

Schlussbericht zum Vorhaben

GALILEOnautic 2

Autonomes Navigieren und optimiertes Manövrieren von kooperierenden Schiffen in sicherheitskritischen Bereichen

Vorgelegt vom Institut für Regelungstechnik der RWTH Aachen University (Förderkennzeichen: 50 NA 1808)



18. Januar 2022

Autoren

Kapitel 1	<i>Projektvorstellung</i>	J.-J. Gehrt, R. Zweigel
Kapitel 2	<i>Maritimes hochgenaues Navigationsfilter</i>	J.-J. Gehrt, S. Liu, M. Nitsch, D. Stenger
Kapitel 3	<i>Integres Navigationsfilter</i>	S. Liu, J.-J. Gehrt
Kapitel 4	<i>Navigation im Netzwerk und Umgebungswahrnehmung</i>	J. Lin, J.-J. Gehrt, M. Kosch
Kapitel 5	<i>Projektergebnis</i>	J.-J. Gehrt, R. Zweigel

Inhaltsverzeichnis

Autoren	iii
Abkürzungsverzeichnis	vii
1 Projektvorstellung	1
1.1 Aufgabenstellung	3
1.2 Voraussetzungen des Vorhabens	7
1.3 Konsortium	7
1.4 Planung und Ablauf des Vorhabens	9
1.5 Änderungen des Vorhabens während der Projektlaufzeit	12
1.6 Technische Ausgangsbasis	13
1.7 Zusammenarbeit mit anderen Stellen	15
2 Maritimes hochgenaues Navigationsfilter	17
2.1 Einleitung und Stand der Technik	17
2.2 Wissenschaftlich-technische Arbeiten	20
2.2.1 Verwendete Koordinatensysteme	20
2.2.2 Filterdesign	21
2.2.3 Sensordaten-Vorverarbeitung	24
2.2.4 Schätzung von differenziellen Daten	26
2.2.5 Zustandsraummodell Navigationsfilter	29
2.2.6 Initialisierung	35
2.2.7 Kompensation totzeitbehafteter Messungen	35
2.2.8 Zustandsautomat	36
2.2.9 Minaturisierung für Lotsassistentz	36
2.2.10 Parametrierung	39
2.3 Experimentelle Validierung	41
2.3.1 Optimierte Satellitenkonstellation	41
2.3.2 Trägerphasen-gestütztes Navigationsfilter	44
2.3.3 Kompass-gestütztes Navigationsfilter	50
2.3.4 Navigationsfilter auf DENEb	52
2.3.5 Navigationsfilter zur Lotsassistentz mit Galileo-GPS DGNSS	54

3	Integres Navigationsfilter	59
3.1	Einleitung und Stand der Technik	59
3.2	Wissenschaftlich-technische Arbeiten	61
3.2.1	Überblick hybrider Strategie zur GNSS-Integritätsüberwachung . .	61
3.2.2	RAIM mit Paritätsraum	62
3.2.3	RSIVIA und Integration von RAIM	63
3.2.4	Filterbank zur Fehlererkennung integrierter Sensoren	68
3.2.5	Gesamtintegrität der Navigationslösung	71
3.3	Experimentelle Validierung	73
3.3.1	GNSS-Fehlererkennung und Ausschluss	73
3.3.2	Sensorübergreifende Fehlererkennung mittels Filterbank	77
3.3.3	Protection Level Navigationslösung	83
4	Navigation im Netzwerk und Umgebungswahrnehmung	85
4.1	Einleitung und Stand der Technik	85
4.1.1	Einleitung und Motivation	85
4.2	Wissenschaftlich-technische Arbeiten	87
4.2.1	Überblick	87
4.2.2	Objekterkennung	88
4.2.3	Multiple Objekt Tracking	90
4.2.4	Statische Kartierung	97
4.2.5	Kooperative Lokalisation	102
4.2.6	Zeitsynchronisation	103
4.3	Experimentelle Validierung	104
4.3.1	Navigation im Netzwerk und Umgebungswahrnehmung mit MESSIN	104
4.3.2	Umgebungswahrnehmung mit DENEb	111
5	Projektergebnis	117
5.1	Bewertung der durchgeführten Arbeiten	117
5.1.1	Notwendigkeit und Angemessenheit	117
5.1.2	Voraussichtlicher Nutzen	118
5.2	Externe Fortschritte	119
5.3	Zusammenfassung	121
5.4	Eigene Veröffentlichungen im Projekt	123

Abkürzungsverzeichnis

AHRS	Attitude & Heading Reference System
AIS	Automatic Identification System
AP	Arbeitspaket
ATC	Aldenhoven Testing Center
AZR	Anwendungszentrum für Regelungstechnik
BMWi	Bundesministerium für Wirtschaft und Energie
BRZ	Bruttoraumzahl
BSH	Bundesamt für Seeschifffahrt und Hydrographie
CENELEC	European Committee for Electrotechnical Standardization
CG	Center of Gravity
DGNSS	Differential GNSS
DGON	Deutsche Gesellschaft für Ortung und Navigation
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DOF	Degrees of Freedom
DOP	Dilution of Precision
DP	Dynamische Positionierung
DVL	Doppler Velocity Log
ECDIS	Electronic Chart Display and Information System
ECEF	Earth-Centered, Earth-Fixed
EKF	Extended Kalman Filter
ENC	Elektronische Navigationskarte

Abkürzungsverzeichnis

EOT	Engine-Order-Telegraph
FDI	Fault Detection and Isolation
FDE	Fault Detection and Exclusion
FMCW	Frequency Modulated Continuous Wave
FOM	Figure of Merit
FTC	Fault Tolerant Control
FTS	Fast Time Simulation
GATE	GALILEO Test- und Entwicklungsumgebung
GBAS	Ground-Based AugmentationService
GIS	Geographische Informationssysteme
GMT	Gesellschaft für Maritime Technologie
GNC	Guidance, Navigation, Control
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HAS	High Accuracy Service
HDOP	Horizontal Dilution of Precision
HF	Hybridfähre
HMI	Human Maschine Interface
HPL	Horizontal Protection Level
HSW	Hochschule Wismar
HTTP	Hypertext Transfer Protocol
IMCA	International Marine Contractors Association
IMO	International Maritime Organization
IMM	Interagierende Mehrfachmodell
IMU	Inertial Measurement Unit
INS	Integriertes Navigationssystem

IoT	Internet-of-Things
IP	Interior-Point
LIDAR	Light Detection And Ranging
LMS	Least Mean Square
LoS	Line-of-Sight
LTE	Long Term Evolution
LQG	Linear Quadratic Gaussian
MAL	Maneuver Automation Level
MAROS	Maritime Robotik und Sensorik
MAS	Maneuver Assistance System
MEMS	Micro-Electro-Mechanical Systems
MESSIN	USV in Katamaranbauweise
MIMO	Multiple-Input-Multiple-Output
MJLS	Markov Jump Linear System
MPC	Model Predictive Control
MRAC	Model Reference Adaptive Control
MSCW	Maritimen Simulationszentrum Warnemünde
MVT	Maritime Verkehrstechnik
MMAE	Multiple Model Adaptive Estimator
NED	North-East-Down
NLP	Nichtlineare Optimierung
NMEA	National Marine Electronics Association
NMMT	Nationaler Masterplan Maritime Technologien
NLOS	Non-Line-of-Sight
OS	Own Ship
OSM	Open Street Map

Abkürzungsverzeichnis

PID	Proportional-Integral-Differenzial Regler
PL	Protection Level
PNT	Position, Navigation, and Timing
PoC	Proof-of-Concept
POD	Antriebsgondel
PPP	Precise Point Positioning
PPS	Pulse-Per-Second
PVT	Position, Velocity, and Time
RCP	Rapid-Control-Prototyping
RTCM	Radio Technical Commission for Maritime Services
RTK	Real-Time-Kinematik
ROT	Rate Of Turn
ROTI	Rotation Indicator
RWTH	Rheinisch-Westfälische Technische Hochschule
RAIM	Receiver Autonomous Integrity Monitoring
RSIVIA	Robust Set Inversion via Interval Analysis
SAPOS	Satellitenpositionierungsdienst der deutschen Landesvermessung
SBAS	Satellite-Based Augmentation Service
SEA GATE	Galileo-Testgebiet für maritime Anwendungen
SDS	Ship Data System
SES	Ship Engine Simulator
ShM	Shared Memory
SHS	Ship Handling Simulator
SMIS	USV aus dem F&E Projekt 'Subsea Monitoring via Intelligent Swarms'
SNAME	Society of Naval Architects and Marine Engineers
SST	Ship Safety & Security Trainer

SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SOG	Speed over Ground
SQP	Sequential Quadratic Programming
TCP	Transmission Control Protocol
TG	Target
UBR	Universität Bremen
UDP	User Datagram Protocol
UHF	Ultra High Frequency
URO	Universität Rostock
USV	Unmanned Surface Vehicle
UTC	Coordinated Universal Time
VDR	Voyage Data Recorder
VDOP	Vertical Dilution of Precision
VK	Versuchsträger Kooperation
VM	Versuchsträger Manöver
VPN	Virtual Private Network
VSIMM	Variable-struktur interagierendes Mehrfachmodell
VTSS	Vessel Traffic Services Simulator
WGS84	World Geodetic System 1984
WDOP	Weighted DOP
WORHP	Europäischer NLP-Solver der ESA - 'We Optimize Really Huge Problems'
XML	Extensible Markup Language
ZD	Zentrale Dienste

1 Projektvorstellung

Autoren: Jan-Jöran Gehrt, René Zweigel

Der Seehandel hat in den letzten 10 Jahren jährlich um 2-4 % zugenommen [1] - ausgenommen ist hierbei das 2020. Um diesem bereits hohen und stetig wachsenden maritimen Verkehrsaufkommen sowie zunehmenden Schiffsgrößen gerecht zu werden und vor allem auch einen sicheren Verkehrsfluss zu garantieren, sind modernste Technik, eine Erhöhung der Automatisierungsstufen und kooperatives Handeln zwischen allen maritimen Teilnehmern erforderlich. So muss der aktuelle Standort, die Fahrtrichtung, die momentane Geschwindigkeit und der aktuelle Zustand jedes Schiffes permanent und zuverlässig bekannt sein. Zudem sind ein sicherer und stets verfügbarer Datenaustausch sowie zusätzliche bordeigene Sensoren zur zuverlässigen Umfeldwahrnehmung notwendig. Auf Basis dieser Messdaten ist eine Zustandsbeschreibung des bewegten Systems möglich und es können Verhaltensvorhersagen getroffen werden. Ist das bei Teilnehmern eines Netzwerks möglich, dann kann darüber hinaus ein kooperatives Handeln erfolgen, das bspw. zentral koordiniert ist. Dadurch können Szenarien sicher automatisiert werden wie das Passieren zweier Schiffe in Hafenanlagen, Binnengewässern oder engen Kanälen.

Gerade im Hinblick auf sicheres automatisiertes Manövrieren mit Schiffen ist eine robuste und hochverfügbare Orts- und Lagebestimmung notwendig, wobei das neue europäische Satellitennavigationssystem Galileo eine entscheidende Rolle hierfür spielt. Mit bis jetzt insgesamt 22 Satelliten im Orbit hat das System die volle Einsatzfähigkeit - den Full Operational Capability (FOC) Status - erreicht und ist über den europäischen GNSS Galileo Open Service (OS) seitdem zur Navigation für jedermann nutzbar. Galileo ist ein rein ziviles System mit einer garantierten Verfügbarkeit, so dass Galileo für autonome Mobilitätsanwendungen uneingeschränkt verwendbar ist. Galileo erfährt eine stetig wachsende Relevanz: Gegenwärtig sind bereits 126 verschiedene Modelle Galileo-fähiger Mobiltelefone am Markt verfügbar. Im maritimen Kontext sind heute bereits 15 % der GNSS-Empfänger Galileo-fähig [2].

In Abb. 1.1 wird der Bezug zwischen den Vorteilen, die das neue europäische Satellitennavigationssystem Galileo für die maritime Anwendung bietet und den in der Abbildung

1 Projektvorstellung

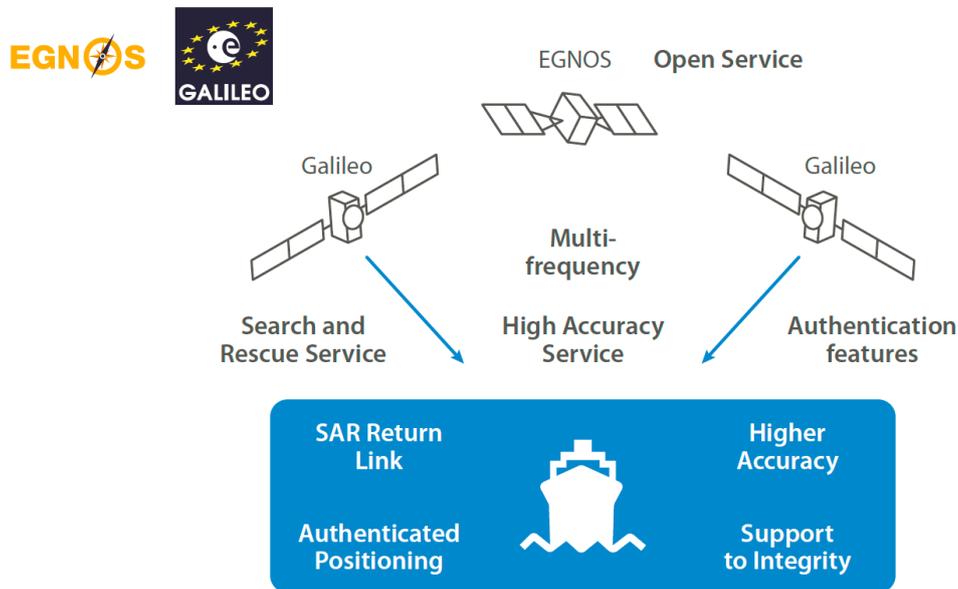


Abbildung 1.1: Europäisches GNSS in der maritimen Anwendung, [2]

aufgeführten maritimen Schwerpunkten hergestellt. Beispielsweise bietet der Galileo OS beim Empfang von jedes einzelnen Galileo-Satelliten standardmäßig zwei unterschiedliche Frequenzen, so dass Laufzeitfehler der Satellitensignale durch die Ionosphäre sehr genau ausgeglichen werden können. Mit der interoperablen Auslegung des Galileo-Systems in Bezug auf GPS, ist es möglich, GPS und Galileo gleichzeitig zu verwenden und damit die generelle Satellitenverfügbarkeit deutlich zu erhöhen. Mit der breitbandigen Galileo-Frequenz E5 ist es möglich, sich robust und hochgenau zu lokalisieren, ebenso wie bei Verwendung der neuen Galileo-Korrekturen im Format RTCM V3.2 und der zukünftigen Korrekturen mittels EGNOS V3. Diese Eigenschaften (u.a.) des Galileo-Systems erlauben neue Anwendungen bei der maritimen Navigation, bilden die Grundlage für autonome Funktionen und erlauben eine robuste Lokalisierung von maritimen Fahrzeugen. Da sich ein deutlicher Mehrwert durch die gleichzeitige Nutzung von mehreren Satellitennavigationssystemen ergibt, wurde im Projekt die Kombination von Galileo und GPS (Multikonstellation) inklusive Korrekturdiensten weiterentwickelt und in den realen automatisierten Einsatz gebracht.

Neben den Satellitensignalen müssen bordeigene Sensordaten und zentral erhältliche, terrestrische Daten einbezogen werden, um Ort und Lage robust und permanent verfügbar ermitteln zu können. Bordeigene Sensorik hat ebenfalls einen hohen Stellenwert und muss um genauere und robustere Technik gegenüber dem Stand der Technik erweitert werden, um gerade im Nahbereich von 0 bis 300 Metern und unter allen Wetter- und Umweltbedingungen ein exaktes Bild der Umgebung zu ermitteln, wie es z. B. für ein automatisiertes Anlegesystem oder für ein aneinander Vorbeifahren (Passieren) im Hafengebiet notwen-

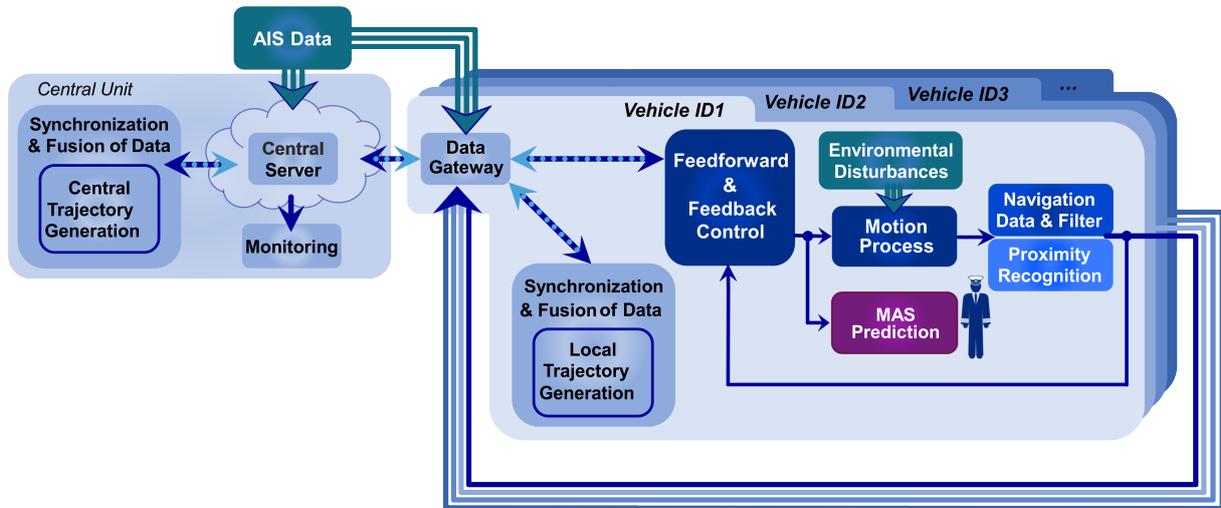


Abbildung 1.2: Entwickeltes Systemkonzept in GALILEOnautic 2 für vernetztes, kooperatives Manövrieren, [3]

dig ist. Störend vor allem auf die Geschwindigkeitsinformationen in flachen, begrenzten Gewässern wirken sich u.a. eigene Verwirbelungen, Flachwassereffekte, Banking. Die Positionsbestimmung kann in der Nähe von Hafengebäuden aufgrund von Mehrwegeausbreitung der Satellitensignale beeinflusst sein.

Abb. 1.1 betont unter anderem die Wichtigkeit satellitenbasierter Navigation als Grundlage für generelle Navigationsaufgaben, für intelligente Schiffsführungssysteme und für automatisierte/autonome Konzepte. GALILEOnautic 2 griff genau diese Punkte auf und entwickelte Konzepte des Vorgängerprojekts GALILEOnautic (FKZ 50 NA 1610) weiter, mit dem Ziel der Entwicklung eines modularen Systems, das von assistierter, intelligenter Unterstützung des Nautikers bis hin zur automatisierten Durchführung von abgegrenzten Teilaufgaben/Wiederholungsaufgaben einsetzbar ist. Hierbei wurde die GNSS-basierte hochgenaue Navigationslösung um gezielte Integritätsprüfungen erweitert. Außerdem wurde die Vernetzung von mehreren Schiffen über ein zentral koordiniertes Netzwerk untersucht, um kooperativ die jeweilige Navigationsaufgabe sicher und effizient zu erfüllen.

1.1 Aufgabenstellung

Ziel von GALILEOnautic 2 war die Weiterentwicklung der technischen Vorarbeiten aus GALILEOnautic, die Anpassung der entsprechenden Methoden an relevante Anwendungen und die Überführung in industrielle Systeme und deren Demonstration auf realen Schiffen.

Abb. 1.2 visualisiert das vernetzte, kooperative maritime Gesamtsystem mit einem per-

1 Projektvorstellung

manentem Datenaustausch zwischen allen Netzwerkteilnehmern und einer zentralen Koordinationskomponente. Die Abbildung zeigt zudem die wesentlichen Komponenten und Module des Gesamtsystems. In der Struktur ist jedes Schiff mit einem Data Gateway ausgestattet über das Daten gesammelt, empfangen und gesendet werden können. Hierüber besteht die Verbindung zur zentralen, koordinierenden Einheit, der *Central Unit*. In der zentralen Einheit erfolgt eine permanente Trajektorienberechnung unter Berücksichtigung aller Netzwerkteilnehmer, aktueller Verkehrssituation und Umwelteinflüsse, so dass sicheres und kooperatives Handeln möglich ist. Die Trajektorien für jedes einzelne Schiff werden über das Data Gateway an die Schiffe übertragen, die diese wiederum lokal robust einregeln. An Bord erfolgt darüber hinaus unter anderem noch die Nahfeldererkennung, die Unterstützung durch das Manöver Assistenz System (MAS) und die Zustandsbestimmung mittels eines maritimen Navigationsfilters. Die im Teilvorhaben der RWTH adressierten Entwicklungsaufgaben waren dabei das Navigationsfilter, die Umfelderkennung und die Synchronisierung und Fusion der Informationen sowohl lokal als auch in der zentralen Einheit.

Das Gesamtsystem sollte in GALILEOnautic 2 so konzipiert werden, dass die freie Wahl zwischen Automatisierungsstufe SAE-Level 1 bis 3 gemäß [4] ermöglicht wird, deren Potential aufgezeigt wird und darüber hinaus für die Automatisierungsstufe SAE-Level 4 ein konkretes, jedoch abgegrenztes Anwendungsbeispiel umgesetzt wird. Da in der Stufe 4 die eigenständige Behandlung von Fehlern durch das System selbst ein Kernelement ist, mussten die entsprechenden Fehlerquellen/-szenarien für die spezielle Anwendung möglichst vollständig identifiziert und geeignete Eingriffs- und Notfallmaßnahmen definiert und umgesetzt werden. Um die Funktionalität der Einzelmodule, als auch die beschriebene Modularität des Systems nachzuweisen, sollten daher in GALILEOnautic 2 fünf Anwendungsszenarien untersucht werden:

Assistiertes Manövrieren mit dem Fährschiff HF-Berlin auf SAE-Level 1: Untersuchung einer assistierten Fahrt zwischen Gedser und Rostock mit dem Fährschiff HF-Berlin, sh. Abb. 1.3. Dabei sollte weder steuernd in den Prozess eingegriffen werden noch das Navigationssystem der HF-Berlin um ein Navigationsfilter zur Multisensorfusion erweitert werden. Die RWTH war daher in diesen Entwicklungsprozess nicht involviert.

Assistierte Lotsung in Bremerhaven auf SAE-Level 1: Untersuchung einer assistierten Lotsung in Bremerhaven mit einem Lotsenschiff, sh. Abb. 1.4, unter Einsatz eines innovativen Lotsassistentensystem. Das Lotsassistentensystem sollte durch hochgenaue GNSS-basierte Sensorik erweitert werden, dessen Navigationsinformationen dem Lotsen zur Verfügung gestellt und zur Echtzeit-Berechnung von zukünftigen Trajektorien genutzt werden. Die Trajektorie wiederum sollte dem Lotsen helfen, die Bewegung des Schiffs und entsprechende Kollisionsrisiken in naher Zukunft einzuschätzen. Die RWTH sollte dazu ein mi-

1.1 Aufgabenstellung



Abbildung 1.3: Hybrid Ferry Berlin, Quelle: Siemens / Ulrich Wirrwa.

niaturisiertes Navigationsfilter entwickeln, das als mobiles Filter geeignet ist und von einem Lotsen im Alltag an Bord gebracht werden kann.



Abbildung 1.4: Lotsenschiff und Versuchsfahrzeug bei Abschluss GALILEOnautic 2, Quelle: TRENZ GmbH.

1 Projektvorstellung

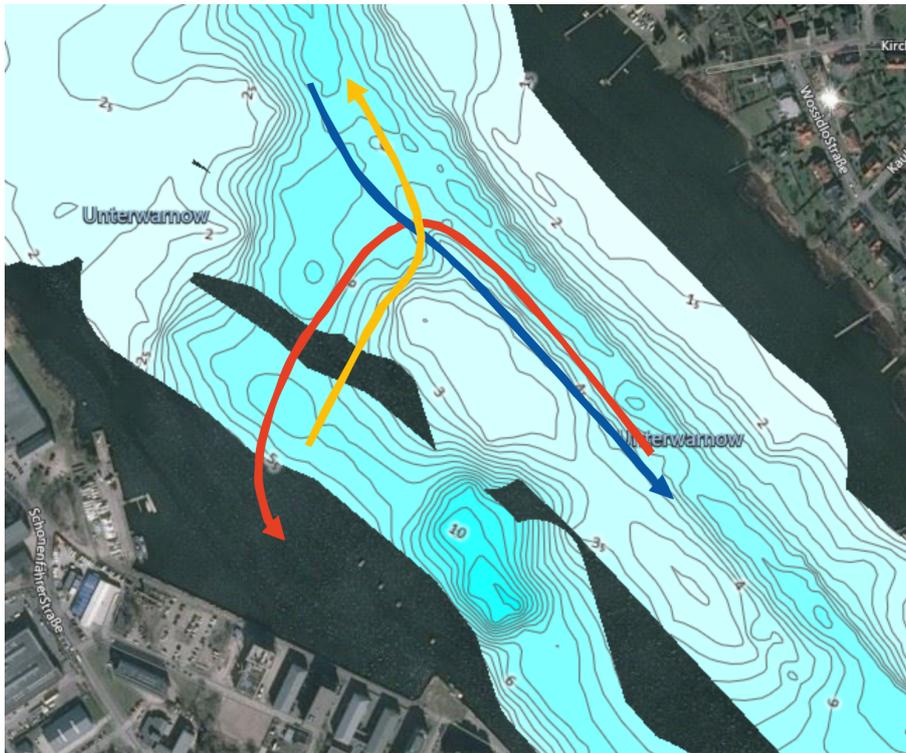


Abbildung 1.5: Geplantes Begegnungsszenario DENEb mit den vernetzten USV-MESSIN und USV-SMIS und einem unvernetzten, störenden Schlauchboot; Grafik erstellt durch Universität Rostock.

Kooperatives Manövrieren mit Forschungsschiff DENEb auf SAE-Level 2: Darstellung einer teilautomatisierten Hafendurchfahrt im vernetzten, kooperativen Netzwerk mit dem Forschungsschiff DENEb und weiteren vernetzten Fahrzeugen und einem nicht-vernetzten Kollisionspartner, sh. Abb. 1.5. Hier sollten die kooperativen Navigations- und Manövrieraufgaben durch das Schiff, die Umfeldbeobachtung allerdings weiterhin durch den Operator auf dem Schiff erfolgen. Die RWTH sollte hierzu ein robustes und integriertes Navigationsfilter beitragen, sowie die Synchronisierung der Bewegungs- und Umfelddaten im Netzwerk sicherstellen.

Bedingt automatisiertes Anlegen mit Forschungsschiff DENEb auf SAE-Level 3: Automatisiertes Anlegen auf Grundlage einer hochgenauen, georeferenzierten Navigationslösung. Dazu sollte die DENEb ein Anlegemanöver vollziehen, bei dem sie sich über einen Bogen in die Anlegestelle hineindreht und anschließend mit den Querstrahlrudern an die Pier schiebt. Hierbei dient die Navigationslösung sowohl als Messgröße der lokalen Fahrregelung des Forschungsschiffes, als auch als Messgröße für die Berechnung der übergeordneten Anlegetrajektorien. Umfeldsensorik wird zur Wahrnehmung von festen infrastrukturellen Hindernissen wie Tonnen und Pier benötigt. Die RWTH sollte dazu ein hochgenaues, robustes und integriertes Navigationsfilter beitragen. Zudem sollte untersucht werden, inwiefern

1.2 Voraussetzungen des Vorhabens

LiDAR-gestützte Umgebungserkennung beim Anlegeprozess unterstützen kann.

Automatisiertes Anlegen mit unbemanntem Fahrzeug MESSIN zwischen SAE-Level 3 und 4: Automatisiertes Anlegen in einem Szenario wie in Abb. 1.6, wobei das Fahrzeug vollkommen selbstständig agieren und auch im Notfall eigenständig reagieren können muss. Gegenüber dem Anlegeszenario mit der DENEb ist in dem hier beschriebenen Szenario daher nicht der menschliche Fahrzeugführer für die Beobachtung der Umgebung verantwortlich – hierbei sollte die Umgebung von der Umfeldsensorik beobachtet werden. Die Aufgabe der RWTH war es, ein hochgenaues, robustes und integriertes Navigationsfilter bereitzustellen und darüber hinaus ein Verkehrslagenerkennungssystem zu entwickeln, das ein zuverlässiges Bild der Anlegesituation wiedergibt und auf dessen Grundlage ein sicheres Anlegemanöver durchgeführt werden kann.



Abbildung 1.6: Unbemanntes Fahrzeug (USV) Messin auf der Warnow, bei Anlegemanöver an der Pier des Breitling e.V. in Rostock, Quelle: Universität Rostock.

1.2 Voraussetzungen des Vorhabens

Es gab keine kritischen Voraussetzungen, die relevant für die Durchführbarkeit und damit für den Projektbeginn im Allgemeinen waren.

1.3 Konsortium

Um die Transition der Konzepte aus dem Vorgängerprojekt GALILEOnautic [5] von Demonstratoren hin zu industriell relevanten Anwendungen realisieren zu können wurde ein

1 Projektvorstellung

schlagkräftiges Konsortium aus hochspezialisierten Hochschulen und Industriepartnern gebildet:

RWTH Aachen University

Im Projekt übernahm die RWTH die Rolle der Projektkoordination. Als wissenschaftlichen Beitrag entwickelte sie ein maritimes eng gekoppeltes, Trägerphasen-gestütztes Navigationsfilter zur Berechnung von hochgenauen Schiffszuständen, die die Basis für die Schiffsregelung und die Trajektoriengenerierung bildete. Das Navigationsfilter wurde zudem miniaturisiert, sodass es Teil eines mobilen Lotsenassistenzsystems des Industriepartners TRENZ GmbH werden konnte. Des Weiteren entwickelte die RWTH neuartige Integritätsmethoden zur Ermittlung der übergreifenden Integrität einer maritimen Multi-sensorfusion. Des Weiteren entwickelte die RWTH ein Modul zur Umgebungswahrnehmung und dynamischen Kartierung durch Fusion von Umfeldsensoren. In diesem Kontext wurde ein LiDAR-basiertes Umfelderkennungssystem mit Objekterkennung (u.a. auch Pierkantenerkennung) und Verfolgung von dynamischen Objekten entwickelt.

Universität Bremen

Die Universität Bremen führte Ihre Entwicklung zur zeit- und energieoptimalen Bestimmung von Trajektorien inkl. Stellgrößen aus dem Vorgängerprojekt fort. Dabei sollten die Algorithmen so skalierbar gemacht werden, dass sie sowohl auf Einzelschiffebene als auch im Netzwerk zur Kollisionsvermeidung zwischen vernetzten Schiffen, mit anderen nicht vernetzten Schiffen oder mit anderen festen Hindernissen eingesetzt werden können. Diese Entwicklung sollte von den Industriepartnern nutzbar gemacht werden, um in naher Zukunft die Trajektorienoptimierung auch im heterogenen Mischverkehr aus manuellem und automatisiertem Schiffsverkehr einzusetzen. Die Aufgabe der Universität Bremen umfasste zudem die Erweiterung der Algorithmen um eine Risikobewertung und die Umsetzung vereinfachter Modellbildung, die das Übertragen der Algorithmen von einem Schiff auf das nächste vereinfachen sollte.

Universität Rostock

Die Universität Rostock erweiterte Ihre Methoden zur robusten Bahnregelung um fehler-tolerante Reglersysteme, die Aktorfehler detektieren, identifizieren und betroffene Aktoren ausschließen können. Des Weiteren wurden Algorithmen zur dynamischen Positionierung mit dem Forschungsschiff DENEb und Regler zum Anlegen mit DENEb und unbemanntem Demonstrator entwickelt. Darüber hinaus war die Universität Rostock für die Vorbereitung und Inbetriebnahme der Demonstratoren in Rostock zuständig, insbesondere auch für die Integration der GALILEOnautic Methoden in das Forschungsschiff DENEb, indem sie die Schnittstellenimplementierung und das Datenmanagement maßgeblich betreute.

Hochschule Wismar

Die Hochschule Wismar entwickelte im Projekt das Manöver Assistenzsystem (MAS) wei-

1.4 Planung und Ablauf des Vorhabens

ter. Die Weiterentwicklung betraf zum einen die Integration von MAS in das automatisierte Systemkonzept von GALILEOnautic, in dem es die Rolle der Kommunikationsplattform zur Kommunikation mit dem nautischen Personal in Form grafischer Nutzeroberflächen einnahm. Darüber hinaus sollte MAS so angepasst werden, dass es zukünftig leicht in Produkte der Industriepartner integriert werden kann. Die Hochschule Wisamr untersuchte zudem RADAR-basierte Nahfeldsensorik zur Detektion von Hindernissen, Fremdverkehr usw. sowie Infrastrukturseitig gestützte Ultra-Wideband (UWB) Sensorik zur Unterstützung der Automatisierungsalgorithmen beim Anlegeprozess.

CGI Deutschland Ltd. & Co. KG

Die CGI Deutschland Ltd. & Co. KG setzte Ihre Entwicklung eines robusten und sicheren Kommunikationsnetzwerk bestehend aus Wasserfahrzeugen, Landstellen und sonstigen Beteiligten fort. Dabei wurde insbesondere die Entwicklung eines skalierbaren Netzwerks mit theoretisch unendlich vielen Netzwerkteilnehmern adressiert. CGI konzipierte zudem eine Kontrolleinheit auf Einzelschiffebene zur Prüfung der Validität der Einzelmodule des GALILEOnautic-Konzepts.

Raytheon Anschütz GmbH

Die Raytheon Anschütz GmbH wirkte als Systemintegrator aus Sicht der automatisierten Schifffahrt und brachte dazu ihr intelligentes Brückensystem SYNAPSIS in Form eines modularen Containers aus dem Projekt ACTRESS ein, der als mobile, zusätzliche Brücke auf dem Forschungsschiff DENEb installiert wurde.

TRENZ GmbH

Die TRENZ GmbH wirkte als Systemintegrator aus Sicht neuartiger assistierter Lotsung. Dabei entwickelte TRENZ GmbH ihr Pilot Information Assistant (PIA) System weiter. Dazu wurden Teile des GALILEOnautic-Konzepts in PIA integriert. Die hochgenaue RWTH-Sensorfusion wurde in minaturisierter Form eingebracht, die Trajektorienoptimierung der Universität Bremen als intelligente Bewegungsvorhersage mit Risikobewertung nutzbar gemacht. Darüber hinaus entwickelte TRENZ einen LiDAR-basierten Abstandssensor, der insbesondere in engen Schleusendurchfahrten den Lotsen unterstützt.

1.4 Planung und Ablauf des Vorhabens

Das Vorhaben GALILEOnautic 2 wurde mit einer Laufzeit von 2,5 Jahren beginnend am 1. Oktober 2018 geplant. Insgesamt wurden acht Hauptarbeitspakete definiert.

AP1000 Projektmanagement

(verantwortlich RWTH)

1 Projektvorstellung

AP2000 Maritime, integrale Lokalisierung und Netzwerkregelung

(verantwortlich RWTH)

- AP2100 GNSS-basierte maritime Orts- und Lagebestimmung mittels Sensorfusion
- AP2300 Integritätsbewertung der GNSS-Signale und Sensordaten
- AP2400 Koordiniertes, kooperatives Manövrieren in Netzwerken

AP3000 Echtzeitfähige Optimalsteuerung für vernetzte maritime Systeme

(verantwortlich Universität Bremen)

- AP3100 Modellanpassung und Online-Adaption auf Basis von Modellfehlern
- AP3200 Erweiterung der Trajektorienberechnung und Übertragung auf lokale Systeme
- AP3300 Risikobewertung von Trajektorien und störungssichere Neuplanung

AP4000 Robuste, fehlertolerante Schiffsregelung

(verantwortlich Universität Rostock)

- AP4100 Bedatung Reglerentwurfsmodell
- AP4200 Hybride Schiffsregelung
- AP4300 Fehlerdiagnose und Fehlertoleranz
- AP4500 Erprobung mit den USV und SHS

AP5000 Simulationsgestützte und assistierte Schiffsführung

(verantwortlich Hochschule Wismar)

- AP5100 Schiffsdatenerfassung und Generierung von Testdaten
- AP5200 Entwicklung von Umfelderkennungssystemen für das Nahfeld im Hafen und zum Docking
- AP5400 Entwicklung von Störmodellen und Sensordatenintegration
- AP5500 Umsetzung Manöver-Assistenz-System (MAS) auf einer Fähre
- AP5600 Automatisches Docking (AD)

AP6000 Optimierte Datenübertragung und Anbindung an externe Dienste

(verantwortlich CGI Deutschland Ltd. & Co. KG)

- AP6100 Kommunikationsnetzwerk
- AP6200 Externe Datendienste

- AP6300 Kommunikationssysteme für Szenarien

AP7000 Brückensystementwicklung für teilautomatisierte Schiffssysteme

(verantwortlich Raytheon Anschütz GmbH)

- AP7100 Schnittstellenimplementierung am Brückensystem SYNAPSIS
- AP7200 Integration und funktionale Einbindung der physikalischen und generischen Schiffsmodelle
- AP7300 Integration des Trajektoriengenerators und Umsetzung automatisierter Fahrfunktionen
- AP7400 Überwachung von Sensorik und Aktorik durch Abgleich mit Schiffsmodell

AP8000 Maritime Informationsplattform

(verantwortlich TRENZ GmbH)

- AP8100 Virtuelles Boarding
- AP8200 Echtzeitfähige Risikoanalyse
- AP8300 Echtzeit Navigationsunterstützung
- AP8400 Erweiterung des PilotPlugs für hochgenaue Positionsbestimmung

AP9000 Überführung der Ansätze auf ein Forschungsschiff

(verantwortlich RWTH)

- AP9100 Schnittstellenerstellung zu ACTRESS und Vorbereitung Brückenintegration auf dem Forschungsschiff
- AP9200 Messdatenerfassung und Modellierung
- AP9300 Szenarienumsetzung vernetztes Manövrieren im Hafen (SAE-Level 2)
- AP9400 Szenarienumsetzung automatisiertes Anlegen (SAE-Level 3 und 3+)
- AP9500 Abschlussdemonstration

Basierend auf den Arbeitspaketen wurde der Projektzeitplan in Abbildung 1.7 definiert.

Hinweis: In manchen Hauptarbeitspaketen fehlen Zwischennummern innerhalb der Unterarbeitspakete. Die fehlenden Nummern repräsentieren Arbeitspakete, die während der Antragstellung und in enger Absprache mit dem Projektträger in andere Arbeitspakete integriert oder gestrichen wurden.

1 Projektvorstellung

AP	Q4 2018			Q1 2019			Q2 2019			Q3 2019			Q4 2019			Q1 2020			Q2 2020			Q3 2020			Q4 2020			Q1 2021		
	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
1100	(RWTH) Projektmanagement und Projektdokumente																													
1110																														
1120																														
1130																														
1140																														
2000	(RWTH) Maritime, integrale Lokalisierung und Netzwerkregelung																													
2100																														
2200																														
2300																														
2400																														
3000	(UBR) Echtzeitfähige Optimalsteuerung für vernetzte maritime Systeme																													
3100																														
3200																														
3300																														
3400																														
4000	(URO) Robuste, fehlertolerante Schiffsregelung																													
4100																														
4200																														
4300																														
4400																														
4500																														
5000	(HSW) Simulationsgestützte und assistierte Schiffsführung																													
5100																														
5200																														
5300																														
5400																														
5500																														
5600																														
6000	(SCISYS) Optimierte Datenübertragung und Anbindung an externe Dienste																													
6100																														
6200																														
6300																														
7000	(RAN) Brückensystementwicklung für teilautomatisierte Schiffssysteme																													
7100																														
7200																														
7300																														
7400																														
7500																														
8000	(TRENZ) Maritime Informationsplattform																													
8100																														
8200																														
8300																														
8400																														
9000	(RWTH) Überführung und Demonstration der Automatisierungsansätze auf ein Forschungsschiff																													
9100																														
9200																														
9300																														
9400																														
9500																														

Abbildung 1.7: Projektzeitplan in GALILEOnautic 2 bei Projektbeginn

1.5 Änderungen des Vorhabens während der Projektlaufzeit

Im Kontext der hochgenauen Sensorfusion wurden Arbeiten zur Erhöhung der Zuverlässigkeit des eng gekoppelten, Trägerphasen-gestützten Navigationsfilters ergänzt (RWTH). Im Bereich der Umfeldwahrnehmung und Hinderniserkennung wurden wichtige Lücken im Nahbereich bis 80 Meter durch LiDAR Technologie geschlossen (RWTH) und Lücken im mittleren Entfernungsbereich zwischen 80 und 300 Meter durch Erweiterung der UWB-Technologie adressiert (Hochschule Wismar). Es wurden zudem Arbeiten formuliert, die das MAS auch auf die unbemannten Versuchsträger übertragen, um in dem Fall in dem die DENEb nicht umgebaut werden kann, alle wichtigen Entwicklungen mit Alternativfahrzeugen zeigen zu können (Hochschule Wismar). Weiter wurden Arbeiten zur Integration

von Kamerasystemen zur Erhöhung der Akzeptanz u.a. beim nautischen Personal ergänzt (Raytheon Anschütz GmbH).

AP2000 Maritime, integrale Lokalisierung und Netzwerkregelung

(verantwortlich RWTH)

- AP2500 Erweiterung und Integritätsüberwachung der Trägerphasen-gestützten Navigation in Multikonnexion (GPS & Galileo)
- AP2600 Umfeldlokalisierung mittels LiDAR

AP3000 Echtzeitfähige Optimalsteuerung für vernetzte maritime Systeme

(verantwortlich Universität Bremen)

- AP3500 Erweiterung der Risikobewertung um komplexe Lotsungsszenarien mit Schlepperschiffen
- AP3600 Erweiterung der Softwaremodule zur Trajektorienberechnung zur Verwendung standardisierter Datenquellen

AP5000 Simulationsgestützte und assistierte Schiffsführung

(verantwortlich Hochschule Wismar)

- AP5700 Strategische Erweiterung des Manöverassistenzsystems (MAS)
- AP5800 Ausbau und Erweiterung des UWB Systems für lange Reichweiten

AP7000 Brückensystementwicklung für teilautomatisierte Schiffssysteme

(verantwortlich Raytheon Anschütz GmbH)

- AP7500 Integration eines Kamerasystems für die Nahbereichsnavigation zur Peilung von Objekten und Erzeugung eines Positionsdatums auf Basis kamerabasierter optischer terrestrischer Navigation

Basierend auf den ergänzenden Arbeitspaketen wurde der Projektzeitplan aktualisiert. Der finale Projektzeitplan ist in Abbildung 1.8 dargestellt, wobei die Erweiterungen mit neuen Arbeitspaketen gelb gekennzeichnet sind. Gegenüber dem Projektzeitplan Abbildung 1.7 wurden zudem einige der ursprünglichen Arbeitspakete neu allokiert worden.

1.6 Technische Ausgangsbasis

Im Projekt GALILEOnautic 2 wurden keine bekannten Konstruktionen, Verfahren und Schutzrechte Dritter für die Durchführung des Vorhabens benutzt. Das Projekt GALILEOnautic 2 knüpfte an das Vorgängervorhaben GALILEOnautic an.

1 Projektvorstellung

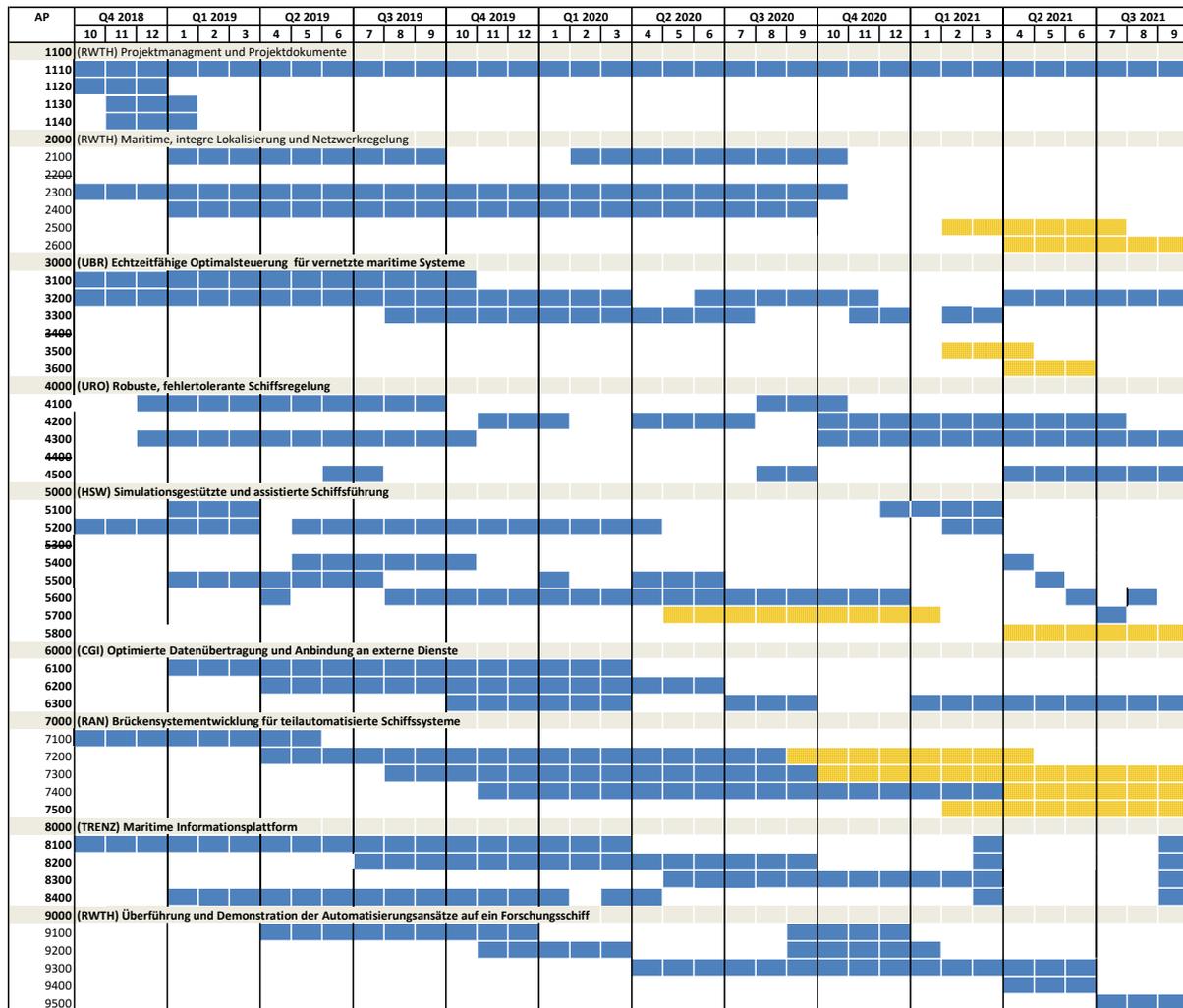


Abbildung 1.8: Projektzeitplan in GALILEOnautic 2 nach kosteneutraler Verlängerung (blau) und Aufstockung (gelb)

In GALILEOnautic wurden vernetzte Demonstratoren aufgebaut, die miteinander kooperativ navigieren und optimiert manövrieren können. Es wurde darüber hinaus gezeigt, dass nicht vernetzte Schiffe vom Netzwerk berücksichtigt werden können [5]. An folgenden wissenschaftlichen und technischen Stand wurde im Speziellen in GALILEOnautic 2 aus Sicht der RWTH angeknüpft: Im Projekt GALILEOnautic wurde eine GNSS-basierte Satellitennavigation für die maritime Anwendung entwickelt zur hochgenauen und hochverfügbaren Positionsbestimmung. Es wurde eine Sensorfusion umgesetzt, welche die GNSS-Beobachtungen Pseudoranges und Deltaranges mit Beschleunigungen und Drehraten einer inertialen Messeinheit (IMU) und einem maritimen Geschwindigkeitssensor Doppler Velocity Log (DVL) fusionierte. Neben der Integration maritimer Sensorik wurde insbesondere die Integration des GNSS-Systems Galileo untersucht. Außerdem wurde ein Synchronisationsmodul entwickelt, welches in einem vernetzten kooperativen System ei-

ne einheitliche, GNSS-basierte Zeitbasis sicherstellte und die auftretenden Datenströme entsprechend synchronisierte.

1.7 Zusammenarbeit mit anderen Stellen

Im Rahmen des Projekts wurde mit assoziierten Partnern zusammengearbeitet. Diese Partner sind im Folgenden aufgezählt.

- **Bundesamt für Seeschifffahrt und Hydrographie**

Das Bundesamt für Seeschifffahrt und Hydrographie (BSH) wurde Anfang 2020 aktiver assoziierter Partner des Projektkonsortiums und in diesem Zuge in den Kooperationsvertrag aufgenommen. Auf dieser Basis beteiligte sich das BSH maßgeblich am Umbau des Forschungsschiffs DENEBO und stellte anschließend das Forschungsschiff für insgesamt vier Erprobungen im Zeitraum März 2021 bis September 2021 zur Verfügung.

- **Scandlines Deutschland GmbH**

Scandlines unterstützte das Projekt durch Zugang zu den VDR-Schiffsdaten, die auf den Fährn zwischen Rostock und Gedser aufgezeichnet wurden, und kostenfreie Mitfahrten. Außerdem konnten auf dieser Route die Ansätze zum assistierten Navigieren und Manövrieren umgesetzt und evaluiert werden.

- **Hafenlotsengesellschaft Bremerhaven**

Die Hafenlotsengesellschaft Bremerhaven unterstützte das Projekt durch Beratung und Bewertung der Assistentenentwicklungen aus Sicht der Experten des Manövrierens schlechthin: den Lotsen.

2 Maritimes hochgenaues Navigationsfilter

Autoren: Jan-Jöran Gehrt, Shuchen Liu, Maximilian Nitsch, David Stenger

2.1 Einleitung und Stand der Technik

Zum heutigen Zeitpunkt werden in der Schifffahrt die Navigationszustände (teilweise redundant) durch einzelne Sensoren, die im Standalone-Betrieb arbeiten, bestimmt. Die Position und der Course Over Ground (COG) werden mittels GNSS-basiertem Electronic Position Fixing System (EPFS) ermittelt. Das sogenannte Heading, also die Orientierung des Schiffskörpers gegenüber geografischem Nordpol und die Drehrate Rate Of Turn (ROT) mittels Kompass - Ab einer bestimmten Schiffsgröße gemäß den Forderungen der IMO in [6] mit einem Inertial Navigation System (INS)-basierten elektronischen, nicht magnetischen Kompass. Die Geschwindigkeit des Schiffs gegenüber Grund Speed Over Ground (SOG) wird häufig mit einem Doppler Velocity Log (Doppler Velocity Log (DVL)) ermittelt. Das automatisierte System gemäß Abb. 1.2 fordert jedoch eine synchrone, hochfrequente, hochverfügbare Bereitstellung der Navigationszustände mit hoher Genauigkeit. In einer Sensorfusion können die Messinformationen der einzelnen Sensoren, die bisher im Standalone-Betrieb genutzt wurden, optimal fusioniert werden, sodass das durch Sensorrauschen induzierte Zustandsrauschen minimiert wird und komplementäre Informationen genutzt werden um den Zustandsvektor zu erweitern. Ein gängiges Vorgehen ist, die Informationen des robusten, jedoch lediglich kurzzeitstabilen INS mit den Informationen des langzeitstabilen GNSS zu fusionieren, um eine langzeitstabile, robuste Zustandsbestimmung zu erhalten.

In [5] wurde die Entwicklung eines INS-basierten eng gekoppelten Navigationsfilters mit zusätzlicher Navigationszustandsstützung durch ein DVL in GALILEOnautic entwickelt und vorgestellt. Dabei wurde das Navigationsfilter durch die GPS- und Galileo-Beobachtungen

2 Maritimes hochgenaues Navigationsfilter

Pseudorange und Deltarange gestützt. Beobachtungskorrekturen wurden je nach Verfügbarkeit mit differentiellen Daten des SAPOS-Dienstes, mit Zweifrequenzkorrekturen oder Korrekturwerten ermittelt via Klobucharmodell durchgeführt. Das DVL wurde lose gekoppelt integriert, d.h. es wurden die vom DVL ermittelten Geschwindigkeiten im eigenen Sensorkoordinatensystem genutzt. Die Fusion wurde mittels eines Extended Kalman-Filters (EKF) umgesetzt. Die Initialisierung des Orientierungszustands wurde mit Hilfe eines mitgeführten Magnetometers durchgeführt. In [7] wurde gezeigt, dass mit den untersuchten GNSS-Beobachtungskorrekturen Genauigkeiten im Bereich von 50cm erreicht werden konnten. Es wurde zudem gezeigt, dass das DVL die Navigationslösung soweit stützen kann, dass auch bei GNSS-Ausfall weiterhin eine zuverlässige, vom Guidance-Navigation-Control (GNC)-System nutzbare, Positionslösung ausgegeben werden kann.

Fehler nach Initialisierung der Filterzustände konnten in GALILEOnautic durch die hohe Dynamik der dort untersuchten unbemannten Versuchsträger beobachtet werden und konvergierten schnell. Da insbesondere die Beobachtbarkeit von Orientierung und orientierungsbezogenen Sensorfehlern stark von der Dynamik des Fahrzeugs abhängt [8, S. 203 ff.], leidet die Beobachtbarkeit der Orientierung bei trägen Fahrzeugen, wie bspw. dem Forschungsschiff DENEBO. Um dieser Problematik zu begegnen muss ein Sensor zur hochgenauen Orientierungsbestimmung in das Navigationsfilter integriert werden - Zumindest zur Ermittlung einer exakten initialen Orientierung und orientierungsbezogenem Sensorfehler. Daher wurde in GALILEOnautic 2 die Integration eines in der Schifffahrt gebräuchlichen Hemispherical Resonant Gyroscope (HRG) in das Navigationsfilter entwickelt.

Satellitenkonstellationsbedingt liegt die größte Unsicherheit der GNSS-basierten Navigationszustandsbestimmung in der Vertikalen. In GALILEOnautic 2 wurde daher untersucht inwiefern die Integration von Pegelstandsdaten einer Messstation der Wasserstraßen- und Schifffahrtsverwaltung des Bundes (WSV) die Navigationslösung genauer und robuster machen kann.

Die Genauigkeit der Navigationslösung hängt allgemein maßgeblich von der Methodik der Korrektur der GNSS-Beobachtungen ab. Damit durch die in GALILEOnautic entwickelte GPS-Galileo-Dualkonstellation nicht nur eine besonders robuste, sondern auch eine genaue Navigationslösung ermittelt werden kann, wurde daher in GALILEOnautic 2 differenzielles GNSS (DGNSS) in Multikonstellation mit Galileo untersucht. Dazu wurde das Format Radio Technical Commission for Maritime Services (RTCM) in der neuesten Version 3.2 implementiert und die Dualkonstellationsalgorithmen entsprechend erweitert.

Für den Hafenbetrieb fordert die International Maritime Organization (IMO) eine Positionsgenauigkeit von 1 m und für das Anlegen 0,1 m [9]. Um ein zuverlässiges und robustes Navigationssystem mit der genannten Positionsgenauigkeit bereitzustellen reicht DGNSS

jedoch nicht aus. Über DGNSS hinaus ist die Integration der GNSS-Trägerphase in das Navigationsfilter erforderlich. GNSS-Trägerphasen sind durch eine ganzzahlige Anzahl unbekannter Zyklen (Mehrdeutigkeiten) für jeden Satelliten beaufschlagt. Um mit einem Navigationssystem, das die Trägerphasen integriert, eine hohe Genauigkeit zu erreichen, muss die unbekannte Anzahl von Zyklen geschätzt werden. Eine gängige Technik besteht darin, zunächst die Float-Mehrdeutigkeiten zu schätzen (wobei die ganzzahlige Natur der Mehrdeutigkeiten vernachlässigt wird) und sie anschließend mit LAMBDA zu korrigieren [10]. In GALILEOnautic 2 wurde darauf aufbauend die enge Kopplung der Trägerphase mit dem INS im Navigationsfilter untersucht. Es wurde dabei insbesondere die Herausforderung der nicht kontinuierlich verfügbaren differenziellen Daten adressiert, sowie die Verkürzung der Konvergenzzeit durch Integration des Pegelstands.

Um eine zeitlich exakte Fusion von IMU, GNSS und DVL zu erreichen, wurde in [7] die Verzögerung der DVL-Messung ermittelt, indem der digitale GNSS-Trigger, der so genannte Pulse-Per-Second (PPS) und die geschätzte Signallaufzeit des DVL im Wasser verwendet werden, die vom selbigen bereitgestellt wird. Die nachträgliche Integration des Navigationsfilters in bereits bestehende Schiffe, wie das Forschungsschiff DENEb, erfordert jedoch, dass die Algorithmen mit dem NMEA-Nachrichtenformat, das im Kommunikationsnetz der Schiffe verwendet wird, umgehen können. In GALILEOnautic 2 wurden daher die Entwicklungen zur Integration des DVL in das Navigationsfilter fortgeführt, sodass die NMEA-Nachrichten des in der DENEb verbauten DVLs, bereitgestellt durch die Brückensoftware SYNAPSIS des Projektpartners Raytheon Anschütz genutzt werden können.

Um die vielen unterschiedlichen GNSS-Korrektur- und Navigationsfiltermodi, die mit den umfangreichen Entwicklungen einhergehen im automatisierten System zu managen, wurde ein Zustandsautomat entwickelt, der in Abhängigkeit verfügbarer Satelliten (-signale) und Korrekturinformationen zwischen Klobuchar-, Zweifrequenz- und differenzieller Korrektur wechseln kann. Bei Verfügbarkeit differenzieller Daten wird wiederum in abhängig verfügbarer bzw. nutzbarer Trägerphasen unterschieden zwischen Pseudorange-basiertem DGNSS, Trägerphasen-basiertem Real-Time-Kinematik (RTK)-float und RTK-fix.

Neben der robusten Implementierung des Navigationsfilters auf Echtzeithardware, wurde in GALILEOnautic 2 auch die Implementierung auf kostengünstiger Hardware u.a. für den mobilen Einsatz auf Lotsungen untersucht. In der Literatur wurden dazu bislang vor allem lose gekoppelte Fusionsalgorithmen implementiert. In GALILEOnautic 2 wurde daher die Implementierung des eng gekoppelten Navigationsfilters auf Massenmarkthardware, einem Raspberry Pi 4, untersucht.

2.2 Wissenschaftlich-technische Arbeiten

2.2.1 Verwendete Koordinatensysteme

Die Notation der Koordinatensysteme in diesem Abschlussbericht orientiert sich an den Notationen von [8], wobei zwischen drei Kategorien von Koordinatensystemen unterschieden wird. Das Objektkoordinatensystem wird benötigt, um die Zustands- oder Messgröße eines Objekts oder Körpers zu beschreiben. Das Referenzkoordinatensystem ist das Bezugssystem gegenüber dem die Zustands- oder Messgröße des Objektkoordinatensystems ausgedrückt werden soll. Das Darstellungskordinatensystem ist das Koordinatensystem, in dem eine betrachtete Zustands- oder Messgröße beschrieben wird. Die Notation einer dreidimensionalen Zustands- oder Messgröße \mathbf{a} folgt damit der eindeutigen Schreibweise:

$$\mathbf{a} \begin{matrix} \text{Darstellungskordinatensystem} \\ \text{Referenzkoordinatensystem} \text{ Objektkoordinatensystem} \end{matrix} \quad (2.1)$$

Abb. 2.1 visualisiert die vier in diesem Abschlussbericht genutzten Koordinatensysteme:

- (i) Earth-Centred-Inertial (ECI): erdzentriert, Achsen sind gegenüber Universum fix, x^i -Achse liegt in Äquatorialebene, z^i -Achse stimmt mit Rotationsachse der Erde überein
- (e) Earth-Centred-Earth-Fixed (ECEF): erdzentriert, x^e -Achse liegt in Äquatorialebene und schneidet den Nullmeridian, z^e -Achse stimmt mit Rotationsachse der Erde überein rotiert mit der Erde
- (b) Körper: mitbewegtes Koordinatensystem, x^b -Achse entlang des Fahrzeugs nach vorn ausgerichtet, z^b zeigt nach Richtung Erdmittelpunkt, y^b folgt Rechte-Hand-Regel
- (n) Navigation: mitbewegtes Koordinatensystem, teilt sich Ursprung mit Körperkoordinatensystem, wobei x^n -Achse stets Richtung geografischem Nordpol zeigt

In Abb. 2.1 ist das ECEF-Koordinatensystem in kartesischer $[x^e, y^e, z^e]$ und geodätischer Form mit Breitengrad ϕ^e , Längengrad λ^e und Höhe h^e dargestellt ist. Es werden die zusammenfallenden Ursprünge der erdzentrierten ECEF- und ECI-Koordinatensysteme sowie des mitbewegten Körper- und Navigationskoordinatensystems sichtbar. Über die wichtigsten Koordinatensysteme (i),(e),(b) und (n) hinaus sind noch das Antennenkoordinatensystem (a) und das Doppler Velocity Log (DVL)-Koordinatensystem (d) zu benennen. Das Antennenkoordinatensystem hat seinen Ursprung im Phasenzentrum der Antenne des GNSS-Empfängers. Die Orientierung der Achsen ist beliebig, da das System nicht zur Orientierungsbestimmung genutzt wird. Das Antennenkoordinatensystem dient allem voran als Objektkoordinatensystem. Der Ursprung des DVL-Koordinatensystems ist der Schnittpunkt der orthogonal auf den Schallköpfen stehenden Richtungsvektoren. Wie

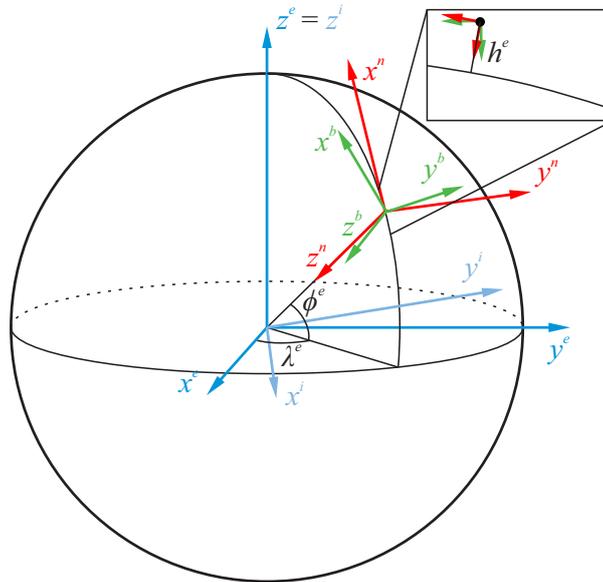


Abbildung 2.1: Koordinatensysteme: Darstellung ECI-Koordinatensystem (i), ECEF- Koordinatensystem (e) in kartesischen Koordinaten $[x^e, y^e, z^e]$ und geodätischen Koordinaten $[\phi^e, \lambda^e, h^e]$, Körperkoordinatensystem (b) und Navigationskoordinatensystem (n), [11].

beim Körperkoordinatensystem wird das DVL möglichst so im Fahrzeug platziert, dass die x^d -Achse entlang des Fahrzeugs nach vorne ausgerichtet ist und die z^d -Achse Richtung Fahrzeugboden nach unten zeigt. Weitere Details, u.a. zur Umrechnung der Systeme untereinander, sind in [11] zu finden.

2.2.2 Filterdesign

Abb. 2.2 zeigt ein vereinfachtes Blockschaltbild des Navigationsfilters. Das INS-basierte, eng gekoppelte Navigationsfilter schätzt die Navigationszustände dreidimensionale Position, -Geschwindigkeit und -Orientierung des Körperkoordinatensystems. \mathbf{p}_{eb}^e ist die dreidimensionale Position des Körperkoordinatensystems relativ zum ECEF-Koordinatensystem, ausgedrückt in geodätischen Koordinaten. \mathbf{v}_{eb}^n ist die dreidimensionale Geschwindigkeit des Körperkoordinatensystems relativ zum ECEF-Koordinatensystem, ausgedrückt im Navigationskoordinatensystem. Die Orientierungsquaternion beschreibt die Verdrehung zwischen Navigationskoordinatensystem und Körperkoordinatensystem und wird durch \mathbf{q}_b^n dargestellt.

Da die Berechnung der Navigationszustände auf den Messwerten einer IMU basiert, müssen zusätzlich der dreidimensionale Beschleunigungsbias \mathbf{b}_a und dreidimensionale Gyroskopbias \mathbf{b}_g geschätzt werden, um damit die gemessene dreidimensionale Beschleunigung

2 Maritimes hochgenaues Navigationsfilter

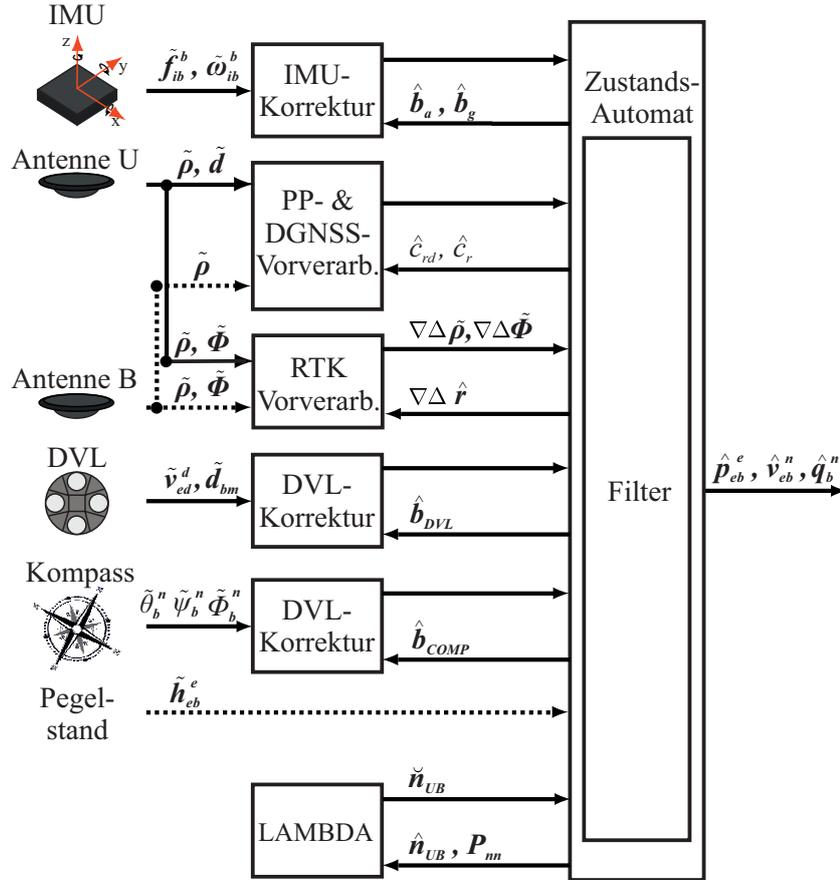


Abbildung 2.2: Blockschaltbild des maritimen Navigationsfilters, adaptiert von [11]

$\tilde{\mathbf{f}}_{ib}^b$ und dreidimensionale Drehrate $\tilde{\boldsymbol{\omega}}_{ib}^b$ der IMU zu korrigieren, bevor sie in den Strapdown-Algorithmus geführt werden. Da eine enge Kopplung von GNSS-Observationen durchgeführt wird, müssen darüber hinaus Empfängeruhrenbias c_r und -drift c_{rd} im Zustandsvektor geschätzt werden, um damit Pseudo- und Deltarange korrigieren zu können.

Es werden GNSS-Beobachtungen der Antenne U des Schiffs und der Antenne B einer Referenzstation verarbeitet. Der gestrichelte Signalpfeil der Referenzantenne zeigt an, dass die Daten der Antenne B über eine Internetverbindung empfangen werden. Die Point Positioning (PP)- und Differenzielle GNSS (DGNSS)-Verarbeitung wurden im Abschlussbericht von GALILEONautic [5] vorgestellt.

Zur Schätzung der Mehrdeutigkeit der Trägerphase wird der Zustandsraum der vorangegangenen Kapitel um den Vektor der doppel-differenzierten Float-Mehrdeutigkeiten \mathbf{n}_{UB} erweitert, wobei die Länge des Vektors von der Anzahl aktuell nutzbarer doppel-differenzierter Trägerphasen abhängt. Zur Realisierung von GPS-Galileo-Dualkonstellation RTK wird kein zusätzlicher Zustand zur Schätzung des Inter System Bias (ISB) der Pseudorange eingeführt. Details werden in [11] erläutert. Um keine Inter System Ambiguity (ISA) der Trägerphasen im Navigationsfilter schätzen zu müssen, wird jedes GNSS getrennt be-

handelt, sodass jeweils ein Referenzsatellit für GPS und ein zweiter Referenzsatellit für Galileo bestimmt wird. Dies hat zur Folge, dass im Zustandsraum zwischen Zuständen und Messgrößen von GPS und Galileo unterschieden wird. Es ergibt sich der vollständige Zustandsvektor des Navigationsfilters:

$$\mathbf{x} = [\mathbf{p}_{eb}^e \ \mathbf{v}_{eb}^n \ \mathbf{q}_b^n \ \mathbf{b}_a \ \mathbf{b}_g \ c_b \ c_d \ \hat{\mathbf{b}}_{DVL} \ \hat{\mathbf{b}}_{COMP} \ \mathbf{n}_{UB}^{GPS} \ \mathbf{n}_{UB}^{Galileo}]^T . \quad (2.2)$$

\mathbf{n}_{UB}^{GPS} ist der Vektor der über die Empfänger U und B doppeldifferenzierten GPS-Trägerphasen-Mehrdeutigkeiten und $\mathbf{n}_{UB}^{Galileo}$ analog der Vektor der über die Empfänger U und B doppeldifferenzierten Galileo-Trägerphasen-Mehrdeutigkeiten.

Es wird angenommen, dass die gesuchte Mehrdeutigkeit \mathbf{n}_{UB} ganzzahlig ist. Somit können Trägerphasen von gerade erst akquirierten Satelliten, die mit einer halbzyklischen Mehrdeutigkeit beaufschlagt sind, nicht in die Float- und sich anschließende Fix-Schätzung einbezogen werden.

Die PP-Vorverarbeitung ermöglicht eine GNSS-gestützte Zustandsschätzung, auch wenn keine differenziellen Daten verfügbar sind. Die PP- und DGNSS-Vorverarbeitung werden getrennt von der Real Time Kinematic (RTK)-Vorverarbeitung durchgeführt. Hauptaufgabe der (RTK)-Vorverarbeitung ist die Schätzung der GNSS-Observationen der Referenzantenne B , das Bilden von Wide Lane (WL)-Kombinationen, die exakte Synchronisation der Messepochen von U und B , das Bilden der Doppeldifferenzen der Pseudorange $\nabla\Delta\rho$ und Trägerphasen $\nabla\Delta\Phi$ sowie die Detektion von Cycle-Slips und weiterer Unregelmäßigkeiten der Trägerphase. Der Cycle-Slip-Detektor basiert auf den Arbeiten von [12] und [13] und wird in diesem Abschlussbericht nicht weiter diskutiert.

Die Beschriftung des Signalflusspfeils der DVL-Komponente im Blockschaltbild gibt an, dass sowohl die DVL-Geschwindigkeit $\tilde{\mathbf{v}}_{ed}^d$ als auch die aus der Doppler-Verschiebung abgeleitete Relativgeschwindigkeit $\tilde{\mathbf{d}}_{bm}$ verarbeitet werden kann. Bei systematisch fehlerhaften DVL-Messungen wird der Zustandsvektor des Filters um Bias-Zustände \mathbf{b}_{DVL} des DVL erweitert. Die DVL-Messungen werden dann durch $\hat{\mathbf{b}}_{DVL}$ korrigiert.

Pegelstanddaten werden über die gleiche Internetverbindung empfangen wie die differenziellen Daten der Antenne B (angedeutet durch den gestrichelten Signalpfeil). Die Umrechnung eines Pegel in Metern über Normalhöhennull (m. ü. NHN) mit Bezugspegel Amsterdam in das World Geodetic System 1984 (WGS84)-System zur Generierung der für das Kalman-Update erforderlichen Messung \tilde{h}_{eb}^e wurde in [14] implementiert und wird hier nicht weiter diskutiert.

Vom Kompass werden die Eulerwinkel in Gier-, Nick- und Rollrichtung als Messgrößen für das Filter geliefert. Bei systematisch fehlerhaften Kompass-Messungen wird der Zustandsvektor des Filters um Bias-Zustände \mathbf{b}_{COMP} des Kompass erweitert, wobei die Kompass-Messungen dann durch $\hat{\mathbf{b}}_{COMP}$ korrigiert werden.

2 Maritimes hochgenaues Navigationsfilter

Die Float-Schätzung \mathbf{n}_{UB} des Navigationsfilters wird zusammen mit der geschätzten Kovarianz \mathbf{P}_{nn} in das LAMBDA-Verfahren eingespeist. Daraus hervorgehende Kandidaten der Fix-Mehrdeutigkeit $\check{\mathbf{n}}_{UB}$ werden an einen Zustandsautomaten übergeben.

Der Zustandsautomat bestimmt einen diskreten Filtermodus. Dieser kann entweder im PP/DGNSS-Modus, im Float-Modus, im Fix-Modus oder in einer Mischung aus Float und Fix arbeiten. Die Transitionen zwischen den diskreten Zuständen des Automaten schalten in Abhängigkeit von der Verfügbarkeit von Doppeldifferenzen und der Qualität des Fix-Kandidaten aus dem LAMBDA-Verfahren.

Zur Implementierung des LAMBDA-Verfahrens wurde die LAMBDA-Toolbox [15] genutzt und lediglich für die Echtzeit-Speicherallokation leicht angepasst. Details zur Implementierung des Zustandsautomaten sind in [11] zu finden und werden hier nicht erläutert.

2.2.3 Sensordaten-Vorverarbeitung

Optimierte Satellitenkonstellation

Die Auswirkung der Satellitengeometrie auf die Navigationslösung wird mithilfe des Konzepts der Dilution of Precision (DOP) quantifiziert [8, pp. 424-429]. Die DOP wird dann verwendet, um die Unsicherheit der verschiedenen Teile der Navigationslösung mit der Pseudorange-Unsicherheit in Beziehung zu setzen. Im Allgemeinen hängt die Berechnung der DOP nur von der Anzahl der sichtbaren Satelliten und ihrer geometrischen Konstellation ab. Die detaillierte Berechnung ist [8, pp. 424-429] zu finden.

Die Verwendung der DOP basiert auf der Annahme, dass alle Satellitenmessungen die gleiche Qualität haben, was in der Realität nicht immer der Fall ist. Daher wird die gewichtete DOP (WDOP) eingeführt, die die geometrische Satellitenkonstellation und die Kovarianzmatrix der Messungen kombiniert, um den Positionierungsfehler vorherzusagen, dessen Berechnung in [16] zu finden ist.

In der vorliegenden Arbeit werden DOP und WDOP als Qualitätsindex verwendet, um die optimale Teilmenge aller verfügbaren Satelliten auszuwählen, die sowohl die beste Satellitenkonstellation als auch die beste Messqualität erzielt, um den Positionsfehler des Navigationsfilters zu minimieren.

Die Auswahl der optimalen Teilmenge erfolgt durch Brute-Force, d. h. DOP und WDOP aller möglichen Teilmengen der Satelliten werden berechnet und die Teilmenge mit dem kleinsten DOP oder WDOP wird ausgewählt. Dies wurde in einem Echtzeittest validiert, dessen Ergebnis in Abschnitt 2.3 dokumentiert ist.

Differenzielles GALILEO und differentielle GPS-Galileo Dualkonstellation

Mit Hilfe differenzieller Korrekturen können die Pseudoranges für die Dualkonstellation von GPS und Galileo bzgl. ihres Ionosphärenfehlers $\frac{f_2}{f_1} I_r$ und des Troposphärenfehlers T_r korrigiert werden. Hierfür werden die RTCM 3.3 Korrekturdaten einer virtuellen Referenzstation (VRS) des SAPOS Dienstes genutzt. Über einen NTRIP Client werden die Daten vom Server über eine Mobilfunkverbindung bezogen.

Zur Korrektur einer Pseudorange wird ein Offset $\delta\rho_B$ für jeden Satelliten berechnet. Dieser basiert auf der Differenz der Euklidischen Distanz zwischen einem Satelliten (Index S) und der Basisstation (Index B) sowie der gemessenen Pseudorange an der Basisstation $\tilde{\rho}_B$:

$$\delta\rho_B = \|\mathbf{p}_{e,S}^e - \mathbf{p}_{e,B}^e\|_2 - \tilde{\rho}_B, \quad (2.3)$$

wobei $\mathbf{p}_{e,B}^e$ die bekannte Position der Basisstation und $\mathbf{p}_{e,S}^e$ die bekannte Position des Satelliten darstellen. Beide sind ECEF-Koordinaten dargestellt.

Unter der Annahme langsam variierender Fehlerterme wird der Pseudorange-Offset $\delta\rho_B$ von der zugehörigen Pseudorange-Messung des Users $\tilde{\rho}_U$ subtrahiert. Nachdem die Pseudoranges von GPS und GAL individuell korrigiert wurden, muss noch der Inter-Systembias (ISB) mittels der GGTO kompensiert werden. Details sind in [17] zu finden. Nach der Kompensation des ISB werden die Observablen von GPS und Galileo in einem gemeinsamen Messvektor \mathbf{z} zusammengeführt.

GPS-Galileo Dualkonstellation Real Time Kinematik

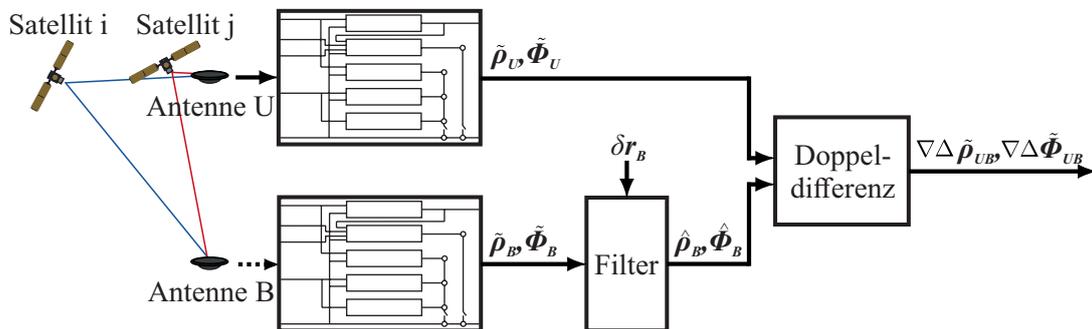


Abbildung 2.3: RTK-Vorverarbeitung mit Filter zur Schätzung der GNSS-Beobachtungsdaten von Empfänger B , aus [11]

Abb. 2.3 zeigt eine vereinfachte Darstellung des Signalpfades der RTK-Vorverarbeitung. Zunächst werden die GNSS-Beobachtungen von Antenne U und Referenzantenne B auf unabhängigen Pfaden vorverarbeitet. Der örtlich separierte Empfänger B überträgt seine

2 Maritimes hochgenaues Navigationsfilter

Messepoche über eine Mobilfunkverbindung, angedeutet durch den gestrichelten Signalpfeil.

Zur Vorverarbeitung wird jeweils eine Instanz der PP-/DGNSS-Vorverarbeitung aus [5] verwendet, bei der die Satellitenposition aus den Ephemeriden berechnet wird und Korrekturen an den Beobachtungen vorgenommen werden können. Wenn sowohl L1- als auch L2C-Signale zur Verfügung stehen und keine Galileo-GPS RTK-Dualkonstellation genutzt werden soll, wird hier zusätzlich eine GPS-WL-Kombination gebildet. Galileo-WL Kombinationen zusammen mit GPS-WL wurde im Projekt nicht untersucht. Da zur Bestimmung der Mehrdeutigkeiten im Float- und Fix-Modus des Navigationsfilters GPS und Galileo getrennt behandelt werden, darf in den beiden Instanzen der PP-DGNSS-Vorverarbeitung GPS und Galileo nicht, wie im vorherigen Abschnitt erläutert, synchronisiert werden. Daher wird in der RTK-Vorverarbeitung die Korrektur der Pseudorange des zusätzlichen GNSS mittels GGTO nicht durchgeführt. Somit werden asynchrone Observations von GPS und Galileo in Signalrichtung im sich anschließenden Filter, der Synchronisation und Doppeldifferenzbildung geführt.

Die vorverarbeiteten GNSS-Beobachtungen der Referenzantenne B werden an ein Kalman-Filter weitergeleitet, das sie als Messwerte für den Kalman-Aktualisierungsschritt verwendet. Das Kalman-Filter propagiert die Pseudorange ρ_B und die Trägerphase Φ_B anhand der zeitlichen Veränderung des geometrischen Abstands (Entfernungsrates) δr_B zwischen der Referenzantenne B und den verfügbaren Satelliten. Das Prozessmodell des Filters wird mit der Frequenz der Beobachtungen des Empfängers U ausgeführt, so dass eine Synchronisation der Beobachtungen von U und B inherent gegeben ist. Es folgt eine Doppeldifferenzierung. Formulierungen zur Bildung von Doppeldifferenzen sind bspw. in [8] zu finden. Das Filter zur Schätzung der differenziellen Daten wird im Folgenden erläutert.

2.2.4 Schätzung von differenziellen Daten

Zustandsraum

Da sich die Referenzantenne B selbst nicht bewegt, kann die Vorhersage der Änderung der Pseudorange ρ_B und der Trägerphasen Φ_B im Prinzip nur auf der Änderung der Satellitenpositionen beruhen. Die Satellitenpositionen können für jede beliebige Wochenzeit (TOW) und Pseudorange bei bekannten Ephemeriden bestimmt werden. Tatsächlich ändern sich aber die GNSS-Beobachtungen nicht nur aufgrund der Satellitenbewegung, sondern auch aufgrund von sich ändernden ionosphärischen und troposphärischen Fehlern der Signale sowie von sich zeitlich ändernden Satelliten- und Empfängeruhrfehlern. Daher wird ein lineares zeitdiskretes Kalman-Filter (KF) mit dem folgenden Zustandsvektor

entworfen:

$$\mathbf{x}_B^i = [\rho_B^i \ \Phi_B^i \ E_d^i \ I_d^i]^T, \quad (2.4)$$

wobei der hochgestellte Index i anzeigt, dass es sich um den Zustandsvektor des Satelliten i handelt. E_d^i ist der Entfernungsratenfehler in $\frac{m}{sec}$, der sowohl für die Pseudorange als auch für die Trägerphase positiv ist und die zeitliche Veränderung der Troposphäre und der Satelliten- und Empfängeruhrfehler einschließt. I_d^i ist die zeitliche Variation des Ionosphärenfehlers, der getrennt von E_d^i betrachtet werden muss, da er sich positiv auf die Pseudorange und negativ auf die Trägerphase auswirkt. Für jeden Satelliten i , einer Messepoche von B , wird eine eigene Instanz des Filters gebildet.

Das zeitdiskrete lineare Prozessmodell des Zustandsraums für den Zustandsvektor, Gl. 2.4, ist:

$$\underbrace{\begin{pmatrix} \rho_{B,k}^i \\ \Phi_{B,k}^i \\ E_{d,k}^i \\ I_{d,k}^i \end{pmatrix}}_{\mathbf{x}_{B,k}^i} = \underbrace{\begin{pmatrix} \rho_{B,k-1}^i + \delta r_B^i + T_s \cdot (E_{d,k-1}^i + I_{d,k-1}^i) \\ \rho_{B,k-1}^i + \delta r_B^i + T_s \cdot (E_{d,k-1}^i - I_{d,k-1}^i) \\ E_{d,k-1}^i \\ I_{d,k-1}^i \end{pmatrix}}_{\mathbf{f}_k(\mathbf{x}_{k-1}, u_{k-1})} + \mathbf{w}_{k,B} \quad \mathbf{w}_B \sim \mathcal{N}(0, \mathbf{Q}_B). \quad (2.5)$$

T_s ist die Zeit zwischen zwei Epochen $k-1$ und k . Die zeitdiskrete Entfernungsrates δr_B^i beschreibt die Änderung der Pseudorange ρ_B^i und der Trägerphase Φ_B^i zwischen $k-1$ und k aufgrund der Änderung der Satellitenposition. Sie ist genau bestimmt durch die Differenz des euklidischen Abstandes zwischen der Position der Referenzantenne B und der Position des Satelliten i zwischen der vergangenen Epoche TOW_{k-1} und der aktuellen Epoche TOW_k :

$$\delta r_B^i = \left(\|\mathbf{p}_{ea,B}^s - \mathbf{p}_{es}^{e,i}(TOW_k)\| - \|\mathbf{p}_{ea,B}^s - \mathbf{p}_{es}^{e,i}(TOW_{k-1})\| \right). \quad (2.6)$$

Die tatsächliche Dynamik der Entfernungsratenfehler E_d^i und I_d^i ist unbekannt. Daher wird ihre Veränderung als eine Folge von weißem Rauschen (Random Walk) modelliert. \mathbf{w}_B wird als normalverteiltes mittelwertfreies Prozessrauschen mit der Kovarianz \mathbf{Q}_B modelliert und stellt die Modellunsicherheiten dar. Da δr_B aus der Satellitenposition berechnet wird und daher von der Variable TOW und der Pseudorange von U abhängt, kann das Prozessmodell immer dann ausgeführt werden, wenn eine Messepoche von U verfügbar ist. Für jede Messepoche von U steht also eine (geschätzte) Messepoche von B zur Verfügung, so dass die kontinuierliche, synchrone Bildung von Doppeldifferenzen möglich ist. T_s entspricht also der Ausgangsfrequenz des Empfängers U und beträgt im Anwendungsbeispiel dieser Arbeit 10 Hz.

2 Maritimes hochgenaues Navigationsfilter

Das Messmodell lautet:

$$\mathbf{z}_i = \underbrace{\begin{pmatrix} \rho_{B,k}^i \\ \Phi_{B,k}^i \end{pmatrix}}_{\mathbf{h}(x_k)} + \mathbf{v}_{k,B} \quad \mathbf{v}_B \sim \mathcal{N}(0, \mathbf{R}_B). \quad (2.7)$$

\mathbf{v}_B wird als normalverteiltes, mittelwertfreies Messrauschen mit der Kovarianz \mathbf{R}_B modelliert. Trifft eine Messepoche von B ein, wird zunächst das Alter der Messepoche im Vergleich zur aktuellen Messepoche von U durch Vergleich der jeweiligen TOWs bestimmt. Der Aktualisierungsschritt mit dem Messmodell Eq. 2.7 wird dann für den entsprechenden Zustand in der Vergangenheit durchgeführt. Gespeicherte zeitdiskrete Entfernungsraten werden anschließend zur Vorhersage des Zustands zum aktuellen TOW von U durch unkorrigierte Vorwärtsprädiktion verwendet.

Ausgabekriterium

Um zu entscheiden, ob die geschätzte Pseudorange und die Trägerphase eine ausreichende Genauigkeit für die weitere Verarbeitung in der Doppeldifferenzbildung und später im Navigationsfilter aufweisen, wird die Norm der Innovation \mathbf{y}_B bei jedem Schritt k überprüft:

$$y_{B,k}^i = \|\tilde{\mathbf{z}}_{B,k}^i - [\hat{\rho}_{B,k}^{i-} \hat{\Phi}_{B,k}^{i-}]^T\|_2 \quad (2.8)$$

Dabei sind $\hat{\rho}_{B,k}^{i-}$ und $\hat{\Phi}_{B,k}^{i-}$ die A-priori-Zustände und $\tilde{\mathbf{z}}_{B,k}^i$ ist die zugehörige Messung der B -Messepoche. Fällt $y_{B,k}^i$ unter einen vordefinierten Schwellenwert t_y , wird der Satellit i zur weiteren Verarbeitung zugelassen. Der Schwellenwert t_y wird gleich 0,5 m gewählt.

Filterparametrierung

Tabelle 2.1: Räumliche und zeitliche Korrelation der Common-Mode Fehler [8, S. 437].

Source	Variation über 100 Sek.	Variation über 100 km
Satellite clock	0.1 m	-
Ephemeris	0.01 m	0.002 m
Ionosphere	0.1-0.4 m	0.2-0.5 m
Troposphere	0.1-1.5 m	0.1-1.5 m

Die Unsicherheiten des Prozessmodells werden anhand der räumlichen und zeitlichen Variation der GNSS-Common-Mode Fehler in Tab. 2.1 bestimmt. Die Grundlage für

die Gewichtung des Rauschens der Prozessgleichungen untereinander sind die dort angegebenen Variationen über 100 s. Das Rauschen des Prozessmodells der Pseudorange und Trägerphase wird durch die Ephemeridenfehler bestimmt. Das Rauschen der Entfernungsratenfehlerschätzung E_d^i wird dagegen durch die Unsicherheiten der Troposphären-, Satellitenuhr- und Empfängeruhrfehler bestimmt. Über Tab. 2.1 wird für die Empfängeruhr der Referenzstation konservativ eine Variation von 0,5 m über 100 s geschätzt. Das Rauschen der Änderung des ionosphärischen Fehlers I_d^i wird ebenfalls anhand von Tab. 2.1 bestimmt. Somit kann die Gesamtvarianz \mathbf{Q}_B des Prozessmodells geschätzt werden:

$$\mathbf{Q}_{B,k} = \text{diag} \left[(0.01 \text{ m})^2, (0.01 \text{ m})^2, (1.5 \text{ m} + 0.1 \text{ m} + 0.5 \text{ m})^2, (0.4 \text{ m})^2 \right]. \quad (2.9)$$

Das Rauschen der Messung der Kovarianzmatrix \mathbf{R}_B wird entsprechend den erwarteten Genauigkeiten des Signaltrackings eines GNSS-Empfängers aus [18] geschätzt. Das Rauschen der GPS-L1-, bzw. Galileo-E1-Pseudoranges wird statisch auf 3 m gesetzt, das der Trägerphasen auf ein Prozent eines (Wide-Lane-) Zyklus:

$$\mathbf{R}_B = \text{diag}[(3 \text{ m})^2, (0.008 \text{ m})^2]. \quad (2.10)$$

2.2.5 Zustandsraummodell Navigationsfilter

Das folgende Kapitel basiert auf [11]. Detaillierte Erläuterungen zum Zustandsraummodell sind daher dort zu finden. Das Navigationsfilter ist als Unscented Kalman-Filter (UKF) implementiert. Die UKF-Implementierung wurde in [19] beschrieben. Im Navigationsfilter wird ein nichtlineares zeitdiskretes Zustandsraummodell der folgenden Form verwendet:

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{G}_k \mathbf{w}_k \quad \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k), \quad (2.11)$$

wobei k der Abtastschritt ist. \mathbf{f}_k ist die nichtlineare, zeitdiskrete Zustandsübergangsfunktion. \mathbf{x} ist der Zustandsvektor und \mathbf{u} ist der Vektor der Eingänge. Die Matrix \mathbf{G} wird als Shapematrix bezeichnet und bildet das Prozessrauschen \mathbf{w} auf den Zustandsvektor \mathbf{x} ab. Die mehrdimensionale Normalverteilung von \mathbf{w} ist mittelwertfrei mit Kovarianz \mathbf{Q} , wobei \mathbf{Q} die Kovarianzmatrix des Prozessrauschens ist. Die Abbildung der Zustände \mathbf{x} auf die Messgrößen wird im Messmodell formuliert:

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{v} \quad \mathbf{v} \sim \mathcal{N}(0, \mathbf{R}), \quad (2.12)$$

wobei \mathbf{z} der Vektor der Messungen, \mathbf{h} das nichtlineare, zeitinvariante Messmodell und \mathbf{v}_k das Messrauschen ist. Die mehrdimensionale Normalverteilung von \mathbf{v} ist mittelwertfrei mit Kovarianz \mathbf{R} , wobei \mathbf{R} die Kovarianzmatrix des Messrauschens ist.

Prozessmodell

Das Prozessmodell beschreibt die Änderung des Zustands des Navigationsfilters $\mathbf{x} = [\mathbf{p}_{eb}^e \ \mathbf{v}_{eb}^n \ \mathbf{q}_b^n \ \mathbf{b}_a \ \mathbf{b}_g \ c_b \ c_d \ \hat{\mathbf{b}}_{DVL} \ \mathbf{n}_{UB}^{GPS} \ \mathbf{n}_{UB}^{Galileo}]^T$ in Form eines Gleichungssystems aus Differentialgleichungen, wobei die Beschreibung der Änderung der Bewegungszustände Position, Geschwindigkeit und Orientierung mittels Strapdown formuliert wird, sh. z.B. [11].

IMU-Bias als auch Empfängeruhrenbias werden als zeitkorrelierte Fehler interpretiert. Dieser systematische Fehler wird gemäß des Vorschlags von [20, S. 200] in einem einfachen Modell umgesetzt:

$$\begin{pmatrix} \dot{c}_r \\ \dot{c}_{rd} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} c_r \\ c_{rd} \end{pmatrix} + \begin{pmatrix} w_r \\ w_{rd} \end{pmatrix}. \quad (2.13)$$

Hierbei wird die Uhrendrift als Integral weißen Rauschens, also als sogenannter Random-Walk modelliert. Das Rauschen des Empfängeruhrenfehlers w_r und das der Empfängeruhrendrift w_{rd} werden als normalverteiltes, mittelwertfreies Rauschen modelliert und in \mathbf{w}_r zusammengefasst.

Die Änderung des Beschleunigungs- und Gyroskop-Bias \mathbf{b}_a und \mathbf{b}_g wird durch die Bias-Instabilität geprägt und wird gemäß [20, S. 200] vereinfacht als Random-Walk beschrieben:

$$\dot{\mathbf{b}}_a = \mathbf{w}_{b_a}, \quad (2.14)$$

$$\dot{\mathbf{b}}_g = \mathbf{w}_{b_g}. \quad (2.15)$$

Dabei ist \mathbf{w}_{b_a} die Beschleunigungssensorbias-Instabilität und wird als normalverteiltes, mittelwertfreies Rauschen modelliert. \mathbf{w}_{b_g} ist das entsprechende Pendant des Gyroskops.

Dass das Forschungsschiff DENEb nur die NMEA-VBW-Nachricht ihres verbauten DVLs zur Verfügung stellt, ist mit erheblichen Nachteilen verbunden:

- Keine Geschwindigkeit in z-Richtung des DVL-Körperkoordinatensystems
- Schlechtere Qualität der Geschwindigkeitsmessung
- Maximale Geschwindigkeitsauflösung von 0.1 kn
- Keine Möglichkeit, die DVL-Messung mit einem GNSS-PPS zu triggern, um die Messverzögerung zu ermitteln
- Keine Informationen über die Messqualität

Es wird daher ein Zustand \mathbf{b}_{DVL} eingeführt der mögliche Messfehler des DVLs beschreibt. Der Geschwindigkeitsmessfehler \mathbf{b}_{DVL} der Geschwindigkeitskomponenten des DVL wird

als mehrdimensionaler zeitkorrelierter Bias modelliert. Da die tatsächliche Charakteristik bzw. Zeitkonstante aber unbekannt ist, wird keine Gauß-Markov-Sequenz, sondern ein Random-Walk-Prozess zur Modellierung gewählt.

Die als Float approximierten Mehrdeutigkeiten \mathbf{n}_{UB}^{GPS} und $\mathbf{n}_{UB}^{Galileo}$ werden als Random-Walk-Prozess mit normalverteiltem mittelwertfreiem Rauschen \mathbf{w}_n modelliert, da sie maximal zeitkorreliert sind und ihre Korrelationszeit somit gegen unendlich tendiert. Die Länge der Zustandsvektoren ist n_{CP}^{GPS} bzw. $n_{CP}^{Galileo}$, variabel und hängt von der Verfügbarkeit nutzbarer Trägerphasen des GPS- bzw. Galileo-GNSS ab.

Zusammen mit dem Prozessmodell des Navigationsfilters mit Strapdown-Algorithmus und Uhrenfehlermodell, das bereits in GALILEOnautic entwickelt wurde, ergibt sich folgendes vollständiges Prozessmodell:

$$\underbrace{\begin{pmatrix} \dot{\mathbf{p}}_{eb}^e \\ \dot{\mathbf{v}}_{eb}^n \\ \dot{\mathbf{q}}_b^n \\ \dot{\mathbf{b}}_a \\ \dot{\mathbf{b}}_g \\ \dot{c}_r \\ \dot{c}_{rd} \\ \dot{\mathbf{b}}_{DVL} \\ \dot{\mathbf{b}}_{COMP} \\ \dot{\mathbf{n}}_{UB}^{GPS} \\ \dot{\mathbf{n}}_{UB}^{Galileo} \end{pmatrix}}_{\mathbf{x}} = \underbrace{\begin{pmatrix} \mathbf{C}_n^e \mathbf{v}_{eb}^n \\ \mathbf{C}_b^n (\tilde{\mathbf{f}}_{ib}^b - \mathbf{b}_a) - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \mathbf{v}_{eb}^n + \mathbf{g}_l^n \\ \frac{1}{2}\boldsymbol{\Omega} \left(0, [(\tilde{\boldsymbol{\omega}}_{ib}^b - \mathbf{b}_g) - \mathbf{C}_n^b (\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n)] \right) \mathbf{q}_b^n \\ 0 \\ 0 \\ c_{rd} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}}_{\mathbf{f}(\mathbf{x}, \mathbf{u})} + \mathbf{G} \underbrace{\begin{pmatrix} \mathbf{w}_a \\ \mathbf{w}_g \\ \mathbf{w}_{b_a} \\ \mathbf{w}_{b_g} \\ \mathbf{w}_r \\ \mathbf{w}_{DVL} \\ \mathbf{w}_{COMP} \\ \mathbf{w}_{nGPS} \\ \mathbf{w}_{nGalileo} \end{pmatrix}}_{\mathbf{w}}. \quad (2.16)$$

Die Matrix \mathbf{C}_n^e ist eine Kosinusrichtungsmatrix und beschreibt die Rotation vom Koordinatensystem (n) in das Koordinatensystem (e). \mathbf{C}_b^n ist ebenfalls eine Kosinusrichtungsmatrix und beschreibt die Drehung von (b) nach (n) usw. Die Änderung der Geschwindigkeit wird durch die lokale Gravitation \mathbf{g}_l^n korrigiert. Die Orientierung im Zustandsvektor wird als Quaternion \mathbf{q}_b^n formuliert. $\boldsymbol{\Omega}$ ist der kreuzproduktbildender Operator, sh. bspw. [11]. Die dritte Gleichung beschreibt demnach die Propagation der Orientierungsquaternion, durch Quaternionmultiplikation mit der um den Bias \mathbf{b}_g , die Erddrehrate $\boldsymbol{\omega}_{ie}^n$ und die Transportrate $\boldsymbol{\omega}_{en}^n$ korrigierte von der IMU gemessene Drehrate $\tilde{\boldsymbol{\omega}}_{ib}^b$.

2 Maritimes hochgenaues Navigationsfilter

Für die Shapematrix ergibt sich:

$$\mathbf{G} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{C}_b^n & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_b^n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{3 \times 3} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{3 \times 3} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{2 \times 2} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{3 \times 3} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{3 \times 3} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n_{CP}^{GPS} \times n_{CP}^{GPS}} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n_{CP}^{Galileo} \times n_{CP}^{Galileo}} \end{pmatrix} \quad (2.17)$$

Messmodelle einfache Pseudorange und Deltorange

Die Messmodelle zur engen Kopplung von Pseudorange und Deltorange wurden in GALILEOnautic entwickelt und dem dazugehörigen Abschlussbericht [5] und den eigenen Publikationen [21], [22], [23] sowie [11] vorgestellt.

Messmodelle doppeldifferenzierte Pseudorange und Trägerphase

Das Pseudorange-Messmodell berechnet vom Navigationsfilter geschätzte doppeldifferenzierte Entfernungsmessungen $\nabla \Delta \hat{r}$ für den anschließenden Vergleich mit der tatsächlichen Pseudorangemessung $\nabla \Delta \tilde{\rho}_{UB}^{ij}$ im Kalman-Aktualisierungsschritt. Das Modell der Pseudorangemessung in Form der algebraischen Gleichungen des Zustandsraums, Gl. 2.12, in Abhängigkeit von der aktuellen Zustandsschätzung \mathbf{x} für die Satelliten i und j der Antenne U und B ist:

$$\underbrace{\nabla \Delta \rho_{UB}^{ij}}_z = \underbrace{\nabla \Delta r_{UB}^{ij}}_{h(\mathbf{x})} + v_{\nabla \Delta \Phi}^{ij}, \quad (2.18)$$

mit:

$$\nabla \Delta r_{UB}^{ij} = \|\mathbf{p}_{es}^{e,i} - \mathbf{p}_{ea,U}^e\|_2 - \|\mathbf{p}_{es}^{e,i} - \mathbf{p}_{ea,B}^e\|_2 - \left(\|\mathbf{p}_{es}^{e,j} - \mathbf{p}_{ea,U}^e\|_2 - \|\mathbf{p}_{es}^{e,j} - \mathbf{p}_{ea,B}^e\|_2 \right). \quad (2.19)$$

Die Position der Referenzantenne B $\mathbf{p}_{ea,B}^e$ ist aus den über eine Mobilfunkverbindung übertragenen differenziellen Daten im RTCM-Format bekannt. Die Position der Antenne U

$\mathbf{p}_{ea,U}^e$ wird durch Transformation der geschätzten Position des Körperkoordinatensystems \mathbf{p}_{eb}^e unter Verwendung des Orientierungszustands \mathbf{q}_b^n und des Hebelarms \mathbf{l}_{ab}^b berechnet:

$$\mathbf{p}_{ea}^e = \mathbf{p}_{eb}^e + \mathbf{C}_e^n \mathbf{C}_b^n \mathbf{l}_{ba}^b. \quad (2.20)$$

Der Hebelarm \mathbf{l}_{ba}^b zeigt vom Ursprung des Körperkoordinatensystems (b) zum Antennenkoordinatensystem (a) und wird in den Koordinaten des Körperkoordinatensystems ausgedrückt. \mathbf{C}_e^n ist die Kosinusrichtungsmatrix, die einen Vektor vom Navigationskoordinatensystem (n) zum ECEF-Koordinatensystem (e) dreht und eine Funktion des Positionszustands \mathbf{p}_{eb}^e ist. \mathbf{C}_b^n ist die Kosinusrichtungsmatrix, die einen Vektor vom Körperkoordinatensystem zum Navigationkoordinatensystem dreht und eine Funktion des Orientierungszustands \mathbf{q}_n^b ist.

Das Trägerphasen-Messmodell berechnet die vom Navigationsfilter geschätzte doppel-differenzierte Trägerphase $\nabla\Delta\hat{\Phi}_{UB}^{ij}$ zum anschließenden Vergleich mit der tatsächlichen Messung $\nabla\Delta\tilde{\Phi}_{UB}^{ij}$ im Kalman-Aktualisierungsschritt. Das Messmodell $\nabla\Delta\Phi$ in Form der algebraischen Gleichungen des Zustandsraums in Abhängigkeit von der aktuellen Zustands-schätzung \mathbf{x} für die Satelliten i und j der Antenne U und B ist:

$$\underbrace{\nabla\Delta\Phi_{UB}^{ij}}_z = \underbrace{\nabla\Delta r_{UB}^{ij} - \lambda\nabla\Delta N_{UB}^{ij}}_{h(\mathbf{x})} + v_{\nabla\Delta\Phi}^{ij}, \quad (2.21)$$

wobei $\nabla\Delta N_{UB}^{ij}$ der Zustand der doppel-differenzierten Float-Mehrdeutigkeit für die Satelliten i und j der Antenne U und B ist und dem Zustandsvektor \mathbf{x} entnommen wird. Die verbleibenden doppel-differenzierten Modellfehler des differenziellen Ansatzes werden in $v_{\nabla\Delta\Phi}^{ij}$ zusammengefasst, das als normalverteiltes Rauschen mit Nullmittelwert gemäß Gl. 2.12 formuliert wird.

Messmodell DVL

Für die Implementierung des DENEb-DVL in das bestehende Navigationsfilter wird angenommen, dass die fehlende Geschwindigkeit in z -Richtung des Körperkoordinatensystems gleich Null ist; x - und y -Geschwindigkeit werden direkt aus der NMEA VBW-Nachricht übernommen. Die resultierende Messung ist:

$$\tilde{\mathbf{z}}_{DVL} = \begin{pmatrix} \text{VBW: Longitudinal ground speed} \\ \text{VBW: Transverse ground speed} \\ 0 \end{pmatrix} \quad (2.22)$$

Das DVL-Messmodell zur losen Kopplung mit DVL-Biaszustand errechnet die vom Navigationsfilter geschätzte DVL-Geschwindigkeit $\hat{\mathbf{v}}_{ed}^d$ zum anschließenden Vergleich mit der

2 Maritimes hochgenaues Navigationsfilter

tatsächlich gemessenen $\tilde{\mathbf{v}}_{ed}^d$ im Kalman-Aktualisierungsschritt. In Form der algebraischen Gleichungen des Zustandsraums, Gl. 2.12, wird die DVL-Geschwindigkeit auf Grundlage der aktuellen Zustandsschätzung \mathbf{x} wie folgt bestimmt:

$$\underbrace{\mathbf{v}_{ed}^d}_z = \underbrace{\mathbf{C}_b^d (\mathbf{C}_n^b \mathbf{v}_{eb}^n + \mathbf{w}_{ib}^b \times \mathbf{l}_{bd}^b)}_{h(\mathbf{x})} + \mathbf{b}_{DVL} + \mathbf{v}_{DVL}. \quad (2.23)$$

Messmodell Kompass

Die Litef IMU stellt Messungen des Yaw-, Pitch- und Roll-Winkel zur Verfügung. Im EKF wird die Orientierung in Quaternionen \mathbf{q}_b^n geschätzt. Entsprechend wird zur Integration der Litef Messungen zunächst eine Konvertierung der Orientierungsschätzung in Eulerwinkel gemäß der ZYX Konvention vorgenommen. Dies führt zu den geschätzten Yaw-, Pitch- und Rollwinkeln ψ_b^n , θ_b^n und ϕ_b^n . Diese Umrechnung ist Mehrdeutig, z. B. gilt: $\sin(\psi_b^n) = \sin(\psi_b^n + n \cdot 360^\circ)$. Die Mehrdeutigkeit wird aufgelöst, indem n so gewählt wird, dass die Differenz zwischen geschätzten und gemessenen Winkel betragsmäßig minimal wird.

Mit den so geschätzten Eulenkeln lautet das Messmodell:

$$\underbrace{\mathbf{z}_{COMP}}_z = \underbrace{[\psi_b^n, \theta_b^n, \phi_b^n]^T}_{h(\mathbf{x})} + \mathbf{b}_{COMP} + \mathbf{v}_{COMP}. \quad (2.24)$$

Das Messrauschen ist \mathbf{v}_{COMP} .

Zur Berechnung des Kompassupdate ist es ebenfalls notwendig, die Kovarianz der Orientierungsschätzung in Eulerwinkel umzurechnen. Dies erfolgt durch analytische Linearisierung der nicht-linearen Umrechnungsgleichung von Quaternionen in Eulerwinkel um die aktuelle Schätzung der Orientierung.

Messmodell Pegelstand

Das Messmodell des Pegelstands berechnet die vom Navigationsfilter geschätzte WGS84-Höhe des Körperkoordinaten-Ursprungs \hat{h}_{eb}^e für den anschließenden Vergleich mit einer gemessenen Höhe \tilde{h}_{eb}^e im Kalman-Aktualisierungsschritt. Die gemessene Höhe wird aus einer Pegelstandsmessung abgeleitet. Der entsprechende Algorithmus wurde in [14] vorgestellt. Das Messmodell h_{eb}^e in Form der algebraischen Gleichungen des Zustandsraums, Gl. 2.12, wird aus der aktuellen Zustandsschätzung \mathbf{x} wie folgt gewonnen:

$$\underbrace{h_{eb}^e}_z = \underbrace{\mathbf{p}_{eb,3}^e}_{h(\mathbf{x})} + v_h. \quad (2.25)$$

Dabei ist $\mathbf{p}_{eb,3}^e$ der dritte Eintrag des dreidimensionalen Positionszustands, der bereits in den geodätischen Daten des WGS84-Koordinatensystems ausgedrückt ist, sh. [11]. Das Messrauschen einer Meeresspiegelmessung wird durch v_h berücksichtigt und als normalverteiltes, mittelwertfreies Rauschen gemäß Gl. 2.12 formuliert.

2.2.6 Initialisierung

Die Initialisierung der Navigationszustände Position, Geschwindigkeit und Orientierung sowie der IMU-Biase und der GNSS-Empfängeruhrenfehler wurde bereits im Abschlussbericht zu GALILEOnautic [5] vorgestellt. Der DVL-Geschwindigkeitsbias \mathbf{b}_{DVL} und der Orientierungsbias des Kompass \mathbf{b}_{COMP} werden mit Null initialisiert, da zunächst unbekannt. Die initialen Float-Mehrdeutigkeitszustände $\nabla\Delta N_{UB,0}^{ij}$, die dem Zustandsvektor hinzugefügt werden, werden wie folgt bestimmt:

$$\nabla\Delta N_{UB,0}^{ij} = \frac{\nabla\Delta\hat{r}_{UB}^{ij} - \nabla\Delta\tilde{\Phi}_{UB}^{ij}}{\lambda}, \quad (2.26)$$

wobei $\nabla\Delta\hat{r}_{UB}^{ij}$ der Wert ist, der aus der aktuellen Zustandsschätzung \mathbf{x} des Navigationsfilters über den $\nabla\Delta r_{UB}^{ij}$ -Term in Gl. 2.18 zu berechnen ist. Der Term $\nabla\Delta\tilde{\Phi}_{UB}^{ij}$ ist die aus den Messungen von U und B mit Satellit i und j ermittelte doppeldifferenzierte Trägerphase. λ ist die Wellenlänge des betrachteten Signals.

Die angenommene anfängliche Standardabweichung einer Float-Schätzung beträgt 2 Zyklen, was weniger als 2 m für GPS-L1/L2 WL Trägerwellenlänge, bzw. weniger als 50 cm für GPS-L1 bzw. Galileo-E1 Trägerwellenlängen entspricht. Diese konservative Schätzung ergibt sich aus der Überlegung, dass die Positionslösung des Navigationsfilters im PP/DGNSS-Modus bereits eine Genauigkeit von etwa 50 cm erreicht hat. Wird der Float-Zustand $\nabla\Delta N_{UB}^{ij}$ initialisiert und zum Zustandsvektor \mathbf{x} hinzugefügt, so wird gleichzeitig die Kovarianzmatrix \mathbf{P} auf der Hauptdiagonalen um das Quadrat der anfänglichen Standardabweichung erweitert. Außerdem wird angenommen, dass die anfänglichen Kovarianzen gleich Null sind.

2.2.7 Kompensation totzeitbehafteter Messungen

Durch Vorverarbeitungsaufwand und den Kommunikationsweg der Messdaten bspw. über serielle Schnittstellen zwischen Sensor und verarbeitender Echtzeit-Hardware treffen zu fusionierende Messdaten mit einer Messverzögerung am Navigationsfilter ein. Kalman-Aktualisierungsschritte können dann nicht am aktuellsten Filterzustand durchgeführt werden, sondern müssen in der Vergangenheit vorgenommen werden. Das dazu anzuwendende

2 *Maritimes hochgenaues Navigationsfilter*

Verfahren wurde bereits im Abschlussbericht des Projekts GALILEOnautic [5] vorgestellt und musste für die Entwicklungen in GALILEOnautic 2 Insbesondere im Hinblick auf die Verarbeitung der Mehrdeutigkeiten der Trägerphasen erweitert werden. Dabei muss die Speicherung der Zustände um die Zuordnung der Mehrdeutigkeiten zu den SVIDs der Satelliten und den gewählten Referenzsatelliten der Doppeldifferenzierung ergänzt werden. Details sind in [11] zu finden.

2.2.8 Zustandsautomat

Der Zustandsautomat gibt den angegebenen Filterzustand zurück. Dieser kann entweder im PP/DGNSS-Modus, im Float-Modus, im Fix-Modus oder in einer Mischung aus Float und Fix arbeiten. Die Übergänge zwischen den diskreten Zuständen richten sich nach der Verfügbarkeit von Doppeldifferenzen und der Qualität des Fix-Kandidaten aus dem LAMBDA-Verfahren. Die Qualität des Kandidaten wird durch die q -Norm und die Ambiguity Dilution of Precision (ADOP) [24] bestimmt. Die q -Norm beschreibt den Abstand zwischen den beiden besten Kandidaten für einen Mehrdeutigkeitsfix, gemessen in der Metrik der Kovarianzmatrix. Je kleiner die q -Norm ist, desto größer ist der Abstand zwischen den beiden Kandidaten und desto höher ist die Wahrscheinlichkeit, dass der beste Kandidat der richtige ist. Die Algorithmen wurden in GALILEOnautic 2 so implementiert, dass Kandidaten akzeptiert werden, wenn ihre q -Norm kleiner als 0,3 ist und der ADOP der Float-Mehrdeutigkeit kleiner als ein Drittel Zyklus für länger als eine Sekunde ist. Eine Diskussion über die Parametrisierung der q -Norm findet sich in [25] und ADOP in [26].

2.2.9 Minaturisierung für Lotsassistentz

Für mobile Anwendungen wie zur Lotsassistentz wurde das bestehende Navigationssystem aus dSPACE SCALEXIO, dem geodätischem AsteRx 3 GNSS Empfänger von Septentrio und der 3DM-GX5-25 IMU von LORD Microstrain miniaturisiert. Aus Anwendersicht ergeben sich folgende Anforderungen an ein mobiles Navigationssystem:

- Geringer Preis
- Kompakte Bauform
- Geringes Gewicht
- Tragbarkeit
- Einfacher Aufbau

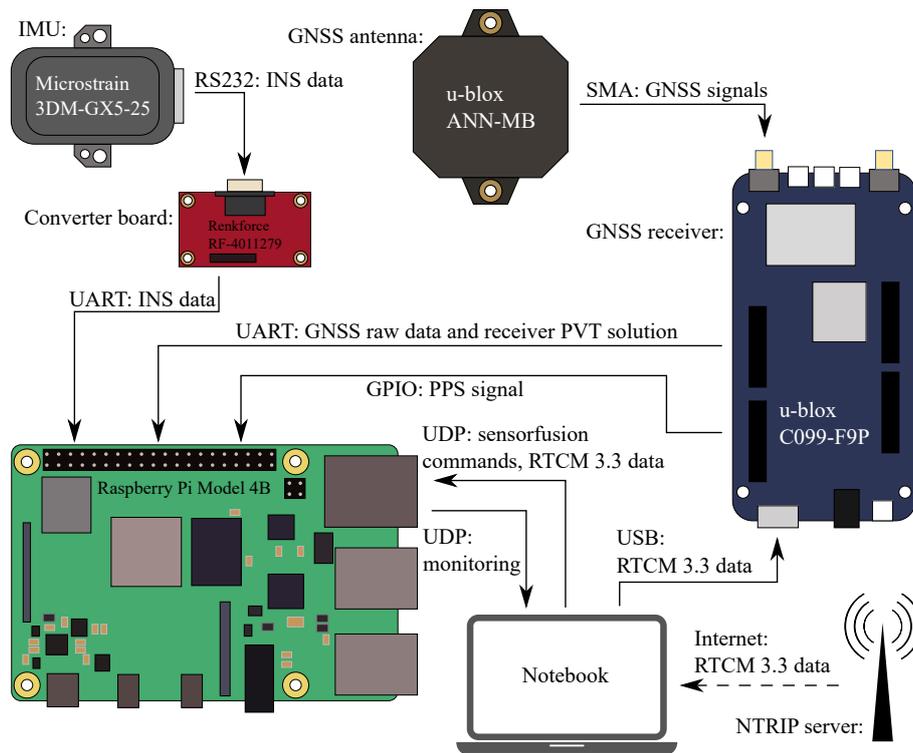


Abbildung 2.4: Hardwarekomponenten des mobilen Navigationsfilters, aus [27]

- Ausreichend hohe Güte der Navigationslösung

Um diese Ziele zu erreichen wurde das bestehende Navigationsfilter auf einen Raspberry Pi 4B portiert. Der Raspberry Pi verfügt über einen Prozessor der auf der weitverbreiteten ARM-Architektur basiert. Der geodätische GNSS-Empfänger wurde durch das C099-Board der Firma u-blox ersetzt, welches den RTK-fähigen ZED-F9P Multi-Band Empfänger integriert. Somit kann eine cm-genaue Positionslösung zur Verfügung gestellt werden. Anstelle einer geodätischen Antennen wird eine ANN-MB Patch-Antenne von u-blox verwendet. Nachteil dieser kompakteren Antenne ist ihre höhere Anfälligkeit gegenüber Multipfadfehlern. Die inertielle Messeinheit von LORD Microstrain wurde beibehalten.

Die Portierung des Navigationsfilter auf den ARM-Prozessor wurde durch ein C++ Multi-Threading Framework realisiert. Das unter MATLAB/Simulink implementierte Navigationsfilter kann code-generiert und in das C++ Framework eingebettet werden, wodurch eine aufwändige Neuimplementierung vermieden wird. Die Schwierigkeit hierbei besteht in der deutlich aufwändigeren Behandlung zeitkritischer Operationen auf dem ARM-Chip. Um das Problem zu lösen implementiert das Multi-Threading Framework fünf parallel laufende Threads (siehe Abb. 2.5). Der erste Thread prozessiert die ankommenden GNSS-Daten des Empfängers und dekodiert die Nachrichten. Die Daten werden auf dem Heap-

2 Maritimes hochgenaues Navigationsfilter

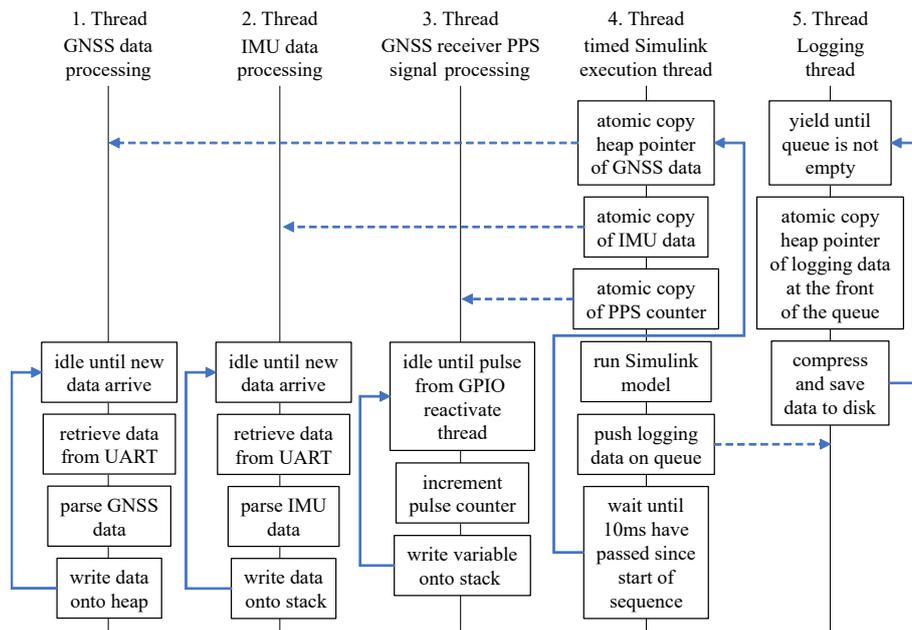


Abbildung 2.5: C++ Multithreading Framework welches auf dem ARM-Prozessor des Raspberry Pi ausgeführt wird, aus [27]

Speicher zur Verfügung gestellt. Der zweite Thread prozessiert und dekodiert die Daten der IMU und schreibt diese direkt auf den Stack-Speicher, da die Datenmenge klein ist. Der dritte Thread implementiert einen Interrupt um den PPS Puls des GNSS-Empfängers abzutasten. Dieser wird für die Synchronisierung von IMU und GNSS genutzt (siehe). Bei erkannter steigender Flanke wird ein global zugänglicher Sekundenzähler (pulse counter) inkrementiert. Der vierte Thread kann als Haupt-Thread betrachtet werden. Dieser fragt im ersten Schritt die Sensordaten auf Heap- und Stackspeicher ab. Anschließend wird das aus Simulink code-generierte Navigationsfilter mit den neuen Daten ausgeführt. Die Navigationslösung sowie weitere relevante Daten werden in eine Datenqueue geschrieben, die zum Datenlogging dient. Der Prozess wird mit 10 ms (100 Hz) ausgeführt. Je nach Dauer der Ausführung des Navigationsfilters wartet der Thread die verbleibende Zeit bis Ablauf der 10ms ab und startet den Vorgang erneut. Die Navigationslösung wird vom Haupt-Thread mittels UDP an etwaige Empfänger verschickt. Ein fünfter Thread fragt die Datenqueue ab und speichert die Daten auf der SD-Karte des Raspberry Pi zwecks Post-Prozessierung.

Abbildung 2.6 zeigt die entwickelte tragbare Lösung für die Lotassistenz. Der Raspberry Pi, der u-blox, die Renkforce Konverplatine und ein LTE-Router wie in Abb. 2.4 sind in einem Koffer untergebracht. Die inertielle Messeinheit ist in einer wasserdichten Box montiert auf der die GNSS-Antenne befestigt wird. Die Box kann mittels einer Multi-Klemme an der Reling eines Schiffes montiert werden. Das Demonstrationssetup bestehend aus



Abbildung 2.6: Sensorbox montiert an der Reling der "Weserlotse" (links) und Koffer mit den Hardwarekomponenten des mobilen Navigationsfilters (rechts)

Sensorbox und Koffer bietet noch deutliches Potenzial bzgl. einer weiteren Miniaturisierung. Die meisten Komponenten können auf einem eigenen Elektronikboard untergebracht werden. Die Box für IMU und GNSS-Antenne ist großzügig ausgelegt worden. Auch hier kann eine kompaktere Lösung entwickelt werden.

2.2.10 Parametrierung

IMU-Parametrierung

Die Parameterierung der Kovarianzen des Prozessmodells der Kovarianzmatrix Q_k wurde im Abschlussbericht zum Projekt GALILEOnautic bereits vorgestellt. Bzgl. der Erweiterung des Prozessmodells um den Geschwindigkeitsbias des DVL wird ein kleines konstantes Rauschen für die Biase der x - und y -Komponente von $0.1 \frac{m}{s}$ angenommen. Die Trägerphasenmehrdeutigkeiten sind tatsächlich konstant, sodass das Prozessmodell die Wirklichkeit exakt beschreibt. Um jedoch eine monoton abnehmende Kovarianz der Mehrdeutigkeiten zu verhindern und eine uneingeschränkte Dynamik der Kovarianzschätzung zu gewährleisten, wird die Standardabweichung σ_n des Rauschens w_n der Mehrdeutigkeiten nicht gleich Null gewählt, sondern erhält den sehr kleinen Wert $(0.001 \text{ cycles})^2$.

Parametrierung GNSS-Rauschmodell

Das GNSS-Rauschmodell v_{GNSS} erfordert die Bestimmung der Messkovarianzmatrix, die mit R_{GNSS} bezeichnet wird. Die Messkovarianzmatrix wird mit Hilfe des Sigma-Epsilon-Modells und des vom Empfänger gemessenen Träger-Rauschleistungsverhältnisses C/N_0 jedes einzelnen Signals berechnet. Details sind bspw. [17] und [28] zu entnehmen.

2 Maritimes hochgenaues Navigationsfilter

Da sich alle Satelliten bei der Bildung der Doppeldifferenzen auf einen Referenzsatelliten beziehen, kann die Kovarianz der Messgrößen nicht vernachlässigt werden. Die Berücksichtigung der Doppeldifferenzierung in der Kovarianz des Messrauschens $\mathbf{R}_{\nabla\Delta}$ wird nach [29] umgesetzt. Dementsprechend folgt die Kovarianzmatrix des doppelt differenzierten Pseudobereichs und der Trägerphase der folgenden Entwurfsregel:

$$\mathbf{R}_{\nabla\Delta,k} = \mathbf{D}_{\nabla\Delta} \mathbf{R}_k \mathbf{D}_{\nabla\Delta}^T, \quad (2.27)$$

wobei $\mathbf{D}_{\nabla\Delta}$ eine rechteckige Matrix mit $n_{sig} - 1$ Zeilen und $2n_{sig}$ Spalten ist, die die lineare Abbildung einer Änderung in den Beobachtungen des Referenzsatelliten auf die Doppeldifferenz darstellt. n_{sig} ist die Anzahl der Signale der betrachteten Observation (Pseudorange oder Trägerphase). Die Matrix \mathbf{R}_k ist eine Diagonalmatrix und gibt die Varianzen der einzelnen Differenzen an. \mathbf{R}_k wird dynamisch als Funktion signalindividueller C/N_0 bestimmt. Es handelt sich um ein normalisiertes Sigma-Epsilon-Modell. Der ursprüngliche Ausdruck wird in [30] diskutiert. Hier wird eine vereinfachte Alternative implementiert, die intuitiv parametrisiert werden kann:

$$R_k^i = \sigma_\epsilon^2 \left(\sigma_k^i(CN_0) \right)^2, \text{ with } \sigma_k^i(CN_0) = \sqrt{\frac{10^{-\frac{CN_0^i}{10}}}{\min\left(10^{-\frac{CN_0^k}{10}}\right)}}, \quad (2.28)$$

wobei $\sigma_k^i(CN_0)$ die normierte Standardabweichung ist. σ_ϵ^2 ist zu parametrieren und entspricht dann der Standardabweichung des Signals mit der besten C/N_0 . Die Standardabweichung aller anderen Signale i ist demnach um ein durch ihren C/N_0 bestimmtes Vielfaches größer. Unter der Annahme, dass alle systematischen Fehler durch die Doppeldifferenz (fast) eliminiert wurden, ergibt sich für Pseudorange eines in diesem Projekt genutzten Septentrio AsteRx 3 mit PolarX Antenne $\sigma_\epsilon = 0.5$ m und für Trägerphasen $\sigma_\epsilon = 0.05$ m bzw. $0,05 \frac{m}{\lambda}$ Zyklen.

Parametrierung DVL-Rauschmodell

Die Parametrierung eines DVL, das dynamisch seine Messgüte bspw. in Form einer Figure Of Merit (FOM) zur Verfügung stellt wurde im Abschlussbericht des Projekts GALILEONautic [5] erläutert. Da Informationen zur Messqualität wie die FOM beim DENEb-DVL nicht verfügbar sind, wird die Standardabweichung von \mathbf{v}_{DVL} in Bezug auf die mittleren Schätzungen in [7] als Konstante $0.2 \frac{m}{s}$ parametrisiert.

Parametrierung Pegel-Rauschmodell

Ein Kalman-Aktualisierungsschritt mit dem Modell der Wasserstandsmessung wird entweder separat durchgeführt, wie in [14] erläutert oder zusammen mit einem Aktualisie-

rungsschritt mit der Trägerphasenmessung durchgeführt, wie in [31] erläutert. Im letzteren Fall wird eine gemeinsame Kovarianzmatrix des Messrauschens $\mathbf{R}_{\nabla\Delta,h,k}$ definiert, wenn Pegelmessungen verfügbar sind:

$$\mathbf{R}_{\nabla\Delta,h,k} = \begin{bmatrix} \mathbf{R}_{\nabla\Delta,k} & \mathbf{0} \\ \mathbf{0} & \sigma_p^2 \end{bmatrix}. \quad (2.29)$$

Dabei ist $\mathbf{R}_{\nabla\Delta,k}$ die Kovarianz der doppel-differenzierten Trägerphasen und σ_p die Standardabweichung der Pegelstandsmessung. Es wird davon ausgegangen, dass die Standardabweichung im Wesentlichen durch den Wellengang beeinflusst wird, der lokal unterschiedlich sein kann. Für den Rostocker Hafen wurde eine Standardabweichung $\sigma_p = 0.2$ m angenommen.

Parametrierung Kompass-Rauschmodell

Es wird angenommen, dass das Rauschen in Gleichung (2.24) der Kompassmessungen unkorreliert ist. Entsprechend gilt:

$$\mathbf{v}_{COMP} = \text{diag}([v_{yaw}, v_{pitch}, v_{roll}]) = \text{diag}([2^\circ, 2^\circ, 2^\circ]). \quad (2.30)$$

Sie ist höher, als das im Datenblatt angegebene Sensorrauschen, um die Abweichungen des Tatsächlichen Rauschens von der im EKF vorausgesetzten idealen Annahmen von mittelwertfreiem normalverteilten Rauschen zu kompensieren.

2.3 Experimentelle Validierung

Dieses Kapitel zeigt Ergebnisse der Entwicklungen zum maritimen Navigationsfilter. Es werden Ergebnisse mit einem Straßenfahrzeug (RWTH-Versuchsfahrzeug), mit dem unbemannten Versuchsfahrzeug MESSIN, dem Forschungsschiff DENEBO und dem mobilen Lotsassistentensystem gezeigt.

2.3.1 Optimierte Satellitenkonstellation

Messaufbau

Der Navigationsfilter mit Satellitenkonstellation-Optimierung wird auf einem 900-MHz-Single-Core Rapid-Control-Prototyping (RCP) Gerät, MicroAutoBox aus dem Hersteller

2 Maritimes hochgenaues Navigationsfilter

dSPACE, implementiert. Die GNSS-Beobachtungen werden von einem Septentrio AstRx3-HDC-Empfänger mit einer Frequenz von 10 Hz ausgegeben. Der Empfänger und die RCP-Einheit werden über eine serielle Schnittstelle erreicht. Für die Trägheitsmessung wird ein 9 DOF-IMU-Sensor der Marke LORD MicroStrain 3DM-GX4-25 verwendet, der über die serielle Schnittstelle angeschlossen ist und Beschleunigungen, Winkelgeschwindigkeiten und Magnetometer liefert. Der Empfänger stellt einen Pulse-Per-Second (PPS) bereit. Mit Hilfe der PPS werden die Kommunikations- und Verarbeitungsverzögerungen des Empfängers kompensiert. Als Referenz wird eine Real-Time-Kinematik (RTK) Lösung durch Septentrio-Empfänger geschätzt.

Ergebnisse

Abb. 2.7 zeigt die Validierungsergebnisse der Echtzeitfähigkeit der DOP-Optimierung unter der Worst-Case-Situation für die Optimierung. Unter Worst-Case versteht man die Einstellung, die zur maximalen Belastung der Berechnungseinheit führt. Die Optimierung erlaubt die Berücksichtigung von bis zu 15 Satelliten aufgrund der Speicherbegrenzung der Berechnungseinheit. Zusätzlich wird die DOP-Optimierung unter folgenden Bedingungen durchgeführt: alle 10 Sekunden (wenn keine andere Bedingung ausgelöst wird), Verlust eines oder mehrerer Satelliten in der optimalen Einstellung, letzte Optimierung ist aufgrund mangelnder Rechenkapazität fehlgeschlagen. Die Ergebnisse zeigen, dass es insgesamt 729 Optimierungsversuche während ca. 900 Sekunden gibt. 636 davon werden erfolgreich ausgeführt. Wenn ein Versuch fehlschlägt, wird er in der nächstmöglichen Epoche wiederholt. Außerdem sind während der Fahrt bis zu 18 Satelliten sichtbar. Gemeinsam sind 13 bis 15 Satelliten sichtbar, die unter dem von uns gesetzten Limit (15 Satelliten) liegen. Wenn man bedenkt, dass dies das Ergebnis der Worst-Case-Situation ist, verifiziert dieser Test die Echtzeitfähigkeit der Implementierung.

Abb. 2.8 zeigt die referenzgefahrene Flugbahn in der Stadt Aachen, die insgesamt 900 Sekunden lang ist und vom Septentrio-Empfänger generiert und aufgezeichnet wurde. Erwähnenswert ist, dass das Fahrzeug von ca. 300 bis 400 Sekunden auf der Straße durch den Laufenberg fährt, der von hohen Bäumen und Gebäuden umgeben ist, was zu einer sehr begrenzten Verfügbarkeit von Satelliten und einer verminderten Qualität der Satellitensignale führt.

Abb. 2.9 zeigt den 2D-Fehler der Navigationslösung gegenüber der RTK-Referenz in Meter an. Es ist zu beobachten, dass im offenen Bereich ohne Signalstörungen (mit Ausnahme von 300 bis 400 Sekunden) der EKF ohne DOP oder WDOP die beste Genauigkeit aufweist. Das bedeutet, wenn die Signale keine signifikanten Interferenzen aufweisen, ist die optimale Untersektionswahl für die Pseudorange-Anwendung einfach die Verwendung aller

2.3 Experimentelle Validierung

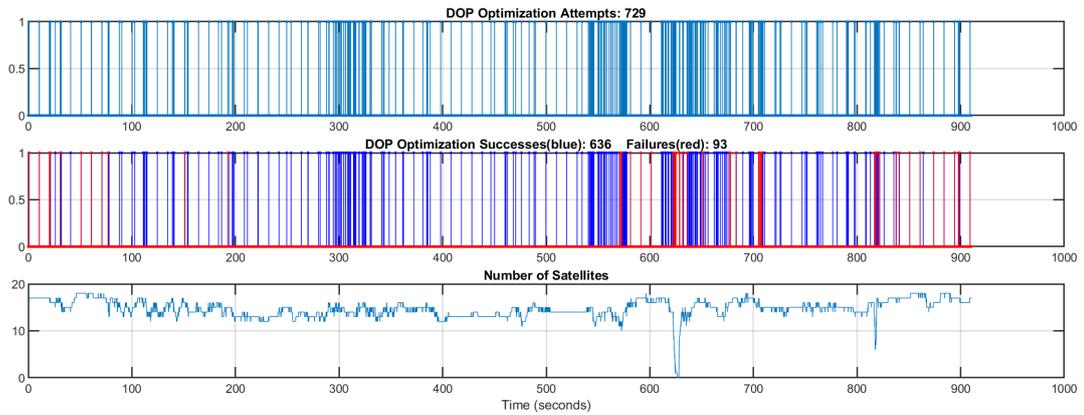


Abbildung 2.7: Validierung der Echtzeitfähigkeit der DOP-Optimierung während der Fahrt: Gesamtzahl der Versuche, erfolgreiche und fehlgeschlagene Fälle und Anzahl der Satelliten

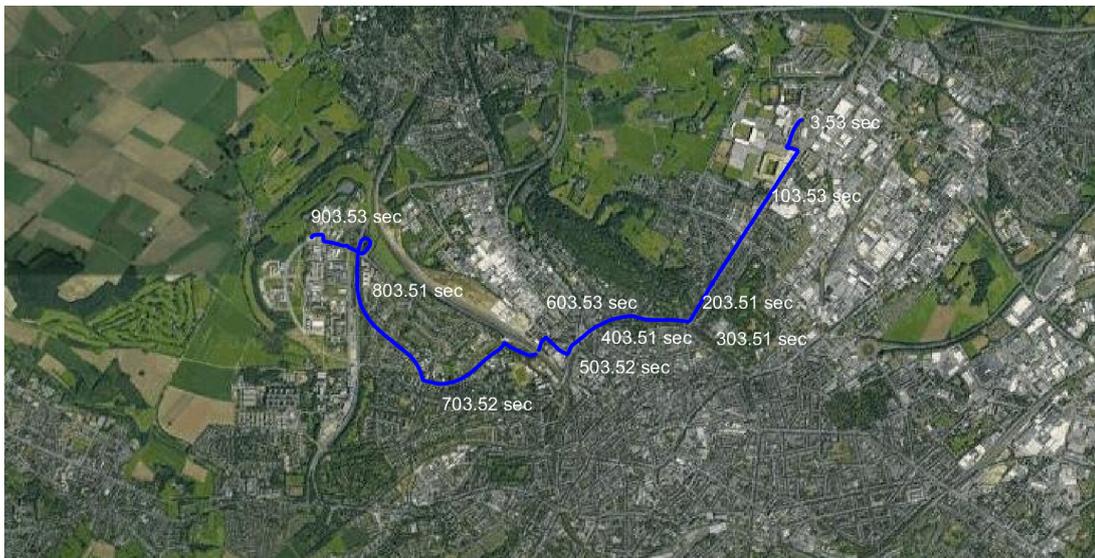


Abbildung 2.8: Referenzfahrstrecke in der Stadt Aachen. Kartendaten: ©2020 GeoBasis-DE/BKG(©2009), Google.

2 Maritimes hochgenaues Navigationsfilter

verfügbaren Satelliten. Wenn die Signalinterferenzen auftreten, kann die Genauigkeit verbessert werden, indem man eine Teilmenge aller verfügbaren Satelliten auswählt. Das Ergebnis von DOP/WDOP ist die Genauigkeit einer Navigationslösung, bei der 10 Satelliten aus allen verfügbaren Satelliten ausgewählt werden, wenn mehr als 10 Satelliten verfügbar sind. Im Durchschnitt hat WDOP, bei dem die Signalqualität berücksichtigt wird, eine bessere Genauigkeit als DOP, bei dem nur die geometrische Satellitenkonstellation berücksichtigt wird.

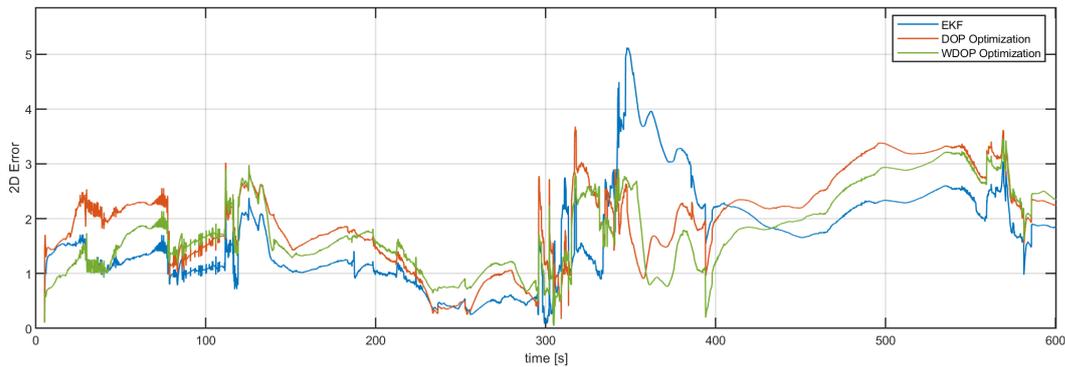


Abbildung 2.9: 2D-Fehler im Meter während der Testfahrt.

2.3.2 Trägerphasen-gestütztes Navigationsfilter

Versuchsaufbau

Das Trägerphasen-gestützten Navigationsfilter wurde mit MESSIN evaluiert. Galileo-GPS Dualkonstellation konnte dabei nicht evaluiert werden, da der für Rostock bzw. Mecklenburg-Vorpommern zuständige Satellitenpositionierungsdienst der Deutschen Landesvermessung (SAPOS) bis Ende der Projektlaufzeit die Bereitstellung differenzieller Galileo-Daten nicht umgesetzt hat. Eine Evaluierung von eng gekoppeltem RTK in Galileo-GPS Dualkonstellation wurde jedoch mit einem PKW in Aachen durchgeführt und in [11] veröffentlicht. Es folgen Ergebnisse zu eng gekoppeltem GPS-basiertem RTK mit L1-L2C Wide-Lane (WL)-Kombinationen, die in [31] detailliert diskutiert werden.

Abb. 2.10 zeigt das Versuchsfahrzeug MESSIN des Projektpartners Universität Rostock, ist. Es ist mit IMU- und GNSS-Sensorik ausgestattet. Die Hebelarm zwischen IMU und GNSS-Antenne als auch zwischen IMU und DVL, sind rot dargestellt und wurden mit einer Leica Laser-Totalstation ermittelt:

$$\begin{aligned} \mathbf{l}_{ba}^b &= [0.05 \text{ m}, -0.398 \text{ m}, -0.735 \text{ m}]^T, \\ \mathbf{l}_{bd}^b &= [-0.321 \text{ m}, 0.683 \text{ m}, 0.826 \text{ m}]^T. \end{aligned}$$



Abbildung 2.10: Unbemanntes Fahrzeug (USV) MESSIN, aus [11]; USV MESSIN ist Eigentum der Universität Rostock.

GNSS-Messepochen wurden bei 10 Hz von einem Septentrio asterX3 GNSS-Empfänger [32] erzeugt, der mit einer Septentrio PolaNt-x MF-Antenne ausgestattet war. Eine LORD MicroStrain 3DM-GX4-25 IMU [33] lieferte Beschleunigungen und Drehraten bei 100 Hz sowie ein dreidimensionales Magnetfeld zur Initialisierung der Orientierung des rekursiven Filters. Die differenziellen Daten wurden über SAPOS Mecklenburg-Vorpommern im Format RTCM 3.1 mit 1 Hz von einem NTRIP Server empfangen. Für die Bildung von Doppeldifferenzen mit GNSS-Messepochen des Empfängers B wurden die Nachrichtentypen 1004 (Beobachtungen, GPS-L1 / L2C) und 1005 (Position der Referenzantenne B) decodiert.

Die Ergebnisse werden mit einem in RTK-Fix betriebenen Referenzempfänger Septentrio asterX3 verglichen (im Folgenden als "Ground Truth" bezeichnet). Um die Qualität der Float-Schätzung \mathbf{n}_{UB} und der Fixed-Schätzung $\check{\mathbf{n}}_{UB}$ zu referenzieren, muss eine Ground Truth der Mehrdeutigkeiten erstellt werden. Diese wird aus der RTK-PVT des Septentrio asterX3 und den für die hier vorgestellten Algorithmen ermittelten Doppeldifferenzen berechnet.

Abb. 2.11 zeigt in (a) die von MESSIN gefahrene Ground-Truth Trajektorie und in (b) den Skyplot der verfügbaren GPS-Satelliten während des Tests. SVID 7 erscheint nach 160 Sekunden. Es wird der Satellit mit dem größten Elevationswinkel als Referenzsatellit gewählt. Daher ist SVID 4 der Referenzsatellit in der folgenden Auswertung.

Das erste Diagramm in Abb. 2.12 zeigt die vom Referenzempfänger aufgezeichnete Geschwindigkeit entlang der gefahrenen Trajektorie. Der zweite Plot zeigt die geometrische

2 Maritimes hochgenaues Navigationsfilter

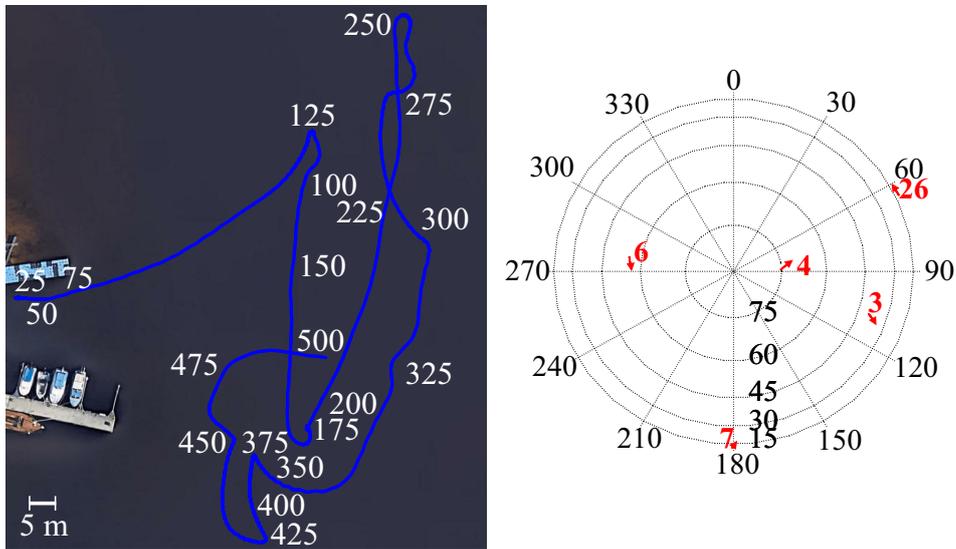


Abbildung 2.11: Trägerphasen-gestütztes Navigationsfilter: 2D-Trajektorie von MESSIN im Rostocker Hafen, mit Zeitstempeln in Sekunden und Skyplot mit der Konstellation der für GPS Wide-Lane (WL)-Kombination verwendeten Satelliten, aus [31]. Kartendaten: ©2021 GeoBasis/BKG(©2009), Google

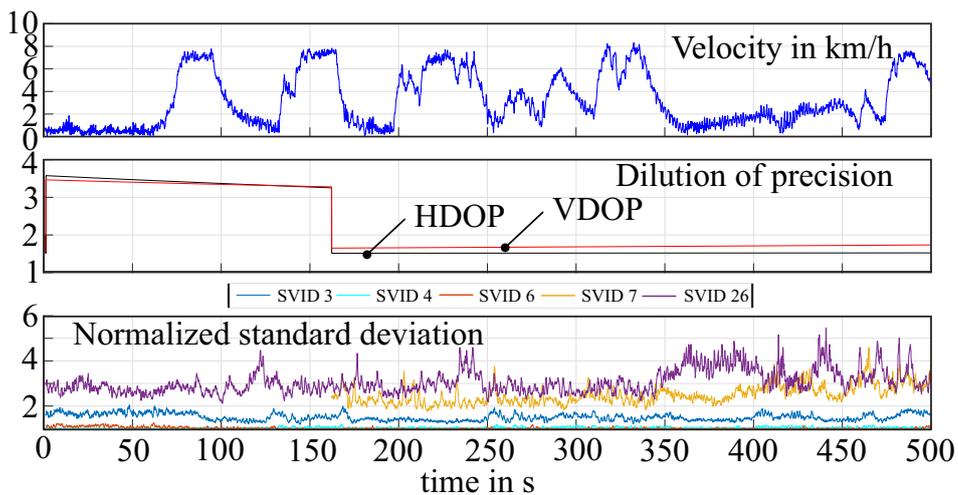


Abbildung 2.12: Trägerphasen-gestütztes Navigationsfilter: Ground-Truth Geschwindigkeit, DOP der verfügbaren Konstellation und normalisierte Standardabweichung $\sigma_k^i(CN0)$ für jede SVID, aus [31]

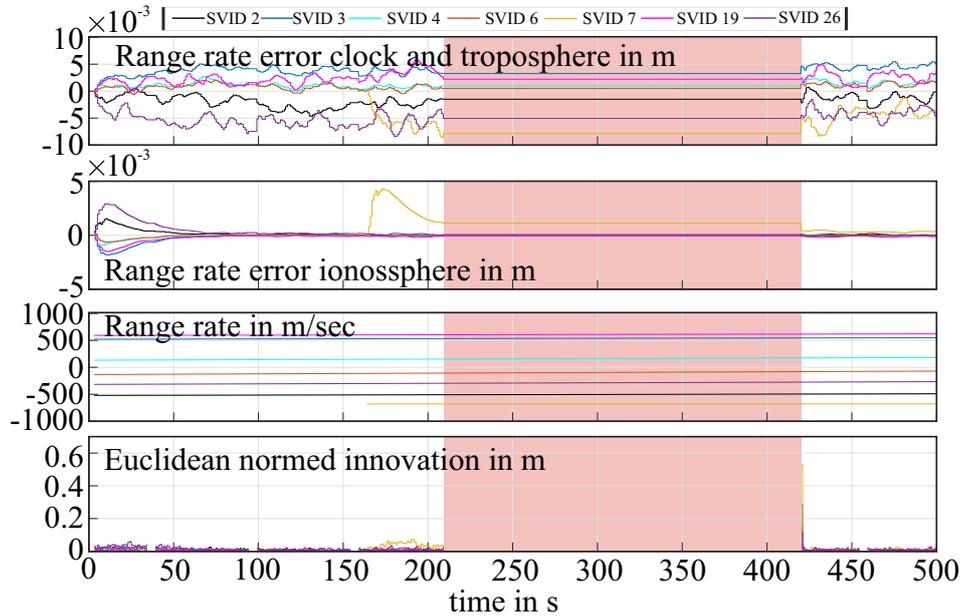


Abbildung 2.13: Auswertungsergebnisse lokale Schätzung von GNSS-Beobachtungen mit Abbruch der Mobilfunkverbindung zum SAPOS-Dienst über 210 s (rot hervorgehoben), aus [31]

Konstellation in Form der vertikalen Dilution Of Precision (VDOP), die die Verstärkung des Rauschens bei der Messung in der vertikalen Positionslösung darstellt und die horizontalen DOP (HDOP), die die entsprechende horizontale Verstärkung darstellt. Die Darstellung zeigt eine unerwünschte geometrische Konstellation bis zum Auftreten von SVID 7. Die letzte Darstellung zeigt die einheitenlose normierte Standardabweichung $\sigma_k^i(CN0)$ aus Gl. 2.28. Es zeigt sich allgemein, dass mit der in GALILEONautic 2 entwickelten Parametrierungsmethode SVIDs mit großen Elevationswinkeln (vgl. Abb. 2.11) eine kleinere Standardabweichung zugeordnet wird als solchen mit kleinen Elevationswinkeln.

Ergebnisse lokale Schätzung differenzieller Daten

Abb. 2.13 zeigt die Auswertung des Kalman-Filters zur Schätzung differentieller Wide Lane-Messeperioden von Empfänger B . Der künstlich erzeugte Verlust von differentiellen Daten über 210 Sekunden ist rot hervorgehoben. Tatsächlich werden während des gesamten Versuchslaufs zusätzlich Verbindungsabbrüche von 5 s in Abständen von 55 s beobachtet. Die Ursache für diese regelmäßigen Verbindungsabbrüche ist unbekannt, könnte aber beispielsweise im NTRIP-Server selbst oder im verwendeten LTE-Mobilfunk-Router liegen. Die beiden oberen Diagramme zeigen die geschätzten Entfernungsratenfehler der Uhren und der Troposphäre \mathbf{E}_d und Ionosphäre \mathbf{I}_d . Da beide Fehler als Random Walk

2 Maritimes hochgenaues Navigationsfilter

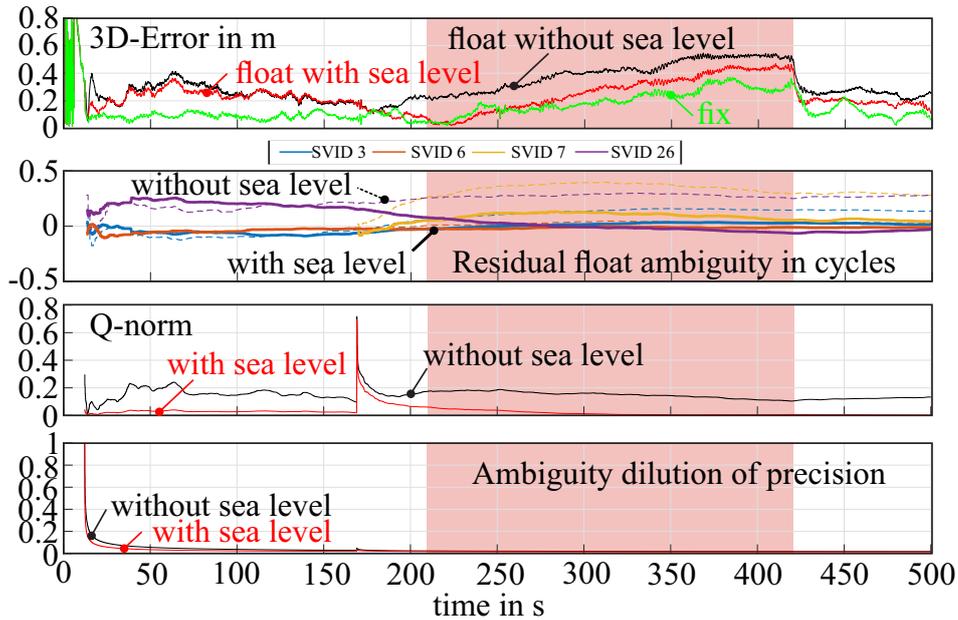


Abbildung 2.14: Auswertungsergebnisse Wide-Lane Float-Mehrdeutigkeitschätzung mit und ohne Integration des Pegelstands und Abbruch der Verbindung zum SAPOS-Dienst von über 210 s (rot hervorgehoben), aus [31]

modelliert wurden, bleiben die Zustände während Phasen des Abbruchs der Mobilfunkverbindung oder in Phasen, in denen die Zustände keine Kalman-Aktualisierung erfahren, konstant. Wie erwartet, wird der Fehler der troposphärischen Entfernungsraten E_d^i für Satelliten mit zunehmender Elevation negativ und für Satelliten mit abnehmender Elevation positiv eingeschätzt (da mit zunehmender Elevation der Weg durch die Atmosphäre kürzer und damit die Signalverzögerung kleiner wird). Die GPSL1-L2C Wide-Lane-Kombination reduziert den ionosphärischen Fehler erheblich, weshalb die Schätzung des Fehlers der ionosphärischen Entfernungsraten gegen Null konvergiert. Die dritte Grafik zeigt die Entfernungsraten δr_B , die aus den Ephemeridendaten gemäß Gl. 2.6. Wie erwartet, sind die Entfernungsraten für Satelliten mit großer Elevation klein und für Satelliten mit kleiner Elevation groß. Das letzte Diagramm zeigt die Norm der Innovation \mathbf{y}_B aller SVIDs, die vom Kalman-Filter gemäß Gleichung verfolgt werden. 2.8. Da in Phasen fehlender differenzieller Messepochen des Empfängers B keine Kalman-Aktualisierungsschritte berechnet werden, wird auch keine Innovation ermittelt. Die Norm der Innovation \mathbf{y}_B konvergiert schnell, so dass man davon ausgehen kann, dass das Zustandsraummodell die Dynamik der tatsächlichen GNSS-Beobachtungen gut repräsentiert.

Ergebnisse Navigationsfilter

Abb. 2.14 zeigt die Ergebnisse des Navigationsfilters. Auch hier ist der künstlich erzeugte Verlust von differenziellen Daten über 210 Sekunden in rot hervorgehoben. Es werden drei Betriebsmodi des Navigationsfilters verglichen:

1. *Float-Mehrdeutigkeitsschätzung ohne Pegeldaten* (schwarz): Das Navigationsfilter schätzt Wide-Lane-Mehrdeutigkeiten \mathbf{n}_{UB} . Die Kalman-Aktualisierungsschritte werden mit dem Modell der doppeldifferenzierten Pseudorange Gl. 2.18 und Trägerphasen Gl. 2.21 durchgeführt. Die in Abb. 2.2 gezeigte LAMBDA-Rückkopplungsschleife wird durchtrennt, sodass das Navigationsfilter die von LAMBDA geschätzten ganzzahligen Mehrdeutigkeiten $\check{\mathbf{n}}_{UB}$ nicht verwenden kann.
2. *Float-Mehrdeutigkeitsschätzung mit Pegeldaten* (rot): Zusätzlich zur Betriebsart *float without sea level* Kalman-Updates mit dem Meeresspiegelmodell von Gl. 2.25 durchgeführt.
3. *Fix-Mehrdeutigkeitsschätzung* (grün): Die LAMBDA-Rückkopplungsschleife in Abb. 2.2 ist geschlossen. Der Fixkandidat, der die in Abschnitt 2.2.2 erläuterten Bedingungen erfüllt, wird in den Navigationsfilter eingespeist und zusammen mit der doppeldifferenzierten Trägerphase als Entfernungsmessung analog zu Gl. 2.18 genutzt. Die notwendige vorherige Float-Mehrdeutigkeitsschätzung wird durch die Betriebsart *Float-Mehrdeutigkeitsschätzung mit Pegeldaten* erzeugt.

Das erste Diagramm zeigt den 3D-Positionsschätzungsfehler der drei Betriebsarten des Navigationsfilters gegenüber Ground-Truth. Das zweite Diagramm zeigt die Residuen der Float-Mehrdeutigkeitsschätzung gegenüber Ground-Truth, das dritte Diagramm die q-Norm und das letzte Diagramm die ADOP. Die Auswertung zeigt, dass die Integration des Pegelstands in die Float-Mehrdeutigkeitsschätzung die 3D-Positionsschätzung deutlich verbessert. Vor allem aber werden die Float-Mehrdeutigkeitsschätzung selbst und die q-Norm verbessert. Auch die über alle Mehrdeutigkeiten ermittelte Unsicherheit der Float-Mehrdeutigkeitsschätzung (ADOP) profitiert sichtbar von der Pegelintegration. In der 210 Sekunden dauernden Phase des Verlusts von differenziellen Daten verschlechtert sich die Positionsschätzung aller drei Betriebsarten deutlich. Der Fehler der Mehrdeutigkeitsschätzung bleibt jedoch gering, wenn der Pegelstand integriert wird. Eine tiefere Diskussion ist in [31] zu finden.

2.3.3 Kompass-gestütztes Navigationsfilter

Versuchsaufbau

Der Versuchsaufbau gleicht dem Versuchsaufbau aus dem vorherigen Kapitel. Für die Integration des Kompass muss kein Hebelarm berücksichtigt werden.

Ergebnisse

Um die Robustheit zu erhöhen wird das Filter zusätzlich um drei Biaszustände für alle drei Winkel erweitert. Im Folgenden sind die Ergebnisse des Filters mit Bias-Schätzung dargestellt. Unter Einbeziehung der Messung der Litef IMU wird ein 2-D Fehler von 0.111 m (0.112 m ohne Litef Messung) erreicht (siehe Abb. 2.15). Durch das zusätzliche Update bleibt die Positionslösung ähnlich gut. Jedoch zeigt Abb. 2.16, dass durch das zusätzliche Heading-Update die Filterlösung für das Heading näher an die Messung gebracht wird. Abb. 2.17 zeigt die Ergebnisse der Bias Schätzung. Pitch und Roll Bias werden zu etwa -1.3 und -2.4 ° geschätzt. Der geschätzte Bias des Yaw-Winkels bleibt unter 0.4 °.

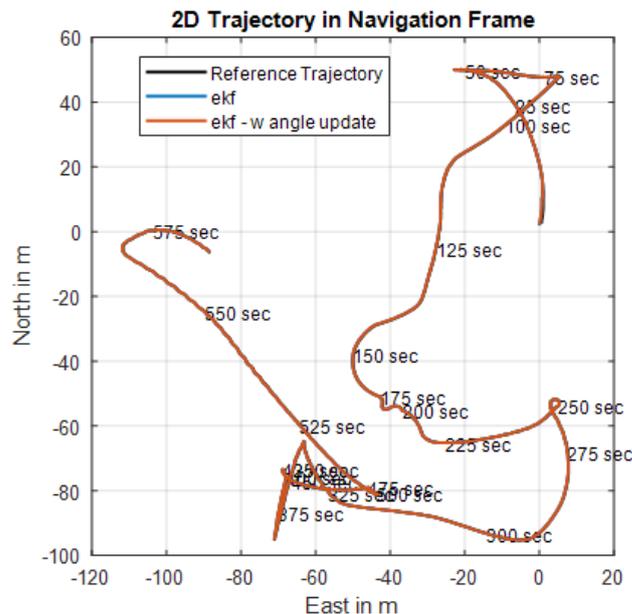


Abbildung 2.15: Kompass-gestütztes Navigationsfilter: Filterlösung und Referenztrajektorie

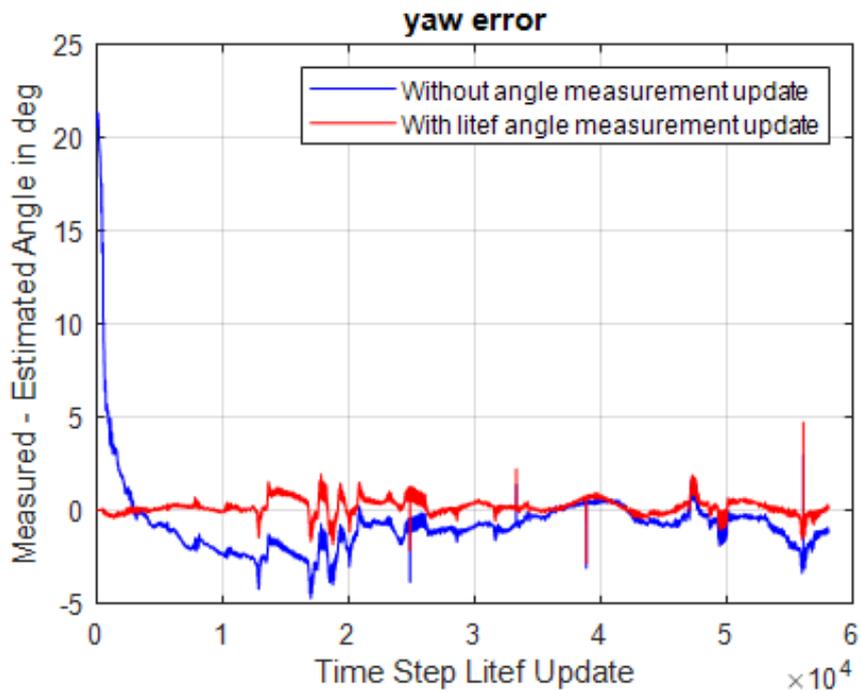


Abbildung 2.16: Kompass-gestütztes Navigationsfilter: Vergleich zwischen dem mit Litef-IMU gemessenen Heading und der Filter Lösung mit und ohne Verwendung des Winkel-Updates

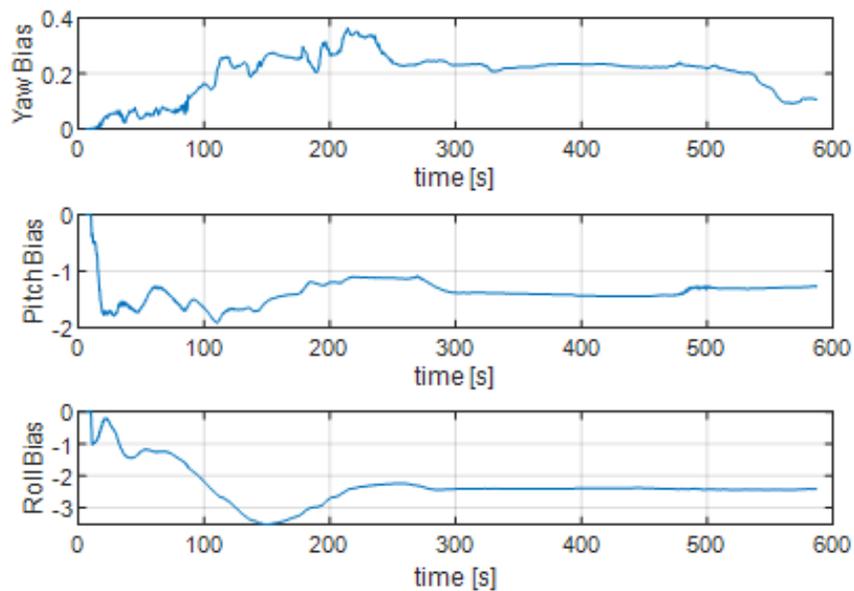


Abbildung 2.17: Kompass-gestütztes Navigationsfilter: Durch das Filter geschätzter Bias in den Litef-Winkelmessungen

2.3.4 Navigationsfilter auf DENEb

Versuchsaufbau

Die DENEb fungiert als Träger für die Raytheon Anschütz SYNOPSIS Software für intelligente Brückensteuerung. Dieses hochmoderne INS bildet die Schnittstelle zwischen der bordeigenen Sensorik und der Aktorsteuerung und dient damit als Plattform für fortschrittliche Anwendungen der Schiffsautomation. Zu den Standardfunktionen von SYNOPSIS gehört z.B. ein konsistentes Sensordatenmanagement.

In GALILEOnautic 2 wurde eine zusätzliche Navigationsbrücke an Bord des Forschungsschiffs gebracht auf der SYNOPSIS installiert wurde. Dabei handelte es sich um eine Brücke in Originalgröße, die im Forschungsprojekt *ACTRESS* in einen mobilen 10'-Offshore-Container integriert wurde und bspw. auch einen Raytheon Anschütz Standard 30 Kompass enthält. Der Datenaustausch an Bord des Schiffes erfolgte gemäß dem NMEA 0183-Standard für die Anbindung elektronischer Geräte, um die Verwertung der Projektergebnisse zu erleichtern. Diese Schnittstelle liefert NMEA VBW DVL Geschwindigkeitsinformation, die wie in Kap. 2.2.5 erläutert, in das Navigationsfilter eingebunden werden.

Abb. 2.18 zeigt die Sensorkonfiguration. Es wurde eine Septentrio PolaNt-x MF-Antenne auf dem Peildeck der DENEb positioniert und eine LORD MicroStrain 3DM-GX4-25 IMU auf der Brücke verbaut. Das genutzte DVL ist in der Nähe des Bugstrahlruders platziert. Der Hersteller des DVL sowie die tatsächliche Größe der Hebelarme dürfen nicht veröffentlicht werden. Ein Raspberry Pi diente als Client für externe Daten. RTCM-Nachrichten wurden hier empfangen, verarbeitet und zur GNSS-Beobachtungskorrektur an die Rechenplattform dSPACE MicroAutoBox II weitergeleitet. Der Zugriff auf *pegelonline.wsv.de* zur Nutzung aktueller Pegelstände erfolgte über denselben Raspberry Pi.

Wie in den Kapiteln zuvor, wurde der Septentrio AsteRx3-Empfänger im Modus RTK-Fix als Ground-Truth genutzt.

Ergebnisse

Abb. 2.19 zeigt die wichtigsten Ergebnisse der Erprobung mit dem Forschungsschiff DENEb aus [14]. Der blau hinterlegte Bereich visualisiert einen virtuellen GNSS-Ausfall. In der oberen Darstellung ist der zweidimensionale Positionsfehler für ein Navigationsfilter ohne DVL-Integration (schwarz), eines mit DVL-Integration, jedoch ohne Schätzung des DVL Messbias (grau) und eines mit DVL-Integration und gleichzeitiger Schätzung des DVL Messbias (rot) dargestellt. Die zweite Darstellung zeigt den Arbeitsmodus des



Abbildung 2.18: Forschungsschiff DENE B des BSH: Versuchsaufbau mit IMU, DVL und GNSS-Antenne, aus [11]

Referenzempfängers. In der dritten Darstellung ist die absolute dreidimensionale Geschwindigkeit dargestellt.

Der Vergleich der Positionsschätzung zeigt, dass nach Integration des DVL in das Navigationsfilter, die Positionsgenauigkeit signifikant schlechter wird, auch wenn die Berücksichtigung eines Geschwindigkeitsbias im Prozessmodell (rot) die Positionslösung gegenüber einer Messachtung dieses systematischen Fehlers verbessert. Es wird aber auch sichtbar, dass bei GNSS-Ausfall das DVL wie schon in [7] gezeigt, die Navigationslösung stabilisiert.

Anhand der in der unteren Darstellung gezeigten die y -Komponente der DVL-Geschwindigkeit wird sichtbar, dass die vom DVL gemessene Quergeschwindigkeit der DENE B von mangelhafter Qualität ist. Dazu werden drei Geschwindigkeiten verglichen. Die mit dem DVL gemessene Geschwindigkeit ist Gelb dargestellt. Rot dargestellt ist das die Geschwindigkeitsschätzung des Navigationsfilters mit integriertem DENE B-DVL. Eine Navigationsfilterimplementierung ohne integriertes DVL, bei dem jedoch die aufgrund der aktuellen Zustandsschätzung über Gl. 2.23 erwartete DVL Geschwindigkeit ermittelt wird, ist schwarz dargestellt. Da gemäß der ersten Darstellung des Filters ohne DVL-Integration (schwarz)

2 Maritimes hochgenaues Navigationsfilter

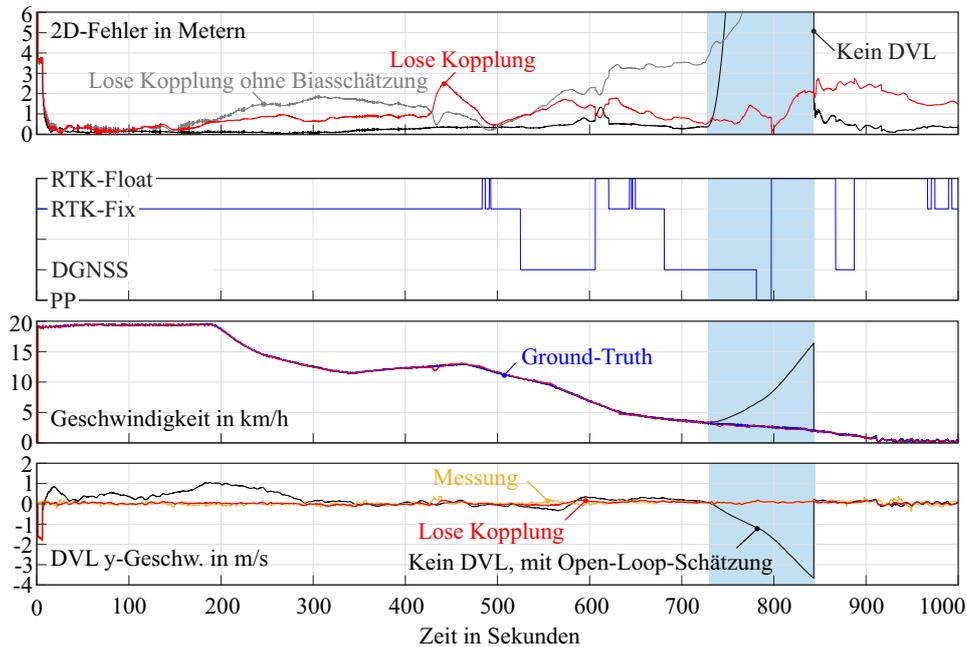


Abbildung 2.19: DENEb Erprobung: Einfluss der DVL-Geschwindigkeit auf die Zustands-schätzung des Navigationsfilters, aus [14]

bis zum GNSS-Ausfall die höchste Genauigkeit erreicht, wäre bei einer zuverlässigen, genauen DVL-Geschwindigkeitsmessung zu erwarten, dass Open-Loop-Schätzung und tatsächliche Messung nahe beieinander liegen. Dass dies nicht Fall ist, weist die mangelhafte Messqualität des DVL nach.

2.3.5 Navigationsfilter zur Lotsassistentz mit Galileo-GPS DGNSs

Versuchsaufbau

Für eine erste Validierung unter schwierigen Empfangsbedingungen wurde ein Messaufbau wie in Abb. 2.6 dargestellt durch einen Koffer mit Hardwarekomponenten und einer Sensorbox wie in Abb. 2.20 realisiert. Die Sensorbox wurde auf einem Auto montiert.

Ergebnisse

Abb. 2.21 stellt die Positionierungsgenauigkeit des eng-gekoppelten Navigationsfilters gegenüber der RTK-Referenz des u-blox GNSS-Empfängers dar. Das Navigationsfilter nutzt differentiell-korrigierte GNSS Pseudorange und Doppler-Beobachtungen. Es zeigt sich, dass mit enger Kopplung ein mittlerer 2D-Fehler von 1.27 m erreicht wird. Das dargestellte



Abbildung 2.20: Messaufbau Versuchsfahrzeug für Validierung des tragbaren Navigationssystems, aus [27]

Tabelle 2.2: Fehlermetriken für fünf Messfahrten in Aachen. Test # 5 wurde unter freiem Himmel (ähnlich Hafenumgebung) durchgeführt, aus [27]

Test #	Mean # Sats	Mean/Std 2D-Fehler [m]	Mean/Std 3D-Fehler [m]
1	12.21	1.54 (1.29)	2.20 (1.29)
2	14.89	1.19 (0.93)	2.80 (0.48)
3	12.95	1.27 (0.67)	3.26 (1.76)
4	12.14	1.48 (0.76)	1.97 (0.75)
5	13.00	0.42 (0.21)	1.39 (0.10)

2 Maritimes hochgenaues Navigationsfilter

Szenario kann als äußerst anspruchsvoll angesehen werden, da es im südlichen Bereich zu starken Mehrwegeffekten (Semi-Urban) sowie mehreren Brückendurchfahrten kommt.

Tab. 2.2 zeigt die Fehlermetriken für fünf Messfahrten in Aachen mit dem Versuchsfahrzeug. Der mittlere 2D-Fehler bewegt sich im Bereich 1.19 m bis 1.54 m. Versuchsfahrt # 5 repräsentiert Hafen-ähnliche Bedingungen, da nur unter freiem Himmel gefahren wurde. Hier beträgt der mittlere 2D-Fehler 42 cm.

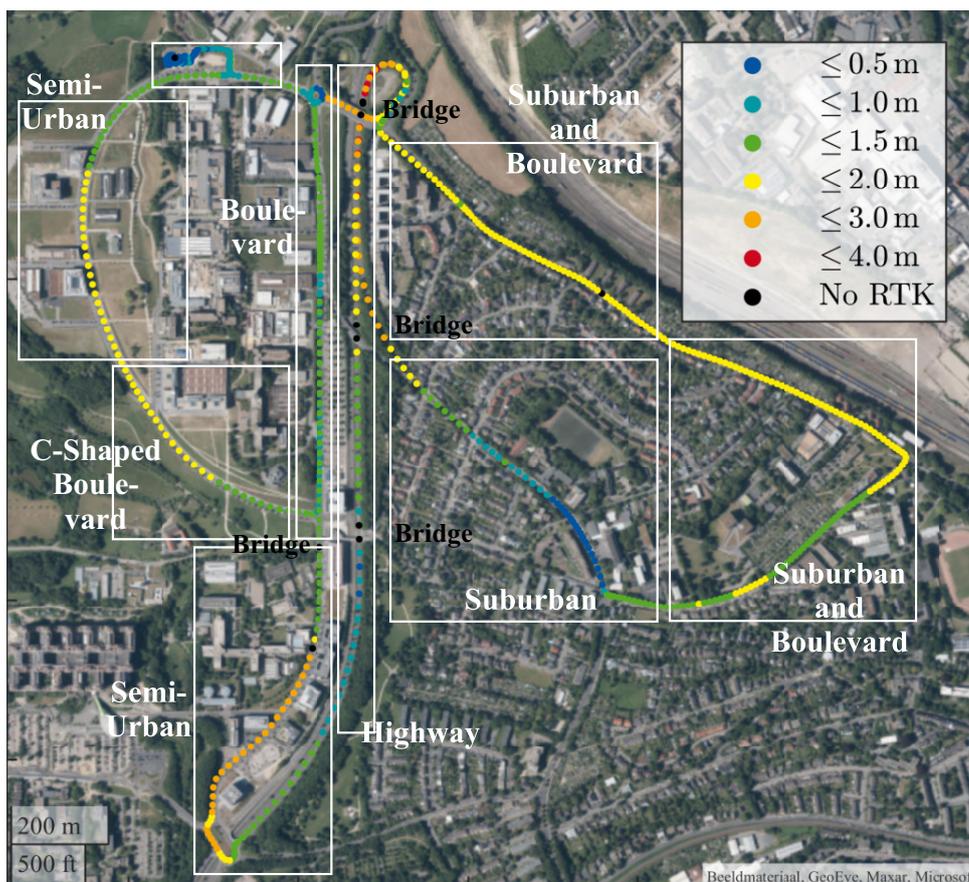


Abbildung 2.21: Vogelperspektive auf eine der gefahrenen Trajektorien am Campus- Melaten, Aachen, aus [27], Kartendaten: GeoEye Inc.

Abb. 2.22 zeigt die Heading-Lösung sowie den Fehler der Heading-Lösung gegen die aus Geschwindigkeitsmessung ermittelte Heading-Lösung des u-blox. Es zeigt sich, dass Roll- und Pitchwinkel im erwarteten Bereich liegen und der Yaw-Winkel (Heading) qualitativ mit der gefahrenen Richtung übereinstimmt. Eine Berechnung des mittleren Heading-Fehlers ergibt einen Mittelwert von 0.88° und einer Standardabweichung von 0.37° . Der Fehler liegt somit im von der IMO geforderten Genauigkeitsbereich von 1.5° für dynamische Anwendungen [34] für die gegebene Referenzlösung.

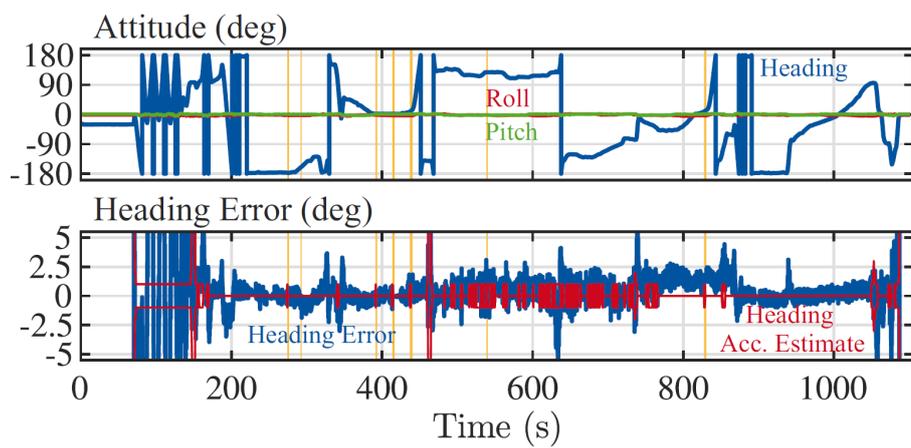


Abbildung 2.22: Heading-Schätzung und Fehler gegenüber u-blox Referenz, aus [27]

3 Integres Navigationsfilter

Autor: Shuchen Liu, Jan-Jöran Gehrt

3.1 Einleitung und Stand der Technik

Bei autonomen Fahrzeugen ist das Kernmodul das Guidance, Navigation, Control (GNC)-System, bei dem das Fahrzeug einer von einem Führungsmodul erzeugten Referenztrajektorie durch die vom Steuermodul bereitgestellten Steuerkräfte und -momente folgt. Dabei ist zu beachten, dass die grundlegende Eingangsgröße der Trajektoriengenerierung und -steuerung die Schätzung des Fahrzeugzustands durch ein, wie im vorherigen Kapitel 2 vorgestellten, Navigationsmodul ist. Bei sicherheitskritischen Anwendungen muss die Zustandsschätzung jedoch nicht nur hochgenau sondern auch vertrauenswürdig sein, d.h. das Navigationssystem muss integer sein.

In einem INS/GNSS-Navigationssystem darf das Risiko von Sensorausfällen nicht unterschätzt oder ignoriert werden. In [35] werden typische Ausfälle von Global Navigation Satellite System (GNSS) und Integriertes Navigationssystem (INS) vorgestellt, sowie deren Einfluss auf die Zustandsschätzung. Des Weiteren wird in [36] ein typischer DVL-Fehler mit aufgezeichneten Daten von einem Schiff gezeigt, das in GALILEOnautic 2 als Versuchsplattform diente. Um eine robuste Zustandsschätzung zu erreichen, müssen Sensorausfälle zunächst erkannt und dann entweder identifiziert und ausgeschlossen oder unterdrückt werden. Andernfalls können Sensorausfälle zu fehlerhaften Zustandsschätzungen führen, zu Fehlfunktionen des Navigationssystems und schließlich sogar Unfälle des autonomen Fahrzeugs verursachen. Unter diesem Gesichtspunkt betont [35], dass die Systemintegrität als eine grundlegende Anforderung bei der Entwicklung von Navigationssystemen betrachtet werden sollte.

Bei der Entwicklung eines Integritätssystems sind zwei Aufgaben zu berücksichtigen, [8]:

1. Erkennung und Ausschluss von Fehlern;
2. Integritätsbestimmung der Navigationslösung, d.h. Schutz der Lösung.

3 Integres Navigationsfilter

In GALILEOnautic 2 wurde daher ein Integritätssystem als Erweiterung des Multisensor-Navigationssystems aus Kapitel 2 entwickelt, das INS, GNSS, DVL und Kompass integriert. In der Luftfahrtnavigationen werden üblicherweise Receiver Autonomous Integrity Monitoring (RAIM) und seine Varianten eingesetzt, wobei RAIM ein statistikbasiertes Verfahren ist, das ursprünglich für Pseudorange-Messungen von Satelliten entwickelt wurde. In den letzten Jahren hat sich jedoch im Bereich der maritimen Navigation der Schwerpunkt der Integritätssysteme allmählich von der Integrität des GNSS allein, zur Integrität der Multisensor-Fusion verschoben. Ein Integritätssystem, das die Fault Detection and Exclusion (FDE) aller integrierten Sensoren innerhalb eines Multisensor-Navigationssystems realisiert und die Integrität des gesamten Navigationssystems bestimmt, gibt es jedoch in maritimen Anwendungen nicht.

Um jedoch die FDE aller integrierten Sensoren zu realisieren, muss die FDE des GNSS vorrangig erreicht werden, und zwar aus zwei Gründen: Erstens treten die durch die umgebende Hafen- und Schiffsinfrastruktur verursachten GNSS-Signalstörungen, wie Mehrwegeeffekte und Non-Line-of-Sight (NLOS)-Signale, viel häufiger auf als Ausfälle anderer integrierter Sensoren; zweitens darf das Satellitennavigationssystem nicht als einzelner Sensor betrachtet werden. Stattdessen ist jeder Satellit als eine Informationsquelle zu betrachten, deren Integrität durch Überprüfung der Konsistenz aller verfügbaren Satelliten überwacht werden kann, bevor sie mit den Messungen anderer Sensoren zusammengeführt werden. Mit den eigenen Veröffentlichungen [37], [38] und [14], wurde in GALILEOnautic 2 ein GNSS Fault Detection and Isolation (FDI)-Schema vorgeschlagen, das den konventionellen RAIM-Ansatz und einen auf Intervallmethoden-basierenden alternativen Ansatz Robust Set Inversion via Interval Analysis (RSIVIA) umfasst. Dieses Schema kombiniert die Vorteile beider Ansätze und gleicht gleichzeitig deren Nachteile aus. Des Weiteren wurde mit der eigenen Veröffentlichung [39], ein Fehlerausschlusschema entwickelt, das speziell für maritime Anwendungen konzipiert ist.

Ausgehend von der Annahme, dass die Integrität von GNSS bestätigt wurde, wird die Integrität anderer Sensoren mittels einer Filterbankstruktur (auch bekannt als Multiple Model Adaptive Estimator (MMAE)) überprüft. Die eigene Veröffentlichung [36] beweist die Eignung der Anwendung des Filterbank-Ansatzes für FDI bei Inertial Measurement Unit (IMU)- und DVL-Ausfällen. Darüber hinaus wurde in GALILEOnautic 2 der Ansatz erweitert, um Kompassfehler einzubeziehen, was in diesem Sachbericht dokumentiert wird. Bei der Filterbankstruktur ist kein Fehlerausschlusschema erforderlich, da die Gewichtung der einzelnen Sensoren umgekehrt zu ihrer Fehlerwahrscheinlichkeit ist. Auf diese Weise wird ein Sensor automatisch aus der resultierenden Navigationslösung ausgeschlossen, wenn seine Fehlerwahrscheinlichkeit extrem hoch ist.

Der allerletzte Schritt zur Vervollständigung des Integritätssystems besteht darin, die

Integrität der resultierenden Navigationslösung zu bestimmen, d. h. das Schutzniveau der geschätzten Fahrzeugzustände Position, Geschwindigkeit und Orientierung abzuschätzen. In GALILEOnautic 2 wurde dazu die filterresidualbasierte Methode angewendet.

3.2 Wissenschaftlich-technische Arbeiten

3.2.1 Überblick hybrider Strategie zur GNSS-Integritätsüberwachung

In der vorliegenden Arbeit wird ein FDE-Schema vorgeschlagen, das die Vorteile von RAIM und RSIVIA nutzt und deren Nachteile ausgleicht (siehe Abb. 3.1). Sobald ein neuer Satz von GNSS-Messungen verfügbar ist, werden Fehlererkennung und -identifizierung (FDI) von RAIM iterativ durchgeführt. Schließlich wird eine Fehlermeldung von RAIM generiert und als Auslöser für RSIVIA verwendet, wenn RAIM immer noch einen Fehler erkennt, aber nicht in der Lage ist, ihn zu identifizieren. Dies reduziert die Rechenlast und ermöglicht eine rechtzeitige Fehlererkennung. Wenn RSIVIA für die weitere FDI-Aufgabe ausgelöst wird, wird es in einem iterativen Prozess ausgeführt, der sich auf die RSIVIA-GNSS-Aktualisierung in Abbildung 3.1 bezieht: Es beginnt mit der Annahme, dass kein Fehler im beobachteten Messraum vorliegt. Immer wenn ein leerer Vertrauensbereich zurückgegeben wird, nimmt RSIVIA einen weiteren Fehler in den Messungen an. Dieser iterative Prozess wird so lange fortgesetzt, bis eine nicht leere Vertrauensregion geschätzt wird. Die restlichen fehlerhaften Messungen werden durch Überprüfung der Konsistenz der Messungen mit der resultierenden Vertrauensregion identifiziert.

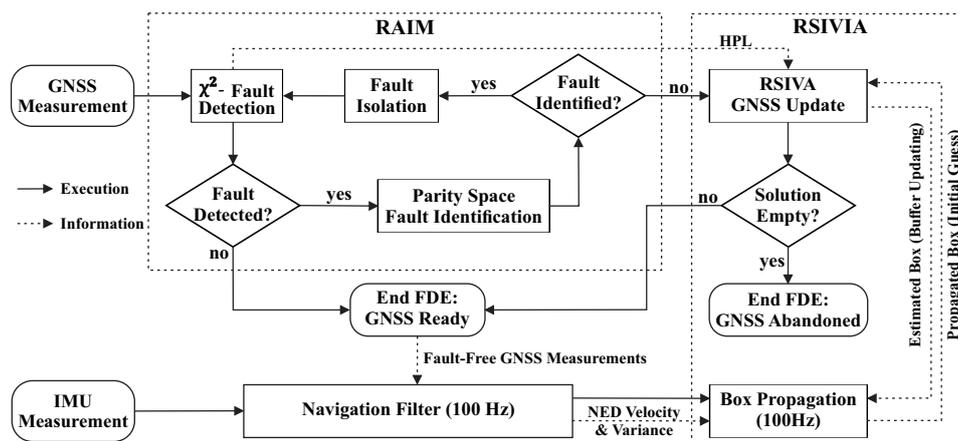


Abbildung 3.1: Schema für GNSS FDE unter Einbeziehung von RAIM und RSIVIA, aus [38]

Um die Rechenlast zu verringern, wird RSIVIA mit einer beliebig großen Box initialisiert,

und die erste Vertrauensregion wird nur mit GNSS-Messungen geschätzt. Danach werden die geschätzten Geschwindigkeitsinformationen und ihre Varianz aus dem Navigationsfilter verwendet, um die Vertrauensregion aus dem letzten Schritt zu propagieren. Auf diese Weise beginnen die weiteren RSIVIA-Schritte mit der propagierten Trust Box als Anfangsschätzung, deren Größe viel kleiner ist als eine beliebige große Box.

3.2.2 RAIM mit Paritätsraum

In der eigenen Veröffentlichung [40] wurde die Entwicklung einer FDE-Erweiterung auf der Grundlage von RAIM vorgestellt. Die notwendigen Gleichungen des auf Residuen basierenden RAIM für die Erkennung von Pseudorange Fehlern sind in [41], [40] angegeben. Die Verwendung von paritätsraumbasiertem RAIM zur Fehlererkennung wird in [42], [40] vorgestellt. Dabei wird nur die notwendige Theorie erläutert, die zum Verständnis der Integration von RAIM in Abschnitt 3.2.3 beiträgt.

Im Allgemeinen verwendet RAIM die Pseudorange-Residuen, um fehlerhafte GNSS-Messungen zu erkennen und zu identifizieren. Die Pseudoranges werden auf der Grundlage der geschätzten Referenzfahrzeugzustände vorhergesagt. Die Pseudorange-Residuen werden als Differenz zwischen gemessenen und vorhergesagten Pseudoranges berechnet. Laut Statistik erfüllt bei ν unabhängigen Standardnormalzufallsvariablen die Summe ihrer Quadrate die Chi-Quadrat-Verteilung mit ν Freiheitsgraden (DOF) - Unter der Annahme, dass das Pseudorange-Messrauschen die Gauß-Verteilung mit weißem Mittelwert und verschiedenen Standardabweichungen erfüllt. Der 3D-Positions- und Uhrenfehler muss zunächst unter Anwendung der Methode der kleinsten Quadrate anhand von Pseudorange-Messungen geschätzt werden. Dabei muss der normierte vorhergesagte Residuum der Pseudoranges die Chi-Quadrat-Verteilung mit $\nu = N - 4$ DOF erfüllen. Wenn die Verteilung nicht erfüllt ist, erklärt RAIM, dass ein Fehler vorliegt. Die Fehleridentifikation erfolgt iterativ mit Hilfe des Paritätsraums unter Verwendung der Bayes-Regel, wobei davon ausgegangen wird, dass alle Satelliten die gleiche vorherige Wahrscheinlichkeit haben, fehlerhaft zu sein, [42], [40]. Es sei darauf hingewiesen, dass RAIM auch das Horizontal Protection Level (HPL) schätzt, das eine Funktion der Pseudorange-Varianzen, der geometrischen Satellitenkonstellation und der vordefinierten Parameter, d. h. der Fehlalarmrate und der Wahrscheinlichkeit einer verpassten Erkennung, ist. Das geschätzte HPL wird in Abschnitt 3.2.3 verwendet, um die minimale Fehlergrenze von RSIVIA zu parametrieren.

3.2.3 RSIVIA und Integration von RAIM

Grundlagen der Intervallanalyse

Die Intervallanalyse (auch Intervallberechnung genannt) ist die Operation auf Intervallen anstelle algebraischer Operation mit Zahlen, obwohl die grundlegenden Operatoren die gleichen sind wie bei algebraischen Operationen: $+$, $-$, \times , \div , \sin , \tan , \exp . Die Berechnung von Intervallen ist in [43] wie folgt definiert:

$$[a] \diamond [b] = [\{a \diamond b \in \mathbb{R} | a \in [a], b \in [b]\}], \quad (3.1)$$

wobei $[a]$ und $[b]$ Intervalle sind. Ein hochdimensionales Intervall ist als Box definiert. \diamond kann jeder der oben gelisteten algebraischen Operatoren sein. Wendet man die Intervallanalyse auf die satellitengestützte Navigation an, wird die Gleichung für die Pseudorange-Messungen wie folgt ausgedrückt:

$$[\rho_i] = \sqrt{([x_{n,i}^s] - [x_n^a])^2 + ([x_{e,i}^s] - [x_e^a])^2 + ([x_{d,i}^s] - [x_d^a])^2} + [c_b], \quad \forall i \in \{1, 2, \dots, N\}, \quad (3.2)$$

wobei $\mathbf{x}^s = [x_{n,i}^s, x_{e,i}^s, x_{d,i}^s]^T$ die i -te Satelliten Position ist. $\mathbf{x}^a = [x_n^a, x_e^a, x_d^a, c_b]^T$ ist die Antennenposition und der Uhrenfehler des Empfängers im NED-Koordinatensystem. N ist die Anzahl der verfügbaren Pseudorange-Messungen.

Messintervallbestimmung

Im Kontext der satellitengestützten Pseudorange-Navigation ist der Messvektor $\tilde{\boldsymbol{\rho}} = [\tilde{\rho}_1, \dots, \tilde{\rho}_i, \dots, \tilde{\rho}_N]$. Ausgehend von den gemessenen Pseudoranges $\tilde{\boldsymbol{\rho}}$ soll deren Intervall $[\boldsymbol{\rho}]$ geschätzt werden. In [44] wird ein Ansatz vorgeschlagen, der die untere und obere Grenze der Messungen schätzt und dabei fehlerhafte Messungen toleriert. Das Zulassen einer bestimmten Anzahl fehlerhafter Messungen zur Schätzung der Vertrauensbox ist als q -relax definiert, wobei q die Anzahl der tolerierten fehlerhaften Messungen ist.

Betrachten wir eine Menge von N verfügbaren Messungen mit der Bedingung q -relax, d.h. mindestens $N - q$ Messungen müssen fehlerfrei sein. Es wird angenommen, dass die Wahrscheinlichkeit, dass eine Messung fehlerhaft ist, der Binomialverteilung entspricht. Die Wahrscheinlichkeit, dass die q -relax-Bedingung erfüllt ist, wird berechnet als:

$$P(n_{ff} \geq N - q) = \sum_{k=N-q}^N P(n_{ff} = k) = \sum_{k=N-q}^N \frac{N!}{k!(N-k)!} p_{ff}^k (1 - p_{ff})^{N-k}, \quad (3.3)$$

wobei das Konfidenzniveau $P(n_{ff} \geq N - q)$ vordefiniert ist. Daher kann p_{ff} als einzige Unbekannte in Gleichung (3.3) geschätzt werden, was die Wahrscheinlichkeit darstellt,

3 Integres Navigationsfilter

dass jeder Satellit fehlerfrei ist. Die Wahrscheinlichkeitsdichtefunktion des Messrauschens wird als $f(e)$ bezeichnet, die praktisch als weiße Gaußverteilung angenommen wird. Auf diese Weise wird für jede Messung p_{ff} berechnet als

$$p_{ff} = P(\rho \in [\tilde{\rho} + a, \tilde{\rho} + b]) = \int_a^b f(e) de. \quad (3.4)$$

Mit p_{ff} aus Gl. (3.3), können die untere und obere Grenze durch Minimierung der Breite des Intervalls $[a, b]$ berechnet werden.

RSIVIA mit GNSS-Messungen

Um die Vertrauensbox des Zustandsvektors \mathbf{x}^a zu schätzen, beginnt RSIVIA mit der möglichen Anfangsschätzung $[\mathbf{x}_0^a]$, die beliebig groß sein kann und die wahre Lösung von \mathbf{x}^a als Teil des Intervalls garantiert. RSIVIA versucht, die Größe der Anfangsschätzung mit einem Kontraktor \mathcal{C} zu reduzieren. Ein Kontraktor ist ein Operator $\mathbb{I}\mathbb{R}^n \rightarrow \mathbb{I}\mathbb{R}^n$, die eine Box $\mathcal{C}[\mathbf{x}] \subseteq [\mathbf{x}]$ zurückgibt. Die Operation ist mit einer Nebenbedingung (in unserem Fall Gl. (3.2)) beschränkt und verliert keinen Vektor, der konsistent zu der Nebenbedingung ist [45]. Wenn die Größe der aktiven Box nicht weiter durch einen Kontraktor reduziert werden kann, wird sie in zwei kleine Boxen halbiert und die Kontraktor-Operation wird für alle verbleibenden kleinen Boxen wiederholt. Dieser Prozess endet, bis die Breite aller verbliebenen Boxen kleiner als eine vordefinierte Fehlergrenze ϵ ist. Der detaillierte Aufbau des RSIVIA-Verfahrens in GNSS-Anwendungen ist in [46] und [44] gegeben und umfasst eine Vorwärts- und Rückwärtskontraktion unter Verwendung der in Gleichung (3.2) angegebenen Einschränkungen. Diese RSIVIA-Operation wird als Zeile 4 in Alg. 1 zusammengefasst.

Alg. 1 gibt den gesamten Prozess einer Boxbegrenzungs-Aktualisierung an, wenn eine neue GNSS-Messung verfügbar ist. Dieser Prozess beginnt mit einer fehlerfreien Annahme ($q = 0$) und versucht, die Vertrauensregion mit zunehmendem q zu schätzen. Dieser Vorgang wird unter dem Begriff GNSS-Aktualisierung zusammengefasst. Die resultierende Box $[\mathbf{x}^a]$ wird in Gl. (3.2) angewendet, um das Intervall jeder Pseudorange mit der entsprechenden Box der Satellitenposition $[x_{n,i}^s, x_{e,i}^s, x_{d,i}^s]$ vorherzusagen. Die fehlerhafte Messung wird erkannt, wenn das vorhergesagte Pseudorange-Intervall keinen Schnittpunkt mit dem gemessenen Intervall hat. Diese Konsistenzprüfung liefert einen Fehlervektor \mathbf{f} , der aus N Elementen besteht, die 0 oder 1 sind und angeben, ob diese Messung fehlerfrei ist.

Dennoch bleiben in diesem Prozess Fragen offen, z.B. wie die Anfangsschätzung $[\mathbf{x}_0^a]$ und die minimal akzeptable Fehlergrenze ϵ unter Berücksichtigung der Korrektheit der Fehleridentifikation und des Rechenaufwands gewählt werden sollten. Diese Fragen werden in den Abschnitten 3.2.3 und 3.2.3 beantwortet.

Algorithm 1 GNSS Update mittels Pseudorange-Messungen [38]

Function gnss_update(in:[\mathbf{x}_0^a], \mathbf{x}^s , $\boldsymbol{\rho}$, ϵ , out:[\mathbf{x}^a], \mathbf{f})

```

1: Initialization: [ $\mathbf{x}^a$ ]  $\leftarrow \emptyset$ ,  $q \leftarrow 0$ , get number of satellites  $N$ 
2: while ( [ $\mathbf{x}^a$ ] =  $\emptyset$  &  $N - q \geq 4$  ) do
3:   ([ $\mathbf{x}^s$ ], [ $\boldsymbol{\rho}$ ])  $\leftarrow$  get_bounds( $\mathbf{x}^s$ ,  $\boldsymbol{\rho}$ ,  $q$ )
4:   [ $\mathbf{x}^a$ ]  $\leftarrow$  rsivia(in:[ $\mathbf{x}_0^a$ ], [ $\mathbf{x}^s$ ], [ $\boldsymbol{\rho}$ ],  $q$ ,  $\epsilon$ )
5:   if [ $\mathbf{x}^a$ ] =  $\emptyset$  then
6:      $q \leftarrow q + 1$ 
7:   else
8:      $\mathbf{f} \leftarrow$  check_consistency([ $\mathbf{x}^a$ ], [ $\mathbf{x}^s$ ], [ $\boldsymbol{\rho}$ ])
9:   end if
10: end while
11: return  $\mathbf{x}^a$ ,  $\mathbf{f}$ 

```

Integration von Geschwindigkeitsinformationen in RSIVIA

In [45] wird ein allgemeines Vorgehen für die garantierte Integration von Zustandsgleichungen vorgeschlagen. In [45] wird ein Beispiel gegeben, bei dem DVL-Messungen und differentielle Zustandsbeschränkungen verwendet werden, um Röhren von gefahrenen Trajektorien eines autonomen Unterwasserfahrzeugs (AUV) zu schätzen. Eine Röhre ist definiert als eine Hüllkurve, die eine unsichere Trajektorie umschließt. Zur Schätzung der Röhre wird ein differentieller Rohrkontraktor $\mathcal{C}_{\frac{d}{dt}}^{\rightarrow}$ verwendet, der aus einem Vorwärtskontraktor $\mathcal{C}_{\frac{d}{dt}}^{\rightarrow}$ und einem Rückwärtskontraktor $\mathcal{C}_{\frac{d}{dt}}^{\leftarrow}$ besteht, [45].

Aus mehreren Gründe wurde in GALILEONautic 2 nur die Vorwärtskontraktion $\mathcal{C}_{\frac{d}{dt}}^{\rightarrow}$ unter Verwendung der Ausgangsgeschwindigkeitsinformationen des Navigationsfilters angewendet. Die ausführliche Diskussion dazu befindet sich in [38].

$$[\mathbf{x}^a](t + dt) = [\mathbf{x}_0^a](t + dt) \cap ([\mathbf{x}^a](t) + dt \cdot [\dot{\mathbf{x}}^a](t)). \quad (3.5)$$

Alle Operationen werden in NED-Koordinaten durchgeführt, so dass $\dot{\mathbf{x}}^a$ bereits geschätzt ist: $\dot{\mathbf{x}}^a = [\mathbf{v}_{cb}^n \ c_d]^T$. Die Begrenzungs-Box $[\dot{\mathbf{x}}^a]$ wird mit dem Output $\dot{\mathbf{x}}^a$ und seiner geschätzten Kovarianz aus dem Navigationsfilter geschätzt. Der detaillierte Entwurf des Kontraktors und die Diskretisierung sind in [45] angegeben.

In der Praxis wird die GNSS-Messung mit einer Rate von ca. 10 Hz und mit einer Zeitverzögerung von 50 ~ 300 Millisekunden aktualisiert. Diese Zeitverzögerung ist das Ergebnis der Signalverarbeitung durch den GNSS-Empfänger, nachdem die Satellitensignale von der Antenne empfangen wurden. Das Navigationsfilter läuft jedoch mit 100 Hz, da für autonome Fahrzeuge eine hochfrequente Navigationslösung erforderlich ist. In Anbetracht

3 Integres Navigationsfilter

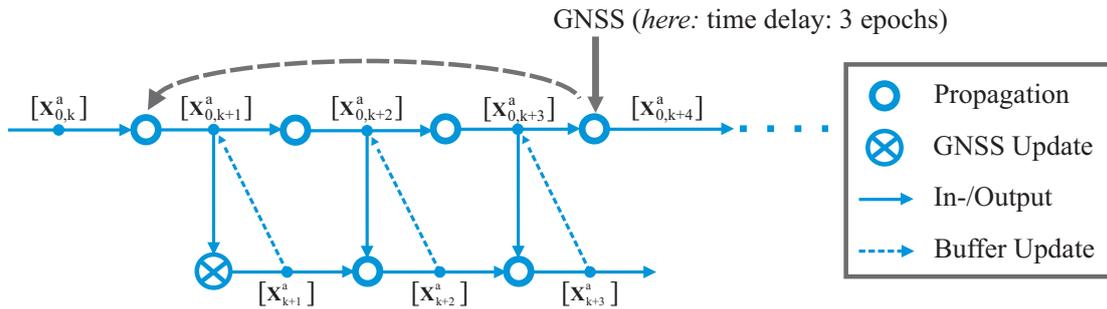


Abbildung 3.2: Verbreitung und GNSS-Aktualisierung unter Berücksichtigung der Messverzögerung, aus [38]

der nicht zu vernachlässigenden Verzögerung wird in Abb. 3.2 eine Struktur vorgeschlagen, die als Beispiel eine GNSS-Zeitverzögerung von 30 Millisekunden annimmt.

Abb. 3.2 zeigt, dass, wenn kein GNSS-Update verfügbar ist, die Box im $k + 1$ -Schritt mit Hilfe von $[\mathbf{x}_k^a]$ und $[\dot{\mathbf{x}}_k^a]$ aus dem letzten Schritt propagiert wird. Ohne zusätzliche Information über die Anfangsbox im $k + 1$ -Schritt wird die Anfangsbox $[\mathbf{x}_{0,k+1}^a]$ als $[-\mathbf{inf}, \mathbf{inf}]$ angenommen, so dass der Teil nach dem \cap -Symbol in Gl. 3.5 die Berechnung dominiert. Sobald eine GNSS-Aktualisierung verfügbar ist, wird der Zeitpunkt der GNSS-Messung durch Vergleich der aktuellen Zeit und der gemessenen Zeitverzögerung geschätzt. Die im Buffer gespeicherte propagierte Box wird lokalisiert und als Anfangsschätzung $[\mathbf{x}_0^a]$ für die GNSS-Aktualisierung verwendet, die in Abschnitt 3.2.3 beschrieben wird. Nach der GNSS-Aktualisierung werden alle Boxen bis zur aktuellen Epoche erneut propagiert und der entsprechende Buffer wird ersetzt.

Abb. 3.3 zeigt ein eindimensionales Beispiel, das die Änderung der geschätzten oberen und unteren Grenze vor und nach der GNSS-Aktualisierung veranschaulicht. In diesem Beispiel wird die GNSS-Messung zur Epoche $k + 4$ (aktuell) mit einer Zeitverzögerung von 3 Epochen empfangen, so dass die GNSS-Aktualisierung in der Vergangenheit (zur Epoche $k + 1$) durchgeführt wird. Die Grenzen zur $k + 1$ -Epoche (schwarz) werden als Anfangsschätzung für die GNSS-Aktualisierung verwendet und engere Grenzen (blau) werden geschätzt. Die blauen Grenzen werden unter Verwendung der gespeicherten Geschwindigkeitsinformationen im Speicher bis zum aktuellen Zeitpunkt ($k + 4$ -Epoche) fortgeschrieben.

RAIM Integration

Durch die Integration von Geschwindigkeitsinformationen beginnt RSIVIA nicht mehr mit einer willkürlichen Box, was die Rechenlast verringert. Dies kann weiter verbessert werden,

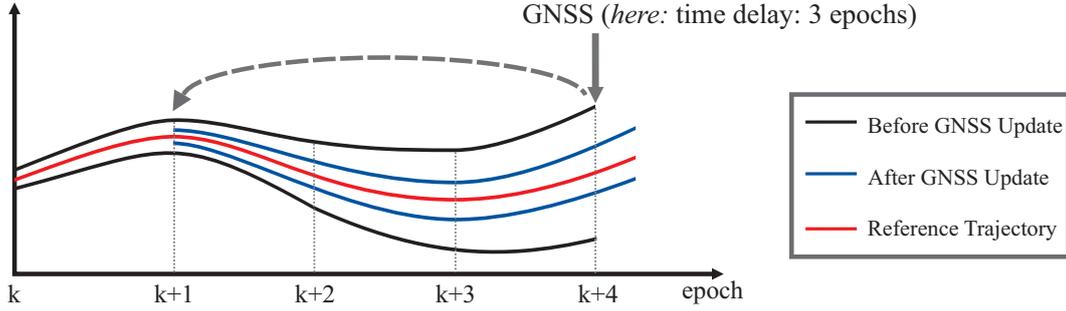


Abbildung 3.3: Eindimensionales (1D) Beispiel für die Integration von Geschwindigkeitsinformationen und die daraus resultierende Varianz bei der GNSS-Aktualisierung, aus [38]

indem RSIVIA nur dann ausgelöst wird, wenn eine GNSS-Aktualisierung erforderlich ist, anstatt es bei jeder neuen GNSS-Messung auszuführen. Ob eine GNSS-Aktualisierung notwendig ist, wird durch das Ergebnis der RAIM-Fehlererkennung entschieden.

Algorithm 2 Integration von Geschwindigkeits- und RAIM-Informationen in RSIVIA im Zeitschritt k [38]

Function estimate_box(in: $\mathbf{x}_k^s, \boldsymbol{\rho}_k, t_d, \mathbf{x}_{k-1}, \mathbf{P}_{k-1}$, out: $[\mathbf{x}_k^a], \mathbf{f}_k$)

- 1: Store \mathbf{x}_{k-1} and \mathbf{P}_{k-1} into buffer
 - 2: **if** New GNSS measurement **then**
 - 3: $(f_a, \epsilon) \leftarrow \text{raim}(\mathbf{x}_k^s, \boldsymbol{\rho}_k, \mathbf{x}_{k-1})$
 - 4: **if** $f_a = \text{true} \mid t_c - t_u > t_{u,max}$ **then**
 - 5: Estimate GNSS delay steps: $n_d \leftarrow t_d/T_0$
 - 6: $([\mathbf{x}_{k-n_d}^a], \mathbf{f}_k) \leftarrow \text{gnss_update}([\mathbf{x}_{k-n_d}^a], \mathbf{x}_k^s, \boldsymbol{\rho}_k, \epsilon)$
 - 7: **for** $i = k - n_d + 1$ to k **do**
 - 8: $([\mathbf{x}_i^a]) \leftarrow \mathcal{C}_{\frac{d}{dt}}^{\rightarrow}(\mathbf{x}_{i-1}, \mathbf{P}_{i-1}, [\mathbf{x}_{i-1}^a])$
 - 9: **end for**
 - 10: **else**
 - 11: $([\mathbf{x}_k^a]) \leftarrow \mathcal{C}_{\frac{d}{dt}}^{\rightarrow}(\mathbf{x}_{k-1}, \mathbf{P}_{k-1}, [\mathbf{x}_{k-1}^a])$
 - 12: **end if**
 - 13: **else**
 - 14: $([\mathbf{x}_k^a]) \leftarrow \mathcal{C}_{\frac{d}{dt}}^{\rightarrow}(\mathbf{x}_{k-1}, \mathbf{P}_{k-1}, [\mathbf{x}_{k-1}^a])$
 - 15: **end if**
 - 16: **return** $[\mathbf{x}_k^a], \mathbf{f}_k$
-

Dennoch sollte eine Situation vermieden werden, in der über einen längeren Zeitraum keine GNSS-Aktualisierung durchgeführt wird, wenn von RAIM kein Fehler erkannt wird. Daher

3 Integres Navigationsfilter

wird ein Parameter $t_{u,max}$ eingeführt, der die maximal zulässige Dauer zwischen zwei GNSS-Aktualisierungen definiert. Die detaillierte Implementierung ist in Alg. 2 gezeigt, wobei t_c und t_u die aktuelle Zeit bzw. die Zeit der letzten GNSS-Aktualisierung ist. t_d ist die Zeitverzögerung der GNSS-Messung. T_o ist die Abtastzeit des Navigationsfilters.

Außerdem wird in Alg. 2 eine neue Methode zur Parametrierung der minimalen Fehler-schranke ϵ von RSIVIA vorgeschlagen, die ein Parameter ist, der in Alg. 1 verwendet wird. In [47] wird einen Solver mit Name *GOMNE* verwendet, der die Fehlergrenze auf die Hälfte des vorherigen Wertes ($\epsilon \leftarrow \epsilon/2$) reduziert, wenn das q -relax steigt. Dennoch musste der Anfangswert von ϵ bei dieser Methode parametrierung werden. [48] diskutiert die Wahl der Fehlerschranke und der Pseudorange-Qualität durch den Vergleich verschiedener Einstellungen von ϵ . Eine deterministische Berechnung der Fehlerschranke ϵ ist jedoch nicht gegeben. Durch die Einbeziehung der Ausgabe von RAIM werden sowohl die Satellitenkonstellation als auch die Messqualität berücksichtigt (Sec. 3.2.2).

3.2.4 Filterbank zur Fehlererkennung integrierter Sensoren

Überblick über die eng-gekoppelte Filterbank

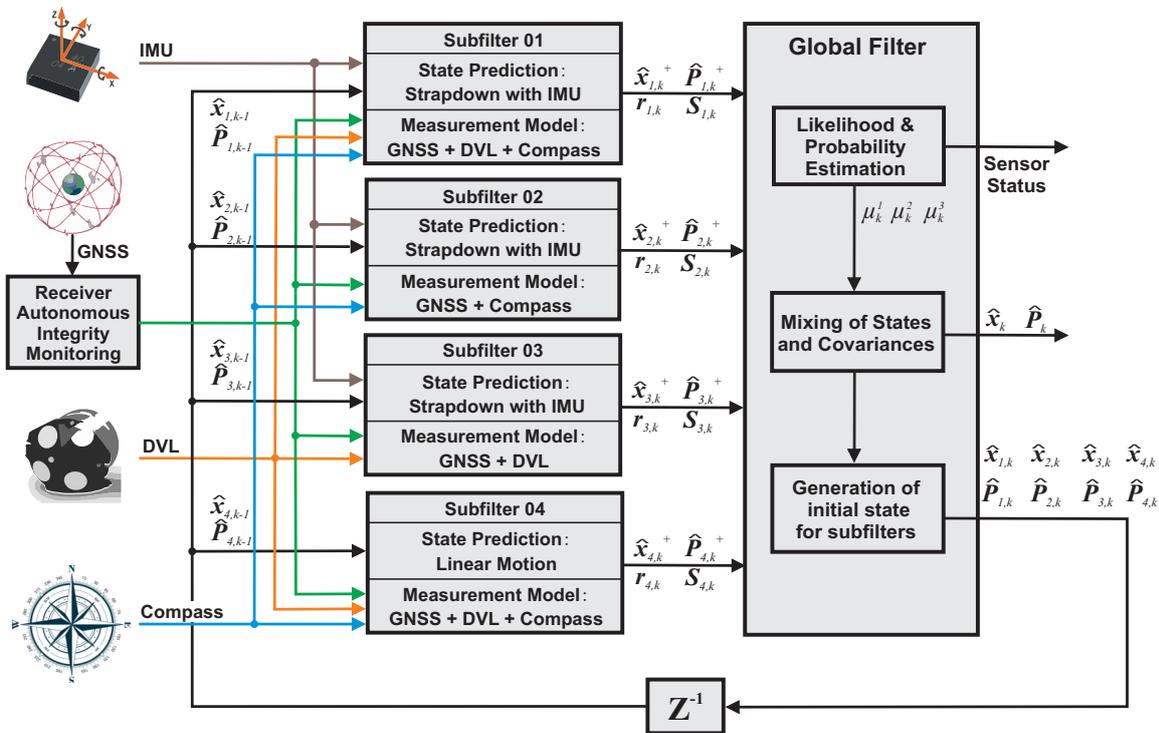


Abbildung 3.4: Konzept einer Filterbankstruktur mit IMU, GNSS und DVL

Abb. 3.4 zeigt einen Überblick über die Struktur der entworfenen Filterbank unter Berücksichtigung möglicher IMU-, DVL- und Kompass-Ausfälle. In Abb. 3.4 bezieht das erste Subfilter 01 alle verfügbaren Messungen (IMU, GNSS, DVL und Kompass) ein und schätzt Zustände. Subfilter 02 schließt die DVL-Messungen aus, um DVL-Ausfälle zu unterdrücken. Unter der Annahme, dass das Rauschen der DVL-Messungen einer mittelwertfreien Gaußverteilung entspricht und in Subfilter 01 korrekt modelliert ist, deutet eine höhere Wahrscheinlichkeit von Subfilter 02 auf einen potenziellen DVL-Fehler hin. Ähnlich schließt Subfilter 03/04 die Kompass bzw. IMU-Messungen aus, um Kompass- bzw. IMU-Fehler zu unterdrücken.

Wahrscheinlichkeitsrechnung und Wahrscheinlichkeitsschätzung

Auf Grundlage der Kalman-Filter-Zustands- und Kovarianzschätzung ist die Wahrscheinlichkeit jedes Teilfilters

$$p_{i,k}(\tilde{\mathbf{z}}_{i,k} | \mathbf{x}_{i,k}^-) = \frac{1}{\sqrt{(2\pi)^L |\mathbf{S}_{i,k}|}} e^{-\frac{1}{2} \mathbf{r}_{i,k}^T \mathbf{S}_{i,k}^{-1} \mathbf{r}_{i,k}}, \quad (3.6)$$

die die Wahrscheinlichkeitsdichte der aktuellen Messung $\tilde{\mathbf{z}}_{i,k}$ bezüglich des propagierten a priori Zustandsvektors $\mathbf{x}_{i,k}^-$ des Subfilters i berechnet. Dabei ist $i \in \{1, \dots, m\}$ die Folge der Subfilter und m die Anzahl der Subfilter. $p_{i,k}$ stellt die Ähnlichkeit zwischen der Messung und den a priori geschätzten Zuständen dar. Dabei ist L die Länge des Messvektors $\tilde{\mathbf{z}}_k$, $\mathbf{r}_{i,k}$ ist der Innovationsvektor der Messung und $\mathbf{S}_{i,k}$ ist seine geschätzte Kovarianzmatrix:

$$\begin{aligned} \mathbf{r}_{i,k} &= \tilde{\mathbf{z}}_{i,k} - \mathbf{H}_{i,k} \mathbf{x}_{i,k}^-, \\ \mathbf{S}_{i,k} &= \mathbf{H}_{i,k} \mathbf{P}_{i,k}^- \mathbf{H}_{i,k}^T + \mathbf{R}_{i,k}, \end{aligned} \quad (3.7)$$

wobei $\mathbf{P}_{i,k}^-$ die A-priori-Schätzung der Zustands-Kovarianz ist.

Die Wahrscheinlichkeit der Subfilter wird geschätzt, wobei die resultierende Wahrscheinlichkeit aus dem letzten Schritt berücksichtigt und μ_k^i mit der berechneten Wahrscheinlichkeit aktualisiert wird, [20]:

$$\mu_k^i = \frac{p_{i,k}(\tilde{\mathbf{z}}_{i,k} | \mathbf{x}_{i,k}^-) \mu_{k-1}^i}{\sum_{i=1}^m p_{i,k}(\tilde{\mathbf{z}}_{i,k} | \mathbf{x}_{i,k}^-) \mu_{k-1}^i}. \quad (3.8)$$

Die Anwendung fehlerhafter Sensormessungen, z.B. DVL, führt zu einer größeren Messinnovation $\mathbf{r}_{i,k}$ der Subfilter 01 und 03 (Gl. (3.7)). Somit nimmt die Wahrscheinlichkeit $p_{i,k}$ der Subfilter 01 und 03 logarithmisch mit der Amplitude ihrer Messinnovation $\mathbf{r}_{i,k}$ ab (Gl. (3.6)). Infolgedessen hat die Wahrscheinlichkeit des Teilfilters ohne fehlerhafte Messung,

3 Integres Navigationsfilter

d. h. Teilfilter 02, bei Anwendung von Gleichung (3.8) die höchste Wahrscheinlichkeit (ungefähr 1), was auf DVL-Ausfälle hinweist.

Generierung eines gemeinsamen Zustandsvektors:

Unter Verwendung der resultierenden Wahrscheinlichkeit als Gewichtung werden die Ausgangszustände $\bar{\mathbf{x}}_k$ und ihre Kovarianzen $\bar{\mathbf{P}}_k$ der Filterbank wie folgt erzeugt:

$$\begin{aligned}\bar{\mathbf{x}}_k &= \sum_{i=1}^m \mu_k^i \mathbf{x}_{i,k}^+ , \\ \bar{\mathbf{P}}_k &= \sum_{i=1}^m \mu_k^i (\mathbf{P}_{i,k}^+ + (\bar{\mathbf{x}}_k - \mathbf{x}_{i,k}^+)(\bar{\mathbf{x}}_k - \mathbf{x}_{i,k}^+)^T) .\end{aligned}\tag{3.9}$$

Subfilter Initialisierung

Das Interagierende Mehrfachmodell (IMM) erfordert, dass die Subfilter nach einem Mischschritt jedes Mal anders initialisiert werden [20]. Die normalisierte Zustandswahrscheinlichkeit, die zur Gewichtung der Zustände verwendet wird, wird wie folgt berechnet:

$$\mu_k^{ji} = \frac{M_{ij} \mu_k^j}{\sum_{j=1}^m M_{ij} \mu_k^j}\tag{3.10}$$

wobei i und j die Sequenz der Subfilter im aktuellen bzw. letzten Schritt sind. Der Sequenzübergang in der Filterbank wird als Markov Jump Linear System (MJLS) angenommen, das durch eine Übergangswahrscheinlichkeitsmatrix \mathbf{M} beschrieben wird, wobei jedes Element M_{ij} die Wahrscheinlichkeit definiert, dass die Sequenz des Subfilters von i zu j [20] wechselt. In der Praxis ist \mathbf{M} ein vordefinierter Parameter. Der Zustandsvektor $\bar{\mathbf{x}}_{i,k}$ und seine Kovarianz $\bar{\mathbf{P}}_{i,k}$, die zur Initialisierung des Subfilters i -te beim nächsten Schritt verwendet werden, werden wie folgt berechnet:

$$\begin{aligned}\bar{\mathbf{x}}_{i,k} &= \sum_{j=1}^m \mu_k^{ji} \mathbf{x}_{j,k}^+ , \\ \bar{\mathbf{P}}_{i,k} &= \sum_{j=1}^m \mu_k^{ji} (\mathbf{P}_{j,k}^+ + (\mathbf{x}_{j,k}^+ - \bar{\mathbf{x}}_{i,k})(\mathbf{x}_{j,k}^+ - \bar{\mathbf{x}}_{i,k})^T).\end{aligned}\tag{3.11}$$

Multi-Rate-Sensormessungen und Messverzögerung

Die in das Navigationssystem integrierten Sensoren werden mit unterschiedlichen Raten abgetastet und haben Messverzögerungen. Die detaillierten Abtastraten der verwendeten Sensoren sind in Abschnitt 3.3.2 angegeben. In Abb. 3.5 wird eine neue Struktur der Filterbank zur Lösung dieses Problems gezeigt die in GALILEOnautic 2 erarbeitet wurde. Sie zeigt den Zustandsübergang innerhalb des entworfenen Filterbankkonzepts anhand

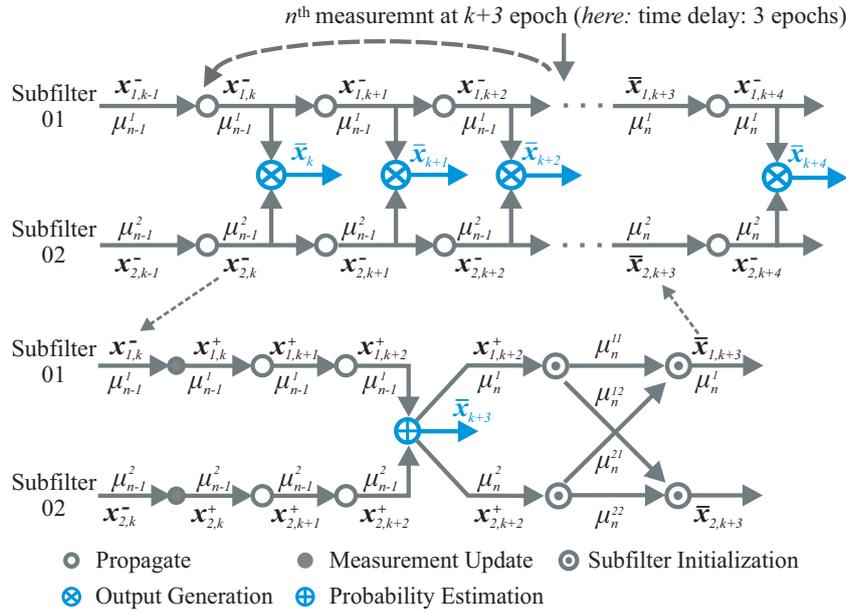


Abbildung 3.5: Filterbank: Zustandsübergang (grau) und Ausgangsgenerierung (blau) innerhalb der Filterbank: Beispiel mit 2 Subfiltern, die sich mit Multiraten-Sensormessungen und Messverzögerungen befassen, aus [36]

eines Beispiels mit zwei Subfiltern, um es einfach zu halten. Die Methodik lässt sich jedoch problemlos auf drei oder mehr Subfilter erweitern.

In Abb. 3.5 werden die Navigationszustände so lange propagiert, bis die n -te Messung verfügbar ist. Davor wird die Filterbankausgabe mit Hilfe von Gl. (3.9) mit der gespeicherten Wahrscheinlichkeit μ_{n-1}^i erzeugt, die durch die $(n-1)$ -te Messupdate geschätzt wird. Wenn die n -te Messung zur $k+3$ -Epoche eintrifft, wird die Zeitverzögerung dieser Messung als 3 Epochen berechnet. Daher wird die Messwertkorrektur zur Epoche k durchgeführt, und die Zustände werden bis zur aktuellen Epoche $k+3$ fortgeschrieben. Dabei werden die Wahrscheinlichkeiten der Subfilter mit Hilfe von Gl. (3.8) auf μ_n^i aktualisiert und die Anfangszustände der Subfilter für den nächsten Schritt mit Hilfe von Gl. (3.11) berechnet. In dieser Struktur wird die Ausgabe der Filterbank in jedem Schritt erzeugt, während die Subfilter nur dann neu initialisiert werden, wenn neue Messungen vorliegen.

3.2.5 Gesamtintegrität der Navigationslösung

Das Protection Level (PL) ist eine statistische Zustandsfehlergrenze, die berechnet wird, um zu gewährleisten, dass die Wahrscheinlichkeit des absoluten Zustandsfehlers kleiner oder gleich dem Integritätsrisiko (IR) ist. Die bestimmte Wahrscheinlichkeit dieser Ober-

3 Integres Navigationsfilter

grenze ist das Integritätsrisiko. Es gibt mehrere Varianten des PL, z. B. den auf Snapshot-Resten basierenden Integritätsüberwachungsalgorithmus, wie z.B. RAIM, oder das EKF-basierte PL [49].

In GALILEOnautic wurde die folgende neigungs-basierte HPL in unserem EKF-Integritätsüberwachungsschema verwendet.

$$PL_x = \max(Xslope_i \sigma_i) \sqrt{\lambda_k^2} + K(P_{md}) \sqrt{P_x}, \quad (3.12)$$

wobei P_x die Varianz des Zustands x ist. $K(P_{md})$ ist ein Inflationsfaktor, um das spezifizierte Integritätsrisiko zu erfüllen. $xslope_i$ ist die Empfindlichkeit des Zustandsfehlers gegenüber der Fehler des i -ten Satelliten. σ_i ist die Standardabweichung der i -ten Messung. Es gilt:

$$Xslope_{k,i,x} = \sqrt{\frac{\mathbf{K}_k^T \mathbf{x}_x^T \mathbf{x}_x \mathbf{K}_k}{\mathbf{S}_k^{-1}}}, \quad (3.13)$$

wobei \mathbf{x}_x ein Vektor ist, der es ermöglicht, den gewünschten Zustand aus dem vollständigen Zustandsvektor auszuwählen.

Das PL kann mit der Filterbank gemischt werden:

$$PL_{mixed} = \sum_{i=1}^m \mu_k^{i_k} PL_{i,k}, \quad (3.14)$$

wobei $PL_{i,k}$ der PL des Subfilters i zum Zeitpunkt k ist.

Horizontal Protection Level

Ein besonderes Protection Level ist das Horizontal Protection Level (HPL), das eine statistische Positionsfehlergrenze darstellt. Das HPL ist eines der am häufigsten verwendeten PL, das mit Referenzdaten zudem gut validiert werden kann. Das HPL ist eine Kombination aus dem östlichen und dem nördlichen Positionsfehler.

Das folgende neigungs-basierte HPL wird in unserem EKF-Integritätsüberwachungsschema verwendet.

$$HPL = \max(Hslope_i \sigma_i) \sqrt{\lambda_k^2} + K(P_{md}) \sqrt{(P_{ee} + P_{nn})}, \quad (3.15)$$

wobei P_{ee} und P_{nn} die Positionsabweichungen in östlicher bzw. nördlicher Richtung sind. $K(P_{md})$ ist ein Inflationsfaktor, der verwendet wird, um das angegebene Integritätsrisiko

zu erfüllen. $Hslope_i$ steht für die Empfindlichkeit des horizontalen Positionsfehlers (HPE) gegenüber dem Bias des i -ten Satelliten. σ_i ist die Standardabweichung der i -ten Messung.

$$Hslope_{k,i} = \sqrt{\frac{\mathbf{f}_i^T \mathbf{K}_k^T \mathbf{x}_e^T \mathbf{x}_e \mathbf{K}_k \mathbf{f}_i + \mathbf{f}_i^T \mathbf{K}_k^T \mathbf{x}_n^T \mathbf{x}_n \mathbf{K}_k \mathbf{f}_i}{\mathbf{f}_i^T \mathbf{S}_k^{-1} \mathbf{f}_i}} \quad (3.16)$$

wobei x_e und x_n Vektoren sind, die es erlauben, den gewünschten Zustand aus dem vollständigen Zustandsvektor auszuwählen. Der Fehlervektor \mathbf{f}_i ist:

$$\mathbf{f}_i = E[\mathbf{r}_i]. \quad (3.17)$$

Die Ableitungen des horizontalen Schutzniveaus sind in [49] zu finden. Der HPL_{mixed} wird berechnet mit:

$$HPL_{mixed} = \sum_{i=1}^m \mu_k^{i_k} HPL_{i,k}, \quad (3.18)$$

wobei $HPL_{i,k}$ die HPL des Subfilters i zum Zeitpunkt k ist.

3.3 Experimentelle Validierung

3.3.1 GNSS-Fehlererkennung und Ausschluss

Versuchsaufbau

Abb. 3.6 zeigt die Verteilung der Sensoren auf dem Testfahrzeug. In diesem Messaufbau werden zwei Antennen verwendet. Eine davon wird vom GATE-System verwendet, um eine Referenzposition des Testbusses für die Erzeugung von GNSS-Beobachtungsdaten für die simulierten virtuellen Satelliten zu bestimmen. Die andere Antenne wird vom Navigationsfilter verwendet, um die Satellitensignale von GPS, Galileo und den virtuellen Satelliten zu empfangen. Das getestete Navigationssystem arbeitet unabhängig von der GATE-Antenne. Die Position der GNSS-Antenne, die vom Navigationsfilter im IMU-Koordinatensystem verwendet wird, wird gemessen und in Abb. 3.6 dargestellt, was für die Umwandlung der gemessenen GNSS-Beobachtungsdaten in den IMU-Koordinatensystem erforderlich ist. Die Sensordaten werden auf einer 900-MHz-Single-Core-RCP-Einheit (Rapid Control Prototyping), der MicroAutoBox II von dSPACE, aufgezeichnet. Die GNSS-Signale werden von einem Septentrio AstRx3 HDC-Empfänger mit einer Rate von 10 Hz empfangen und dekodiert. Die Kommunikation zwischen dem Empfänger und der RCP-Einheit erfolgt über eine serielle Schnittstelle. Für Inertialmessungen wird ein LORD

3 Integres Navigationsfilter

MicroStrain 3DM-GX4-25 IMU-Sensor der Industrieklasse mit 9 DOF verwendet, der über eine serielle Schnittstelle angeschlossen ist und Beschleunigungen, Winkelgeschwindigkeiten und Magnetometermessungen mit einer Rate von 100 Hz liefert. Der Empfänger liefert einen Impuls pro Sekunde (PPS). Mit Hilfe des PPS werden die Kommunikations- und Verarbeitungsverzögerungen des Empfängers gemessen. Um das reale Testszenario in einer Post-Processing-Umgebung zu reproduzieren, werden alle Sensoren und GNSS-Korrekturdaten aufgezeichnet.



Abbildung 3.6: Messaufbau und Sensorverteilung am Testbus der IFEN GmbH, aus [38]

Abb. 3.7 zeigt die Vogelperspektive der gefahrenen Trajektorie. Das Experiment wird im sogenannten „T-Cross“ in Berchtesgaden durchgeführt, weil es die beste Teststrecke für die Sichtbarkeit aller drei Basisstationen des GATE-Systems ist. Es sollte beachtet werden, dass sich die gefahrene Strecke in diesem Experiment in offenem Gelände befindet. Daher wird davon ausgegangen, dass die Messungen von anderen Satelliten, abgesehen von den erzeugten Ereignissen, fehlerfrei sind. Das Testszenario ist jedoch nur durch die Wiedergabe der aufgezeichneten Daten in der Post-Processing-Umgebung reproduzierbar, da sich die Umgebung in der Realität ändert, z. B. die Position der realen Satelliten, die Verzögerung in der Ionosphäre und der Troposphäre. Die Post-Processing-Umgebung läuft auf der MATLAB- & Simulink-Plattform auf einem Laptop mit einer Intel Core i7-7700HQ CPU @ 2,80 GHz.

Während dieser Fahrt treten zwei Ereignisse auf, bei denen es sich um vom GATE-System absichtlich erzeugte Entfernungfehler handelt. Diese Fehler werden in Abb. 3.8 visualisiert, die die Pseudorange-Residuen der GPS- und Galileo-Signale während dieser Fahrt zeigt. Die Pseudorange-Residuen werden als Differenz zwischen den gemessenen und den wahren Pseudoranges berechnet. Hier werden die wahren Pseudoranges mit der hoch genauen RTK-Referenzlösung und den Satellitenpositionen geschätzt. Abb. 3.8 zeigt, dass von Sekunde 108 bis 172 und von Sekunde 188 bis 252 die Pseudorange-Residuen der vier Galileo-Satelliten E10, E16, E17 und E23 extrem hoch sind. Dies sind die beiden Zeiträume, in denen gefürchtete Ereignisse auftreten, was durch den Experiments-Bericht der



Abbildung 3.7: Referenztrajektorie unter Verwendung eines RTK-fähigen GNSS-Empfängers, ©2019 GeoBasis-DE/BKG (©2009), Google, aus [38]

IFEN GmbH bestätigt wird. Diese beiden Zeiträume sind in den folgenden Abbildungen durch die grauen gestrichelten Linien gekennzeichnet.

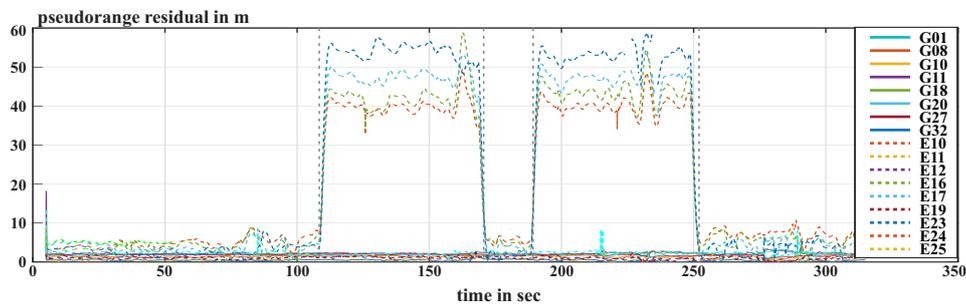


Abbildung 3.8: Pseudorange-Residuen, aus [38]

Ergebnisse

In [50] wird im Kapitel über die experimentelle Validierung die Situation beschrieben, dass RAIM mit Paritätsraum nicht in der Lage ist, alle Fehler in einem sehr kurzen Zeitraum zu identifizieren, während RSIVIA in der Lage ist, alle fehlerhaften Pseudoranges zu identifizieren. Der Vergleich zwischen RAIM und RSIVIA unter diesen Bedingungen ist in [50] enthalten und wird daher in der vorliegenden Veröffentlichung nicht wiederholt. In diesem Abschnitt werden die GNSS-Aktualisierung sowie die Integration von Geschwindigkeitsinformationen und RAIM im Hinblick auf die Korrektheit der Identifizierung, die

3 Integres Navigationsfilter

Verbesserung der Genauigkeit und den Rechenaufwand bewertet. Um diese zu vergleichen, werden in diesem Abschnitt drei Einstellungen verwendet:

- Einstellung 04: Die GNSS-Aktualisierung erfolgt mit 1 Hz. Jeder RSIVIA-Prozess beginnt mit einer anfänglichen Schätzung einer beliebigen großen Box.
- Einstellung 05: Die GNSS-Aktualisierung erfolgt mit 1 Hz. Die Vertrauensbox wird mit Geschwindigkeitsinformationen propagiert (Abschnitt 3.2.3). RSIVIA startet mit der propagierten Trustbox.
- Einstellung 06: Der gesamte in Abschnitt 3.2.3 vorgeschlagene Ansatz wird hier angewendet. Falls keine RAIM-Fehlermeldung vorliegt, beträgt die maximale Dauer ohne GNSS-Aktualisierung $t_{u,max}$ 60 Sekunden.

Abb. 3.9 zeigt die experimentellen Ergebnisse. Die erste Unterabbildung zeigt das Ergebnis der Fehlererkennung bei den drei Einstellungen und die Verfügbarkeit der Satelliten während der Testfahrt. Wie bereits in Abb. 3.8 gezeigt, steigt der Pseudorange-Fehler zu Beginn jedes Ereignisses allmählich an. Diese Art des Pseudorange-Fehlers wird in [51] als gefährlichstes Fehlermodell für die Snapshot-Integritätsüberwachung vorgestellt, da die Zustandsschätzung verfälscht wird, bevor der Fehler erkannt wird.

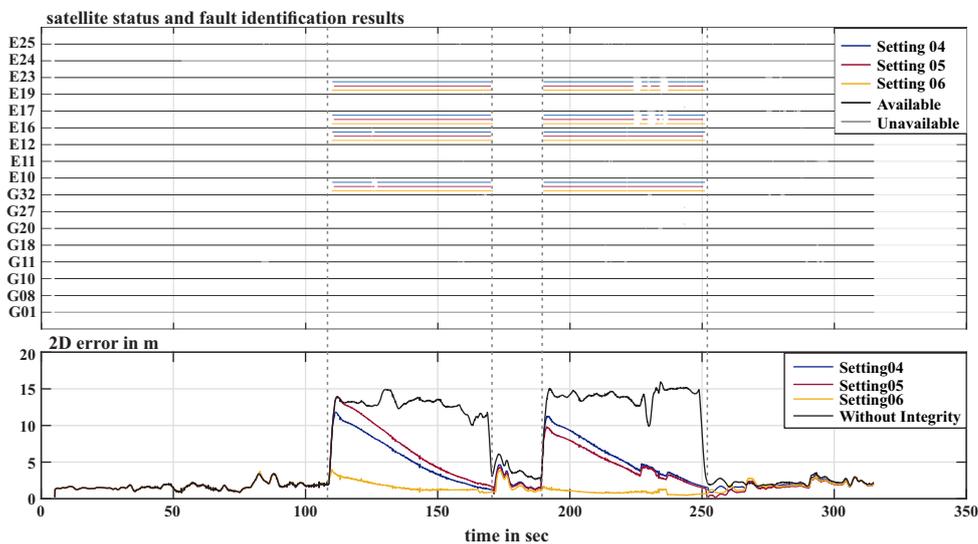


Abbildung 3.9: Validierung der Fehlererkennung und Verbesserung der Genauigkeit, aus [38]

Dies kann anhand von Abb. 3.9 überprüft werden. Wenn das erste Ereignis eintritt (bei 108,4 Sekunden), wird die fehlerhafte Pseudorange zuerst mit Einstellung 06 (bei 109,87 Sekunden), dann mit Einstellung 04 (bei 110,20 Sekunden) und schließlich mit Einstellung

05 (bei 110,67 Sekunden) identifiziert. Der Grund für die Verzögerung bei den Einstellungen 04 und 05 ist, dass die GNSS-Aktualisierung aufgrund der hohen Rechenlast mit 1 Hz erfolgt, während die GNSS-Messungen mit 10 Hz empfangen werden. Außerdem wird die GNSS-Aktualisierung bei Einstellung 06 rechtzeitig durch RAIM ausgelöst. Obwohl der Zeitunterschied bei der ersten Fehlererkennung zwischen den drei Einstellungen gering ist, führt er zu einem erheblichen Unterschied bei der Zustandsschätzung.

Die zweite Unterabbildung zeigt den 2D-Fehler als Maß für die Qualität der Zustandsschätzung. Ohne FDE bleibt der 2D-Fehler während der gefürchteten Ereignisse zwischen 10 und 16 Metern. Mit den Einstellungen 04 und 05 erhöht sich der 2D-Fehler auf 14 bzw. 12 Meter und konvergiert langsam gegen die Genauigkeit ohne gefürchtete Ereignisse. Mit der Einstellung 06 steigt der 2D-Fehler leicht auf 3,7 Meter an, da der Fehler früher erkannt wird, und konvergiert schnell auf 1 Meter.

Abschließend wird die Berechnungszeit bewertet. Die Nachbearbeitung dauert 81,70 Sekunden ohne Integritätsüberwachung, was die Basis für die Berechnungszeit darstellt. Mit den Einstellungen 04, 05 und 06 dauert die Nachbearbeitung 276,68, 171,51 bzw. 121,89 Sekunden. In Anbetracht der Tatsache, dass die GNSS-Aktualisierung sowohl bei Einstellung 04 als auch bei Einstellung 05 mit 1 Hz erfolgt, ist die Anzahl der GNSS-Aktualisierungen gleich. Durch die Einführung der Geschwindigkeitsinformationen zur Verringerung der Größe der anfänglichen Schätzung wird die durchschnittliche Berechnungszeit jeder GNSS-Aktualisierung um 53,94 % verringert, was für die zukünftige Echtzeitimplementierung sehr wichtig ist. Im Gegensatz dazu reduziert RAIM die Gesamtrechenzeit, indem es die Anzahl der GNSS-Aktualisierungen verringert, anstatt die durchschnittliche Rechenzeit zu reduzieren. Daher kann es sein, dass RAIM die Echtzeitberechnungsleistung nicht sehr stark verbessert. Die Einführung von RAIM ermöglicht jedoch eine rechtzeitige Fehlererkennung, was die Genauigkeit der Navigationslösung verbessert.

3.3.2 Sensorübergreifende Fehlererkennung mittels Filterbank

Versuchsaufbau

Abb. 3.10 zeigt die Sensorverteilung am Testfahrzeug und visualisiert den Hebelarm der GNSS-Antenne und des DVL:

$$\begin{aligned} \boldsymbol{l}_a^b &= [\quad 0.050 \text{ m}, \quad -0.398 \text{ m}, \quad -0.935 \text{ m}]^T, \\ \boldsymbol{l}_d^b &= [-0.321 \text{ m}, \quad 0.683 \text{ m}, \quad 0.826 \text{ m}]^T. \end{aligned}$$

Die entworfene Filterbank ist auf einer 900-MHz-Single-Core-RCP-Einheit (Rapid Control Prototyping) dSPACE MicroAutoBox implementiert. Die GNSS-Messdaten werden

3 Integres Navigationsfilter

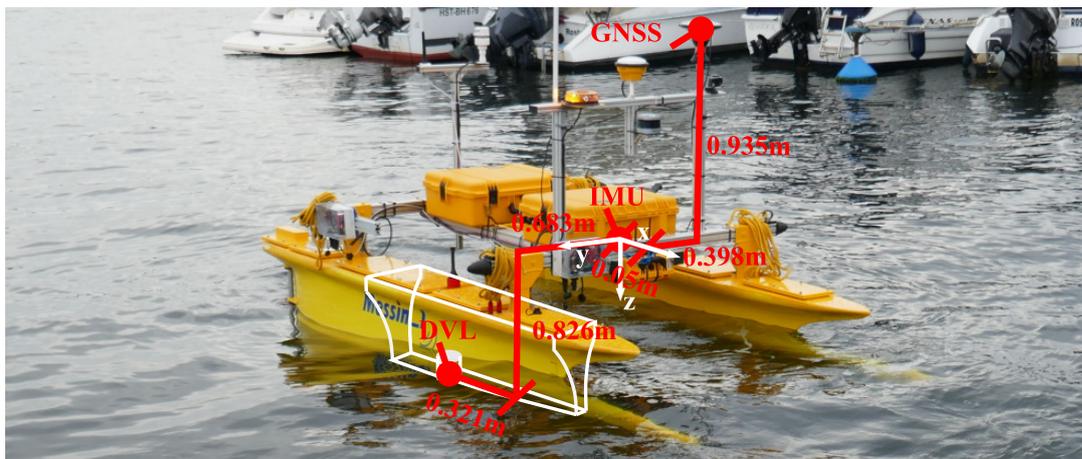


Abbildung 3.10: Messaufbau am Testfahrzeug MESSIN: Verteilung der Sensoren relativ zum IMU-Bezugssystem, wobei sich die IMU im gelben Kasten und das DVL im Rumpf befindet, aus [38]; USV MESSIN ist Eigentum der Universität Rostock

von einem Septentrio AstRx3 HDC-Empfänger mit einer Rate von 10 Hz geliefert. Für die inertielle Messung wird ein LORD MicroStrain 3DM-GX5-25 IMU-Sensor der Industrieklasse verwendet, der Beschleunigungen, Rotationsraten und Magnetometermessungen mit einer Rate von 100 Hz liefert. Sowohl der Empfänger als auch die IMU kommunizieren über eine serielle Schnittstelle mit der RCP-Einheit. Der DVL ist ein NORTEK DVL1000 und ist über Ethernet mit der RCP-Einheit verbunden. Die DVL-Messung wird mit einem vom Empfänger gelieferten PPS ausgelöst. Anhand der PPS werden die Kommunikations- und Verarbeitungsverzögerungen sowohl des Empfängers als auch des DVL geschätzt. Als Referenz wird die Real-Time-Kinematik (RTK) Lösung des Empfängers verwendet.

Ergebnisse bei DVL-Messfehlern

In diesem Abschnitt werden die experimentellen Ergebnisse eines Echtzeit-Validierung-Szenarios vorgestellt. In diesem Test werden verschiedene Formen von DVL-Ausfällen simuliert und zu den Messungen hinzugefügt, die von dem entwickelten Filterbank-Ansatz in der Echtzeit-Testkampagne verwendet werden. In der Zwischenzeit arbeiten alle drei in Abschnitt 3.2.4 genannten Subfilter online, um die Echtzeitfähigkeit des entwickelten Ansatzes auf der Zielhardware zu validieren.

Abb. 3.11 zeigt die gefahrene 2D-Trajektorie in lokalen NED-Koordinaten, deren Ursprung der Startpunkt der Referenztrajektorie ist. Die schwarze Linie ist die RTK-Referenztrajektorie. Die grüne und die orangefarbene Linie sind die geschätzten Trajektorien aus dem entworfenen Filterbank-Ansatz (online) bzw. dem konventionellen EKF (offline). Hier

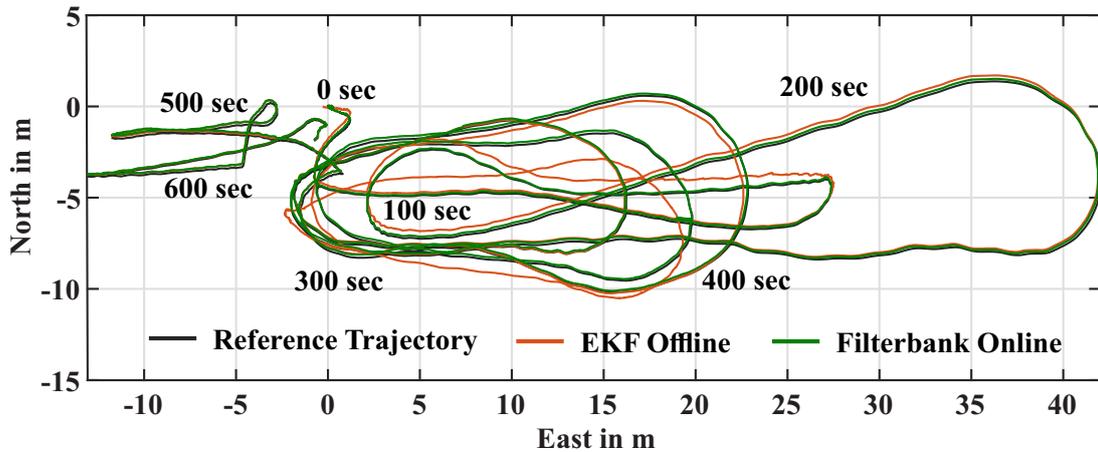


Abbildung 3.11: 2D-Trajektorie im lokalen NED-Koordinatensystem, aus [36]

verwendet der EKF das gleiche Prozess- und Messmodell wie der Subfilter 01 und arbeitet in der Post-Processing-Umgebung mit den aufgezeichneten Daten. Bemerkenswert ist, dass der Subfilter 01 nach dem IMM-Prozess mit der Ausgabe der Filterbank (Gl. (3.11)) initialisiert wird, während der EKF allein ohne Verbindung zur Filterbank arbeitet.

Die erste Unterabbildung in Abb. 3.12 zeigt fünf Formen von Fehlern, die simuliert und zur Geschwindigkeit in y -Richtung des DVL-Bezugssystems addiert werden: Sprung, Rampe, Impuls, Vibration und Achsenversatz, wenn Objekte in Küstennähe oder Meerestiere den Sensor berühren. Die Zeitintervalle, in denen die DVL-Fehler simuliert werden, sind grau hinterlegt.

Die zweite Unterabbildung in Abb. 3.12 zeigt den 2D-Fehlervergleich zwischen EKF (orange) und der entworfenen Filterbank (grün) bei DVL-Ausfällen. Vor dem Auftreten des ersten Fehlers haben sowohl EKF als auch Filterbank eine 2D-Genauigkeit von 0,1 m. Nach dem Auftreten eines Fehlers steigt der 2D-Fehler von EKF auf bis zu 2 m beim Vibrationsfehler. In der Zwischenzeit ist der 2D-Fehler der Filterbank stabil und schwankt zwischen 0,1 und 0,2 m. Bemerkenswert ist, dass es einen Impuls im 2D-Fehler der Filterbank bei etwa 340 Sekunden gibt, der durch eine Berechnungsüberlastung der RCP-Einheit verursacht wird.

Tabelle 3.1: Filterbank: Tabelle der 2D-Fehler in Meter

2D Error in m	Avarage	95% Accuracy	Standard Deviation
EKF	0.329	1.274	0.401
Filter bank	0.123	0.184	0.046

3 Integres Navigationsfilter

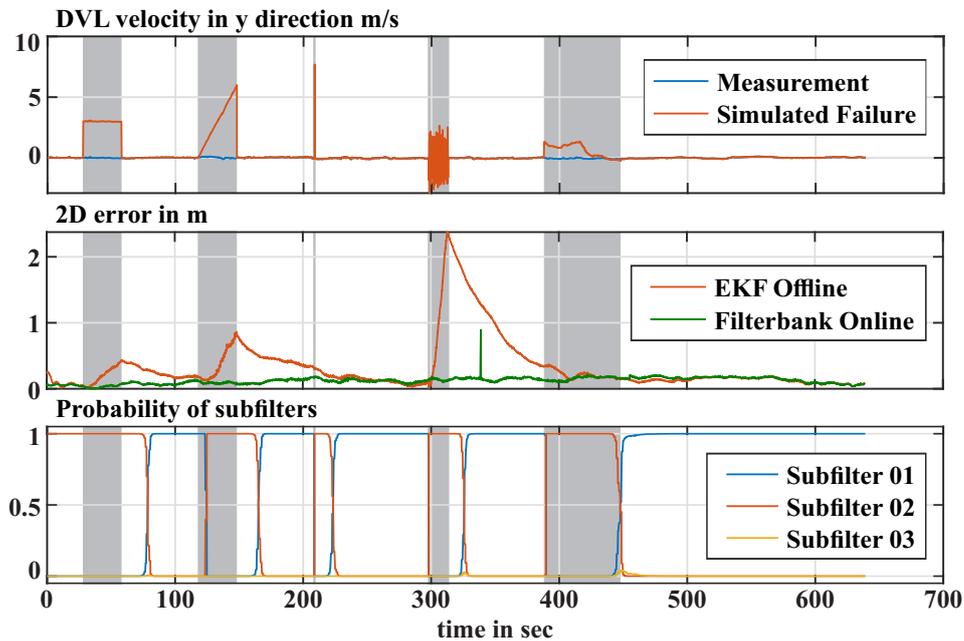


Abbildung 3.12: Ergebnisse der Echtzeit-Bewertung: DVL-Messung und simulierte Ausfälle (Schritt, Rampe, Impuls, Vibration und Fehlausrichtung), 2D-Fehler und geschätzte Wahrscheinlichkeit von drei Teilfiltern, aus [36]

Die verschiedenen Genauigkeitsindikatoren sind in Tab. I. Im Allgemeinen verbessert die Filterbank die durchschnittliche 2D-Genauigkeit um 167%, die 95%ige Genauigkeit, die für sicherheitskritische Anwendungen wichtig ist, um 592% und die Standardabweichung, die für Führungs- und Kontrollsysteme in autonomen Anwendungen wichtig ist, um 771%.

Die dritte Unterabbildung in Abb. 3.12 zeigt die geschätzte Wahrscheinlichkeit der drei Subfilter. Dass die Wahrscheinlichkeit von Subfilter 02/03 sich 1 nähert, deutet auf einen DVL/einen IMU-Fehler hin. Es ist zu beobachten, dass alle simulierten DVL-Fehler korrekt erkannt werden. Bei einem Rampenfehler steigt die Wahrscheinlichkeit von Subfilter 02 erst, wenn der Fehler groß genug ist. Der Grund dafür, dass Subfilter 02 mit einer relativ hohen Wahrscheinlichkeit startet, liegt darin, dass im Echtzeittest die Filterbank schon vor längerer Zeit initialisiert wurde und die USV seitdem stillsteht. Dadurch unterstützen die DVL-Messungen die Zustandsabschätzung nicht, was zu einer hohen Wahrscheinlichkeit für Subfilter 02 führt.

Ergebnisse bei IMU-Messfehlern

In diesem Abschnitt wird der entworfene Filterbank-Ansatz in der Post-Processing-Umgebung mit den aufgezeichneten Daten der gleichen Testkampagne wie in Abschnitt 3.3.2 bewertet. In diesem Test werden drei Einstellungen evaluiert:

- EFK: wie Subfilter 01. Es arbeitet eigenständig;
- Filterbank: alle drei Subfilter wie in Abb. 3.4 dargestellt. Subfilter 03 enthält nicht die IMU-Bias-Schätzung;
- Filterbank verbessert: alle drei Subfilter wie in Abb. 3.4 gezeigt. Subfilter 03 enthält die IMU-Bias-Schätzung.

Mit diesen Einstellungen können die Fähigkeit zur Fehlererkennung und die Verbesserung der 2D-Genauigkeit bei IMU-Fehler sowie die Notwendigkeit der Einführung einer IMU-Bias-Schätzung validiert werden. Die Bewertung bei IMU-Fehler wird in der Post-Processing-Umgebung durchgeführt, da in diesem Test zwei Filterbänke (insgesamt 6 Subfilter) verglichen werden sollen, was die Berechnungskapazität der Zielhardware übersteigt.

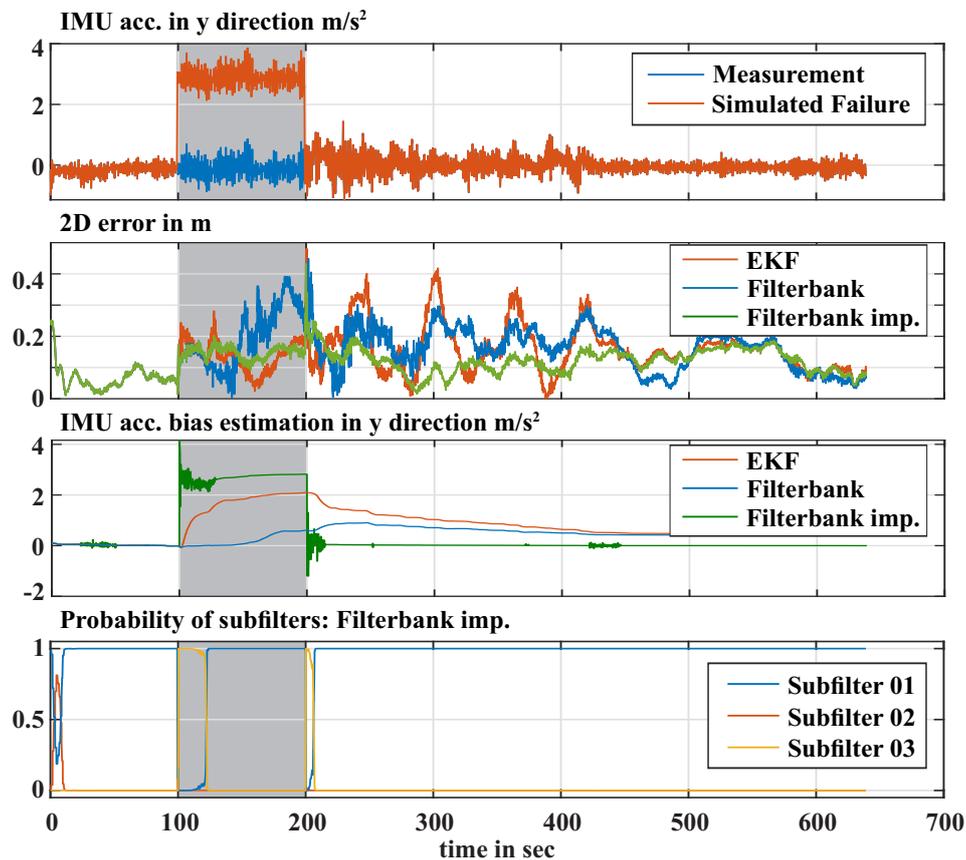


Abbildung 3.13: Auswertung im Post-Processing: IMU-Messung und simulierter Schrittfehler, 2D-Fehler, Wahrscheinlichkeit von Subfiltern und IMU-Bias-Schätzung, aus [36]

Abb. 3.13 zeigt die experimentellen Ergebnisse. Die erste Unterabbildung zeigt, dass von 100 bis 200 Sekunden eine IMU-Fehler mit einer Amplitude von 3 m/s^2 zur Beschleunigungsmessung in y-Richtung hinzugefügt wird. In der zweiten Unterabbildung ist zu

3 Integres Navigationsfilter

erkennen, dass die IMU-Fehler einen geringeren Einfluss auf die 2D-Genauigkeit haben als die DVL-Fehler in der zweiten Unterabbildung in Abb. 3.12. Dennoch zeigt die „Filterbank improved“ die beste Genauigkeit unter allen drei Einstellungen.

Die dritte Unterabbildung zeigt die Schätzung der Beschleunigungsmesserverzerrung in y-Richtung für alle drei Einstellungen. Die „Filterbank improved“ ist in der Lage, die Beschleunigungsmessfehler in y-Richtung ca. 25 Sekunden nach dem Auftreten des IMU-Fehlers zu schätzen. Außerdem konvergiert die Bias-Schätzung des EKF langsam gegen die simulierte IMU-Bias. Und die Filterbank ohne IMU-Bias-Schätzung konvergiert sogar noch langsamer als EKF, da der Subfilter 03 ohne Bias-Schätzung die Filterbank bei IMU-Fehler dominiert.

Die vierte Unterabbildung zeigt die Wahrscheinlichkeitsabschätzung von „Filterbank improved“. Zu Beginn des IMU-Fehlers wird die Wahrscheinlichkeit von Subfilter 03 zu 1, was auf eine erfolgreiche Identifizierung des IMU-Fehlers hinweist. Die Wahrscheinlichkeit von Subfilter 03 sinkt jedoch nach 25 Sekunden auf 0, während der IMU-Fehler weiter anhält. Der Grund dafür ist, dass nach 25 Sekunden die IMU-Fehler durch die „verbesserte Filterbank“ korrekt geschätzt wird. Da die geschätzte IMU-Fehler modelliert und vom Strapdown-Algorithmus in den Subfiltern 01 und 02 der „Filterbank verbessert“ verwendet wird, wird die IMU-Messung zusammen mit der korrekt geschätzten IMU-Fehler als „fehlerfrei“ betrachtet. Ähnlich verhält es sich, wenn am Ende des simulierten IMU-Fehlers die Wahrscheinlichkeit von Subfilter 03 wieder auf 1 ansteigt, bis „Filterbank improved“ die IMU-Fehler korrekt schätzt. Dies führt zu einem Fehlalarm bei der Erkennung von IMU-Fehlern und kann dadurch gelöst werden, dass die IMU-Messung für einen langen Zeitraum ausgeschlossen wird, sobald ein Fehler erkannt wird. Allerdings kann die Langzeit-Lageschätzung nur mit GNSS und DVL nicht beibehalten werden. Nach den Erkenntnissen der Autoren könnte die beste Lösung darin bestehen, zusätzliche Orientierungsmessungen einzuführen, z. B. einen Kompass, zu bewältigen.

Ergebnisse bei Kompassfehlern

Abb. 3.14 zeigt die experimentellen Ergebnisse. Die erste Unterabbildung zeigt, dass von 100 bis 300 Sekunden eines Kompass-Bias mit einer Amplitude von 15 Grad zur Yaw-Winkel wird. In der zweiten Unterabbildung ist zu erkennen, dass die Filterbank-Methode eine bessere Genauigkeit der Orientierungsschätzung aufweist als der EKF unter Kompass Fehlerfälle. In der dritten Unterabbildung wird gezeigt, dass die Positioniergenauigkeit sowohl von EKF als auch von Filterbank ohne signifikanten Unterschied ist. Aber sie sind beide dezimetergenau. Die vierte Unterabbildung zeigt die Wahrscheinlichkeit jedes Subfilters während des Tests. Dass der Subfilter 02 (ohne Kompassmessung) von 100 bis 300

Sekunden eine Wahrscheinlichkeit von etwa 1 hat, zeigt eine erfolgreiche Fehlererkennung an.

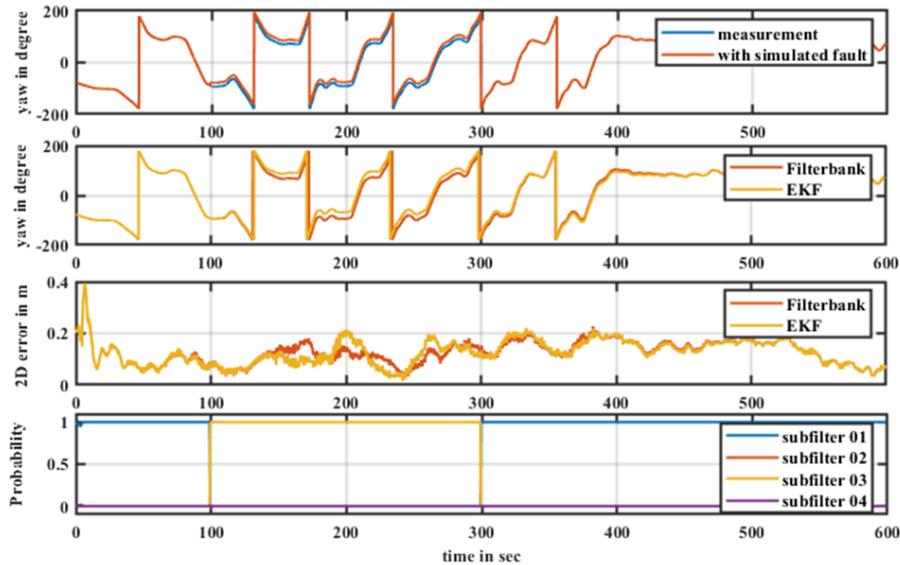


Abbildung 3.14: Auswertung im Post-Processing: IMU Messung und simulierter Schrittfehler, geschätzter Gierwinkel, 2D-Fehler und Wahrscheinlichkeit von Subfiltern

3.3.3 Protection Level Navigationslösung

Abb. 3.15 zeigt den Fehler der geschätzten Position in Nord-, Ost- und Abwärtsrichtung in Metern. Die Bestimmung des PL ist validiert, wenn der Positionierfehler zwischen der unteren und oberen Grenze liegt. Außerdem ist das PL in Abwärtsrichtung deutlich höher als in anderen Richtungen, was auf den bekannten Nachteil der Satellitennavigation zurückzuführen ist: Der vertikale Fehler ist normalerweise größer als der horizontale, da die vertikale Satellitenverteilung schlecht ist. Da das vertikale PL für maritime Anwendungen nicht kritisch ist und von der International Maritime Organization (IMO) nicht gefordert wird, ist dies akzeptabel. Die geschätzte HPL, die nach der Initialisierungsphase unter 2,5 Meter liegt und damit die IMO-Anforderung erfüllt.

3 Integres Navigationsfilter

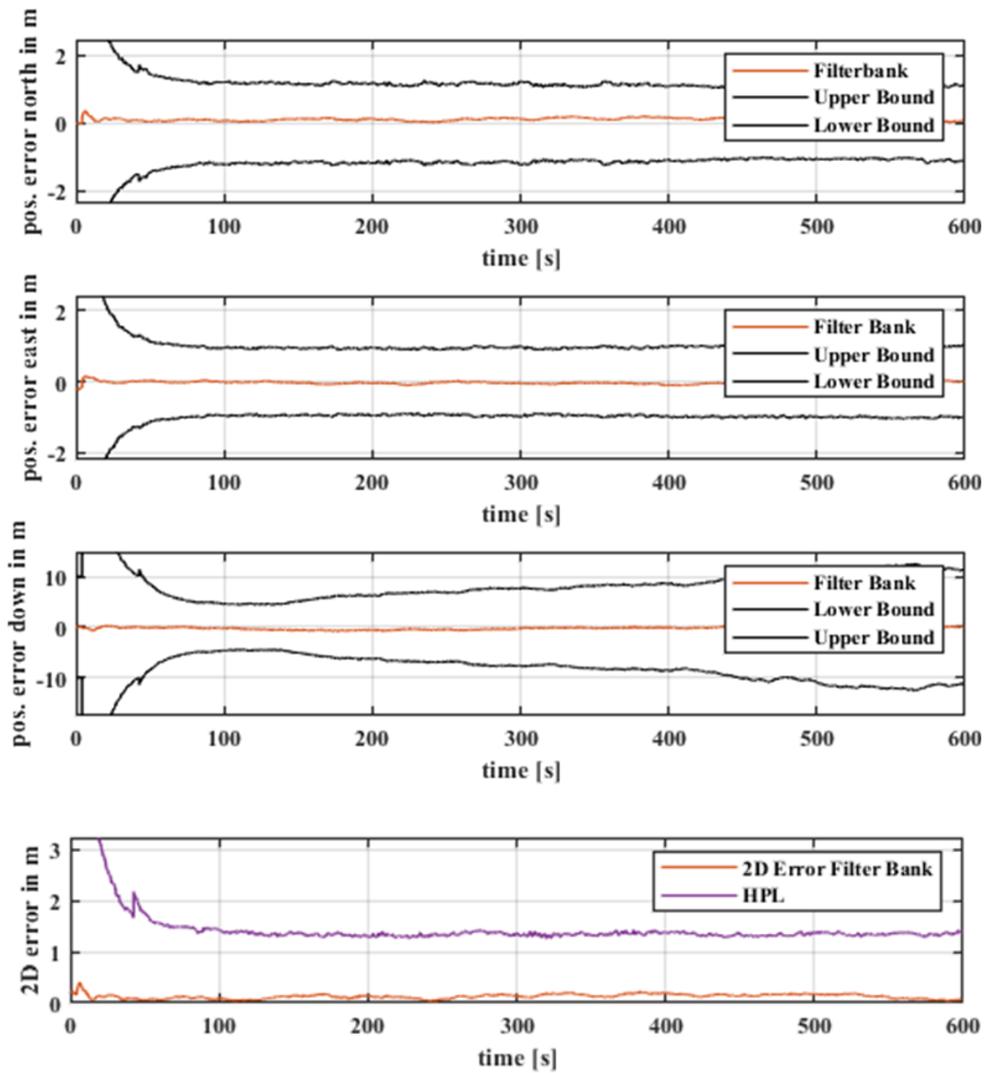


Abbildung 3.15: Auswertungsergebnisse in Post-Processing: Schätzfehler der Positionierung und deren geschätzte PL

4 Navigation im Netzwerk und Umgebungswahrnehmung

Autoren: Jiaying Lin, Jan-Jöran Gehrt

4.1 Einleitung und Stand der Technik

4.1.1 Einleitung und Motivation

Bei der Entwicklung maritimer Autonomie ist die Wahrnehmung der Umgebung eine grundlegende Aufgabe, um eine vollautomatische Fahrzeugführung und -steuerung zu ermöglichen. Beim autonomen Anlegen muss das Schiff in der Lage sein, die dynamischen Hindernisse in der Umgebung zu erkennen, verfolgen und den belegten Raum zu bestimmen, um fahrbare Trajektorien zu planen. Im Manövriererszenario muss das Schiff in der Lage sein, die kinematischen Bewegungszustände der Umgebung zu schätzen und vorherzusagen, um Kollisionen zu vermeiden. Zusammenfassend lässt sich sagen, dass intelligente Schiffe im Hinblick auf die Umweltwahrnehmung kritische Objekte in der Umgebung erkennen und verfolgen und die statische Umgebung in Echtzeit abbilden müssen, um eine sichere Trajektorienplanung und -regelung zu ermöglichen.

Intelligente Schiffe können in ein vernetztes System integriert werden, um den Austausch und die Erweiterung von Wahrnehmungsinformationen zu verbessern. Abbildung 4.1 illustriert das Konzept des vernetzten autonomen Systems, welches im Projekt GALILEOnautic 1 vorgeschlagen wurde, [52]. Mehrere intelligente Schiffe, ausgestattet mit Sensoren für Navigation und Wahrnehmung kommunizieren über ein drahtloses Netzwerk miteinander. Auf diese Weise können sie ihre aktuelle Position und Informationen ihrer Umgebungswahrnehmung austauschen. Durch die Kommunikation untereinander haben die vernetzten Schiffe mehrere Vorteile: Das Erfassungsfeld jedes vernetzten Schiffes kann integriert werden, so dass mehr Bereiche der Umgebung erkundet werden. Dieser Informationsaustausch kann den begrenzten Erfassungsbereich der exterozeptiven Sensoren ausgleichen

4 Navigation im Netzwerk und Umgebungswahrnehmung

und eine bessere Manöverplanung und Kollisionsvermeidung ermöglichen. Außerdem können vernetzte Schiffe mit einer genauen geodätischen Position die Positionierung anderer vernetzter Schiffe durch Relativpositionierung verbessern.

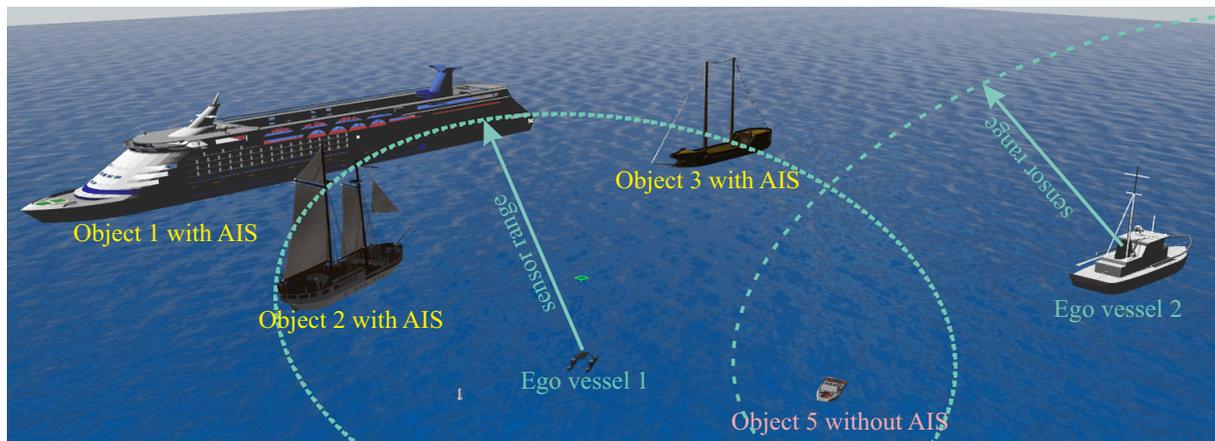


Abbildung 4.1: Konzept des Systems der vernetzten Schiffe. Zwei vernetzte Schiffe sind mit Wahrnehmungssensoren ausgestattet und können die Wahrnehmungsaufgaben erfüllen. Indem sie miteinander kommunizieren, können sie den gesamten Wahrnehmungsbereich erweitern und die Erkennungen in der Umgebung vervollständigen, adaptiert von [53]

Stand der Technik

Im maritimen Bereich werden automatische Identifikationssysteme (AIS) häufig zur See-raumüberwachung eingesetzt. AIS ist ein internationales automatisches Ortungssystem für die Kommunikation von Schiff zu Schiff, von Schiff zu Land und von Land zu Schiff. Mit AIS ausgestattete Schiffe können ihre Maritime Mobile Service Identity (MMSI), ihre Position, Größe, Geschwindigkeit und ihren Kurs übermitteln. Die Aktualisierungsrate beträgt 2 bis 30 Sekunden (alle 3 Minuten, wenn das Schiff vor Anker liegt), je nach Art des AIS-Geräts, der Geschwindigkeit und der Kursänderungsrate des Schiffs, [54], [55]. Heutzutage sind Schiffe mit einer Bruttoreaumzahl von 300 und mehr verpflichtet, AIS-Geräte zu verwenden. Diese AIS Nachrichten können als zusätzliche Quelle genutzt werden, um den Automatisierungsgrad zu verbessern, insbesondere bei der Erkennung von Anomalien, der Routenabschätzung, der Kollisionsvorhersage und der Pfadplanung, [56]. Dennoch gibt es immer noch kritische Probleme mit den reinen AIS-Lösungen: die Aktualisierungsrate des AIS ist unzureichend für hochdynamische Objekte im Nahbereich; außerdem sind einige AIS Nachrichten aufgrund menschlicher Fehlbedienung falsch oder leer. Laut [57] zeigten 30% der Schiffe falsche Statusinformationen an, und es gab wahrscheinlich noch mehr Beispiele, die von der Forschung unentdeckt blieben. Darüber hinaus

hängt die Qualität der Positionsbestimmung des AIS-Senders von den GNSS-Empfängern ab, die im Standalone-Modus eine Genauigkeit von 5 bis 10 m haben. Dies ist auch für das autonome Fahren von kleinen dynamischen Schiffen entscheidend. Schließlich sind nicht alle Schiffe und Hindernisse wie Bojen und kleine Boote mit AIS ausgestattet.

Exterozeptive Sensoren wie radar- oder laserbasierte Systeme werden häufig in maritime Überwachungssysteme integriert, um die Einschränkungen der reinen AIS-Lösung zu kompensieren. [58] hat ein künstliches neuronales Netz für die Erkennung und Verfolgung mehrerer Schiffe vorgeschlagen. Die Zustände jedes einzelnen Schiffes werden dabei mithilfe des erweiterten Kalman-Filters (EKF) geschätzt. [59] hat einen Bayes'schen Ansatz vorgeschlagen, der sogenannte Belief-Propagation verwendet, um die AIS- und Radarnachrichten zu verschmelzen. Diese Ansätze sind nur in Simulationen unter der Annahme korrekter und unverfälschter Messungen erprobt. Ihnen fehlt die experimentelle Validierung in der realen Welt, wo Echtzeitfähigkeit, unzureichende Informationen und Störungen berücksichtigt werden müssen. [60] hat ein probabilistisches Datenassoziationsfilter für die Assoziation von AIS und Radarerfassung auf einem USV implementiert, wobei die Robustheit der Assoziation und die Genauigkeit der Schätzung noch unzureichend sind.

Neben dem Radar hat die Verwendung von LiDAR für die Meeresüberwachung in den letzten Jahren dank seiner hohen Genauigkeit, hohen Datendichte und Lichtunabhängigkeit zugenommen. [61] und [62] haben LiDAR in das Schiffsüberwachungssystem integriert, um eine zuverlässigere Objekterkennung im Nahfeld zu erreichen.

4.2 Wissenschaftlich-technische Arbeiten

4.2.1 Überblick

Wie in Abbildung 4.2 dargestellt, wird für jedes Ego-Schiff LiDAR als primärer Sensor verwendet. Dazu werden folgenden Sensoren oder Geräten als Systemeingänge verwendet: AIS, Radar, GNSS, und INS. Als Ausgabe wird eine Belegungskarte erstellt, die die Pose und Größe von 3D Bounding Boxes mit Bewegungsabschätzung enthält. Die Pipeline besteht aus drei Hauptteilen: Objekterkennung, Multiple Objekt Tracking (MOT) und statische Umgebungsdarstellung.

In Bezug auf die Objekterkennung wird in Abschnitt 4.2.2 die Verarbeitung von Punktwolken erörtert, so dass die potenziellen Objekte aus den Punktwolken extrahiert und als sogenannte Bounding Boxes dargestellt werden können, in die die Punktwolken eingepasst werden. In GALILEOnautic 2 wurden dazu sowohl eine konventionelle Methode als auch die Methode des Deep-Learning untersucht.

4 Navigation im Netzwerk und Umgebungswahrnehmung

Wie beim MOT, das in Abschnitt 4.2.3 besprochen wird, werden die erkannten Bounding Boxes mit den vorhandenen Objektschätzungen verknüpft. Für die Koordinatentransformation wird die 6D-Pose des Ego-Schiffs mit Hilfe des in Kap. 2 und Kap. 3 erläuterten Navigationsfilters genutzt. Messungen von anderen Wahrnehmungssensoren, wie z. B. AIS und Radar, können dann in die Bounding-Box-Listen integriert werden, die mithilfe einer Reihe von Filterbänken verfolgt werden, um die Pose und Bewegungen jedes Objekts zu schätzen.

Im Anschluss an das MOT erörtert Abschnitt 4.2.4 die statische Umgebungsdarstellung. Von den ursprünglichen LiDAR-Punktwolken werden die Punkte innerhalb der Bounding-Boxes abgezogen. Die verbleibenden Punktwolken werden dann als statische Hindernisse behandelt und als Eingabe für eine globale Occupancy Grid Map verwendet. Der belegte Raum wird mit einer Reihe von Polygonen zusammengefasst, die die Umrisse der statischen Infrastrukturen beschreiben. Die generierten Polygone und die geschätzten Bounding-Boxes aus dem MOT bilden die gesamte Ausgabe der Pipeline.

Um die Wahrnehmung von einem einzelnen Ego-Schiff aus zu erweitern, wurde in GALILEOnautic 2 die Möglichkeit eines vernetzten Multi-Agenten-Systems untersucht, bei dem mehr als ein Schiff, das mit der gleichen Art von Wahrnehmungssensoren ausgestattet ist, seine Schätzungen über ein drahtloses Netzwerk austauschen kann. Die Informationen der lokalen Umgebungswahrnehmung verschiedener Schiffe werden in die zentrale Datenbank integriert, um die endgültige kooperative Umgebungswahrnehmungsschätzung zu erzeugen.

Die gesamte Wahrnehmungspipeline wird in Abschnitt 4.3 ausgewertet. Verschiedene Testfahrten wurden mit dem unbemannten Wasserfahrzeug MESSIN und dem Forschungsschiff DENEK im Rostocker Hafen, Deutschland, durchgeführt. In diesem Sachbericht wird die vorgeschlagene Pipeline in verschiedenen Szenarien, z.B. beim automatischen Anlegen und Manövrieren evaluiert.

4.2.2 Objekterkennung

Zuerst werden die Punktwolken mit Hilfe der modifizierten Euklidischen Clustering Methode zu Clustern gruppiert. Anschließend werden diese Cluster durch Bounding-Boxen dargestellt und in eine Objektliste umgewandelt, welche die Position, Ausrichtung und Objektgröße enthält. Jede Bounding-Box wird aus der Form extrahiert, die am besten zur Punktwolke des Clusters passt, um zuverlässige Werte für die geometrischen Eigenschaften zu erzeugen.

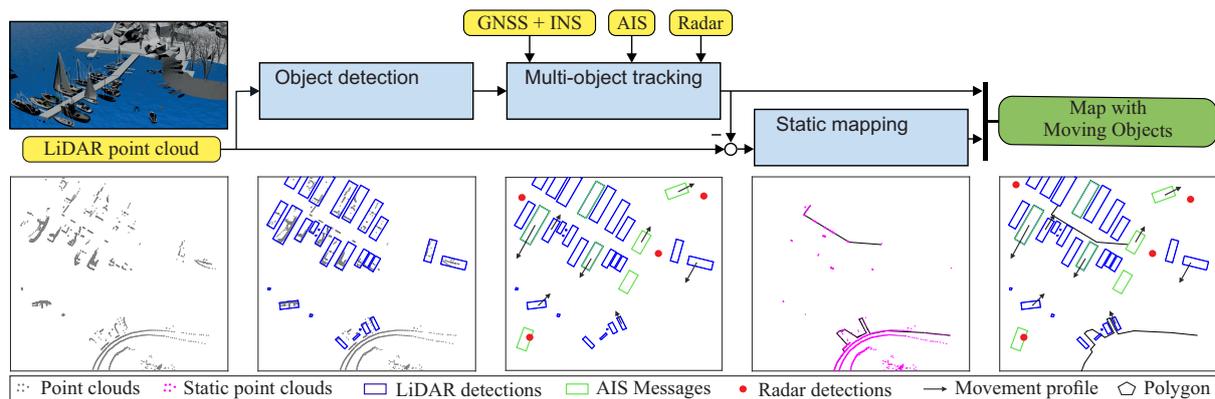


Abbildung 4.2: Vorgeschlagene Pipeline für die Umgebungswahrnehmung auf einem einzelnen Ego-Schiff. In der ersten Zeile werden die Eingaben in gelben Textfeldern, die Ausgaben in grünen Textfeldern und die Module in blauen Textfeldern dargestellt. Die zweite Zeile zeigt das entsprechende erwartete Ergebnis jedes Moduls, aus [63]

Punktwolke Vorverarbeitung

Bei der Vorverarbeitung werden irrelevante oder verrauschte Punkte in der Punktwolke herausgefiltert. Da die Wasseroberfläche im LiDAR-Punktwolken-Scan keine Reflexion erzeugt, ist eine Bodenentfernung für einen maritimen Anwendungsfall nicht erforderlich. Daher werden die Punktwolken lediglich mit einem Pass-Through-Filter, einem VoxelGrid-Filter und einem Ausreißer-Entfernungsfiler gefiltert, um die redundanten, überlappenden Punkte zu entfernen und das weitere Verfahren zu beschleunigen. Alle diese Filter werden in der Punktwolkenbibliothek [64] berechnet. Der Pass-Through-Filter schneidet Punkte ab, die außerhalb eines bestimmten Bereichs liegen. In unserem Fall wird der Pass-Through-Filter verwendet, um die zum Ego-Vessel gehörenden Punktwolken und die Punktwolken außerhalb eines vertikalen Bereichs in der z -Achse abzuschneiden. Mittels eines VoxelGrid-Filters werden die Punktwolken entsprechend der definierten Gittergröße heruntergetastet und gefiltert. Der Filter zur Entfernung von Ausreißern entfernt verrauschte Punkte auf der Grundlage einer statistischen Analyse der Nachbarschaft jedes Punktes. Die Anwendung dieser Filter kann den Berechnungsaufwand enorm reduzieren, während die Wahrnehmung weiterhin in hoher Auflösung erfolgen kann.

Clustering

Die Clustering-Methoden gruppieren die Punktwolken anhand ihres euklidischen Abstands. In dieser Arbeit wurde das modifizierte euklidische Clustering Methode verwendet, [65]. Beim modifizierten euklidischen Clustering werden alle Punkte zunächst mithilfe ei-

4 Navigation im Netzwerk und Umgebungswahrnehmung

nes k-d-Tree [66] sortiert. Danach werden die Punkte in einer Nachbarschaft innerhalb eines Schwellenradius R_{MC} als ein Cluster gruppiert. Da die laterale Auflösung des LiDAR von der Entfernung des Punktes abhängt, können die entfernten Punkte zum selben Objekt gehören, auch wenn sie einen größeren Abstand zueinander haben. Daher ist der Schwellenwert R_{MC} variabel und hängt von der Entfernung des Clusters ab.

Shape Fitting

Das Modul Shape Fitting wandelt die Cluster in eine Objektliste um, die Position, Ausrichtung und Objektgröße enthält. In der maritimen Anwendung werden hauptsächlich zwei Grundformen erkannt: elliptische Zylinder (z. B. Boote und Bojen) und rechteckige Würfel (z. B. Ladungen und Container). Und bei ein und demselben Schiff können während der Fahrt verschiedene Teile, z. B. der elliptische Bug und das rechteckige Heck, gescannt werden. Aufgrund der Einschränkungen von LiDAR kann nur ein Teil der Objekte gescannt werden, so dass der sichtbare Teil eine L-Form oder eine elliptische Kurve ist. Je nach Geometrie wird jedes LiDAR-Punktcluster entweder mit einer L-Form oder einer Ellipse angepasst.

Abbildung 4.3 verallgemeinert die Problemstellung sowohl der L-Form als auch der Ellipsenfitting: jeder dieser Algorithmen nimmt als Eingabe eine Liste von Punkten in der x - und y -Achse. Hier vernachlässigen wir die Nick- und Rollwinkel, da sie für die Flugbahnplanung und Kollisionsvermeidung nur von geringer Bedeutung sind. Daher werden die 3D-Punkte in jedem Cluster für die Formanpassung in 2D projiziert. Das Ergebnis dieser Algorithmen ist der zentrale Punkt der Form $P_0(x_0, y_0)$, der Kurs ϕ (entlang der längsten Seite der angepassten Form) sowie die Abmessungen w und l . Da es sich bei den Gefäßformen in der Regel um unvollständige Ellipsen handelt, werden die Länge der Form und der zentrale Punkt P_0 durch die Bestimmung des minimalen und maximalen Punktes entlang der Hauptachse angepasst, so dass die unvollständigen elliptischen Grenzen bestimmt werden können. Die Höhe der Box oder des elliptischen Zylinders h ist definiert als der vertikale Abstand zwischen dem höchsten und dem niedrigsten Punkt.

4.2.3 Multiple Objekt Tracking

Die Detektionen aus verschiedenen Sensoren werden in dem weiteren Multiple Objekt Tracking (MOT) Module bearbeitet für die Objektbeobachtung. Wie in Abbildung 4.4 dargestellt, besteht das Modul MOT aus zwei Hauptteilen: Datenmanagement und Objektverfolgung. Das Datenmanagementmodul erhält die Ergebnisse der LiDAR Objekterkennung in Abschnitt 4.2.2 zusammen mit den Messungen von AIS und Radar als

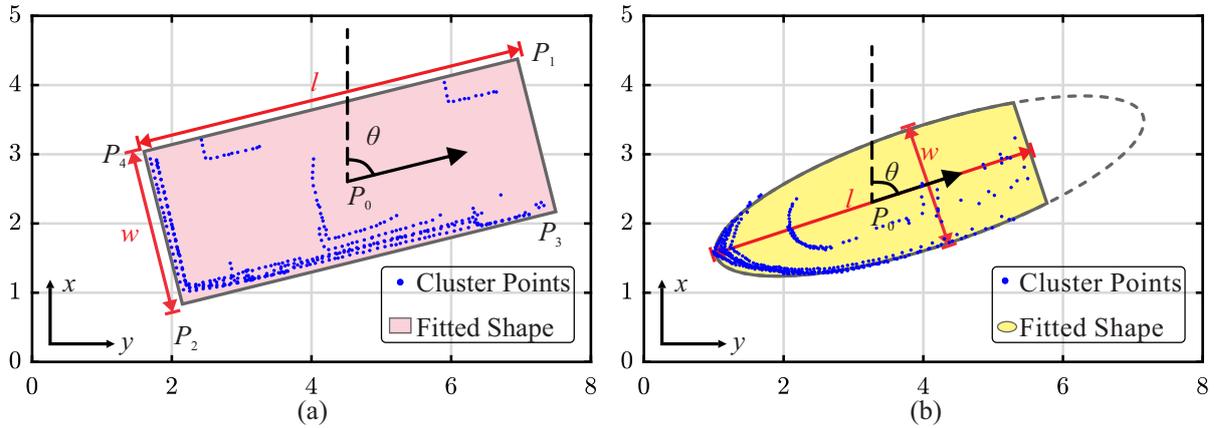


Abbildung 4.3: Allgemeine Beschreibung der LiDAR Objekterkennung mittels Shape Fitting Methode, aus [67]

Eingaben. In *Koordinatentransformation* werden die erkannten Objekte von den lokalen Schiffskörperkoordinaten in geodätische Koordinaten transformiert. Hier wird die 6D-Pose des Ego-Fahrzeugs mithilfe des Navigationsfilters geschätzt. In der *Objekt Assoziation* werden diese Objekte mit der im Datenpuffer verwalteten vorhandenen Objektschätzung verknüpft. Jedes assoziierte Objekt wird dann an die entsprechende *Filterbank* zur separaten Verfolgung weitergeleitet. Da eine Filterbank für ein Objekt zuständig ist, lässt sich dieses System leicht auf so viele Objekte wie möglich erweitern. Hier verwenden wir das Konzept der Filterbanken, um mit fehlerhaften oder fehlenden Messungen fertig zu werden, insbesondere mit den AIS Nachrichten mit vielen möglichen Fehlern in jedem Feld. Jede Filterbank besteht aus mehreren adaptiven Erweiterte Kalman Filter (EKF) als Subfiltern, die proportional zu den aktuellen Messqualitäten integriert werden.

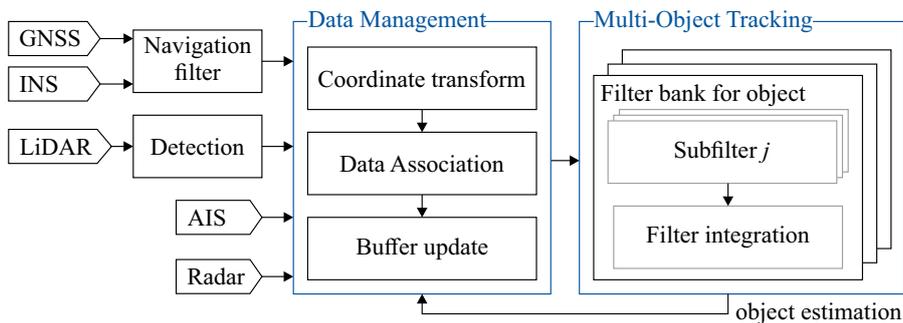


Abbildung 4.4: Aufbau der MOT und Gliederung dieses Kapitels, adaptiert von [53]

Datenmanagement

Sobald neue LiDAR- oder AIS-Messungen verfügbar sind, wird der Datenspeicher aktualisiert. Für das Objekt Assoziieren wird in unserer frühen Arbeit eine auf der Evidenz-

4 Navigation im Netzwerk und Umgebungswahrnehmung

theorie basierende Methode verwendet [68]. Im Vergleich zu traditionellen Bayes'schen Netzen erlaubt der Evidenztheorie unvollständige Informationen in der Evidenz, was unseren Bedingungen von unvollständigen oder falschen Messungen entspricht. Die neuen Messungen, entweder von LiDAR oder AIS, werden als Set $\mathcal{A} = \{a_1, a_2, \dots, a_m\}$ definiert, und das vorhandene Objekt im Speicher wird als Set $\mathcal{B} = \{b_1, b_2, \dots, b_n\}$ definiert. Ziel ist es, die Assoziation zwischen den Set \mathcal{A} und \mathcal{B} anhand verschiedener Evidenzen zu finden. Der Algorithmus besteht aus drei Schritten:

Schritt 1 Ähnlichkeit bestimmen: Die Ähnlichkeit von jedem a_i in \mathcal{A} ($i = 1, 2, \dots, m$) und b_j in \mathcal{B} ($j = 1, 2, \dots, n$) soll verglichen werden. Da die Evidenztheorie unvollständige Informationen zulässt, können drei Hypothesen für die Ähnlichkeit definiert werden: s_{ij} für ähnlich, n_{ij} für Nicht-ähnlich und u_{ij} für unbekannte Informationen, und $s_{ij} + n_{ij} + u_{ij} = 1$. Für den Vergleich können drei Arten von Informationen verwendet werden: Position (bezeichnet als $s_{ij,p}$), Größe $s_{ij,s}$ und Richtung $s_{ij,h}$. Die Ähnlichkeit jedes Informationstyps wird zunächst mit Hilfe der Mahalanobis-Distanz d_{ij} bestimmt und dann in Ähnlichkeit umgewandelt:

$$s_{ij} = -e^{\lambda d_{ij}}, \quad n_{ij} = 1 - s_{ij}, \quad u_{ij} = 0, \quad (4.1)$$

wobei λ ein vordefinierter Koeffizient zur Steuerung der Konvergenzgeschwindigkeit ist. Auf diese Weise können drei Ähnlichkeiten $s_{ij,p}$, $s_{ij,s}$ und $s_{ij,h}$ in einem Bereich von $[0, 1]$ bestimmt werden. Für den Fall, dass Informationen fehlen, z.B. wenn die Kursinformationen in einer AIS-Nachricht fehlen, wird die Ähnlichkeit wie folgt festgelegt:

$$s_{ij} = 0, \quad n_{ij} = 0, \quad u_{ij} = 1. \quad (4.2)$$

Schritt 2 Evidenzkombination: Wir wenden die DS-Kombinationsregel an, um die Ähnlichkeit verschiedener Beweise zu kombinieren. Die Kombination von zwei Evidenzen wird wie folgt durchgeführt, wobei die Kombination von Positions- und Größenevidenz als Beispiel dient: Der Konfliktindikator K wird bestimmt, indem die Summe der widersprüchlichen Beweise berücksichtigt wird

$$K = s_{ij,p} \cdot n_{ij,s} + n_{ij,p} \cdot s_{ij,s} \quad (4.3)$$

Unter Verwendung dieses Konfliktindikators kann die kombinierte Glaubensfunktion dann normalisiert werden:

$$\begin{aligned} s_{ij,comb} &= \frac{1}{1 - K} \cdot (s_{ij,p} \cdot s_{ij,s} + s_{ij,p} \cdot u_{ij,s} + u_{ij,p} \cdot s_{ij,s}) \\ n_{ij,comb} &= \frac{1}{1 - K} \cdot (n_{ij,p} \cdot n_{ij,s} + n_{ij,p} \cdot u_{ij,s} + u_{ij,p} \cdot n_{ij,s}) \\ u_{ij,comb} &= \frac{1}{1 - K} \cdot (u_{ij,p} \cdot u_{ij,s}) \end{aligned} \quad (4.4)$$

Die kombinierte Funktion $s_{ij,kamm}$, $n_{ij,kamm}$ und $u_{ij,kamm}$ kann als Evidenz behandelt und für die weitere Evidenzkombination verwendet werden.

Schritt 3 Entscheidungsfindung: nach der Bestimmung von s_{ij} , $i = 1, 2, \dots, m$, $j = 1, 2, \dots, n$, wird das $s_{ij,max}$ mit der maximalen Ähnlichkeit als übereinstimmendes Paar betrachtet, wenn $s_{ij,max} > s_{Grenzwert}$, wobei $s_{Grenzwert}$ ein vordefinierter Grenzwert ist. Andernfalls werden die Objekte als neue Objekte markiert und in den Objektpuffer eingetragen.

Objektverfolgen mittels Filterbank Technik

Nach dem Datenassoziiieren werden alle Objekte im Datenspeicher separat weiterverarbeitet. Abbildung 4.5 zeigt die detaillierte Struktur jeder Filterbank: ein Satz adaptiver erweiterter Kalman-Filter (AEKF) wird unter Verwendung einzelner Sensormessungen und auch einer Kombination davon ausgeführt. Alle AEKF haben dasselbe Systemmodell, verfügen aber über unterschiedliche Korrekturmechanismen, die auf verschiedenen Messungen basieren, so dass jeder Filter in der Bank eine Zustandsschätzung erzeugt. Anschließend wird unter Verwendung der Innovationsinformationen jedes AEKF eine Wahrscheinlichkeit für die erzeugten Zustandsschätzungen berechnet. Schließlich wird jede Filterzustandsschätzung entsprechend der Wahrscheinlichkeit gewichtet und als gewichtete Summe gemischt, um die endgültige Schätzung zu erhalten. Auf diese Weise können falsche Sensormessungen herausgefiltert werden. Der Ausgang der Filterintegration wird für weitere Filterschätzungen und als Validierung für die Datenassoziiierung verwendet.

Wie in Abbildung 4.5 gezeigt, besteht der Zustandsvektor \mathbf{x} in jedem Subfilter aus 8 Elementen:

$$\mathbf{x} = [p_x \quad p_y \quad v_x \quad v_y \quad \phi \quad \dot{\phi} \quad l \quad w]^T, \quad (4.5)$$

wobei p_x und p_y die 2D Mittelposition des Objekts bezeichnen; v_x und v_y beschreiben die Geschwindigkeit; ϕ und $\dot{\phi}$ beschreiben den Kurs und die Drehrate; und l , w bezeichnen die Länge und Breite der Objekte.

Das in dieser Arbeit verwendete Modell ist ein allgemeines nichtlineares Starrkörpermodell:

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{G}_k \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k) \quad (4.6)$$

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k) \quad (4.7)$$

wobei k der Zeitindex ist, \mathbf{x}_k den in (4.5) definierten Zustandsvektor bezeichnet. \mathbf{F}_k und \mathbf{G}_k sind die Jacobean-Matrizen der nichtlinearen Funktionen für Zustandsübergänge bzw.

4 Navigation im Netzwerk und Umgebungswahrnehmung

Rauschen. Die Zustände werden mit dem Modell der konstanten Geschwindigkeit vorhergesagt, wobei angenommen wird, dass die Geschwindigkeit innerhalb eines Zeitschritts konstant ist. \mathbf{y}_k ist der Messvektor und \mathbf{h}_k ist die nichtlineare Gleichung von \mathbf{x}_k . \mathbf{w}_k und \mathbf{v}_k sind additives Gaußsches Rauschen, wobei \mathbf{Q}_k und \mathbf{R}_k die Prozess- bzw. Messrauschkovarianzmatrizen bezeichnen.

Die Zustände werden mit Hilfe des einheitlichen Modells vorhergesagt, daher kann \mathbf{F}_k wie folgt bestimmt werden

$$\mathbf{F}_k = \begin{pmatrix} 1 & 0 & \delta t & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & \delta t & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \delta t & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (4.8)$$

wobei δt die Zeitdifferenz zwischen zwei Zeitstempeln ist. Tabelle 4.1 zeigt die verfügbare Messinformation aus LiDAR und AIS Sensoren. In den folgenden Unterabschnitten werden die Korrekturteile der einzelnen Subfilter und ihre Integration im Einzelnen erläutert.

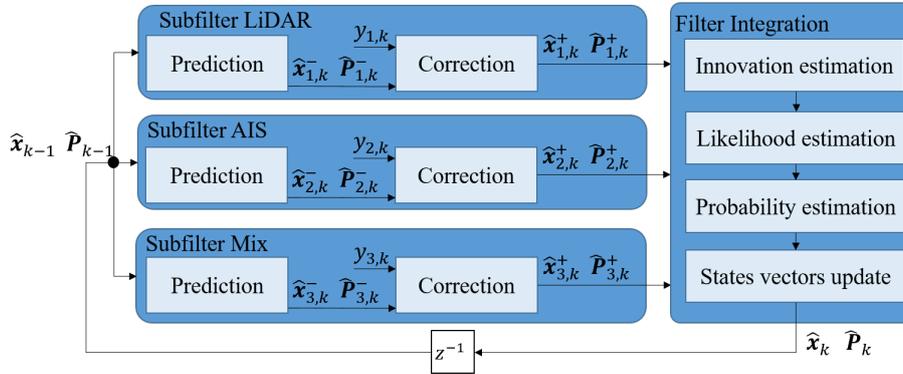


Abbildung 4.5: Struktur der Filter Bank, [53]

Subfilter LiDAR Die LiDAR-Messungen $\mathbf{y}_{1,k}$ können wie folgt definiert werden (der Index 1 bezeichnet den ersten Teilfilter)

$$\mathbf{y}_{1,k} = [p_x \quad p_y \quad \phi \quad l \quad w]^T, \quad (4.9)$$

Bei der Verwendung eines LiDAR-Sensors werden aufgrund der diskreten Natur der LiDAR-Messungen immer Teile der erfassten Objekte fehlen. Wie in Abbildung 4.6 dargestellt, hängt die Größe der gemessenen Form von dem am weitesten entfernten Laserpunkt

Tabelle 4.1: Arten der von LiDAR und AIS empfangenen Informationen

Sources	Position	Size	Heading	SOG	COG
LiDAR $\mathbf{y}_{k,1}$	p_x, p_y	w, l, h	ϕ	-	-
AIS $\mathbf{y}_{k,2}$	p_x, p_y	w, l	ϕ	$\sqrt{v_x^2 + v_y^2}$	$\arctan(v_y, v_x)$
	evidence 1	evidence 2	evidence 3		

ab, der von der Oberfläche reflektiert wird. Wenn sich das Objekt von Punkt A nach B bewegt, geht die Breiteninformation verloren, was zu einer fehlerhaften Größemessung und damit zu einer verzerrten Bestimmung des zentralen Punktes führt. Daher sollte die Rauschkovarianzmatrix \mathbf{R}_k auf der Grundlage der Sichtbarkeit des Objekts durch LiDAR angepasst werden. Der prozentuale Anteil der Sichtbarkeit \mathbf{P}_{vis} (5×1 Vektor für 5 Mess-typen in Tabelle 4.1, 0 für nicht sichtbar, 1 für volle Sichtbarkeit) in Breite und Länge hängt von der relativen Lage des Objekts ab:

$$\mathbf{P}_{vis} = \mathbf{P}_{vis}(p_x - p_{x0}, p_y - p_{y0}, \phi - \phi_0), \quad (4.10)$$

wobei p_{x0} , p_{y0} und ϕ_0 den Mittelpunkt der Detektion definieren. Der \mathbf{R}_k ist eine Funktion vom \mathbf{P}_{vis} :

$$\mathbf{R}_{1,k} = \mathbf{R}_{L,k0} + r(1 - \mathbf{P}_{vis})(1 - \mathbf{P}_{vis})^T, \quad (4.11)$$

wobei $\mathbf{R}_{L,k0}$ die Rauschkoeffizientenmatrix bei voller Sichtbarkeit ist, und $r \geq 0$ einen positiven definiten konstanten Verstärker bezeichnet. Da der zusätzliche Term positiv ist, ist $\mathbf{R}_{1,k}$ immer positiv semidefinit, was die Stabilität des Subfilters garantiert.

Subfilter AIS Der Subfilter AIS, bezeichnet als Subfilter 2, hat dasselbe Systemmodell wie (4.6). Nach Tabelle 4.1 lassen sich die AIS-Messungen $\mathbf{y}_{2,k}$ (der tiefgestellte Index A bezeichnet Messungen von AIS) wie folgt zusammenfassen:

$$\mathbf{y}_{2,k} = \begin{bmatrix} p_x & p_y & \sqrt{v_x^2 + v_y^2} & \arctan(v_y, v_x) & \phi & l & w \end{bmatrix}^T, \quad (4.12)$$

Um zeitlich fehlerhafte AIS-Messungen zu vermeiden, wird die Rauschkovarianzmatrix \mathbf{R}_k gemäß der innovationsbasierten adaptiven Schätzmethode, [69], konsequent angepasst. Die Innovationssequenz \mathbf{v}_k ist definiert als:

$$\mathbf{v}_k = \mathbf{y}_{2,k} - \hat{\mathbf{y}}_{2,k} \quad (4.13)$$

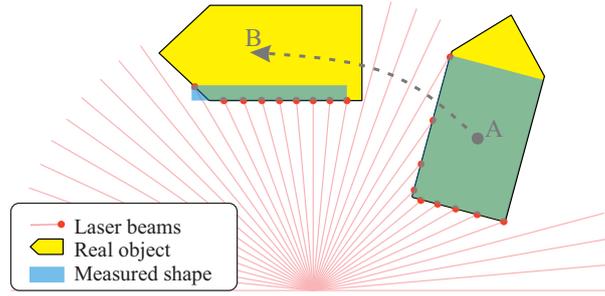


Abbildung 4.6: Problemstellung bei der LiDAR Objekterkennung: Die Sichtbarkeit der Größe hängt von der Position des Objekts zum Sensor ab, adaptiert von [67]

wobei $\mathbf{y}_{2,k}$ der reale Messwert und $\hat{\mathbf{y}}_{2,k}$ der aus (4.12) ermittelte geschätzte Messwert ist. Durch Mittelwertbildung eines gleitenden Fensters über N Innovationssequenzen entwickelt sich die Adaption mit der Zeit weiter:

$$\mathbf{C}_k = \frac{1}{N} \sum_{\kappa=k-N+1}^k \mathbf{v}_\kappa \cdot \mathbf{v}_\kappa^T \quad (4.14)$$

Das $\mathbf{R}_{A,k}$ kann wie folgt angepasst werden:

$$\mathbf{R}_{A,k} = \mathbf{C}_k - \mathbf{H}_k \hat{\mathbf{P}}_k \mathbf{H}_k^T, \quad (4.15)$$

wobei \mathbf{H}_k die Jacobimatrix von \mathbf{h}_k ist und $\hat{\mathbf{P}}_k$ die vorhergesagte Kovarianz der Zustandsmatrix ist. *Subfilter Mix* Der Subfilter Mix berücksichtigt sowohl die AIS- als auch die LiDAR-Messungen in der Korrekturstufe. Wie in Abbildung 4.7 gezeigt, werden die LiDAR-Messungen kontinuierlich mit einer Rate von 0,1 s empfangen, während die AIS-Messungen nur alle 2 bis 30 s verfügbar sind. Daher wenden alle Unterfilter ereignisbasierte Korrekturschritte an, d. h. alle Unterfilter haben eine Grundrate von 0,1 s für die Vorhersage und korrigieren ihre Schätzung nur mit Messungen, sobald eine neue Messung verfügbar ist. Für den Fall der Subfiltermischung ist die Messfunktion $\mathbf{y}_{3,k}$ jeweils gleich (4.12) und (4.9), wenn nur eine Art der Messungen verfügbar ist; sind beide Messungen verfügbar, ist die $\mathbf{y}_{3,k}$

$$\mathbf{y}_{3,k} = \begin{bmatrix} \mathbf{y}_{1,k}^T & \mathbf{y}_{2,k}^T \end{bmatrix}^T, \quad (4.16)$$

$$\mathbf{R}_{3,k} = \begin{bmatrix} \mathbf{R}_{1,k}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_{2,k} \end{bmatrix}$$

Filter Integration Die Filterintegration integriert die Schätzung der drei Subfilter unter Verwendung eines verallgemeinerten Pseudo-Bayes-Algorithmus erster Ordnung, um das

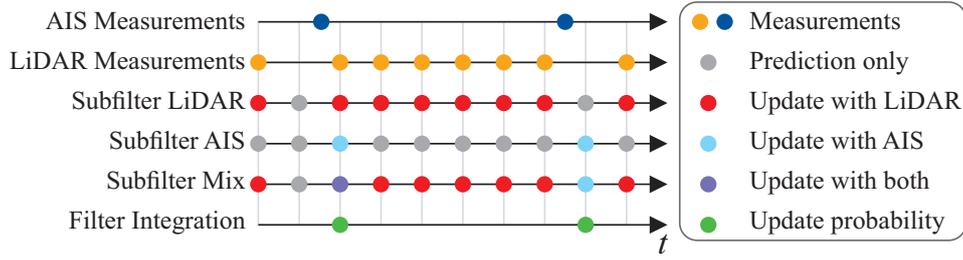


Abbildung 4.7: Multi-Rate-Filterstruktur, [53]

optimale Gewicht für die Schätzkombination zu bestimmen. Zunächst wird die Innovation $\mathbf{r}_{i,k}$ der Kovarianz $\mathbf{S}_{i,k}$ jedes Subfilters i ($i = 1, 2, 3$) wie folgt bestimmt:

$$\begin{aligned}\mathbf{r}_{i,k} &= \mathbf{y}_{i,k} - \hat{\mathbf{y}}_{i,k} \\ \mathbf{S}_{i,k} &= \mathbf{H}_{i,k} \mathbf{P}_{i,k}^- \mathbf{H}_{i,k}^T + \mathbf{R}_{i,k}\end{aligned}\quad (4.17)$$

Anschließend kann die Wahrscheinlichkeit für jeden Subfilter berechnet werden

$$p_{i,k} = \frac{1}{\sqrt{(2\pi)^n |\mathbf{S}_{i,k}|}} e^{-\frac{1}{2} \mathbf{r}_{i,k}^T \mathbf{S}_{i,k}^{-1} \mathbf{r}_{i,k}} \quad (4.18)$$

Mit Hilfe der Wahrscheinlichkeitsrechnung wird die Wahrscheinlichkeit jedes Subfilters bestimmt:

$$\mu_{i,k} = \frac{p_{i,k} \mu_{i,k-1}}{\sum_i p_{i,k} \mu_{i,k-1}}, \quad (4.19)$$

mit deren Hilfe die endgültige Zustandsschätzung von Schritt k aktualisiert werden kann:

$$\mathbf{x}_k = \sum_i \mu_{i,k} \mathbf{x}_{i,k} \quad (4.20)$$

Die endgültige Zustandsschätzung wird dann zurückgeführt und für die Vorhersage des nächsten Zeitstempels verwendet.

4.2.4 Statische Kartierung

Für die Kartierung der statischen Umgebung werden Punktwolken als Eingabe verwendet und ein Polygon geschätzt, das die Umrisse der statischen Hafenstruktural darstellt. Wie in Abbildung 4.2 gezeigt, werden die Punktwolken im Anschluss an das *MOT* anhand der Punkte in den Begrenzungsrahmen subtrahiert. Die verbleibenden Punktwolken zeigen die statische Umgebung an, z. B. Pier, Strand oder andere Hafonstruktionen. Diese

4 Navigation im Netzwerk und Umgebungswahrnehmung

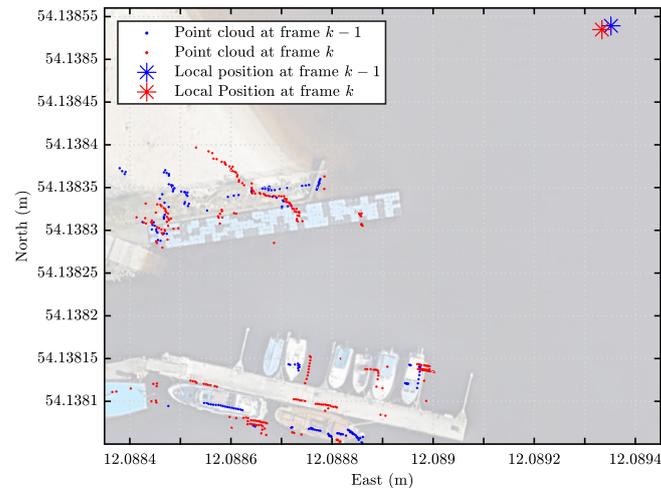


Abbildung 4.8: LiDAR-Punktwolken aus zwei aufeinander folgenden Zeitpunkten $k - 1$ und k , [70]. Kartendaten: ©2020 GeoBasis-DE/BKG(©2009), Google.

statische Umgebung sollte auch für die Aufgabe des autonomen Andockens konstituiert werden.

Die statische Kartierung umfasst drei Hauptalgorithmen, nämlich die Kartierung von Belegungsrastern, die Polyongenerierung und die Kantenanpassung. Die occupancy grid map vervollständigt den belegten Raum in der Umgebung durch die Integration der gegenwärtigen und vergangenen Punktwolken. Diese belegten Flächen werden als mehrere Polygone zusammengefasst. Für automatisierte Andockaufgaben ist es wichtig, die relative Position des Ego-Schiffs zum Dock zu bestimmen. Daher wird ein Verfahren zur Kantenerkennung entwickelt, um die Kantenlinie des Piers zu identifizieren. In den folgenden Abschnitten werden die drei oben genannten Algorithmen vorgestellt.

Occupancy Grid Mapping

Aufgrund der Roll- und Nickbewegung des Ego-Schiffs ändert sich das Sichtfeld des auf den Schiffen montierten LiDAR von Bild zu Bild stark. Abbildung 4.8 zeigt LiDAR-Messungen von zwei aufeinanderfolgenden Zeitstempeln $k - 1$ und k mit der Abtastzeit $\Delta t = 0,1$ s. Das Schwanken des Schiffes verursacht die gleiche Bewegung des auf dem Ego-Schiff montierten LiDAR-Sensors, was zu stark wechselnden LiDAR-Bildern führt. Verschiedene Bereiche der Umgebung werden gescannt, so dass sich die Punktwolken nicht vollständig überschneiden. Die Betrachtung eines einzelnen Bildes der Punktwolke für die statische Umgebungsabschätzung würde zu einer unvollständigen Darstellung der Umgebung führen.

Daher wird in diesem Fall ein Occupancy Grid Map (OGM) verwendet, um die aktuellen Informationen mit denen aus der Vergangenheit zu integrieren. Die OGM wurden in den letzten drei Jahrzehnten häufig für die Wahrnehmung mobiler Roboter verwendet. [71]. Wie in Abbildung 4.9 gezeigt, wird der Raum in 3D-Voxel-Gitter unterteilt, die in absoluten geodätischen Koordinaten lokalisiert sind und als $V = \{v_1, v_2, \dots, v_G\}$ bezeichnet werden. Jedes Voxelgitter v_i speichert eine probabilistische Schätzung seines Zustands, bezeichnet als $P(v_i)$. Diese Schätzung wird durch die Messungen der Punktwolken aktualisiert, die als \mathbf{y}_k bezeichnet werden. Je nach Wahrscheinlichkeit $P(v_i)$ können die Voxelgitter als besetzt, frei oder unbekannt angesehen werden. Dabei sind für jede Entscheidung drei Wahrscheinlichkeitsschwellen vorgegeben: $p_{besetzt}$, p_{free} , $p_{unknown}$. Wenn nicht weniger als ein LiDAR Punkt im Voxelgitter landet, wird die Besetzungswahrscheinlichkeit dieses Gitters als $P(v_i|\mathbf{y}_k) = p_{occupied}$ festgelegt. Wenn der Laserstrahl durch die Voxel-Gitter hindurchgehen kann, ohne eine Reflexion zu erzeugen, wird die Wahrscheinlichkeit dieser Gitter als $P(v_i|\mathbf{y}_k) = p_{free}$ gesetzt. Die Wahrscheinlichkeit der übrigen Gitter wird als unbekannt $P(v_i|\mathbf{y}_k) = p_{unknown}$ gesetzt. Abhängig von der Winkel- und Entfernungsauflösung des LiDAR, in Abbildung 4.9 als $\Delta\phi$ und Δd bezeichnet, ist es möglich, dass das Nachbargitter v_j eines belegten Gitters v_i ebenfalls belegt ist. Daher wird das $P(v_j)$ ebenfalls mit p_{belegt} aktualisiert, wenn die Winkel- und Abstandsdifferenz von v_i und v_j innerhalb von $\Delta\phi$ und Δd liegt, wie in der blau schattierten Fläche in Abbildung 4.9 dargestellt. Diese Berechnung kann auf 3D erweitert werden, indem man die Höhenwinkelauflösung des Sensors berücksichtigt.

Bei der Aktualisierung des Belegungsrasters wird die log-odds-Darstellung $L(\cdot)$ verwendet [72], wobei das log odds ratio des aktuellen Zeitstempels k definiert ist als

$$L(v_i|\mathbf{y}_k) = \log \frac{P(v_i|\mathbf{y}_k)}{1 - P(v_i|\mathbf{y}_k)} \quad (4.21)$$

und die log odds bis zum aktuellen Frame $1 : k$ können bestimmt werden durch

$$L(v_i|\mathbf{y}_{1:k}) = L(v_i|\mathbf{y}_k) + L(v_i|\mathbf{y}_{1:k-1}) - L_0 \quad (4.22)$$

wobei L_0 durch $p_{unknown}$ unter Verwendung von Gleichung (4.21) bestimmt wird. Danach wird die aktuelle Wahrscheinlichkeit der Occupancy Grid Map aktualisiert durch

$$P(v_i|\mathbf{y}_{1:k}) = 1 - \frac{1}{1 + \exp(l(v_i|\mathbf{y}_{1:k}))} \quad (4.23)$$

Durch den Vergleich des aktuellen $P(v_i|\mathbf{y}_{1:k})$ mit dem vordefinierten Schwellenwert $p_{occupied}$ lässt sich feststellen, ob dieses Voxelgitter besetzt ist.

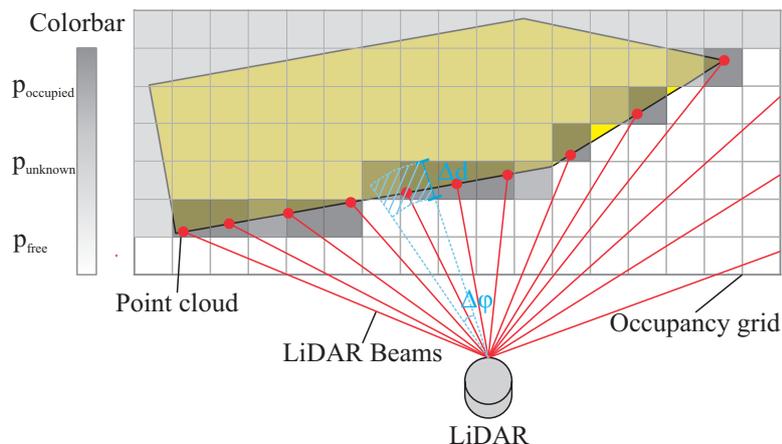


Abbildung 4.9: Beispiel eines Occupancy Grid Map, [70]

Polygon Generierung

Obwohl das OGM den belegten Raum in hoher Auflösung beschreiben kann, kann die Entfremdung des aktuellen belegten Raums über das OGM mit den weiteren Automatisierungsaufgaben hohe Kommunikationslast kosten. Um den Kommunikationsverkehr zu erleichtern, fassen wir den belegten Raum mit Polygonen zusammen, die den Umriss des OGM beschreiben. Wie in Abbildung 4.10 gezeigt, werden die belegten Zellen im interessierenden Gebiet zunächst als Ganzes betrachtet und mit einer konvexen Hülle beschrieben. Anschließend wird der Freiraum der Hülle anhand der Position des Ego-Fahrzeugs definiert, die als Referenzpunkt in Abbildung 4.10 (a) vermerkt ist. Das bedeutet, dass wir für jeden Azimutwinkel einer vollen Drehung den nächstgelegenen gemessenen Punkt nehmen und ihn als Grenze des Freiraums für den jeweiligen Azimutwinkel definieren können. Nachdem dies für eine volle Drehung durchgeführt wurde, kann ein Polygon zwischen den Punkten spline-interpoliert werden. Anschließend wird der Douglas-Peucker-Algorithmus zur weiteren Vereinfachung der Freiraumgrenze eingesetzt. Nach Subtraktion der konvexen Hülle mit dem Freiraum können die belegten Polygone bestimmt werden, wie in Abbildung 4.10 (b) gezeigt.

Pier Erkennung

Beim autonomen Anlegen muss das Schiff die Kante der Pier identifizieren, um seine relative Position zur Pier zu bestimmen. In den meisten Fällen kann die Kante des Piers als Linie dargestellt werden. Daher sollten wir die Linie, die zur Pierkante gehört, aus den Messungen extrahieren, um die relative Position zu bestimmen. Abbildung 4.11 zeigt ein Beispiel für die Punktwolken, als sich das DENEb-Forschungsschiff der Anlegestelle

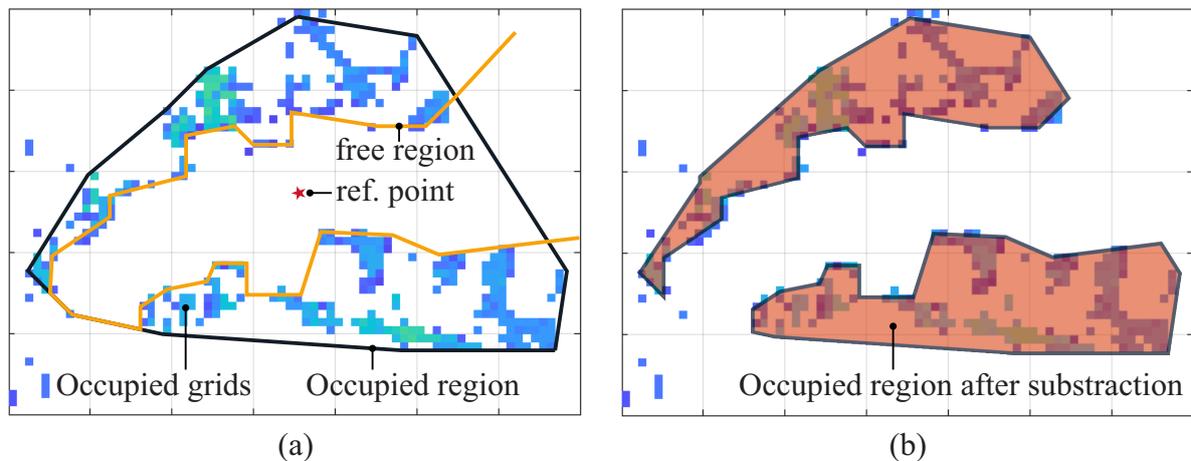


Abbildung 4.10: Prozess der Polygon Erzeugung. Anhand des OGM kann eine konvexe belegte Region bestimmt werden, wie in (a) dargestellt. Außerdem kann eine freie Region um den Referenzpunkt bestimmt werden. Die belegten Polygone werden durch Subtraktion der belegten Region mit den freien Regionen erzeugt, wie in (b) dargestellt, [63]

im Rostocker Hafen näherte, wo die landseitige Infrastruktur mit LiDAR erfasst und als Punktwolken dargestellt werden kann. Um die relative Position des Schiffes zur Pier zu bestimmen, müssen wir die nächstgelegene Linie aus den Punktwolken extrahieren, die die Kante der Pier darstellt.

In dieser Arbeit wird eine auf Random sample consensus (RANSAC) basierende Methode zur Extraktion von Kantenlinien vorgeschlagen. RANSAC ist eine Methode, die ein Modell an experimentelle Daten mit groben Fehlern anpasst [73]. Wie bei der Geradenanpassung ist RANSAC in der Lage, Ausreißer (grobe Fehler) zu erkennen und herauszufiltern, so dass die Geraden mit Punkten angepasst wird, die zu kleinen Residuen der angepassten Geraden führen. Beim Algorithmus RANSAC wird zunächst eine Teilmenge der ursprünglichen Punktwolken zufällig ausgewählt, als hypothetische Ausreißer behandelt und mit Hilfe der Methode der kleinsten Quadrate an eine Linie angepasst. In einem zweiten Schritt prüft der Algorithmus die verbleibenden Punkte, ob sie mit dem hypothetischen Linienmodell des ersten Schritts konsistent sind. Punkte innerhalb der Restschwelle werden als Ausreißer behandelt, und das Linienmodell wird aktualisiert. Andernfalls werden die Punkte als Ausreißer behandelt und herausgefiltert. Die Menge der darin enthaltenen Elemente wird als Konsensmenge definiert. Das Programm RANSAC wiederholt die beiden oben genannten Schritte, bis ein Linienmodell mit genügend Elementen in der Konsensmenge bestimmt ist.

Wie in Abbildung 4.11 gezeigt, stellt die aus den Punktwolken extrahierte, nächstgelegene

ne Linie die Kante der Anlegestelle dar. Es kann jedoch mehr als ein Punktwolkenatz vorhanden sein, der mit Linien angepasst werden kann. Da RANSAC die Suche nach angepassten Linien mit einer Zufallsstichprobe beginnt, kann es nicht garantieren, dass die angepasste Linie die nächstliegende ist. Daher wenden wir den Algorithmus RANSAC wiederholt an, finden alle möglichen Linienmodelle in den Punktwolken und wählen die Linie aus, die dem Ego-Schiff am nächsten liegt, und behandeln sie als Kante des Piers.

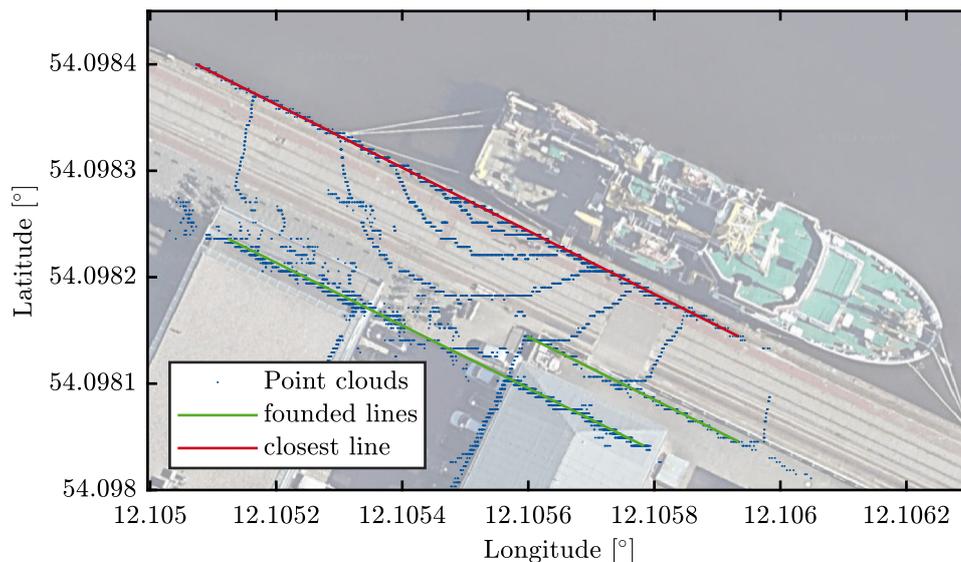


Abbildung 4.11: Beispiel für die Piererkennungsmethode während eines Andockszenarios. Die Kante des Piers kann durch die nächstgelegene Linie dargestellt werden, die aus den Punktwolken extrahiert wurde. Die Google-Karte wird zum Vergleich im Hintergrund verwendet. Kartendaten: ©2020 GeoBasis-DE/BKG(©2009), Google.

4.2.5 Kooperative Lokalisation

In GALILEOnautic 2 wurde die Wahrnehmung und Lokalisierung von vernetzten Mehrfachagentensystemen untersucht. Die Verkehrsteilnehmer werden in vier Klassen unterteilt.

- Vernetzte Schiffe: sind definiert als Schiffe, die innerhalb des lokalen Netzwerks mit dem zentralen Server als vernetzte Schiffe verbunden sind. Dies sind die intelligenten Schiffe, die mit Navigations- und Wahrnehmungssensoren mit bekannten Messgeräuschen ausgestattet sind. Diese vernetzten Schiffe bestimmen ihre Ego-Position mit hoher Genauigkeit und nehmen die Umgebung innerhalb der Reichweite der Wahrnehmungssensoren wahr. Über das LTE-Netzwerk kommunizieren diese Schif-

fe über die Zentralstation miteinander und tauschen ihre Systemzustände in Echtzeit mit einer vordefinierten Aktualisierungsrate aus.

- AIS Vessels: werden als Verkehrsteilnehmer definiert, die nicht mit dem lokalen Netz verbunden sind, aber ihre aktuelle Ego-Position über bordeigene AIS-Geräte an andere AIS-Empfänger übermitteln können. Im Vergleich zu den vernetzten Schiffen haben die mit AIS ausgerüsteten Schiffe unbekannte Messcharaktere, seltene Abstraten und mögliche Messfehler.
- Schwimmende Hindernisse: sind definiert als schwimmende Objekte (wie Bojen und kleine Boote), die weder mit dem lokalen Netz verbunden noch mit AIS-Geräten ausgestattet sind. Diese Hindernisse können keine vorherigen Informationen liefern und können nur von Wahrnehmungssensoren erkannt werden.
- Statische Hindernisse: werden als statische Objekte definiert, die weder mit AIS noch mit Netzwerkgeräten ausgestattet sind. Alle anderen Terrains und Hafeninfrastrukturen werden als statische Hindernisse betrachtet.

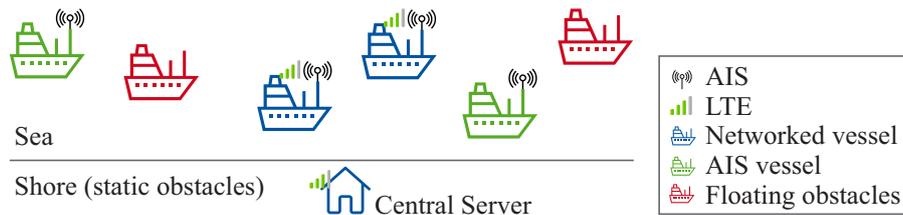


Abbildung 4.12: Klassifizierung der Verkehrsteilnehmer

4.2.6 Zeitsynchronisation

Zur Synchronisierung der Netzwerkteilnehmer haben wir zunächst alle in den vernetzten Systemen verwendeten Uhren synchronisiert, so dass die Multi-Agenten dieselbe Zeitbasis mit vernachlässigbarer Uhrendrift haben. Hier haben wir die Network Time Protocol (NTP) Technik angewandt, um alle Uhren zu synchronisieren. NTP ist ein Netzwerkprotokoll für die Zeitsynchronisierung innerhalb des Netzwerks, das aus NTP-Servern und -Clients besteht. Die Clients senden Pakete an den Server, der diese Pakete sofort zurückschickt. Durch die Berechnung der Zeitverzögerung der Round-Trip-Pakete können die Clients ihre Zeit mit dem Server synchronisieren. NTP verwendet ein Schichtensystem, um die Zeitquellen zu organisieren, und jede Schicht wird als Stratum bezeichnet. Die Schicht 0 bezeichnet hochpräzise Uhren wie Atomuhren oder GNSS. Die Geräte der Schicht $k + 1$ synchronisieren ihre Zeit mit denen der Schicht k innerhalb weniger Mikrosekunden.

In GALILEOnautic 2 wurde ein NTP-Server im lokalen Netz eingerichtet, der sich mit der von der oberen Schicht aus dem Internet empfangenen UTC-Zeit synchronisiert. Die anderen Agenten im lokalen Netz werden innerhalb weniger Mikrosekunden mit diesem NTP-Server synchronisiert.

4.3 Experimentelle Validierung

4.3.1 Navigation im Netzwerk und Umgebungswahrnehmung mit MESSIN

Versuchsaufbau

Die Algorithmen wurden während des Projekts mit MESSIN mehrmals getestet, wie in Abbildung 4.13 dargestellt ist. Der MESSIN wurde mit einem Velodyne Puck VLP-16 LiDAR-Sensor an der Front, einem AIS-Empfänger, einem Septentrio AsteRx-3 GNSS-Empfänger und einem Trägheitsnavigationssystem (INS) ausgestattet. Die Algorithmen werden auf einem Industrie PC Vecow 5100 mit Intel® Core i7 Prozessoren durchgeführt, wobei ROS Melodic auf Ubuntu 20.04 verwendet wird.

Um die Genauigkeit der Objektverfolgung zu überprüfen, verwendeten wir ein Schlauchboot, das mit einem AIS-Sender der Klasse B und einem Septentrio AsteRx-3 GNSS-Empfänger ausgestattet war (rechte Seite in Abbildung 4.13). Der AIS-Sender kann alle 30 sec eine AIS Nachricht senden. Die RTK Lösung des Septentrio-GNSS-Empfängers wird nur als Positionsreferenz verwendet.

Während der Testfahrt haben wir die Algorithmen durch verschiedenen Szenarien getestet. Im Manövrieren Szenario wurde der MESSIN in zum Wasserkanal gefahren, um möglich mehr fahrende Schiffe zu beobachten. Im Anlegen Szenario wurde der MESSIN in dem Alegenpier gefahren, wo die statische hindernisse erkannt werden müssen. Im Netzwerkbeobachtung Szenario wurde das Schlauchboot als ein zweites vernetzte Schiffe betrachtet. Der Zentralserver soll in diesem Fall Netzwerkbeobachtungen aus zwei vernetzte Schiffe integrieren.

Szenario Manövrieren

Während der Testfahrt ist der MESSIN in der Nähe eines der Hauptkanäle im Rostocker Hafen gefahren, so dass mehr Schiffe erkannt und verfolgt werden können. Die aktuellen



Abbildung 4.13: Versuchsaufbau links: Ausrüstungen auf dem MESSIN. Rechts: Sensorausrüstungen auf dem Schlauchboot, [67]; USV MESSIN und dargestelltes Schlauchboot sind Eigentum der Universität Rostock.

Ergebnisse der Multi-Objektverfolgung können gleichzeitig mit dem ROS Visualisierungstool rviz beobachtet werden (Abbildung 4.14). Die LiDAR-Punktwolke (weiße Punkte) wird zunächst zu Clustern gruppiert, die dann in Boundingboxen (als rote Boxen dargestellt) eingepasst werden, um die LiDAR-Messungen für den MOT zu generieren. Diese Boundingboxen werden dann mit der bestehenden Objektbank verknüpft, wobei die Methode in Abschnitt 4.2.3 verwendet wird. Anschließend werden alle erkannten Objekte mit der in Abschnitt 4.2.3 vorgeschlagenen Methode verfolgt, und die Echtzeit-Ergebnisse werden mit grünen Ellipsen dargestellt.

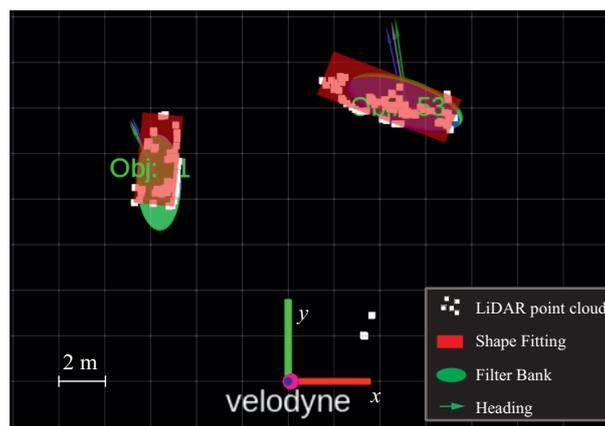


Abbildung 4.14: Visualisierung der LiDAR Punkte, Boundingboxen und MOT Objekte mit der ROS rviz Toolbox

Abbildung 4.15 zeigt die Positionierungsergebnisse des MOT in einer Testfahrt. der MES-

4 Navigation im Netzwerk und Umgebungswahrnehmung

SIN bewegt sich in einem beliebigen Fahrplan, während das Schlauchboot mit hoher Dynamik um den MESSIN gefahren wird. Während der Fahrt hat das LiDAR ein Schiff mit AIS und ein kleines Boot ohne AIS erkannt. Alle empfangenen AIS-Nachrichten sind mit Sternen dargestellt, die LiDAR-Messungen mit grauen Punkten. Anhand dieser Messungen demonstrieren wir die Positionsschätzung der Filterbank: Alle erkannten Objekte in der Ansicht können mit einer Rate von 10 Hz verfolgt werden, selbst wenn einige Objekte, z.B. das Schlauchboot, eine hohe dynamische Bewegung aufweisen. Die mit AIS ausgestatteten Objekte können ebenfalls mit dem AIS-Subfilter verfolgt werden, auch wenn sie zu Beginn nicht im Blickfeld des Sensors sind (das Schiff mit AIS wird als blauer Stern und Punkt dargestellt). Allerdings ist die Genauigkeit aufgrund falscher Kurs- und Geschwindigkeitsinformationen eingeschränkt. Auch dieses Schiff wird zunächst von LiDAR erfasst, bevor es mit der aktuellen AIS-Nachricht abgeglichen wird. Daher wird zu Beginn nur der Subfilter LiDAR ausgeführt (rote Punkte). Sobald dieses Schiff mit der AIS-Nachricht übereinstimmt, wird die gesamte Filterbank ausgeführt, was zu einer robusten Schätzung führt (gelbe Punkte). Mit Hilfe von LiDAR-Messungen können wir die Positionsgenauigkeit der MOT weiter verbessern. Wenn die Schiffe nicht mit AIS ausgestattet sind oder keine AIS-Messungen senden, wird nur der Subfilter LiDAR durchgeführt, um eine vollständige Verfolgung in der Sensoransicht zu gewährleisten.

Abbildung 4.16 zeigt das Verfolgungsergebnis des Schlauchbootes im Vergleich zur Referenz. Um die Leistung und den Nutzen der gesamten Filterbank zu überprüfen, zeigt Abbildung 4.16 auch die drei reinen Subfilter-Ergebnisse ohne gemeinsame Rückkopplung. Obwohl das Subfilter-LiDAR eine relativ genaue Positionierung bieten kann, kann es aufgrund schlechter Sicht oder turbulenter Störungen gelegentlich zu großen Fehlern kommen. Wie in Abbildung 4.16 gezeigt, werden AIS-Nachrichten etwa alle 30 s oder sogar noch länger mit einer Positionsgenauigkeit von 5 bis 10 m empfangen. Aufgrund dieser begrenzten Positionierungsgenauigkeit und der seltenen Aktualisierungshäufigkeit liefert die reine AIS-Lösung die schlechteste Positionierungs- und Verfolgungsleistung. Allerdings sind in der Nachricht genaue Informationen über die konstante Größe des Schiffes enthalten. Daher integriert der Subfilter-Mix die genaue LiDAR-Positionsmessung und die AIS-Größeninformationen als Messungen. Dennoch kann es, wie oben erwähnt, durch falsche oder fehlende Messungen zu Abweichungen kommen. Im Vergleich zu den Einzelfilterlösungen kann die Filterbank die beste Messkonstellation in jeder Periode nutzen und den geringsten 2D-Positionsfehler verursachen. Normalerweise liegt die Wahrscheinlichkeit der Subfilter Mix nahe bei eins, so dass die Filterbank die Position und Größe mit hoher Genauigkeit schätzen kann. Die LiDAR-Messungen fehlen bei 60 s und 160 s. Die Filterbank kann die Wahrscheinlichkeit des Subfilters AIS erhöhen, so dass die Schätzung immer noch anhand der AIS-Messungen aktualisiert werden kann.

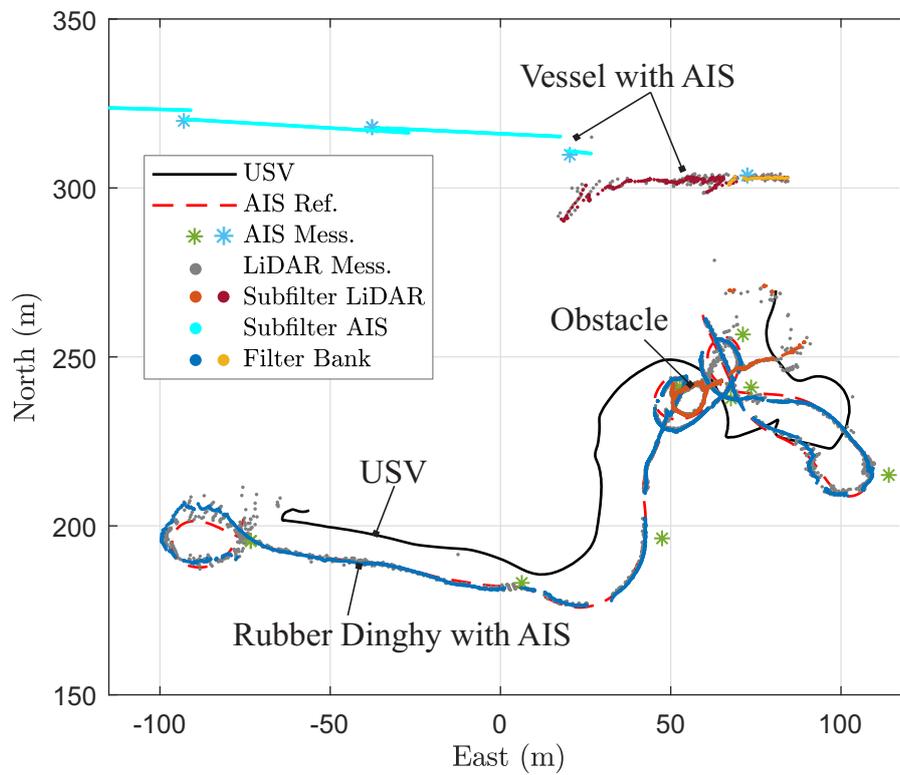


Abbildung 4.15: Ergebnisse des Algorithmus in der realen Welt. Die NED-Koordinaten basieren auf dem geodätischen Referenzpunkt $[54.138247^\circ, 12.08941316^\circ, 0 \text{ m}]$, [67]

4 Navigation im Netzwerk und Umgebungswahrnehmung

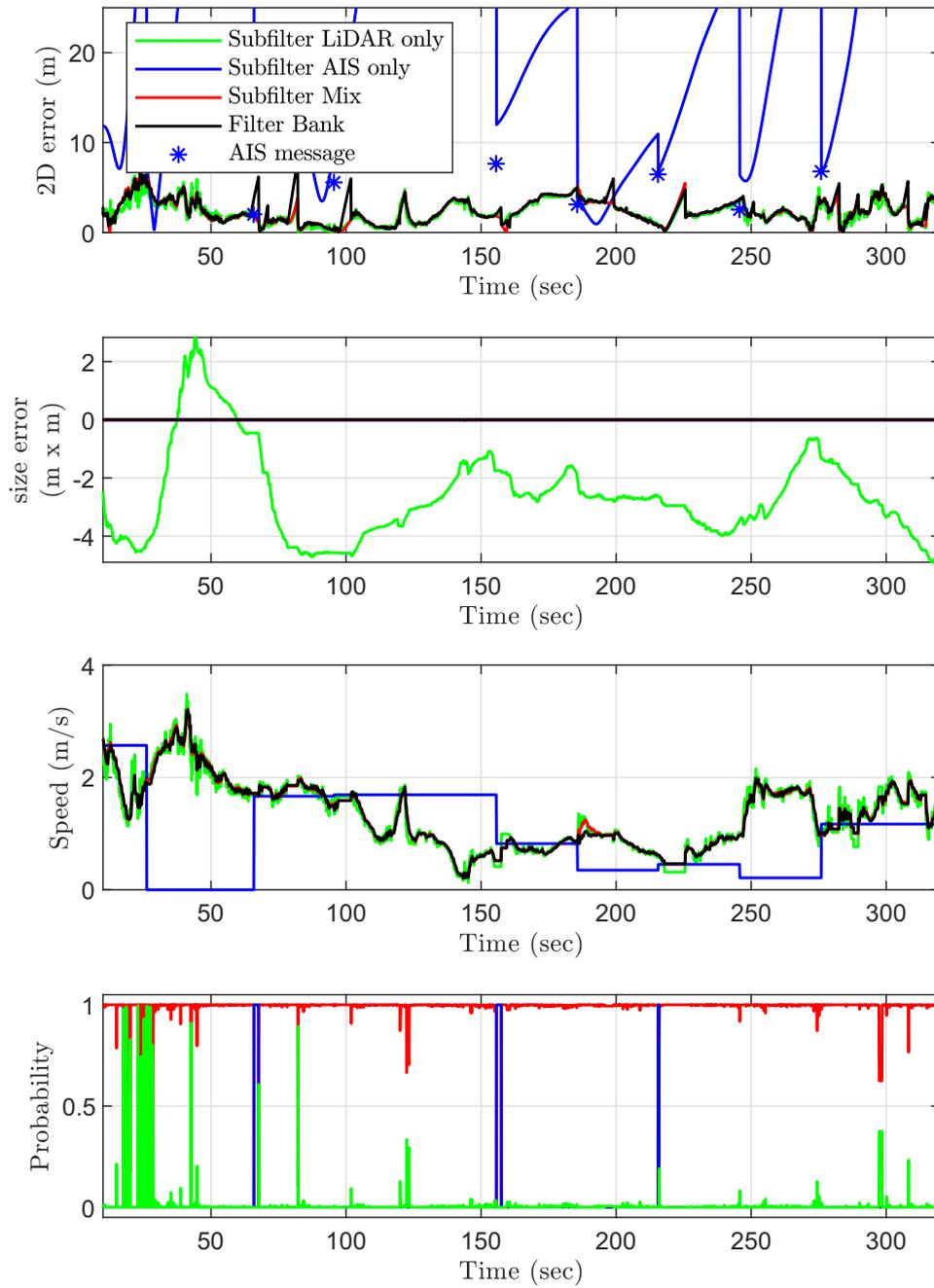


Abbildung 4.16: Tracking-Ergebnisse des Schlauchbootes im Vergleich zur Referenz, [67]

Szenario Anlegen

Das USV wird in den Pier gefahren, während ein Schlauchboot vor dem USV herfährt. In diesem Szenario wird die Performance der OGM, der Objektzuordnung und der dynamischen Objektverfolgung validiert. Abbildung 4.17 zeigt eines der Evaluierungsszenarien, bei dem wir die statische Umgebung zusammen mit den geparkten Schiffen als Referenz mit der Leica Nova MS60 MultiStation gescannt haben und dabei eine Millimetergenauigkeit erreichten. Nach dem Vergleich mit fünf über RTK GNSS gemessenen Referenzpunkten wurden die gescannten Punkte in ein gemeinsames geodätisches Referenzsystem transformiert. Die statischen Formen der Piers und die Bounding Boxes wurden manuell nachgezeichnet.

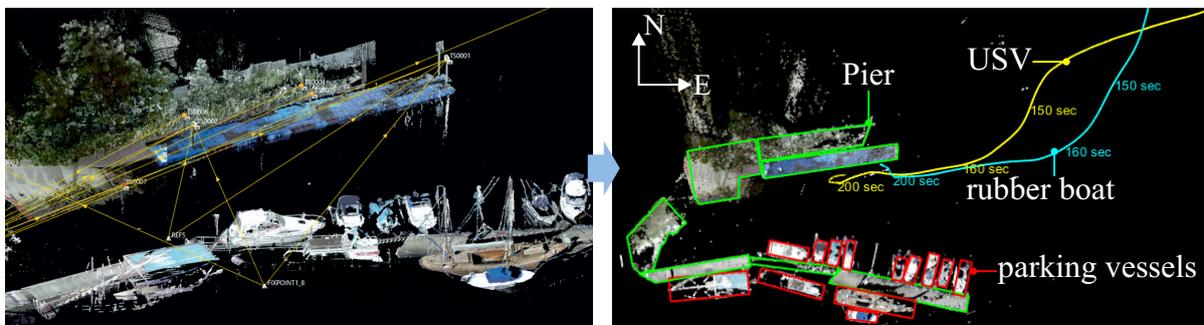


Abbildung 4.17: Performance des OGM und Objektverfolgung im Anlegen-Szenario, [63]

Abbildung 4.18 zeigt die Ergebnisse des Algorithmus, wenn das USV in einer Zeitsequenz in die Nähe des Docks gefahren wird. Das erste Bild der vom LiDAR wahrgenommenen Sequenz ist in Abbildung 4.18(a) dargestellt. Hier wird ein OGM mit einer Zellauflösung von 10cm durchgeführt. Nach 2 Sekunden wird mittels OGM eine deutlich klarere Wahrnehmung des Ufers aufgebaut, obwohl das USV noch nicht weit gefahren ist (Abbildung 4.18 (b)). Bei der Verwendung von OGM werden auch die Belegungsinformationen der vergangenen Frames berücksichtigt. In diesem Fall werden die oszillierenden Bewegungen der USV zu einem Vorteil statt zu einer Störung. Ein Teil des unbekanntes Raums wird gescannt und vervollständigt. Je näher das USV an das Dock herangefahren wird, desto mehr Details des Golfs werden wahrgenommen, wie in Abbildung 4.18 (c). Da auch der freie Raum um das Objekt herum berücksichtigt wird, bleibt die Kante des Objekts in OGM immer noch scharf, obwohl einige Objekte wie parkende Schiffe geringfügige Bewegungen auf dem Wasser aufweisen. Außerdem ist zu beobachten, dass das OGM eine vollständige 3D-Darstellung des Piers bieten kann, bevor das USV in ihn einfährt. Das dynamische Objekt ist in der orangefarbenen Punktwolke in Abbildung 4.18 dargestellt, bei dem es sich um das Schlauchboot handelt. Die Geschwindigkeits- und Entfernungsschwellenwerte für dynamische und kritische Objekte werden auf 0,2 m/s und 30 m festgelegt.

4 Navigation im Netzwerk und Umgebungswahrnehmung

In diesem Fall wird das Schlauchboot verfolgt, sobald es sich in Bewegung setzt.

Um die Genauigkeit des OGM quantitativ zu validieren, wurde die OGM mit der Google Map in 2D verglichen, als dargestellt in Abbildung 4.18 (d). Die OGM in NED-Koordinaten wird in geodätische Koordinaten umgewandelt und auf der Google Map[©] eingezeichnet. Es wird erwartet, dass der ständig besetzte Raum auf der Google Map, wie z. B. das große parkende Schiff, die Küstenlinie und der Golf, in der OGM als besetzt bestimmt werden sollte. In Abbildung 4.18 (d) wird der Rand des besetzten Raums in verschiedenen Höhen mit OGM dargestellt, was den Erwartungen der Umweltwahrnehmung entspricht. Es sollte erwähnt werden, dass die Genauigkeit des OGM im Vergleich zur Reallokation von der Genauigkeit der globalen Positions- und Orientierungsmessungen abhängt.

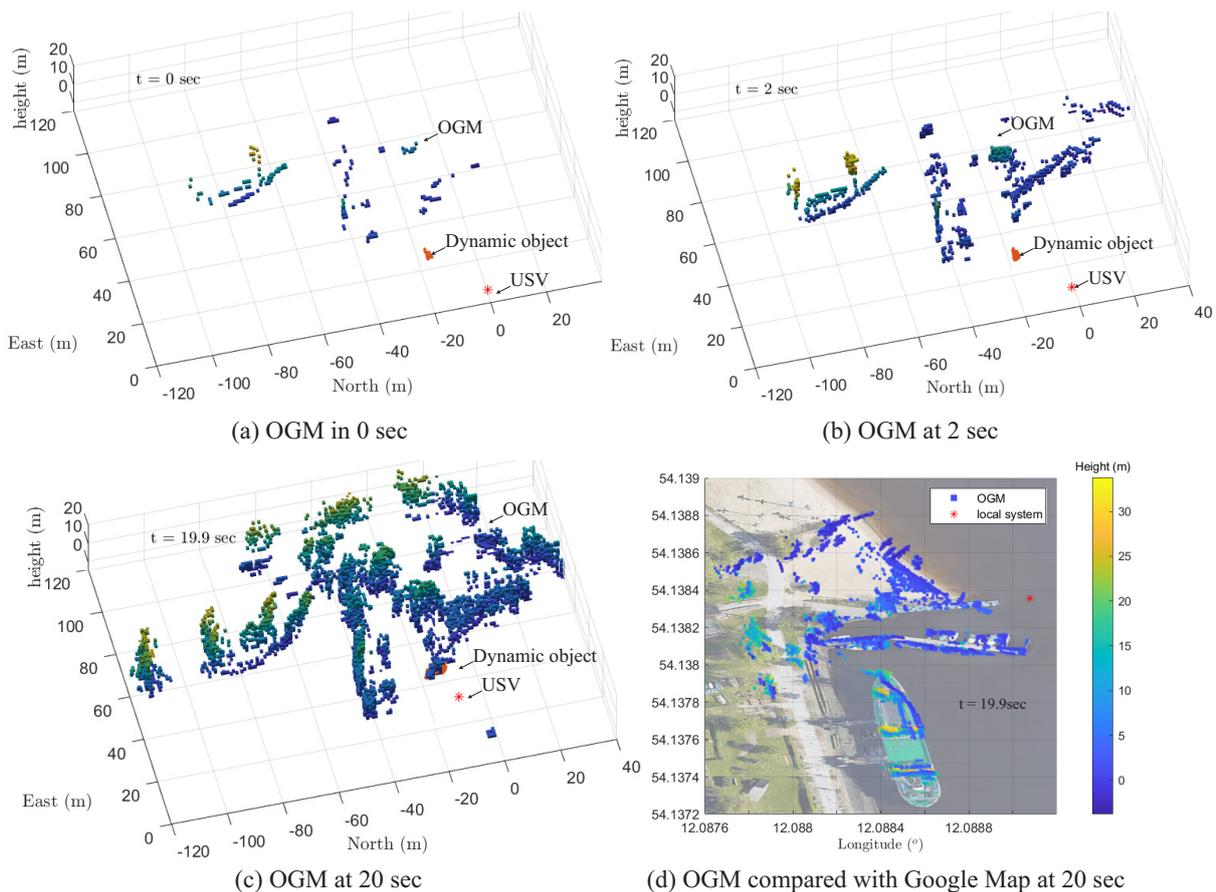


Abbildung 4.18: Performance des OGM und Objektverfolgung im Anlegen-Szenario, [70]. Kartendaten: ©2020 GeoBasis-DE/BKG(©2009), Google.

Abbildung 4.19 zeigt zwei Zeitstempel der Umgebungswahrnehmung während dieser Zeitspanne. Bei 164 s wurde die Mehrheit der parkenden Schiffe erkannt und verfolgt, bevor das USV in den Hafen einfuhr, wobei sich die meisten Erkennungen mit der Referenz

überschnitten. Mehrere Schiffe, die weit von der USV entfernt waren, wurden jedoch nicht erkannt und als Teil der statischen Umgebung behandelt und durch statische Polygone dargestellt. Als das USV in den Hafen einfuhr, wurde die statische Karte weiter vervollständigt, und es wurden mehr Schiffe erkannt, die sich tiefer im Hafen positionierten.

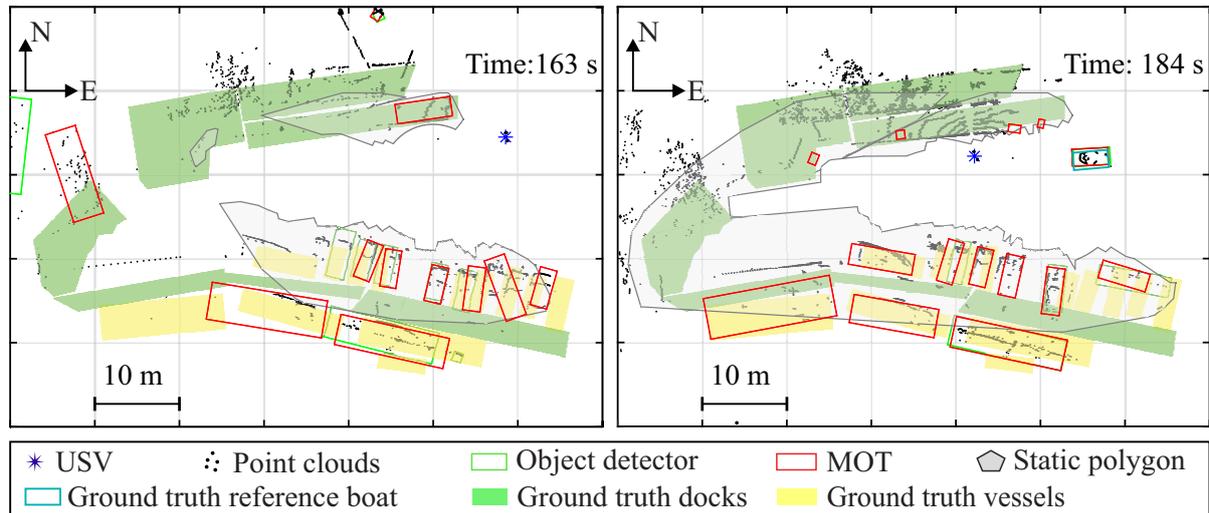


Abbildung 4.19: Performance des OGM und Objektverfolgung im Anlegen-Szenario, [63]

Szenario Netzwerkbeobachtung

Um die Netzwerkbeobachtung Szenario zu realisieren, wurde das Schlauchboot erst als ein vernetztes Schiff ausgestattet. Abbildung 4.13 rechts zeigt die Ausstattung des Schlauchboots, welches mit Septentrio GNSS Empfänger und Micro Strain G5 IMU montiert ist. Außerdem ist ein Velodyne Puck VLP 16 LiDAR ausgestattet für Umgebungswahrnehmung. Durch LTE und CGI Gateway kann das Schlauchboot mit Zentral Server kommunizieren

Abbildung 4.20 zeigt die Ergebnisse der Netzwerkbeobachtung während einer Testfahrt. Der MESSIN und das Schlauchboot fahren mit einem beliebigen Fahrplan und konnten die Objekte in der Umgebung detektieren. Die detektierten Objekte wurden als Bounding Boxes an Zentral Server gesendet. In dem Zentral Server wurden die Objektbeobachtungen integriert und visualisiert.

4.3.2 Umgebungswahrnehmung mit DENEb

In GALILEOnautic 2 wurden die Wahrnehmungsalgorithmen zur Unterstützung des autonomen Anlegens des Forschungsschiffs DENEb eingesetzt. In dem autonomen Anlegenszenario wurde hauptsächlich die Pier-Erkennung verwendet und evaluiert. Das Modul zur

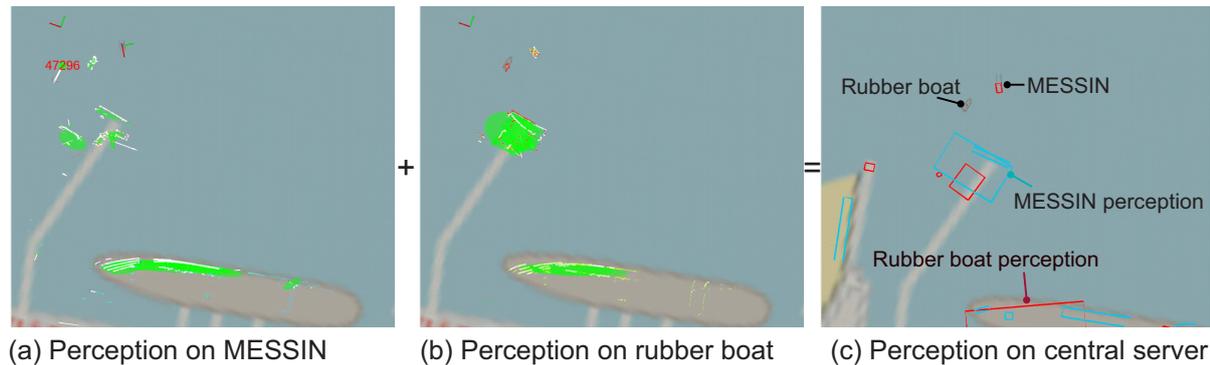


Abbildung 4.20: Ergebnisse der Netzwerkbeobachtung. Kartendaten: [74]

Umgebungswahrnehmung empfängt Messungen von LiDAR, Radar und AIS und soll die Aufgaben der Umgebungswahrnehmung erfüllen, die Voraussetzung für das automatische Anlegen sind. In den folgenden Abschnitten werden die Versuchsaufbauten und Ergebnisse erörtert.

Versuchsaufbau

Wie in Abbildung 4.21 dargestellt, wurde die DENEb mit einem Velodyne Puck VLP-16 LiDAR-Sensor, einem AIS-Empfänger, einem X-Band-Marineradar von Raytheon Anschütz, einem Septentrio AsteRx-3 GNSS-Empfänger und einer Microstrain GX5 SIMU ausgestattet. Der Navigationsfilter bestimmt die 6D-Position der DENEb mit einer Positionsgenauigkeit von 20 cm und einer Orientierungsgenauigkeit von 1° , indem er die GNSS-, IMU- und Kompassmessungen verwendet. Die Kante des anzudockenden Piers wurde mit GNSS gemessen RTK-Lösung mit einer Genauigkeit von unter 5 cm gemessen. Die referenzierte relative Pose zwischen dem Pier und der DENEb wurde durch Subtraktion der Schiffspose vom Navigationsfilter mit der Kantenlinie des Piers bestimmt. Die Referenzlage hat somit eine Genauigkeit von etwa 25 cm.

Da die Größe der DENEb nicht zu vernachlässigen ist, sollten wir die Position aller Wahrnehmungssensoren messen, um eine genaue lokale Positionierung und Umgebungswahrnehmung zu gewährleisten. Zu diesem Zweck verwendeten wir eine Leica MS 60 MultiStation, um den DENEb zu scannen und Punktwolken mit einer Auflösung von ca. $1,5 \text{ cm} \times 1,5 \text{ cm} \times 1,5 \text{ cm}$ und einer Genauigkeit von ca. 5 cm zu erzeugen. Abbildung 4.21(b) zeigt die verwendete MultiStation während der Sensorpositionierung und des Rumpfschannens. Anschließend bestimmen wir mit demselben Gerät die Position des Sensors in Bezug auf den Rumpf mit einer Genauigkeit von weniger als 1 cm. Durch den Vergleich des gescannten Rumpfes mit einem Schiffsmodell, einschließlich des Schwerpunkts, haben wir alle Sensorpositionen im Schiffsrumpf bestimmt, deren Ursprung im

Schwerpunkt liegt. Die Positionen der einzelnen Sensoren auf der Grundlage des Körperrahmens sind in Tabelle 4.2 dargestellt. Während des Tests fuhr die DENEb von der Pier in der Nähe des Bundesamtes für Seeschifffahrt und Hydrographie (BSH) im Rostocker Hafengebiet ab und dockte an derselben Pier an. Während des Tests sollte die DENEb die relative 2D-Position des Zentrums zur Pier bestimmen, die dann gesteuert werden kann. Außerdem sollte die statische Umgebung kartiert und die dynamischen Objekte in der Umgebung erkannt und verfolgt werden.



Abbildung 4.21: Sensoreinstellungen am Forschungsschiff DENEb des BSH: (a) und Messung der Sensorenposition mit der Leica MultiStation (b)

Tabelle 4.2: Location of the used sensors on DENEb

Sensor	Location in body frame		
	x^b [m]	y^b [m]	z^b [m]
GNSS + IMU	-3.95	4	6.5
LiDAR	2.38	-16.79	0.01
Radar	1.42	-18.14	8.11
Compass	1.42	-18.14	0.1
AIS	1.3	-16.7	0.17

Ergebnisse

Abbildung 4.22 zeigt das Andockszenario der DENEb und den entsprechenden Schnappschuss der statischen Kartierungsergebnisse in ROS gazebo. Wie in Abbildung 4.22 (a) dargestellt, konnte die Infrastruktur an Land als Punktwolke erkannt werden, während sich die DENEb in der Nähe des Piers befand. Durch die Verwendung der OGM und der statischen Polygongenerierung, wie in Abschnitt 4.2.4 beschrieben, konnte die vorgeschlagene Erkennungsmethode die statische Infrastruktur in der Ansicht erkennen und sie

4 Navigation im Netzwerk und Umgebungswahrnehmung

mit Polygonen darstellen. Für einen qualitativen Vergleich wurde eine Satellitenkarte als Hintergrund verwendet.

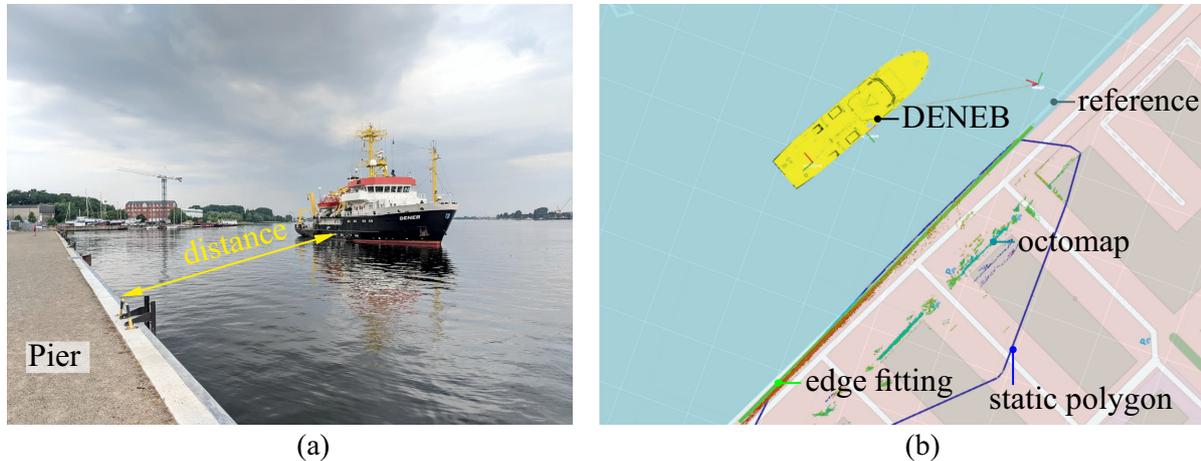


Abbildung 4.22: Anlegeszenario mit dem Forschungsschiff DENEBO des BSH und die entsprechende Visualisierung in ROS rviz. Kartendaten: [74]

Abbildung 4.23 zeigt vier typische Zeitstempel des Andockszenarios. Da die DENEBO mit dem bordseitigen GNSS-System AIS-Messungen erzeugen kann, kann sie zur Validierung des AIS-Teils des MOT-Algorithmus verwendet werden. Durch die Verwendung der AIS-Messungen konnte der MOT-Algorithmus die Position der DENEBO mit 10 Hz schätzen, selbst wenn die AIS-Messungen nur gelegentlich verfügbar waren. Während sich die DENEBO dem Pier näherte, konnten immer mehr Details der statischen Infrastruktur erkannt werden, und der OGM wurde nach und nach vervollständigt. Auch die Kante des Piers wurde erkannt und mit einer Linie versehen.

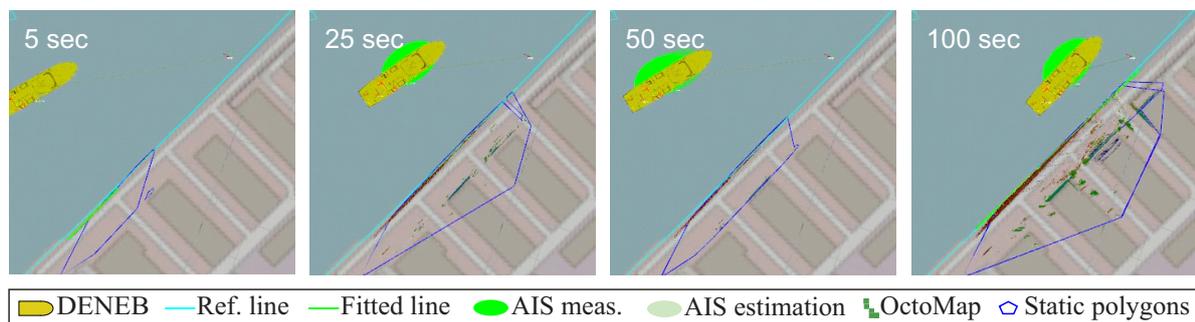


Abbildung 4.23: Wahrnehmungsergebnisse zu vier typischen Zeitpunkten während des Andockszenarios. Kartendaten: [74]

Um die Genauigkeit der statische Kartierung zu bewerten, verglichen wir die durch die Piererkennung ermittelte relative Position mit der Referenz, wie in Abbildung 4.24 dargestellt. Die Kante konnte erkannt werden, als die DENEBO weniger als 60 m vom Pier

4.3 Experimentelle Validierung

entfernte. Als sich die DENEb dem Pier näherte, verringerten sich der Abstand und der Kursfehler weiter, da mehr Details erkannt wurden. Die Fehler nahmen jedoch zu, wenn der Abstand weniger als 5 m betrug. Dies ist auf die Montageposition des LiDAR zurückzuführen. Die Kante des Piers fiel in den Blindbereich des Sensors, da die DENEb an den Pier angedockt war. Daher wurde am letzten 2 m keine Kantenlinie gefunden. Dieses Problem kann jedoch durch die Installation der Wahrnehmungssensoren in der Nähe der Schiffsreling oder durch die Ausstattung des Schiffes mit mehr Wahrnehmungssensoren gelöst werden. Insgesamt konnte die Kantenanpassung die relative Lage des Schiffes mit einer Abstandsgenauigkeit von 0,44 m und einer Kursgenauigkeit von $1,32^\circ$ bestimmen, was der Genauigkeit der referenzierten Lage nahe kommt.

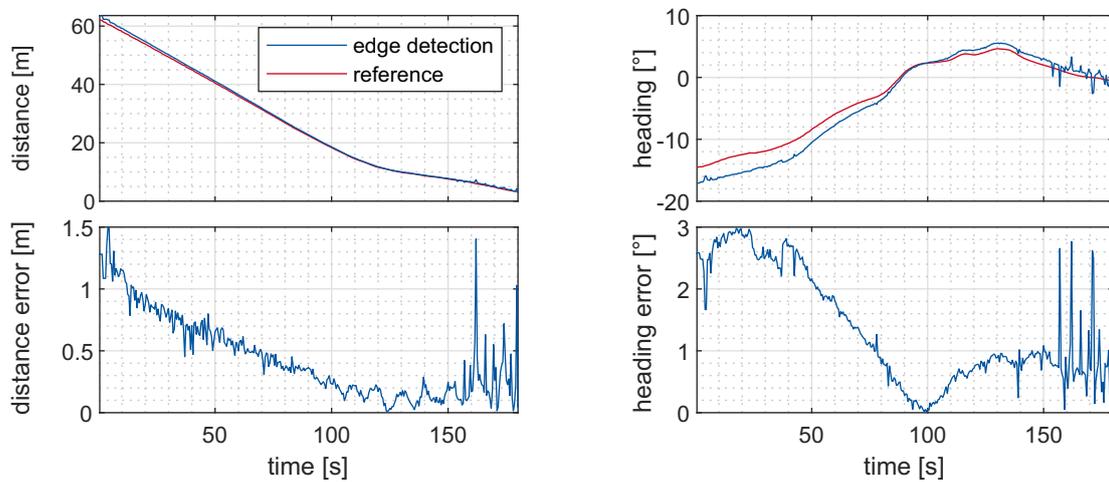


Abbildung 4.24: Ergebnisse der Piererkennung und die entsprechenden Fehler beim Andockszenario

5 Projektergebnis

Autoren: Jan-Jöran Gehrt, René Zweigel

5.1 Bewertung der durchgeführten Arbeiten

5.1.1 Notwendigkeit und Angemessenheit

Um die Vorteile des europäischen Satellitennavigationssystems für Automatisierungsentwicklungen nutzbar zu machen und diese Entwicklungen in die deutsche Industrie zu tragen, müssen anwendungsnahe Forschungs- und Entwicklungsprojekte in enger Zusammenarbeit von Industrie und Wissenschaft durchgeführt werden. Insbesondere ist es dabei wichtig von der simulativen Validierung und der Umsetzung im kleinen Maßstab in den Bereich der für die Industrie relevanten Anwendungen zu kommen. Dieser Aspekt wurde in GALILEOnautic 2 adressiert. Dazu wurde das Systemkonzept des Vorgängerprojekts GALILEOnautic skalierbar gemacht und auf industriell relevante Versuchsträger neu zum Konsortium hinzugekommener Projektpartner gebracht.

Zu den technischen Entwicklungen im Projekt konnte die RWTH im Rahmen des Projekts GALILEOnautic 2 einen wichtigen Beitrag leisten, in dem es ein hochgenaues und robustes Navigationsfilter integrierte, das GPS- und Galileo-Satellitensignale gleichzeitig nutzte und damit die Vorteile beider Navigationssysteme vereinte. Die Modularität des Filters erlaubte die Integration maritimer Sensoren und Korrekturdaten: in GALILEOnautic 2 erfolgte die Integration eines DVLs, eines Kompasses und externer Korrekturdienste von SAPOS und WSV. Des Weiteren wurde nachgewiesen, dass mit entsprechender Algorithmik LiDAR-gestützte Erkennungssysteme zuverlässige Objekterkennung und -verfolgung in der maritimen Anwendungsumgebung liefern, die sich bspw. eignet um damit automatisierte Anlegemanöver zu fahren.

Die gewährte Zuwendung für die RWTH im Rahmen des Förderprojekts GALILEOnautic 2 war notwendig, da für das Vorhaben mit Ausnahme der vorhandenen Grundausstattung

5 Projektergebnis

keine Mittel aus der Grundfinanzierung zur Verfügung standen, die für das Forschungsprojekt eingesetzt werden konnten. Der finanzielle und zeitliche Aufwand für das Vorhaben lag im Bereich dessen, was für vergleichbare Forschungsvorhaben üblich ist.

Es wurde ein hoher Versuchsaufwand betrieben und umfangreiche Integrationstests vor allem im Hafen Rostock durchgeführt, die es der RWTH ermöglichten, ihre Expertisen im Themenfeld Navigationsfilter und Umfeldwahrnehmung für maritime Anwendungen auszubauen. Die erzielten Ergebnisse und Erkenntnisse wurden in Berichten und Projektdokumenten dokumentiert und mittels Veröffentlichungen und wissenschaftlichen Publikationen Dritten zugänglich gemacht. Die entsprechenden Messdaten können für zukünftige Entwicklungen zu Vergleichszwecken wiederverwendet werden. Es lag kein außergewöhnlicher Personalaufwand vor.

5.1.2 Voraussichtlicher Nutzen

Ein Beispiel für die Verwertung der entwickelten Ansätze und Methoden ist die Etablierung der Lehrveranstaltung *Navigation und Sensorfusion in der Regelungstechnik* im Master *Automatisierungstechnik* an der RWTH Aachen University. Diese Veranstaltung hat zum Ziel, den Studierenden die Grundlagen der Satelliten-basierten Navigation und Sensordatenfusion zu vermitteln. Darüber hinaus lernen die Studierenden anhand der realen Ergebnisse aus GALILEOnautic 2 wie diese Grundlagen in der Praxis erfolgreich umgesetzt werden können.

Die FDE-Funktion des Integritätssystems unterstützte das Navigationsfilter, indem es die Messung mit optimaler Qualität auswählt und diejenigen mit begrenzter Qualität ausschließt. Dadurch wird die Genauigkeit und Zuverlässigkeit der Navigationslösung verbessert. Die Abschätzung eines Zuverlässigkeitsbereichs der Positionsangabe wurde untersucht und validiert. Eine mögliche zukünftige Verwendung dieses Zuverlässigkeitsbereichs ist die Integration in das Modul zur Generierung von Trajektorien. Indem der ermittelte Zuverlässigkeitsbereich als zuverlässige Schätzung der Unsicherheit der Navigationslösung behandelt wird, kann dadurch das durch Navigationsungenauigkeiten verursachte Risiko besser bewertet werden.

Da die Algorithmen zur Umgebungswahrnehmung modular entwickelt wurden, können die Algorithmen zur Wahrnehmung und Netzwerkbeobachtung zusammen oder separat für andere Anwendungen verwendet werden, bspw. für Projekte der automobilen Mobilität. Während des Projekts wurden zudem umfangreiche Datensätze als Referenz für Navigations- und Wahrnehmungsansätzen aufbereitet, dokumentiert und veröffentlicht.

5.2 Externe Fortschritte

In diesem Abschnitt werden externe Fortschritte in Bezug auf projektrelevante Entwicklungen dargestellt, die in dem Vorhabenzeitraum erzielt wurden. Diese Fortschritte werden im Kontext der in GALILEOnautic 2 erreichten Ergebnissen bewertet.

Satelliten-basierte eng-gekoppelte Sensorfusion

Die Entwicklungen im Bereich hochgenauer eng-gekoppelter Navigationsfilter durch Trägerphasenintegration und Mehrdeutigkeitsauflösung wird in Deutschland beispielsweise durch die Firma ANavS GmbH vorangetrieben. Insbesondere werden die Bereiche Automobil, Haushaltsroboter und Drohnen adressiert. Seit 2019 ist ANavS im Forschungsprojekt *PREPare Ships* aktiv und beschäftigt sich in diesem Zusammenhang auch mit der maritimen Navigation.

Trägerphasen-gestützte Systeme für den Einsatz in der maritimen Anwendung wurden im Förderzeitraum auch in anderen Projekten untersucht. Dabei wurden meist insbesondere neuartige PPP-Verfahren untersucht, die die PPP-Korrekturen in der State-Space Darstellung formulieren und nutzen, z.B. in *SCIPPPER*. Um der langen Konvergenzzeit des PPP zu begegnen, wird seit 2019 im Projekt *PREPare SHIPS* die Kombination von PPP und RTK untersucht, wobei die schlechte Verfügbarkeit bzw. Zuverlässigkeit der für RTK benötigten differenziellen Daten durch die Hochverfügbarkeit der PPP-Daten und andersherum die lange Konvergenzzeit von PPP durch die kurze Konvergenzzeit von RTK ausgeglichen werden soll. Die in GALILEOnautic 2 entwickelten Algorithmen kompensieren Verbindungsabbrüche von differenziellen Daten durch lokalen Schätzung eben dieser Daten, sodass die Integration von PPP hier nicht notwendig war.

Das aktuelle Satellite-Based Augmentation Services (SBAS) EGNOS-System liefert lediglich Korrekturen für GPS-Satelliten. Zukünftige Entwicklungen weisen in Richtung des SBAS V3, mit dem ebenfalls Galileo-Korrekturen möglich sein sollen. Insbesondere die höhere Verfügbarkeit gegenüber dem Satellitenpositionierungsdienst der deutschen Landvermessung (SAPOS) könnte ein entscheidender Vorteil sein. Darüber hinaus ist auf europäischer Ebene geplant, dass ab 2022 auf dem E6-B Signal aller Galileo-Satelliten Korrekturen für Galileo-Signale zur Verfügung stehen. Dieser ursprünglich als kostenpflichtig entwickelte High Accuracy Service (HAS) wird nach einem Beschluss aus dem Jahr 2018 zukünftig kostenlos sein. Mit dem HAS von Galileo sollen Genauigkeiten kleiner als 20 Zentimeter mit PPP-ähnlichen Verfahren erreichbar sein.

Die Versorgung mit differenziellen Daten, die in GALILEOnautic 2 via LTE umgesetzt wurde, könnte zukünftig über das neue maritime Kommunikationsformat VHF Data Exchange System (VDES) umgesetzt werden. Aktive Forschung dazu wurde im Projekt

5 Projektergebnis

SCIPPER betrieben. Die Idee wird aktuell auch im Projekt *PREPare SHIPS* fortgeführt.

Integrität

Der Schwerpunkt der Überwachung der Navigationsintegrität in jüngster Zeit oder derzeit liegt nach wie vor auf GNSS. Nicht viele Projekte haben die Integrität des gesamten Navigationssystems und die Sensorfusion berücksichtigt.

Im Projekt *LAESSI* wurde die Verbesserung der Sicherheit von Binnenschiffen angestrebt, wobei ein Integritätssystem als wesentlich angesehen werden sollte. Wie in [75] erwähnt, wurde die geschätzte Kovarianz der Systemzustände in einem integrierten Integritätssystem genutzt. Das *LAESSI*-Projekt stützte sich jedoch ausschließlich auf GNSS-Navigation. In *GALILEOnautic 2* hingegen wurden alle Sensoren des Navigationsfilters in die Integritätsuntersuchung einbezogen.

Das EGNSS-Projekt *Hull to Hull (H2H)* betraf die Entwicklung eines intelligenten Assistentensystems zur Unterstützung der Betreiber von Binnenschiffen. Das kommerzielle INS Ekinox-2E wurde für die Sensorfusion verwendet. Ekinox-2E bietet eine integrierte Integritätsprüfungsfunktion. Ob die kommerzielle Integritätsprüfungsfunktion ausreichend war, wurde nicht bewertet und lag außerhalb des Rahmens dieses Projekts.

Im Projekt *SCIPPER* ging es um die automatische Einfahrt in eine Wasserstraßenschleuse, was strenge Sicherheitsmaßnahmen erfordert. Die Integritätsüberwachung des Navigationssystems wurde durch den Vergleich der mittels Navigationslösung geschätzten Baseline, mit den zuvor eingemessenen, tatsächlichen Antennenpositionen realisiert. Dieses Integritätssystem ist auf die Funktion beschränkt zu entscheiden, ob die geschätzte Navigationslösung zutreffend ist.

Umfelderkenntung

Neben der klassischen Integration von ARPA und AIS haben viele Studien Videokameras (Infrarot- und elektro-optische) zur maritimen Überwachung eingesetzt, [76]. In Bezug auf die Objekterkennung wurden neuronale Netze, wie das berühmte You only look once (YOLO) Framework [77], zur Extraktion der Objekte aus dem Bild eingesetzt [78]. Bei diesen auf maschinellem Lernen basierenden Verfahren werden zuvor markierte Datensätze verwendet, um die neuronalen Netze zu trainieren, so dass die neuronalen Netze die Muster der Objekte (Schiffe oder Bojen) erkennen können. Videokameras reagieren jedoch sehr empfindlich auf Beleuchtungs- und Wetteränderungen und haben den Nachteil, dass sie nur eine geringe Genauigkeit bei der Tiefenerkennung aufweisen und sehr empfindlich auf ungünstige Wetterbedingungen reagieren, so dass die Objektextraktion bei stürmischer See oft problematisch ist [76]. Daher werden Videokameras häufig zur Objektklassifizierung im Rahmen einer Multisensorfusion verwendet, wobei die Objektposition hauptsächlich über Radar oder LiDAR [79, 80] gemessen wird.

Viele Arbeiten haben LiDAR-Sensoren für maritime Wahrnehmungszwecke integriert [81, 79]. Neuronale Netze, die LiDAR-Punktwolken verwenden, wie in GALILEOnautic 2 entwickelt, sind im Automobilssektor bekannt [82, 83, 84, 85, 86, 87]. Zum Trainieren und Evaluieren der neuronalen Netze stehen viele Benchmarks für urbane Szenarien zur Verfügung, wie der KITTI-Datensatz [88] und der DENSE-Datensatz [89]. Für die maritimen Szenarien gibt es jedoch zum heutigen Zeitpunkt keine vergleichbaren Implementierungen. Außerdem stehen der Öffentlichkeit keine markierten Datensätze für ein mögliches Training und eine Evaluierung der neuronalen Netze zur Verfügung.

Für die Multisensorfusion modularisieren viele Studien Ihre Sensordaten zur Umgebungswahrnehmung, indem sie ihre Messungen separat verarbeiten. Die Erkennungen der verschiedenen Sensoren werden dann bzgl. der Position der Objekte zugeordnet. Die Autoren in [80, 90] haben ein Sensorfusionssystem für die maritime Objektverfolgung unter Verwendung von LiDAR, Radar, Infrarot- und elektro-optischen Kameras vorgeschlagen. Die Messungen dieser Sensoren werden zunächst getrennt verarbeitet. Das heißt, die Punktwolken von Radar und LiDAR werden geclustert und die Bilder von Infrarot- und elektro-optischen Kameras werden merkmalsbasiert segmentiert, so dass vier Listen mit erkannten Objekten erstellt werden können.

5.3 Zusammenfassung

Die RWTH übernahm im Projekt GALILEOnautic 2 die Rolle der Projektkoordination, entwickelte ein eng-gekoppeltes maritimes integriertes Navigationsfilter und war für die Fusion von Umfelderkennungsinformationen, die LiDAR-gestützte Umfeldwahrnehmung und die Datensynchronisierung im Rahmen der Netzwerkregelung verantwortlich.

Kapitel 2 stellte die Entwicklung des hochgenauen Navigationsfilters vor. Es wurde ein eng-gekoppeltes Filter entwickelt, das auf inertialen Messungen (Beschleunigungen, Drehraten, etc.) basiert und durch GNSS-Signale und Geschwindigkeitsinformationen eines DVL, sowie durch einen elektronischen Kompass und Daten von externen Diensten (SAPOS und WSV) gestützt wird. Auf GNSS-Ebene wurden Pseudorange, Deltarange und Trägerphasen integriert. Zur Integration der Trägerphasen wurde die Schätzung der sogenannten Mehrdeutigkeiten eingeführt, so dass zentimetergenaue Positionierung ermöglicht wurde. Die Arbeiten zur DVL-Integration bezogen sich insbesondere auf die Integration des bereits auf dem Forschungsschiff DENEb verbauten DVL und den damit einhergehenden Integrationsherausforderungen. Zur Erhöhung der Konvergenzgeschwindigkeit bei der Orientierungsschätzung wurde zudem ein elektronischer Kompass integriert. In der experimentellen Untersuchung konnte gezeigt werden, dass hohe Positions- und Orien-

5 Projektergebnis

tierungsgenauigkeiten im Zentimeterbereich bzw. im Bereich von einem Grad erreicht wurden.

In Kapitel 3 wurde die Entwicklung eines Integritätssystems als Erweiterung des enggekoppelten Navigationsfilters vorgestellt. Hauptziel des Integritätssystems war die FDE von Multi-Pseudorange-Beobachtungen. Es wurde eine RSIVIA-basierte FDE-Strategie entwickelt, die RAIM mit Paritätsraum und die Integration von Geschwindigkeitsschätzung aus dem Navigationsfilter umfasst. In experimentellen Untersuchungen mit Daten aus dem GATE Berchtesgaden konnte gezeigt werden, dass dieser Ansatz sowohl die Genauigkeit als auch die Robustheit des Navigationsfilters deutlich verbessert und die Rechenlast im Vergleich zum traditionellen RSIVIA deutlich reduziert.

Ausgehend von der Annahme, dass die im Navigationsfilter verwendeten GNSS-Messungen damit integer sind, wurde für die FDE weiterer integrierter Sensoren ein Filterbank-Ansatz entwickelt. Eine Herausforderung hierbei war, dass die integrierten Sensoren unterschiedliche Abtastraten und Messverzögerungen haben. Die experimentelle Validierung zeigte, dass der entwickelte Ansatz in der Lage ist, verschiedene Arten von Sensorausfällen zu identifizieren und die Genauigkeit der Navigationslösung erheblich zu verbessern. Bei sicherheitskritischen Anwendungen erhielt das autonome System eine Warnung vor Sensorausfällen, ohne die Genauigkeit der Navigationslösung zu verlieren, was die Integrität des Navigationssystems erhöhte.

In Kapitel 4 wurden Umgebungswahrnehmung und Netzwerbeobachtung vernetzter Wasserfahrzeuge im Hafengebiet vorgestellt. Für die Umgebungswahrnehmung wurden Methoden für die Multi-Sensorfusion untersucht, welche die gleichzeitige Wahrnehmung der Umgebung und die Verfolgung der dynamischen Objekte ermöglichen. Außerdem wurden Methoden entwickelt, die die Umgebungswahrnehmung mehrerer vernetzter Teilnehmer in einer zentralen Recheneinheit integrierten. Um automatisiertes Manövrieren in einem dynamischen Verkehrsraum mit sich stetig verändernder Verkehrslage zu ermöglichen, wurde die Umgebungswahrnehmung anderer Verkehrsteilnehmer mittels LiDAR, AIS und Radar umgesetzt. Beim Anlegen wurde die statische Umgebung zunächst durch OGM und anschließend durch Polygone zusammengefasst. Die Verfolgung dynamischer Objekte wurde mittels Filterbank-Ansätzen realisiert. Bei Erprobungsfahrten mit dem Forschungsschiff DENEBA konnte gezeigt werden, dass die Umgebungswahrnehmung die Pier mit einer relativen Genauigkeit im erforderlichen Submeterbereich erkennen konnte, so dass zusammen mit der hochgenauen Sensorfusion und der Integritätsprüfung Anlegen realisiert werden konnte.

5.4 Eigene Veröffentlichungen im Projekt

- [S. Liu et al.] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Filter bank approach of tightly-coupled gnss and velocity sensor aided ins for navigation integrity enhancement in maritime applications,” in *In Proceedings of ECC 2021*, 2021.
- [J. Lin et al.] J. Lin, G. Campa, C.-E. Framing, J.-J. Gehrt, R. Zweigel, and D. Abel, “Adaptive shape fitting for lidar object detection and tracking in maritime applications,” *International Journal of Transport Development and Integration*, vol. 5, no. 2, pp. 105–117, jun 2021.
- [M. Nitsch et al.] M. Nitsch, J.-J. Gehrt, R. Zweigel, and D. Abel, “Embedded tightly coupled ins/dgps-dgal navigation filter on a mass-market single-board computer,” in *In Proceedings of ICL-GNSS 2021*, 2021.
- [J.-J. Gehrt et al.] J.-J. Gehrt, R. Zweigel, and D. Abel, “Fast GNSS ambiguity resolution under frequent and persistent outages of differential data for vessel navigation,” in *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*. Institute of Navigation, oct 2021, pp. 4116–4127.
- [J. Lin et al.] J. Lin, A. Puthiyavinayagam, S. Liu, M. Kurowski, J.-J. Gehrt, R. Zweigel, and D. Abel, “Real-time Multi-Object Tracking using Adaptive Filtering and Filter-Banks for Maritime Applications, ” *In Proceedings of ECC 2021*, 2021.
- [J.-J. Gehrt et al.] J.-J. Gehrt, W. Liu, D. Stenger, S. Liu, and D. Abel, “Environmentally Dependent Adaptive Parameterization of a GNSS-aided Tightly-Coupled Navigation Filter,” in *Proceedings of European Navigation Conference 2020*. IEEE, nov 2020, pp. 1–10.
- [J.-J. Gehrt et al.] J.-J. Gehrt, S. Liu, M. Nitsch, W. Bruhn, S. Rohde, D. Abel, and R. Zweigel, “Robust and Reliable Multi-Sensor Navigation Filter for Maritime Application,” in *Proceedings of IFAC World Congress 2020*, vol. 53, no. 2. Elsevier BV, 2020, pp. 14 482–14 487.
- [J. Lin et al.] J. Lin, L. Koch, M. Kurowski, J.-J. Gehrt, D. Abel, and R. Zweigel, “Environment perception and object tracking for autonomous vehicles in a harbor scenario,” in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, sep 2020.
- [S. Liu et al.] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Integrity of Dual-Constellation aided Navigation Filter in Safety-Critical Maritime Applications,” *European journal of navigation*, vol. 19, no. 3, pp. 10–17, 2019.

Literaturverzeichnis

- [S. Liu et al.] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Dual-Constellation Aided High Integrity and High Accuracy Navigation Filter for Maritime Applications,” in *Proceedings of the 2019 International Technical Meeting of The Institute of Navigation*. Institute of Navigation, feb 2019.
- [S. Liu et al.] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Identification of Multi-Faults in GNSS Signals using RSIVIA under Dual Constellation,” in *Summer Workshop on Interval Methods*, 2019.
- [J.-J. Gehrt et al.] J.-J. Gehrt, M. Nitsch, D. Abel, and R. Zweigel, “High Accuracy Navigation Filter with Dual Antenna enabling Double-Differencing with Dual-Constellation,” in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS +)*. Institute of Navigation, oct 2019, pp. 2186–2197.
- [J.-J. Gehrt et al.] J.-J. Gehrt, R. Zweigel, S. Roy, C. Büskens, M. Kurowski, T. Jeinsch, A. Schubert, M. Gluch, O. Simanski, E. Pairet-Garcia, W. Bruhn, F. Diegel, and D. Abel, “Optimal Maneuvering and Control of Cooperative Vessels within Harbors,” in *Journal of Physics: Conference Series, International Maritime and Port Technology and Development Conference and International Conference on Maritime Autonomous Surface Ships 13–14 November 2019, Trondheim, Norway*, vol. 1357. IOP Publishing, oct 2019, pp. 1–12.
- [J. Lin et al.] J. Lin, J.-J. Gehrt, R. Zweigel, and D. Abel, “Cooperative Localization of Networked Multi-agent System,” in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation*. Institute of Navigation, oct 2019.
- [J. Lin et al.] J. Lin, Z. Lou, R. Zweigel, and D. Abel, “Cooperative localization of a networked multi-vehicle system,” *IFAC-PapersOnLine*, vol. 52, no. 8, pp. 428–433, 2019.
- [S. Liu et al.] S. Liu, S. Roy, E. Pairet-Garcia, J.-J. Gehrt, and F. Siemer, “Case Study : Networked Control for Optimal Maneuvering of Autonomous Vessels,” *Laxenburg / IFAC*, vol. 52, no. 8, pp. 440–445, 2019.
- [R. Zweigel et al.] R. Zweigel, J. Gehrt, S. Liu, S. Roy, C. Buskens, M. Kurowski, T. Jeinsch, A. Schubert, M. Gluch, O. Simanski, E. Pairet-Garcia, F. Siemer, and D. Abel, “Optimal maneuvering and control of cooperative vehicles as case study for maritime applications within harbors,” in *2019 18th European Control Conference (ECC)*. IEEE, jun 2019, pp. 3022–3027.
- [J.-J. Gehrt et al.] J.-J. Gehrt, R. Zweigel, T. Konrad, and D. Abel, “DVL-aided Navigation Filter for Maritime Applications,” in *Proceedings of 11th IFAC CAMS 2018*,

vol. 51, no. 29. Elsevier BV, 2018, pp. 418 – 423, 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2018.

Literaturverzeichnis

- [1] United Nations Publications, “Review of maritime transport 2020,” tech. rep., United Nations, 2020. ISBN 978-92-1-112993-9.
- [2] European Global Navigation Satellite System Agency, *GSA GNSS Market Report 2019*. Luxembourg: Publications Office of the European Union, 2019.
- [3] J.-J. Gehrt, R. Zweigel, S. Roy, C. Büskens, M. Kurowski, T. Jeinsch, A. Schubert, M. Gluch, O. Simanski, E. Pairet-Garcia, W. Bruhn, F. Diegel, and D. Abel, “Optimal Maneuvering and Control of Cooperative Vessels within Harbors,” in *Journal of Physics: Conference Series, International Maritime and Port Technology and Development Conference and International Conference on Maritime Autonomous Surface Ships 13–14 November 2019, Trondheim, Norway*, vol. 1357, pp. 1–12, IOP Publishing, oct 2019.
- [4] S. International, “Sae-standard j3016,” tech. rep., SAE International, 2014.
- [5] R. Zweigel, J.-J. Gehrt, S. Liu, S. Roy, M. Kurowski, B. Kolewe, R. Damerius, T. Jeinsch, O. Simanski, M. Gluch, A. Schubert, H. Kupas, F. Siemer, S. Recher, and E. Pairet-Garcia, “Galileonautic - autonomes navigieren und optimiertes manövrieren von kooperierenden schiffen in sicherheitskritischen bereichen : Schlussbericht zum vorhaben, autoren des berichts: R. zweigel, j.-j. gehrt, s. liu, s. roy, m. kurowski, b. kolewe, r. damerius, t. jeinsch, o. simanski, m. gluch, a. schubert, h. kupas, f. siemer, s. recher, e. pairet-garcia ; durchführende institution(en): Rwth aachen university, universität bremen, universität rostock, hochschule wismar, scisys deutschland gmbh,” tech. rep., 2019.
- [6] International Maritime Organization, “SOLAS Chapter V: Safety of Navigation,” tech. rep., IMO, London, 2002.
- [7] J.-J. Gehrt, R. Zweigel, T. Konrad, and D. Abel, “DVL-aided Navigation Filter for Maritime Applications,” in *Proceedings of 11th IFAC CAMS 2018*, vol. 51, pp. 418 – 423, Elsevier BV, 2018. 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2018.

Literaturverzeichnis

- [8] P. D. Groves, *Principles of GNSS, inertial, and multisensor integrated navigation systems*. GNSS technology and application series, Boston: Artech house, second edition ed., 2013.
- [9] International Maritime Organization, “Imo resolution a.915(22) Revised maritime policy and requirements for a future GNSS,” tech. rep., IMO, London, 2002.
- [10] P. J. G. Teunissen, P. J. de Jonge, and C. C. J. M. Tiberius, “The LAMBDA-Method for fast GPS surveying,” in *International Symposium "GPS Technology Applications"*, Bucharest, Romania, 1995.
- [11] J.-J. Gehrt, *Robuste und hochgenaue Zustandsschätzung für die autonome Schifffahrt*. PhD thesis, Rheinisch-Westfälisch Technische Hochschule (RWTH) Aachen, 2021.
- [12] A. Lipp and X. Gu, “Cycle-slip detection and repair in integrated navigation systems,” in *Proceedings of 1994 IEEE Position, Location and Navigation Symposium - PLANS'94*, pp. 681–688, IEEE, 1994.
- [13] P. Henkel and N. Oku, “Cycle Slip Detection and Correction for Heading Determination with Low-Cost GPS/INS Receivers,” in *VIII. Hotine-Marussi Symposium on Mathematical Geodesy*, pp. 291–299, Springer International Publishing, 2015.
- [14] J.-J. Gehrt, S. Liu, M. Nitsch, W. Bruhn, S. Rohde, D. Abel, and R. Zweigel, “Robust and Reliable Multi-Sensor Navigation Filter for Maritime Application,” in *Proceedings of IFAC World Congress 2020*, vol. 53, pp. 14482–14487, Elsevier BV, 2020.
- [15] S. Verhagen and B. Li, *LAMBDA software package: Matlab implementation, Version 3.0*. Mathematical Geodesy and Positioning, Delft University of Technology, 2012.
- [16] B. Tapley, *Statistical orbit determination*. Amsterdam Boston: Elsevier Academic Press, 2004.
- [17] J.-J. Gehrt, R. Zweigel, T. Konrad, and D. Abel, “Dvl-aided navigation filter for maritime applications**this paper results from the joint research project galileonautic, which is supported by the german federal ministry for economic affairs and energy (grant 50na1510).,” *IFAC-PapersOnLine*, vol. 51, no. 29, pp. 418 – 423, 2018. 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2018.
- [18] J. Farrell, *Aided navigation : GPS with high rate sensors*, vol. 1. Edition. New York : McGraw-Hill, 2008. ISBN 9780071493291.
- [19] J.-J. Gehrt, M. Nitsch, D. Abel, and R. Zweigel, “High Accuracy Navigation Filter with Dual Antenna enabling Double-Differencing with Dual-Constellation,” in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The*

- Institute of Navigation (ION GNSS +)*, pp. 2186–2197, Institute of Navigation, oct 2019.
- [20] J. Wendel, *Integrierte Navigationssysteme: Sensordatenfusion, GPS und Inertiale Navigation*. Walter de Gruyter, 2011.
- [21] J.-J. Gehrt, R. Zweigel, T. Konrad, and D. Abel, “How the Parallel Use of GPS and Galileo Benefits Railway Applications,” *InsideGNSS*, pp. 40–45, 2018.
- [22] J.-J. Gehrt, T. Konrad, J. Lin, M. Breuer, D. Abel, and R. Zweigel, “High Precision Localization with Dual-Constellation for Railway Applications,” in *Proc. of the 2018 International Technical Meeting (ION ITM 2018)*, pp. 35–45, Institute of Navigation, feb 2018.
- [23] T. Konrad, J.-J. Gehrt, J. Lin, R. Zweigel, and D. Abel, “Advanced State Estimation for Navigation of Automated Vehicles,” *Annual Reviews in Control*, vol. 46, pp. 181–195, 2018.
- [24] P. Teunissen and D. Odijk, “Ambiguity dilution of precision: Definition, properties and application,” in *Proceedings of the 10th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1997)*, Kansas City, MO, pp. pp. 891–899., 1997.
- [25] A. Leick, L. Rapoport, and D. Tatarnikov, *GPS Satellite Surveying*. BLACKWELL, 2015. ISBN: 1118675576.
- [26] D. Odijk and P. J. G. Teunissen, “ADOP in closed form for a hierarchy of multi-frequency single-baseline GNSS models,” *Journal of Geodesy*, vol. 82, pp. 473–492, jan 2008.
- [27] M. Nitsch, J.-J. Gehrt, R. Zweigel, and D. Abel, “Embedded tightly coupled ins/dgps-dgal navigation filter on a mass-market single-board computer,” in *Proceedings of ICL-GNSS 2021*, pp. 1–7, 2021.
- [28] M. Breuer, T. Konrad, and D. Abel, “High Precision Localisation in Customised GNSS Receiver for Railway Applications,” in *Proc. of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2016)*, pp. 779–787, 2016.
- [29] G. Blewitt, *Geodetic Applications of GPS - Lecture Notes for Nordic Autumn School*, ch. Basics of the GPS Technique: Observation Equations, pp. 9–55. National Land Survey, Gävle, Sweden, 1997. ISSN 0280-5731, LMV-raport 1997:16.
- [30] S. Carcanague, O. Julien, W. Vigneau, C. Macabiau, and G. Hein, “Finding the right algorithm: Low-cost, single-frequency GPS/GLONASS RTK for road users,” *Inside GNSS*, vol. 8, no. 6, pp. 70–80, 2013.

- [31] J.-J. Gehrt, R. Zweigel, and D. Abel, “Fast GNSS ambiguity resolution under frequent and persistent outages of differential data for vessel navigation,” in *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*, pp. 4116–4127, Institute of Navigation, oct 2021.
- [32] Septentrio Satellite Navigation, *Firmware User Manual for SSR3 3.4.0*, 2014.
- [33] LORD Corporation MicroStrain® Sensing Systems, *LORD DATASHEET 3DM®-GX4-25 Attitude and Heading Reference System (AHR5)*. 459 Hurricane Lane , Suite 102 Williston, VT 05495 USA, 2018.
- [34] “116(73) Performance standards for marine transmitting heading devices (THDs),” tech. rep., International Maritime Organization (IMO), 2000.
- [35] U. I. Bhatti and W. Y. Ochieng, “Failure Modes and Models for Integrated GPS/INS Systems,” *Journal of Navigation*, vol. 60, no. 2, pp. 327–348, 2007.
- [36] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Filter bank approach of tightly-coupled gnss and velocity sensor aided ins for navigation integrity enhancement in maritime applications,” in *In Proceedings of ECC 2021*, 2021.
- [37] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Integrity of Dual-Constellation aided Navigation Filter in Safety-Critical Maritime Applications,” *European journal of navigation*, vol. 19, no. 3, pp. 10–17, 2019.
- [38] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Identification of multi-faults in GNSS signals using RSIVIA under dual constellation,” *Acta Cybernetica*, vol. 25, pp. 69–84, feb 2021.
- [39] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Dual-Constellation Aided High Integrity and High Accuracy Navigation Filter for Maritime Applications,” in *Proceedings of the 2019 International Technical Meeting of The Institute of Navigation*, Institute of Navigation, feb 2019.
- [40] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Dual-constellation aided high integrity and high accuracy navigation filter for maritime applications,” *Proceeding of the 2019 International Technical Meeting of The Institute of Navigation (ION ITM)*, pp. 762–774, 2019.
- [41] R. Brown, “A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods,” *Journal of The Insitute of Navigation*, vol. 39, no. 3, pp. 301–316, 1992.
- [42] B. Pervan, D. Lawrence, C. Cohen, and B. Parkinson, “Parity space methods for autonomous fault detection and exclusion using GPS carrier phase,” *Position Location and Navigation Symposium*, pp. 649–656, 1996.

- [43] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis*. Springer, 2001.
- [44] V. Drevelle and P. Bonnifait, “A set-membership approach for high integrity height-aided satellite positioning,” *GPS Solutions*, vol. 15, no. 4, pp. 357–368, 2011.
- [45] S. Rohou, L. Jaulin, L. Mihaylova, F. Le Bars, and S. Veres, “Guaranteed computation of robot trajectories,” *Robotics and Autonomous Systems*, vol. 93, pp. 76–84, 2017.
- [46] V. Drevelle and P. Bonnifait, “High integrity GNSS location zone characterization using interval analysis,” *Proceedings of the 22nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2009)*, pp. 2178–2187, 2009.
- [47] L. Jaulin, M. Kieffer, E. Walter, and D. Meizel, “Guaranteed robust nonlinear estimation with application to robot localization,” *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 32, no. 4, pp. 374–381, 2002.
- [48] H. Dbouk and S. Schön, “Comparison of different bounding methods for providing GPS integrity information,” in *2018 IEEE/ION Position Location and Navigation Symposium (PLANS)*, (Piscataway, NJ), pp. 355–366, IEEE, 2018.
- [49] N. Zhu, D. Betaille, J. Marais, and M. Berbineau, “Extended kalman filter (EKF) innovation-based integrity monitoring scheme with c / n0 weighting,” in *2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI)*, IEEE, sep 2018.
- [50] S. Liu, J.-J. Gehrt, D. Abel, and R. Zweigel, “Integrity of dual-constellation aided navigation filter in safety-critical maritime applications,” *European journal of navigation*, vol. 19, no. 3, pp. 10–17, 2019.
- [51] C. Tanil, S. Khanafseh, M. Joerger, and B. Pervan, “Sequential integrity monitoring for kalman filter innovations-based detectors,” in *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2018)*, pp. 2440–2455, Institute of Navigation, 2018.
- [52] R. Zweigel, J.-J. Gehrt, S. Liu, S. Roy, C. Bueskens, M. Kurowski, T. Jeinsch, A. Schubert, M. Gluch, O. Simanski, E. Pairet-Garcia, F. Siemer, and D. Abel, “Optimal Maneuvering and Control of Cooperative Vehicles as Case Study for Maritime Applications within Harbors,” in *18th European Control Conference (ECC)*, (Naples, Italy), pp. 3022–3027, 2019.

- [53] J. Lin, A. Puthiyavinayagam, S. Liu, M. Kurowski, J.-J. Gehrt, R. Zweigel, and D. Abel, “Real-time Multi-Object Tracking using Adaptive Filtering and FilterBanks for Maritime Applications,” in *2021 European Control Conference (ECC)*, pp. 1–6, 2021.
- [54] B. J. Tetreault, “Use of the automatic identification system (ais) for maritime domain awareness (mda),” in *Proceedings of OCEANS 2005 MTS/IEEE*, pp. 1590–1594 Vol. 2, 2005.
- [55] MarineTraffic, “Global Ship Tracking Intelligence | AIS Marine Traffic,” Feb. 2019. URL: <https://www.marinetraffic.com>. Last accessed: 22.10.2020.
- [56] E. Tu, G. Zhang, L. Rachmawati, E. Rajabally, and G. Huang, “Exploiting ais data for intelligent maritime navigation: A comprehensive survey from data to methodology,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1559–1582, 2018.
- [57] A. Harati-Mokhtari, A. Wall, P. Brooks, and J. Wang, “Automatic identification system (AIS): Data reliability and human error implications,” *The Journal of Navigation*, vol. 60, no. 3, pp. 373–389, 2007.
- [58] L. P. Perera, P. Oliveira, and C. Guedes Soares, “Maritime traffic monitoring based on vessel detection, tracking, state estimation, and trajectory prediction,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1188–1200, 2012.
- [59] D. Gaglione, P. Braca, and G. Soldi, “Belief propagation based ais/radar data fusion for multi - target tracking,” in *2018 21st International Conference on Information Fusion (FUSION)*, pp. 2143–2150, July 2018.
- [60] B. H. Eriksen, E. F. Wilthil, A. L. Flåten, E. F. Brekke, and M. Breivik, “Radar-based maritime collision avoidance using dynamic window,” in *2018 IEEE Aerospace Conference*, pp. 1–9, 2018.
- [61] J. Han, J. Kim, and N. Son, “Persistent automatic tracking of multiple surface vessels by fusing radar and lidar,” in *OCEANS 2017 - Aberdeen*, pp. 1–5, 2017.
- [62] L. Stanislas and M. Dunbabin, “Multimodal sensor fusion for robust obstacle detection and classification in the maritime RobotX challenge,” *IEEE Journal of Oceanic Engineering*, vol. 44, no. 2, pp. 343–351, 2019.
- [63] J. Lin, P. Diekmann, C.-E. Framing, R. Zweigel, and D. Abel, “Maritime environment perception based on deep learning,” *submitted in IEEE Transactions on Intelligent Transportation Systems*, 2021.

- [64] R. B. Rusu and S. Cousins, “3D is here: Point Cloud Library (PCL),” in *IEEE International Conference on Robotics and Automation (ICRA)*, (Shanghai, China), May 9-13 2011.
- [65] R. Pugliese, T. Konrad, R. Zweigel, and D. Abel, “Object Detection and Potential Field-based Trajectory Planning on LiDAR 3D-Point Clouds for Autonomous Vehicles in Uncertain Environments,” in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019)*, (Miami, USA), pp. 2622 – 2636, Insitut of Navigation, Sept. 2019.
- [66] J. L. Bentley, “Multidimensional binary search trees used for associative searching,” *Commun. ACM*, vol. 18, p. 509–517, Sept. 1975.
- [67] J. Lin, G. Campa, C.-E. Framing, J.-J. Gehrt, R. Zweigel, and D. Abel, “Adaptive shape fitting for lidar object detection and tracking in maritime applications,” *International Journal of Transport Development and Integration*, vol. 5, pp. 105–117, jun 2021.
- [68] J. Lin, J.-J. Gehrt, R. Zweigel, and D. Abel, “Cooperative Localization of Networked Multi-agent System,” in *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation*, Institute of Navigation, oct 2019.
- [69] B. Jin, J. Guo, D. He, and W. Guo, “Adaptive kalman filtering based on optimal autoregressive predictive model,” *GPS Solutions*, vol. 21, p. 307–317, Apr. 2017.
- [70] J. Lin, L. Koch, M. Kurowski, J.-J. Gehrt, D. Abel, and R. Zweigel, “Environment perception and object tracking for autonomous vehicles in a harbor scenario,” in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1513–1518, IEEE, sep 2020.
- [71] A. Elfes, “Using occupancy grids for mobile robot perception and navigation,” *Computer*, vol. 22, pp. 46–57, jun 1989.
- [72] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. The MIT Press, 2005.
- [73] M. A. Fischler and R. C. Bolles, “Random sample consensus,” *Communications of the ACM*, vol. 24, pp. 381–395, jun 1981.
- [74] O. contributors, “Planet dump retrieved from <https://planet.osm.org>.” "<https://www.openstreetmap.org>", 2017.
- [75] D. Medina, A. Heßelbarth, R. Büscher, R. Ziebold, and J. García, “On the Kalman Filtering Formulation for RTK Joint Positioning and Attitude Quaternion Determination,” in *Proc. of the 2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pp. 597–604, IEEE, apr 2018.

- [76] D. K. Prasad, D. Rajan, L. Rachmawati, E. Rajabally, and C. Quek, "Video processing from electro-optical sensors for object detection and tracking in a maritime environment: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 8, pp. 1993–2016, 2017.
- [77] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2016.
- [78] B. Liu, S. Z. Wang, Z.X. Xie, J. Zhao, and M. Li, "Ship Recognition and Tracking System for Intelligent Ship Based on Deep Learning Framework," *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation*, vol. 13, no. 4, pp. 699–705, 2019.
- [79] J. Han, Y. Cho, J. Kim, J. Kim, N. sun Son, and S. Y. Kim, "Autonomous collision detection and avoidance for ARAGON USV: Development and field tests," *Journal of Field Robotics*, vol. 37, no. 6, pp. 987–1002, 2020.
- [80] M. Haghbayan, F. Farahnakian, J. Poikonen, M. Laurinen, P. Nevalainen, J. Plosila, and J. Heikkonen, "An efficient multi-sensor fusion approach for object detection in maritime environments," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2163–2170, 2018.
- [81] L. Elkins, D. Sellers, and W. R. Monach, "The autonomous maritime navigation (amn) project: Field tests, autonomous and cooperative behaviors, data fusion, sensors, and vehicles," *Journal of Field Robotics*, vol. 27, no. 6, pp. 790–818, 2010.
- [82] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "PointPillars: Fast Encoders for Object Detection from Point Clouds," in *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 12697-12705*, 2019.
- [83] C. He, H. Zeng, J. Huang, X.-S. Hua, and L. Zhang, "Structure Aware Single-stage 3D Object Detection from Point Cloud," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [84] Y. Yan, Y. Mao, and B. Li, "SECOND: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, pp. 1–17, 2018.
- [85] M. Imad, O. Doukhi, and D.-J. Lee, "Transfer learning based semantic segmentation for 3d object detection from point cloud," *Sensors*, vol. 21, no. 12, p. 3964, 2021.
- [86] S. Shi, X. Wang, and H. Li, "Pointrenn: 3d object proposal generation and detection from point cloud," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

- [87] Y. Chen, S. Liu, X. Shen, and J. Jia, “Fast point r-cnn,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, 2019.
- [88] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: the KITTI dataset,” *The International Journal of Robotics Research* 32(11):1231-1237, 2013.
- [89] T. Gruber, F. Julca-Aguilar, M. Bijelic, and F. Heide, “Gated2Depth: Real-Time Dense Lidar From Gated Images,” in *The IEEE International Conference on Computer Vision (ICCV)*, 2019.
- [90] O. K. Helgesen, E. F. Brekke, H. H. Helgesen, and O. Engelhardtson, “Sensor Combinations in Heterogeneous Multi-sensor Fusion for Maritime Target Tracking,” in *2019 22th International Conference on Information Fusion (FUSION)*, pp. 1–9, 2019.

Berichtsblatt

1. ISBN oder ISSN	2. Berichtsart	
3a. Titel des Berichts Schlussbericht zum Vorhaben GALILEOnautic 2		
3b. Titel der Publikation Schlussbericht zum Vorhaben GALILEOnautic 2		
4a. Autoren des Berichts (Name, Vorname(n)) Gehrt, Jan-Jöran; Liu, Shuchen; Lin, Jiaying; Nitsch, Maximilian; Stenger, David; Kosch, Martin; Zweigel, René		5. Abschlussdatum des Vorhabens 30.09.2021
4b. Autoren der Publikation (Name, Vorname(n)) Gehrt, Jan-Jöran; Liu, Shuchen; Lin, Jiaying; Nitsch, Maximilian; Stenger, David; Kosch, Martin; Zweigel, René		6. Veröffentlichungsdatum 19. Januar 2022
		7. Form der Publikation Digital
8. Durchführende Institution(en) (Name, Adresse) Institut für Regelungstechnik RWTH Aachen University Campus-Boulevard 30 52074 Aachen Germany		9. Ber.Nr. Durchführende Institution
		10. Förderkennzeichen ¹⁾ 50 NA 1808
		11a. Seitenzahl Bericht 147
		11b. Seitenzahl Publikation 147
13. Fördernde Institution (Name, Adresse) Deutsche Raumfahrtagentur im Deutschen Zentrum für Luft- und Raumfahrt e.V. Königswinterer Str. 522-524 53227 Bonn		12. Literaturangaben 90
		14. Tabellen 5
		15. Abbildungen 69
16. Zusätzliche Angaben		
17. Vorgelegt bei (Titel, Ort, Datum) Technische Informationsbibliothek (TIB) - Deutsche Forschungsberichte, Welfengarten 1B, 30167 Hannover 19. Januar 2022		
18. Kurzfassung Die zunehmende Verkehrsdichte auf Schifffahrtswegen und die damit einhergehenden hohen Anforderungen an eine sichere Schifffahrt erfordern in Zukunft einen Ausbau der Automatisierungsfunktionen an Bord. Abseits der großen Traversen ergeben sich - insbesondere in engen Fahrwassern und Häfen, besondere Anforderungen an eine automatisierte Schiffsführung. Die Ergebnisse eines Navigationssystems zur Bestimmung der aktuellen Position, Geschwindigkeit und Orientierung spielen dabei eine maßgebliche Rolle. Gerade im Hinblick auf sicheres automatisiertes Manövrieren ist eine hochverfügbare, hochgenaue und vertrauenswürdige/integre Orts-, Geschwindigkeits- und Lagebestimmung notwendig. Im Förderprojekt GALILEOnautic 2 - gefördert vom Bundesministerium für Wirtschaft und Energie mit dem Förderkennzeichen 50NA1808 - wurde daher ein System zur Navigation, Umwelterkennung und Schiffsregelung entwickelt. Die Arbeiten standen im Kontext automatisierter Manöver in Häfen. Zur Demonstration der Projektergebnisse wurde das Forschungsschiff DENEb des Bundesamtes für Seeschifffahrt und Hydrographie derart technisch erweitert, dass eine automatisierte Steuerung möglich war - die DENEb war damit zentraler Versuchsträger im Projekt. Die RWTH Aachen University entwickelte im Projekt Trägerphasen-gestützte Navigationsalgorithmen, die zentimetergenaue Lokalisierung im Hafen ermöglichten. Darüber hinaus wurden Filterbank-basierte, Sensor-übergreifende Integritätsmethoden entwickelt, die die verwendeten Sensordaten auf Ihre Integrität hin überwachten und eine Aussage zur aktuellen Zuverlässigkeit des Navigationssystems geben konnten. Die RWTH zeigte zudem, dass mittels Datenfusion von maritimer Umfeldsensoren und AIS-Daten mit LiDAR- und Radarsensoren ein dynamisches Verkehrslagebild inklusive Infrastruktur erstellt werden konnte, das als Basis für sichere, vernetzte und kooperative Navigation diene.		
19. Schlagwörter Autonomes Fahren, autonome Schifffahrt, automatisches Anlegen, Navigation, Integrität, Umgebungswahrnehmung, GNSS		
20. Verlag		21. Preis

¹⁾ Hinweis für Bearbeiter/in: Auf das Förderkennzeichen des BMWi soll auch in der Veröffentlichung hingewiesen werden.

--	--

Document Control Sheet

1. ISBN or ISSN	2. Type of Report
3a. Report Title Final report on the GALILEOnautic 2 project	
3b. Title of Publication Final report on the GALILEOnautic 2 project	
4a. Author(s) of the Report (Family Name, First Name(s)) Gehrt, Jan-Jöran; Liu, Shuchen; Lin, Jiaying; Nitsch, Maximilian; Stenger, David; Kosch, Martin; Zweigel, René	5. End of Project 30th September, 2021
4b. Author(s) of the Publication (Family Name, First Name(s)) Gehrt, Jan-Jöran; Liu, Shuchen; Lin, Jiaying; Nitsch, Maximilian; Stenger, David; Kosch, Martin; Zweigel, René	6. Publication Date January 19, 2022
8. Performing Organization(s) (Name, Address) Institute of Automatic Control RWTH Aachen University Campus-Boulevard 30 52074 Aachen Germany	9. Originator's Report No.
13. Sponsoring Agency (Name, Address) Deutsche Raumfahrtagentur im Deutschen Zentrum für Luft- und Raumfahrt e.V. Königswinterer Str. 522-524 53227 Bonn Germany	10. Reference No. 50 NA 1808
	11a. No. of Pages Report 147
	11b. No. of Pages Publication 147
	12. No. of References 90
	14. No. of Tables 5
	15. No. of Figures 69
16. Supplementary Notes	
17. Presented at (Title, Place, Date) Technische Informationsbibliothek (TIB) - Deutsche Forschungsberichte, Welfengarten 1B, 30167 Hannover January 19, 2022	
18. Abstract The increasing traffic density on shipping routes and the associated high demands on safe shipping will require an expansion of on-board automation functions in the future. Away from the major traverses - especially in narrow fairways and harbors - there are special requirements for automated ship navigation. The results of a navigation system for determining the current position, speed and orientation play a decisive role here. Especially with regard to safe automated maneuvering, a highly available, highly accurate and trustworthy/integre location, speed and position determination is necessary. In the GALILEOnautic 2 project - funded by the German Federal Ministry for Economic Affairs and Energy with the grant number 50NA1808 - a system for navigation, environmental recognition and ship control was therefore developed. The work was carried out in the context of automated maneuvers in ports. To demonstrate the project results, the research vessel DENEb of the Federal Maritime and Hydrographic Agency was technically upgraded in such a way that automated control was possible - the DENEb was thus the central test vehicle in the project. In the project, RWTH Aachen University developed carrier-phase-based navigation algorithms that enabled localization in the harbor with centimeter accuracy. In addition, filter bank-based, cross-sensor integrity methods were developed that monitored the integrity of the sensor data used and were able to provide a statement on the current reliability of the navigation system. RWTH also showed that by means of data fusion of maritime environment sensor technology and AIS data with LiDAR and radar sensor technology, a dynamic traffic situation snap-shot including infrastructure information could be created, which served as a basis for safe, networked and cooperative navigation.	
19. Keywords autonomous driving, autonomous shipping, automated docking, navigation, integrity, environmental perception, GNSS	

20. Publisher	21. Price
---------------	-----------