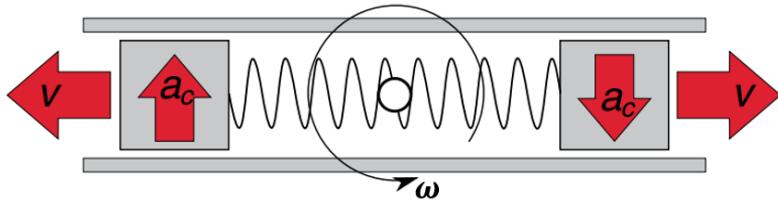


Abbildung 2.4: Stimmgabelgyroskop [08]

Man sieht, dass zwei Massen durch einen Anregungsaktor immer gegensinnig radial mit der Geschwindigkeit v schwingen können. Wenn der Sensor eine Rotation mit Winkelgeschwindigkeit ω erfährt, sind die Coriolisbeschleunigungen gleich durch einen Beschleunigungssensor messbar. Da die Störgrößen gleichzeitig und gegensinnig auf zwei Massen wirken, kann solche Struktur durch eine Differenzmethode einen großen Teil der Störgrößen eliminieren [09]. Dadurch ist das Messergebnis des Beschleunigungssensors, der solche Struktur verwendet, viel genauer, als den Beschleunigungssensor, der nur eine Masse nutzt.

MEMS-Inertialsensoren

Die Inertialsensoren werden häufig durch den Einsatz von Verfahren der MEMS-Technologie generiert werden. Bei modernen MEMS-Inertialsensoren werden für eine Messung der Beschleunigung und der Winkelgeschwindigkeit um xyz -Achse ein 3-Achsen-Beschleunigungssensor und ein 3-Achsen-Drehratensensor in einem sehr kleinen Chip integriert, wie z.B. beim Sensorchip InvenSense MPU-6050 alle Inertialsensoren in einem lediglich $4\text{mm} \times 4\text{mm} \times 0.9\text{mm}$ kleinen Gehäuse integriert werden (siehe Abbildung 2.5).

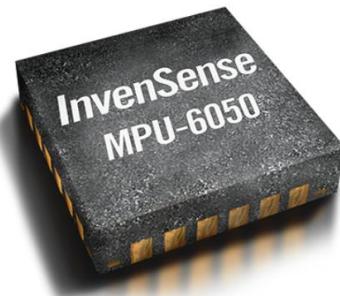


Abbildung 2.5: InvenSense MPU-6050 MEMS-Inertialsensor

(3-Achsen-Beschleunigungssensor + 3-Achsen-Drehratensensor) [10]

2.2.3 Radencoder

Radencoder oder Raddrehzahlsensoren misst eine Raddrehzahl n in rpm (Umdrehungen pro Minute) oder eine Rotationsgeschwindigkeit ω in rad/s der Fahrzeugräder. Diese Drehzahlsignal werden häufig an das ABS-, ASR- oder ESP-Steuergerät des Fahrzeugs

weitergeleitet, um die Bremskraft je Rad individuell zu regeln. Dadurch ist ein Blockieren (bei ABS) oder Durchdrehen der Räder (bei ASR bzw. ESP) verhindert [11].

In dieser Arbeit werden die Geschwindigkeit vierer Räder des Busses zur Schätzung der longitudinalen Geschwindigkeit des Busses verwendet. Die Radgeschwindigkeit v_{Rad} in m/s lässt sich durch die Rotationsgeschwindigkeit ω in rad/s mal den Radradius R in m ermitteln:

$$v_{Rad} = \omega \cdot R \quad (2.3)$$

Hier werden die Messprinzipien der Radencoder nach [11] kurz erläutert. Raddrehzahlsensoren können je nach Messprinzipien in passive Drehzahlsensoren und aktive Drehzahlsensoren geteilt. Seit 1998 werden statt den passiven Raddrehzahlsensoren fast ausschließlich nur noch aktive Drehzahlsensoren in Automobiltechnik eingesetzt [11]. Deshalb werden hier nur aktive Raddrehzahlsensoren vorgestellt.

Ein aktiver Raddrehzahlsensor (siehe Abbildung 2.6 links) besteht aus einem Silizium-IC und einem Impulsrad. Als Impulsrad dient ein Multipolring (siehe Abbildung 2.6 rechts), bei dem wechselweise magnetisierte Kunststoffelemente ringförmig auf einem metallischen Träger angeordnet sind [11]. Der IC des Sensors wird durch die ständig wechselnde Folge von Nord- und Südpole des rotierenden Impulsrades gestört. Dadurch ist die Frequenz der vom wechselnden Magnetfeld induzierten Stromimpulse proportional zur Raddrehzahl. Der IC kann die Stromimpulse weiter verarbeiten und eine Drehzahl ausgeben. Bei Bosch sind in der Mehrzahl ein Hall-Messelement in dem IC integriert, das schon auf eine sehr kleine Änderung des Magnetfeldes reagiert und somit eine Raddrehzahl bis zum Radstillstand (0,1 km/h) detektieren kann [11].

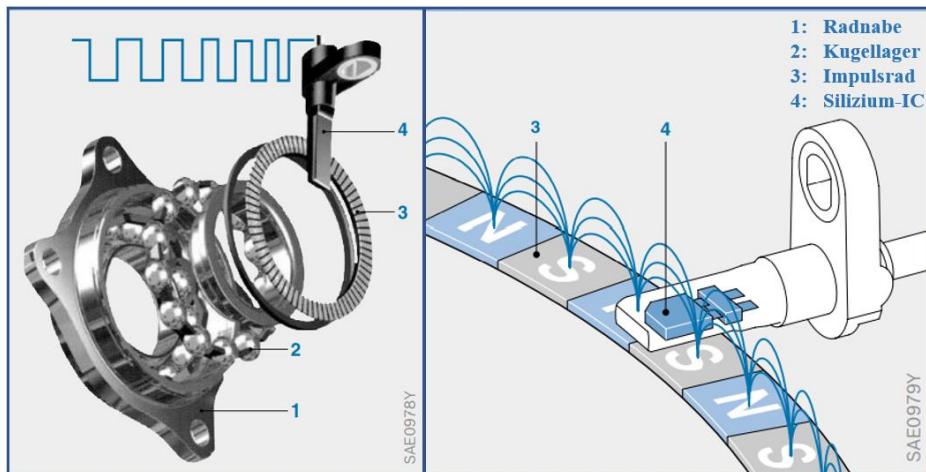


Abbildung 2.6: aktiver Raddrehzahlsensor aus [11]

2.2.4 Lenkwinkelsensor

Lenkwinkel der Fahrzeugräder ist eine wichtige Größe des Fahrzeugs und fahrdynamisch von Interesse. In dieser Arbeit sind die Lenkwinkel zweier vorderen Fahrzeugräder benötigt, um die Geschwindigkeit der vorderen Fahrzeugräder, die von Radencoder ermittelt sind, in

die Geschwindigkeit des Fahrzeugs umzuwandeln, sowie ein Schwimmwinkel des Fahrzeugs zu schätzen.

Bevor den Lenkwinkelsensor vorstellen zu können, muss man zunächst die Begriffe vom Lenkradwinkel und Lenkwinkel richtig unterscheiden. Der Lenkwinkel δ_v ist der Einschlagwinkel der gelenkten Vorderräder gegen die Längsrichtung des Fahrzeugs. δ_v ist der Mittelwert aus den Lenkwinkeln zweier Vorderräder, nämlich dem Lenkwinkel δ_{vl} des linken Vorderrads und dem Lenkwinkel δ_{vr} des rechten Vorderrads [12]:

$$\delta_v = \frac{\delta_{vl} + \delta_{vr}}{2} \quad (2.4)$$

Der Lenkradwinkel ist der Winkel, welchen ein Fahrer am Lenkrad einstellt, damit bestimmten Lenkwinkel der Vorderräder zu erreichen sind.

Ein Lenkwinkelsensor oder Lenkradwinkelsensor misst eine aktuelle Umdrehung des Lenkrads des Fahrzeugs, nämlich den Lenkradwinkel. Aus dem gemessenen Lenkradwinkel sind die Lenkwinkel zweier vorderen Fahrzeugeräder ermittelbar, weil zwischen dem Lenkradwinkel und dem Lenkwinkel ein Übersetzungsverhältnis zwischen 15 und 20 liegt. Das bedeutet, dass der Lenkradwinkel 15 bis 20 mal so groß wie der Lenkwinkel am Rad ist [12]. Dies Übersetzungsverhältnis vom Lenkradwinkel auf den Lenkwinkel ist normalerweise nicht konstant, dafür muss ein Lenkwinkel-Kennlinie vorab ermittelt [12] und von Automobilhersteller oder Zulieferer geliefert werden.

Zur Messung des Lenkradwinkels können die Winkelsensoren mit verschiedenen Messprinzipien in die Lenkwinkelsensoren eingesetzt werden. Die meisten modernen Lenkwinkelsensoren nutzen den magnetoresistiven Effekte, deren Messprinzipien nach [11] werden hier kurz erläutert.

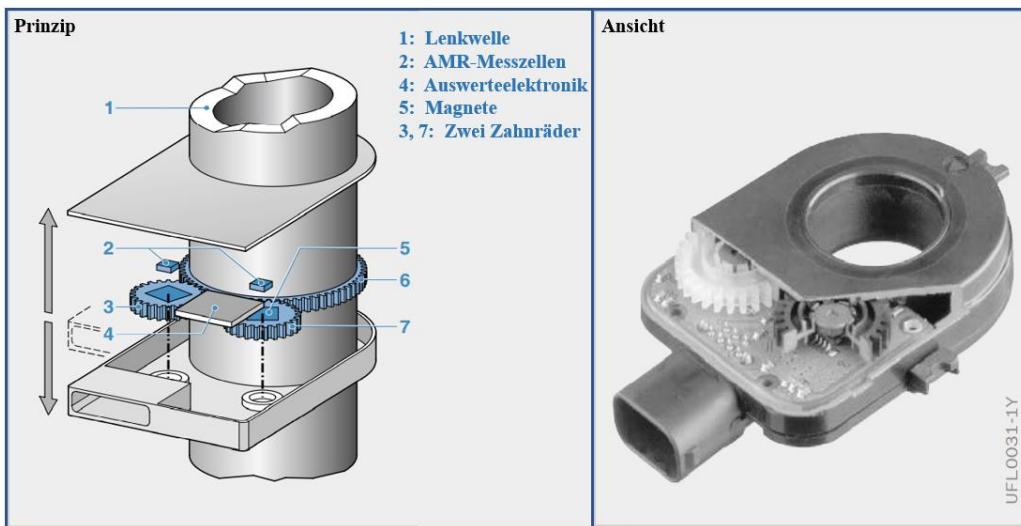


Abbildung 2.7: AMR-Lenkwinkelsensor aus [11]

Der Aufbau eines Lenkwinkelsensors, der mit einem „Anisotrop magnetoresistiven Sensor (AMR)“ arbeitet, sind in Abbildung 2.7 dargestellt. Wichtig sind hier zwei AMR-Messzellen

und zwei Zahnräder, in denen Magnete eingelassen sind. Ein Zahnrad ist auf der Lenkwellen antreibt, dabei ergibt sich der Lenkradwinkel durch das Messen der Winkel zweier Zahnräder. Das Magnetfeld verändert sich durch die Änderung der Winkel zweier Zahnräder, und der elektrischer Widerstand der AMR-Messzellen verändert sich durch die Änderung eines äußeren Magnetfelds. Dadurch ist das Messen des Lenkradwinkels durch das Messen des elektrischen Widerstands der AMR-Messzellen möglich. Durch bestimmte mathematische Algorithmen kann der Lenkradwinkel in einer Auswerteelektronik berechnet werden.

2.3 Grundlagen der Fahrzeugdynamik

Da das kinematische Modell der relativen Bewegung zweier Busse sowie das Schätzverfahren des Schwimmwinkels des Busses sich auf der Fahrzeugdynamik basieren, ist zunächst eine kurze Vorstellung der Grundlagen der Fahrzeugdynamik nach [12] erforderlich.

Da der Bus beim Platooning nicht nur die translatorische Bewegung in Längsrichtung hat, sondern auch in eine Kurve fährt, muss für die Kurvenfahrt die Querdynamik des Fahrzeugs mit Hilfe des Fahrzeugmodells analysiert werden. Neben der translatorischen Bewegung in Längsrichtung muss man auch die translatorische Bewegung in Querrichtung sowie die Gierrotation (Rotation um z-Achse des Fahrzeugs) berücksichtigen.

In diesem Kapitel werden zuerst zwei Fälle der Kurvenfahrten unterschiedet, nämlich die seitenkraftfreie Kurvenfahrt bei niedrigen Geschwindigkeiten sowie die schnelle Kurvenfahrt mit einer bestimmten Seitenkraft. Anschließend folgt die Herleitung der auf das Fahrzeug wirkenden Beschleunigung. Zuletzt werden die fahrdynamischen Gleichungen nach dem Prinzip von d'Alembert aufgestellt.

2.3.1 Seitenkraftfreie Kurvenfahrt

In Abbildung 2.8 zeigt das Fahrzeugmodell bei seitenkraftfreier Kurvenfahrt, welches sich seitenkraftfrei auf einem Kreis mit dem Momentanpol M bewegt. Die Räder können die Seitenkraft nur generieren, wenn es zwischen der Richtung der Radgeschwindigkeit und der Längsrichtung des Rades ein Winkel gibt. Diesen Winkel nennt man Schräglauwinkel α des Rades (siehe Abbildung 2.10). Bei seitenkraftfreier Kurvenfahrt sind die Richtungen der Geschwindigkeit jeder Räder ($v_{vl}, v_{vr}, v_{hl}, v_{hr}$) mit der Längsrichtung des jeweiligen Rades überlagert. Daher wird keine Seitenkraft generiert. Nach der Ackermann Lenkgeometrie müssen die Geschwindigkeiten der Hinterräder sowie die Geschwindigkeiten der Vorderräder senkrecht auf den Strahlen zum Momentanpol stehen. Dazu müssen zwei Vorderräder mit unterschiedlichen Lenkradwinkel δ_{vl} und δ_{vr} gelenkt werden. Der Lenkwinkel des kurveninneren Rades muss größer als den des kurvenäußeren Rades. Den Lenkwinkel δ_v bei seitenkraftfreier Kurvenfahrt, welcher der Mittelwert der beiden Lenkwinkel δ_{vl} und δ_{vr} ist, bezeichnet man als Ackermannlenkwinkel.

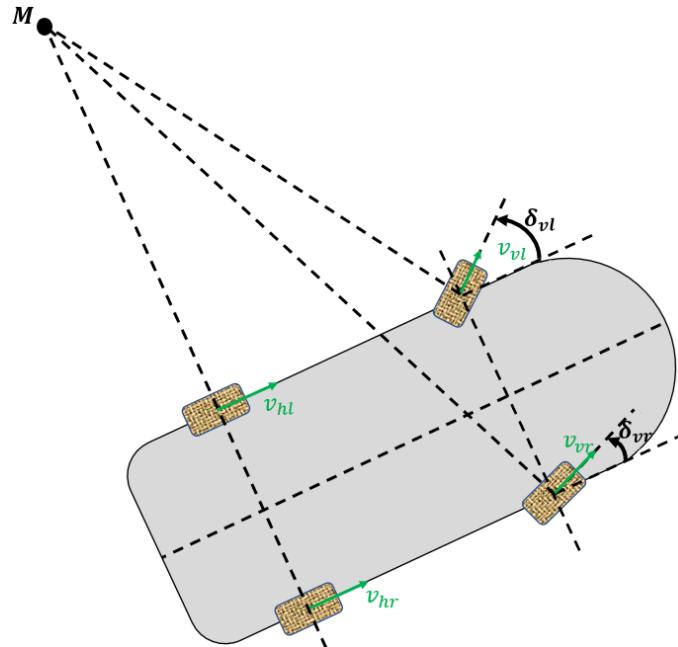


Abbildung 2.8: Fahrzeugmodell bei seitenkraftfreier Kurvenfahrt

2.3.2 Kurvenfahrt mit Seitenkraft

Bei schneller Kurvenfahrt wird durch die Schräglauwinkel der Räder eine Seitenkraft erzeugt, welche das Fahrzeug in Querrichtung beschleunigt. In Abbildung 2.9 sieht man, dass die Radgeschwindigkeiten nicht mehr mit der Längsrichtung des Rades überlagert sind. Um eine einfachere Analyse der Querdynamik in diesem Fall durchzuführen, wird das Fahrzeugmodell in Abbildung 2.9 häufig durch ein Einspurmodell, wie Abbildung 2.10 dargestellt, ersetzt.

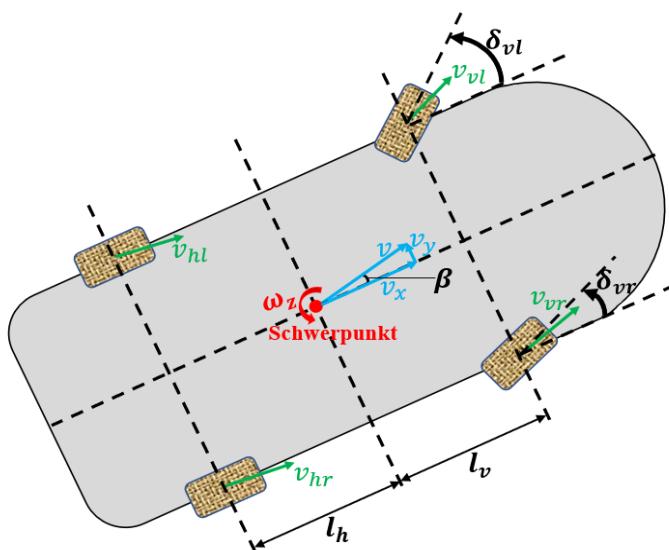


Abbildung 2.9: Fahrzeugmodell bei einer Kurvenfahrt mit Seitenkraft

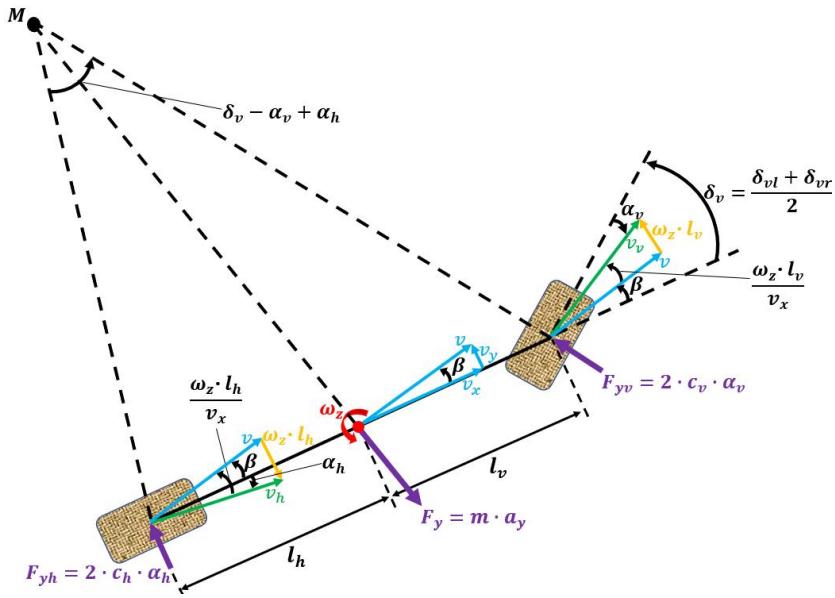


Abbildung 2.10: Einspurmodell bei einer Kurvenfahrt mit Seitenkraft nach [12]

Beim Einspurmodell sind nur ein Vorderrad und ein Hinterrad zu sehen, welches aber jeweils die Eigenschaften beider Räder einer Achse hat, zum Beispiel wirken auf dem Vorderrad nun die gesamten Seitenkräfte, die zuvor auf zwei Vorderrädern wirken. Es wird nun angenommen, dass der Gesamtschwerpunkt des Fahrzeugs auf Fahrbahnhöhe liegt, und die absoluten Abstände des Schwerpunkts des Fahrzeugs zur Vorder- und Hinterachse bekannt sind, nämlich l_v und l_h . Außerdem ist der Lenkwinkel δ_v des Vorderrads der Mittelwert der Lenkwinkel der beiden Vorderräder. Das Fahrzeug bewegt sich nun translatorisch mit der Schwerpunktgeschwindigkeit v .

Normalerweise ist die Richtung der Schwerpunktgeschwindigkeit des Fahrzeugs mit der Längsachse des Fahrzeugs überlagert. Aber bei einer Kurvenfahrt gibt es häufig ein Winkel β zwischen der Richtung der Schwerpunktgeschwindigkeit des Fahrzeugs und der Fahrzeulgängsachse. Diesen Winkel β nennt man Schwimmwinkel, welcher sich nach der longitudinalen Geschwindigkeit v_x und der lateralen Geschwindigkeit v_y des Fahrzeugs berechnen lässt:

$$\beta = \text{atan} \left(\frac{v_y}{v_x} \right) \approx \frac{v_y}{v_x} \quad (2.5)$$

Da das Fahrzeug sich um den Schwerpunkt mit der Giergeschwindigkeit ω_z dreht, wird die Schwerpunktgeschwindigkeit v an der Vorderachse mit der Geschwindigkeit $\omega_z \cdot l_v$ überlagert. Beide Geschwindigkeiten v und $\omega_z \cdot l_v$ können sich vektoriell zur Geschwindigkeit v_v addieren. Da der Winkel zwischen der Schwerpunktgeschwindigkeit v und der Geschwindigkeit v_v meistens sehr klein ist, lässt sich dieser Winkel näherungsweise durch die folgende Formel berechnen:

$$\tan \left(\frac{\omega_z \cdot l_v}{v} \right) \approx \sin \left(\frac{\omega_z \cdot l_v}{v} \right) \approx \frac{\omega_z \cdot l_v}{v} \approx \frac{\omega_z \cdot l_v}{v_x} \quad (2.6)$$

Die Schwerpunktgeschwindigkeit \boldsymbol{v} an der Hinterachse wird gleichfalls mit der Geschwindigkeit $\omega_z \cdot l_h$ addiert, und führt zur Geschwindigkeit \boldsymbol{v}_h . Der Winkel zwischen der Schwerpunktgeschwindigkeit \boldsymbol{v} und der Geschwindigkeit \boldsymbol{v}_h ist dabei $\frac{\omega_z \cdot l_h}{v_x}$.

Nun kann man die Gleichungen für die Schräglaufwinkel der Vorder- und Hinterachse, wie Abbildung 2.10 dargestellt, direkt aufstellen [12]:

$$\alpha_v = \delta_v - \frac{\omega_z \cdot l_v}{v_x} - \beta \quad (2.7)$$

$$\alpha_h = \frac{\omega_z \cdot l_h}{v_x} - \beta \quad (2.8)$$

2.3.3 Beschleunigung bei Kurvenfahrt

Die Beschleunigung des Fahrzeugs, die bei einer Kurvenfahrt auf den Schwerpunkt des Fahrzeugs wirkt, lässt sich über die folgende Formel ermitteln [12]:

$$\boldsymbol{a} = \dot{\boldsymbol{v}} + \boldsymbol{\omega} \times \boldsymbol{v} \quad (2.9)$$

wobei \boldsymbol{a} den Vektor der Beschleunigung bezeichnet, \boldsymbol{v} ist der Vektor der Schwerpunktgeschwindigkeit, und $\boldsymbol{\omega}$ der Vektor der Winkelgeschwindigkeit des Fahrzeugs.

Mit der Schwerpunktgeschwindigkeit \boldsymbol{v} und des Schwimmwinkels β kann man den Geschwindigkeitsvektor \boldsymbol{v} bestimmen:

$$\boldsymbol{v} = \begin{pmatrix} v \cdot \cos \beta \\ v \cdot \sin \beta \\ 0 \end{pmatrix} \quad (2.10)$$

Die zeitliche Ableitung von \boldsymbol{v} gilt dann:

$$\dot{\boldsymbol{v}} = \begin{pmatrix} \dot{v} \cdot \cos \beta - v \cdot \dot{\beta} \cdot \sin \beta \\ \dot{v} \cdot \sin \beta + v \cdot \dot{\beta} \cdot \cos \beta \\ 0 \end{pmatrix} \approx \begin{pmatrix} \dot{v} \cdot \cos \beta \\ \dot{v} \cdot \sin \beta \\ 0 \end{pmatrix} \quad (2.11)$$

Hier wird angenommen, dass sich der Schwimmwinkel β in kurzem Zeitintervall nicht ändert.

Da das Fahrzeug nur eine Giergeschwindigkeit ω_z hat, ergibt sich der Vektor der Winkelgeschwindigkeit des Fahrzeugs zu:

$$\boldsymbol{\omega} = \begin{pmatrix} 0 \\ 0 \\ \omega_z \end{pmatrix} \quad (2.12)$$

Man setzt (2.10)(2.11) und (2.12) in (2.9), damit wird \boldsymbol{a} bestimmt:

$$\boldsymbol{a} = \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} = \begin{pmatrix} \dot{v} \cdot \cos \beta - \omega_z \cdot v \cdot \sin \beta \\ \dot{v} \cdot \sin \beta + \omega_z \cdot v \cdot \cos \beta \\ 0 \end{pmatrix} = \begin{pmatrix} \dot{v}_x - \omega_z \cdot v_y \\ \dot{v}_y + \omega_z \cdot v_x \\ 0 \end{pmatrix} \quad (2.13)$$

2.3.4 D'Alembert basierte Gleichungen

Die Querbeschleunigung bzw. die Zentripetalbeschleunigung a_y des Fahrzeugs wirkt auf den Schwerpunkt des Fahrzeugs, und lässt sich durch den folgenden Ausdruck berechnen:

$$a_y = \frac{v^2}{R} \quad (2.14)$$

wobei v die Schwerpunktgeschwindigkeit, und R der Radius der Kurvenfahrt bezeichnet. Diese Beschleunigung ist auf den Momentanpol der Kurvenfahrt gerichtet. Die Seitenkraft F_y , welche auch als bei der Kurvenfahrt wirkende Trägheitskraft betrachtet wird, lässt sich nach dieser Querbeschleunigung a_y und dem Gewicht des Fahrzeugs m bestimmen:

$$F_y = m \cdot a_y = m \cdot \frac{v^2}{R} \quad (2.15)$$

Die Richtung der Seitenkraft F_y ist der Querbeschleunigung entgegengesetzt, nämlich vom Momentanpol nach außen gerichtet (siehe Abbildung 2.10).

Nach dem Prinzip von d'Alembert können die Gleichungen für das Einspurmodell zur Beschreibung des dynamischen Verhaltens des Fahrzeugs basierend auf ein Gleichgewicht von Kräfte sowie ein Gleichgewicht von Momenten aufgestellt werden.

Die erste Gleichung berücksichtigt die Seitenkräfte. Um das Fahrzeug auf einer Kurve mit konstantem Radius verbleiben zu können, müssen die Seitenkräfte an der Vorder- und Hinterachse der auf das Fahrzeug wirkenden Seitenkraft F_y entgegenwirken. Die Seitenkräfte an der Vorder- und Hinterachse, nämlich F_{yv} und F_{yh} , welche jeweils in die Y-Richtung des Koordinatensystems der Vorder- und Hinterachse weist, werden von den Rädern des Fahrzeugs aufgebracht, wenn sie unter einem Schräglauftwinkel α_v und α_h abrollen. In Abbildung 2.10 zeigen diese drei Kräfte, F_y , F_{yv} und F_{yh} , in unterschiedliche Richtungen. Diese Kräfte müssen zuerst jeweils in die Y-Richtung des fahrzeugfesten Koordinatensystems projektiert werden, dadurch kann man die Gleichung für ein Gleichgewicht aller Kräfte in Y-Richtung des fahrzeugfesten Koordinatensystems aufstellen:

$$F_y \cdot \cos \beta = F_{yv} \cdot \cos \delta_v + F_{yh} \quad (2.16)$$

Da der Schwimmwinkel β im Vergleich zum Lenkwinkel δ_v sehr klein ist, wird (2.16) durch die folgende Gleichung vereinfacht:

$$F_y = F_{yv} \cdot \cos \delta_v + F_{yh} \quad (2.17)$$

Zur Beschreibung des dynamischen Verhaltens des Fahrzeugs bei Kurvenfahrt gibt es eine weitere Gleichung, die basierend auf dem Gleichgewicht aller auf das Fahrzeug wirkenden Momente aufgestellt wird:

$$I_z \cdot \dot{\omega}_z = F_{yv} \cdot \cos \delta_v \cdot l_v - F_{yh} \cdot l_h \quad (2.18)$$

wobei I_z das Trägheitsmoment (auch Massenträgheitsmoment oder Inertialmoment) des Fahrzeugs um die z-Achse durch den Fahrzeugschwerpunkt bezeichnet, und $\dot{\omega}_z$ die zeitliche Änderung der Giergeschwindigkeit, bzw. die Gierbeschleunigung ist.

Das dynamische Verhalten des Fahrzeugs lässt sich nun mit der Gleichung (2.17) und (2.18) definieren, welche jedoch mehrere Unbekannte, nämlich die Seitenkräfte an der Vorder- und Hinterachse (F_{yv} und F_{yh}), enthalten.

Die unbekannten Seitenkräfte an der Vorder- und Hinterachse (F_{yv} und F_{yh}), welche jeweils die Summe der Seitenkräfte von linken und rechten Rädern einer Achse ist,

$$F_{yv} = F_{yvl} + F_{yvr} \quad (2.19)$$

$$F_{yh} = F_{yhl} + F_{yhr} \quad (2.20)$$

wird durch die Schräglaufwinkel der Vorder- und Hinterräder (α_v und α_h) generiert. Nach [12] ist der Zusammenhang zwischen Schräglaufwinkel und Seitenkraft des Rades bei geringen Massenänderungen des Fahrzeugs sowie kleinem Schräglaufwinkel (bis etwa 3°) üblicherweise linear:

$$\frac{F_{yvl}}{\alpha_v} = \frac{F_{yvr}}{\alpha_v} \approx c_v, \quad \frac{F_{yhl}}{\alpha_h} = \frac{F_{yhr}}{\alpha_h} \approx c_h \quad (2.21)$$

wobei die Konstante c_v und c_h in $N/^\circ$ oder N/rad die Schräglaufseitensteifigkeit an der Vorder- und Hinterachse bezeichnet. Für Querbeschleunigungen bis $4 \frac{m}{s^2}$ erhält man üblicherweise eine gute Linearisierung, während bei größeren Querbeschleunigungen der lineare Zusammenhang nicht mehr genau ist [12]. Dafür muss man die varierten Schräglaufseitensteifigkeiten c_v und c_h online schätzen [19].

Da die Querbeschleunigung des Linienbusses meistens unter $4 \frac{m}{s^2}$ liegt, wird hier lediglich der lineare Zusammenhang berücksichtigt. In diesem Fall kann man die Seitenkräfte an der Vorder- und Hinterachse in Abhängigkeit von dem Schräglaufwinkel und der Schräglaufseitensteifigkeit bestimmen, indem man (2.21) in (2.19) und (2.20) setzt:

$$F_{yv} = 2 \cdot c_v \cdot \alpha_v \quad (2.22)$$

$$F_{yh} = 2 \cdot c_h \cdot \alpha_h \quad (2.23)$$

Setzt man (2.5)(2.7)(2.8)(2.13)(2.15)(2.22)(2.23) in (2.17) und (2.18):

$$m \cdot (\dot{v}_y + \omega_z \cdot v_x) = 2 \cdot c_v \cdot \left(\delta_v - \frac{\omega_z \cdot l_v}{v_x} - \frac{v_y}{v_x} \right) \cdot \cos \delta_v + 2 \cdot c_h \cdot \left(\frac{\omega_z \cdot l_h}{v_x} - \frac{v_y}{v_x} \right) \quad (2.24)$$

$$I_z \cdot \dot{\omega}_z = 2 \cdot c_v \cdot \left(\delta_v - \frac{\omega_z \cdot l_v}{v_x} - \frac{v_y}{v_x} \right) \cdot \cos \delta_v \cdot l_v - 2 \cdot c_h \cdot \left(\frac{\omega_z \cdot l_h}{v_x} - \frac{v_y}{v_x} \right) \cdot l_h \quad (2.25)$$

werden das dynamische Verhalten des Fahrzeugs, welches auf eine Lenkwinkelvorgabe δ_v reagiert, mit den mehreren bekannten Fahrzeugparametern, wie Beispielweise der Masse, des Trägheitsmoments, der Schräglaufseitensteifigkeit, sowie der Lage des Schwerpunktes, definiert.

2.4 Kalman Filterung

Das Kalman Filter ist ein Zustandsschätzer. Es prädiziert rekursiv die Zustände mittels des Prozessmodells, und verbessert die Schätzung der Zustände basierend auf Beobachtungen. Das Ziel des Algorithmus des Kalman Filters ist eine Minimierung der Kovarianz des Schätzfehlers unter Berücksichtigung der Unsicherheit des Prozessmodells und der Beobachtungen. Dabei ist das Kalman Filter ein linearer erwartungstreuer Schätzer mit kleinster Varianz (*BLUE: Best Linear Unbiased Estimator*). Deshalb wird das Kalman Filter häufig für Informationsfusion eingesetzt, wenn die Informationen aus verschiedenen Sensoren zusammen verwendet werden, um eine beste Zustandsschätzung zu realisieren.

In diesem Kapitel wird zuerst die Theorie des Kalman Filters für den einfachsten Fall bzw. für lineare Systeme erläutert. Wenn das Systemmodell (inklusive Prozessmodell und Beobachtungsmodell) nichtlinear ist, ist das normale Kalman Filter nicht mehr geeignet. Zur Behandlung der Nichtlinearität des Modells werden hier anschließend zwei Varianten des normalen Kalman Filters vorgestellt. Sie sind Extended Kalman Filter und Unscented Kalman Filter.

2.4.1 Kalman Filter für lineare Systeme

Für das Kalman Filter wird häufig ein lineares, zeitdiskretes und zeitvariantes Systemmodell im Zustandsraum berücksichtigt:

$$\mathbf{x}_k = \mathbf{A}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{u}_k + \mathbf{G}_k \mathbf{v}_k \quad (\text{Prozessmodell}) \quad (2.26)$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\mu}_k \quad (\text{Beobachtungsmodell}) \quad (2.27)$$

Dabei ist \mathbf{x}_k der geschätzte Zustandsvektor, \mathbf{u}_k der deterministische Eingangsvektor, und \mathbf{z}_k der Beobachtungsvektor zum Zeitpunkt k .

\mathbf{A}_k , \mathbf{B}_k , \mathbf{G}_k , \mathbf{H}_k sind hier die Systemmatrix, die Steuermatrix, die Prozessrauschmatrix und die Beobachtungsmatrix. Sie sind linear und zeitvariant.

Das Prozessrauschen \mathbf{v}_k und das Beobachtungsrauschen $\boldsymbol{\mu}_k$ sind mittelwertfreie, weiße Gaußsche Rauschprozesse mit Kovarianzmatrizen \mathbf{Q}_k und \mathbf{R}_k :

$$\mathbf{v}_k \sim N(0, \mathbf{Q}_k), \quad \boldsymbol{\mu}_k \sim N(0, \mathbf{R}_k) \quad (2.28)$$

wobei:

$$\mathbf{Q}_k = E\{\mathbf{v}_i \mathbf{v}_i^T\}, \quad \mathbf{R}_k = E\{\boldsymbol{\mu}_i \boldsymbol{\mu}_i^T\} \quad (2.29)$$

Normalerweise ist das Beobachtungsrauschen $\boldsymbol{\mu}_k$ additives Rauschen, während das Prozessrauschen \mathbf{v}_k nicht additiv. Es wird durch die lineare Prozessrauschmatrix \mathbf{G}_k propoliert.

Es wird angenommen, dass das Systemrauschen und das Beobachtungsrauschen jeweils für unterschiedliche Zeitpunkte unkorreliert sind [03]:

$$E\{\boldsymbol{v}_i \boldsymbol{v}_j^T\} = 0, \quad E\{\boldsymbol{\mu}_i \boldsymbol{\mu}_j^T\} = 0 \quad \text{für } i \neq j \quad (2.30)$$

Außerdem wird es auch angenommen, dass das Prozessrauschen und das Beobachtungsrauschen zueinander unkorreliert sind [03]:

$$E\{\boldsymbol{v}_i \boldsymbol{\mu}_j^T\} = 0 \quad \text{für alle } i, j \quad (2.31)$$

Der Erwartungswert des Anfangszustands und die zugehörige Kovarianzmatrix sind bekannt:

$$\hat{\boldsymbol{x}}_0 = E\{\boldsymbol{x}_0\}, \quad \boldsymbol{P}_0 = Cov\{\boldsymbol{x}_0\} = E\{\boldsymbol{x}_0 \boldsymbol{x}_0^T\} \quad (2.32)$$

Für das Kalman Filter ist die Eigenschaften des Rauschens sehr wichtig. Nur unter die Voraussetzungen (2.28)(2.29)(2.30)(2.31) kann das Kalman-Filter eine Least-Square-Schätzung des jeweils aktuellen Zustandsvektor \boldsymbol{x}_k bei bekannten Eingangsgrößen, bekannten Beobachtungen und bekannten Anfangszustand (2.32) liefern.

Die iterative Struktur des Algorithmus des Kalman Filters wird in Tabelle 2.1 dargestellt.

Initialisierung:

$$\hat{\boldsymbol{x}}_0 = E\{\boldsymbol{x}_0\}, \quad \boldsymbol{P}_0 = Cov\{\boldsymbol{x}_0\} = E\{\boldsymbol{x}_0 \boldsymbol{x}_0^T\}$$

Für $k = 1, \dots, \infty$:

Prädiktionsschritt:

$$\begin{aligned} \hat{\boldsymbol{x}}_k^- &= \boldsymbol{A}_k \hat{\boldsymbol{x}}_{k-1} + \boldsymbol{B}_k \boldsymbol{u}_k \\ \boldsymbol{P}_k^- &= \boldsymbol{A}_k \boldsymbol{P}_{k-1} \boldsymbol{A}_k^T + \boldsymbol{G}_k \boldsymbol{Q}_k \boldsymbol{G}_k^T \end{aligned}$$

Innovationsschritt:

$$\begin{aligned} \boldsymbol{K}_k &= \boldsymbol{P}_k^- \boldsymbol{H}_k^T (\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{R}_k)^{-1} \\ \hat{\boldsymbol{x}}_k &= \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k (\mathbf{z}_k - \boldsymbol{H}_k \cdot \hat{\boldsymbol{x}}_k^-) \\ \boldsymbol{P}_k &= (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k) \boldsymbol{P}_k^- \end{aligned}$$

Tabelle 2.1: Algorithmus des Kalman Filters

Das Algorithmus des Kalman Filter lässt sich in zwei Schritte zerlegen.

Beim Prädiktionsschritt wird eine Prädiktion $\hat{\boldsymbol{x}}_k^-$ des Zustandsvektors mittels des Prozessmodells (2.26) durchgeführt. Die Prädiktionswert $\hat{\boldsymbol{x}}_k^-$ nennt man auch A-priori-Schätzwert, der als Vorwissen beim nächsten Schritt berücksichtigt wird. Außerdem wird die Kovarianzmatrix \boldsymbol{P}_{k-1} des Schätzfehlers vom letzten Zeitpunkt $k-1$ auch durch das Prozessmodell propagiert. Es führt zu einer vergrößerten Kovarianzmatrix \boldsymbol{P}_k^- , die als Kovarianzmatrix des Prädiktionsfehlers betrachtet wird.

Beim Innovationsschritt wird das Vorwissen $\hat{\boldsymbol{x}}_k^-$ durch die neuen Beobachtungen korrigiert, indem $\hat{\boldsymbol{x}}_k^-$ und die aktuellen Beobachtungen \mathbf{z}_k gewichtet summiert werden:

$$\underbrace{\hat{x}_k}_{\substack{A \text{ posteriori} \\ \text{Schätzung}}} = \underbrace{\hat{x}_k^-}_{\substack{\text{Vorwissen aus} \\ \text{der Prädiktion}}} + \underbrace{\mathbf{K}_k}_{\substack{\text{Verstärkungsmatrix}}} \cdot \underbrace{(\mathbf{z}_k - \mathbf{H}_k \cdot \hat{x}_k^-)}_{\text{Innovation}} \quad (2.33)$$

Das Ergebnis \hat{x}_k nennt man A-posteriori-Schätzwert, weil dieser Schätzwert die neuen Beobachtungen berücksichtigt. Der Term $\mathbf{H}_k \cdot \hat{x}_k^-$ bildet eine erwartete virtuelle Beobachtung, wenn der Prädiktionswert \hat{x}_k^- in lineare Messabbildung \mathbf{H}_k eingesetzt wird. Die Differenz zwischen der Beobachtung und der virtuellen Beobachtung nennt man eine Innovation. Die Korrektur des Zustandsvektors ist proportional zur aktuellen Innovation.

Die Wichtungsmatrix bzw. die Verstärkungsmatrix (Kalman Gain) \mathbf{K}_k wird so gewählt, dass die Kovarianz des Schätzfehlers zwischen dem wahren Zustandsvektor \mathbf{x}_k und dem geschätzten Zustandsvektor \hat{x}_k minimiert wird [03]:

$$\mathbf{P}_k = E\{\|\hat{x}_k - \mathbf{x}_k\|^2 \mid \mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_k\} \rightarrow \min \quad (2.34)$$

Aus dem Algorithmus des Kalman Filters kann man sehen, dass die Unsicherheit der Prädiktion \mathbf{P}_k^- nach dem Innovationsschritt mit Hilfe der neuen Beobachtung um $\mathbf{K}_k \mathbf{H}_k \mathbf{P}_k^-$ reduziert wird. Das bedeutet, dass der Prädiktionswert durch die neuen Beobachtungen korrigiert wird. Falls keine neue Beobachtung reinkommt, wird der Innovationsschritt nicht durchgeführt. In diesem Fall schätzt das Kalman Filter den Zustandsvektor lediglich durch den Prädiktionsschritt. Die Unsicherheit des geschätzten Zustandsvektors wird nach der Propagierung des Prozessmodells immer größer.

Mit Hilfe der Kovarianzmatrizen des Rauschens \mathbf{Q}_k und \mathbf{R}_k nimmt der Filter eine Gewichtung vor, in welcher der Prädiktionswert und die neue Beobachtung zueinander gewichtet werden müssen. Je kleiner das Beobachtungsrauschen \mathbf{R}_k ist, desto mehr werden diese Beobachtung beim Schätzung gewichtet. Das bedeutet, dass wir mehr auf die Beobachtungen vertrauen. Wenn die Unsicherheit der neuen Beobachtung sehr groß, nämlich \mathbf{R}_k sehr groß ist, hat der Prädiktionswert \hat{x}_k^- eine größere Gewichtung als die neue Beobachtung \mathbf{z}_k . In diesem Fall wird die neue Beobachtung weniger von Kalman Filter vertraut.

2.4.2 Extended Kalman Filter

Beim normalen Kalman Filter sind nur lineare Prozessmodell und Beobachtungsmodell zulässig. Um den Zustandsvektor eines nichtlinearen Systems zu schätzen, kann man das Extended Kalman Filter (EKF) verwenden. Für das EKF wird von einem nichtlinearen, zeitdiskreten Systemmodell ausgegangen:

$$\mathbf{x}_k = \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{v}_k) \quad (\text{Prozessmodell}) \quad (2.35)$$

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{u}_k) + \boldsymbol{\mu}_k \quad (\text{Beobachtungsmodell}) \quad (2.36)$$

Das Prozessrauschen \mathbf{v}_k und das Beobachtungsrauschen $\boldsymbol{\mu}_k$ sind hier ebenfalls mittelwertfreie, weiße Gaußsche Rauschprozesse mit Kovarianzmatrizen \mathbf{Q}_k und \mathbf{R}_k . Gleichfalls werden die Voraussetzungen (2.28)(2.29)(2.30)(2.31) hier auch erfüllt. Es wird

wieder angenommen, dass das Beobachtungsrauschen additives Rauschen ist, während das Prozessrauschen $\boldsymbol{\nu}_k$ in nichtlineares Prozessmodell eingesetzt wird.

Für die Behandlung der Nichtlinearität wird das nichtlineare Prozessmodell $\mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{u}_k, \boldsymbol{\nu}_k)$ und das nichtlineare Beobachtungsmodell $\mathbf{h}_k(\mathbf{x}_k, \mathbf{u}_k)$ um den Arbeitspunkt linearisiert. Dementsprechend werden die Jacobi Matrizen für die Systemmatrix \mathbf{A}_k und die Beobachtungsmatrix \mathbf{H}_k hergeleitet:

$$\mathbf{A}_k = \left. \frac{\partial \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{0})}{\partial \mathbf{x}_{k-1}} \right|_{\mathbf{x}_{k-1} = \hat{\mathbf{x}}_{k-1}} \quad (2.37)$$

$$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}_k(\mathbf{x}_k, \mathbf{u}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \hat{\mathbf{x}}_k^-} \quad (2.38)$$

Das Prozessrauschen $\boldsymbol{\nu}_k$ wird aufgrund der Mittelwertfreiheit bei Berechnung der Jacobi Matrix \mathbf{A}_k zu Null gesetzt [03].

Gleichfalls wird die Jacobi Matrix auch für die Prozessrauschmatrix \mathbf{G}_k berechnet:

$$\mathbf{G}_k = \left. \frac{\partial \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{0})}{\partial \boldsymbol{\nu}_k} \right|_{\mathbf{x}_{k-1} = \hat{\mathbf{x}}_{k-1}} \quad (2.39)$$

Das Algorithmus des Extended Kalman Filters ist eine Variante vom normalen Kalman Filter, wenn man anstelle der linearen Systemmatrizen die entsprechenden zeitvarianten Jacobi Matrizen verwendet (siehe Tabelle 2.2).

Initialisierung:

$$\hat{\mathbf{x}}_0 = E\{\mathbf{x}_0\}, \quad \mathbf{P}_0 = Cov\{\mathbf{x}_0\} = E\{\mathbf{x}_0 \mathbf{x}_0^T\}$$

Für $k = 1, \dots, \infty$:

Prädiktionsschritt:

$$\begin{aligned} \hat{\mathbf{x}}_k^- &= \mathbf{f}_k(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_k) \\ \mathbf{P}_k^- &= \mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T \end{aligned}$$

Innovationsschritt:

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \\ \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}_k^-, \mathbf{u}_k)) \\ \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \end{aligned}$$

Tabelle 2.2: Algorithmus des Extended Kalman Filters

Denn viele Systeme in der realen Welt nichtlinear sind, ist das EKF bei Zustandsschätzung sehr verbreitet. Jedoch hat das EKF die folgenden Probleme:

1. Die Berechnung der Jacobi Matrizen ist manchmal sehr kompliziert, oder sogar nicht möglich, wenn das Modell nicht differenzierbar ist.
2. Der Zustand beim EKF lässt sich immer durch eine Normalverteilung mit dem Mittelwert und der Kovarianz beschreiben. Wenn der Zustand durch eine nichtlineare Systemfunktion propagiert wird, ändert sich die Gestalt der normalverteilten Wahrscheinlichkeitsdichte des Zustands [05]. Es ist nicht mehr aussagekräftig genug, den propagierten Zustand durch eine Normalverteilung zu beschreiben (siehe Abbildung 2.11).

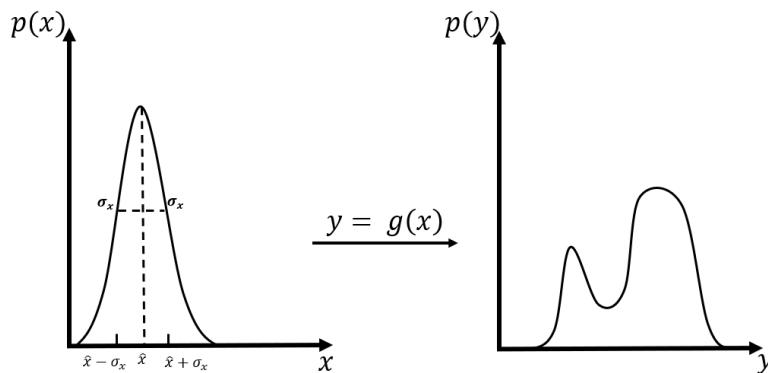


Abbildung 2.11: Verfälschung einer Normalverteilung nach der Propagierung durch eine nichtlineare Funktion $g(\dots)$

3. Beim EKF werden der Mittelwert und die Kovarianzmatrix des Zustands durch berechnete Jacobi Matrizen hergeleitet. Zur Berechnung der Jacobi Matrizen wird das nichtlineare System jedoch durch Taylorreihe 1. Ordnung um den Arbeitspunkt linearisiert. Je stärker die Nichtlinearität des Systems ist, desto größer wird der Approximationsfehler sein, also zusätzliche Unsicherheit wird eingebracht. In diesem Fall ist das EKF nicht optimal (im Sinne einer minimalen Kovarianz des Schätzfehlers). Im Extremfall kann das EKF divergent sein [04].

Das dritte Problem vom EKF kann häufig durch iteratives EKF (IEKF) gelöst werden, wodurch für jeden Zeitschritt die Linearisierung des nichtlinearen Systems durch iterative Filterschritte verbessert wird. Jedoch ist beim IEKF schwer, die Laufzeit schätzen zu können. Außerdem kann eine Konvergenz des IEKF nicht garantiert sein. Aber IEKF ist schon ein Verbesserungsverfahren vom EKF.

2.4.3 Unscented Kalman Filter

Unscented Kalman Filter (UKF) liefert ein ganz unterschiedliches Verfahren, um eine Kalman Filterung für nichtlineares System zu entwickeln. Beim UKF lässt sich eine

Verteilung des Zustandsvektors mit Mittelwert und Kovarianzmatrix durch mehrere charakteristische Punkte („Sigma-Punkte“) beschreiben.

Wenn der Zustandsvektor \mathbf{x} mit einer Dimension L einen Erwartungswert $\hat{\mathbf{x}}$ und eine Kovarianzmatrix \mathbf{P} hat, kann für diesen Zustandsvektor normalerweise $2L + 1$ Sigma-Punkte \mathbf{x}_i mit entsprechenden Wichtung W_i nach folgende Gleichungen generiert werden [04]:

$$\mathbf{x}_0 = \hat{\mathbf{x}} \quad (2.40)$$

$$\mathbf{x}_i = \hat{\mathbf{x}} + \left(\sqrt{(L + \lambda)\mathbf{P}} \right)_i \quad \text{für } i = 1, \dots, L \quad (2.41)$$

$$\mathbf{x}_i = \hat{\mathbf{x}} - \left(\sqrt{(L + \lambda)\mathbf{P}} \right)_{i-L} \quad \text{für } i = L + 1, \dots, 2L \quad (2.42)$$

$$W_0^{(m)} = \frac{\lambda}{L + \lambda} \quad (2.43)$$

$$W_0^{(c)} = \frac{\lambda}{L + \lambda} + (1 - \alpha^2 + \rho) \quad (2.44)$$

$$W_i^{(m)} = W_i^{(c)} = \frac{\lambda}{2(L + \lambda)} \quad \text{für } i = 1, \dots, 2L \quad (2.45)$$

wobei die Wichtungen $W_i^{(m)}$ und $W_i^{(c)}$ jeweils für die Rekonstruktion des Mittelwerts und der Kovarianz genutzt werden. Das Ergebnis der „Wurzel“ der symmetrischen Matrix $(L + \lambda)\mathbf{P}$ ist eine untere Dreiecksmatrix, die durch Cholesky-Zerlegung berechnet wird:

$$\text{Falls } \mathbf{A}\mathbf{A}^T = (L + \lambda)\mathbf{P}, \text{ dann } \sqrt{(L + \lambda)\mathbf{P}} = \mathbf{A} \quad (2.46)$$

Das Zeichen $(\mathbf{A})_i$ bedeutet hier i -te Spalte der Matrix \mathbf{A} .

$\lambda = \alpha^2(L + \kappa) - L$ ist ein Skalierungsparameter. α, κ, ρ sind hier drei Tuningparameter bei Generierung der Sigma-Punkte. Sie haben folgende Bedeutungen [04]:

1. α bestimmt Streubreite der Sigma-Punkte um den Mittelwert, meist kleiner Wert > 0 , z.B. $\alpha = 10^{-3}$.
2. κ ist ein zweiter Skalierungsparameter, häufig $\kappa = 0$ gesetzt.
3. ρ berücksichtigt das Vorwissen über die Verteilung vom Zustand \mathbf{x} , z.B. für Normalverteilungen $\rho = 2$ ist optimal.

Um den Mittelwert und die Kovarianzmatrix durch nichtlineares System zu propagieren, braucht das EKF eine Linearisierung des nichtlinearen Systems, während eine Linearisierung beim UKF nicht benötigt wird. Denn beim UKF wird die generierten Sigma-Punkte des Zustandsvektors direkt durch das nichtlineare System transformiert. Danach kann man aus solchen propagierten Sigma-Punkten den Mittelwert und die Kovarianz der Verteilung des propagierten Zustandsvektors mit Gewichten $W_i^{(m)}$ und $W_i^{(c)}$ wieder rekonstruieren.

Zum Beispiel, wenn der Zustandsvektor \mathbf{x} durch eine nichtlineare Funktion $\mathbf{f}(\dots)$ in einen neuen Zustandsvektor \mathbf{y} zu transformieren ist,

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) \quad (2.47)$$

ist es schwer den Mittelwert und die Kovarianz von \mathbf{y} direkt zu bestimmen. Aber wenn man zuerst die Sigma-Punkte χ_i des Zustandsvektors \mathbf{x} statt des Zustandsvektors \mathbf{x} durch diese nichtlineare Funktion $f(\dots)$ in die neuen Sigma-Punkten γ_i transformiert, können der Mittelwert $\bar{\mathbf{y}}$ und die Kovarianz \mathbf{P}_y vom neuen Zustandsvektor \mathbf{y} über transformierte Sigma-Punkte γ_i wie folgend berechnet werden:

$$\gamma_i = f(\chi_i) \quad \text{für } i = 0, \dots, 2L \quad (2.48)$$

$$\bar{\mathbf{y}} \approx \sum_{i=0}^{2L} W_i^{(m)} \gamma_i \quad (2.49)$$

$$\mathbf{P}_y \approx \sum_{i=0}^{2L} W_i^{(c)} (\gamma_i - \bar{\mathbf{y}}) (\gamma_i - \bar{\mathbf{y}})^T \quad (2.50)$$

Das ganze Prozess nennt man „Unscented Transform“ (Julier and Uhlmann, 1996). Man kann sehen, dass beim Unscented Transform eine Näherung für die Verteilung des Zustandsvektors \mathbf{x} anstelle für das nichtlineare System $f(\mathbf{x})$ durchgeführt wird (siehe Abbildung 2.12).

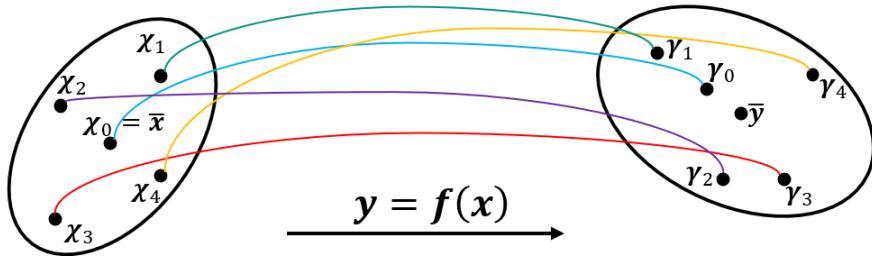


Abbildung 2.12: Unscented Transform durch eine nichtlineare Funktion $f(\dots)$

Beim Algorithmus des UKF wird solche Unscented Transform für jeden Zeitschritt zweimal durchgeführt, weil solche Sigma-Punkte zuerst durch das nichtlineare Prozessmodell, danach durch das nichtlineare Beobachtungsmodell propagiert werden. Für das nichtlineare Prozessmodell (2.35) und Beobachtungsmodell (2.36) kann der Algorithmus des UKF nach [04] und [06] wie Tabelle 2.3 dargestellt entworfen werden (Anmerkung: „aug“ = erweitert, engl. *augmented*):

Initialisierung:

$$\hat{\mathbf{x}}_0 = E\{\mathbf{x}_0\}, \mathbf{P}_0 = Cov\{\mathbf{x}_0\} = E\{\mathbf{x}_0 \mathbf{x}_0^T\}$$

$$\hat{\mathbf{x}}_0^{aug} = \begin{bmatrix} E\{\mathbf{x}_0\} \\ E\{\mathbf{v}_0\} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{x}}_0 \\ \mathbf{0} \end{bmatrix}, \mathbf{P}_0^{aug} = \begin{bmatrix} \mathbf{P}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_0 \end{bmatrix}$$

Für $k = 1, \dots, \infty$:

Generierung der Sigma-Punkte:

$$\hat{\mathbf{x}}_{0,k-1}^{aug} = \hat{\mathbf{x}}_{k-1}^{aug}$$

$$\begin{aligned}\hat{x}_{i,k-1}^{aug} &= \hat{x}_{k-1}^{aug} + \left(\sqrt{(L + \lambda) P_{k-1}^{aug}} \right)_i && \text{für } i = 1, \dots, L \\ \hat{x}_{i,k-1}^{aug} &= \hat{x}_{k-1}^{aug} - \left(\sqrt{(L + \lambda) P_{k-1}^{aug}} \right)_{i-L} && \text{für } i = L + 1, \dots, 2L \\ \rightarrow \quad \hat{\chi}_{k-1}^{aug} &= \begin{bmatrix} \hat{\chi}_{k-1}^x \\ \hat{\chi}_{k-1}^v \end{bmatrix} = [\hat{x}_{0,k-1}^{aug} \ \hat{x}_{1,k-1}^{aug} \dots \hat{x}_{2L,k-1}^{aug}]\end{aligned}$$

Prädiktionsschritt:

$$\begin{aligned}\hat{\chi}_{i,k}^{x-} &= f_k(\hat{\chi}_{i,k-1}^x, \hat{\chi}_{i,k-1}^v, u_k) && \text{für } i = 0, \dots, 2L \\ \hat{x}_k^- &= \sum_{i=0}^{2L} W_i^{(m)} \hat{\chi}_{i,k}^{x-} \\ P_k^- &= \sum_{i=0}^{2L} W_i^{(c)} (\hat{\chi}_{i,k}^{x-} - \hat{x}_k^-) (\hat{\chi}_{i,k}^{x-} - \hat{x}_k^-)^T\end{aligned}$$

Innovationsschritt:

$$\begin{aligned}\hat{\mathbf{z}}_{i,k}^{x-} &= h_k(\hat{\chi}_{i,k}^{x-}, u_k) && \text{für } i = 0, \dots, 2L \\ \hat{\mathbf{z}}_k^- &= \sum_{i=0}^{2L} W_i^{(m)} \hat{\mathbf{z}}_{i,k}^{x-} \\ P_{zz,k} &= \sum_{i=0}^{2L} W_i^{(c)} (\hat{\mathbf{z}}_{i,k}^{x-} - \hat{\mathbf{z}}_k^-) (\hat{\mathbf{z}}_{i,k}^{x-} - \hat{\mathbf{z}}_k^-)^T + R_k, \text{ hier } R_k \text{ ist additiv.} \\ P_{xz,k} &= \sum_{i=0}^{2L} W_i^{(c)} (\hat{\chi}_{i,k}^{x-} - \hat{x}_k^-) (\hat{\mathbf{z}}_{i,k}^{x-} - \hat{\mathbf{z}}_k^-)^T \\ K_k &= P_{xz,k} P_{zz,k}^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k (\mathbf{z}_k - \hat{\mathbf{z}}_k^-) \\ P_k &= P_k^- - K_k P_{zz,k} K_k^T\end{aligned}$$

wobei:

L = Dimension des erweiterten Zustandsvektor

$\lambda = \alpha^2(L + \kappa) - L$ = Skalierungsparameter

$x_k^{aug} = \begin{bmatrix} x_k \\ v_k \end{bmatrix}$ = wahrer Wert des um das Prozessrauschen v_k erweiterten Zustandsvektors

$\hat{x}_k^{aug} = \begin{bmatrix} \hat{x}_k \\ \hat{v}_k \end{bmatrix}$ = Schätzwert (Mittelwert) des erweiterten Zustandsvektors

$\hat{\chi}_{i,k}^x, \hat{\chi}_{i,k}^v$ = Sigma-Punkte des Zustandsvektors \hat{x}_k und des Prozessrauschens \hat{v}_k

$\hat{\mathbf{x}}_{i,k}^{aug} = \begin{bmatrix} \hat{\mathbf{x}}_{i,k}^x \\ \hat{\mathbf{x}}_{i,k}^v \end{bmatrix}$ = um die Sigma-Punkte des Prozessrauschen \mathbf{v}_k erweiterte Sigma-Punkte

$\hat{\mathbf{x}}_k^{aug} = \begin{bmatrix} \hat{\mathbf{x}}_k^x \\ \hat{\mathbf{x}}_k^v \end{bmatrix} = [\hat{\mathbf{x}}_{0,k}^{aug} \ \hat{\mathbf{x}}_{1,k}^{aug} \dots \hat{\mathbf{x}}_{2L,k}^{aug}]$ = Sigma-Punkte Matrix

\mathbf{Q}_k = Kovarianzmatrix des nicht additiven Prozessrauschen

\mathbf{R}_k = Kovarianzmatrix des additiven Beobachtungsrauschen

$\hat{\mathbf{x}}_{i,k}^{x-}$ = Sigma-Punkte nach der Transformation durch Prozessmodell

$\hat{\mathbf{x}}_k^-$ = Prädizierter Zustandsvektor

\mathbf{P}_k^- = Fehlerkovarianz des prädizierten Zustandsvektors

$\hat{\mathbf{Z}}_{i,k}^{x-}$ = Sigma-Punkte nach der Transformation durch Beobachtungsmodell

$\hat{\mathbf{z}}_k^-$ = Prädizierte virtuelle Beobachtung

$\mathbf{P}_{zz,k}$ = Fehlerkovarianz der prädizierten virtuellen Beobachtung

$\mathbf{P}_{xz,k}$ = Kreuz-Kovarianzmatrix zwischen prädiziertem Zustandsvektor und prädizierter virtuellen Beobachtung

\mathbf{z}_k = Beobachtung

$\mathbf{z}_k - \hat{\mathbf{z}}_k^-$ = Innovation

$\hat{\mathbf{x}}_0$ = Erwartungswert des Anfangszustandsvektors

\mathbf{P}_0 = Anfangswert der Konvarianzmatrix des Zustandsvektors

Tabelle 2.3: Algorithmus des Unscented Kalman Filters

Wie in Abschnitt 2.4.1 bereits erwähnt, wird es häufig so angenommen, dass das Beobachtungsrauschen \mathbf{u}_k additives Rauschen, und das Prozessrauschen \mathbf{v}_k nicht additives Rauschen und durch das nichtlineare Prozessmodell transformierbar ist. Deshalb wird der Zustandsvektor \mathbf{x}_k im Algorithmus um auf das nichtlineare Prozessmodell wirkenden Prozessrauschen \mathbf{v}_k erweitert. Dadurch kann die nichtlineare Transformation des Prozessrauschen auch im Algorithmus berücksichtigt werden. Für jeden Zeitschritt werden zunächst die Sigma-Punkte für den Zustandsvektor, der für den letzten Zeitschritt geschätzt ist, generiert. Danach werden der Prädiktionsschritt und der Innovationsschritt durchgeführt, indem die Sigma-Punkte durch das Prozessmodell und das Beobachtungsmodell propagiert werden.

Die Rechenkomplexität des UKF hat ein gleiches Niveau wie das EKF, jedoch ist das UKF im Vergleich zum EKF bei Behandlung des nichtlinearen Systems viel genauer [04] [05]. Außerdem benötigt das UKF keine Berechnung der Jacobi Matrizen. Dadurch kann das UKF sehr flexible in viele komplexen nichtlinearen Systeme eingesetzt werden.

3 Konzeption der Informationsfusion für relative Positionierung beim Platooning

Um sicherere und robustere Schätzung der relativen Pose zweier Busse zu erhalten, kann ein Fusionssystem normalerweise so entworfen werden, dass die Beobachtungen aus Kamera- und Lidar-Tracker durch einen rekursiven Least-Squares- (RLS-) Schätzer fusioniert werden, wobei die Schätzung der relativen Pose rekursiv durch Hinzunahme neuer Beobachtungen verbessert werden. Damit wird eine beste Schätzung mit kleinster Unsicherheit ermittelt [03]. Aber die Schätzung der relativen Pose ist mit dieser Methode nicht erzielbar, wenn keine Beobachtungen ins Fusionssystem reinkommt. Um die relative Pose des vorausfahrenden Busses für die Szenarien, in denen keine Beobachtungen vom Kamera- und Lidar-Tracker gebietet werden, auch noch zu schätzen, kann man die Technologie der Kalman-Filterung verwenden.

In dieser Arbeit wird das Fusionssystem mittels der Technologie der Kalman Filterung bzw. des Unscented Kalman Filters (UKF) entworfen. Das Kalman-Filter ist auch ein rekursiver Least-Squares-Schätzer, jedoch fusioniert das Kalman-Filter nicht nur die Beobachtungen, sondern auch ein Prozessmodell, das sich durch ein kinematisches Modell sowie mehrere fahrdynamischen Kenngrößen, z.B. longitudinale Geschwindigkeit, Giergeschwindigkeit und Schwimmwinkel, zweier Busse beschreiben lässt.

Manche fahrdynamische Kenngrößen können direkt durch Sensoren gemessen werden, wie Beispielsweise kann der Giergeschwindigkeit (Drehung um die z-Achse durch den Fahrzeugschwerpunkt) des Busses durch IMU direkt gemessen werden. Aber manche fahrdynamische Kenngrößen sind nicht messbar, oder nur mit großen Aufwand und Kosten messbar, sie werden aus den Messdaten der verschiedenen internen Sensoren, die auf dem jeweiligen Bus montiert sind, extrahiert. Beispielweise kann der longitudinale Geschwindigkeit des Busses aus den Messdaten der IMU, Radencoder und Lenkwinkelsensor rekonstruiert werden.

Da nur der verfolgende Bus mit einem Prozessor ausgestattet ist, werden die Messdaten der internen Sensoren des führenden Busses zuerst durch eine Fahrzeug-zu-Fahrzeug (V2V) Kommunikation in den Prozessor des verfolgenden Busses übertragen, und diese übertragenen Messdaten und die Messdaten der internen Sensoren des verfolgenden Busses werden dort je nach Bedarf zur Extrahierung der fahrdynamischen Kenngrößen verwendet, oder direkt ins Fusionssystem für die Schätzung der relativen Pose eingesetzt werden.

Auf diese Weise werden die Beobachtungen der Umfeldsensoren (Lidar, Kamera) sowie die Messdaten der internen Sensoren (IMU, Radencoder, Lenkwinkelsensor) des führenden und verfolgenden Busses im Prozessor des verfolgenden Busses zentralisiert fusioniert (siehe Abbildung 3.1).

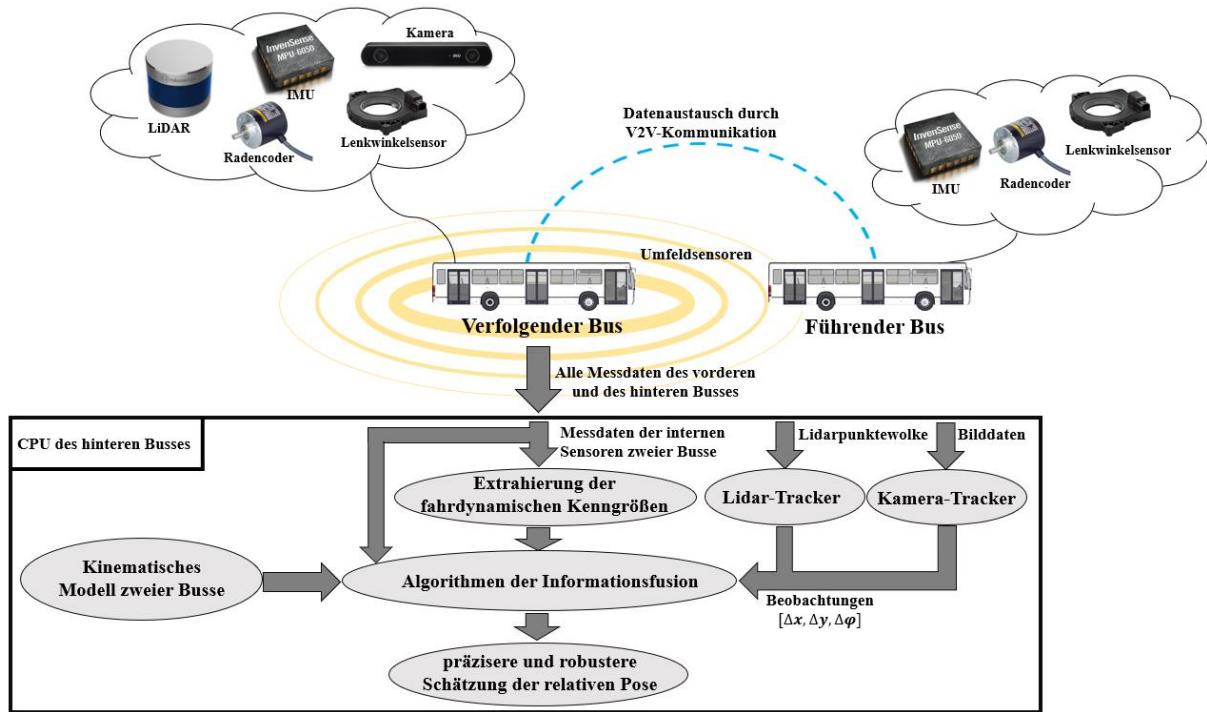


Abbildung 3.1: Konzept der Informationsfusion zur besseren Schätzung der relativen Pose

Ein ähnliches UKF basiertes Fusionskonzept für die kooperative relative Positionierung zweier Fahrzeuge mit dem Radar, GNSS, IMU, Geschwindigkeitssensor und V2V-Kommunikation findet sich in [21]. In [23] werden lediglich die Daten vom Lidar und Radar mittels des EKF/UKF fusioniert, um die Fahrzeuge auf der Straße zu detektieren und zu tracken. [22] nutzt die V2V-Kommunikation, um die Messdaten der GPS/INS vom Ziel-Fahrzeug nach Ego-Fahrzeug zu übertragen, und fusioniert alle Messdaten der GPS/INS zweier Fahrzeuge mittels des Partikel Filters, um die relative Pose zweier Fahrzeuge zu schätzen.

In diesem Kapitel wird zunächst das kinematische Modell für die relative Bewegungen zweier Busse erläutert. Anschließend wird in Abschnitt 3.2 und 3.3 vorgestellt, wie man aus den messbaren Größen die nicht messbaren fahrdynamischen Kenngrößen, bzw. die longitudinale Geschwindigkeit und den Schwimmwinkel zweier Busse, rekonstruiert. In Abschnitt 3.4 erfolgt ein UKF basiertes Informationsfusionskonzept, welches die Beobachtungen aus Kamera- und Lidar-Tracker, das kinematische Modell sowie die fahrdynamischen Kenngrößen zusammen fusioniert. Zuletzt befasst sich Abschnitt 3.5 mit der Methode der Zeitsynchronisierung aller Sensoren.

3.1 Kinematisches Modell der relativen Bewegung zweier Busse

Um die relative Pose zweier Busse ohne Beobachtungen auch weiter zu schätzen, ist es notwendig, die neue relative Pose zum Zeitpunkt t_k mittels des kinematischen Modells der

relativen Bewegung zweier Busse aus der relativen Pose zum letzten Zeitpunkt t_{k-1} herzuleiten.

Man berücksichtigt nun zuerst zwei Busse im Koordinatensystem $O_{k-1}X_{k-1}Y_{k-1}$ zum Zeitpunkt t_{k-1} (siehe Abbildung 3.2). Der Ursprung O_{k-1} dieses Koordinatensystems liegt im Schwerpunkt des verfolgende Busses. Die X_{k-1} -Achse dieses Koordinatensystems ist immer mit der Längsachse des verfolgende Busses überlagert.

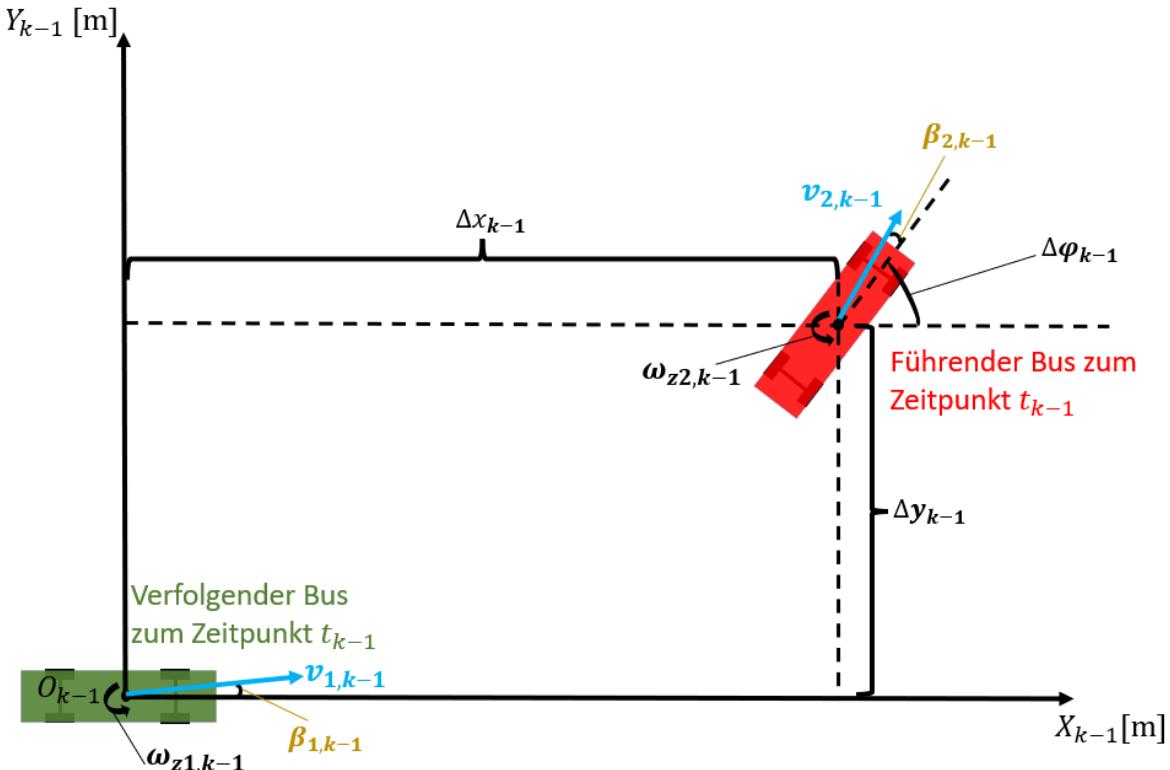


Abbildung 3.2: Darstellung der relativen Pose zweier Busse zum Zeitpunkt t_{k-1}

Es wird zuerst angenommen, dass die relative Position $[\Delta x_{k-1}, \Delta y_{k-1}]$ zwischen den Schwerpunkten zweier Busse sowie der relative Gierwinkel $\Delta\varphi_{k-1}$ zwischen den beiden Bussen zum Zeitpunkt t_{k-1} bekannt sind. Es ist auch bekannt, dass der verfolgende und der führende Bus zum Zeitpunkt t_{k-1} die Giergeschwindigkeit $\omega_{z1,k-1}$ und $\omega_{z2,k-1}$ im Schwerpunkt hat. Außerdem bewegt sich der verfolgende und der führende Bus jeweils mit einer bekannten Schwerpunktgeschwindigkeit $v_{1,k-1}$ und $v_{2,k-1}$, wobei der Schwimmwinkel $\beta_{1,k-1}$ und $\beta_{2,k-1}$ entsteht. (Anmerkung: Tiefstellung „1“ = Variable des verfolgenden Busses, Tiefstellung „2“ = Variable des führenden Busses)

Nun werden die Bewegungen zweier Busse vom Zeitpunkt t_{k-1} zum Zeitpunkt t_k berücksichtigt (siehe Abbildung 3.3). Die Zeitdauer der Bewegungen ist $\Delta t = t_k - t_{k-1}$.

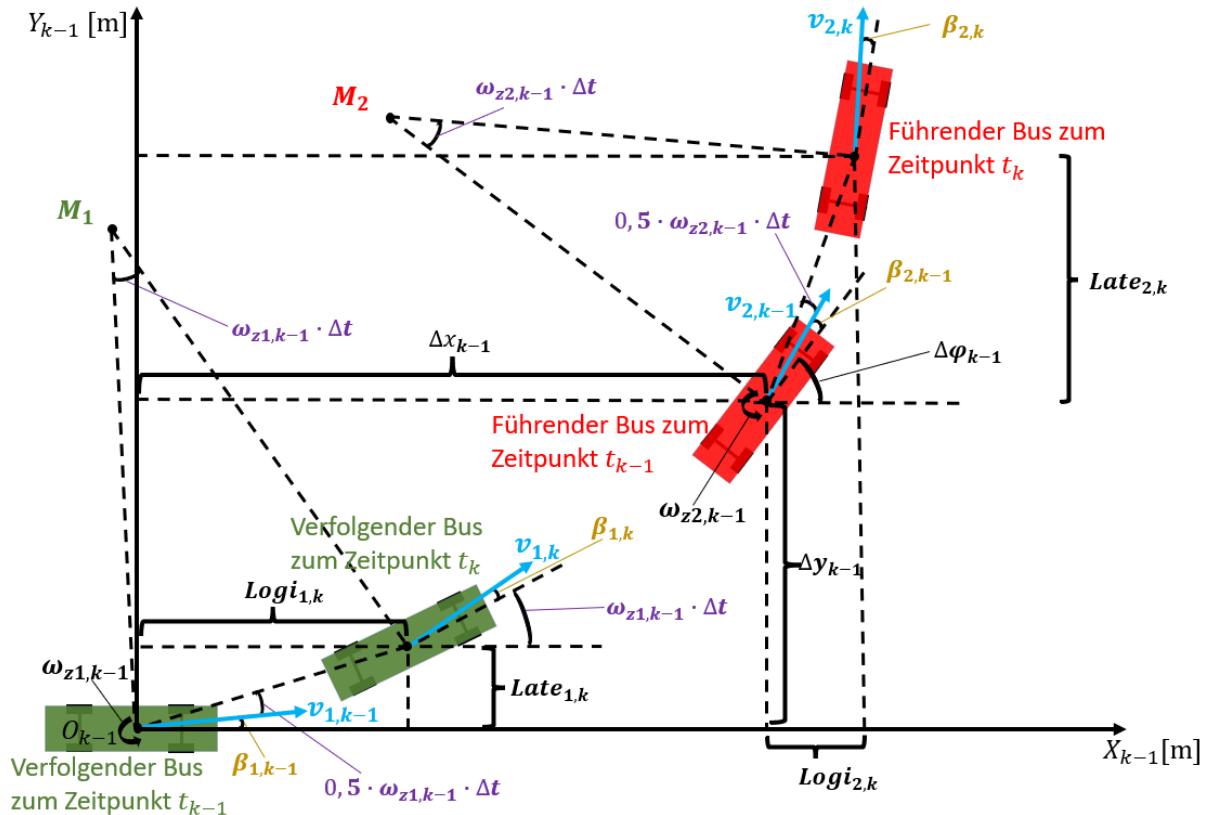


Abbildung 3.3: Darstellung der Bewegungen zweier Busse vom Zeitpunkt t_{k-1} zum Zeitpunkt t_k

Wird der Bus als ein starrer Körper betrachtet, lässt sich die momentane Bewegung des Busses als eine Kreisbewegung um einen Momentanpol M mit dem Radius beschreiben. Die Bewegungsrichtung oder die Richtung der Schwerpunktgeschwindigkeit des Busses muss dabei senkrecht auf den Strahlen zum Momentanpol stehen. Nach [17] kann die Translation des verfolgenden Busses um X_{k-1} - und Y_{k-1} -Achse (siehe Abbildung 3.3) relativ zur seinen letzten Busposition wie folgende Gleichungen ermittelt werden:

$$Logi_{1,k} = v_{1,k-1} \cdot \Delta t \cdot \cos(0,5 \cdot \omega_{z1,k-1} \cdot \Delta t + \beta_{1,k-1}) \quad (3.1)$$

$$Late_{1,k} = v_{1,k-1} \cdot \Delta t \cdot \sin(0,5 \cdot \omega_{z1,k-1} \cdot \Delta t + \beta_{1,k-1}) \quad (3.2)$$

Der führende Bus bewegt sich auch um X_{k-1} - und Y_{k-1} -Achse relativ zur seinen letzten Busposition nach folgenden Gleichungen:

$$Logi_{2,k} = v_{2,k-1} \cdot \Delta t \cdot \cos(\Delta\varphi_{k-1} + 0,5 \cdot \omega_{z2,k-1} \cdot \Delta t + \beta_{2,k-1}) \quad (3.3)$$

$$Late_{2,k} = v_{2,k-1} \cdot \Delta t \cdot \sin(\Delta\varphi_{k-1} + 0,5 \cdot \omega_{z2,k-1} \cdot \Delta t + \beta_{2,k-1}) \quad (3.4)$$

Die neue relative Position zweier Busse im Koordinatensystem $O_{k-1}X_{k-1}Y_{k-1}$ ergibt sich zu:

$$\Delta x_k^{\#} = \Delta x_{k-1} + Logi_{2,k} - Logi_{1,k} \quad (3.5)$$

$$\Delta y_k^{\#} = \Delta y_{k-1} + Late_{2,k} - Late_{1,k} \quad (3.6)$$

Diese neue relative Position $[\Delta x_k^{\#}, \Delta y_k^{\#}]$ zweier Busse wird mit Bezug auf das Koordinatensystem $O_{k-1}X_{k-1}Y_{k-1}$ ermittelt. Aber vom Zeitpunkt t_{k-1} zum Zeitpunkt t_k hat

der verfolgende Bus sich bewegt und um $\omega_{z1,k-1} \cdot \Delta t$ dreht. Deshalb ist das neue Koordinatensystem $O_k X_k Y_k$ zum Zeitpunkte t_k auch anders als das alte Koordinatensystem $O_{k-1} X_{k-1} Y_{k-1}$ (siehe Abbildung 3.4). Man muss daher die relative Position $[\Delta x_k^\#, \Delta y_k^\#]$ ins neue Koordinatensystem $O_k X_k Y_k$ transformieren:

$$\Delta x_k = \cos(\omega_{z1,k-1} \cdot \Delta t) \cdot \Delta x_k^\# + \sin(\omega_{z1,k-1} \cdot \Delta t) \cdot \Delta y_k^\# \quad (3.7)$$

$$\Delta y_k = -\sin(\omega_{z1,k-1} \cdot \Delta t) \cdot \Delta x_k^\# + \cos(\omega_{z1,k-1} \cdot \Delta t) \cdot \Delta y_k^\# \quad (3.8)$$

Die ermittelte relative Position $[\Delta x_k, \Delta y_k]$ ist der longitudinale Abstand und der lateraler Versatz zwischen Schwerpunkten zweier Busse zum Zeitpunkt t_k .

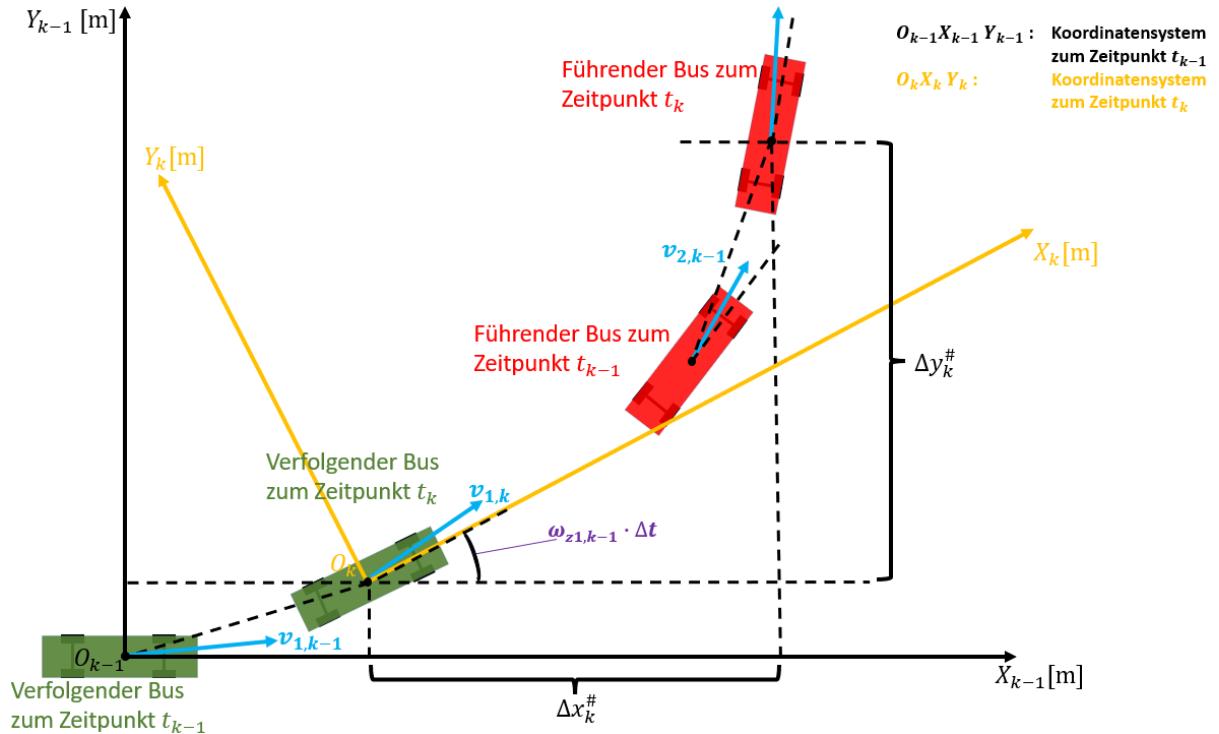


Abbildung 3.4: Darstellung der Koordinatentransform vom Zeitpunkt t_{k-1} zum Zeitpunkt t_k

Die Differenz des Gierwinkels zum Zeitpunkt t_k lässt sich durch die folgende Formel berechnen:

$$\Delta \varphi_k = \Delta \varphi_{k-1} + (\omega_{z2,k-1} - \omega_{z1,k-1}) \cdot \Delta t \quad (3.9)$$

Die Formeln (3.1) ~ (3.9) bilden dabei das kinematische Modell der relativen Bewegung zweier Busse. Man kann sehen, dass mehrere dynamische Kenngrößen ins kinematische Modell eingesetzt werden müssen, um die neue relative Pose $[\Delta x_k, \Delta y_k, \Delta \varphi_k]$ aus der alten relativen Pose $[\Delta x_{k-1}, \Delta y_{k-1}, \Delta \varphi_{k-1}]$ zu ermitteln. Diese dynamischen Kenngrößen sind die Schwerpunktgeschwindigkeit v , der Schwimmwinkel β und die Giergeschwindigkeit ω_z des jeweiligen Busses. Die Giergeschwindigkeit ω_z des Busses kann direkt durch IMU gemessen werden, während die Schwerpunktgeschwindigkeit v und der Schwimmwinkel β aus den Messdaten der internen Sensoren des jeweiligen Busses rekonstruiert werden müssen. Die

Schwerpunktgeschwindigkeit v kann über die folgende Formel ermittelt werden, sobald der Schwimmwinkel β und die longitudinale Geschwindigkeit v_x des Busses bekannt sind:

$$v = \sqrt{v_x^2 + v_y^2} = \sqrt{v_x^2 + (v_x \cdot \tan \beta)^2} = v_x \cdot \sqrt{1 + \tan^2 \beta} \quad (3.10)$$

Deshalb werden nun die longitudinale Geschwindigkeit v_x und der Schwimmwinkel β gesucht. Sie werden aus den Messdaten der Radencoder, IMU und Lenkwinkelsensor des jeweiligen Busses rekonstruiert, indem ein Geschwindigkeit-Schätzer und ein Schwimmwinkelschätzer verwendet werden. Ein Blockschaltbild findet sich in Abbildung 3.5, wobei $\omega_{vl}, \omega_{vr}, \omega_{hl}, \omega_{hr}$ die gemessene Rotationsgeschwindigkeit der linken und rechten Vorderräder sowie der linken und rechten Hinterräder bezeichnet, und δ_{vl}, δ_{vr} sind der Lenkwinkel der linken und rechten Vorderräder. Die longitudinale Beschleunigung, die laterale Beschleunigung sowie die Giergeschwindigkeit werden mit der IMU gemessen und durch $a_{x,imu}, a_{y,imu}, \omega_{z,imu}$ bezeichnet. Die Schätzwerte für die longitudinale Geschwindigkeit und den Schwimmwinkel sind $v_{x,est}$ und β_{est} (Anmerkung: „est“ = estimation).

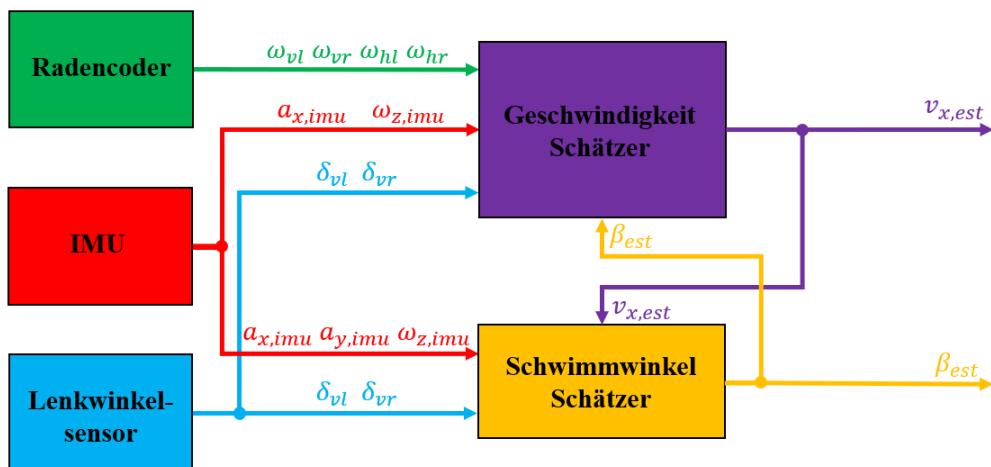


Abbildung 3.5: Rekonstruktion der longitudinalen Geschwindigkeit und des Schwimmwinkels des jeweiligen Busses

In Abschnitt 3.2 und 3.3 wird die Schätzung der longitudinalen Geschwindigkeit und des Schwimmwinkels des Busses genauer erläutert.

3.2 Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs

Die longitudinale Geschwindigkeit v_x des Fahrzeugs ist häufig mit den Sensoren nicht direkt messbar, oder nur mit großen Aufwand und Kosten direkt messbar. Sie wird häufig mittels der Messdaten der Radencoder oder der Beschleunigungssensoren geschätzt. Jedoch hat jeder einzelne Sensor seine eigene Limitierung und Nachteile.

Zum Beispiel kann die longitudinale Geschwindigkeit des Fahrzeugs häufig mit der Geschwindigkeit der Räder geschätzt werden, weil die Radgeschwindigkeit meistens mit der longitudinalen Geschwindigkeit des Fahrzeugs identisch ist. Dabei entstehen keine akkumulierte Fehler bei Schätzung der longitudinalen Geschwindigkeit. Aber wenn ein Rad einen Radschlupf hat, kann die Geschwindigkeit dieses Rades von der longitudinalen Geschwindigkeit des Fahrzeugs stark abweichen. In diesem Fall können die Messdaten dieses Rades nicht zur Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs verwendet, bis dieses Rad ohne Radschlupf weiter fahren kann.

Ist die Ausgangsgeschwindigkeit des Fahrzeugs bekannt, kann auch über die zeitliche Integration der einzelnen mit dem Beschleunigungssensor gemessenen longitudinalen Beschleunigung a_x die longitudinale Geschwindigkeit v_x des Fahrzeugs bestimmt werden. Im Vergleich zu Radencoder wird der Beschleunigungssensor nur wenig von der Umwelt und dem Zustand der Fahrbahn beeinflusst. Jedoch enthält jeder Messwert des Beschleunigungssensors aufgrund der Drift oder Rauschen eine gewisse Messabweichung. Diese Messabweichungen summieren sich bei der Integration und können mit der Zeit erhebliche Abweichungen bei Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs verursachen.

Um die Limitierung und Nachteile der einzelnen Sensoren zu überwinden und seine Vorteile auszunutzen, wird meist komplementäre Fusion für solche Sensoren erforderlich. In der Literatur [18] findet sich eine grundsätzliche Methode, die die Radencoder und die IMU mittels des Kalman Filters fusioniert, um die longitudinale Geschwindigkeit des Fahrzeugs besser zu schätzen. Abbildung 3.6 zeigt das Blockschaltbild des Geschwindigkeit-Schätzers nach [18], der aus einem Kalman Filter und einem Detektor des Radschlupfs besteht.

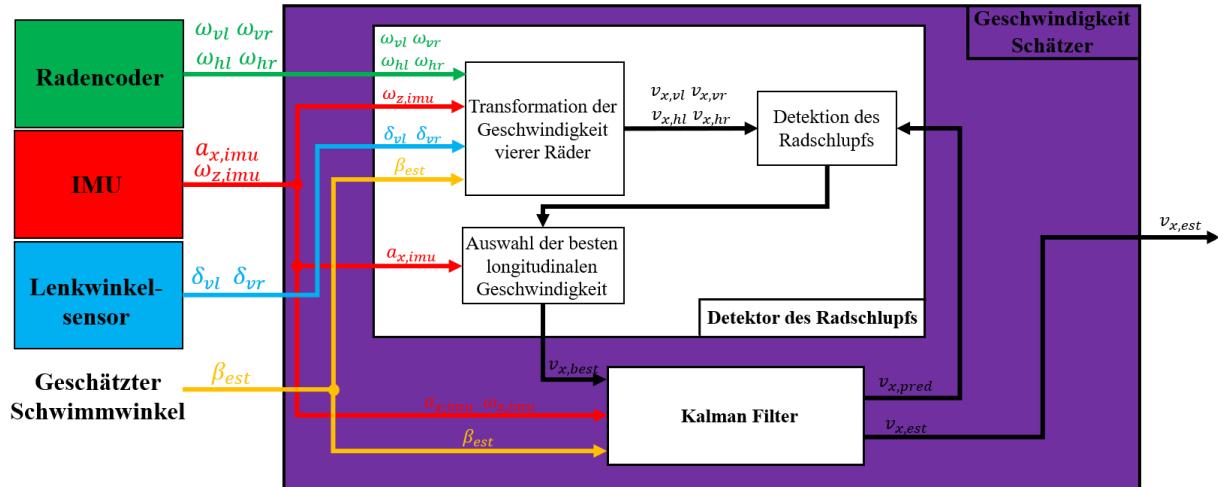


Abbildung 3.6: Blockschaltbild des Geschwindigkeit-Schätzers nach [18]

Im Detektor des Radschlupfs werden die Geschwindigkeiten vierer Räder ($v_{vl}, v_{vr}, v_{hl}, v_{hr}$) zunächst vom reifenfesten Koordinatensystem ins fahrzeugfeste Koordinatensystem transformiert (siehe Abschnitt 3.2.1). Anschließend wird jedes Rad geprüft, ob es einen Radschlupf hat (siehe Abschnitt 3.2.2). Wenn das Rad einen Radschlupf hat, wird es nicht mehr als Kandidaten für die Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs

berücksichtigt. Unter den restlichen Kandidaten ohne Radschlupf wird die beste longitudinalen Geschwindigkeit des Fahrzeugs $v_{x,best}$ ausgewählt. Sie dient als eine beste Beobachtung für die tatsächliche longitudinale Geschwindigkeit des Fahrzeugs v_x (siehe Abschnitt 3.2.3). Zuletzt fusioniert das Kalman Filter die Messdaten der IMU mit der vom Detektor des Radschlupfs ermittelten Beobachtung (siehe Abschnitt 3.2.4). Ist keine geeignete Beobachtung vom Detektor des Radschlupfs geliefert, weil jedes Rad einen Radschlupf hat, wird die longitudinale Geschwindigkeit des Fahrzeugs in diesem Fall lediglich über die zeitliche Integration der longitudinalen Beschleunigung bestimmt. Daher ist die Frequenz der Schätzwerte für die longitudinale Geschwindigkeit des Fahrzeugs identisch mit der Messfrequenz der IMU.

3.2.1 Transformation der Radgeschwindigkeit

Die translatorische Radgeschwindigkeit ($v_{vl}, v_{vr}, v_{hl}, v_{hr}$) in m/s kann durch die Rotationsgeschwindigkeit des Rades in rad/s mal den Radradius in m ermittelt werden:

$$v_{ij} = \omega_{ij} \cdot R_{ij} \quad (3.11)$$

wobei $i = v, h$ die Vorder- und Hinterachse, und $j = l, r$ die linken und rechten Räder bezeichnet. Zum Beispiel bedeutet v_{vl} die Radgeschwindigkeit des linken Vorderrades. ω_{ij} und R_{ij} sind die entsprechende Rotationsgeschwindigkeit und der Radius des Rades.

Zur Vereinfachung der Analyse wird in diesem Abschnitt angenommen, dass bei jedem Rad kein Schräglauftwinkel entsteht, das bedeutet, dass jede Radgeschwindigkeit v_{ij} in die X-Richtung des jeweiligen reifenfesten Koordinatensystems gerichtet ist (siehe Abbildung 3.7). Die Lenkwinkel der beiden Vorderräder (δ_{vl}, δ_{vr}) werden mit dem Lenkwinkelsensor bestimmt, während die Lenkwinkel der beiden Hinterräder (δ_{hl}, δ_{hr}) immer Null bleiben.

In Abschnitt 2.3.2 ist der Abstand mit dem absoluten Betrag vom Schwerpunkt zur Vorder- und Hinterachse durch l_v und l_h bezeichnet. Um zu unterscheiden, ob die Achse in die longitudinale Richtung vor dem Schwerpunkt oder nach dem Schwerpunkt liegt, werden hier L_v und L_h definiert, welche absolute Größen mit l_v und l_h identisch sind, jedoch enthalten L_v und L_h ein zusätzliches Vorzeichen. L_v wird als positive Größe definiert, weil die Vorderachse vor dem Schwerpunkt liegt. Dementsprechend ist L_h negativ.

s_{vl}, s_{vr}, s_{hl} und s_{hr} sind der laterale Versatz vom Schwerpunkt zu jeweiligem Rad (engl. *half track*). Hier wird es definiert, dass s_{vl} und s_{hl} auf der linken Seite des Schwerpunkts die positiven Werte haben, und s_{vr}, s_{hr} sind negativ.

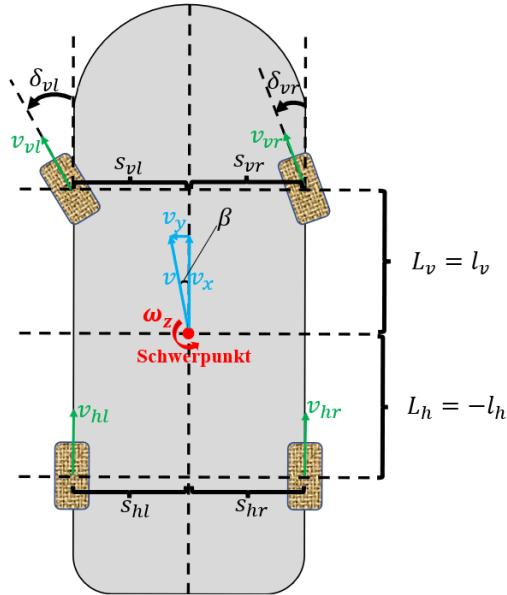


Abbildung 3.7: Transformation der Radgeschwindigkeit

Aus den Radgeschwindigkeiten kann man die longitudinale Geschwindigkeit des Fahrzeugs ermitteln, indem diese Radgeschwindigkeiten zuerst vom jeweiligen reifenfesten Koordinatensystem ins fahrzeugfeste Koordinatensystem, nämlich in den Schwerpunkt des Fahrzeugs, transformiert werden. Zur Analyse dieser Transformation wird zunächst der folgende Fall berücksichtigt. Das Fahrzeug bewegt sich nun mit einer Schwerpunktgeschwindigkeit v und dreht sich mit einer Giergeschwindigkeit ω_z . Für die Schwerpunktgeschwindigkeit v entsteht hier ein bekannter Schwimmwinkel β , damit die longitudinale und laterale Geschwindigkeit des Fahrzeugs v_x und v_y bestimmt werden können:

$$v_x = v \cdot \cos \beta; \quad v_y = v \cdot \sin \beta = v_x \cdot \tan \beta \quad (3.12)$$

Die Schwerpunktgeschwindigkeit v lässt sich über die folgende Formel vom fahrzeugfesten Koordinatensystem ins unterschiedliche reifenfeste Koordinatensystem transformieren:

$$\begin{aligned} v_{ij} &= (v_x - s_{ij} \cdot \omega_z) \cdot \cos \delta_{ij} + (v_y + L_i \cdot \omega_z) \cdot \sin \delta_{ij} \\ &= (v_x - s_{ij} \cdot \omega_z) \cdot \cos \delta_{ij} + (v_x \cdot \tan \beta + L_i \cdot \omega_z) \cdot \sin \delta_{ij} \end{aligned} \quad (3.13)$$

Zur Verdeutlichung wird hier ein Beispiel gegeben. Die Schwerpunktgeschwindigkeit v kann über die folgende Formel ins reifenfeste Koordinatensystem des linken Vorderrades ($i = v$ und $j = l$) transformiert werden [18]:

$$v_{vl} = (v_x - s_{vl} \cdot \omega_z) \cdot \cos \delta_{vl} + (v_x \cdot \tan \beta + L_v \cdot \omega_z) \cdot \sin \delta_{vl} \quad (3.14)$$

Formel (3.14) wird weiter umgestellt, so dass die longitudinale Geschwindigkeit v_x auf die linke Seite der Formel gebracht wird [18]:

$$v_x = \frac{v_{ij} + (s_{ij} \cdot \cos \delta_{ij} - L_i \cdot \sin \delta_{ij}) \cdot \omega_z}{\cos \delta_{ij} + \tan \beta \cdot \sin \delta_{ij}} \quad (3.15)$$

Über die Formel (3.15) lässt sich die beliebige Radgeschwindigkeit v_{ij} mit $i = v, h$ und $j = l, r$ vom jeweiligen reifenfesten Koordinatensystem in die longitudinalen Geschwindigkeit v_x des fahrzeugfesten Koordinatensystems transformieren. Die mit den Radgeschwindigkeiten vierer Räder ermittelten longitudinalen Geschwindigkeit des Fahrzeugs, welche durch $v_{x,vl}, v_{x,lr}, v_{x,hl}, v_{x,hr}$ bezeichnet sind, werden zuletzt ausgegeben (siehe Abbildung 3.6).

3.2.2 Detektion des Radschlups

Beim Übertragen einer Antriebskraft für Bremsen oder Beschleunigen kann eine Differenzgeschwindigkeit zwischen Reifen und Fahrbahn auftreten. Um diese Differenzgeschwindigkeit beschreiben zu können, wird der Radschlupf in % definiert.

Beim Bremsen des Fahrzeugs ist der Radschlupf s_{Brem} in % wie folgt definiert [12]:

$$s_{Brem} = \frac{v_{R,x} - R \cdot \omega}{v_{R,x}} \quad (3.16)$$

Hier bedeutet $v_{R,x}$ die translatorische Geschwindigkeit des Radmittelpunktes in X -Richtung des reifenfesten Koordinatensystems. $R \cdot \omega$ ist dabei die Radgeschwindigkeit. Da beim Bremsen die Radgeschwindigkeit $R \cdot \omega$ immer kleiner als $v_{R,x}$ ist, ist die Zähler in (3.16) immer positiv, somit ist der Schlupf s_{Brem} auch immer positiv. $s_B = 0\%$ bedeutet keinen Radschlupf, das heißt, die Radgeschwindigkeit $R \cdot \omega$ ist so schnell wie sich der Radmittelpunkt translatorisch bewegt. $s_B = 100\%$ bedeutet, dass sich das Rad nicht mehr dreht, während sich der Radmittelpunkt translatorisch noch bewegt.

Gleichfalls wird der Radschlupf s_{Besch} in % beim Beschleunigen des Fahrzeugs definiert [12]:

$$s_{Besch} = \frac{R \cdot \omega - v_{R,x}}{R \cdot \omega} \quad (3.17)$$

Beim Beschleunigen des Fahrzeugs ist die Radgeschwindigkeit $R \cdot \omega$ immer größer als $v_{R,x}$, somit ist der Radschlupf s_{Besch} immer positiv. $s_A = 0\%$ bedeutet keinen Radschlupf, und $s_A = 100\%$ bedeutet, dass sich der Radmittelpunkt translatorisch nicht bewegt, während sich das Rad noch dreht.

Tritt beim Rad ein Radschlupf auf, weicht die über die Formel (3.15) ermittelte longitudinale Geschwindigkeit des Fahrzeugs stark vom tatsächlichen Wert (siehe Abbildung 3.8). In diesem Fall soll diese Radgeschwindigkeit zur Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs beim Kalman Filter nicht fusioniert werden.

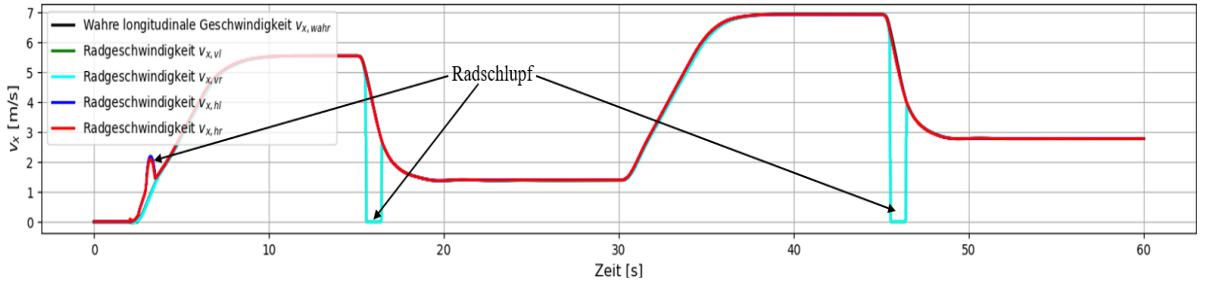


Abbildung 3.8: Radschlupf beim Bremsen und Beschleunigen

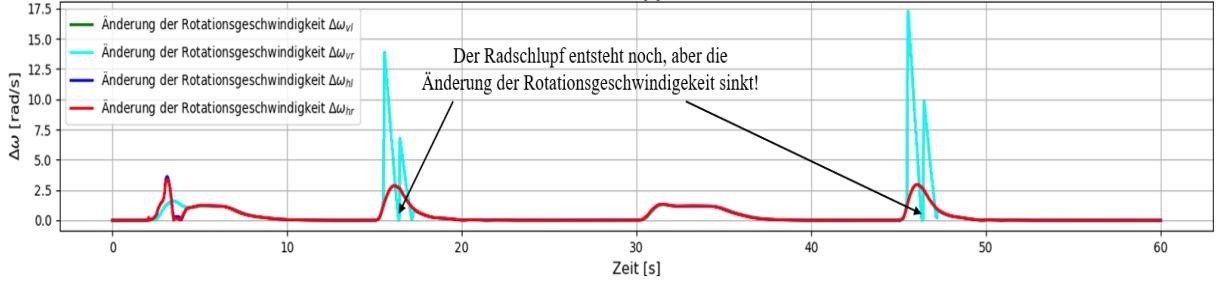


Abbildung 3.9: Änderungen der Rotationsgeschwindigkeit beim Radschlupf

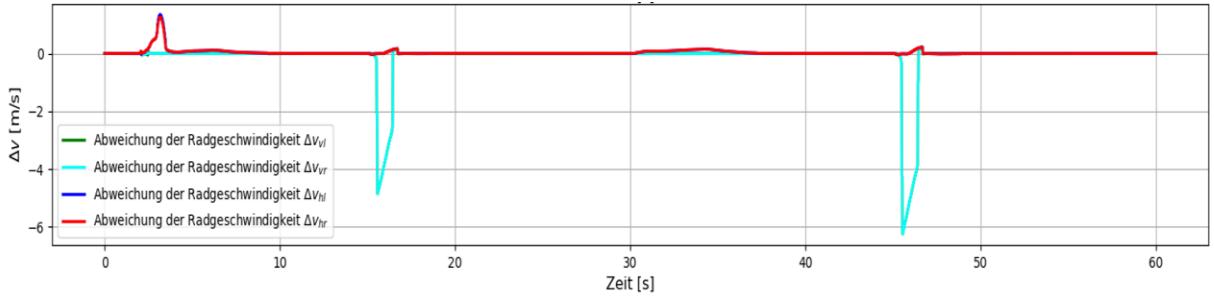


Abbildung 3.10: Abweichungen zwischen tatsächlichen und prädizierten Radgeschwindigkeit

Eine schnelle Erkennung des Radschlups ist daher sehr wichtig, um die ungültige Radgeschwindigkeit auszuschließen. In dieser Arbeit werden zwei Kriterien nach [18] zur Detektion des Radschlups verwendet.

Kriterium I:

An einem Rad ergibt sich zum gegenwärtigen Zeitpunkt t_k ein Radschlupf, sobald die gegenwärtige Rotationsgeschwindigkeit ω_k dieses Rades eine zu große Änderung relativ zum Mittelwert der vergangenen Rotationsgeschwindigkeiten $(\omega_{k-n}, \dots, \omega_{k-1})$ hat. Mathematisch kann das Kriterium wie folgt definiert:

$$\left| \omega_k - \frac{1}{n_{ZF}} \sum_{i=k-n_{ZF}}^{i=k-1} \omega_i \right| > \Delta\omega_{threshold} \quad (3.18)$$

wobei n_{ZF} die Länge des Zeitfensters bezeichnet. $\Delta\omega_{threshold}$ ist der unteren Schwellenwert der Änderung der Rotationsgeschwindigkeit für den Radschlupf. Als ein Einstellparameter soll $\Delta\omega_{threshold}$ nicht zu groß sein, damit der Radschlupf früh wie möglich detektiert werden kann. Jedoch darf dieser Schwellenwert auch nicht zu klein sein, ansonsten wird das normale Messrauschen beim Rotationsgeschwindigkeit auch als Radschlupf betrachtet.

Nachdem der Radschlupf zum Zeitpunkt t_k auftritt, ist die Änderung der Rotationsgeschwindigkeit meistens nicht mehr stark. Im diesem Fall ergibt sich nach dem Kriterium **I** kein Radschlupf, obwohl der Radschlupf nach dem Zeitpunkt t_k tatsächlich noch entsteht (siehe Abbildung 3.9). Aus diesem Grund ist Kriterium **I** zur Detektion des Radschlupfs nicht ausreichend, man muss noch Kriterium **II** verwenden.

Kriterium **II**:

An einem Rad ergibt sich zum gegenwärtigen Zeitpunkt t_k ein Radschlupf, wenn die tatsächliche Radgeschwindigkeit $\omega_k \cdot R$ stark vom prädizierten Radgeschwindigkeit $v_{R,pred}$ abweicht:

$$|\omega_k \cdot R - v_{R,pred}| > \Delta v_{threshold} \quad (3.19)$$

wobei $\Delta v_{threshold}$ den unteren Schwellenwert für die Abweichung zwischen tatsächlichen und prädizierten Radgeschwindigkeit bezeichnet. Die prädizierte Radgeschwindigkeit $v_{R,pred}$ des jeweiligen Rades wird ermittelt, indem die mit dem Kalman Filter prädizierte longitudinale Geschwindigkeit des Fahrzeugs $v_{x,pred}$ über die Formel (3.13) ins reifenfeste Koordinatensystem des jeweiligen Rades transformiert wird.

Vergleicht man Abbildung 3.9 und Abbildung 3.10, kann man sehen, dass der Radschlupf des Rades durch die große Abweichung zwischen tatsächlichen und prädizierten Radgeschwindigkeit noch detektiert werden kann, auch wenn die Änderungen der Rotationsgeschwindigkeit schon stark sinkt. Kriterium **II** ist daher eine gute Ergänzung vom Kriterium **I**.

3.2.3 Auswahl der besten Radgeschwindigkeit

Mit den Radgeschwindigkeiten vierer Räder kann man die entsprechende longitudinalen Geschwindigkeiten des Fahrzeugs, nämlich $v_{x,vl}, v_{x,vr}, v_{x,hl}, v_{x,hr}$, ermitteln. Nach der Detektion des Radschlupfs werden nur die Räder, die keinen Radschlupf hat, als Kandidaten für die Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs berücksichtigt. Das heißt, mehrere ungültige Werte von $v_{x,vl}, v_{x,vr}, v_{x,hl}, v_{x,hr}$ werden direkt ausgeschlossen. Unter den restlichen Kandidaten wird eine beste longitudinale Geschwindigkeit ausgewählt.

Wie in Abschnitt 3.2.2 schon erwähnt, ist die translatorische Geschwindigkeit des Radmittelpunktes beim Bremsen des Fahrzeugs immer größer als die Radgeschwindigkeit, und beim Beschleunigen immer kleiner als die Radgeschwindigkeit. Daher sind die Kriterien für die Auswahl der besten longitudinalen Geschwindigkeiten je nach der longitudinalen Beschleunigung des Fahrzeugs a_x wie folgt entworfen (vgl. Tabelle 3.1).

Situation	Bremsen $(a_x < -0.1 \frac{m}{s^2})$	Kleine Änderung der longitudinalen Geschwindigkeit $(-0.1 \frac{m}{s^2} < a_x < 0.1 \frac{m}{s^2})$	Beschleunigen $(a_x > 0.1 \frac{m}{s^2})$
Kriterium	Maximum aller gültigen longitudinalen Geschwindigkeiten	Mittelwert aller gültigen longitudinalen Geschwindigkeiten	Minimum aller gültigen longitudinalen Geschwindigkeiten

Tabelle 3.1: Kriterien für die Auswahl der besten longitudinalen Geschwindigkeiten nach [18]

Die ausgewählte beste longitudinale Geschwindigkeit wird durch $v_{x,best}$ bezeichnet und als eine beste Beobachtung für die tatsächliche longitudinale Geschwindigkeit des Fahrzeugs ins Kalman Filter eingespeist.

3.2.4 Fusionierung mittels des Kalman Filters

Man kann nun die Beobachtung $v_{x,best}$ aus dem Radencoder und die Messdaten der IMU mittels des Kalman Filters fusionieren. Wie zuvor erwähnt, kann die longitudinale Geschwindigkeit des Fahrzeugs über die zeitliche Integration der gemessenen longitudinalen Beschleunigung a_x bestimmt werden. Aber die gemessene longitudinale Beschleunigung a_x kann sowohl die longitudinale Beschleunigung des Fahrzeugs als auch einen Teil der Schwerbeschleunigung $g = 9,81 \frac{m}{s^2}$ enthalten, wenn das Fahrzeug an einer um den Winkel θ gegenüber der Horizontalen geneigten Ebene fährt (siehe Abbildung 3.11).

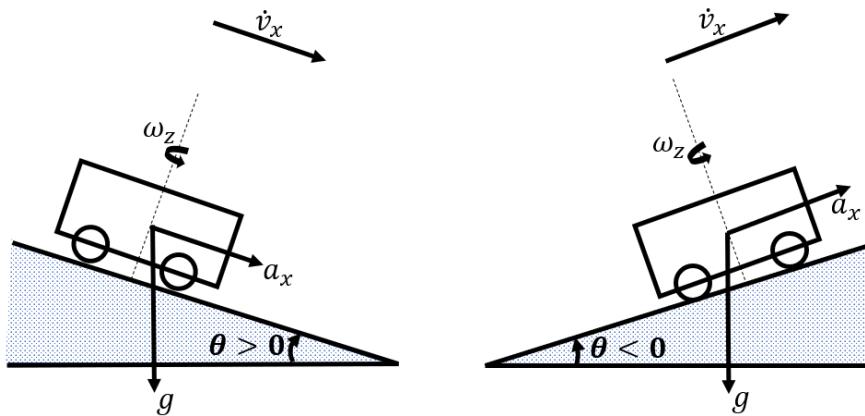


Abbildung 3.11: Fahrt auf der geneigten Ebene für $\theta > 0$ und $\theta < 0$

Um die longitudinale Beschleunigung des Fahrzeugs $a_{x,F}$ zu ermitteln, muss man $g \cdot \sin \theta$ vom a_x abziehen:

$$a_{x,F} = a_x - g \cdot \sin \theta \quad (3.20)$$

Die zeitliche Ableitung der longitudinalen Geschwindigkeit des Fahrzeugs lässt sich über die Formel (2.13) berechnen, wenn sich das Fahrzeug um die Giergeschwindigkeit ω_z dreht:

$$\dot{v}_x = a_{x,F} + v_y \cdot \omega_z = a_x - g \cdot \sin \theta + v_y \cdot \omega_z \quad (3.21)$$

Um die Wirkung der Schwerbeschleunigung auf die longitudinale Beschleunigung des Fahrzeugs berücksichtigen zu können, wird neben der longitudinalen Geschwindigkeit des Fahrzeugs v_x , auch $\sin \theta$ als einen Zustand beim Kalman Filter geschätzt. Der geschätzte Zustandsvektor ist daher $\mathbf{x} = [v_x \ sin \theta]^T$.

Das zeitdiskrete Prozessmodell zum Zeitpunkt t_k wird wie folgt aufgebaut:

$$\begin{aligned} \mathbf{x}_k &= \begin{bmatrix} v_{x,k} \\ \sin \theta_k \end{bmatrix} = \mathbf{A}_k \cdot \mathbf{x}_{k-1} + \mathbf{B}_k \cdot u_k + \mathbf{v}_k \\ &= \begin{bmatrix} v_{x,k-1} + \dot{v}_{x,k} \cdot \Delta t \\ \sin \theta_{k-1} \end{bmatrix} + \mathbf{v}_k = \begin{bmatrix} 1 & -g \cdot \Delta t \\ 0 & 1 \end{bmatrix} \cdot \underline{\mathbf{x}}_{k-1} + \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \cdot u_k + \mathbf{v}_k \end{aligned} \quad (3.22)$$

wobei Δt das Zeitintervall zwischen dem letzten Zeitpunkt t_{k-1} und dem gegenwärtigen Zeitpunkt t_k bezeichnet. $u_k = a_{x,k} + v_{y,k} \cdot \omega_{z,k}$ ist die deterministische Eingangsgröße. Zur Vereinfachung der Analyse wird ein additives Prozessrauschen \mathbf{v}_k ins Prozessmodell hinzugefügt.

Das Beobachtungsmodell lautet wie folgt:

$$z_k = v_{x,best,k} = \mathbf{H}_k \cdot \mathbf{x}_k + \mu_k = [1 \ 0] \cdot \mathbf{x}_k + \mu_k \quad (3.23)$$

Hier ist die Beobachtung $z_k = v_{x,best,k}$ ein Skalar, und das Beobachtungsrauschen ist auch ein Skalar.

Um die Unsicherheit des Prozessmodells sowie der Beobachtung zu beschreiben, werden beim KF die Kovarianzmatrizen \mathbf{Q}_k und R_k eingeführt, wobei \mathbf{Q}_k eine 2×2 Matrix und R_k ein Skalar ist.

3.3 Schätzung des Schwimmwinkels des Fahrzeugs

Der Schwimmwinkel des Fahrzeugs wird mit einem Schätzer online geschätzt. Eine Analyse und ein Vergleich der verschiedenen Methoden der Schwimmwinkelschätzung findet sich in [20], wobei diese Methoden hauptsächlich in drei Untergruppen eingeteilt werden:

1. Kinematische Methoden mit GPS-IMU Fusion [24]

Das GPS-System kann die Größe der Schwerpunktgeschwindigkeit des Fahrzeugs v_G sowie deren Richtung ϑ im erdfesten Koordinatensystem direkt messen, während der Gierwinkel des Fahrzeugs ψ über zeitliche Integration der von IMU gelieferten Giergeschwindigkeit ermittelt werden kann. Dadurch wird der Schwimmwinkel des Fahrzeugs als die Differenz zwischen ϑ und ψ berechnet. Um das Messrauschen der Signale von IMU und GPS zu beseitigen, wird ein kinematisches Modell für die Fusion der Signale von IMU und GPS eingesetzt. Dafür wird häufig das EKF verwendet.

2. Fahrdynamik basierte Methoden [19][25]

Diese Methoden verwenden die d'Alembert basierten fahrdynamischen Gleichungen als Prozessmodell, um den Schwimmwinkel aus den Messdaten der IMU und Lenkwinkelsensor sowie der longitudinalen Geschwindigkeit des Fahrzeugs mittels des EKF/UKF zu rekonstruieren.

3. Neurale Netzwerke basierte Methoden [26]

Bei diesen datengetriebenen Methoden werden keine konkrete physikalische Modelle verwendet. Die Neuronale Netze können aus den Trainingsdaten den Zusammenhang zwischen dem Schwimmwinkel und anderen fahrdynamischen Parametern, wie zum Beispiel, der longitudinalen Geschwindigkeit, der longitudinalen Beschleunigung, der lateralen Beschleunigung, der Giergeschwindigkeit sowie des Lenkwinkels, lernen.

Die erste Methode liefert eine hohe Schätzgenauigkeit, wenn sich das Fahrzeug mit einer hohen Geschwindigkeit oder einer großen Querbeschleunigung bewegt. Jedoch benötigt diese Methode einen kontinuierlichen Zugriff auf die GPS-Signale. Die zweite Methode ist geeignet für die Fahrt mit einer niedrigen Geschwindigkeit und einer kleinen Querbeschleunigung, während bei zu hoher Geschwindigkeit oder zu großer Querbeschleunigung die Schätzgenauigkeit dieser Methode schlechter wird. Die dritte Methode braucht keine physikalischen Modelle, jedoch benötigt sie große Mengen an Trainingsdaten.

In dieser Arbeit wird die zweite Schätzmethode nach [19][25], welche auf Fahrdynamik des Fahrzeugs basiert, verwendet, weil diese Methode keine GPS-Signale und keine Trainingsdaten benötigt. Den Nachteil der zweiten Methode kann man auch beim Bus-Platooning vernachlässigen, weil der Linienbus im urbanen Bereich üblicherweise mit einer niedrigen Geschwindigkeit und einer kleinen Querbeschleunigung ($< 4 \frac{m}{s^2}$) fährt. Diese Methode verwendet ein Unscented Kalman Filter als den Zustandsschätzer, bei welchem die d'Alembert basierten fahrdynamischen Gleichungen als Prozessmodell, die gemessene longitudinale Beschleunigung $a_{x,imu}$ und die Lenkwinkel der Vorderräder δ_{vl} und δ_{vr} als Eingangsgrößen, und die gemessene Querbeschleunigung $a_{y,imu}$ und Giergeschwindigkeit $\omega_{z,imu}$ sowie die vom Geschwindigkeit-Schätzer geschätzte longitudinale Geschwindigkeit $v_{x,est}$ des Fahrzeugs als Beobachtungen verwendet werden (siehe folgende Abbildung).

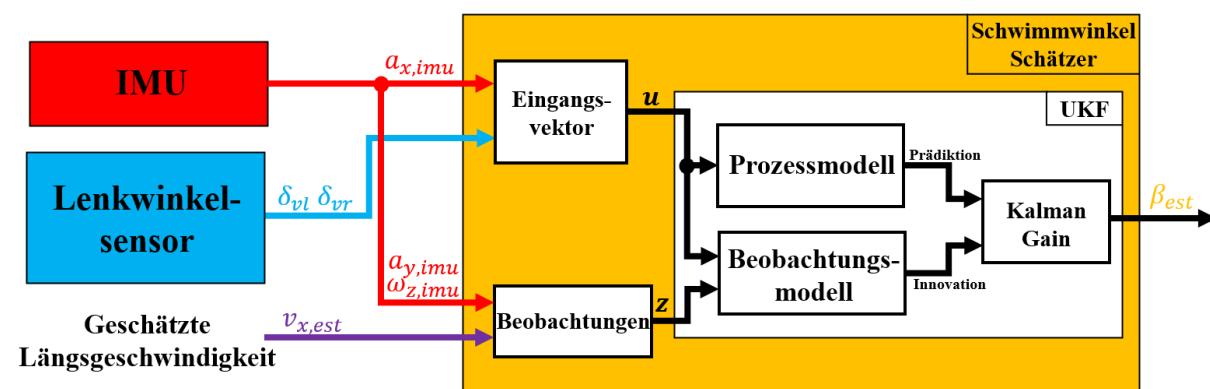


Abbildung 3.12: Blockschaltbild des Schwimmwinkelschätzers

In Abschnitt 2.3.4 werden die d'Alembert basierten Fahrdynamischen Gleichungen (2.24) und (2.25) bereits aufgestellt. Sie werden hier umgestellt, so dass die zeitliche Ableitung der Quergeschwindigkeit und der Giergeschwindigkeit jeweils auf die linke Seite der Formel gebracht wird:

$$\dot{v}_y = \frac{1}{m} \cdot \left(2 \cdot c_v \cdot \left(\delta_v - \frac{\omega_z \cdot l_v}{v_x} - \frac{v_y}{v_x} \right) \cdot \cos \delta_v + 2 \cdot c_h \cdot \left(\frac{\omega_z \cdot l_h}{v_x} - \frac{v_y}{v_x} \right) \right) - \omega_z \cdot v_x \quad (3.24)$$

$$\dot{\omega}_z = \frac{1}{I_z} \cdot \left(2 \cdot c_v \cdot \left(\delta_v - \frac{\omega_z \cdot l_v}{v_x} - \frac{v_y}{v_x} \right) \cdot \cos \delta_v \cdot l_v - 2 \cdot c_h \cdot \left(\frac{\omega_z \cdot l_h}{v_x} - \frac{v_y}{v_x} \right) \cdot l_h \right) \quad (3.25)$$

Aus (2.13) wird auch die zeitliche Ableitung der Längsgeschwindigkeit ermittelt:

$$\dot{v}_x = a_x + \omega_z \cdot v_y \quad (3.26)$$

Mit diesen zeitlichen Ableitungen $\dot{v}_y, \dot{\omega}_z, \dot{v}_x$ lässt sich das zeitdiskrete Gleichungssystem für das Prozessmodell zum Zeitpunkt t_k herleiten:

$$\mathbf{x}_k = \begin{bmatrix} v_{x,k} \\ v_{y,k} \\ \omega_{z,k} \end{bmatrix} = \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{v}_k) = \begin{bmatrix} v_{x,k-1} + \dot{v}_{x,k} \cdot \Delta t \\ v_{y,k-1} + \dot{v}_{y,k} \cdot \Delta t \\ \omega_{z,k-1} + \dot{\omega}_{z,k} \cdot \Delta t \end{bmatrix} + \begin{bmatrix} v_{a_{x,k}} \\ v_{a_{y,k}} \\ v_{\dot{\omega}_{z,k}} \end{bmatrix} \cdot \Delta t \quad (3.27)$$

wobei Δt das Zeitintervall zwischen dem letzten Zeitpunkt t_{k-1} und dem gegenwärtigen Zeitpunkt t_k bezeichnet. Der geschätzte Zustandsvektor \mathbf{x}_k besteht aus $v_{x,k}, v_{y,k}$ und $\omega_{z,k}$. Der deterministische Eingangsvektor gilt:

$$\mathbf{u}_k = \begin{bmatrix} a_{x,k} \\ \delta_{v,k} \end{bmatrix} \quad (3.28)$$

wobei $a_{x,k}$ mit der IMU gemessen wird, und $\delta_{v,k}$ ist das Lenkwinkel der Vorderachse, nämlich der Mittelwert der Lenkwinkel der beiden Vorderräder:

$$\delta_{v,k} = \frac{1}{2} \cdot (\delta_{vl,k} + \delta_{vr,k}) \quad (3.29)$$

Als Prozessrauschen \mathbf{v}_k werden hier die Änderung der longitudinalen und lateralen Beschleunigung $v_{a_{x,k}}$ und $v_{a_{y,k}}$, sowie die mögliche unbekannte Gierbeschleunigung $v_{\dot{\omega}_{z,k}}$ berücksichtigt. Die Unsicherheit des Prozessmodells wird vereinfacht durch das Prozessrauschen \mathbf{v}_k mal das Zeitintervall Δt .

Das Beobachtungsmodell zum Zeitpunkt t_k mit dem Beobachtungsvektor \mathbf{z}_k ergibt sich zu:

$$\begin{aligned}
 \mathbf{z}_k &= \begin{bmatrix} a_{y,imu,k} \\ v_{x,est,k} \\ \omega_{z,imu,k} \end{bmatrix} = \mathbf{h}_k(\mathbf{x}_k, \mathbf{u}_k) + \boldsymbol{\mu}_k = \begin{bmatrix} \frac{1}{m} \cdot (F_{yv,k} \cdot \cos \delta_{v,k} + F_{yh,k}) \\ v_{x,k} \\ \omega_{z,k} \end{bmatrix} + \begin{bmatrix} \mu_{a_y,k} \\ \mu_{v_x,k} \\ \mu_{\omega_z,k} \end{bmatrix} \\
 &= \begin{bmatrix} \frac{1}{m} \cdot \left(2 \cdot c_v \cdot \left(\delta_{v,k} - \frac{\omega_{z,k} \cdot l_v}{v_{x,k}} - \frac{v_{y,k}}{v_{x,k}} \right) \cdot \cos \delta_{v,k} + 2 \cdot c_h \cdot \left(\frac{\omega_{z,k} \cdot l_h}{v_{x,k}} - \frac{v_{y,k}}{v_{x,k}} \right) \right) \\ v_{x,k} \\ \omega_{z,k} \end{bmatrix} \\
 &\quad + \begin{bmatrix} \mu_{a_y,k} \\ \mu_{v_x,k} \\ \mu_{\omega_z,k} \end{bmatrix} \tag{3.30}
 \end{aligned}$$

wobei $a_{y,imu,k}$ und $\omega_{z,imu,k}$ zum Zeitpunkt t_k die mit der IMU gemessenen Querbeschleunigung und Giergeschwindigkeit bezeichnen, und $v_{x,est,k}$ ist die mit dem Geschwindigkeit-Schätzer geschätzte longitudinale Geschwindigkeit des Fahrzeugs. Sie werden als Beobachtungen betrachtet. Der Eingangsvektor \mathbf{u}_k für das Beobachtungsmodell besteht lediglich aus dem Lenkwinkel $\delta_{v,k}$ der Vorderachse.

Man kann sehen, dass die Beobachtungen $v_{x,est,k}$ und $\omega_{z,imu,k}$ eine direkte Abbildung für die Zustände $v_{x,k}$ und $\omega_{z,k}$ sind, während die Messabbildung $a_{y,imu,k} = f(\mathbf{x}_k, \mathbf{u}_k)$ nichtlinear ist, welche den Zusammenhang zwischen der Querbeschleunigung a_y und der auf das Fahrzeug wirkenden Seitenkraft F_y nach (2.15) und (2.17) verwendet.

Da jede Beobachtung eine unbekannte Messunsicherheit enthält, werden diese Messunsicherheiten als additives Beobachtungsrauchen $\boldsymbol{\mu}_k$ direkt auf das Beobachtungsmodell hinzugefügt:

$$\boldsymbol{\mu}_k = \begin{bmatrix} \mu_{a_y,k} \\ \mu_{v_x,k} \\ \mu_{\omega_z,k} \end{bmatrix} \tag{3.31}$$

Basierend auf dem Prozessmodell und Beobachtungsmodell lässt sich der Zustandsvektor \mathbf{x} mit Hilfe der Technologie der Kalman Filterung rekursiv schätzen. Da sowohl das Prozessmodell als auch das Beobachtungsmodell eine starke Nichtlinearität hat, und die Herleitung der entsprechenden Jacobi Matrizen sehr kompliziert ist, wird hier statt des EKF ein Unscented Kalman Filter verwendet. Die Frequenz der Schätzwerte für den Schwimmwinkel ist identisch mit der Messfrequenz von IMU und Lenkwinkelsensor.

Die Schätzgenauigkeit dieser Methode kann sich bei zu hoher Geschwindigkeit und somit zu großer Querbeschleunigung verschlechtern, weil das Prozessmodell und das Beobachtungsmodell in diesem Fall nicht mehr genau sind. Wie in Abschnitt 2.3.4 erwähnt, ist der Zusammenhang zwischen der Seitenkraft und dem Schräglaufwinkel bei zu großer Querbeschleunigung nicht mehr linear, somit ist eine konstante Schräglaufseitensteifigkeit zur Beschreibung dieses Zusammenhangs nicht genau genug. Zum Lösen dieses Problems wird in der Literatur [19] ein rekursiver Least-Squares-Schätzer mit dem Vergessensfaktor entworfen, um die variierte Schräglaufseitensteifigkeit online zu schätzen. In dieser Arbeit wird solcher

Schätzer nicht verwendet, weil der Linienbus bei der Fahrt im urbanen Bereich üblicherweise eine konstante Schräglaufseitensteifigkeit darstellt.

3.4 UKF basierte Fusionsalgorithmen für relative Positionierung

Die Vorbereitung für die zentralisierte Fusion aller Daten zweier Busse im Prozessor des verfolgenden Busses ist bisher fertig. Die dynamischen Kenngrößen des verfolgenden Busses $v_{x1,est}, \omega_{z1,imu}, a_{x1,imu}, \beta_{1,est}$ und des führenden Busses $v_{x2,est}, \omega_{z2,imu}, a_{x2,imu}, \beta_{2,est}$, sowie die vom Lidar- und Kamera-Tracker ermittelte relative Pose zweier Busse $[\Delta x_{lidar}, \Delta y_{lidar}, \Delta \varphi_{lidar}]^T$ und $[\Delta x_{kamera}, \Delta y_{kamera}, \Delta \varphi_{kamera}]^T$ können nun nach dem Messung und Schätzen im Prozessor vorliegen und weiter mit einem Kalman Filter fusioniert werden, um eine beste Schätzung der relative Pose zweier Busse zu erzielen. Ein Blockschaltbild für die zentralisierte Fusion findet sich in Abbildung 3.13.

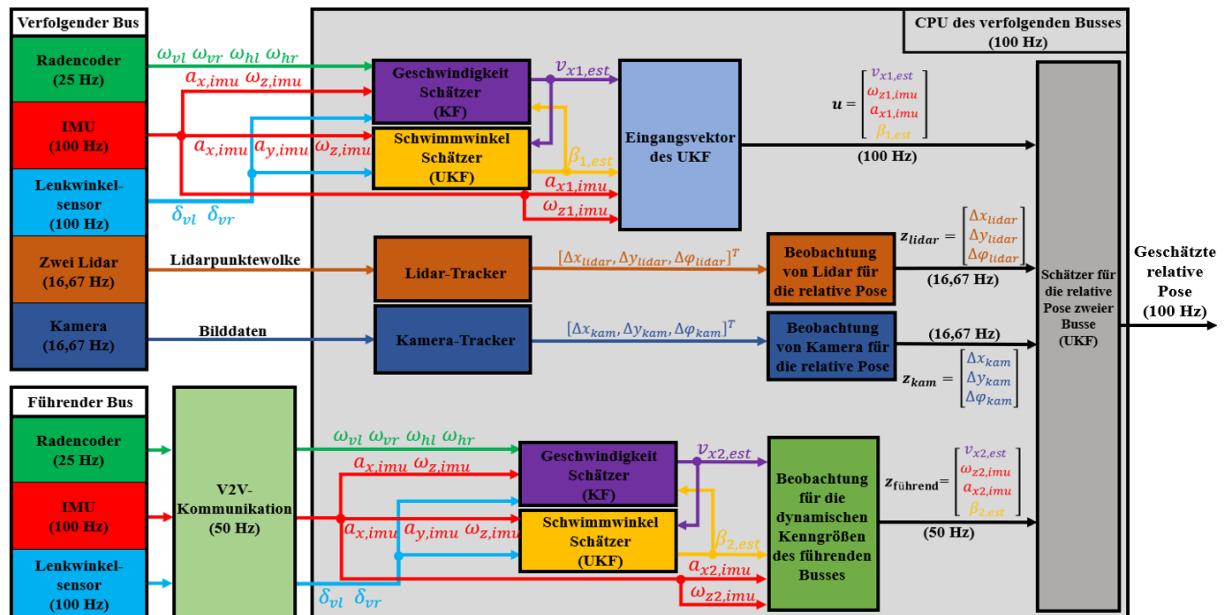


Abbildung 3.13: Blockschaltbild der zentralisierten Fusion zur Schätzung der relativen Pose

Der Prozessor (CPU) hat eine Verarbeitungsfrequenz von 100 Hz, daher soll der Schätzer für die relative Pose zweier Busse für jede 10 Millisekunde einen Schätzwert ausgeben. Jedoch haben die Sensoren sowie die V2V-Kommunikation unterschiedliche Frequenzen (vgl. Tabelle 3.2).

	CPU	Radencoder	IMU	Lenkwinkel-sensor	Lidar	Kamera	V2V-Kommunikation
Frequenz (Hz)	100	25	100	100	16,67	16,67	50

Tabelle 3.2: Frequenzen der Sensoren, CPU und V2V-Kommunikation

Wie zuvor in Abschnitt 3.2 und 3.3 schon erwähnt, ist die Frequenz der Schätzwerte für die longitudinalen Geschwindigkeit und den Schwimmwinkel identisch mit der Messfrequenz von IMU. Daher hat die dynamischen Kenngrößen des verfolgenden Busses, nämlich $v_{x1,est}, \omega_{z1,imu}, a_{x1,imu}, \beta_{1,est}$, eine Frequenz von 100 Hz, das heißt, für jeden Berechnungszyklus (10 ms) des Prozessors kann solche dynamischen Kenngrößen des verfolgenden Busses immer vorliegen. Deswegen können sie als Eingangsvektor \mathbf{u} in jedem Berechnungszyklus ins Prozessmodell eingesetzt werden, um den Prädiktionsschritt durchzuführen:

$$\mathbf{u} = [v_{x1,est} \quad \omega_{z1,imu} \quad a_{x1,imu} \quad \beta_{1,est}]^T \quad (3.32)$$

Die Tiefstellungen der Variablen im Eingangsvektor \mathbf{u} können wie folgt vereinfacht werden:

$$\mathbf{u} = [v_{x1} \quad \omega_{z1} \quad a_{x1} \quad \beta_1]^T \quad (3.33)$$

Aber der Prozessor kann häufig nur eine 50 Hz frequente IMU-Signale vom führenden Bus empfangen, weil die Frequenz der V2V-Kommunikation nur 50 Hz ist. Im schlechteren Fall, z.B. wenn die V2V-Kommunikation eine zu große Zeitverzögerung hat, ist die Frequenz der kommenden IMU-Signale vom führenden Bus sogar niedriger als 50 Hz. Aus diesem Grund können es nicht garantiert sein, dass für jeden Berechnungszyklus (10 ms) die dynamischen Kenngrößen des führenden Busses $v_{x2,est}, \omega_{z2,imu}, a_{x2,imu}, \beta_{2,est}$ immer beim Prozessor vorliegen können. Daher können sie nicht als Eingangsgrößen, sondern nur als Beobachtungen dienen:

$$\mathbf{z}_{\text{führend}} = [v_{x2,est} \quad \omega_{z2,imu} \quad a_{x2,imu} \quad \beta_{2,est}]^T \quad (3.34)$$

Für solche nicht kontinuierlich erfassbare Größen sollen neue Zustände in den geschätzten Zustandsvektor beim Kalman Filter hinzugefügt werden, damit sie in jedem Berechnungszyklus des Prozessors schätzbar sind [21]. Der geschätzte Zustandsvektor muss daher auf der Basis der geschätzten relativen Pose wie folgt erweitert werden:

$$\mathbf{x} = [\Delta x \quad \Delta y \quad \Delta \varphi \quad v_{x2} \quad \omega_{z2} \quad a_{x2} \quad \beta_2]^T \quad (3.35)$$

Das zeitdiskrete Prozessmodell für den Zustandsvektor \mathbf{x} zum Zeitpunkt t_k wird nach dem erstellten kinematischen Modell (3.1) ~ (3.9) in Abschnitt 3.1 wie folgt aufgebaut:

$$\begin{aligned}
 \boldsymbol{x}_k &= \begin{bmatrix} \Delta x_k \\ \Delta y_k \\ \Delta \varphi_k \\ v_{x2,k} \\ \omega_{z2,k} \\ a_{x2,k} \\ \beta_{2,k} \end{bmatrix} = \boldsymbol{f}_k(\boldsymbol{x}_{k-1}, \boldsymbol{u}_k, \boldsymbol{v}_k) \\
 &= \begin{bmatrix} \cos(\omega_{z1,k} \cdot \Delta t) \cdot \Delta x_k^\# + \sin(\omega_{z1,k} \cdot \Delta t) \cdot \Delta y_k^\# \\ -\sin(\omega_{z1,k} \cdot \Delta t) \cdot \Delta x_k^\# + \cos(\omega_{z1,k} \cdot \Delta t) \cdot \Delta y_k^\# \\ \Delta \varphi_{k-1} + (\omega_{z2,k} - \omega_{z1,k}) \cdot \Delta t \\ v_{x2,k-1} + (a_{x2,k} + \omega_{z2,k} \cdot v_{x2,k} \cdot \tan \beta_{2,k}) \cdot \Delta t \\ \omega_{z2,k-1} \\ a_{x2,k-1} \\ \beta_{2,k-1} \end{bmatrix} + \boldsymbol{g}_k(\boldsymbol{v}_k)
 \end{aligned} \tag{3.36}$$

wobei der Eingangsvektor \boldsymbol{u}_k nach (3.33) gilt:

$$\boldsymbol{u}_k = [v_{x1,k} \quad \omega_{z1,k} \quad a_{x1,k} \quad \beta_{1,k}]^T \tag{3.37}$$

$\Delta x_k^\#$ und $\Delta y_k^\#$ können über folgende Formeln ermittelt werden:

$$\Delta t = t_k - t_{k-1} \tag{3.38}$$

$$v_{1,k} = v_{x1,k} \cdot \sqrt{1 + \tan^2 \beta_{1,k}} \tag{3.39}$$

$$v_{2,k} = v_{x2,k} \cdot \sqrt{1 + \tan^2 \beta_{2,k}} \tag{3.40}$$

$$Logi_{1,k} = v_{1,k} \cdot \Delta t \cdot \cos(0,5 \cdot \omega_{z1,k} \cdot \Delta t + \beta_{1,k}) \tag{3.41}$$

$$Late_{1,k} = v_{1,k} \cdot \Delta t \cdot \sin(0,5 \cdot \omega_{z1,k} \cdot \Delta t + \beta_{1,k}) \tag{3.42}$$

$$Logi_{2,k} = v_{2,k} \cdot \Delta t \cdot \cos(\Delta \varphi_{k-1} + 0,5 \cdot \omega_{z2,k} \cdot \Delta t + \beta_{2,k}) \tag{3.43}$$

$$Late_{2,k} = v_{2,k} \cdot \Delta t \cdot \sin(\Delta \varphi_{k-1} + 0,5 \cdot \omega_{z2,k-1} \cdot \Delta t + \beta_{2,k}) \tag{3.44}$$

$$\Delta x_k^\# = \Delta x_{k-1} + Logi_{2,k} - Logi_{1,k} \tag{3.45}$$

$$\Delta y_k^\# = \Delta y_{k-1} + Late_{2,k} - Late_{1,k} \tag{3.46}$$

Die Unsicherheit des Prozessmodells wird durch den Term $\boldsymbol{g}_k(\boldsymbol{v}_k)$ bezeichnet, wobei das Prozessrauschen \boldsymbol{v}_k durch eine nichtlineare Funktion $\boldsymbol{g}_k(\dots)$ propagiert wird. Es ist nicht leicht, die Unsicherheit dieses Prozessmodells mit dem Ausdruck $\boldsymbol{g}_k(\boldsymbol{v}_k)$ zu beschreiben, weil das Prozessmodell selbst schon sehr kompliziert ist. Hier wird zur Vereinfachung der Analyse nur die Änderung der longitudinalen Beschleunigung $v_{a_x,k}$, die unbekannte Gierbeschleunigung $v_{\dot{\omega}_z,k}$ sowie die unbekannte Schwimmbeschleunigung $v_{\dot{\beta},k}$ als das Prozessrauschen berücksichtigt:

$$\boldsymbol{v}_k = \begin{bmatrix} v_{a_x,k} \\ v_{\dot{\omega}_z,k} \\ v_{\beta,k} \end{bmatrix} \quad (3.47)$$

Die Unsicherheit des Prozessmodells wird wie folgt vereinfacht:

$$\begin{aligned} \boldsymbol{g}_k(\boldsymbol{v}_k) &\approx \begin{bmatrix} \frac{\Delta t^2}{2} \cdot (v_{a_{x1},k} \cdot \cos(\Delta\varphi_{k-1} + 0,5 \cdot \omega_{z2,k} \cdot \Delta t + \beta_{2,k}) + v_{a_{x2},k} \cdot \cos(0,5 \cdot \omega_{z2,k} \cdot \Delta t + \beta_{2,k})) \\ \frac{\Delta t^2}{2} \cdot (v_{a_{x1},k} \cdot \sin(\Delta\varphi_{k-1} + 0,5 \cdot \omega_{z2,k} \cdot \Delta t + \beta_{2,k}) + v_{a_{x2},k} \cdot \sin(0,5 \cdot \omega_{z2,k} \cdot \Delta t + \beta_{2,k})) \\ \frac{\Delta t^2}{2} \cdot (v_{\dot{\omega}_{z1},k} + v_{\dot{\omega}_{z2},k}) \\ v_{a_x,k} \cdot \Delta t \\ v_{\dot{\omega}_z,k} \cdot \Delta t \\ v_{a_x,k} \\ v_{\beta,k} \cdot \Delta t \end{bmatrix} \\ &\approx \begin{bmatrix} 0,5 \cdot v_{a_x,k} \cdot \Delta t^2 \cdot (\cos \Delta\varphi_{k-1} + 1) \\ 0,5 \cdot v_{a_x,k} \cdot \Delta t^2 \cdot \sin \Delta\varphi_{k-1} \\ v_{\dot{\omega}_z,k} \cdot \Delta t^2 \\ v_{a_x,k} \cdot \Delta t \\ v_{\dot{\omega}_z,k} \cdot \Delta t \\ v_{a_x,k} \\ v_{\beta,k} \cdot \Delta t \end{bmatrix} \end{aligned} \quad (3.48)$$

Für jeden Berechnungszyklus kann das Kalman Filter den Prädiktionsschritt mittels des Prozessmodells (3.36) durchführen, während der Innovationschritt durchgeführt wird, nur wenn die neuen Beobachtungen reinkommt. Diese Beobachtungen sind in drei Untergruppen einzuteilen:

1. Mit dem Lidar-Tracker ermittelte relative Pose zweier Busse:

$$\mathbf{z}_{lidar,k} = [\Delta x_{lidar,k}, \Delta y_{lidar,k}, \Delta\varphi_{lidar,k}]^T \quad (3.49)$$

Das Beobachtungsmodell lautet:

$$\mathbf{z}_{lidar,k} = [\mathbf{I}_{3 \times 3} \quad \mathbf{O}_{3 \times 4}] \cdot \mathbf{x}_k + \boldsymbol{\mu}_{lidar,k} \quad (3.50)$$

wobei $\mathbf{I}_{3 \times 3}$ die Einheitsmatrix mit der Dimension 3×3 bezeichnet. $\mathbf{O}_{3 \times 4}$ bedeutet eine 3×4 Nullmatrix. $\boldsymbol{\mu}_{lidar} = [\mu_{lidar,\Delta x}, \mu_{lidar,\Delta y}, \mu_{lidar,\Delta\varphi}]^T$ ist das additiven Beobachtungsrauschen vom Lidar-Tracker.

2. Mit dem Kamera-Tracker ermittelte relative Pose:

$$\mathbf{z}_{kamera,k} = [\Delta x_{kamera,k}, \Delta y_{kamera,k}, \Delta\varphi_{kamera,k}]^T \quad (3.51)$$

Das Beobachtungsmodell ist gleich wie den Lidar-Tracker:

$$\mathbf{z}_{kamera,k} = [\mathbf{I}_{3 \times 3} \quad \mathbf{O}_{3 \times 4}] \cdot \mathbf{x}_k + \boldsymbol{\mu}_{kamera,k} \quad (3.52)$$

3. Beobachtungen für die dynamischen Kenngrößen des führenden Busses nach (3.34):

$$\mathbf{z}_{\text{führend},k} = [v_{x2,est,k} \quad \omega_{z2,imu,k} \quad a_{x2,imu,k} \quad \beta_{2,est,k}]^T \quad (3.53)$$

Das entsprechende Beobachtungsmodell ergibt sich zu:

$$\mathbf{z}_{\text{führend},k} = [\mathbf{O}_{4 \times 3} \quad \mathbf{I}_{4 \times 4}] \cdot \mathbf{x}_k + \boldsymbol{\mu}_{\text{führend},k} \quad (3.54)$$

wobei $\boldsymbol{\mu}_{\text{führend}} = [\mu_{v_{x2,est}}, \mu_{\omega_{z2,imu}}, \mu_{a_{x2,imu}}, \mu_{\beta_{2,est}}]^T$ auch das additives Beobachtungsrauschen bezeichnet.

Obwohl alle Beobachtungsmodelle linear sind, hat das Prozessmodell eine starke Nichtlinearität, und eine Berechnung der Jacobi Matrizen ist zu kompliziert. Deshalb wird hier das UKF verwendet, um eine Kalman Filterung für das nichtlineare System zu entwickeln.

Beim UKF wird der Zustandsvektor $\mathbf{x}_k = [\Delta x_k \quad \Delta y_k \quad \Delta \varphi_k \quad v_{x2,k} \quad \omega_{z2,k} \quad a_{x2,k} \quad \beta_{2,k}]^T$, wie in Abschnitt 2.4.3 schon erwähnt, um auf das nichtlineare Prozessmodell wirkende nicht additive Prozessrauschen $\mathbf{v}_k = [v_{a_x,k} \quad v_{\dot{\omega}_z,k} \quad v_{\dot{\beta},k}]^T$ erweitert (Anmerkung: „aug“ = erweitert, engl. *augmented*):

$$\mathbf{x}_k^{\text{aug}} = \begin{bmatrix} \Delta x_k \\ \Delta y_k \\ \Delta \varphi_k \\ v_{x2,k} \\ \omega_{z2,k} \\ a_{x2,k} \\ \beta_{2,k} \\ v_{a_x,k} \\ v_{\dot{\omega}_z,k} \\ v_{\dot{\beta},k} \end{bmatrix} \quad (3.55)$$

Damit können das nicht additive Prozessrauschen \mathbf{v}_k beim UKF auch immer geschätzt werden. Die Kovarianzmatrix \mathbf{P}_k (Dimension 7×7) des Zustandsvektors \mathbf{x}_k wird auch um die Kovarianzmatrix \mathbf{Q}_k (Dimension 3×3) des Prozessrauschens \mathbf{v}_k erweitert:

$$\mathbf{P}_k^{\text{aug}} = \begin{bmatrix} \mathbf{P}_k & \mathbf{0}_{7 \times 3} \\ \mathbf{0}_{3 \times 7} & \mathbf{Q}_k \end{bmatrix} \quad (3.56)$$

Die Kovarianzmatrix des Beobachtungsrauschens $\boldsymbol{\mu}_{\text{lidar},k}$ wird durch $\mathbf{R}_{\text{lidar},k}$ bezeichnet. Für die Beobachtungsrauschen $\boldsymbol{\mu}_{\text{kamera},k}$ und $\boldsymbol{\mu}_{\text{führend},k}$ werden die Kovarianzmatrizen $\mathbf{R}_{\text{kamera},k}$ und $\mathbf{R}_{\text{führend},k}$ berücksichtigt.

3.5 Zeitsynchronisierung der Sensoren

Wenn die Messdaten der mehreren Quellen fusioniert werden müssen, stellt sich die Frage nach der Zeitsynchronisierung der erfassten Messdaten. Die Forderung hinsichtlich der Qualität der Zeitsynchronisierung sind abhängig von konkreten Aufgabenstellungen. Es gibt verschiedene Methoden zur Zeitsynchronisierung der Sensoren. Man unterscheidet dabei grundsätzlich hardwarebasiertes und softwarebasiertes Zeitsynchronisieren [27].

Beim hardwarebasierten Synchronisieren werden alle Sensoren durch einen externen Impuls (Hardwaretrigger) aktiviert, indem jeder der Sensoren das Triggersignal empfangen, um ein Messen zu starten [28]. Beispielsweise kann ein spezielles Interface zwischen Hardwaretrigger und GPS-Empfänger erstellt werden, damit der Hardwaretrigger nach der globalen Weltzeit die Sensoren regelmäßig auslöst [27]. Diese Methode bietet eine hohe Qualität der Zeitsynchronisierung, weil die Messdaten zwingend zu einem deterministischen Zeitpunkt generiert werden können. Jedoch ist der Entwurf eines solchen Hardwaretriggers für Sensorsysteme häufig schwer.

Beim softwarebasierten Zeitsynchronisieren wird jedes Messsignal ein Zeitstempel (z.B Unixzeit) zugeordnet, der gibt an, zu welchem Zeitpunkt das Messsignal vom Prozessor empfangen [27]. Beispielsweise können die Messsignale von der Software *Robotic Operation System (ROS)* gleich zeitgestempelt werden, sobald sie bei *ROS* vorliegen. Diese Methode braucht keinen Hardwaretrigger, jedoch entsteht zwischen diesem Zeitstempel und dem Zeitpunkt, an dem das Messen der Sensoren durchgeführt wird, eine unbekannte Verzögerungszeit. Hinsichtlich der Messdauer, der Übertragungsdauer und der verwendeten Hard- und Software können diese Verzögerungszeiten sehr unterschiedliche Größenordnungen annehmen [28], welche eine Synchronisationsunsicherheit ins Fusionssystem bringt. In dieser Arbeit wird das softwarebasierte Zeitsynchronisieren verwendet, weil ein Hardwaretrigger zurzeit bei Sensorsystemen vom Bus-Platooning nicht vorhanden ist.

Die Messdaten der Sensoren haben verschiedene Frequenzen (vgl. Tabelle 3.2). Daher entsteht zwischen unterschiedlichen Messdaten ein zeitliches Offset. Die Frage stellt sich danach, wie man alle Messdaten in einheitliche Zeitstempel transformieren kann. In dieser Arbeit werden die Zeitstempel der IMU als die zeitliche Referenz für alle anderen Messdaten betrachtet, weil die IMU die höchste Abtastrate hat. Wie in Abschnitt 3.2 und 3.3 schon erwähnt, haben die geschätzten longitudinalen Geschwindigkeit und der geschätzte Schwimmwinkel die gleiche Frequenz wie die IMU, weil die Schätzungen auf der Basis des Prädiktionsschritts immer durchgeführt wird, solange die hochfrequenten IMU Signale beim Kalman Filter vorliegen. Das bedeutet, dass die Zeitstempel der ermittelten fahrdynamischen Kenngrößen $[v_x \ \omega_z \ a_x \ \beta]^T$ des Busses mit den Zeitstempel der jeweiligen IMU des Busses identisch sind. Im Vergleich zu hochfrequenten (100 Hz) fahrdynamischen Kenngrößen $[v_{x1} \ \omega_{z1} \ a_{x1} \ \beta_1]^T$ des verfolgenden Busses haben jedoch die fahrdynamischen Kenngrößen $[v_{x2} \ \omega_{z2} \ a_{x2} \ \beta_2]^T$ des führenden Busses aufgrund der niedrigfrequenten V2V-Kommunikation nur eine niedrige Frequenz (50 Hz).

Für jeden Berechnungszyklus (10ms) des Prozessors kann der Schätzer der relativen Pose die neuen Messsignale ablesen (siehe Abbildung 3.14). Da die fahrdynamischen Kenngrößen des verfolgenden Busses, die als Eingangsgrößen des UKF betrachtet werden, auch jede 10ms neu generieren können, benutzt der Schätzer der relativen Pose die Zeitstempel der fahrdynamischen Kenngrößen des verfolgenden Busses als seine eigene Zeitstempel (siehe $t_0 \sim t_{15}$ in Abbildung 3.14). Solange die neuen fahrdynamischen Kenngrößen des verfolgenden Busses detektiert werden, verschiebt sich die Zeitstempel des Schätzers auch nach vorne. zieht der Schätzer den letzten Zeitstempel t_{k-1} vom gegenwärtigen Zeitstempel t_k ab, wird das Zeitintervall Δt ermittelt und in Formel (3.36) eingesetzt, damit der Prädiktionsschritt beim UKF durchgeführt wird.

Die ermittelten relativen Posen von Lidar und Kamera, sowie die fahrdynamischen Kenngrößen des führenden Busses werden als Beobachtungen ins UKF eingespeist. Aber die Zeitstempel der Beobachtungen richten häufig nicht auf die Zeitstempel des Schätzers aus. Je nach der Forderung hinsichtlich der Qualität der Zeitsynchronisierung können solche niedrigen Beobachtungen durch unterschiedliche Methode, z.B. lineare Interpolation, in Bezug auf die Zeitstempel des Schätzers transformiert werden [27]. Zur Vereinfachung der Algorithmen werden in dieser Arbeit einfach angenommen, dass der Zeitstempel der neu reinkommenden Beobachtungen auf den gegenwärtigen Zeitstempel des Schätzers ausrichtet.

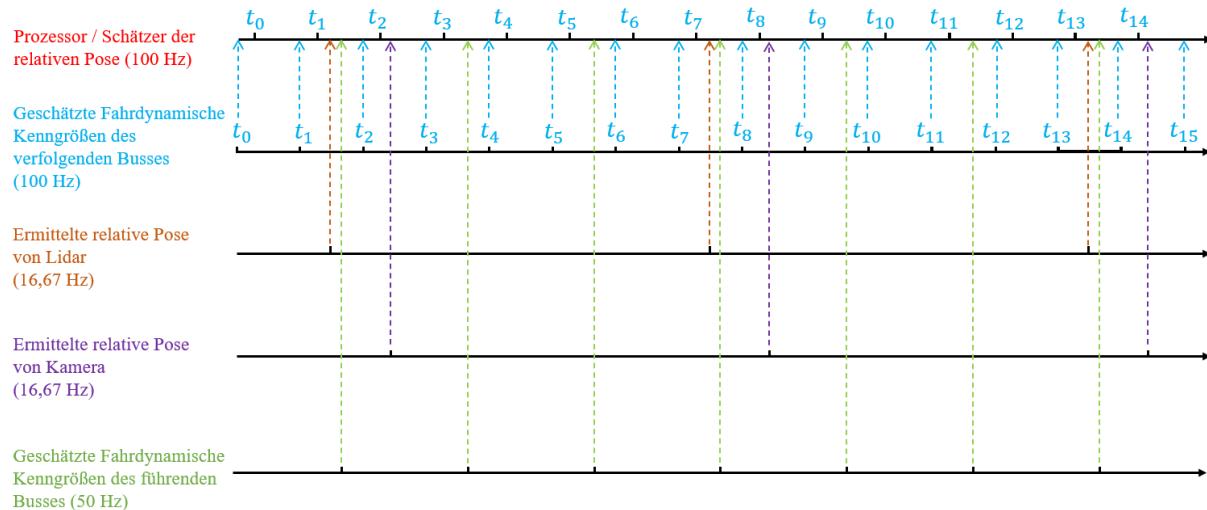


Abbildung 3.14: Darstellung der Zeitstempel des Fusionssystems

4 Implementierung in Simulation

Die entwickelten Algorithmen werden in Simulation implementiert und erprobt. In Simulation wird die Physik der Fahrzeuge, Umwelt sowie Sensorik durch Modelle vereinfacht abgebildet. Die entwickelten Algorithmen können die virtuellen Messwerte der virtuellen Sensoren verarbeiten und die ermittelten Ergebnisse ausgeben, danach lassen sich die ermittelten Ergebnisse sehr leicht mit den tatsächlichen Werten, die von Simulationssoftware geliefert werden, vergleichen. Mit der Simulation ist die Erprobung und Validierung der entwickelten Funktionalität zwar flexibel, schnell und kostengünstig, es ist jedoch nicht aussagekräftig genug, weil die Abbildung der Physik in der Simulationsumgebung nur eine Näherungslösung ist, und dazu kommt der Einfluss der Randbedingungen, die ebenfalls oft idealisiert sind. Um die Performanz der entwickelten Algorithmen besser nachzuweisen, ist der Versuch am Teststand auch notwendig, obwohl es zeit- und kostenaufwändig ist. In dieser Arbeit wird das entwickelte Konzept lediglich in der Simulation implementiert, weil der Versuchsaufbau am Teststand noch nicht fertig vorbereitet zur Verfügung steht.

Für die Simulation des urbanen Platooning wurde die Simulationssoftware *CarMaker 11.0* von *IPG Automotive* verwendet, welche große Bibliotheken von Fahrzeugen, Fahrzeugkomponenten sowie Testszenarien liefert. Für den Benutzer ist es sehr leicht, ein Fahrmanöver für das simulierte Fahrzeug vorzugeben, damit das Fahrzeug mit diesem Manöver im Testszenario fahren kann. Um dem Benutzer die Simulationsdaten in jedem Zeitschritt zur Verfügung zu stellen, bietet *CarMaker* eine C/C++ Schnittstelle, nämlich eine C/C++ Datei (*User.c/User.cpp*), bei der mehrere Funktionen in der Hauptschleife der Simulation zyklisch aufgerufen werden. Der Benutzer kann die selben Funktionen in diese Datei hinzufügen, damit diese Funktionen in der Hauptschleife die Messwerte der Sensoren zyklisch ablesen und verarbeiten können.

4.1 Busmodell

In *CarMaker 11.0* wird ein Modell des *Mercedes-Benz Citaro 2005* für einen typischen Linienbus gebietet, welcher in dieser Arbeit für das Bus-Platooning eingesetzt wird (siehe Abbildung 4.1). Die wichtige Parameter des simulierten Busses sind in Tabelle 4.1 dargestellt.



Abbildung 4.1: Busmodell des Mercedes-Benz Citaro 2005

Parameter	Wert
Gewicht	15597,98 kg
Trägheitsmoment (um die z-Achse durch den Schwerpunkt)	119975,015 kg·m ²
Länge	11,89 m
Breite	2,25 m
Höhe	2,65 m
Radstand (Abstand zwischen der Vorder- und Hinterachse)	5,80 m
Spurbreite (Abstand zwischen rechtem und linkem Rad einer Achse)	Vorderachse: 2,10 m Hinterachse: 2,10 m
Abstand des Schwerpunkts zur Achse	zur Vorderachse: 2,891 m zur Hinterachse: 2,909 m
Schräglaufseitensteifigkeiten (durch Messen bestimmt)	Vorderrad: $\approx 2 \cdot 10^5$ N/rad Hinterrad: $\approx 2 \cdot 10^5$ N/rad

Tabelle 4.1: Wichtige Parameter des Busmodells (Mercedes-Benz Citaro 2005)

4.2 Testszenario, Teststrecke und Fahrmanöver

Da der Linienbus im urbanen Bereich fährt, wird auch ein urbanes Testszenario in *CarMaker* ausgewählt. Abbildung 4.2 zeigt die verwendete Teststrecke im Testszenario des *UrbanRoad*. Nach der Einstellung des Fahrmanövers kann der führende Bus mit einer vorgegebenen variierten Geschwindigkeit im Bereich von 10 bis 30 km/h fahren, und der verfolgende Bus kann dem führenden Bus mit einem bestimmten longitudinalen Abstand ($\Delta x < 5m$) folgen, wodurch die Längs- und Querdynamik des verfolgenden Busses automatisch gesteuert werden. Abbildung 4.3 zeigt die Änderung des longitudinalen Abstands Δx und des lateralen Versatzes Δy zweier Busses sowie die Änderung der wahren longitudinalen Geschwindigkeit des verfolgenden und führenden Busses beim Platooning in Simulation.



Abbildung 4.2: Teststrecke im urbanen Testszenario

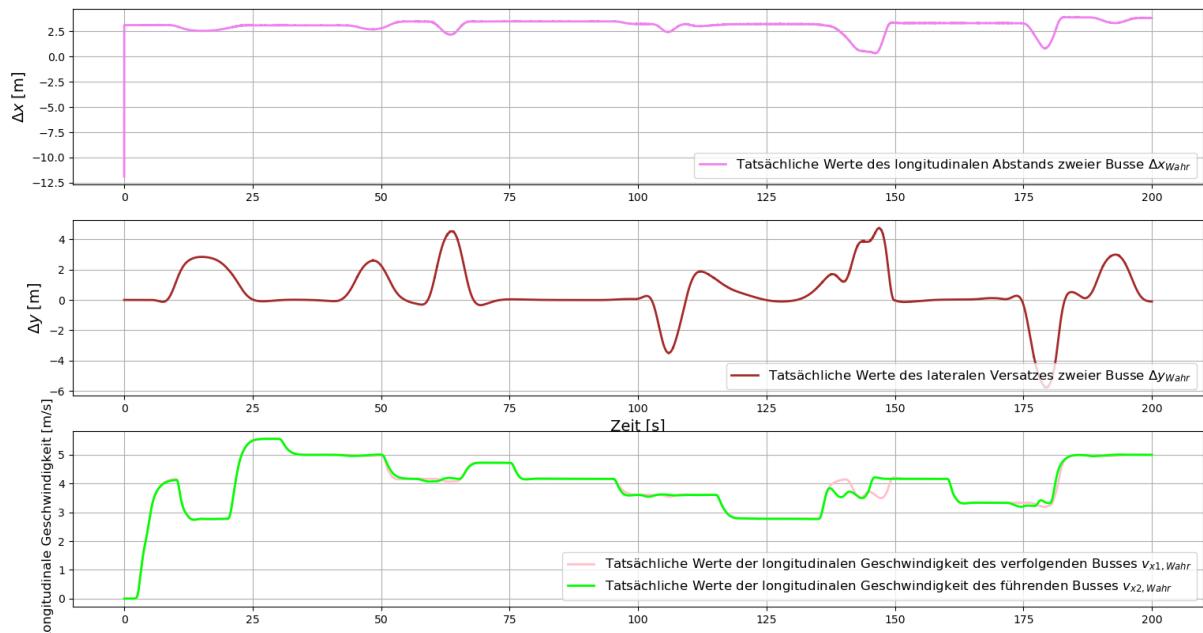


Abbildung 4.3: Änderung des longitudinalen Abstands, des lateralen Versatzes sowie der longitudinalen Geschwindigkeit des verfolgenden und führenden Busses

4.3 Sensorik

Die in dieser Arbeit verwendeten Sensoren für den realen Linienbus sind in Abschnitt 2.2 bereits vorgestellt. In CarMaker 11.0 finden sich keine virtuelle Sensoren für die Radencoder und Lenkwinkelsensoren, deshalb werden statt der Messwerte die tatsächlichen Werte mit additiv überlagerten Rauschen für die Radgeschwindigkeit vierer Räder und die Lenkwinkel zweier Vorderräder in entwickelten Algorithmen eingesetzt.

Die Funktionalität der IMU, Lidarsensoren und Kameratasensoren werden in *CarMaker* über virtuelle Sensoren nachgebildet. In dieser Arbeit ist der simulierte verfolgende Bus mit einer IMU, zwei Lidarsensoren und einem Kameratasensor ausgestattet, während auf dem simulierten führenden Bus nur eine IMU montiert ist. Über *CarMaker* GUI kann man die Einbauort der benötigten Sensoren auf die Fahrzeuge festlegen und die relevanten Sensorparameter, wie z.B. Abtastfrequenz, selbst einstellen.

Für eine Simulation der Lidarsensoren wird in *CarMaker* 11.0 ein Lidar Raw Signal Interface (Lidar RSI) gebietet, wobei die ausgesendeten Lidarstrahlen und deren Kollisionen mit Objekten in der Simulationsumgebung nachgebildet werden. Da die Simulation dieser Lidarstrahlen einen großen Berechnungsaufwand benötigt, wird für eine Parallelisierung der Sensorberechnungen ein GPUSensor-Framework eingesetzt. Für jeden reflektierten Lidarstrahlen werden dabei die relevanten Informationen, z.B. die Laufzeit (engl. Time of Flight) und die Lichtintensität, ermittelt [29]. Sie können anschließend über das Interface dem Benutzer zur Verfügung stehen.

Für das Generieren und Aufnahme der RGB-Bilddaten in *CarMaker* wird das Camera Raw Signal Interface (Camera RSI) verwendet (siehe Abbildung 4.4). Dabei werden die Bilddaten zunächst mittels des GPUSensor-Frameworks mit einer bestimmten Frequenz zyklisch generiert und anschließend als einen Datenstrom über TCP/IP in eine separate Client-Anwendung (*rsds-client-camera.c*) übertragen [29]. Alle Bilddaten werden dort in einem separaten Thread verarbeitet. Die Ergebnisse der Datenverarbeitung, z.B. in dieser Arbeit die relative Pose des führenden Busses, können über ein vom Benutzer definiertes Interface in Hauptschleife der Simulation transportiert werden, und für weitere Nutzung zur Verfügung stehen.

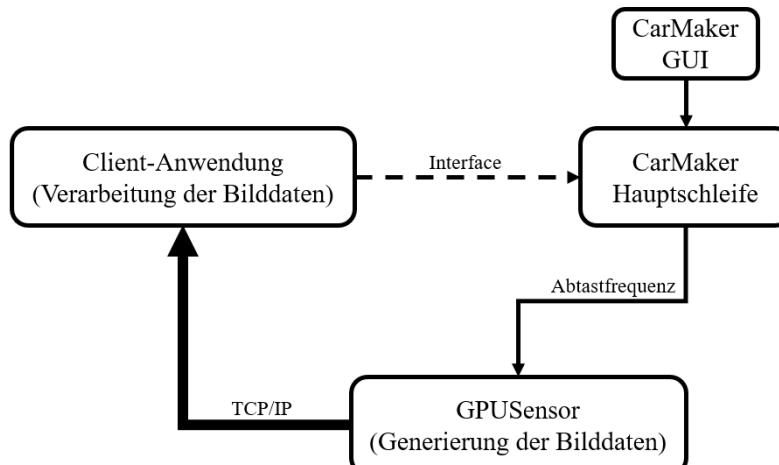


Abbildung 4.4: Blockschalbild des Camera RSI nach [29]

4.4 Datenaustausch zwischen Fahrzeugen mittels des SimNet

Eine normale Simulation in *CarMaker* konzentriert sich nur auf einem detaillierten Ego-Fahrzeug, und zusätzliche Verkehrsobjekte werden nur mit weniger detaillierten Fahrzeugmodellen, wie z.B. Fahrzeugsensoren oder komplexe Fahrzeugsteuermodule, simuliert. Um die Simulation mit mehreren detaillierten Ego-Fahrzeugen durchzuführen, und die Interaktion mehrerer Fahrzeuge zu realisieren, wird das SimNet eingesetzt.

Das SimNet ermöglicht dabei die Simulationen mehrerer Ego-Fahrzeuge, das jeweils in einer unabhängigen CarMaker-Anwendung läuft, jedoch können die Daten, wie z.B. Sensordaten und relevante Fahrzeugdaten, zwischen allen Anwendungen synchronisiert und ausgetauscht werden [30]. Ein Fahrzeug davon wird als eine Master-Anwendung simuliert, welche die ganze CarMaker-Simulation steuern kann, während die Simulationen aller anderen Fahrzeuge nur als Slave-Anwendungen dienen können, die von Master-Anwendung gesteuert werden.

In dieser Arbeit werden der verfolgende Bus in Master-Anwendung und der führende Bus in Slave-Anwendung simuliert. Die Sensordaten der IMU, sowie die Radgeschwindigkeiten und die Lenkwinkel zweier Busse werden über SimNet in die Hauptschleife der Simulation übertragen und in einen Datenpool für die weitere Nutzung gesammelt.

5 Erprobung und Analyse der Ergebnisse

In diesem Kapitel werden die entwickelten Algorithmen zur Schätzung der longitudinalen Geschwindigkeit, des Schwimmwinkels des Fahrzeugs sowie zur Schätzung der relativen Pose zweier Busse in Testumgebungen, die in Kapitel 4 vorgestellt werden, erprobt. Um die Schätzgüte zu bewerten werden dabei Häufigkeitsverteilung des Schätzfehlers und der *Root Mean Square Error* verwendet. Da jeder Schätzer ein normales Kalman Filter oder ein Unscented Kalman Filter verwendet, wird hier nach Literaturen erläutert, wie sie nach der gemessenen Kovarianz des Beobachtungsrauschens ausgelegt werden können, um eine beste Performance des Filters zu erzielen.

Die Ergebnisse für verschiedene Fälle in Simulationen werden dabei visualisiert, berechnet und diskutiert, z.B. wenn der Lidarsensor oder der Kamerasensor ausfällt, oder wenn die V2V-Kommunikation zwischen Bussen ausfällt.

5.1 Methode zur Beurteilung der Schätzgüte

Die Schätzfehler $[e_{\Delta x}, e_{\Delta y}, e_{\Delta \varphi}]$ der relativen Pose sind dabei die Abweichungen zwischen den Schätzwerten und den wahren Werten der relativen Pose:

$$\begin{aligned} e_{\Delta x} &= \Delta x_{\text{schätz}} - \Delta x_{\text{wahr}} \\ e_{\Delta y} &= \Delta y_{\text{schätz}} - \Delta y_{\text{wahr}} \\ e_{\Delta \varphi} &= \Delta \varphi_{\text{schätz}} - \Delta \varphi_{\text{wahr}} \end{aligned} \quad (5.1)$$

Um die Schätzgüte der relativen Pose zu bewerten, werden die Häufigkeitsverteilung sowie der *Root Mean Square Error* des Schätzfehlers der relativen Pose analysiert.

5.1.1 Häufigkeitsverteilung des Schätzfehlers

Die Verteilung des Schätzfehlers lässt sich häufig durch die Wahrscheinlichkeitsdichte des Schätzfehlers beschreiben. Aus der lassen sich Kenngrößen wie der Mittelwert und die Varianz des Schätzfehlers ermitteln. In der Praxis ist die Wahrscheinlichkeitsdichte des Schätzfehlers unbekannt, sie muss anhand einer repräsentativen Stichprobe des Schätzfehlers geschätzt werden. Das Ergebnis der Schätzung ist eine empirische Häufigkeitsverteilung, die grafisch in Form eines Histogramms dargestellt werden kann [09].

Abbildung 5.1 zeigt ein typisches Histogramm für die Häufigkeitsverteilung des Schätzfehlers. Dabei werden die Elemente e_i der Stichprobe des Schätzfehlers nach Größenklassen der Breite Δe sortiert werden:

$$v \cdot \Delta e \leq e_i < (v + 1) \cdot \Delta e \quad (5.2)$$

v bezeichnet hier den Klassenindex. Falls n_v Stichprobe von den insgesamt n Stichprobenelementen im Intervall (5.2) liegen, wird die Häufigkeit h_v des Schätzfehlers in der Klasse v über die folgende Formel bestimmt:

$$h_v = \frac{n_v}{n \cdot \Delta e} \quad (5.3)$$

Es führt dazu, dass für das Histogramm der Häufigkeitsverteilung die Fläche A zwischen Kurve und Abszisse gleich eins ist:

$$A = \sum_{v=1}^m h_v \cdot \Delta e = \sum_{v=1}^m \frac{n_v}{n \cdot \Delta e} \cdot \Delta e = \sum_{v=1}^m \frac{n_v}{n} = 1 \quad (5.4)$$

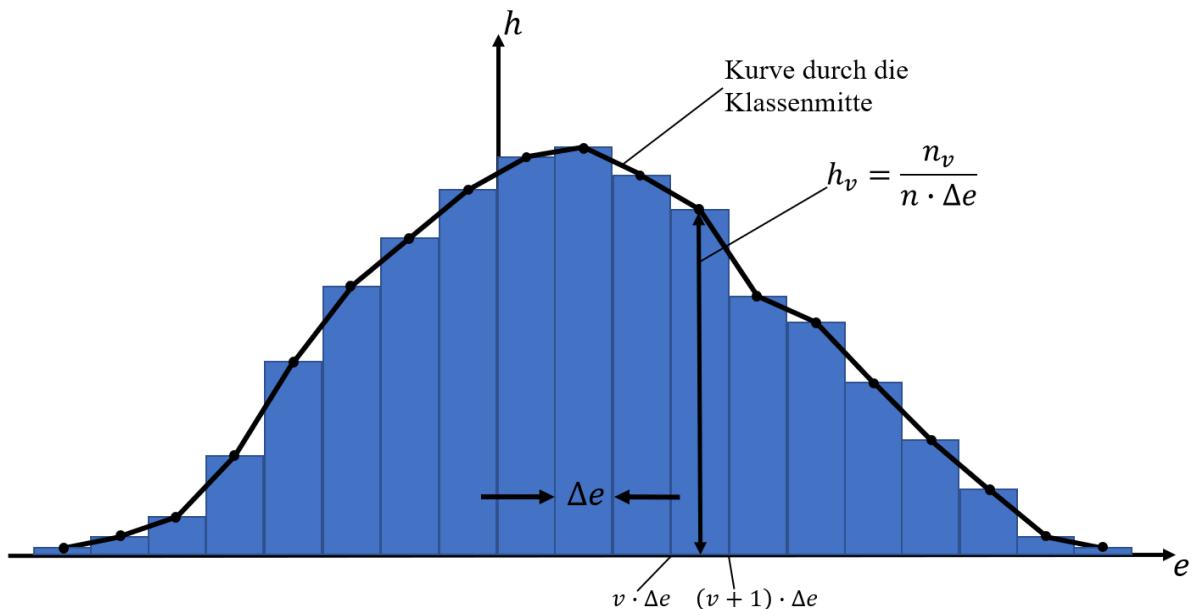


Abbildung 5.1: ein typisches Histogramm für die Häufigkeitsverteilung des Schätzfehlers e nach [09]

Die Klassenbreite Δe wird so gewählt, dass die Kurve durch die Klassenmitte „glatt“ ist. Nach der Literatur [09] kann die Klassenbreite Δe für normalverteilte Stichprobe häufig anhand der Standardabweichung s_e des Schätzfehlers gewählt:

$$\Delta e = \frac{3,49 \cdot s_e}{\sqrt[3]{n}} \quad (5.5)$$

wobei die Standardabweichung s_e der Stichprobe des Schätzfehlers e über die folgende Formel berechnet werden kann:

$$s_e = \frac{1}{n-1} \sum_{i=1}^n (e_i - \bar{e})^2 \quad (5.6)$$

Der Mittelwert \bar{e} des Schätzfehlers gilt dabei:

$$\bar{e} = \frac{1}{n} \sum_{i=1}^n e_i \quad (5.7)$$

5.1.2 Wurzel der mittleren Fehlerquadratsumme

Um die Schätzgüte der Schätzwerte zu beurteilen, wird auch häufig die Wurzel der mittleren Fehlerquadratsumme (engl. *Root Mean Square Error, RMSE*) benutzt. Der *Root Mean Square Error* für die Schätzung der Variable x wird über die folgende Formel definiert:

$$RMSE_x = \sqrt{\frac{1}{n-1} \cdot \sum_{i=1}^n (x_{\text{schätz}} - x_{\text{wahr}})^2} \quad (5.8)$$

wobei $x_{\text{schätz}}$ und x_{wahr} die Schätzwerte und die wahren Werte der Variable x bezeichnet.

Der *Root Mean Square Error* gibt an, wie stark eine Schätzung im Durchschnitt von den tatsächlichen Werten abweicht. Je größer der RMSE ist, desto schlechter ist die Schätzgüte.

5.2 Auslegung des UKF für den Schätzer der relativen Pose in Simulation

Das Verhalten des Unscented Kalman Filters (UKF) im Schätzer der relativen Pose hängt wesentlich von den Kovarianzmatrizen des Prozessrauschens \mathbf{Q}_k und des Beobachtungsrauschens $\mathbf{R}_{\text{lidar},k}$, $\mathbf{R}_{\text{kamera},k}$ und $\mathbf{R}_{\text{führend},k}$ ab. Deshalb spielt die Auslegung der Kovarianzmatrizen eine wichtige Rolle für die Performance des UKF. Meistens werden die Kovarianzmatrizen des Beobachtungsrauchens zuerst festgelegt. Danach kann die Kovarianzmatrix des Prozessrauschens als „Tuning-Parameter“ eingestellt werden.

5.2.1 Bestimmung der Kovarianz des Beobachtungsrauschens

Wie in Abschnitt 3.4 schon vorgestellt, werden für den Schätzer der relativen Pose zweier Busse drei verschiedene Arten der Beobachtungen berücksichtigt, sie sind die mit dem Lidar-Tracker und Kamera-Tracker ermittelte relative Pose ($[\Delta x_{\text{lidar},k}, \Delta y_{\text{lidar},k}, \Delta \varphi_{\text{lidar},k}]^T$ und $[\Delta x_{\text{kamera},k}, \Delta y_{\text{kamera},k}, \Delta \varphi_{\text{kamera},k}]^T$) sowie die beobachteten fahrdynamischen Kenngrößen des führenden Busses ($[v_{x2,\text{est}} \quad \omega_{z2,\text{imu}} \quad a_{x2,\text{imu}} \quad \beta_{2,\text{est}}]^T$).

Für den Kamera-Tracker wird das OBTS Aruco-Marker-Konzept („OBTS“ = One Big Two Small Aruco Markers) nach [14] verwendet. Dabei werden ein großer Aruco-Marker und zwei kleine Aruco-Marker an die Rücken des führenden Busses geklebt (siehe Abbildung 5.2). Die relative Pose zweier Busse kann mit dem Kamera-Tracker erfasst werden, wenn mindestens ein Aruco-Marker vom Kamera-Tracker detektiert wird.



Abbildung 5.2: OBTS Aruco-Marker-Konzept nach [14]

Laut [14] können die Kovarianzen des Beobachtungsrauschen der mit dem Kamera-Tracker ermittelten relativen Pose stark variieren, wenn sich die Anzahl der detektierten Aruco-Markers ändert. Je weniger Aruco-Markers vom Kamera-Tracker detektiert werden, desto größer ist die Kovarianz des Beobachtungsrauschen der ermittelten relativen Pose. In Abbildung 5.3 sind die Änderung des Schätzfehlers (Beobachtungsrauschen) der ermittelten relativen Pose dargestellt, wenn sich die Anzahl der detektierten Aruco-Markers im Bereich von 0 bis 3 ändert. Man kann sehen, dass der Schätzfehler der relativen Pose besonders groß ist, wenn der Kamera-Tracker nur einen oder keinen Aruco-Marker detektiert hat (siehe grauen Bereich in Abbildung 5.3). In diesem Fall wird diese ermittelte relative Pose als ungültige Beobachtung betrachtet und gar nicht vom UKF vertraut.

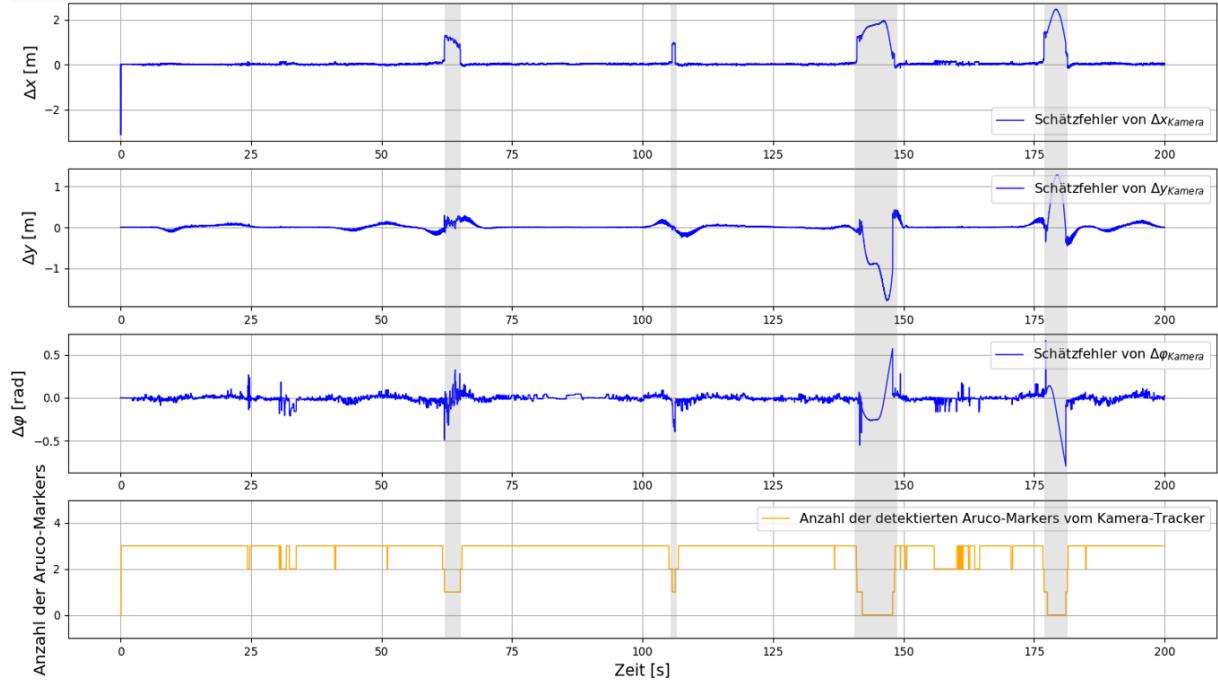


Abbildung 5.3: Änderung des Beobachtungsrauschen des Kamera-Trackers

Im Vergleich dazu verändert sich das Beobachtungsrauschen der mit dem Lidar-Tracker ermittelten relativen Pose nicht so stark, solange der Lidar-Tracker den führenden Bus richtig detektieren kann. Aber wenn der Lidar-Tracker aufgrund des begrenzten Blickfelds und

Reichweite den führenden Bus nicht detektieren kann, bietet er eine komplett falsche Information über die relativen Pose. Wenn diese falsche relative Pose nicht rechtzeitig erkannt und vom Fusionssystem vertraut sind, wird das Fusionsergebnis deutlich verschlechtert. Eine richtige und schnelle Fehlererkennung des Lidar-Trackers spielt daher eine wichtige Rolle für das Fusionsergebnis. Wie im Abschnitt 1.2.4 erläutert, kann der Lidar-Tracker gleich einen Fehler melden, wenn er keine geeigneten Objekte als den führenden Bus detektieren kann. Sobald der Schätzer der relativen Pose solche Fehlermeldung vom Lidar-Tracker empfängt, verlässt sich der Schätzer nicht mehr auf die Beobachtungen des Lidar-Trackers, bis der Lidar-Tracker den führenden Bus wieder detektiert.

In der Praxis können die Kovarianzmatrizen des Beobachtungsrauschens häufig durch drei Weise bestimmt werden:

1. Wenn die Beobachtungen die Messwerte der Sensoren sind, können bei der Festlegung der Kovarianz des Beobachtungsrauschens im Allgemeinen die Herstellerangaben der verwendeten Sensoren herangezogen werden.
2. Wenn die Beobachtungen nicht direkt mit den Sensoren gemessen werden, können die Kovarianzmatrizen des Beobachtungsrauschens häufig über die folgende Formel geschätzt werden [03]:

$$\mathbf{R}_k \approx \frac{1}{n-1} \sum_{i=1}^n (\mathbf{z}_i - \bar{\mathbf{z}}_i) \cdot (\mathbf{z}_i - \bar{\mathbf{z}}_i)^T \quad (5.9)$$

wobei \mathbf{z}_i die Beobachtungen bezeichnet, und $\bar{\mathbf{z}}_i$ ist der Mittelwert der Beobachtungen.

3. Es kommt auch häufig vor, dass die Kovarianzmatrizen des Beobachtungsrauschens als „Tuning-Parameter“ eingesetzt werden. Das qualitative Wissen über die Beobachtungen wird in die Schätzung der Kovarianzmatrizen eingebracht [03].

In dieser Arbeit werden die Kovarianzmatrizen des Beobachtungsrauschens zuerst über die Formel (5.9) grob geschätzt werden, um zu vergleichen, welche Beobachtungen eine größere oder kleinere Kovarianz haben. Danach werden die Kovarianzmatrizen des Beobachtungsrauschens nach dieser groben Schätzung eingestellt. Werden die Elemente der Kovarianzmatrix des Beobachtungsrauschens auf höhere Werte gesetzt, bedeutet, dass sich das UKF weniger auf diese Beobachtung verlässt, weil sie zu viel Beobachtungrauschen enthält:

$$\mathbf{R}_{lidar} = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.05 \end{bmatrix} \quad (5.10)$$

$$\mathbf{R}_{kamera,drei} = \begin{bmatrix} 0.3 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.05 \end{bmatrix} \quad (5.11)$$

$$\mathbf{R}_{kamera,zwei} = \begin{bmatrix} 0.6 & 0 & 0 \\ 0 & 0.05 & 0 \\ 0 & 0 & 0.15 \end{bmatrix} \quad (5.12)$$

$$\mathbf{R}_{\text{führend}} = \begin{bmatrix} 1e^{-4} & 0 & 0 & 0 \\ 0 & 1e^{-4} & 0 & 0 \\ 0 & 0 & 0.8 & 0 \\ 0 & 0 & 0 & 1e^{-2} \end{bmatrix} \quad (5.13)$$

wobei $\mathbf{R}_{\text{kamera,drei}}$ und $\mathbf{R}_{\text{kamera,zwei}}$ die Kovarianzmatrizen des Beobachtungsrauschenks der mit dem Kamera-Tracker ermittelten relativen Pose ($\Delta x, \Delta y, \Delta \varphi$) zweier Busse bezeichnen, wenn der Kamera-Tracker drei oder zwei Aruco-Markers detektiert hat. Man kann sehen, dass die Elemente der Kovarianzmatrix $\mathbf{R}_{\text{kamera,zwei}}$ auf höhere Werte als die Elemente der Kovarianzmatrizen $\mathbf{R}_{\text{kamera,drei}}$ gesetzt sind, weil der Schätzfehler der relativen Pose größer ist, wenn nur zwei Aruco-Marker detektiert werden (vgl. Abbildung 5.3).

Die Elemente der diagonalen Kovarianzmatrix $\mathbf{R}_{\text{führend}}$ bezeichnen dabei die Kovarianz der beobachteten fahrdynamischen Kenngrößen des führenden Busses, nämlich $v_{x2,est}$, $\omega_{z2,imu}$, $a_{x2,imu}$ und $\beta_{2,est}$.

5.2.2 Einstellung der Kovarianzmatrix des Prozessrauschenks

Nach der Festlegung der Kovarianzmatrizen des Beobachtungsrauschenks kann man nun die Kovarianzmatrix \mathbf{Q}_k (Dimension 3×3) des Prozessrauschenks $\mathbf{v}_k = [v_{a_x,k} \ v_{\omega_z,k} \ v_{\beta,k}]^T$ als „Tuning-Parameter“ einstellen, weil die Unsicherheit des Prozessmodells häufig weder messbar noch schätzbar ist. Je größer die Elemente der Kovarianzmatrix \mathbf{Q}_k eingestellt werden, desto größer ist die Unsicherheit des Prozessmodells. In diesem Fall wird das Prozessmodell weniger vom UKF vertraut. Dabei verlässt sich das UKF mehr auf die neu reinkommenden Beobachtungen.

Laut Literatur [07] soll der Schätzfehler des Zustands für etwa 68% der Zeit innerhalb der prädizierten $\pm \sigma$ -Grenzen liegen, wenn das Kalman Filter richtig arbeitet. Dabei bezeichnet σ die Standardabweichung des prädizierten Zustands, nämlich die Wurzel des entsprechenden diagonalen Elementes der Kovarianzmatrix \mathbf{P}_k des Zustandsvektors (vgl. Abschnitt 2.4.3), z.B. die Wurzel des diagonalen Elementes P_{11} der Kovarianzmatrix \mathbf{P}_k bedeutet dabei die prädizierte Standardabweichung des longitudinalen Abstands Δx zweier Busse.

Zunächst wird es versucht, die Kovarianzmatrix \mathbf{Q}_k auf Nullmatrix zu setzen, das bedeutet, dass kein Prozessrauschen auf das Prozessmodell addiert wird. In Abbildung 5.4 sind die Schätzfehler der vom Schätzer geschätzten relativen Pose ($\Delta x_{\text{Fusion}}, \Delta y_{\text{Fusion}}, \Delta \varphi_{\text{Fusion}}$) durch die rote Linien dargestellt. Die prädizierten $\pm \sigma$ -Grenzen für die Schätzfehler der relativen Pose sind dabei durch die schwarzen gestrichelten Linien gezeichnet.

Auffällig ist die Divergenz bei Schätzung der relativen Pose in Abbildung 5.4 zu sehen. Der Grund für die Divergenz der Schätzung liegt darin, dass sich das UKF lediglich auf das Prozessmodell, nicht auf die Beobachtungen verlässt, wenn die Kovarianzmatrix \mathbf{Q}_k des Prozessrauschenks auf Nullmatrix gesetzt ist. In diesem Fall konvergieren die diagonalen Elemente vom Kalman Gain \mathbf{K}_k meistens gegen Null. Das heißt, dass die geschätzte Zustände beim UKF über eine Weile nicht mehr durch die neuen Beobachtungen korrigiert werden können.

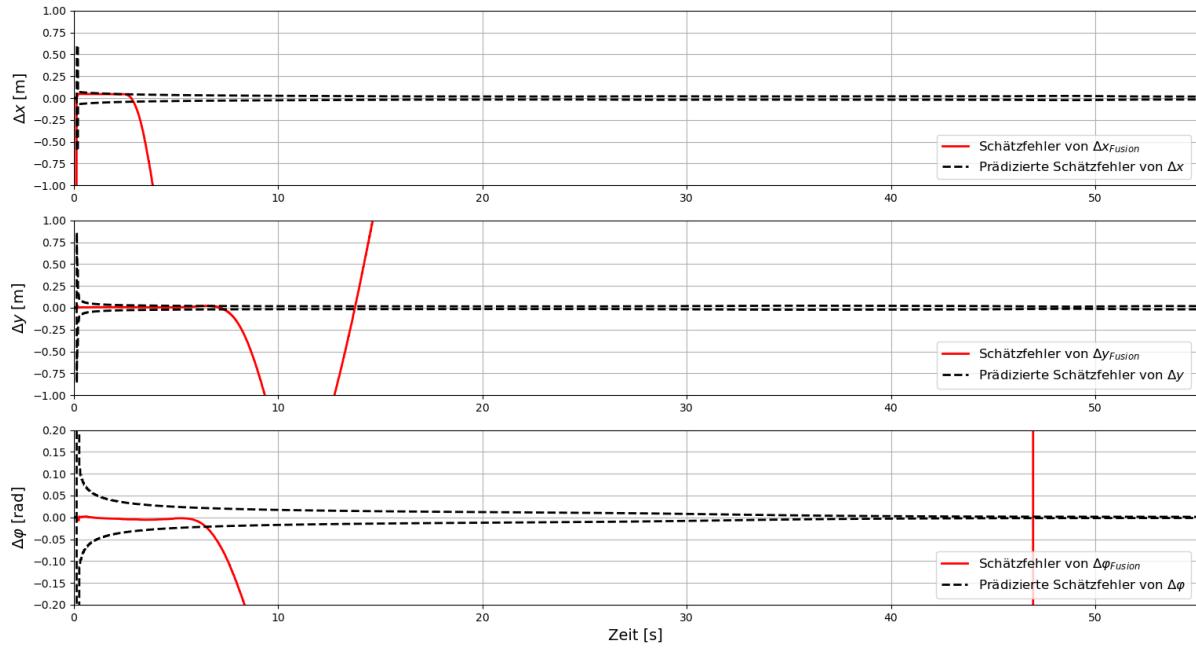


Abbildung 5.4: Darstellung des Schätzfehlers der relativen Pose, wenn Q_k auf Nullmatrix gesetzt ist

Ist die diagonalen Elemente der Kovarianzmatrix Q_k auf sehr kleine Werte, nämlich $1e^{-3}$, gesetzt:

$$Q_k = \begin{bmatrix} 1e^{-3} & 0 & 0 \\ 0 & 1e^{-3} & 0 \\ 0 & 0 & 1e^{-3} \end{bmatrix} \quad (5.14)$$

wird das Problem der Divergenz der Schätzung schon gelöst (siehe Abbildung 5.5). Jedoch liegen die Schätzfehler meistens außerhalb der schwarzen gestrichelten Linien, weil sich das UKF nur wenig auf die neuen Beobachtungen verlässt, somit die Schätzfehler nicht schnell durch die neuen Beobachtungen reduziert werden können.

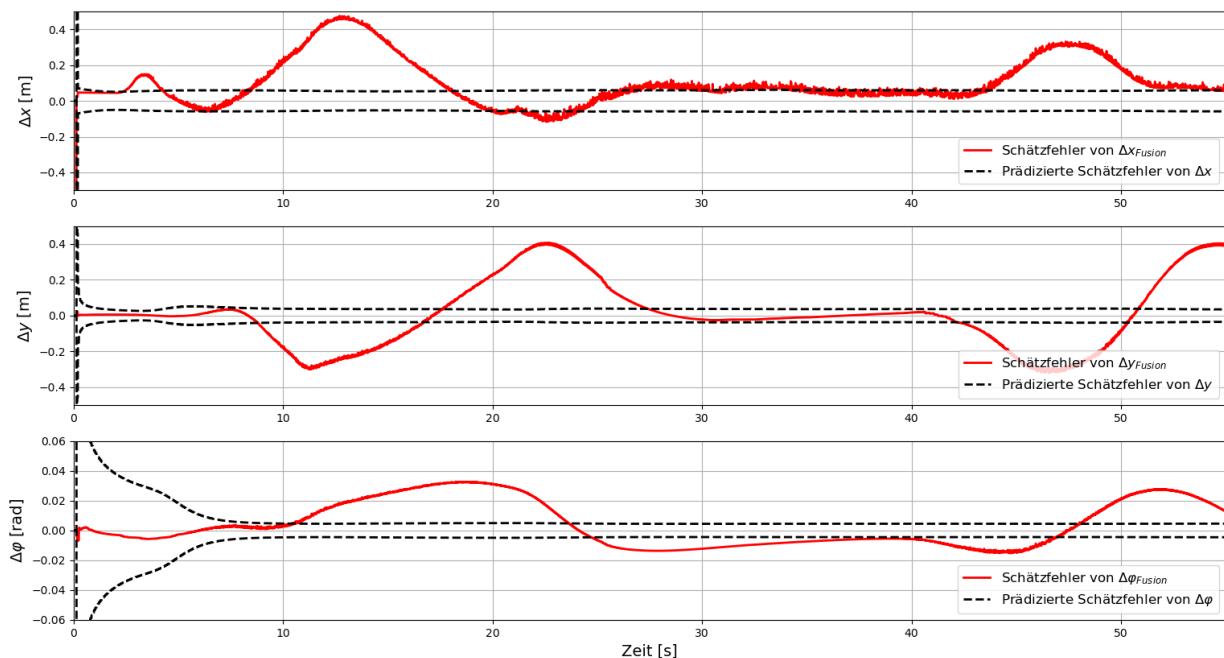


Abbildung 5.5: Darstellung des Schätzfehlers der relativen Pose, wenn Q_k auf sehr kleine Werte gesetzt ist

Wenn die Kovarianzmatrix \mathbf{Q}_k noch größer wie folgt gesetzt ist:

$$\mathbf{Q}_k = \begin{bmatrix} 1.0 & 0 & 0 \\ 0 & 5.0 & 0 \\ 0 & 0 & 1.0 \end{bmatrix} \quad (5.15)$$

bleiben die Schätzfehler der relativen Pose für etwa 68% der Zeit innerhalb der schwarzen gestrichelten Linien (siehe Abbildung 5.6), das heißt, dass die Schätzwerte der relativen Pose die erwarteten Schätzfehler haben. Es ist aussagekräftig nachzuweisen, dass das UKF in diesem Fall richtig arbeitet.

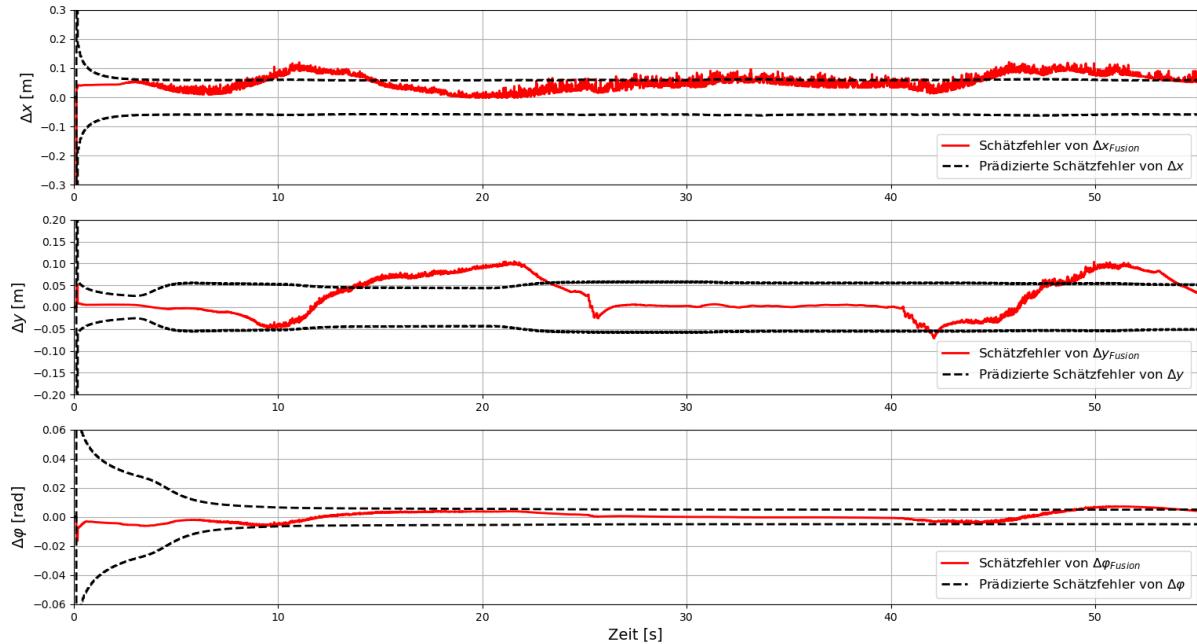


Abbildung 5.6: Darstellung des Schätzfehlers der relativen Pose, wenn \mathbf{Q}_k auf sehr große Werte gesetzt ist

5.2.3 Initialisierung des UKF im Schätzer der relativen Pose

Eine unpassende Initialisierung des UKF kann die Performance des UKF verschlechtern, und manchmal sogar zur Divergenz der Schätzung führen. Bei Initialisierung des UKF müssen die Anfangswerte des Zustandsvektors \mathbf{x}_0 sowie die Anfangswerte der zugehörige Kovarianzmatrix \mathbf{P}_0 eingestellt werden.

Für den Zustandsvektors $\mathbf{x}_k = [\Delta x_k \quad \Delta y_k \quad \Delta \varphi_k \quad v_{x2,k} \quad \omega_{z2,k} \quad a_{x2,k} \quad \beta_{2,k}]^T$ im Schätzer der relativen Pose sind dabei ein geeigneter Anfangszustandsvektor \mathbf{x}_0 gesucht:

$$\mathbf{x}_0 = \begin{bmatrix} \Delta x_0 \\ \Delta y_0 \\ \Delta \varphi_0 \\ v_{x2,0} \\ \omega_{z2,0} \\ a_{x2,0} \\ \beta_{2,0} \end{bmatrix} \quad (5.16)$$

Für die Anfangszustände $[\Delta x_0 \quad \Delta y_0 \quad \Delta \varphi_0]^T$ kann die erste Beobachtung der relative Pose aus dem Lidar- oder Kamera-Tracker eingesetzt werden:

$$\begin{bmatrix} \Delta x_0 \\ \Delta y_0 \\ \Delta \varphi_0 \end{bmatrix} = \mathbf{z}_{kamera,0} \text{ oder } \begin{bmatrix} \Delta x_0 \\ \Delta y_0 \\ \Delta \varphi_0 \end{bmatrix} = \mathbf{z}_{lidar,0} \quad (5.17)$$

Denn es kann nicht garantiert sein, dass bei Initialisierung des UKF die V2V-Kommunikation verbunden ist, und die Beobachtungen für die fahrdynamischen Kenngrößen des führenden Busses beim Prozessor vorliegen können, werden die Anfangszustände $[v_{x2,0} \quad \omega_{z2,0} \quad a_{x2,0} \quad \beta_{2,0}]^T$ einfach auf Null gesetzt:

$$\begin{bmatrix} v_{x2,0} \\ \omega_{z2,0} \\ a_{x2,0} \\ \beta_{2,0} \end{bmatrix} = \mathbf{0}_{4 \times 1} \quad (5.18)$$

Die Anfangswerte der diagonalen Kovarianzmatrix \mathbf{P}_0 des Anfangszustandsvektors können häufig auf hohe Werte gesetzt werden, damit die Schätzung der Zustände schnell konvergieren kann. Jedoch kann die Anfangswerte der Kovarianzmatrix \mathbf{P}_0 nicht zu groß eingestellt werden, weil eine zu große Kovarianzmatrix \mathbf{P}_0 häufig zur Divergenz der Schätzung führen kann. Hier wird \mathbf{P}_0 wie folgt eingestellt:

$$\mathbf{P}_0 = diag([1; 1; 1; 0,1; 0,1; 0,1; 0,1]) \quad (5.19)$$

wobei $diag(\dots)$ eine diagonale Matrix mit den diagonalen Elementen $[1; 1; 1; 0,1; 0,1; 0,1; 0,1]$ bezeichnet.

5.3 Analyse der Ergebnisse in Simulation

Die Ergebnisse der Schätzungen für die longitudinalen Geschwindigkeit, den Schwimmwinkel und die relative Pose zweier Busse in der Simulation werden nun bewertet. Dafür werden die Teststrecke, das Testszenario und das Fahrmanöver in Abschnitt 4.2 verwendet (vgl. Abbildung 4.2 und Abbildung 4.3).

5.3.1 Schätzung der longitudinalen Geschwindigkeit des Fahrzeugs

Der Geschwindigkeit-Schätzer besteht aus dem Detektor des Radschlupfs und dem Kalman Filter. Die Parameter des Detektors des Radschlupfs werden im Simulation wie folgt eingestellt:

$$n_{ZF} = 20; \quad \Delta \omega_{threshold} = 2,0; \quad \Delta v_{threshold} = 0,4 \quad (5.20)$$

Die Auslegung der Parameter des Kalman Filters erfolgt durch die gleich Weise wie in Abschnitt 5.2:

$$\mathbf{Q}_k = \begin{bmatrix} 100 & 0 \\ 0 & 0,1 \end{bmatrix} \quad (5.21)$$

$$\mathbf{R}_k = 0,1 \quad (5.22)$$

$$\mathbf{P}_0 = \begin{bmatrix} 1,0 & 0 \\ 0 & 1,0 \end{bmatrix} \quad (5.23)$$

Die Schätzung der longitudinalen Geschwindigkeit des führenden Busses $v_{x2,est}$ wird durchgeführt, nur wenn die V2V-Kommunikation zweier Busse angeschaltet ist. Die Schätzfehler der geschätzten longitudinalen Geschwindigkeit des verfolgenden Busses $v_{x1,est}$ und des führenden Busses $v_{x2,est}$ sind in Abbildung 5.7 dargestellt. Dabei ist die V2V-Kommunikation zweier Busses immer angeschaltet.

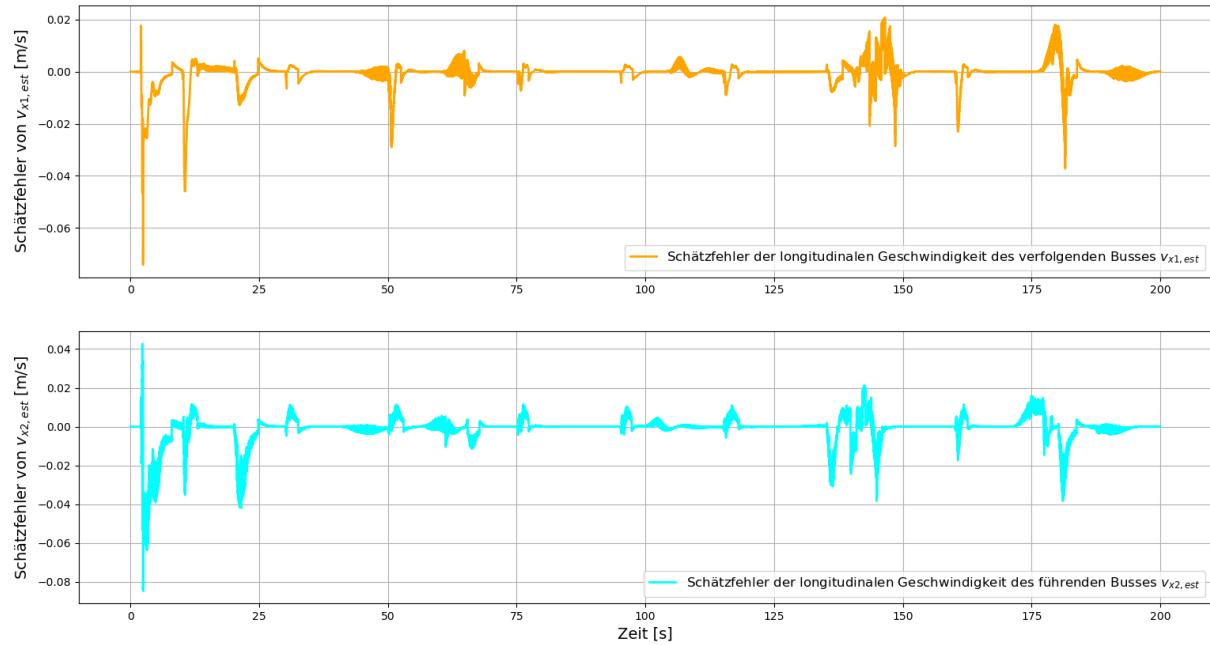


Abbildung 5.7: Schätzfehler der longitudinalen Geschwindigkeit des verfolgenden Busses $v_{x1,est}$ und des führenden Busses $v_{x2,est}$

Tabelle 5.1 zeigt den *Root Mean Square Error* für die Schätzung der longitudinalen Geschwindigkeit.

	<i>Root Mean Square Error</i> [m/s]
$v_{x1,est}$	$4,51e^{-3}$
$v_{x2,est}$	$6,86e^{-3}$

Tabelle 5.1: Root Mean Square Error von $v_{x1,est}$ und $v_{x2,est}$

Man kann sehen, dass der Schätzfehler von $v_{x2,est}$ wenig größer als $v_{x1,est}$ ist, weil die Messdaten für die Schätzung der longitudinalen Geschwindigkeit des führenden Busse aufgrund der V2V-Kommunikation nur eine niedrigere Frequenz (50 Hz) hat.

5.3.2 Schätzung des Schwimmwinkels des Fahrzeugs

Der Schwimmwinkel-Schätzer benutzt dabei ein Unscented Kalman Filter, dessen Parameter in Simulation wie folgt ausgelegt werden:

$$\mathbf{Q}_k = \begin{bmatrix} 1,0 & 0 & 0 \\ 0 & 1,0 & 0 \\ 0 & 0 & 5,0 \end{bmatrix} \quad (5.24)$$

$$\mathbf{R}_k = \begin{bmatrix} 0,5 & 0 & 0 \\ 0 & 1e^{-4} & 0 \\ 0 & 0 & 1e^{-4} \end{bmatrix} \quad (5.25)$$

$$\mathbf{P}_0 = \begin{bmatrix} 1,0 & 0 & 0 \\ 0 & 1,0 & 0 \\ 0 & 0 & 1,0 \end{bmatrix} \quad (5.26)$$

Die Schätzung des Schwimmwinkels des führenden Busses $\beta_{2,est}$ wird durchgeführt, nur wenn die V2V-Kommunikation zweier Busse angeschaltet ist. Abbildung 5.8 zeigen die Verläufe für die Schätzwerte und die wahren Werte des Schwimmwinkels des verfolgenden Busses β_1 und des führenden Busses β_2 . In Abbildung 5.9 sind die zugehörigen Schätzfehler für die Schätzwerte $\beta_{1,est}$ und $\beta_{2,est}$ dargestellt.

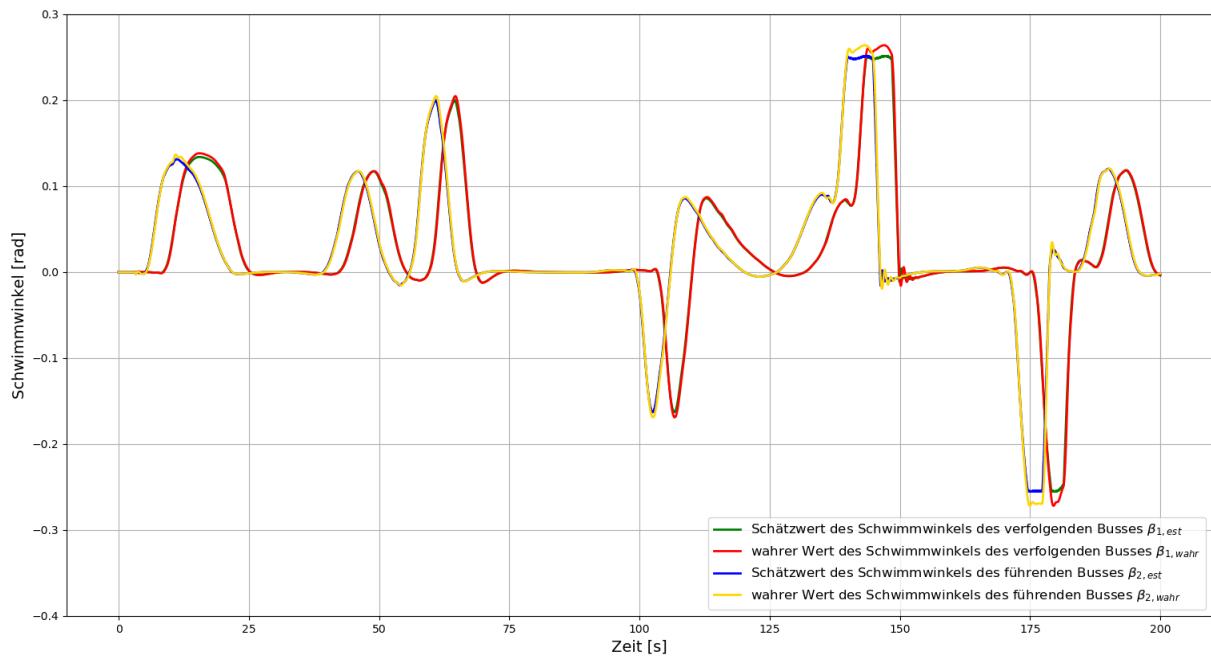


Abbildung 5.8: Verläufe für die Schätzwerte und die wahren Werte des Schwimmwinkels des verfolgenden Busses β_1 und des führenden Busses β_2

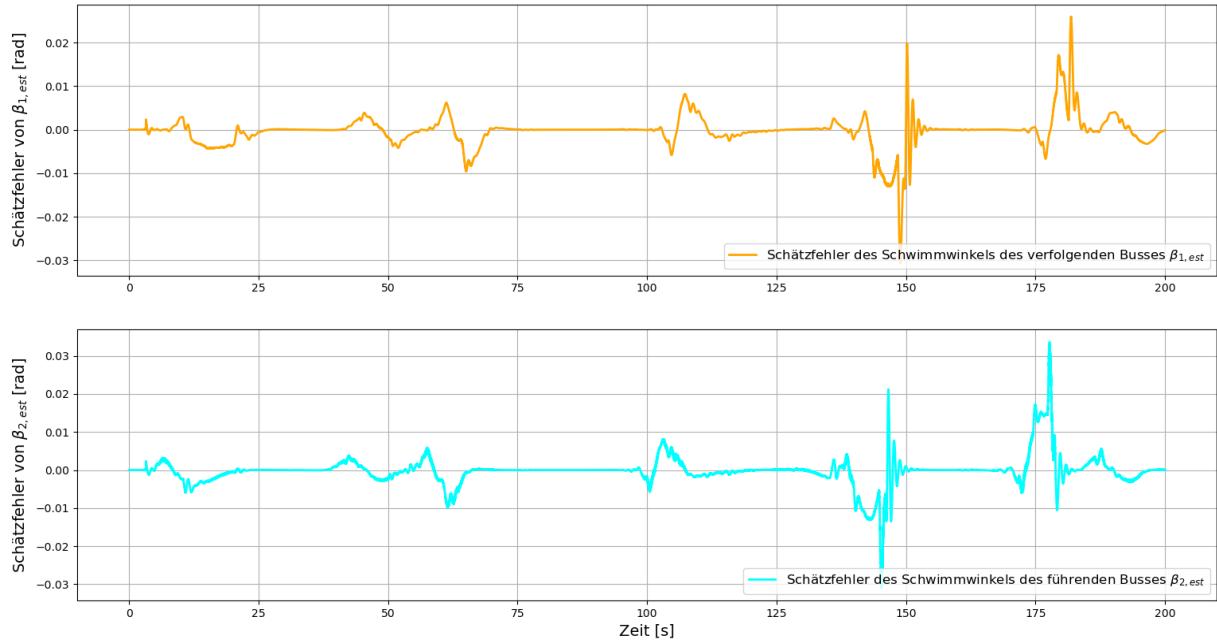


Abbildung 5.9: Schätzfehler des Schwimmwinkels des verfolgenden Busses β_1 und des führenden Busses β_2

Auffällig in der Abbildung 5.8 und Abbildung 5.9 sind die großen Schätzfehlerspitzen, wenn der Schwimmwinkel einen großen Wert hat. Wie bereits in Abschnitt 3.3 erläutert, kann sich die Schätzgenauigkeit des Schwimmwinkel-Schätzers bei großem Schwimmwinkel und somit großer Querbeschleunigung verschlechtern, weil eine konstante Schräglaufseitensteifigkeit (vgl. Tabelle 4.1) in diesem Fall zur Beschreibung des Zusammenhangs zwischen der Seitenkraft und dem Schräglauwinkel nicht genau genug ist.

In Tabelle 5.2 ist der *Root Mean Square Error* für die Schätzung des Schwimmwinkels des verfolgenden Busses und des führenden Busses zu sehen.

	<i>Root Mean Square Error [rad]</i>
$\beta_{1,est}$	$3,49e^{-3}$
$\beta_{2,est}$	$3,70e^{-3}$

Tabelle 5.2: Root Mean Square Error von $\beta_{1,est}$ und $\beta_{2,est}$

5.3.3 Schätzung der relativen Pose beim Platooning

Im Schätzer der relative Pose werden die ermittelten relativen Posen zweier Busse aus zwei Lidarsensoren und der Monokamera, sowie die rekonstruierten fahrdynamischen Kenngrößen des führenden Busses als Beobachtungen betrachtet, welche die Schätzungen für den Zustandsvektor $\mathbf{x}_k = [\Delta x_k \quad \Delta y_k \quad \Delta \varphi_k \quad v_{x2,k} \quad \omega_{z2,k} \quad a_{x2,k} \quad \beta_{2,k}]^T$ korrigieren können. Der Schätzer bekommt die Beobachtungen für die fahrdynamischen Kenngrößen des

führenden Busses $\mathbf{z}_{\text{führend}} = [v_{x2,est} \quad \omega_{z2,imu} \quad a_{x2,imu} \quad \beta_{2,est}]^T$, nur wenn die V2V-Kommunikation zwischen zwei Bussen angeschaltet ist.

In Simulation wird die Performance des Schätzers der relativen Pose in verschiedenen Fällen untersucht und bewertet. Solche Testfälle, die durch verschiedene Buchstaben gekennzeichnet sind, sind in Tabelle 5.3 dargestellt. Z.B. der erste Testfall mit der Bezeichnung „LKV“ bedeutet, dass sowohl zwei Lidar und die Kamera, als auch die V2V-Kommunikation zwischen zwei Bussen angeschaltet sind. Im Gegensatz dazu wird die Kamera im zweiten Testfall mit der Bezeichnung „LV“ ausgeschaltet, und nur zwei Lidar und die V2V-Kommunikation sind eingeschaltet. Der dritte Testfall „KV“ verzichtet dabei auf die Beobachtungen von Lidarsensoren, und beim vierten Testfall „LK“ ist die V2V-Kommunikation zweier Busse abgebrochen. Im fünften Testfall mit der Bezeichnung „V“ sind zwei Lidarsensoren, der Kamerasensor sowie die V2V-Kommunikation zunächst angeschaltet, aber über einer Weile sind zwei Lidarsensoren und der Kamerasensor gleichzeitig ausgeschaltet, jedoch ist die V2V-Kommunikation zweier Busse noch verbunden. Im letzten Testfall „Mode Switch“ wird es untersucht, ob der Schätzer der relativen Pose auch richtig arbeiten kann, wenn zwei Lidarsensoren, der Kamerasensor sowie die V2V-Kommunikation nacheinander frequent ein- und ausgeschaltet werden.

1	LKV	Zwei Lidar an, Kamera an, V2V-Kommunikation an
2	LV	Zwei Lidar an, Kamera aus, V2V-Kommunikation an
3	KV	Zwei Lidar aus, Kamera an, V2V-Kommunikation an
4	LK	Zwei Lidar an, Kamera an, V2V-Kommunikation aus
5	V	Zwei Lidar aus, Kamera aus, V2V-Kommunikation an
6	Mode Switch	Zwei Lidar, Kamera und V2V-Kommunikation werden nacheinander frequent ein- oder ausgeschaltet.

Tabelle 5.3: Testfälle für den Schätzer der relativen Pose in der Simulation

5.3.3.1 Testfall 1: LKV

Für die erste Untersuchung wird der allgemeine Betriebszustand „LKV“ des Schätzers der relativen Pose getestet. Dabei wird es garantiert sein, dass zwei Lidarsensoren, der Kamerasensor sowie die V2V-Kommunikation im ganzen Zeitraum von 200 Sekunden richtig arbeiten können.

In Abbildung 5.10 sind die Verläufe des Schätzfehlers der relativen Pose (Δx , Δy , $\Delta \varphi$) für den Testfall „LKV“ dargestellt. Zum Vergleichszweck sind die Schätzfehler der ermittelten relativen Pose vom Kamera-Tracker, Lidar-Tracker sowie Fusionsalgorithmus jeweils durch blaue, grüne und rote Linien dargestellt (Anmerkung: „FusionLKV“ = Fusionsalgorithmus, wenn Lidarsensoren, Kamerasensor sowie V2V-Kommunikation angeschaltet sind).

Abbildung 5.11 zeigt die gleichen Verläufe des Schätzfehlers, jedoch sind die Verläufe vertikal vergrößert, um die Größe des Schätzfehlers vergleichen zu können.

In Abbildung 5.10 sind mehrere Schätzfehlerspitzen der ermittelten relativen Pose vom Lidar-Tracker und Kamera-Tracker zu sehen (siehe blaue und grüne Linien im gelben Bereich). Diese besonders große Schätzfehler treten auf, wenn der führende Bus plötzlich eine scharfe Abbiegung hat. In diesem Fall können die Kamera- und Lidarsensoren den führenden Bus nicht gut detektieren. Im Vergleich dazu bleiben die Schätzfehler der relativen Pose vom Fusionsalgorithmus „LKV“ immer in einem kleinen Wertebereich. Denn der Lidar-Tracker kann gleich einen Fehler melden, wenn er den führenden Bus nicht detektieren kann (siehe Abschnitt 1.2.4). In diesem Fall wird die ermittelte relative Pose des Lidar-Trackers nicht mehr vom Schätzer der relativen Pose vertraut. Gleichfalls kann der Schätzer der relativen Pose die Glaubwürdigkeit der ermittelten relativen Pose des Kamera-Trackers anhand der Anzahl der detektierten Aruco-Markers beurteilen. Wenn der Kamera-Tracker nur einen oder keinen Aruco-Marker detektiert hat, wird die ermittelten relative Pose nicht mehr vom Schätzer vertraut (siehe Abschnitt 5.2.1). Auf diese Weise wird das Fusionsergebnis des Schätzers nicht von falschen Beobachtungen der relativen Pose des Kamera- und Lidar-Trackers verschlechtert. Der Schätzer kann die relativen Pose zweier Busse durch den Prädiktionsschritt des UKF noch weiter schätzen, auch wenn keine neuen gültigen Beobachtungen der relativen Pose vom Kamera- oder Lidar-Tracker reinkommen.

Aus Abbildung 5.11 ist es deutlich zu sehen, dass der Schätzfehler der relativen Pose vom Fusionsalgorithmus „LKV“ im Vergleich zum Kamera- und Lidar-Tracker am kleinsten ist.

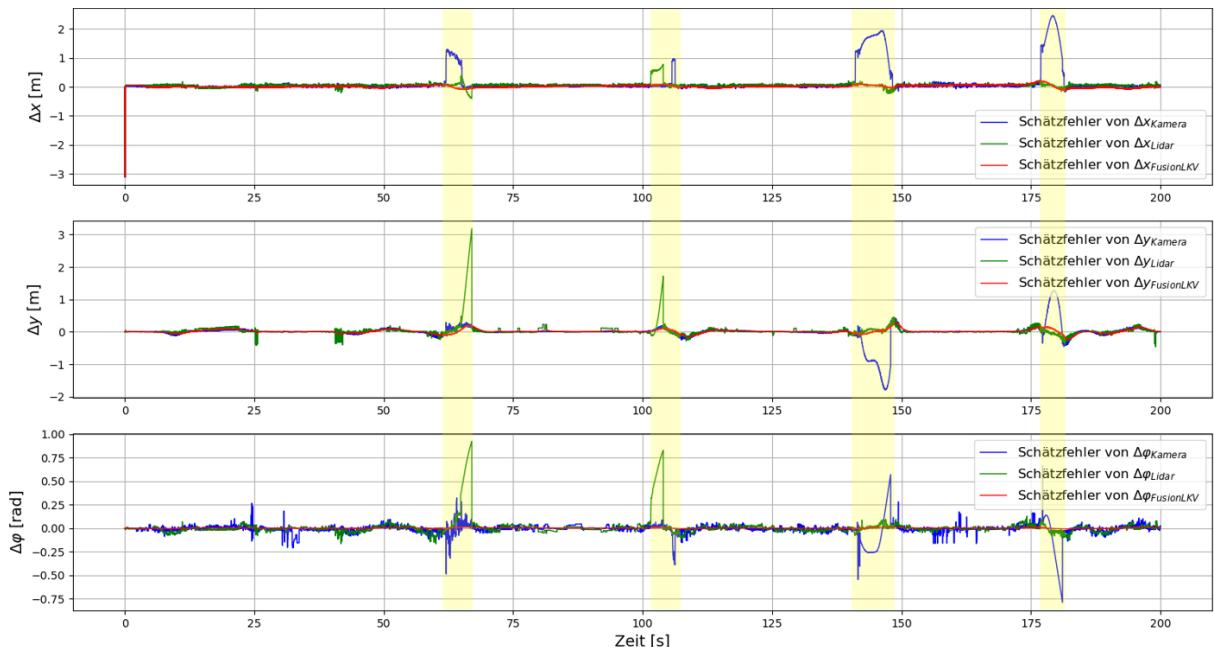


Abbildung 5.10: Verläufe des Schätzfehlers der relativen Pose im Testfall „LKV“

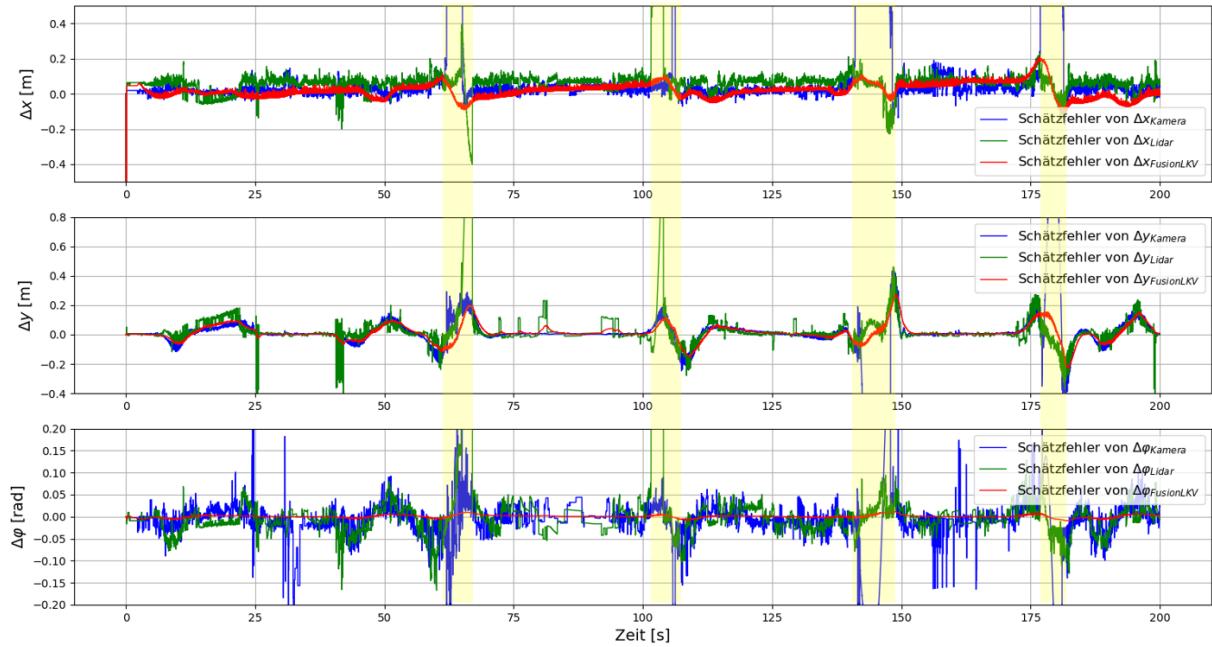


Abbildung 5.11: Verläufe des Schätzfehlers der relativen Pose im Testfall „LKV“ (vertikal vergrößert)

Um die Wahrscheinlichkeitsdichte des Schätzfehlers der relativen Pose zu beschreiben, sind die Häufigkeitsverteilungen für den Schätzfehler von Δx , Δy und $\Delta \varphi$ in Abbildung 5.12, Abbildung 5.13 und Abbildung 5.14 dargestellt. Die Klassenbreite der Häufigkeitsverteilungen für den Schätzfehler von Δx , Δy und $\Delta \varphi$ sind jeweils $5e^{-3}m$, $5e^{-3}m$ und $5e^{-3}rad$. In jede Abbildung zeigt die obere Hälfte die ganzen Ansicht der Häufigkeitsverteilung, während in unterer Hälfte der Abbildung die Häufigkeitsverteilung vertikal vergrößert wird, um den Ausreißer des Schätzfehlers klar darstellen zu können. Aus Abbildung 5.12, Abbildung 5.13 und Abbildung 5.14 kann man sehen, dass die Häufigkeitsverteilung des Schätzfehlers der relativen Pose vom Fusionsalgorithmus im Allgemeinen eine schmalere Form hat und nur wenige Ausreißer enthält. Damit kann es nachgewiesen werden, dass die Schätzung der relativen Pose vom Fusionsalgorithmus als die Schätzung vom Lidar- und Kamera-Tracker genauer ist.

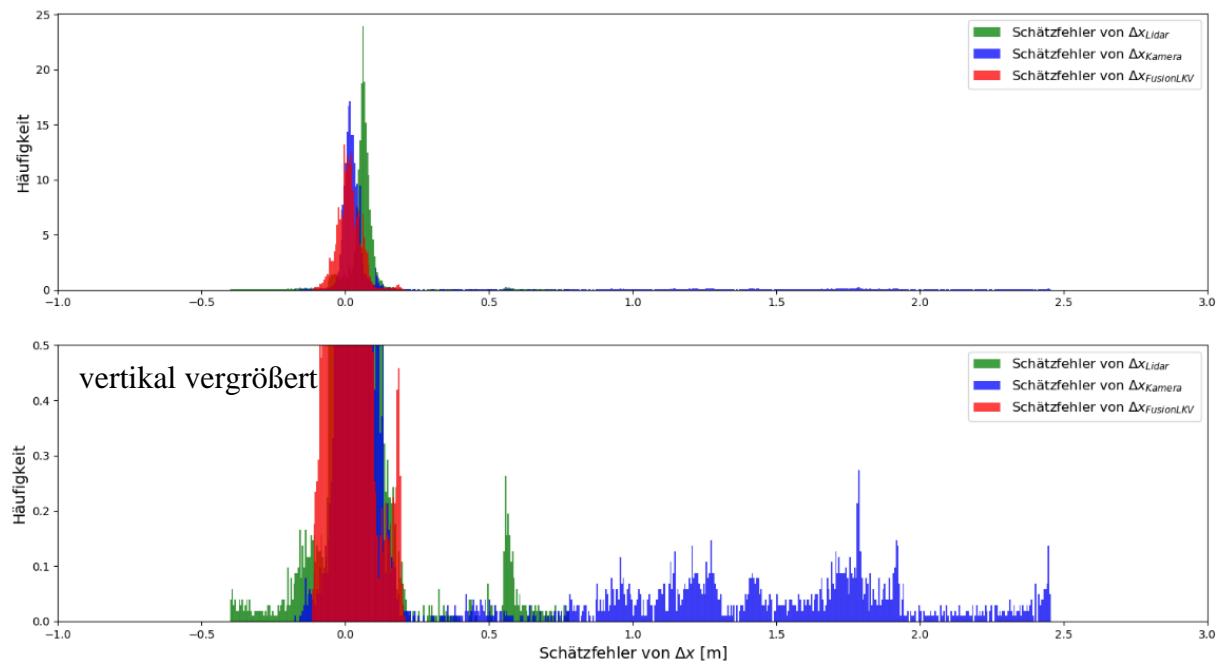


Abbildung 5.12: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „LKV“

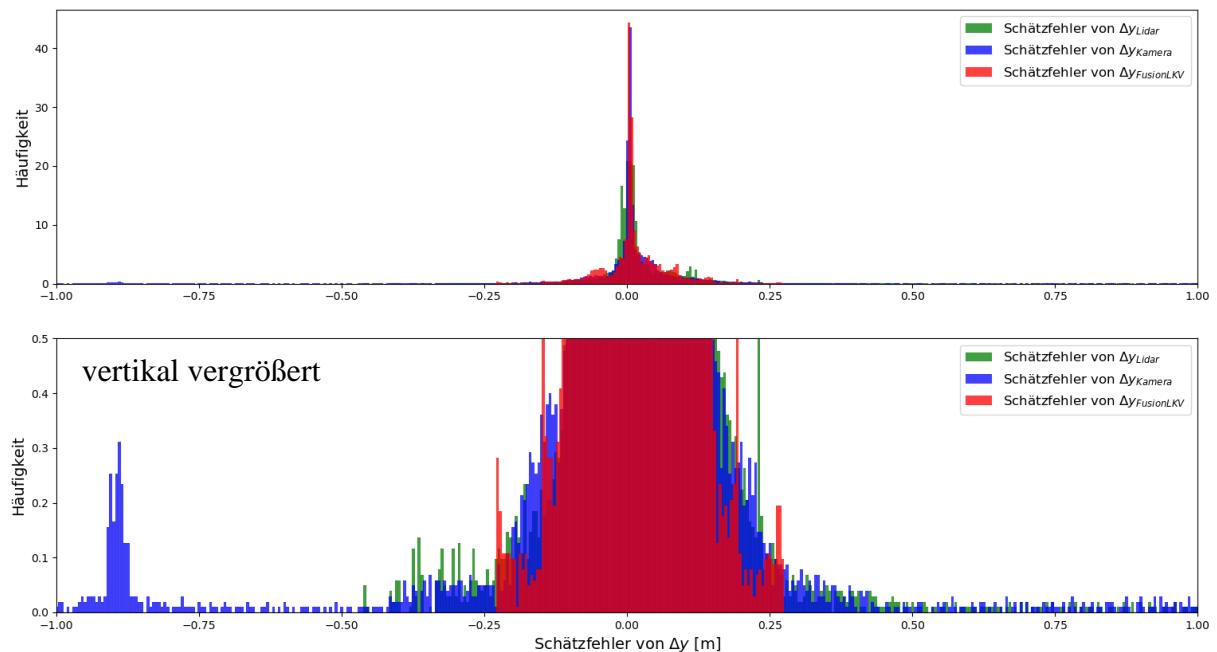


Abbildung 5.13: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „LKV“

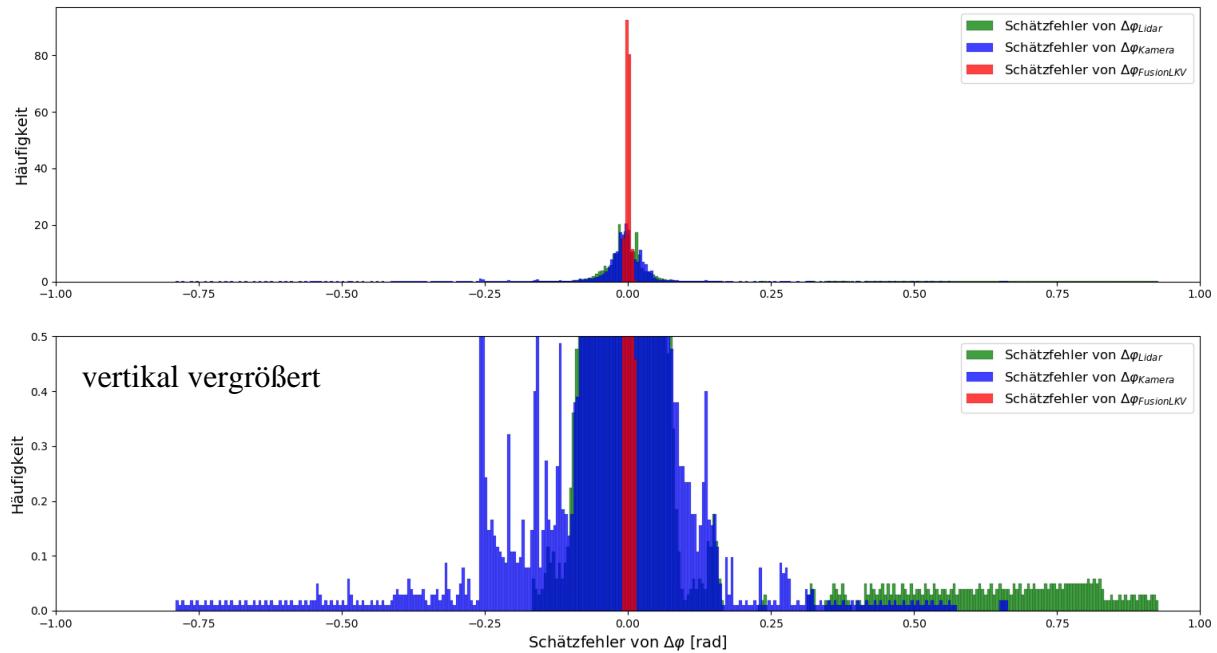


Abbildung 5.14: Häufigkeitsverteilung des Schätzfehlers von $\Delta\varphi$ im Testfall „LKV“

5.3.3.2 Testfall 2: LV

In zweiter Untersuchung wird der Testfall „LV“ durchgeführt. Im Vergleich zum ersten Testfall „LKV“ wird nun die Kamera ausgeschaltet. Das heißt, dass die Schätzung der relativen Pose lediglich durch die Beobachtungen der Lidarsensoren korrigiert wird. Dabei wird die V2V-Kommunikation auch immer angeschaltet. Gleichfalls werden hier die Verläufe des Schätzfehlers der relativen Pose für den Testfall „LV“ in Abbildung 5.15 und Abbildung 5.16 dargestellt. Die Häufigkeitsverteilungen für den Schätzfehler von Δx , Δy und $\Delta\varphi$ sind in Abbildung 5.17, Abbildung 5.18 und Abbildung 5.19 dargestellt.

Man kann sehen, dass bei scharfer Abbiegung der Schätzfehler der relativen Pose vom Fusionsalgorithmus „LV“ auch immer klein halten kann. In Abbildungen für die Häufigkeitsverteilungen ist es auffällig zu sehen, dass die Schätzung der relativen Pose vom Fusionsalgorithmus „LV“ als die Schätzung des Lidar-Trackers genauer ist. Diese Ergebnisse zeigen, dass die Genauigkeit der Schätzung der relativen Pose beim Ausschalten der Kamera nicht deutlich verschlechtert wird, solange die V2V-Kommunikation zweier Busse angeschaltet ist.

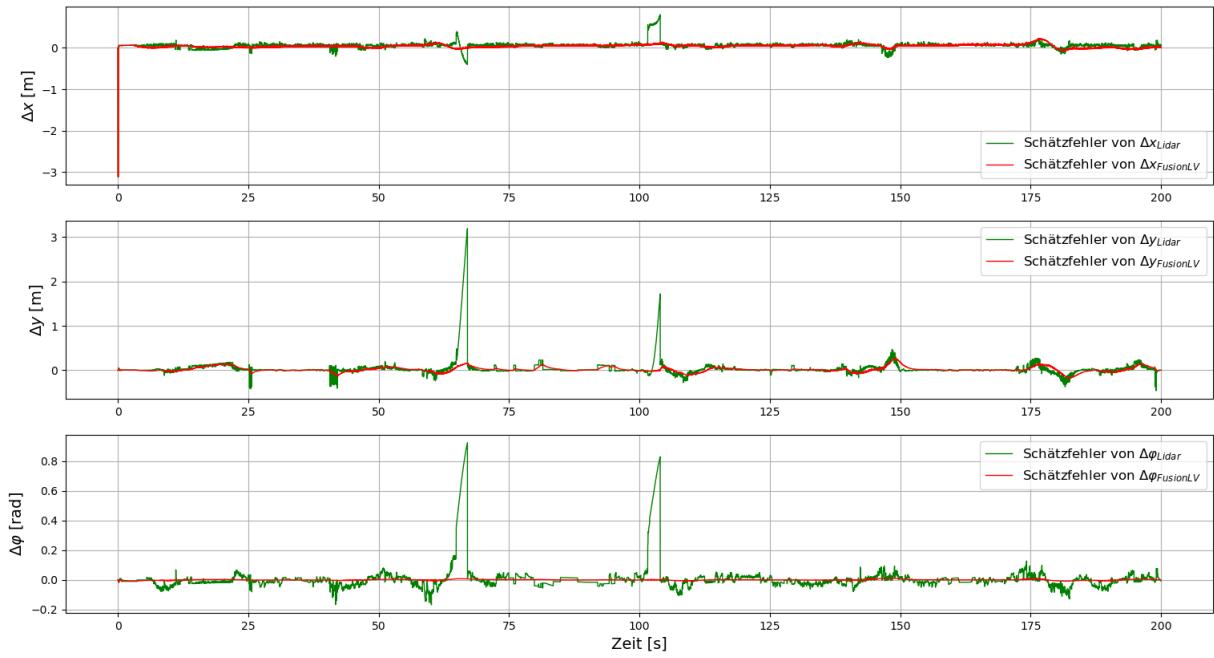


Abbildung 5.15: Verläufe des Schätzfehlers der relativen Pose im Testfall „LV“

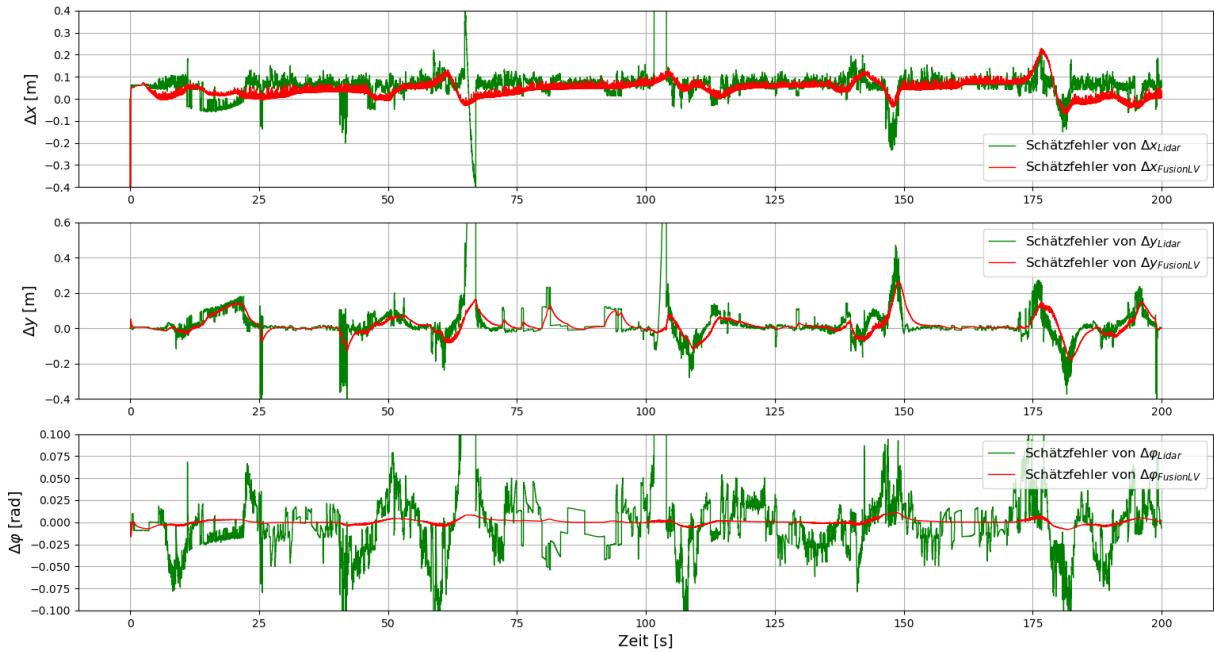


Abbildung 5.16: Verläufe des Schätzfehlers der relativen Pose im Testfall „LV“ (vertikal vergrößert)

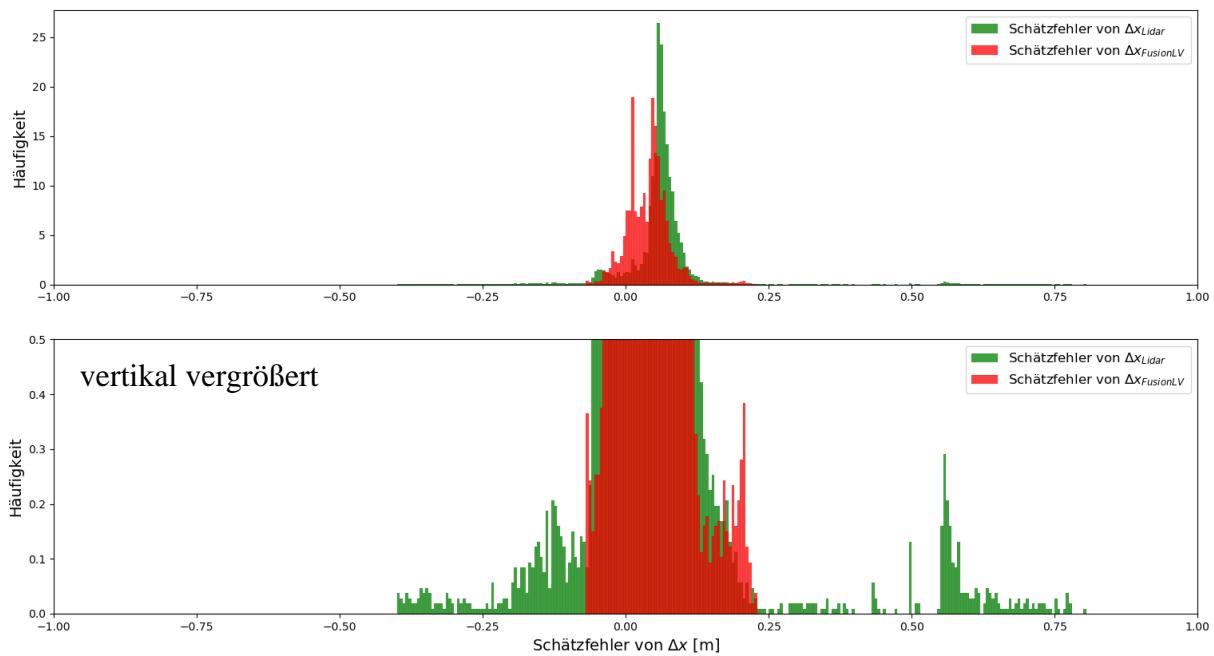


Abbildung 5.17: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „LV“

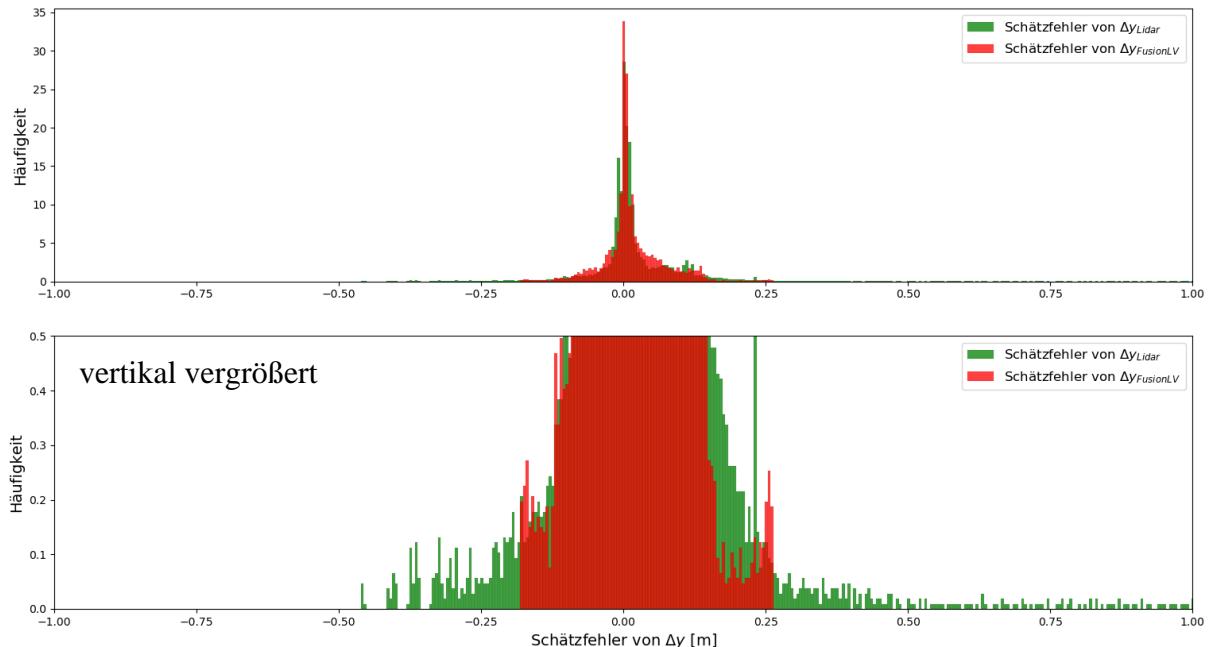


Abbildung 5.18: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „LV“

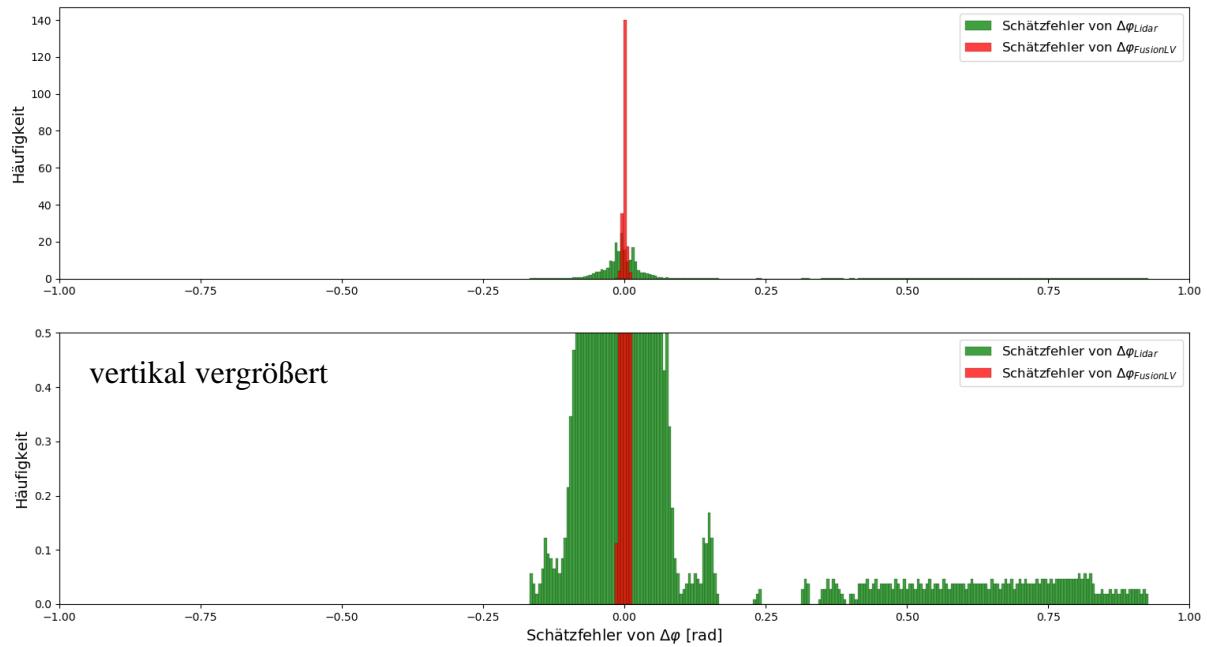


Abbildung 5.19: Häufigkeitsverteilung des Schätzfehlers von $\Delta\varphi$ im Testfall „LV“

5.3.3.3 Testfall 3: KV

Der dritte Testfall „KV“ ist sehr ähnlich wie den zweiten Testfall „LV“. Dabei werden nicht der Kamerasensor, sondern zwei Lidarsensoren ausgeschaltet, und die V2V-Kommunikation zweier Busse ist auch noch angeschaltet. Gleichfalls werden hier die Verläufe des Schätzfehlers der relativen Pose für den Testfall „KV“ in Abbildung 5.20 und Abbildung 5.21 dargestellt. Die Häufigkeitsverteilungen für den Schätzfehler von Δx , Δy und $\Delta\varphi$ sind in Abbildung 5.22, Abbildung 5.23 und Abbildung 5.24 dargestellt. Das Ergebnis zeigt auch deutlich, dass die Schätzung der relativen Pose vom Fusionsalgorithmus „KV“ schon genauer als die Schätzung des Kamera-Trackers ist, obwohl der Schätzer der relativen Pose im diesem Testfall auf die Beobachtungen der Lidarsensoren verzichtet.

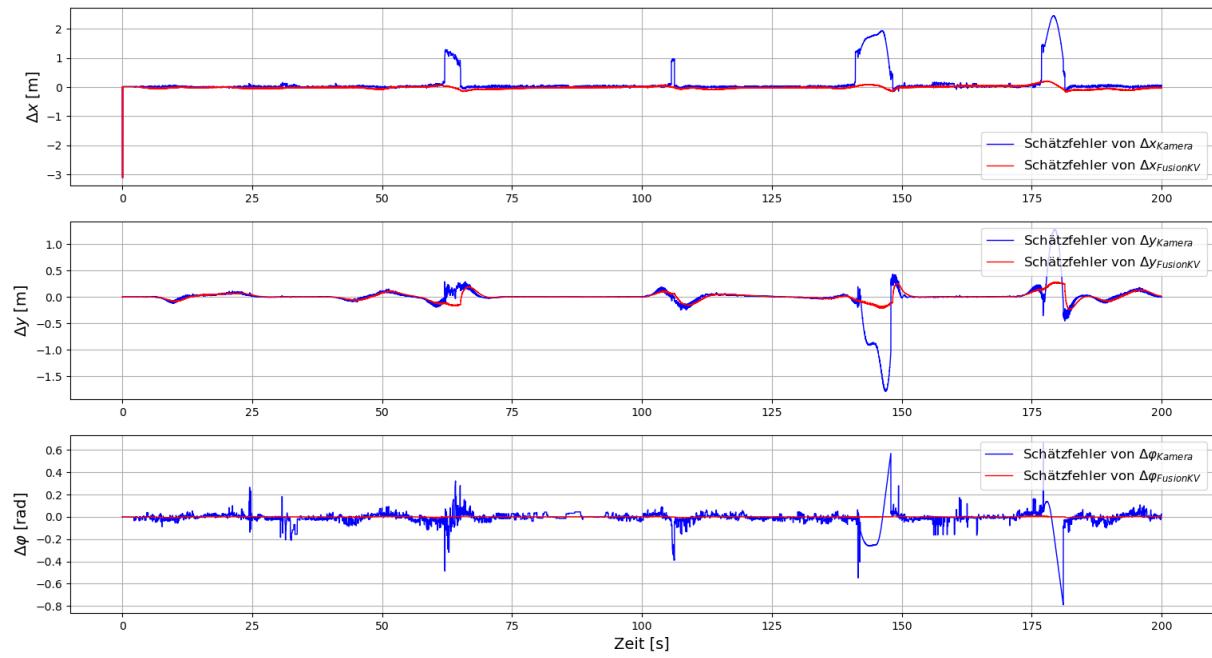


Abbildung 5.20: Verläufe des Schätzfehlers der relativen Pose im Testfall „KV“

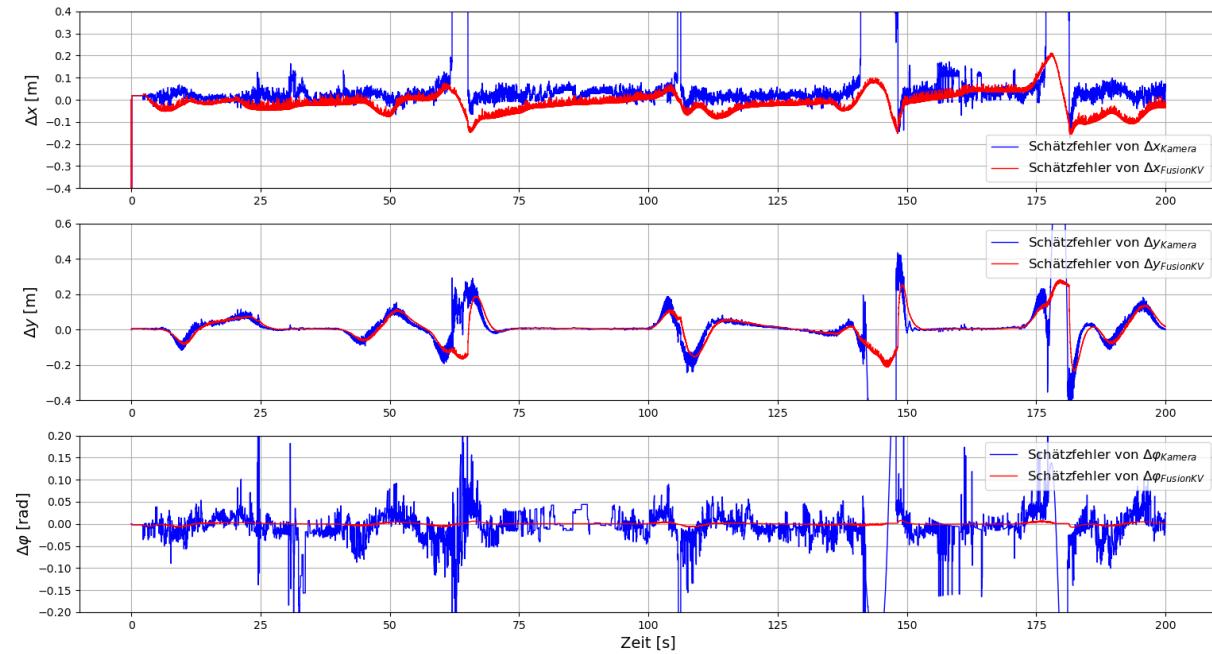


Abbildung 5.21: Verläufe des Schätzfehlers der relativen Pose im Testfall „KV“ (vertikal vergrößert)

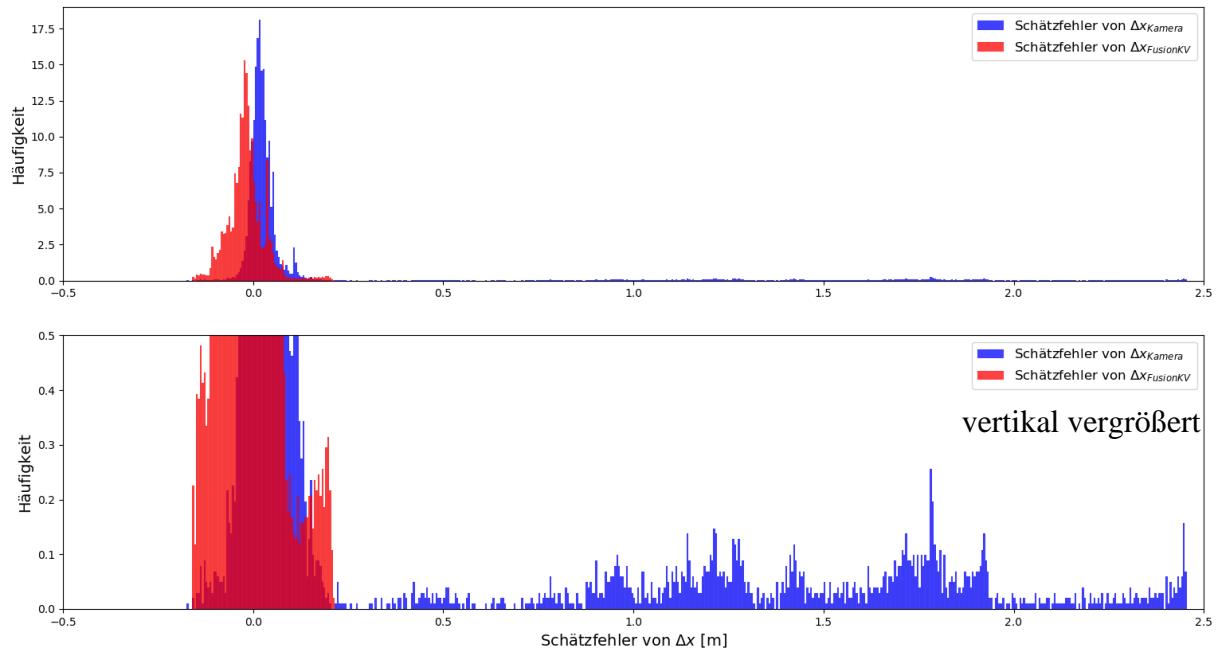


Abbildung 5.22: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „KV“

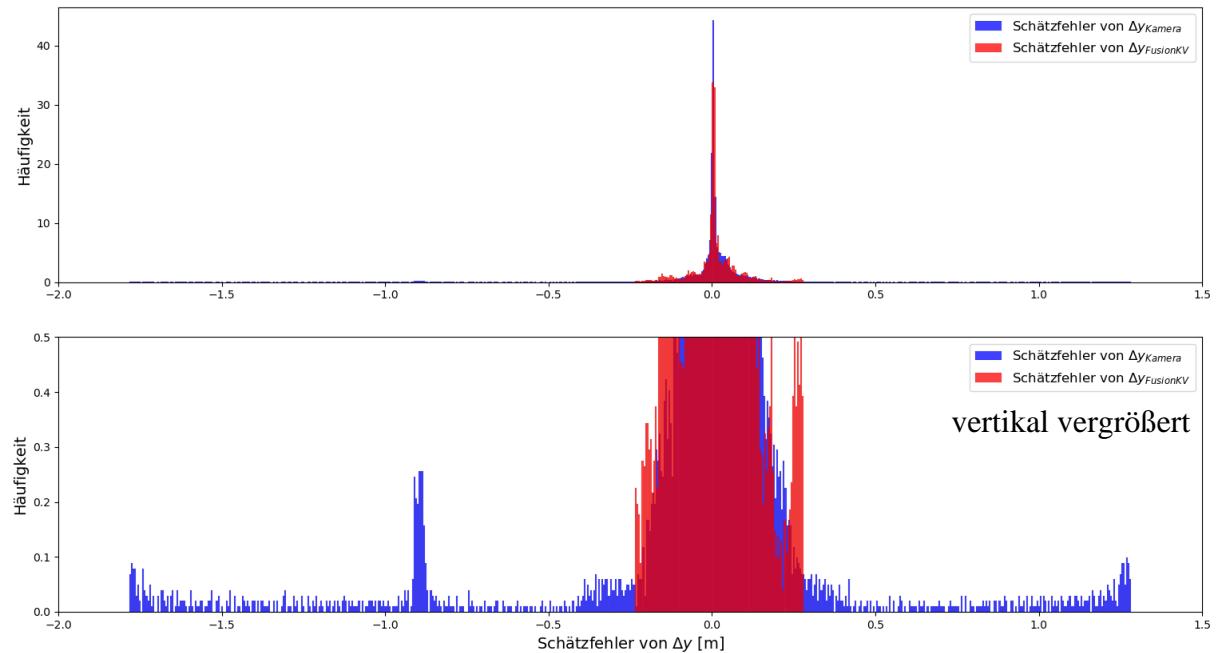


Abbildung 5.23: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „KV“

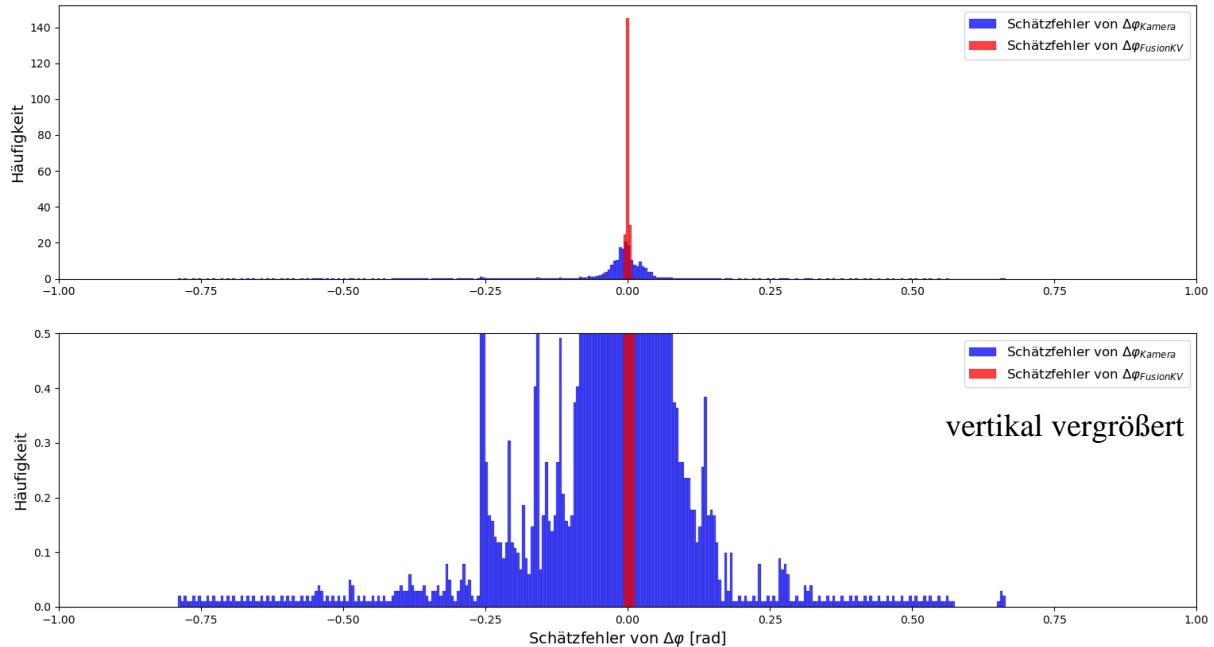


Abbildung 5.24: Häufigkeitsverteilung des Schätzfehlers von $\Delta\phi$ im Testfall „KV“

5.3.3.4 Testfall 4: LK

Im vierten Testfall „LK“ wird es geprüft, ob der Schätzer die relativen Pose auch noch richtig schätzen kann, wenn die V2V-Kommunikation zwischen zwei Bussen ausgeschaltet ist. Dieser Testfall kann in der Praxis sehr häufig auftreten, weil das neue Einschalten oder Ausschalten der V2V-Kommunikation zweier Busse beim Platooning häufig passieren kann. Es kann sein, dass die Lidarsensoren oder die Kamerapersensoren den führenden Bus bereits detektiert haben, während die Kommunikation zweier Busse noch nicht aufgebaut ist. Es ist auch möglich, dass die V2V-Kommunikation zweier Busse beim Platooning plötzlich abgebrochen wird. Wenn die V2V-Kommunikation zweier Busse nicht angeschaltet ist, kommen keine neuen Beobachtungen für die fahrdynamischen Kenngrößen des führenden Busses in den Schätzer rein. In diesem Fall können die fahrdynamischen Kenngrößen des führenden Busses jedoch noch als die Zustände durch die neuen Beobachtungen der relativen Pose vom Kamera- oder Lidar-Tracker weiter geschätzt werden.

Ebenfalls in Abbildung 5.25 und Abbildung 5.26 dargestellt sind die Verläufe des Schätzfehlers der relativen Pose für den Testfall „LK“. Man kann sehen, dass der Schätzfehler der relativen Pose vom Fusionsalgorithmus bei scharfer Abbiegung auch noch in einem kleinen Wertebereich bleibt, solange mindestens ein Umfeldsensor den führenden Bus detektieren kann. Zur Verdeutlichung wird hier ein Beispiel gegeben (siehe den gelben Bereich in Abbildung 5.25). Es ist zu sehen, dass die Lidarsensoren zuerst gescheitert sind, den führenden Bus zu detektieren. Über eine kurze Weile kann die Kamera den führenden Bus auch nicht detektieren, während die Lidarsensoren nun den führenden Bus wieder detektieren können. Mit Hilfe der komplementären Fusion der Kamera- und Lidarsensoren können das Blickfeld und die Reichweite des einzelnen Umfeldsensors schon erweitert. Ist ein

Umfeldsensor gescheitert, dem führenden Bus zu folgen, stehen noch die Daten anderer Umfeldsensoren für die Positionierung zur Verfügung.

Aber es kann nicht garantiert sein, dass immer mindestens ein Umfeldsensor den führenden Bus detektieren kann. Sind die Kamera- und Lidarsensoren gleichzeitig gescheitert, den führenden Bus zu detektieren, bekommt der Schätzer gar keine neuen Beobachtungen der relativen Pose. In diesem Fall ist eine richtige Schätzung der relativen Pose für den Schätzer unmöglich. In Abbildung 5.25 tritt dieser Fall jedoch nicht auf.

Außerdem ist in Abbildung 5.25 und Abbildung 5.26 zu erkennen, dass die Filterwirkung vom Fusionsalgorithmus in diesem Testfall deutlich schwächer ist. Das Beobachtungsrauchen kann nicht so gut, wie die erste drei Testfälle dargestellt, vom Filter ausgefiltert werden, weil sich der Schätzer der relativen Pose viel auf die Beobachtungen der relativen Pose vom Kamera- und Lidar-Tracker, und nur wenig auf die Prädiktion des Prozessmodells verlässt, wenn die V2V-Kommunikation zweier Busse abgebrochen ist.

Gleichfalls sind die Häufigkeitsverteilungen für den Schätzfehler von Δx , Δy und $\Delta \varphi$ in Abbildung 5.27, Abbildung 5.28 und Abbildung 5.29 dargestellt, um die Wahrscheinlichkeitsverteilung des Schätzfehlers zu vergleichen. Es ist zu sehen, dass die Schätzung vom Fusionsalgorithmus „LK“ weniger Ausreißer enthält, und schon genauer als die Schätzung der Kamera- und Lidarsensoren ist.

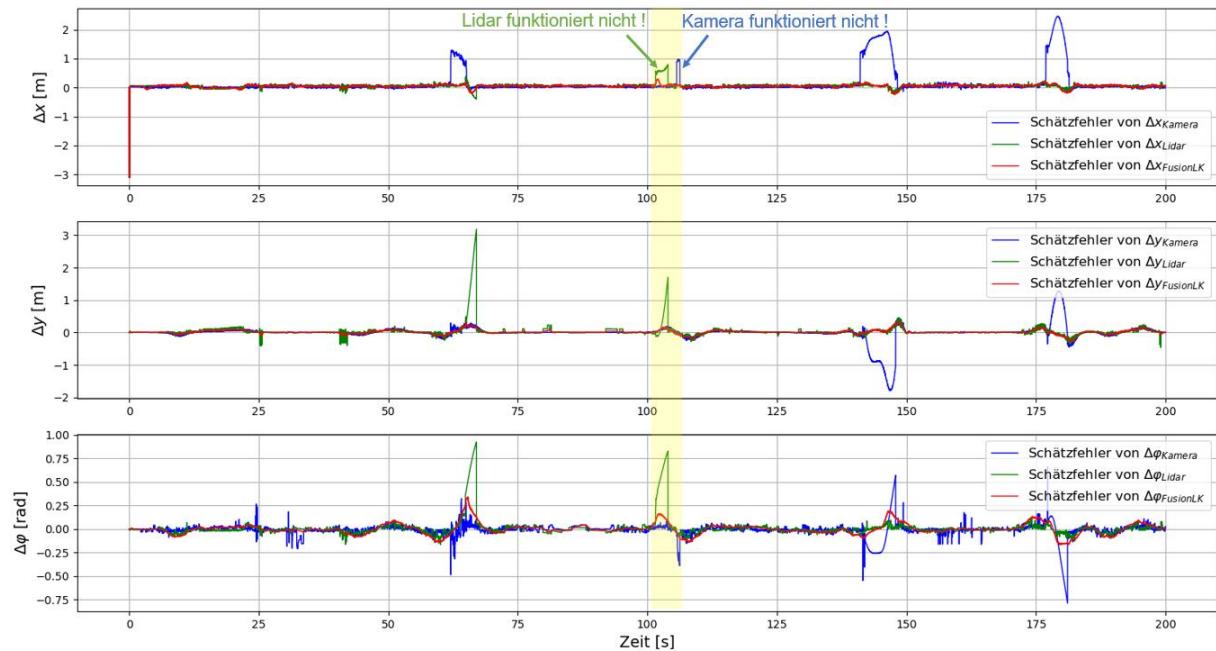


Abbildung 5.25: Verläufe des Schätzfehlers der relativen Pose im Testfall „LK“

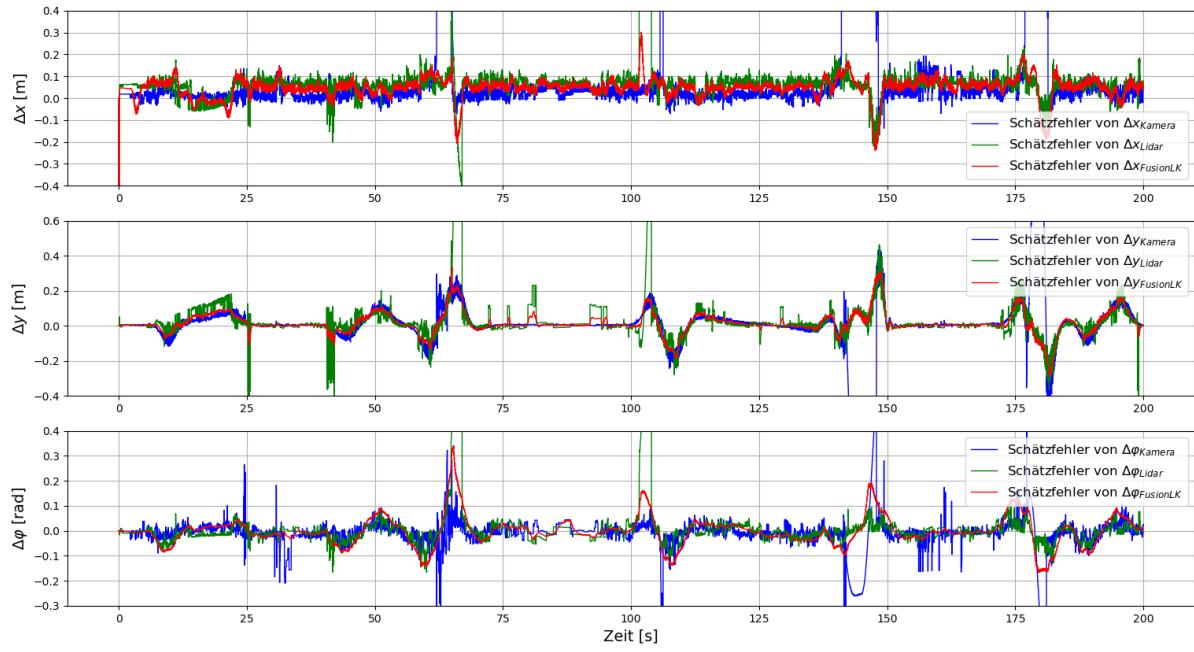


Abbildung 5.26: Verläufe des Schätzfehlers der relativen Pose im Testfall „LK“ (vertikal vergrößert)

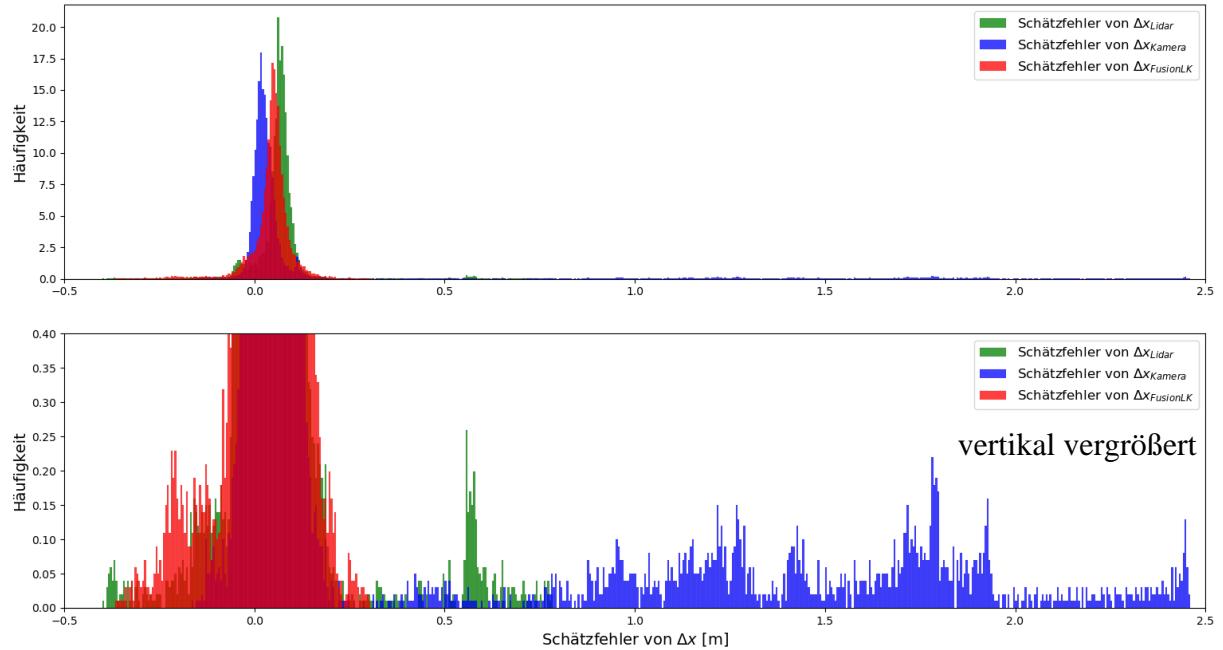


Abbildung 5.27: Häufigkeitsverteilung des Schätzfehlers von Δx im Testfall „LK“

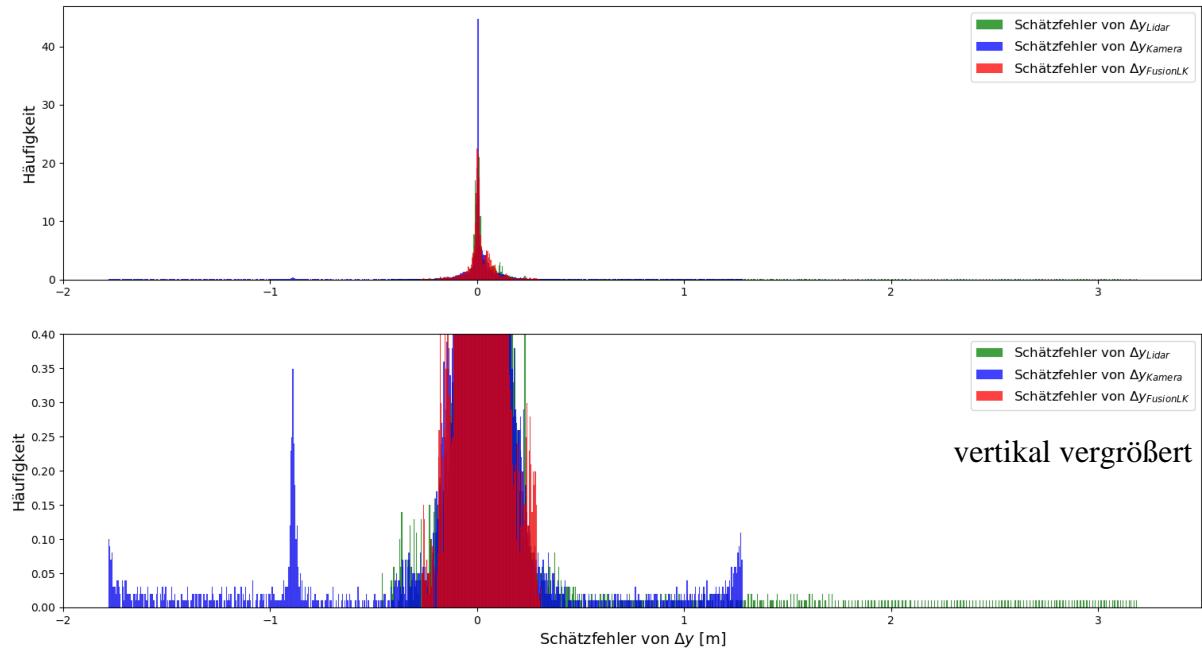


Abbildung 5.28: Häufigkeitsverteilung des Schätzfehlers von Δy im Testfall „LK“

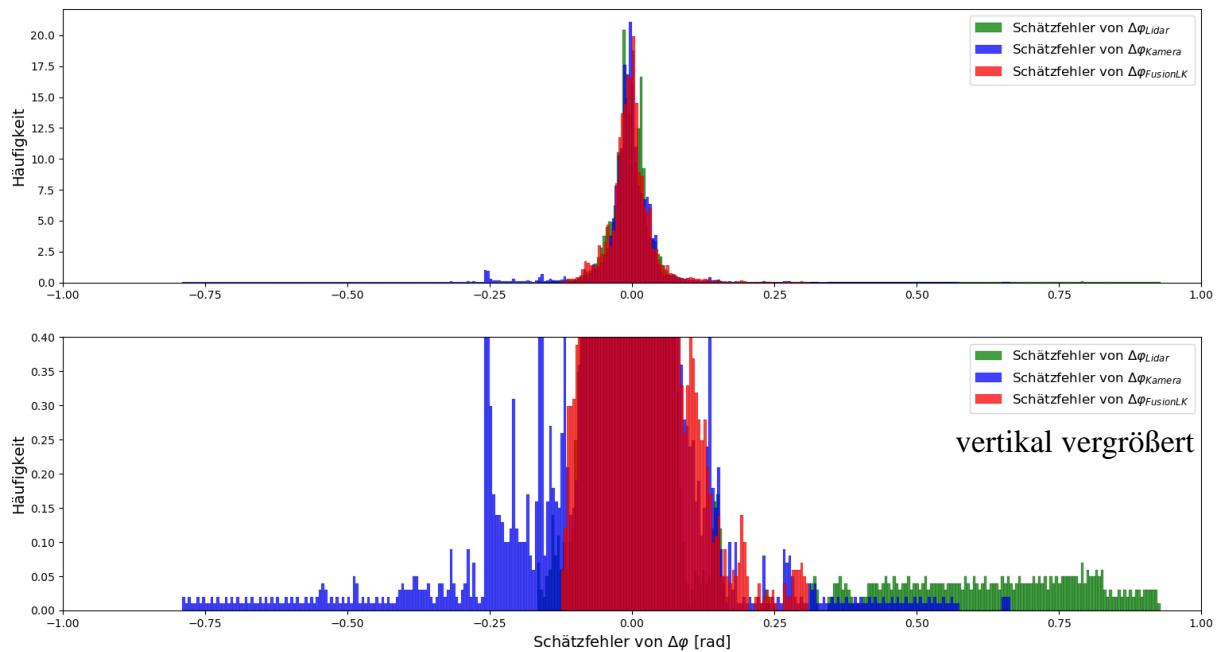


Abbildung 5.29: Häufigkeitsverteilung des Schätzfehlers von $\Delta \varphi$ im Testfall „LK“

5.3.3.5 Vergleich der Wahrscheinlichkeit der Ausreißer

Für die Schätzwerte der relativen Pose kann man Ausreißer, die von den übrigen Werte der Schätzwerte entfernt liegt, definieren. Solche Ausreißer der Schätzwerte der relativen Pose sind nicht erwünscht, weil sie die weitere Trajektorienplanung und Fahrzeugsteuerung maßgeblich verschlechtern können. In diesem Abschnitt werden die Wahrscheinlichkeiten der Ausreißer der geschätzten relativen Pose, die in ersten vier Testfälle untersucht werden, zusammen dargestellt und verglichen.

Für die Schätzung des longitudinalen Abstands Δx kann man zwei Arten der Ausreißer definieren. Hier sind in Abbildung 5.30 die Wahrscheinlichkeiten der Ausreißer, deren Schätzfehler von Δx größer als 0,2m und größer als 0,5m, jeweils durch blaue und orange Balken dargestellt. Gleichfalls zeigt Abbildung 5.31 die Wahrscheinlichkeiten der Ausreißer des lateralen Versatzes Δy , wenn der Schätzfehler von Δy größer als 0,2m und größer als 0,5m ist. Abbildung 5.32 zeigt dabei die zugehörigen Wahrscheinlichkeiten der Ausreißer von $\Delta\varphi$, wenn der Schätzfehler von $\Delta\varphi$ größer als 0,1rad und 0,2rad ist.

Es ist deutlich zu sehen, dass die Wahrscheinlichkeiten der Ausreißer der relativen Pose in Testfälle „LKV“, „LV“, „KV“ und „LK“ schon viel niedriger als die Wahrscheinlichkeiten der Ausreißer von „L“ und „K“, wobei lediglich die Lidarsensoren und der Kamerasonar verwendet werden. Im Vergleich zu Testfälle „LKV“, „LV“, „KV“ enthält die Schätzung der relativen Pose im Testfall „LK“ viel mehr Ausreißer, weil sich die Filterwirkung aufgrund des Ausfalls der V2V-Kommunikation verschlechtert. Der Testfall „LKV“ zeigt im Allgemeinen eine beste Performance, welche nur wenige Ausreißer darstellt.

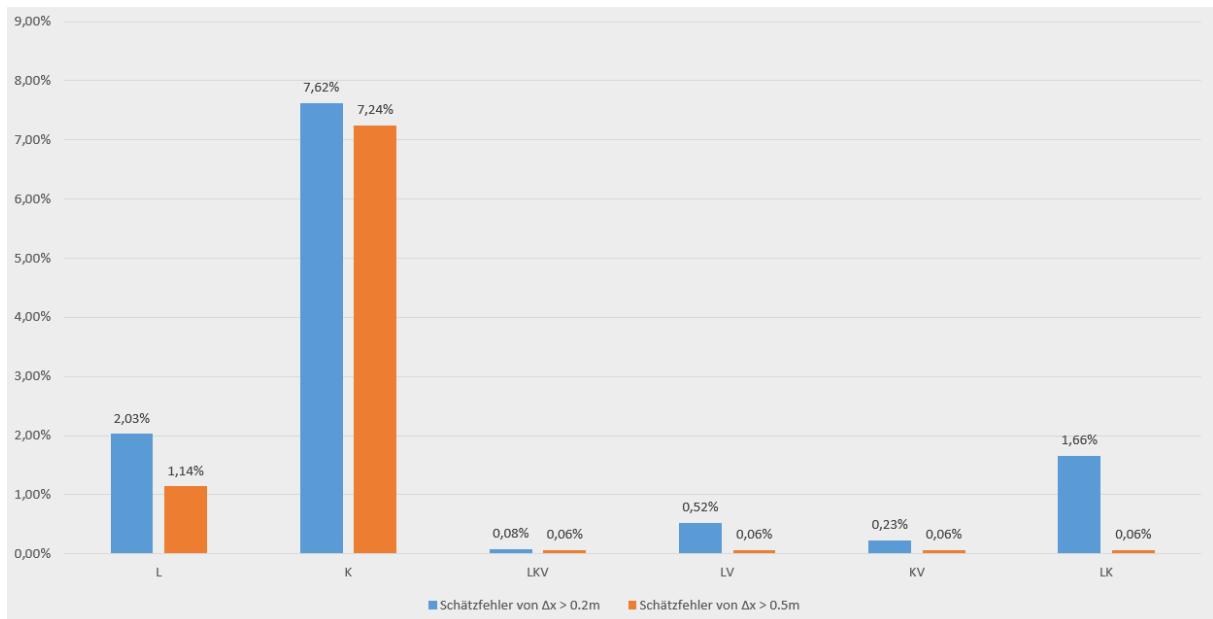


Abbildung 5.30: Wahrscheinlichkeit der Ausreißer der Schätzwerte Δx

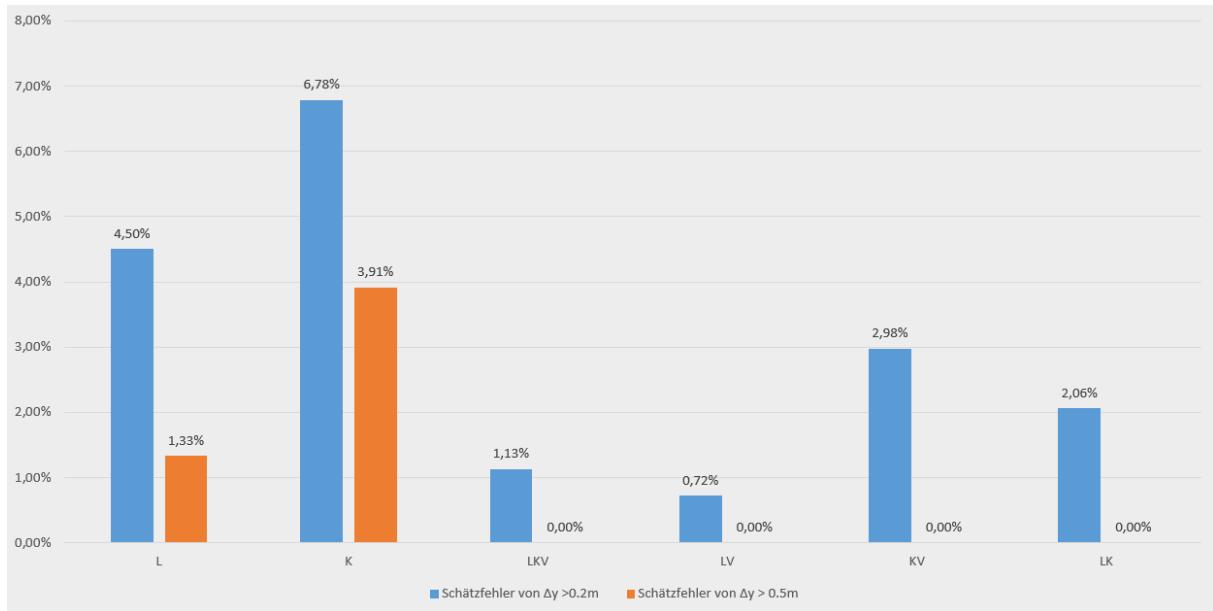


Abbildung 5.31: Wahrscheinlichkeit der Ausreißer der Schätzwerte Δy

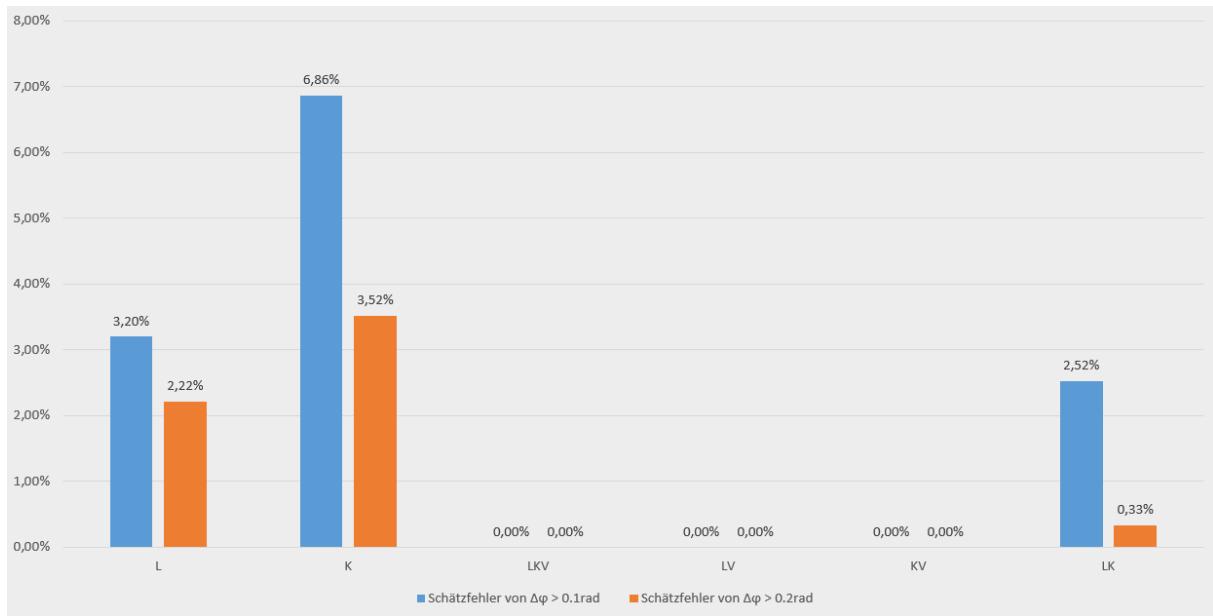


Abbildung 5.32: Wahrscheinlichkeit der Ausreißer der Schätzwerte $\Delta\varphi$

5.3.3.6 Testfall 5: V

Im fünften Testfall sind zwei Lidarsensoren, der Kamerasensor sowie die V2V-Kommunikation zunächst angeschaltet (siehe den Zeitbereich $t = 0 \sim 25s$ mit Bezeichnung „LK“ in Abbildung 5.33), damit die Initialisierung der geschätzten Zustände $[\Delta x \ \Delta y \ \Delta\varphi]^T$ beim UKF erfolgen kann (siehe Abschnitt 5.2.3). Aber über einer Weile (ab $t = 25s$) sind zwei Lidarsensoren und der Kamerasensor gleichzeitig ausgeschaltet, jedoch ist die V2V-Kommunikation zweier Busse noch verbunden (siehe den Zeitbereich mit

Bezeichnung „V“ in Abbildung 5.33). Im diesem Fall können die Zustände für die relative Pose nicht mehr durch die Beobachtungen vom Kamera- und Lidar-Tracker korrigiert werden, sondern lediglich durch den Prädiktionsschritt vom UKF geschätzt werden. Da keine neuen Beobachtungen der relativen Pose beim Schätzer vorliegen, kann sich die Unsicherheit der Schätzung ab $t = 25\text{s}$ immer steigern. Ab $t = 190\text{s}$ werden zwei Lidarsensoren sowie der Kamerasensor wieder eingeschaltet, damit der Schätzfehler der relativen Pose schnell reduziert wird (siehe den Zeitbereich $t = 190\text{--}200\text{s}$ mit Bezeichnung „LKV“).

Durch Abbildung 5.33 wird es nachgewiesen, dass nach dem Ausschalten der Kamera- und Lidarsensoren die relativen Pose zweier Busse mithilfe der V2V-Kommunikation noch weiter geschätzt werden kann, obwohl der Schätzfehler immer höher wird.

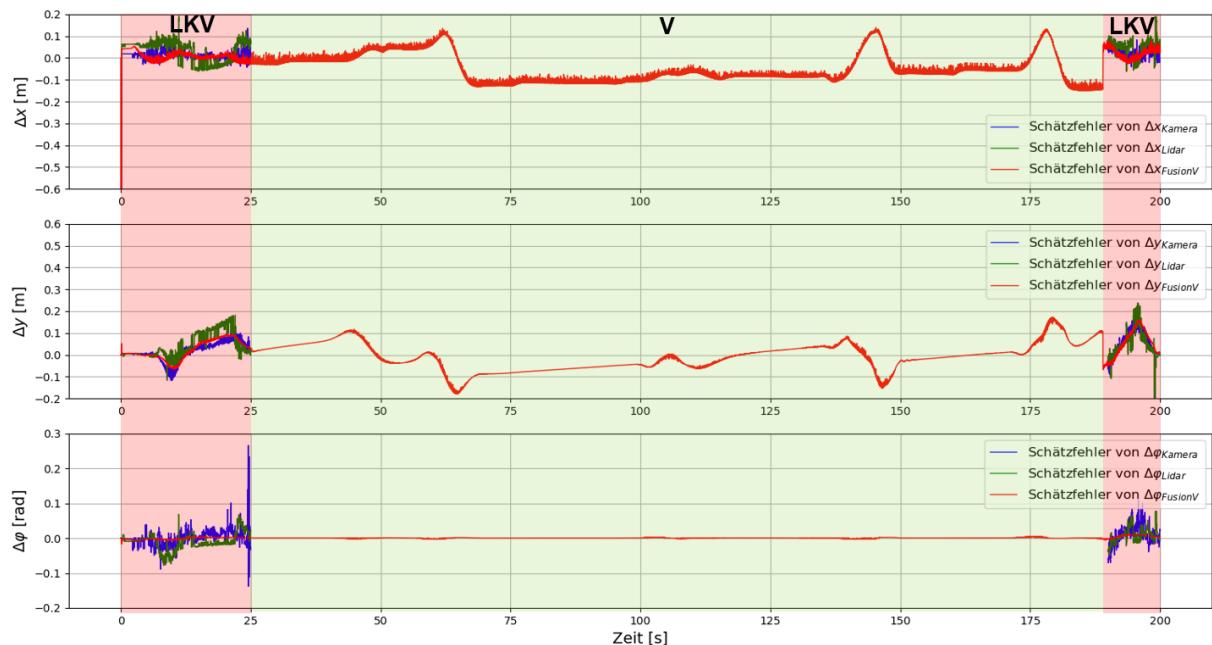


Abbildung 5.33: Verläufe des Schätzfehlers der relativen Pose im Testfall „V“

5.3.3.7 Testfall 6: Mode Switch

Im letzten Testfall „Mode Switch“ wird der Betriebszustand des Schätzers frequent gewechselt, indem zwei Lidarsensoren, der Kamerasensor sowie die V2V-Kommunikation nacheinander frequent ein- und ausgeschaltet werden. Abbildung 5.34 zeigt dabei die Verläufe des Schätzfehlers der relativen Pose, wobei die roten Linien den Schätzfehler des Schätzers bezeichnet, wenn der Schätzer in verschiedenen Betriebszuständen arbeitet. In Abbildung 5.35 wird die Ansicht für die Verläufe des Schätzfehlers vertikal vergrößert.

Auffällig in der Abbildung 5.34 sind die großen Schätzfehlerspitzen im Zeitbereich mit der Bezeichnung „L“, wobei der Schätzer lediglich die Beobachtungen von Lidar verwendet. Wenn der Lidar-Tracker bei scharfer Abbiegung dem führenden Bus nicht folgen kann, ist die Schätzung der relativen Pose vom Schätzer auch gescheitert. Ein gleiches Problem tritt auch beim Betriebszustand „K“ auf, wenn nur die Beobachtungen der Kamera zur Schätzung der relativen Pose benutzt werden. Bei Betriebszuständen „LKV“, „LV“, „KV“, „LK“ und „V“ können der Schätzfehler immer in einem kleinen Wertebereich bleiben.

Durch diesem Testfall kann es nachgewiesen werden, dass dieses Fusionskonzept eine gute Robustheit darstellt, wenn irgend ein Umfeldsensor ausfällt. Wenn die V2V-Kommunikation zweier Busse ausfällt, kann die relative Pose mit Hilfe der Beobachtungen von Lidar und Kamera noch geschätzt werden. Jedoch kann es nicht garantiert sein, dass die Schätzung der relativen Pose bei scharfer Abbiegung noch erfolgen kann, wenn die Kamera- und die Lidarsensoren gleichzeitig den führenden Bus nicht detektieren können. Außerdem ist die Filterwirkung der Schätzwerte viel besser, wenn eine V2V-Kommunikation zwischen Bussen angeschaltet ist.

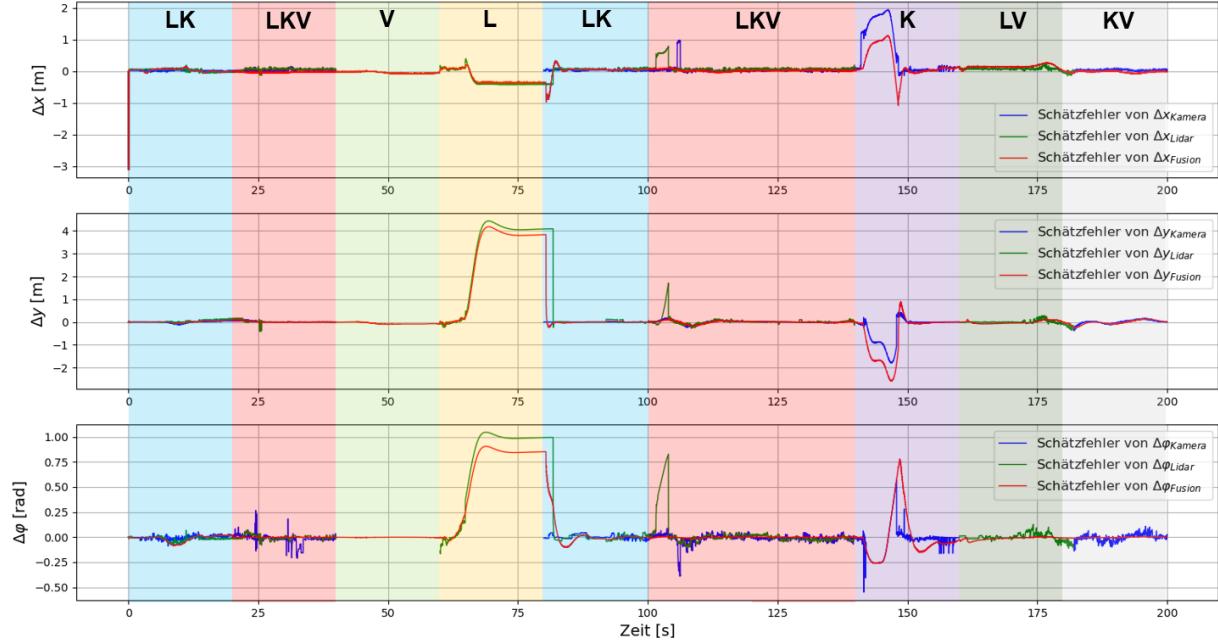


Abbildung 5.34: Verläufe des Schätzfehlers der relativen Pose im Testfall „Mode Switch“

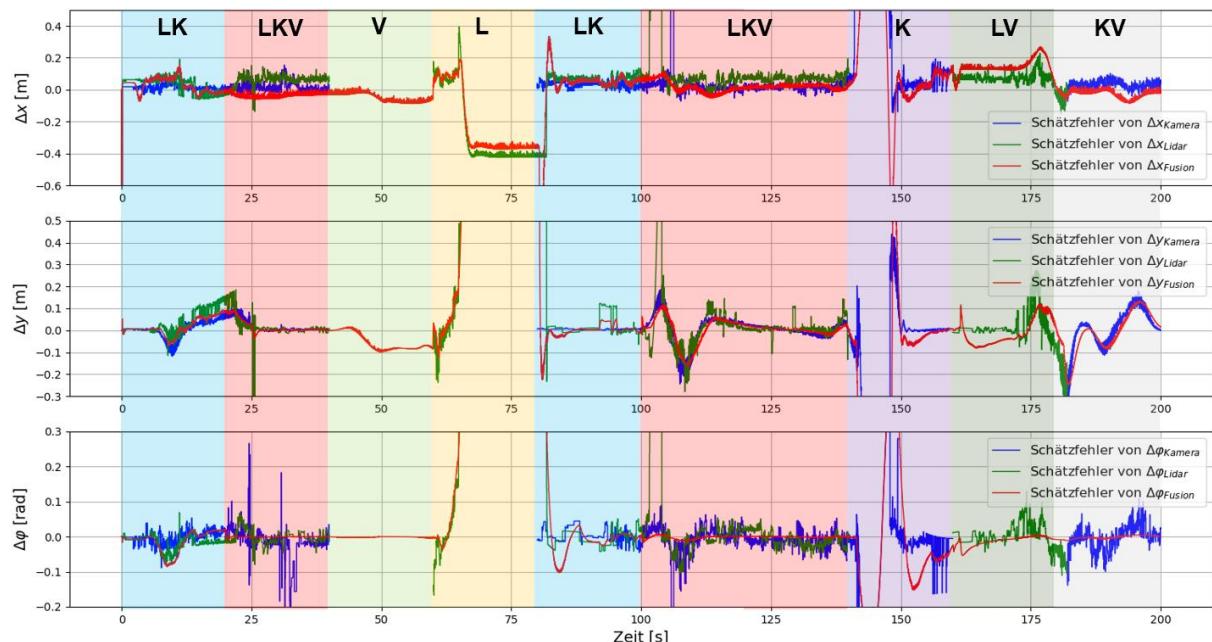


Abbildung 5.35: Verläufe des Schätzfehlers der relativen Pose im Testfall „Mode Switch“ (vertikal vergrößert)

6 Fazit

6.1 Zusammenfassung

In dieser vorliegenden Arbeit wurde zur besseren Schätzung der relativen Pose zweier Busse ein Informationsfusionskonzept entworfen, welches die Beobachtungen der Kamera- und Lidarsensoren sowie die Messdaten der internen Sensoren zweier Busse ausnutzt. Die Messdaten der internen Sensoren des führenden Busses wird durch die V2V-Kommunikation in den verfolgenden Bus übertragen. Alle Messdaten werden beim Prozessor des verfolgenden Busses mit Hilfe des Unscented Kalman Filters zentralisiert fusioniert.

Zur Erprobung des entwickelten Konzepts wird die Simulationsumgebung „CarMaker 11.0“ verwendet, in der sowohl die Busse und die Sensoren, als auch das Testszenario und das Fahrmanöver simuliert werden. Aus den Ergebnissen in der Simulation lässt sich entnehmen, dass das entwickelte Informationsfusionskonzept eine genauere Schätzung der relativen Pose zweier Busse erzielt hat. Wenn die Kamera- oder die Lidarsensoren bei scharfer Abbiegung aufgrund des begrenzten Blickfelds und Reichweite den führenden Bus nicht detektieren, kann das Informationsfusionskonzept jedoch noch eine gute Schätzung der relativen Pose anbieten. Außerdem zeigt das Informationsfusionskonzept eine gute Robustheit, weil der Schätzer noch sicher arbeiten kann, wenn ein Umfeldsensor (Lidar oder Kamera) ausfällt, oder die V2V-Kommunikation zwischen Bussen abgebrochen ist.

6.2 Ausblick

Die Erprobung in der Simulation zeigt, dass die Auslegung der Parameter des Kalman Filters die Performance des Schätzers maßgeblich beeinflussen kann. Wenn der entwickelte Fusionsalgorithmus am Teststand implementiert wird, müssen dabei die Kovarianzen des Beobachtungsrauschens der realen Messsignale durch das gleiche Verfahren, wie in Abschnitt 5.2.1 vorgestellt, bestimmt werden. Die Kovarianzen des Prozessrauschens kann als „Tuning Parameter“ eingestellt werden, bis eine gute Filterwirkung wie in Abschnitt 5.2.2 dargestellt erzielt wird.

In dieser Arbeit werden die Kovarianzen des Beobachtungsrauschens und des Prozessrauschens auf Konstante gesetzt. In der Realität können diese Kovarianzen sehr unterschiedliche Größenordnungen annehmen, wenn sich der Betriebszustand der Sensoren ändert, z.B. die Qualität des Messsignals des Radencoders kann sich mit steigender Drehzahl verbessern [03]. Für eine bessere Informationsfusion müssen dabei genauere Rauschmodelle für die realen Messsignale bestimmt werden.

7 Literaturverzeichnis

- [01] Zeitung für kommunale Wirtschaft (ZfK), „Platooning für Münchner Busse“, 04. 06. 2021. [Online]. Available: <https://www.zfk.de/mobilitaet/oepnv/platooning-fuer-muenchner-busse>. [Zugriff am 25. 08. 2022].
- [02] J. Ignatowitsch, „Bus-Platooning, Nachhaltiger ÖPNV: Busse, die automatisch Kolonne fahren“, Deutschlandfunk Nova, 07. 07. 2021. [Online]. Available: <https://www.deutschlandfunknova.de/beitrag/nachhaltiger-oepnv-busse-die-automatisch-in-kolonne-fahren>. [Zugriff am 25. 08. 2022].
- [03] U. Kiencke, M. Schwarz und T. Weickert, Signalverarbeitung, Zeit-Frequenz-Analyse und Schätzverfahren, München: Oldenbourg Wissenschaftsverlag GmbH, 1. Auflage, 2008.
- [04] E. A. Wan und R. v. d. Merwe, „The Unscented Kalman Filter for Nonlinear Estimation“, in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications and Control Symposium (Cat. No.00EX373)*, Oregon Graduate, 2000.
- [05] T. D. Barfoot, State Estimation in Robotics, Cambridge University Press, 1st Edition, 2017.
- [06] S. Thrun, W. Burgard und D. Fox, Probabilistic Robotics, Massachusetts Institute of Technology: The MIT Press, 1st Edition, 2006.
- [07] P. Zarchan und H. Musoff, Fundamentals of Kalman Filtering: A Practical Approach, American Institute of Aeronautics and Astronautics, 3rd Edition, 2009.
- [08] H.-R. Tränkler und L. M. Reindl, Sensortechnik, Handbuch für Praxis und Wissenschaft, Springer DE, 2. Auflage, 2014.
- [09] F. P. León, Messtechnik, Grundlagen, Methoden und Anwendungen, Karlsruher Institut für Technologie: Springer Vieweg, 11. Auflage, 2019.
- [10] InvenSense - TDK Corporation, „MPU-6050 Six-Axis (Gyro + Accelerometer) MEMS MotionTracking Devices“, [Online]. Available: <https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/>. [Zugriff am 26. 08. 2022].
- [11] K. Reif, Sensoren im Kraftfahrzeug, Bosch Fachinformation Automobil, 3. Auflage, 2017.
- [12] S. Breuer und A. Rohrbach-Ker, Fahrzeughodynamik, Mechanik des bewegten Fahrzeugs, Bochum, Deutschland: Springer Fachmedien Wiesbaden, 1. Auflage, 2015.
- [13] L. Valdy, „Entwurf und Implementierung eines Bildverarbeitungs-Ansatzes zur relativen Positionserkennung zweier Fahrzeuge in Kontext des urbanen Platoonings unter der Nutzung von Marker“, Bachelorarbeit, Institut für Technik der Informationsverarbeitung, Karlsruher Institut für Technologie, 2021.

- [14] D. Dimitrov, „Relative Positionsbestimmung von Bussen mittels Kamera für das urbane Platooning“, Bachelorarbeit, Institut für Technik der Informationsverarbeitung, Karlsruher Institut für Technologie, 2022.
- [15] P.-L. Müller, „Detektion und Tracking des vorausfahrenden Fahrzeugs für Platooning“, Masterarbeit, Institut für Technik der Informationsverarbeitung, Karlsruher Institut für Technologie, 2022.
- [16] P. Palendran, „Entwurf und Evaluation eines Sensorkonzepts zur Umsetzung einer Platooning-Funktionalität mittels Simulation“, Masterarbeit, Institut für Technik der Informationsverarbeitung, Karlsruher Institut für Technologie, 2021.
- [17] C. Han, M. Frey und F. Gauterin, „Modular Approach for Odometry Localization Method for Vehicles with Increased Maneuverability“, in *Sensors 2021*, 21, 79., Institut für Fahrzeugsystemtechnik, Karlsruher Institut für Technologie, 2021.
- [18] M. Klomp, G. Yunlong und F. Bruzelius, „Longitudinal velocity and road slope estimation in hybrid electric vehicles employing early detection of excessive wheel slip“, in *Vehicle System Dynamics*, 52(SUPPL. 1): 172-188, e-AAM Driveline Systems, Trollhättan, Sweden, 2014.
- [19] S.-H. You, J.-O. Hahn und H. Lee, „New adaptive approaches to realtime estimation of vehicle sideslip angle“, in *Control Engineering Practice 17 (2009) 1367–1379*, Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, 2009.
- [20] J. Liu, Z. Wang, L. Zhang und P. Walker, „Sideslip angle estimation of ground vehicles: a comparative study“, in *IET Control Theory Appl.*, 2020, Vol. 14 Iss. 20, pp. 3490-3505, National Engineering Laboratory for Electric Vehicles, Beijing Institute of Technology, China; Faculty of Engineering and Information Technology, University of Technology Sydney, Australia, 2020.
- [21] F. d. P. Müller, E. M. Diaz und I. Rashdan, „Cooperative Positioning and Radar Sensor Fusion for Relative Localization of Vehicles“, in *2016 IEEE Intelligent Vehicles Symposium (IV)*, Institute of Communications and Navigation of the German Aerospace Center (DLR), Wessling, Germany, 2016.
- [22] E. Richter, M. Obst, R. Schubert und G. Wanielik, „Cooperative Relative Localization using Vehicle-To-Vehicle Communication“, in *12th International Conference on Information Fusion*, Chemnitz University of Technology, Germany, 2009.
- [23] W. Farag, „Kalman-filter-based sensor fusion applied to road-objects detection and tracking for autonomous vehicles“, in *Journal of Systems and Control Engineering*, 2021, Vol. 235(7) 1125–1138, College of Engineering and Technology, American University of the Middle East, Kuwait, 2021.
- [24] D. Bevly, J. Ryu und J. Gerdes, „Integrating Ins sensors with Gps measurements for continuous estimation of vehicle sideslip, roll, and tire cornering stiffness“, in *IEEE Trans. Intell. Transp. Syst.*, 2006, 7, (4), pp. 483–493, 2006.

- [25] J. Chen, J. Song und L. Li, „UKF-based adaptive variable structure observer for vehicle sideslip with dynamic correction“, in *IET control Theory Appl.*, 2016, 10, (14), pp. 1641–1652, 2016.
- [26] J. Liu, Z. Wang und L. Zhang, „A time-delay neural network of sideslip angle estimation for in-wheel motor drive electric vehicles“, in *2020 IEEE 91st Vehicular Technology Conf. (VTC2020-Spring)*, Antwerp, Belgium, 2020.
- [27] M. Faizullin, A. Kornilova und G. Ferrer, „Open-Source LiDAR Time Synchronization System by Mimicking GPS-clock“, in *IEEE ISPCS 2022 Conference (International Symposium on Precision Clock Synchronization for Measurement, Control and Communication)*, Skolkovo Institute of Science and Technology, Moscow, Russia, 2022.
- [28] M. Hennes und S. Wursthorn, „Zur Synchronisierung von Multi-Sensor-Systemen – Grundlagen und Realisierungen“, Karlsruher Institut für Technologi, 2015.
- [29] IPG Automotive Group, „CarMaker, Reference Manual Version 11.0.“, Karlsruhe, 2022.
- [30] IPG Automotive Group, „CarMaker SimNet, User Manual Version 11.0.“, Karlsruhe, 2022.
- [31] C. Bergenhem; H. Pettersson; E. Coelingh; C. Englund; S. Shladover und S. Tsugawa, „Overview of Platooning Systems“, in *19th ITS World Congress*, Vienna, Austria, 2012.
- [32] S. Tsugawa; S. Jeschke und S. E. Shladovers, „A Review of Truck Platooning Projects for Energy Savings“, in *IEEE Transactions on Intelligent Vehicles*, Vol. 1, No. 1, 2016.