

PAPER • OPEN ACCESS

## Automated Maneuvering of Networked Vessels in Confined Waters

To cite this article: Nick Eisenblätter *et al* 2025 *J. Phys.: Conf. Ser.* **3123** 012020

View the [article online](#) for updates and enhancements.

### You may also like

- [Mud crabs \(\*Scylla olivacea\*\) fattening in recirculating aquaculture system \(RAS\) using vertical gallons crab house with different feed types](#)  
Zainal Usman, Muhammad Hery Riyadi Alauddin, Anton et al.
- [Effect of Locally Formulated Urea Molasses Multinutrient Block Supplementation on Feed Intake, Growth Performance and Meat Quality of Beef Cattle](#)  
N. S. A. Hanafiah, W. M. S. Wan Ab Karim, W. M. N. A. Wan Zuhaimi et al.
- [Data assimilation between Monte Carlo simulation and experimental results of crystal growth island density in REBCO thin films during the initial growth stages](#)  
Eijiro Okumura, Haruto Uchida, Takumi Takamura et al.



The Electrochemical Society  
Advancing solid state & electrochemical science & technology







**249th  
ECS Meeting**  
May 24-28, 2026  
Seattle, WA, US  
*Washington State  
Convention Center*

# Spotlight Your Science

***Submission deadline:  
December 5, 2025***

**SUBMIT YOUR ABSTRACT**

# Automated Maneuvering of Networked Vessels in Confined Waters

Nick Eisenblätter<sup>1\*</sup>, Tim Rehbronn<sup>2\*</sup>, Martin Kurowski<sup>1</sup>,  
David Benz<sup>2</sup>, Heike Vallery<sup>2</sup>, and Torsten Jeinsch<sup>1</sup>

<sup>1</sup>Institute of Automation, University of Rostock, Germany

<sup>2</sup>Institute of Automatic Control, RWTH Aachen University, Germany

\*These authors contributed equally

E-mail: nick.eisenblaetter@uni-rostock.de, t.rehbronn@irt.rwth-aachen.de

**Abstract.** Maritime Autonomous Surface Ships technology has made substantial progress in recent years, yet challenges remain in safety-critical scenarios such as maneuvering through congested, constrained waterways like ports. This contribution presents an architecture for automated, networked, and cooperative maneuvering of multiple vessels with two core modules: cooperative perception and centralized collision avoidance planning. The environmental perception module combines LiDAR, radar, and GNSS-based navigation to enable accurate self-localization and robust object detection. Its cooperative multi-object tracking fuses onboard detections with AIS data and observations from other vessels. The module distinguishes between static and dynamic obstacles and feeds the shapes as convex polygons to the centralized collision avoidance planning module. The planning module uses optimal control to compute collision-free trajectories under geometric constraints. Cooperative maneuver planning addresses complex right-of-way situations by prioritizing and decomposing maneuvers into sequential sub-scenarios, with local control systems guiding individual vessels. We evaluate the approach experimentally in a maneuvering scenario in the port of Rostock involving two on-site and one simulated vessel. One of them is the 52 m-long automated research vessel Deneb equipped with advanced sensors, which exchanges state and object information via conventional communication links. Results demonstrate the reliable localization of static and dynamic objects as well as collision-free automated maneuver execution. This cooperative behavior is an important step towards safe mixed traffic, including non-networked vessels.

## 1 Introduction

### 1.1 Motivation

Maritime transport has the largest share (67.4%) of freight transport within the European Union [1], and the global trade volume is projected to increase by 2.4% in 2025 [2]. As a result, congested and complex environments such as ports and inland waterways are facing growing operational demands. At the same time, the sector faces a shortage of qualified personnel, particularly for challenging navigation tasks in port areas and inland waterways [3]. This context motivates the development of advanced cooperative and automated navigation systems to enhance operational efficiency and safety in these critical areas.

Maritime Autonomous Surface Ships (MASS) present a promising technical solution, where automated vessels navigate with reduced human intervention. However, achieving safe and reliable autonomous operation in confined and safety-critical environments such as ports and inland waterways is a key challenge. Multiple vessels may navigate in close proximity, requiring robust strategies for maneuver coordination.



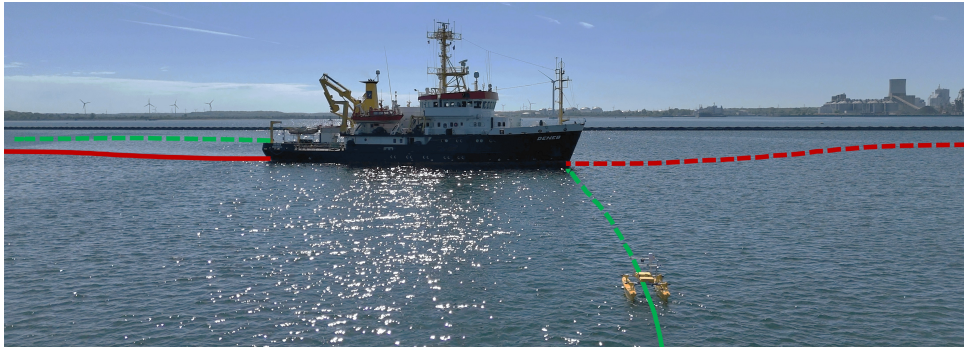


Figure 1: Crossing situation with limited maneuvering space in the port of Rostock between research vessel Deneb (background) and USV Messin (foreground)

Networked autonomous vessels can be part of a cooperative solution by sharing data such as routes, states, as well as detected and tracked objects. Traffic control centers that generate trajectories for feasible encounters based on shared information reduce time for decision making and maneuvering space compared to decentralized scenarios. Such a cooperative strategy is crucial when the maneuvering space is so limited that the basic rules of the International Regulations for Preventing Collisions at Sea (COLREGs) [4] still apply, but the aspects of good seamanship of a navigator strongly impact the situation, e.g. concerning the individual definition of safety distances to other vessels. Figure 1 shows an example in the port of Rostock for a crossing situation between research vessel Deneb (background) and unmanned surface vehicle (USV) Messin in which the entire maneuvering space is less than  $300 \times 300$  meters.

## 1.2 Related Work

Extensive research has been conducted on autonomous shipping [5], where the major activities focus on environmental perception and motion planning, primarily for collision avoidance.

Concerning environmental perception, most recent approaches for MASS rely heavily on radar or camera systems. For instance, [6] employs a YOLOv8 [7] image-only object detection model for vessel detection. However, relying solely on a camera provides useful semantic information but lacks accurate depth information. Authors of [8] improve this approach by fusing a camera with a radar sensor using a multi-stage object detection module, improving detection range and performance. However, both approaches focus on object detection without considering tracking, static mapping, or localization. To further improve detection and enable tracking, the authors of [9] fuse camera, LiDAR, and radar sensors. They apply a YOLOv4 model on image data to detect objects. However, for radar and LiDAR data, only clustering is applied to remove static land mass and find clusters for object candidates. The candidates of LiDAR, radar, and camera are fused on the object level using a Kalman filter. While this architecture utilizes multiple sensors and extends detection by tracking, it lacks static mapping or absolute localization.

In the context of collision avoidance in maritime traffic, mainly methods relating to the individual ship are investigated, without knowing the intentions of other traffic participants [10]. In open seas, this approach is appropriate since ships maintain large distances between each other and sufficient time is available for vehicle prediction and decision-making, e.g. [11] and many others. Collaborative collision avoidance of networked vehicles is even less considered, although it is crucial for safety-critical areas. In this context, a distinction can be made between decentralized and centralized approaches, where [10] provides a comprehensive overview of different studies focusing more on narrow waterways than on safety-critical port areas.

On the traffic management level, extensive developments were pushed forward as part of the STM - Sea Traffic Management initiatives with the aim to allow personnel on-board and on shore to make decisions based on real-time information. The focus was on exchange of routes, tactical traffic management, decision support for navigators and compliance with standards of the International Maritime Organization (IMO), [12], which provides an important basis also for the own research.

Further, it is essential to consider the Operational Design Domains (ODDs) when assessing the applicability and safety of autonomous maritime operations. The ODD defines the specific conditions under which the MASS is intended to operate safely, including environmental, geographical, and operational constraints. These considerations are fundamental to ensure reliable system performance and risk mitigation in real-world scenarios, [13].

### 1.3 Own Preliminary Work

In the GALILEOnautic project series, researchers investigate cooperative maneuvering strategies, especially in confined areas to enable autonomous navigation for multiple vehicles based on the well-known Guidance, Navigation, and Control (GNC) structure established on single vessels, e.g., described in [14]. [15] applied the structure for cooperative multi-vehicle scenarios. A tightly-coupled Extended Kalman Filter (EKF) achieved accurate pose estimation, and radar and LiDAR systems provided environmental perception [16]. Further, trajectory planning and control algorithms have been designed to dynamically optimize trajectories, avoid collisions, and track desired paths, with non-networked vessels taken into account using the Automatic Identification System (AIS). The algorithms were evaluated in real-world scenarios, without considering collision regulations in detail and the maneuvering space limitations related to vehicle size [15]. Furthermore, the perception module had a maximum range of 40 m, limiting adaptability for larger vessels requiring larger maneuvering space. [17] Within project GALILEOnautic2, the strategies were applied to the research vessel Deneb of the German Federal Maritime and Hydrographic Agency (BSH), where the focus was on GNC of a single vessel in maneuvering situations and automated docking system procedures [18, 19]. The navigation filter was updated to utilize carrier phase measurements of GPS and Galileo [20], and the navigation sensors and the filter were monitored with integrity modules [21]. As a further improvement, an enhanced perception system was implemented using deep learning-based object detection, static mapping, and Kalman filter banks to track multiple objects by combining LiDAR detections with AIS signals [22, 23]. The GNC system was adapted to partly operate locally on the vessels while being simultaneously part of a networked agent [24]. A maneuver assistance system (MAS) was realized to visualize the automation process for vessel operators [25].

Despite these advancements, the perception system was restricted by its close range, limited field of view, and low-resolution LiDAR. The overall system was designed and tested for a single-vessel docking maneuver and not for cooperative multi-vessel maneuvering.

### 1.4 Contribution

Building on the GALILEOnautic project series, this contribution presents the system integration and validation of a central traffic control architecture for collaborative automated maneuvering, demonstrated on three vessels. The system addresses key limitations of prior work, particularly sensing range and coverage, by integrating multiple medium-range LiDARs with a long-range radar. The navigation filter is based on a previously developed loosely coupled error-state extended Kalman Filter (ES-EKF) [26], and the tracking algorithm is simplified from an Extended Kalman Filter bank [23] to a single linear Kalman Filter for computational efficiency. In addition, a newly developed 2D grid mapping module, an enhanced object detection model with a retrained network, and improved sensor fusion were implemented to support robust perception. The primary contribution lies in the successful integration of these components into a unified perception pipeline and the validation of its performance in realistic multi-vessel scenarios.

The proposed collision avoidance planner works centrally and is applied to ships of a relevant size. It generates optimal collision-free trajectories based on potentially non-collision-free initial trajectories, taking into account map information, fused objects from the environmental perception, and generalized vehicle dynamics. COLREGs are taken into account and adapted for use in safety-critical areas like ports.

The system is validated in a mixed experimental setup involving two real and one simulated networked vessels, performing coordinated collision avoidance maneuvers in the port of Rostock.

The primary contribution of this work lies in the successful integration of these components - perception pipeline, collision avoidance planner, and multi-vessel coordination — and the validation of their combined performance in realistic operational scenarios.

In the remainder of this contribution, the general system architecture for automated central traffic management is introduced in Section 2. On that basis, Section 3 describes the modules developed for perception at the vehicle level as well as in the central context, and Section 4 presents the methods applied with the centrally operating collision avoidance planner. Further, Section 5 depicts the experimental setup, comprising three vessels, followed by an outline of the corresponding evaluation scenario in the experimental protocol and the description of the data processing and performance indicators. The contribution finishes with the discussion of the experimental results in Section 6 and a conclusion in Section 7.

## 2 System Architecture for Automated Central Traffic Management

When approaching congested waterways and ports, a traffic control center (e.g., the German Vessel Traffic Service, VTS) usually takes over the global management of traffic by planning arrival times and encounter situations in advance and effectively managing them. This procedure is essential for accident prevention



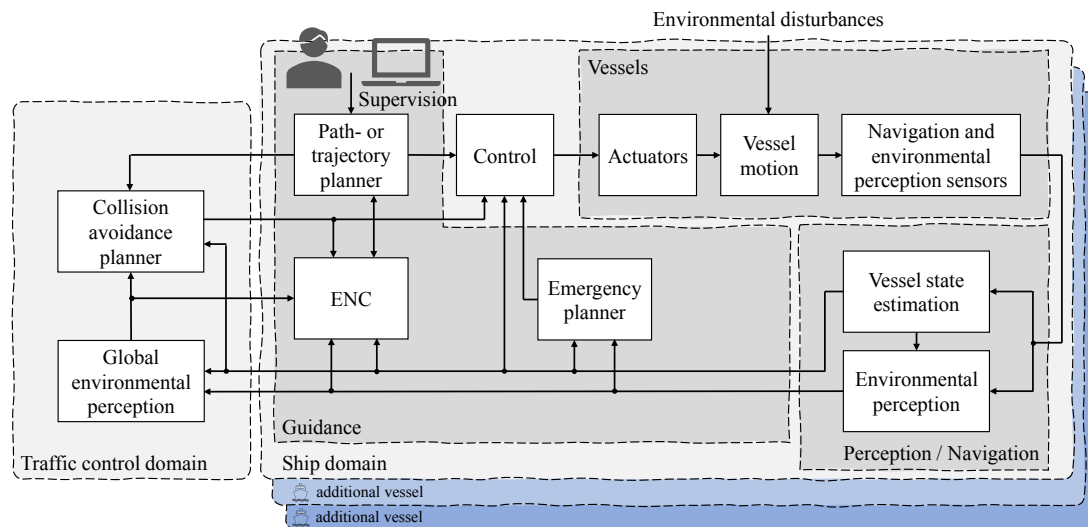


Figure 2: System Architecture for Automated Central Traffic Management

in safety-critical areas when the maneuvering space available to the individual ship is extremely limited. In this established procedure, the coordination of several ships is performed in a superior *traffic control domain*, which then impacts the operation of the vessel through the nautical officers of the individual ships. With regard to MASS operations, a more differentiated view on the traffic management structures is necessary, for example in the case of monitoring and potential remote control by a Remote Operations or Shore Control Center which finally gives instructions or commands to a MASS [27].

The approach presented in this paper and illustrated in Figure 2 is based on that general structure and is intended to show to what extent the centralized traffic management can be used for automated coordination of MASS in confined waters.

A distinction is made between the *ship domain* and the traffic management domain, where the ship domain reflects the well-known *guidance*, *navigation* and *control* structure for a single manned or unmanned vehicle, given in e.g. [14] and [28]. While the traffic control domain is centrally responsible for a specific area, there are potentially several vehicles to be considered in the respective ship domain, which is indicated by the additional blue ship domain layers in the figure.

The central point of the system architecture is the vessels, which are equipped with *actuators* and sensors, where, in contrast to human-centered maritime shipping, future MASS will have to use high-performance systems to be operated safely. The *vessel motion* is additionally subject to *environmental disturbances*. Especially concerning the sensors, a comprehensive set of *navigation and environmental perception sensors* is required for operation with a high automation level, particularly when operating in restricted waters and ports, [29].

The *perception/navigation* module is fed by the signals from the available sensors and can be divided into two main submodules. First, the *vehicle state estimation* submodule determines the optimal state vector by fusing inertial measurements (accelerations and angular rates) from an IMU with GNSS-based position and velocity measurements with RTK corrections in a classical multi-sensor Kalman Filter structure [26]. Further, a dynamic model-based filtering stage can be added subsequently to realize state filtering on the operational level [30]. In this way, the submodule feeds the control and guidance systems of the respective ship with filtered motion data from the real process subject to disturbances. Hence, the estimation must be robust to sensor uncertainties and implemented free of jumps to allow the vehicle's motion states to be used in the feedback control. Second, in case of MASS, aspects of *environmental perception* play a crucial role, applying dedicated sensors such as LiDAR, radar, and other optical systems, as well as the automatic identification system (AIS). The submodule detects static structures and dynamical objects, including their classification as well as possible trajectory predictions of the dynamical objects, which are used in the guidance of the respective ship, e.g., as overlay in an electronic navigational chart (*ENC*). Further, the vehicle state and the classified objects are transmitted traffic control domain. In this paper, we focus on the *vessel state estimation* and *environmental perception* submodules as the two core parts of navigation.

The *guidance* module is used to generate the reference values for the control of the vehicle. In human-

centered shipping, a nautical officer uses a *path or trajectory planner* to generate a berth-to-berth route using information from an ENC, which serves as an initial plan for the next voyage. However, the route cannot be considered collision-free as it only incorporates information at the current time of planning and is not necessarily used on an ad-hoc basis. For unmanned systems or MASS, an optimization takes over this task, which also uses the map basis of an ENC as well as the additional information of the local environmental perception for trajectory generation. In addition, knowledge about the dynamic behavior of the vehicle should be taken into account to generate feasible trajectories [31]. The resulting trajectories also have to be considered as non-collision-free in a global context. Moreover, an *emergency planner* is required for scenarios in which either the online trajectory planner does not find a solution in the required time or it is no longer possible to follow the trajectory due to system faults. This module reacts based on information from the navigation module and is intended to bring the automated vehicle into a safe state in case of an emergency.

Finally, the *control* system summarizes the modules for automatic motion control, with the ability to follow a given path or trajectory. Especially in maneuvering situations [32], model-based approaches for control system design can be used to achieve the required performance, resulting in a multi-variable control system structure including an actuator allocation distributing the control forces and torques to the equipped and operational propulsion and steering gears, [33].

The concept is based on the fact that in the networked scenario under consideration, basic information regarding the vehicle state, the detected local objects, and the initial trajectory of each networked vehicle is transmitted to the traffic control domain. Widely available communication systems, such as mobile radio (e.g., LTE), are used for this purpose. The *global environmental perception* merges the local object detections of the respective vessels to obtain a comprehensive situational awareness of the entire confined area under observation. This information is then used together with the initial trajectories in the *collision avoidance planner*, where finally collision-free trajectories are generated for the networked vessels that conform to the COLREGs and especially their pendants for confined waters and ports.

### 3 Perception Module

#### 3.1 Perception Components and Structure

The perception module uses a variety of sensors and processing strategies to estimate the vessel's pose as well as to detect and track objects in its environment. Thus, it is structured into two core components: vessel state estimation (self-perception) and environmental perception. The overall structure is depicted in Figure 3. First, we transform each local sensor frame into a common NED reference frame using the dynamic transformation provided by the navigation filter. A 3D object detection is applied to the point cloud, estimating nearby vessels' pose, dimension, and class. The detected objects are tracked and fused with AIS data. To account for static objects, we project the 3D LiDAR point cloud into a 2D bird's-eye view and fuse it with the long-range 2D radar point cloud. Additionally, we utilize the object bounding boxes to remove dynamic points, yielding a static point cloud. Lastly, the static point cloud is used for probabilistic occupancy grid mapping. The ego-vessel pose, the detected objects, and static objects are shared via a network interface.

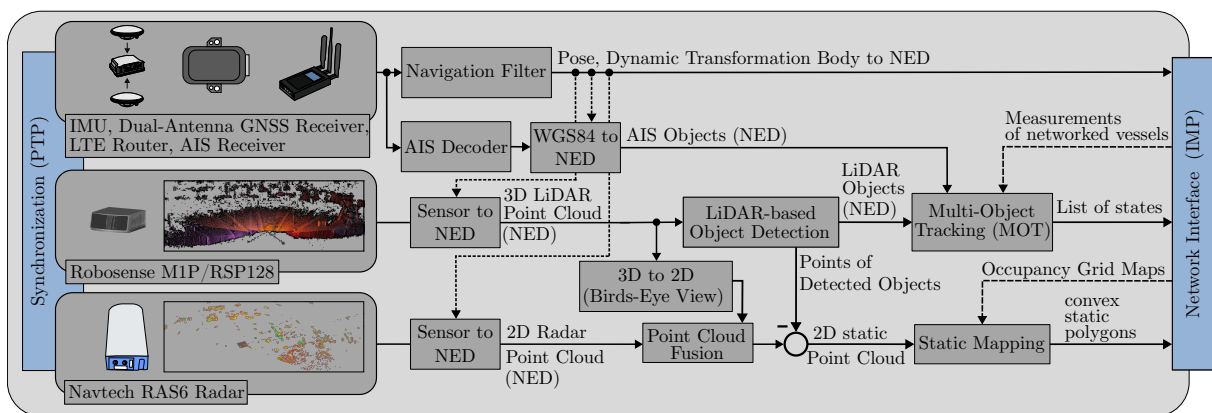


Figure 3: Architecture of the perception module. Sensor data is transformed into a common NED frame. Detected objects are tracked and fused with AIS. A static point cloud is derived by removing dynamic objects and used for occupancy grid mapping. The module outputs are shared with the network.

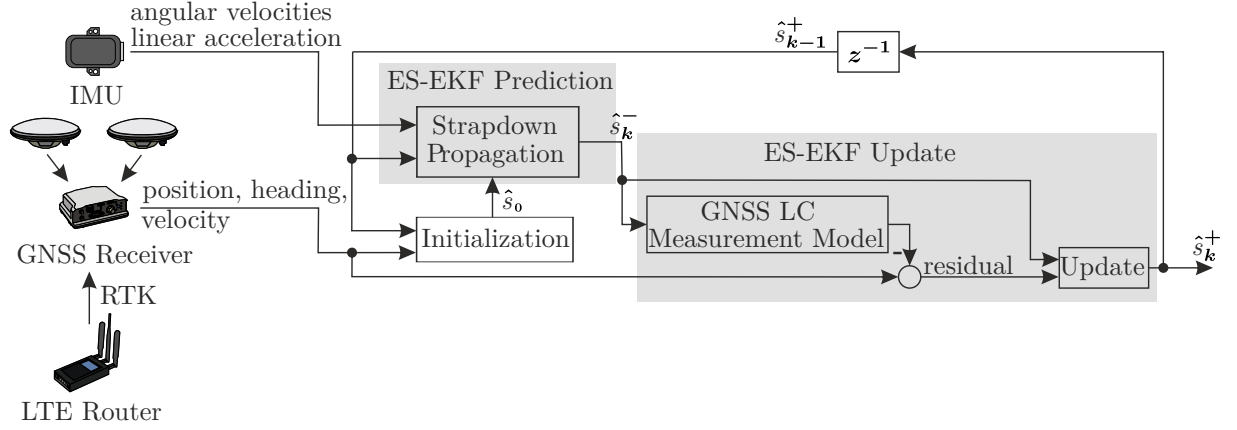


Figure 4: Loosely-coupled (LC) navigation filter based on an Error-State Extended Kalman Filter (ES-EKF) as in [26, 34]. The filter integrates 3D accelerations and rotation rates from an IMU using a strapdown algorithm to propagate a nominal state estimate  $\hat{\mathbf{s}}^-$ , while fusing position, velocity, and heading measurements from a dual-antenna GNSS receiver with RTK corrections via mobile network to update the error state  $\delta\mathbf{s}$ , and corrects the nominal state to  $\hat{\mathbf{s}}^+ = \hat{\mathbf{s}}^- \oplus \delta\mathbf{s}$ .

### 3.2 Vessel State Estimation

To estimate the vessel's state (position, velocity, orientation, and IMU biases), we employ a loosely coupled navigation filter based on an Error-State Extended Kalman Filter (ES-EKF). The ES-EKF maintains a nominal state  $\hat{\mathbf{s}}$  and estimates an additive error state  $\delta\mathbf{s}$ , which is used to correct the nominal state at each update step. The filter fuses a high-frequency inertial measurement unit with a low-frequency GNSS receiver to estimate a high-frequency pose solution. The nominal state is defined as follows:

$$\hat{\mathbf{s}} = [\mathbf{p}_{eb}^e{}^T \mathbf{v}_{eb}^n{}^T \mathbf{q}_b^n{}^T \mathbf{b}_a^T \mathbf{b}_g^T], \quad (1)$$

with the three-dimensional position of the body-frame origin (b) with respect to the Earth-Centered-Earth-Fixed (ECEF) frame (e)  $\mathbf{p}_{eb}^e$ , the three-dimensional velocity of the body frame with respect to the ECEF frame in navigation frame (NED) coordinates (n)  $\mathbf{v}_{eb}^n$ , the quaternion representing the alignment of the body frame to the navigation frame  $\mathbf{q}_b^n$ , the biases of the accelerometer  $\mathbf{b}_a$ , and the biases of the gyroscope  $\mathbf{b}_g$ . The error state  $\delta\mathbf{s}$  is defined as:

$$\delta\mathbf{s} = [\delta\mathbf{p}^T \delta\mathbf{v}^T \delta\mathbf{q}^T \delta\mathbf{b}_a^T \delta\mathbf{b}_g^T] \quad (2)$$

with position error  $\delta\mathbf{p}$ , velocity error  $\delta\mathbf{v}$ , orientation error  $\delta\mathbf{q}$ , and IMU bias errors  $\delta\mathbf{b}_a$ ,  $\delta\mathbf{b}_g$ .

We use the error-state Extended Kalman Filter (ES-EKF) implemented in the work of [26, 34]. The nominal state is propagated at IMU rate using a strapdown algorithm, integrating measured accelerations and angular rates. The ES-EKF estimates  $\delta\mathbf{s}$  based on GNSS measurements of position, velocity, and heading, which are incorporated as measurement updates. The estimated errors are then applied to correct the nominal state:

$$\hat{\mathbf{s}}^+ = \hat{\mathbf{s}}^- \oplus \delta\mathbf{s}. \quad (3)$$

with  $\hat{\mathbf{s}}^-$  indicating the nominal state prior to the update. Both the estimation of  $\delta\mathbf{s}$  and the update appear in the update block in Figure 4.

Additionally, to further refine the GNSS position, we utilize Real-Time Kinematics (RTK) correction data to reach subcentimeter-level precision. The navigation filter also takes a LLH reference point to estimate its current pose within an NED reference frame, with its origin in the reference point. We use this to transform each local sensor frame to the global NED reference frame.

To get all sensor data in a common reference frame, we first transform each sensor's local frame into the imu body frame. Then, we transform the sensor data into the common NED reference frame using the navigation filter solution.

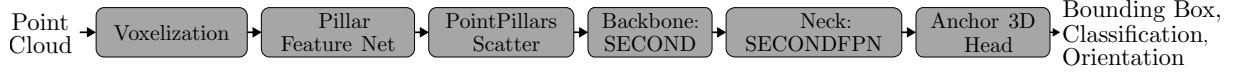


Figure 5: Variation of the PointPillars model [35] using the MMDet3D implementation [36]. The original backbone and neck are replaced with those of SECOND [37] to improve feature extraction and multi-scale representation. The detector learns a nonlinear mapping from point cloud input to bounding box, orientation, and classification of nearby vessels.

### 3.3 Environmental Perception

To estimate the state of vessels in the environment, we use LiDAR-based 3D object detection yielding pose, dimension, and class of detected nearby objects. We transform the LiDAR data to a NED reference frame and perform a 3D object detection similar to [22] but with a different detection architecture. We use a variation of the efficient PointPillars model [35] as implemented in MMDet3D [36]. The model groups the 3D point cloud  $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^N$  with  $\mathbf{p}_i = [x_i, y_i, z_i]^T \in \mathbb{R}^3$  into vertical pillars  $\mathcal{C}_{u,v}$ . For every point  $\mathbf{p}_i \in \mathcal{C}_{u,v}$  we construct the extended feature vector

$$\mathbf{f}_i = [\mathbf{p}_i^\top \quad 1 \quad (\mathbf{p}_i - \bar{\mathbf{p}}_{u,v})^\top \quad (\mathbf{p}_i - \mathbf{c}_{u,v})^\top]^\top, \quad (4)$$

where  $\bar{\mathbf{p}}_{u,v}$  is the pillar mean and  $\mathbf{c}_{u,v}$  its centre. For each pillar  $(u, v)$  we stack its extended point features into the fixed-size tensor  $\mathbf{F}_{u,v} = [\mathbf{f}_1, \dots, \mathbf{f}_{T_{\max}}]^\top \in \mathbb{R}^{T_{\max} \times C}$ , with  $C = 9$  and zero-padding whenever  $|\mathcal{C}_{u,v}| < T_{\max}$ .  $T_{\max}$  depicts the maximum number of points considered in a pillar. The stack is fed to a Pillar Feature Net (PFN) that yields one pillar feature  $\mathbf{g}_{u,v} = \text{PFN}(\mathbf{F}_{u,v})$ . All  $\mathbf{g}_{u,v}$  are scattered into a bird's-eye-view pseudo image  $\mathbf{G} \in \mathbb{R}^{H \times W \times C}$ , enabling efficient 2D convolutions. A single-stage head predicts, per BEV cell, class scores, bounding box offsets to anchors, and a direction bin, and is trained with the combined loss

$$\mathcal{L} = \mathcal{L}_{\text{focal}} + \lambda_{\text{reg}} \mathcal{L}_{\text{smooth } L_1} + \lambda_{\text{dir}} \mathcal{L}_{\text{CE}} \quad (5)$$

with  $\lambda_{\text{reg}}, \lambda_{\text{dir}}$  weighting the loss contributions. The backbone and FPN neck are replaced by SECOND modules [37], while the overall PointPillars pipeline is retained. The model structure is presented in Figure 5.

We train and test the model with simulated data from [38]. We use two different scenarios and extended the simulation with new vessel models used in the real-world evaluation scenario, as well as newly developed plugins for the LiDAR sensors. To improve generalization and robustness, we apply standard LiDAR data augmentation techniques such as object-level noise injection, geometric transformations (translation, rotation, scaling), and flipping [39]. We train the model on four classes *small boat*, *medium boat*, *large boat*, *buoy* with the same anchor sizes as in [22].

The perception module should follow maritime objects over time and predict their future movements. Pose hypotheses for each object originate from a LiDAR-based detection module and AIS (Automatic Identification System) messages. Both sources provide position and heading data, which are transformed into a common North-East-Down (NED) reference frame.

Instead of applying an extended Kalman filter bank with joint probabilistic data association as in [22], we opt for a simpler linear Kalman filter using the StoneSoup tracking framework [40], tailored to a small set of low-dynamic targets such as large vessels in harbor scenarios. Our streamlined approach aims to reduce computational complexity. We define a vessels state as the vector  $\mathbf{x}_k = [x \ \dot{x} \ y \ \dot{y} \ \psi \ \dot{\psi}]^T$ , where  $(x, y)$  indicates position,  $(\dot{x}, \dot{y})$  denote the linear velocities,  $\psi$  is the heading angle, and  $\dot{\psi}$  is the heading rate. All components are expressed in the NED frame. A linear constant-velocity model predicts the state's progression with  $\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}$ .

Noise is modeled as additive Gaussian  $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$  with covariance matrix  $\mathbf{Q}$ .

For sensor measurements from AIS and LiDAR, we employ a linear Gaussian observation model  $\mathbf{z}_k^i = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k^i$  with measurement noise  $\mathbf{v}_k^i \sim \mathcal{N}(\mathbf{0}, \mathbf{R}^i)$ , where  $\mathbf{R}^i$  is sensor-specific covariance matrix, and  $i \in \{\text{LiDAR}, \text{AIS}\}$ . The observation matrix  $\mathbf{H}$  selects the observed state components. The transition and observation matrix are given as

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$



If both LiDAR- and AIS-based observations are available in a given time step, we apply sequential Kalman updates.

The above prediction and correction steps of the filter require the association of measurements to tracks. For data association, we compute the Mahalanobis distance [41]  $d_M^2 = (\mathbf{z}_k - \hat{\mathbf{z}}_k)^\top \mathbf{S}_k^{-1} (\mathbf{z}_k - \hat{\mathbf{z}}_k)$  between predicted tracks and measurements with covariance matrix  $\mathbf{S}_k = \mathbf{H}\mathbf{P}_k\mathbf{H}^\top + \mathbf{R}$  and state covariance matrix  $\mathbf{P}_k$ . We then form a cost matrix  $\mathbf{C}_{i,j} = \text{Mahalanobis}(\text{Track}_i, \text{Measurement}_j)$ . A Hungarian algorithm [42] computes the matches with minimal cost. Tracks are initialized for each measurement that is not associated with an existing track. Tracks that remain without an update for a time threshold of 10s are removed from the track list.

To fuse detected objects from multiple connected agents, we add the LiDAR object detections as additional observations.

A static mapping module accounts for static objects like piers or harbor edges. First, we follow [22] and subtract points of objects detected by the object detection module from the 3D LiDAR point cloud, seen in Figure 3, as those include dynamic objects. This yields a point cloud that represents only static objects. The object detection module shares all detected objects, including dynamic but also static vessels, with the network, such that the planning module can account for both types of obstacles. Second, the 3D point cloud is projected into a birds-eye-view (BEV) perspective and merged with the 2D radar point cloud. This extends the perception range to about 1 km and simplifies the mapping approach to 2D, which suffices to detect land mass and piers.

To represent the static environment, we implement a 2D probabilistic occupancy grid map (OGM) [43]  $\mathbf{G} \in \mathbb{R}^{H \times W}$  of height  $H$  and width  $W$ :

$$\mathbf{G}_{i,j} = \begin{cases} 1, & \text{if cell } (i,j) \text{ is fully occupied} \\ 0, & \text{if cell } (i,j) \text{ is free} \\ 0.5, & \text{if cell } (i,j) \text{ is unknown} \end{cases}.$$

The map discretizes the space into equally sized grid cells where each cell  $c_{i,j}$  stores the probability  $p_{i,j} \in [0, 1]$  that the corresponding area is occupied. The map is initialized with each cell being in the unknown state  $p_{i,j} = 0.5$ .

We determine the current position of the sensor in the grid map and cast rays with a specifiable angular resolution and range. If a ray hits a point in the point cloud, the corresponding cell is marked as occupied. Cells that are in the shadow of this occupied cell remain unknown. Cells a ray can pass are marked as free.

To efficiently update the map, we use log-odds representation  $l_{i,j}$  of occupancy probability [44] of cell  $c_{i,j}$  with

$$l_{i,j} = \log \left( \frac{p_{i,j}}{1 - p_{i,j}} \right).$$

When new measurements arrive, we update the cells additively

$$\begin{aligned} l_{i,j}^{(t)} &= l_{i,j}^{(t-1)} + l_{\text{occ}} \quad , \text{ if cell is occupied} \\ l_{i,j}^{(t)} &= l_{i,j}^{(t-1)} + l_{\text{free}} \quad , \text{ if cell is free} \end{aligned}$$

with  $l_{\text{occ}}$  and  $l_{\text{free}}$  depicting the log-odds increment parameters for occupied and free observations, respectively. In a multi-vessel scenario, we fuse aligned maps from multiple agents by adding the log-odds values of corresponding cells.

To extract static objects from the grid map, we extract contours of unknown and occupied space boundaries from the occupancy grid at each iteration. We implement a contour-finding algorithm on a binary mask of the grid. Given the occupancy grid  $\mathbf{G}$ , we first threshold the grid to obtain a binary grid  $\mathbf{G}_{\text{binary}}$ .

We then use the Marching Squares algorithm [45] implemented in the SciKit-Image library [46] to find  $k$  contours of the binary pseudo image as a set  $\mathcal{C}_k$  of edge points, which are then polygons of static objects:

$$\mathcal{C}_k = \text{MarchingSquares}(\mathbf{G}_{\text{binary}}).$$

To ensure the correct data type for the planning module, all polygons are required to be convex. Non-convex polygons are therefore transformed into their convex hulls using the ConvexHull algorithm

from the SciPy library [47,48]. This transformation over-approximates the original non-convex polygon, as the convex hull is guaranteed to contain all the original vertices and may include additional area not present in the original shape. Overall, the static mapping module generates a 2D occupancy grid map and extracts static objects as convex polygons.

## 4 Collision Avoidance Planning Module

### 4.1 Regulatory Aspects of Port Operations

For collision avoidance of ships, the internationally valid COLREGs are generally used. However, the defined rules cannot be applied to port operations in general. Therefore, basic rules have been adopted by national regulations, in the case of Germany, the German Traffic Regulations for Navigable Maritime Waterways (SeeSchStrO) [49].

If it is not individually defined, the following basic rules apply:

- Shall keep a safe speed (COLREGs Rule 6) / basic principles) limited to the maximum speed allowed in the specific port.
- When meeting, overtaking, or passing other vessels or fixed or floating installations, vessels shall do so at a safe distance (COLREGs Rule 8(d) / basic principles).
- Shall keep as near to the outer limit of the fairway which lies on her starboard side as is safe and practicable (COLREGs Rule 9(a) / basic principles).
- An overtaking vessel shall pass the vessel being overtaken on the latter vessel's port side until she is finally past and clear (COLREGs Rule 13 / overtaking).
- In a head-on situation in a fairway, whether courses are straight head-on or nearly straight head-on, evasive action shall be taken to the starboard side (COLREGs Rule 14 / head-on situation).
- A vessel proceeding along the course of the fairway shall have the right of way over vessels entering, crossing, making turns in that fairway, or leaving their mooring grounds (COLREGs Rules 9, 15, 18 / crossing situation).

These basic rules are then adapted to the specific port. For example, maximum speeds, reporting obligations, or special right-of-way and maneuvering rules are defined. Central coordination by the VTS avoids encounters between more than two ships, especially if they require a large maneuvering space, to keep the complexity of the situations as low as possible. This can lead to waiting times at the pier or a reduction in the vessel speed. The maximum speed in the port of Rostock is 6.5 knots.

Especially in crossing situations, it is crucial to consider the prioritized navigation channel to be able to decide which vessel has the right of way. For example, the encounter situation shown in Figure 1 is particularly challenging, as the Deneb has the Messin on its starboard side, but is moving within the main navigation channel and therefore has priority over the Messin moving into the navigation channel.

### 4.2 Formulation of the Optimization Problem

With the system architecture presented in Figure 2, a centralized approach for trajectory planning avoids collisions. This relies on the networked vessels transmitting their current desired trajectories or paths. In the simplest case, this can e.g. be done by sending the generally used berth-to-berth waypoint route. On this basis, a collision check is carried out automatically at the control center. Once a collision risk is detected, collision avoidance trajectories for the involved vessels are computed based on the initial desired paths.

The calculation method is based on a nonlinear optimal control problem (OCP). The general form of an OCP is given by

$$\begin{aligned}
 & \min_{\mathbf{x}(t), \mathbf{u}(t), t_f} J(\mathbf{x}(t), \mathbf{u}(t), t_f) \\
 & \text{subject to } \dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)), \\
 & \quad \mathbf{x}(t) \in [\mathbf{x}_{\min}, \mathbf{x}_{\max}], \\
 & \quad \mathbf{u}(t) \in [\mathbf{u}_{\min}, \mathbf{u}_{\max}], \\
 & \quad \mathbf{x}(t_0) = \mathbf{x}_0, \\
 & \quad \mathbf{x}(t_f) = \mathbf{x}_f, \\
 & \quad \mathbf{h}(\mathbf{x}(t)) \leq 0
 \end{aligned} \tag{6}$$

This optimization problem aims to find the optimal state trajectory  $\mathbf{x}(t)$ , control input  $\mathbf{u}(t)$ , and final time  $t_f$  that minimize the cost functional  $J(\mathbf{x}(t), \mathbf{u}(t), t_f)$ . The system dynamics are governed by the differential equation  $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$ . The state and control inputs are constrained to lie within the admissible bounds  $[\mathbf{x}_{\min}, \mathbf{x}_{\max}]$  and  $[\mathbf{u}_{\min}, \mathbf{u}_{\max}]$ , respectively. Furthermore, the initial and final states are fixed at  $\mathbf{x}_0$  and  $\mathbf{x}_f$ .

The inequality constraint  $\mathbf{h}(\mathbf{x}(t)) \leq 0$  ensures that additional path or safety constraints, such as collision avoidance, are satisfied throughout the maneuver. The final time  $t_f$  represents the moment at which the maneuver is successfully completed.

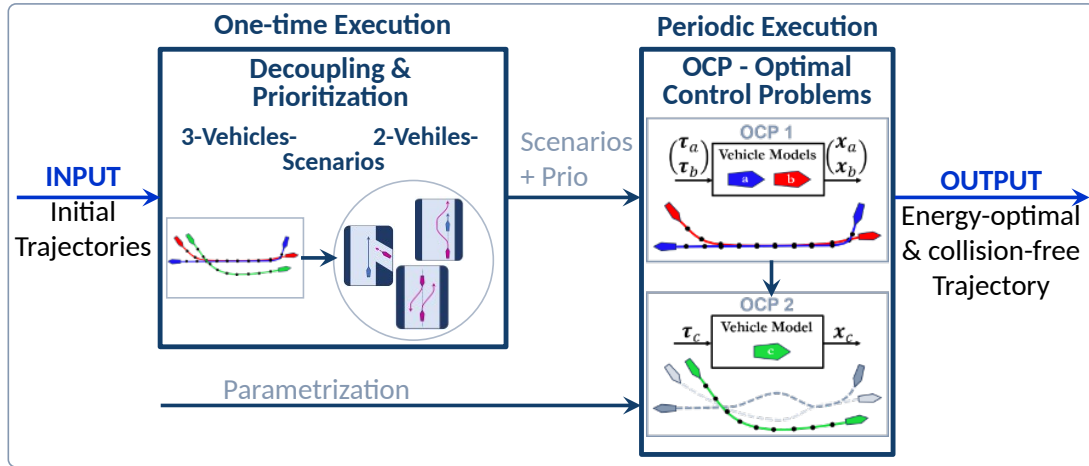


Figure 6: Schematic representation of the approach for collision avoidance trajectory generation with submodules scenario collision risk detection, decoupling and prioritization as well as solving the OCPs

The proposed approach, as illustrated in Figure 6, is based on the analysis of given initial trajectories of multiple vessels, with the objective of identifying and classifying potential collisions. The method is designed to handle multi-ship collision scenarios. Its scope is limited to head-on and crossing encounters, which are particularly relevant in constrained environments. To systematically manage the complexity of such multi-vessel situations, each encounter is decomposed into a set of pairwise two-ship collision subproblems, categorized by collision type. The underlying idea is to reduce a complex multi-ship situation into simpler, more tractable two-vessel interactions.

These individual subproblems are then prioritized, with head-on situations assigned higher priority than crossing ones. This reflects the assumption that such encounters occur within a port area, where vessels involved in head-on interactions are likely to navigate along the prioritized navigation channel. According to COLREGs, these vessels generally have right-of-way over those entering, crossing, or leaving their mooring areas see Section 4.1. The crossing participants are then compared with the results of the head-on situation to ensure that the final solution complies with the COLREGs.

Once the relevant collision scenarios have been identified, they are resolved sequentially based on their assigned priority, while ensuring adherence to applicable COLREGs. The first optimal control problem, denoted as **OCP 1**, formulates a coupled trajectory optimization for a head-on scenario, in which evasive maneuvers for both vessels are computed simultaneously. The resulting collision-free trajectories are then used as dynamic obstacles in a subsequent optimal control problem, **OCP 2**, which computes the evasive path for a third vessel potentially involved in a crossing situation with the head-on participants.

The collision avoidance approach starts by classifying vessel encounters. In the present implementation, only motorized vessels are considered, excluding sailing boats and other maneuvering-limited vessels. Two principal encounter types are distinguished:

- **Head-on situation:** This occurs when two vessels are approaching each other on reciprocal or nearly reciprocal courses, such that both are required to alter course to starboard to avoid collision. Practically, this applies when the relative course between the vessels is approximately  $150^\circ$  to  $180^\circ$ .
- **Crossing situation:** This occurs when two vessels are approaching on crossing courses with a risk of collision. In such cases, the vessel that has the other on its starboard side must give way. This typically applies when the relative course difference is between  $45^\circ$  and  $135^\circ$ .

Based on these definitions, the method applies a threshold of  $150^\circ$  to  $180^\circ$  to classify head-on encounters and uses the range from  $45^\circ$  to  $135^\circ$  to identify crossing situations.

In the next step, the identified and prioritized collision scenarios are addressed through a sequence of OCPs. Although OCPs follow the general structure outlined in Equation (6), they differ in their specific objectives, state definitions, and constraints depending on the scenario. The cost function for OCP 1 is defined as

$$J_h = \min_{\mathbf{x}(t), \mathbf{u}(t)} J_h(\mathbf{x}(t), \mathbf{u}(t)), \quad (7)$$

$$J_h = \int_0^{t_f} \Delta \mathbf{p}_h^\top \mathbf{Q} \Delta \mathbf{p}_h + \mathbf{u}_h^\top \mathbf{R} \mathbf{u}_h dt.$$

Here, the final time  $t_f$  is fixed and corresponds to the end of the maneuver, as determined by the initial trajectories of the vessels. The state vector is defined as  $\mathbf{x} = [\mathbf{p}_h^\top \mathbf{v}_h^\top]^\top$ , where the position and heading vector is given by  $\mathbf{p}_h = [x_1 \ x_2 \ y_1 \ y_2 \ \psi_1 \ \psi_2]^\top$  and the translational and rotational velocity vector by  $\mathbf{v}_h = [u_1 \ u_2 \ v_1 \ v_2 \ r_1 \ r_2]^\top$ . The control input vector is defined as  $\mathbf{u}_h = [X_1 \ X_2 \ Y_1 \ Y_2 \ N_1 \ N_2]^\top$ , representing the surge forces  $X$ , sway forces  $Y$ , and yaw moments  $N$  applied to each vessel. Indices 1 and 2 in the vectors refer to the two vessels involved in the head-on situation. The term  $\Delta \mathbf{p}_h = \mathbf{p}_h - \mathbf{p}_{h,\text{init}}$  describes the deviation from the nominal trajectory, where  $\mathbf{p}_{h,\text{init}}$  represents the initial paths obtained from the initial trajectories or path. The cost function  $J_h$  penalizes both the deviation from the nominal path via the weighting matrix  $\mathbf{Q}$  and the control effort via the matrix  $\mathbf{R}$ .

To formulate collision avoidance constraints within the optimal control framework, vessels are conservatively approximated using circular shapes, as illustrated in Figure 7. Each vessel is initially modeled as a rectangle reflecting its physical dimensions. To incorporate a safety margin, this rectangle is enlarged to form a bounding box that accounts for navigational uncertainty and control limitations. A circle is then fitted around this bounding box, with its radius chosen such that it fully encloses the expanded shape. These circles are subsequently used in the collision constraints to define pairwise minimum separation requirements between vessels.

In head-on encounters, the enlarged bounding rectangle is shifted laterally to the starboard side of each vessel by a constant offset vector  $\mathbf{a}$ , defined in the ship's local reference frame. This ensures that both vessels execute a starboard-side evasive maneuver, in accordance with the COLREGs, thereby reducing collision risk through coordinated and rule-compliant behavior.

The center of the ship is denoted as  $P_i = (x_i, y_i)$ . The center of the corresponding safety circle is computed as

$$P_i = P_i + R(\psi_i) \cdot \mathbf{a},$$

where  $R(\psi_i)$  is the rotation matrix defined by the vessel's heading angle  $\psi_i$ . This geometric transformation ensures that the safety circles are positioned asymmetrically to favor starboard-side passing, as illustrated in Figure 7. Collisions between vehicles 1 and 2 are prohibited via requesting that either disc of vessel 1 does not overlap with either disc of vessel 2 using

$$\|\mathbf{p}_1 - \mathbf{p}_2\|^2 \geq (R_1^D + R_2^D)^2, \quad (8)$$

where  $R_1^D$  and  $R_2^D$  denote the safety radii of vessel 1 and vessel 2, respectively.

In OCP 2, the collision-free trajectories obtained from the solution of OCP 1 are treated as dynamic obstacles and incorporated into the optimization problem as constraints. In contrast to OCP 1, the final time  $t_f$  is considered as a decision variable and is explicitly included in the cost function  $J_c$ , which leads to

$$J_c = \min_{\mathbf{x}(t), \mathbf{u}(t), t_f} J_c(\mathbf{x}(t), \mathbf{u}(t), t_f), \quad (9)$$

$$J_c = \int_0^{t_f} \Delta \mathbf{p}_c^\top \mathbf{Q} \Delta \mathbf{p}_c + \mathbf{u}_c^\top \mathbf{R} \mathbf{u}_c + w t_f dt.$$

Here, the weight  $w$  penalizes excessive maneuver duration, which typically results from speed reductions yielding to the higher-priority vessels in accordance with the right-of-way rules described in Section 4.1. The state vector for the crossing vessel is defined as  $\mathbf{x}(t) = [\mathbf{p}_c^\top(t) \mathbf{v}_c^\top(t)]^\top$ , where the position vector is  $\mathbf{p}_c = [x \ y \ \psi]^\top$  and the velocity vector is  $\mathbf{v}_c = [u \ v \ r]^\top$ . The control input is given by  $\mathbf{u}_c = [X \ Y \ N]^\top$ , and the deviation from the nominal path is defined as  $\Delta \mathbf{p}_c = \mathbf{p}_c - \mathbf{p}_{c,\text{init}}$ .

For the purpose of collision avoidance in OCP 2, these obstacle trajectories are represented using two safety circles per vessel, where one is located forward and one aft of the ship's center, as illustrated in Figure 7. This two-circle approximation is applied uniformly to all vessels, including those originating from OCP 1, in order to enable a more spatially accurate and less conservative representation of occupied



areas during the maneuver. The center coordinates of the front and rear circles are determined from the ship's center position  $(x, y)$ , heading angle  $\psi$ , and a fixed offset distance  $d$ , using

$$\begin{aligned} x_f &= x + d \cdot \cos(\psi), & y_f &= y + d \cdot \sin(\psi), \\ x_r &= x - d \cdot \cos(\psi), & y_r &= y - d \cdot \sin(\psi). \end{aligned} \quad (10)$$

To ensure a minimum clearance between all safety zones, distance constraints are imposed at each time step  $t_k$  between all combinations of circle centers of the crossing and obstacle vessels, applying

$$\begin{aligned} \|\mathbf{p}_{f,i}(t_k) - \mathbf{p}_{f,o}(t_k)\|^2 &\geq (R_{f,i} + R_{f,o})^2, \\ \|\mathbf{p}_{f,i}(t_k) - \mathbf{p}_{r,o}(t_k)\|^2 &\geq (R_{f,i} + R_{r,o})^2, \\ \|\mathbf{p}_{r,i}(t_k) - \mathbf{p}_{r,o}(t_k)\|^2 &\geq (R_{r,i} + R_{r,o})^2, \\ \|\mathbf{p}_{r,i}(t_k) - \mathbf{p}_{f,o}(t_k)\|^2 &\geq (R_{r,i} + R_{f,o})^2. \end{aligned} \quad (11)$$

These constraints ensure collision-free behavior by maintaining sufficient spatial separation between the safety regions of all vessels throughout the maneuver.

In both OCP 1 and OCP 2, static obstacles in the environment are also taken into account. These fixed structures from a map or the global environmental perception are also approximated using circles to ensure compatibility with the collision-avoidance framework. A two-stage procedure limits the number of constraints introduced into the optimization problem and thereby reduces computational complexity: Initially, the trajectory optimization is performed without considering static obstacles. Subsequently, a collision check is conducted between the optimized trajectories and the static environment. If any collisions are detected, the optimization is repeated with the corresponding static obstacle constraints included. This iterative process continues until a set of collision-free trajectories is obtained.

The optimal control problem formulated in (6) incorporates the dynamic behavior of the vessels through the system dynamics constraint  $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t))$ . In both OCP 1 and OCP 2, the function  $f(\cdot)$  is instantiated using a conservative, force-based ship model, which follows the structure of well-established marine vehicle models, [14]. The same underlying model is applied consistently across both problems to ensure dynamically feasible and physically realistic trajectories.

For each vessel, a generic model structure in the form of

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} = \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{\tau} \quad (12)$$

is applied, where  $\boldsymbol{\nu} = [u \ v \ r]^T$  denotes again the velocity vector in surge, sway, and yaw.  $\mathbf{M}$  denotes inertia including added effects,  $\mathbf{C}(\boldsymbol{\nu})$  models Coriolis and centripetal effects and  $\mathbf{D}(\boldsymbol{\nu})$  captures nonlinear hydrodynamic damping.  $\boldsymbol{\tau} = [F_u \ F_v \ M_r]^T$  represents the actuator forces and moment. This model is transformed into the generic nonlinear state-space structure

$$\dot{\boldsymbol{\nu}} = \mathbf{M}^{-1} (\mathbf{D}(\boldsymbol{\nu}) - \mathbf{C}(\boldsymbol{\nu})) \boldsymbol{\nu} + \mathbf{M}^{-1} \boldsymbol{\tau} = \mathbf{F} \cdot \left( \mathbf{A}\boldsymbol{\nu} + \mathbf{C} \begin{bmatrix} vr \\ ur \\ uv \end{bmatrix} + \mathbf{D} \begin{bmatrix} u^3 \\ v^3 \\ r^3 \end{bmatrix} + \mathbf{E} \begin{bmatrix} u^2 \\ v^2 \\ r^2 \end{bmatrix} + \mathbf{B}\boldsymbol{\tau} \right), \quad (13)$$

where the matrices  $\mathbf{A}$  through  $\mathbf{F}$  are of size  $(3 \times 3)$ , vessel-specific and identified experimentally using test data from real-world trials. The parameters used for generating the avoidance trajectories for the involved vehicles are given in Table 1.

Table 1: Identified parameters for vessel Deneb in  $10^3$  SI units

Index	A	B	C	D	E	F
(1,1)	-1.9482	0.00199	1532.75	-0.83975	0.00000	1000.00
(1,2)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(1,3)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(2,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(2,2)	-12.089	0.00132	0.00000	-103.55	0.00000	1000.00
(2,3)	-188.60	0.00000	0.00000	0.00000	0.00000	0.00000
(3,1)	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
(3,2)	-0.47304	0.00000	0.00000	0.00000	0.00000	0.00000
(3,3)	-19.707	0.00474	0.00000	-71369.1	0.00000	1000.00

In OCP 1, this model is applied in coupled form, where the trajectories of both interacting vessels are optimized simultaneously. This joint formulation enables the coordinated computation of collision-free

avoidance maneuvers. In contrast, OCP 2 focuses on a single vessel, whose trajectory is computed while treating the previously planned paths of other vessels as dynamic obstacles. In this case, the model is used independently for the crossing vessel.

It is assumed that all ships have a control system to follow the calculated model-based trajectories. This also includes the ability to execute the planned maneuvers within the limits of the actuators and under environmental influences. An analytical solution to (6) cannot be obtained in general and therefore

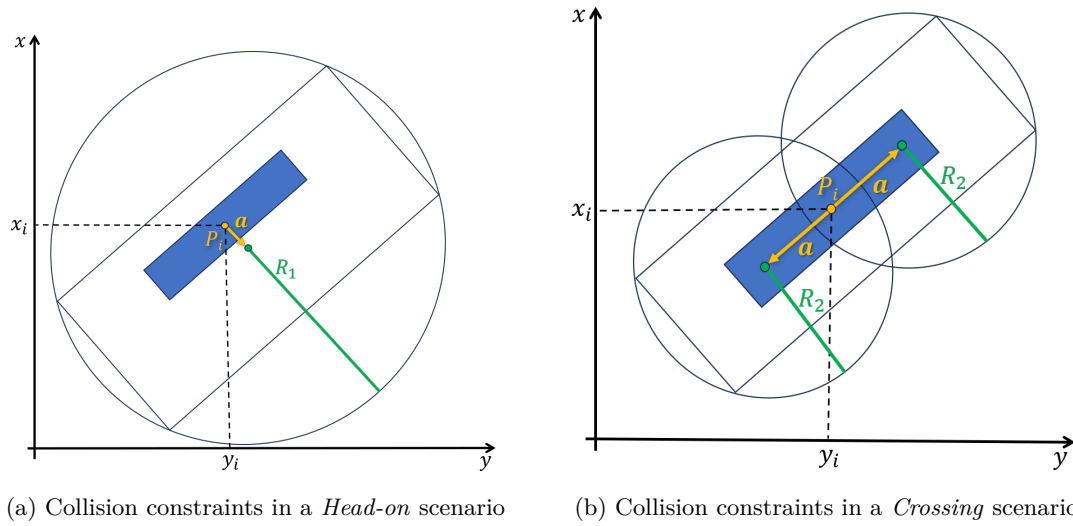


Figure 7: Implementation of collision constraints for vessel-to-vessel interactions through custom circle approximation for the head-on and crossing scenario

must be computed numerically. The first step in the numerical solution of an OCP is to discretize it into a nonlinear programming (NLP) problem. Abstracting OCP (6) as

$$\begin{aligned} \min_{\mathbf{x}(t), \mathbf{u}(t), t_f} \quad & J(\mathbf{x}(t), \mathbf{u}(t), t_f) \\ \text{s.t.} \quad & \text{constraints,} \end{aligned} \quad (14)$$

the problem is discretized using a multiple shooting approach. The time horizon  $[0, t_f]$  is divided into  $N_{FE}$  intervals of equal length  $h = t_f/N_{FE}$ . For each interval  $[t_i, t_{i+1}]$ , the state  $X = [x_0^\top, \dots, x_{N_{FE}}^\top]^\top$  and control  $U = [u_0^\top, \dots, u_{N_{FE}-1}^\top]^\top$  are introduced as decision variables. The system dynamics are integrated over each subinterval using a Runge–Kutta method, which leads to the following NLP formulation

$$\begin{aligned} \min_{X, U, t_f} \quad & J(X, U, t_f) \\ \text{s.t.} \quad & \text{constraints.} \end{aligned} \quad (15)$$

The resulting NLP is implemented and solved using CasADi [17].

OCP1 contains 16,206 optimization variables, 10,800 equality constraints, and 900 inequality constraints. The computation time is around 20 s. OCP2 contains 8,099 optimization variables, 5,400 equality constraints, and 1,800 inequality constraints. The computation time is around 5 s.

## 5 Experimental Setup

### 5.1 Vessels and Instrumentation

To validate the developed modules for cooperative maneuvering in confined waters, three different vessels are used as experimental platforms, consisting of the survey, wreck search and research vessel Deneb of the BSH [50], the experimental catamaran Bernhard Lampe (BeLa) and the USV Messin, depicted in Figure 8. Concerning the system architecture (Figure 2), each vessel is equipped with its local GNC system. Since the Deneb is the largest ship within the experimental setup, the central processing unit (Nuvo-9006E-PoE with Intel Core i9-14900 processor), representing the central traffic control domain, was installed on the Deneb.

The vessels are networked with the central service via 4G/5G mobile network. A MikroTik CCR2004-16G-2S+PC serves as the central router and VPN server. Each vessel is equipped with a Teltonika RUTC50 5G router (downward compatible with 4G/LTE), which also serves as a time server utilizing GNSS and 5G as a time source. During the experiments, the state vectors from the vessel state estimation, the object detections from the environmental perception, as well as the non-collision-free initial trajectories of the vessels and the generated collision avoidance trajectories are transmitted between the vessels and the central service via the established network.



(a) Survey, wreck search and research vessel Deneb of the BSH



(b) Experimental catamaran BeLa of the University of Rostock



(c) USV Messin of the University of Rostock

Figure 8: Three vessels used as experimental platforms for system evaluation

The vessel Deneb measures 52 m in length has a width of 11.4 m and a draft of 3.5 m. The vessel reaches a maximum speed of 11 knots. The Deneb is actuated by a conventional fixed pitch propeller with rudder, a Pump-Jet in the bow, and a stern transverse thruster. Due to the actuator configuration, the Deneb is fully actuated at low speed and is capable of realizing dynamic positioning (DP) functionalities. For detailed information about the vessel automation and the applied guidance and maneuvering control system, the reader is referred to [18] and [33]. In addition, within the experiments presented here, the trajectory tracking task is realized by a model predictive control structure, as it is presented in [51]. For the execution of the guidance and control modules as well as the visualization on board the Deneb, a framework of three industrial computers (Nuvo-9006E-PoE with Intel Core i9-14900 processor) is used, which work in a local network with time synchronization. Finally, for automatic vessel control, dedicated Ethernet interfaces are available to receive the estimated vessel state from the navigation module and to command the actuator set values.

For localization, Deneb was equipped with two Septentrio GNSS antennas, a Mosaic-H receiver operating at 10 Hz and a Lord Microstrain GX5-25 IMU operating at 100 Hz. The GNSS receiver obtains Real-Time Kinematic (RTK) correction data to ensure precise localization. An onboard computer, comprising an AMD Ryzen Threadripper PRO 5975WX processor with 256 GB RAM and two NVIDIA GeForce RTX 4090, processes the sensor data, software modules, and fuses multi-vessel data. The Deneb's sensors include five solid-state Robosense M1P LiDAR sensors with a maximum range of 150 m, a horizontal field of view of 120°, a vertical field of view of 25°, and a scanning rate of 10 Hz. The sensors support PTP synchronization but cannot be triggered simultaneously due to the solid-state technology. A sixth LiDAR sensor is mounted on the observation deck of the vessel for calibration purposes. For long-range perception, a NavTech RAS6 radar covers up to 1000 m scanning in a 2D plane at 4 Hz. The radar provides a 360° horizontal field of view with an azimuth beam width of 1.8° and an elevation beam width of 3.6°. A ZED stereo camera offers visual feedback and supports scenario recordings. All sensors connect to the computing device via Ethernet, and a PTP-transparent switch ensures synchronized data throughout the network.

Extrinsic calibration transforms each local sensor frame to a body reference frame. For a multi-LiDAR setup, the SIMPLE-C tool (currently submitted to RA-L) performs semi-automatic calibration. For the calibration of the remaining sensors, the distance and the roll, pitch, and yaw angle to the nearby LiDAR sensors are measured.

The experimental catamaran Bernhard Lampe of the University of Rostock, shown in Figure 8b, has a length of 7.35 m, a width of 2.44 m, a draft of 0.35 m and is powered by two outboard jet drives with 140 HP each resulting in a maximum speed of approximately 30 knots. Due to repairs, the vessel was not available during the test period. Thus, the vessel has been simulated during the experiments.

The USV Messin depicted in Figure 8c is a catamaran with a length of 3.44 m, a width of 1.65 m, and a draft of 0.35 m. The USV is equipped with two podded drives of 400 W each and achieves a maximum speed of 4 knots. The podded drives are also used for steering. The reader is referred to [52] for detailed information about the applied local guidance and control modules of the USV.

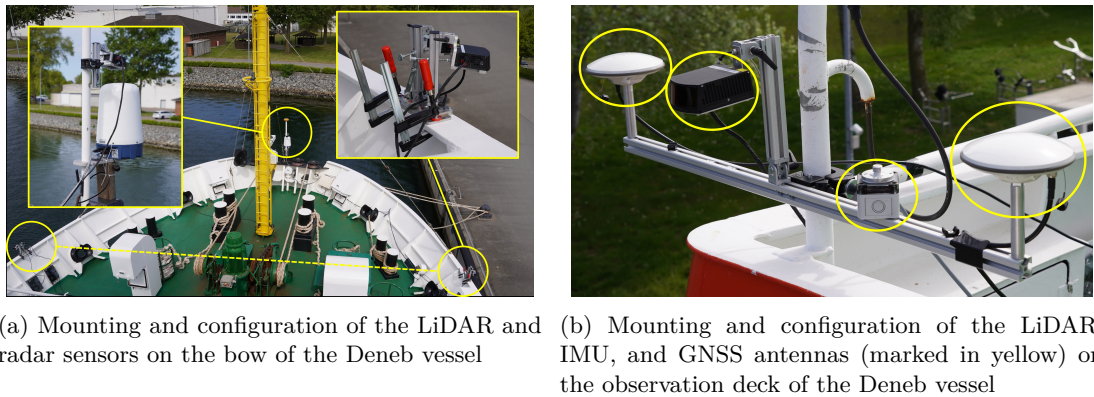


Figure 9: Detailed mounting and configuration of the sensors on the Deneb vessel

The Messin localization system relies on two GNSS antennas linked to a Septentrio Mosaic-H receiver, providing position and orientation data. In addition, a Lord Microstrain GX5-25 IMU delivers acceleration and angular rate measurements at 100 Hz. Environmental perception is managed by a Robosense RSP128 LiDAR sensor, offering 360° coverage through 128 layers, spanning a vertical field of view from  $-25^\circ \sim +15^\circ$  at a 10 Hz scan rate. Onboard processing and perception tasks run on an NVIDIA Jetson Orin. The LiDAR connects to the Jetson Orin over Ethernet; the IMU uses a serial-to-USB interface. The Jetson Orin itself integrates with the USV network via Ethernet for communication and to receive RTK correction data for precise GNSS localization.

### 5.2 Experimental Protocol: Evaluation Scenario

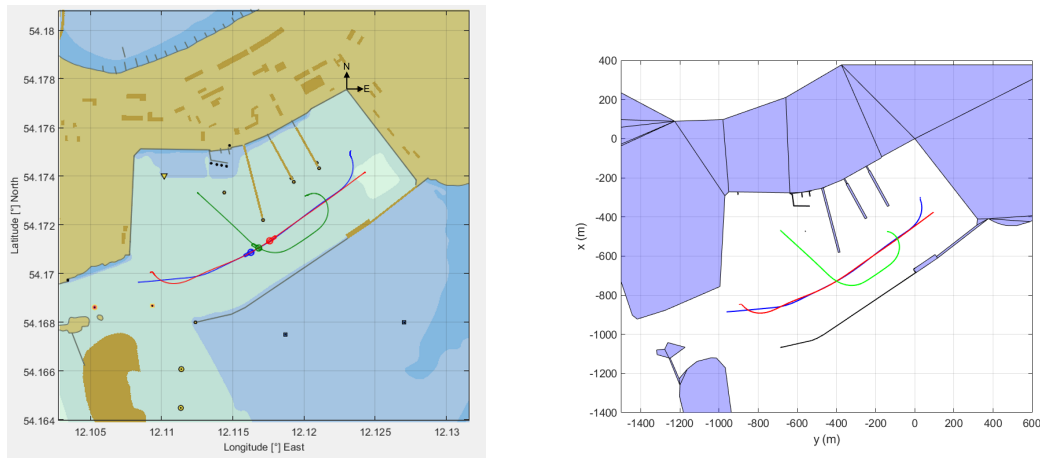
To validate the performance of the developed methods, a complex maneuvering scenario in the port of Rostock was defined as illustrated in Figure 10. Taking ODDs into account, the nautical officers of DENEb and ISSIMS GmbH supported the definition of safety parameters for automatic maneuvering. DENEb serves as the reference vessel, with safety distances of 50 m ahead and 25 m to the sides and astern. Automatic operation is limited to wind speeds below 8 m/s, especially when wind assists forward motion. In DP mode, wind plays a minor role. Each vessel must have a dedicated operator or chief mate on board to take manual control if needed. The scenario is aborted if deviations exceed  $5^\circ$  in heading, 10 m to the trajectory or 1 m/s in speed.

Figure 10a shows the ENC information of the area and the given nautically planned initial trajectories of the three involved vessels, Deneb, BeLa, and Messin. The trajectories are planned generically so that each of the test vessels can follow any trajectory, and thus different vessel constellations can be tested. Due to the speed limitation of USV Messin, the maximum trajectory speed has been set to 4 knots. The initial speed for all trajectories is 0 knots. Additionally, the figure shows the time 350 s after the start of each maneuver, whereby several collisions would inevitably occur without evasive maneuvers.

Following the COLREGs, the situation can be divided into the two core test scenarios *Head-on* (red and blue trajectories) and *Crossing* (red/blue and green trajectories). The collision avoidance maneuvers should be as close as possible to the initial trajectories, but maintain the safety distances between the vessels and the port structures. The critical area, where the collision avoidance maneuver has to be realized, has a size of approximately  $300\text{ m} \times 300\text{ m}$ , and the straight sections of red and blue are approximately 1000 m. As this may cause distances too large to be covered by the medium-range LiDAR sensors, we added a small drive-by scenario. While the Deneb vessel was docked, Messin and the remote operation dinghy passed by with distances of 10 m to 100 m to ensure visibility and enable validation of the object detection.

In the head-on situation, two vessels are approaching each other on nearly reciprocal courses with a risk of collision, both vessels should alter their course to starboard, ensuring they pass on each other's port side, in accordance with COLREGs Rule 14. In the crossing scenario two vessels' paths intersect. According to COLREGs Rule 15, when two vessels cross with risk of collision, the vessel which has the other on her starboard side shall keep out of the way. This applies to the red one. Since the head-on vessels on the red and blue trajectories are moving in the prioritized navigation channel, the vessel on the green trajectory starting from the berth should wait or approach at low speed until the head-on situation has been resolved.





(a) Non collision-free initial trajectories resulting in a *Head-on* (red and blue trajectory) and a *Crossing* (red/blue and green trajectory) collision situation

(b) Approximation of the port area and fixed structures using convex polygons

Figure 10: Overview of the evaluation scenario involving three vessels in a confined area of the port of Rostock

As the catamaran BeLa was not available for the tests at short notice, we adapted the test protocol as follows: Each scenario was conducted independently with the research vessel Deneb and the USV Messin. In the combined test runs, the central traffic management simulated the second ship in the head-on situation as BeLa, while Deneb and Messin operated physically on site.

Prior to the launch of the automated scenario, the three vessels navigate near the respective starting positions of the initial trajectories, as shown in Figure 10a. Afterwards, the scenario was started in a coordinated manner in order to maintain the real collision risk. The central collision avoidance planner transmits the adapted collision-free trajectories, calculated on the basis of the available non-collision-free initial trajectories as described in Section 4, to the three vessels involved.

### 5.3 Sensor Processing and Performance Indicators

Each vessel runs its own localization and perception pipeline, as described in Section 3. The sensors communicate locally with the computing device via ROS2. The resulting detections are then fused within the global environmental perception module onboard the Deneb, which also hosts the collision avoidance planner, described in Section 4. For the initial map constraints used within the OCP, the ENC data was approximated using convex polygons as shown in e.g. [53] and visualized in Figure 10b.

For the perception software modules, the following parameter setup was chosen: For the training of the object detection model, we use 9500 samples in the training set and 1500 samples in the test set. We train the model for 100 epochs and a batch size of 6 using the one-cycle learning rate policy [54] on two NVIDIA RTX 4090 GPUs with 24 GB VRAM each. For the PointPillars detector, we discretise each point cloud into pillars of  $0.5 \times 0.5$  m and retain points within the range  $[-200, -200, -1]$  m,  $[200, 200, 30]$  m. We take the anchors as in [22]:  $[4.28, 1.92, 1.60]$  m for small boats,  $[9.39, 3.15, 7.54]$  m for medium boats,  $[15.74, 4.40, 15.92]$  m for large boats, and  $[0.93, 0.93, 1.47]$  m for buoys. To improve generalisation, we apply standard data augmentations during training like object-level jittering (translations of roughly 1 m and in-place rotations up to  $\pm 45^\circ$ , global scene rotations of  $\pm 45^\circ$ , isotropic scaling between 0.95 and 1.05, and a 50% horizontal BEV flip. We choose  $\lambda_{\text{reg}} = 2.2$  and  $\lambda_{\text{dir}} = 0.2$  in our configuration. For grid mapping, we choose a grid map size of  $1400 \text{ m} \times 1200 \text{ m}$ , a cell size of  $2.5 \text{ m} \times 2.5 \text{ m}$ , and log-odds increments as  $l_{\text{occ}} = 0.7$  and  $l_{\text{free}} = -0.4$ . The sensor data are transformed into a single, fixed NED reference frame, with its origin set to the geodetic reference point at  $(54.177599, 12.122916, 38.4)$  shown in Figure 10. The navigation filter receives IMU messages at 100 Hz and GNSS messages at 10 Hz such that the filter outputs a localization solution at 100 Hz.

Perception performance is assessed qualitatively on six predefined indicators: environment-representation consistency, which requires that the extrinsic calibration of LiDAR and radar sensors is accurate enough to yield a coherent multi-sensor point cloud; detection consistency, which requires that both

Messin and the remote-operation dinghy are covered by at least one valid 3D box throughout the encounter; class plausibility, meaning that predicted labels match the correct size category; tracking stability, defined as uninterrupted ID persistence in at least 95 % of the frames of an encounter; qualitative range accuracy, whereby the relative pose from the navigation filter must visually coincide with the GNSS-derived pose; and static-mapping plausibility, which demands that the resulting static map is topologically consistent with the underlying chart tiles.

## 6 Experimental Results and Discussion

### 6.1 General Observations

The combined environmental perception and collision avoidance tests with multiple vessels involved were carried out in the port of Rostock from May 7 to 9, 2025. The environmental conditions during the real-world test were fair to moderate. The wind speed was averaged 4.8 m/s, ranging from 2.0 m/s up to 7.5 m/s from the northwest direction, approximately  $300^\circ$ . No relevant current or swell has been noted. The visibility was good.

### 6.2 Performance of the Environmental Perception Module

The results of the extrinsic calibration for the multi-LiDAR radar setup on the Deneb vessel are qualitatively depicted in Figure 11. Before calibration, the point clouds of the LiDAR sensor overlap, yielding a chaotic point cloud. Utilizing a simple reflective warning vest with the SIMPLE-C calibration tool (Figure 11a), and without requiring manual parameter adjustment, results in well-aligned point clouds for all LiDAR sensors on Deneb, mapped to a global NED frame (Figure 11b).

The result of the overall transformation of LiDAR and radar data representing the scene at the pier as a consistent global point cloud can be seen in Figure 11b.

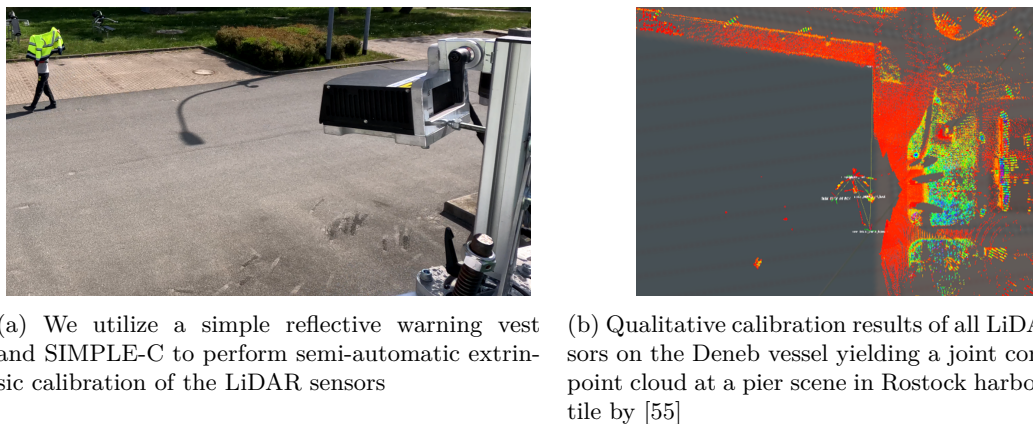


Figure 11: Qualitative results for multi-LiDAR extrinsic calibration using SIMPLE-C tool

Figure 12 illustrates a visual comparison of our navigation filter with GNSS receiver locations for the Deneb vessel. The trajectory estimated by our navigation filter, implemented as an error-state Extended Kalman Filter (ES-EKF), is indicated by the green dots, whereas the GNSS receiver data is marked by the magenta dots.

The result of the LiDAR-based object detection module running on the Deneb can be seen in Figure 13. While Figure 13a shows the camera visualization of the drive-by scenario of Messin and the remote-controlled dinghy, Figure 13b presents the sensor data. At the location of both small vessels, the object detection module detects two objects indicated by the bounding boxes and classifies them as small boats with a classification confidence of 0.84 for Messin and 0.93 for the dinghy.

Figure 14 provides a qualitative overview of the resulting probabilistic occupancy grid mapping (OGM). A grid is overlaid on a map [55], and each cell's state is updated using LiDAR and radar data from the Deneb vessel. Different shades of gray indicate the cell state: light gray denotes free space, dark gray indicates occupied cells, and intermediate tones represent unknown regions. Areas located behind occupied cells or outside the direct sensor line of sight are similarly classified as unknown, as their status cannot be observed directly. Lastly, polygons are extracted from these occupied cells and shown in blue, denoting static objects.

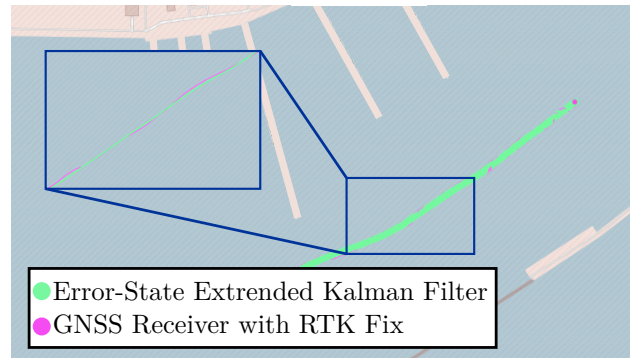
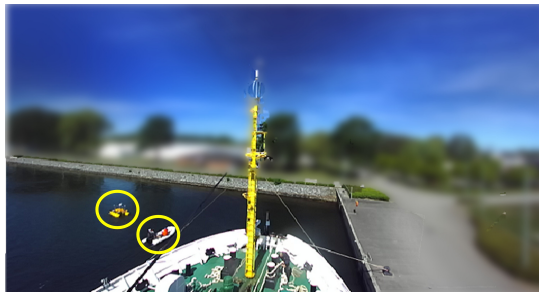
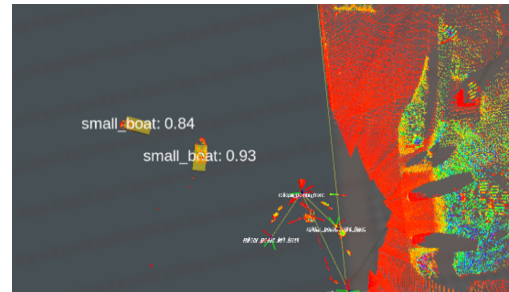


Figure 12: Visual result of the navigation filter (green) and the RTK-supported GNSS localization (magenta) projected on map tiles [55] for the Deneb vessel in the Crossing scenario. The filter is parameterized to compensate for the roll motion of the vessel seen in the sinusoidal trajectory of the GNSS solution.



(a) Camera image of Messin vessel driving near Deneb vessel for testing the object detection module (blurred background)



(b) LiDAR and radar representation of the harbor environment mapped on a satellite map tile [55] with LiDAR-based object detections indicated by bounding boxes with confidence scores. Detected are the USV Messin and the remote control dinghy, correctly classified as small boats.

Figure 13: LiDAR and radar representation of the harbor environment with corresponding camera image. Messin vessel and a dinghy for remote operation drive near the Deneb vessel for object detection tests.

Using the SIMPLE-C calibration tool for the LiDAR sensors and manual measurements for the remaining sensors, we accurately transformed each local sensor frame into a fixed body frame. As seen in the result section, the calibration suffices for the environment-representation consistency. A navigation filter was then employed to determine the vessels' poses, yielding consistent high-frequency estimates. As shown in the results section, the filter produces a trajectory that aligns with the ground-truth GNSS receiver, yielding a sufficient qualitative range accuracy. However, a closer look at Figure 12 shows that the GNSS receiver position follows a sinusoidal pattern. This is due to the roll movement of the vessel. The filter solution compensates for the roll motion by fusing GNSS with IMU data and yields a smooth trajectory presenting the forward motion of the vessel. This is desired as roll motion is a disturbance in vessel control. Due to this roll motion of the GNSS antennas, other performance metrics such as the 2D error are not valid. The navigation solutions for both Deneb and Messin were subsequently made available to the global trajectory planner in each scenario.

In the tested scenarios, we were unable to thoroughly evaluate the LiDAR-based object detection due to sensor range limitations, the low mounting height of the Messin USV's LiDAR, and the minimum safety distance between Messin and Deneb, which prevented them from detecting one another. To further assess the LiDAR-based method, we conducted additional close-proximity drive-by maneuvers, as described in Section 5.2. Based on these tests, the detector successfully recognized both the Messin and the operator dinghy, classifying them correctly as small boats, as seen in the results, indicating a sufficient detection consistency and class plausibility. However, the LiDAR-based object detection module exhibited an inference time of several seconds. One likely reason is the delayed arrival of point cloud data: each

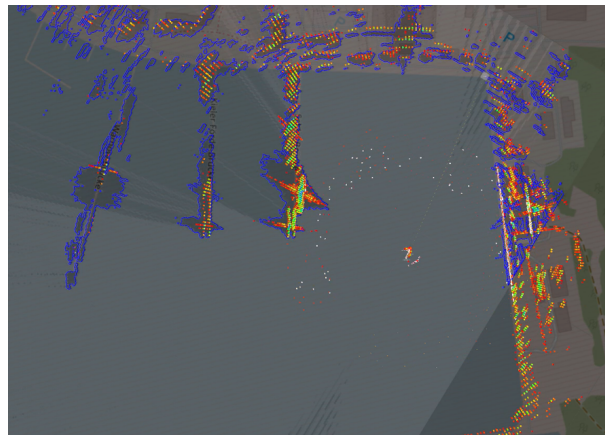


Figure 14: Probabilistic occupancy grid map on radar and LiDAR data in Rostock harbor environment. Shades of grey indicate the probability of occupancy, while blue polygons present extracted static objects from the map.

sensor's output must be aligned using the appropriate transformations. The dynamic transformation between Deneb's body frame and the NED reference point can cause timing issues, and integrating ROS2 with a Python-based detection node may further contribute to inference delays. Figure 13b illustrates this effect, where the point cloud for the actual boat appears slightly offset relative to its bounding box.

Because no objects were detected during the test scenarios, the multi-object tracking algorithm could not be fully assessed. The extended inference time markedly reduced the frequency of available observations, hindering practical tracking. Moreover, since the BeLa vessel was not available and the dinghy lacked both AIS and GNSS data, the scope for multi-object tracking remained limited. Therefore, we could not assess the tracking stability.

The radar could increase the measurement range from about 100 m to 600 m such that the Messin vessel was visible during the scenarios. As there is no object detection module implemented for radar data, the sensor could not be used for long-range 3D object detection.

When evaluating the grid mapping approach, both LiDAR and radar data were fused to generate the map. Although reflections at varying heights can occasionally classify otherwise free cells as unknown due to the birds-eye view projection, this is a rare occurrence in maritime environments unless elevated structures like bridges are present. In such cases, individual adjustments can be made by using only reflections with  $z$ -component below vessel height, and the conservative nature of the algorithm provides a reliable buffer for free-space determination. Due to the radar's high mounting position and its two-dimensional scanning mode, nearby objects may not be accurately detected. Adjusting the LiDAR sensor's pose can help mitigate this issue. Polygon extraction from occupied cells generally captures large clusters and labels them as static obstacles; however, the current implementation tends to register many small polygons instead. Merging closely spaced polygons would yield a more consolidated representation. Despite these details, the grid mapping and polygon extraction method operates in real time, benefiting from a straightforward and efficient design. Overall, the grid mapping results show a sufficient static mapping plausibility.

### 6.3 Performance of the Collision Avoidance Module

Figure 15 shows a sequence of the collision avoidance maneuver during the real-world tests, where the calculated collision-free trajectories are marked in purple. The white circles around the vehicles have a radius of 50 m each, with the origin at the center of gravity of the respective ship, and are only used for visualization for the nautical staff onboard the Deneb. The sequence illustrates the three vehicles following their trajectories, feedback-controlled and successfully executing the evasive maneuver as planned by the central traffic management system. The colors of the past paths represent the respective vehicle speed over ground. It can be seen here that the Messin, as the crossing vehicle, moves more slowly on the trajectory in order to prevent a collision and gives the BeLa and Deneb the right of way in the navigation channel.

In addition to the qualitative results of the avoidance maneuver in the confined area, we analyzed two quantitative performance indicators for scenario execution. The first and decisive measure was how the



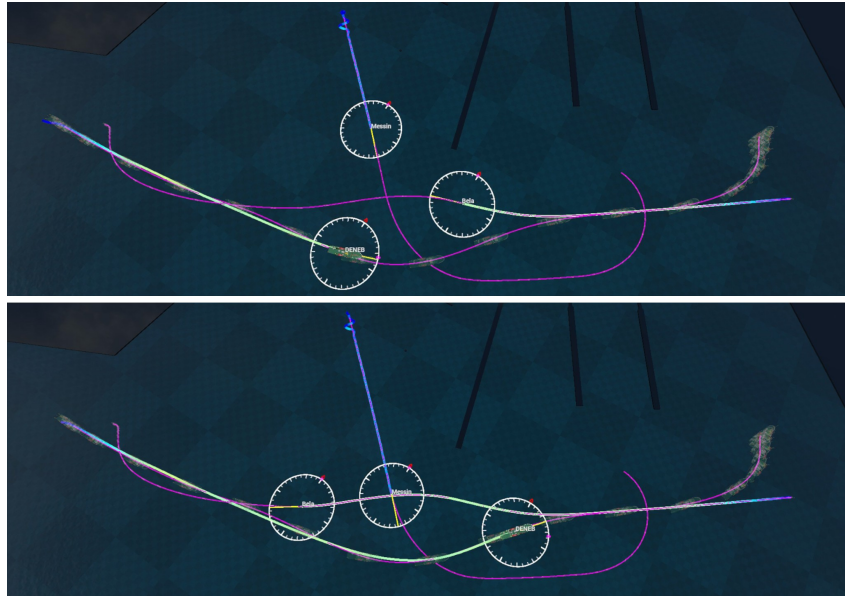


Figure 15: Illustration of the collision avoidance maneuver of three vessels navigating in a head-on and crossing scenario during the real-world tests at two points in time

distances between the vehicles evolved during the scenario, and whether these complied with COLREGs and with the navigators' specifications of good seamanship. The second measure was to what extent the vehicles were able to track the generated trajectories. Figure 16 shows the summarized results of the entire maneuver based on the test run shown in Figure 15.

For further quantification, Figure 16a shows the comparison of the planned (marked with *ref*) and measured (marked with *nav*) positions of the evasion trajectories in the experimental area. To calculate the Euclidean distances between the vehicles involved for further evaluation, the geographical positions have been transformed into the defined local coordinates, depicted in Figure 16b.

Figure 16c shows that the optimization generates valid avoidance trajectories since the minimum distance within the scenario is approximately 55 m between BeLa and Messin at  $t \approx 400$  s. According to the definition, the distance between the two vehicles must not fall below 50 m. The minimum distance between Deneb and BeLa is approximately 92 m and between Deneb and Messin is approximately 168 m.

Finally, Figure 16d shows the comparison of the deviations between the optimized and automatically tracked trajectories. This is important to validate to what extent the vehicles are capable of automatically tracking the given trajectory, as they were generated using a generalized dynamical model for the vehicles involved. The results show that the control deviations remain within a tolerable range. Specifically, the maximum deviation in the case of Deneb is 6.4 m with an average error of 2.6 m. For Messin and BeLa, the values are 11.4 m and 1.3 m as well as 5.6 m and 0.7 m respectively. For Messin in particular, as a vehicle that is not fully actuated, the largest deviations occur at the beginning of the trajectory, moving at a very low speed. After  $t \approx 100$  s, the error decreases abruptly.

#### 6.4 Limitations and Future Research

Given the modular design of our system architecture, limitation concerns the communication between individual modules onboard and between vessels. These communication channels can experience delays or data loss, which may affect critical perception tasks. When the system relies on outdated or incomplete information, it can result in inaccurate trajectory planning or unsafe control actions. For example, the vessel may be directed toward a location that is no longer valid or potentially hazardous. Incomplete information can also reduce the system's overall situational awareness. Hence, in the context of MASS, further research is required to address a range of hazard-related questions, particularly those involving interactions between shore-based infrastructure and vessels. These include vulnerabilities in communication systems, sensor dependencies and the level of redundancy needed to ensure safe operations in various failure scenarios, [27]. Nevertheless, the concept is considered promising, especially for confined and congested waters where vessels are already monitored by established traffic control centres. Such

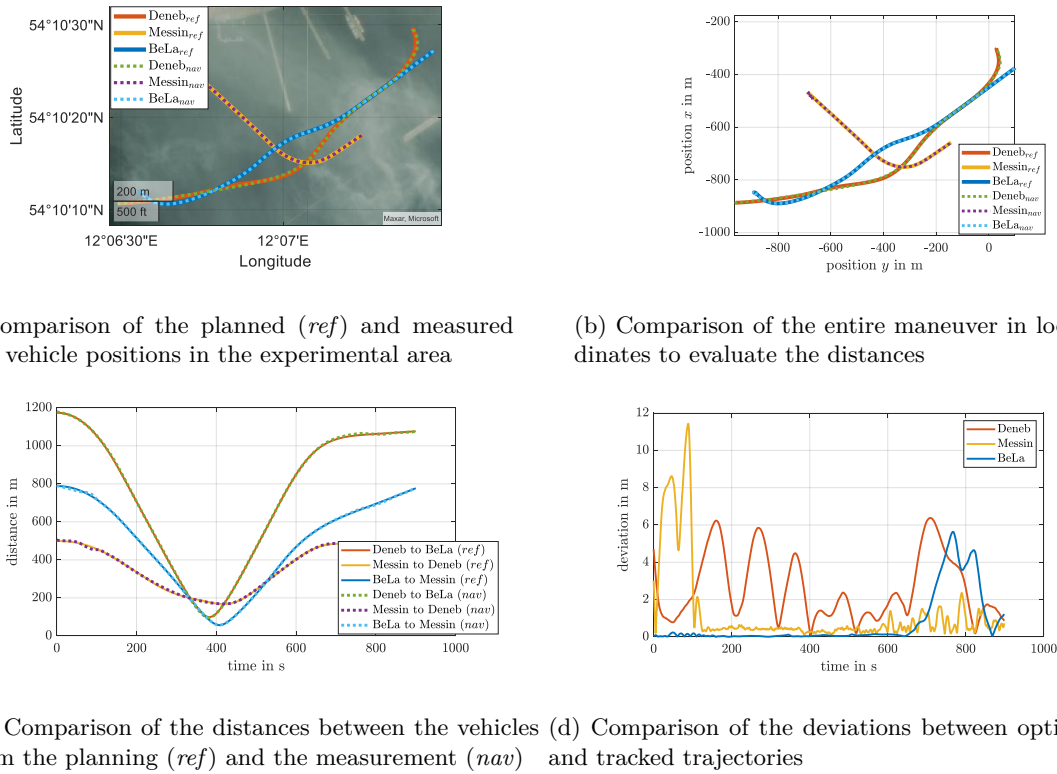


Figure 16: Results of the planned collision avoidance maneuver compared to the real world experiment

environments offer a structured framework within which autonomous transport systems can be more easily integrated and managed.

The object detection capabilities are constrained primarily due to reliance on simulated data for training purposes. Real-world datasets lack sufficient labels necessary for supervised learning, which hampers accurate detection performance. To address this issue, future work could explore self-supervised learning strategies or invest in manual annotation of real-world data. LiDAR technology presents another limitation with its restricted distance view of approximately 100 m. This is insufficient for applications involving port environments that may span much larger areas. Expanding the range or integrating complementary technologies, such as radar, on all vessels may enhance coverage. Radar integration occurred at a late stage in the project, limiting its effective incorporation into the system architecture, such as the object detection module. Fusing radar with LiDAR for enhanced object detection yields promising future research. Further, both perception systems, LiDAR and radar, were unable to operate at their rated measurement distances due to decreased signal-to-noise ratios at extended ranges. Additionally, high inference times for the object detection module precluded assessment of the tracking methods.

The applied optimization approach can require calculation times for complex scenarios of approximately 90 seconds. In terms of the central system architecture under consideration, this is acceptable. However, fast trajectory adaptations based on unknown dynamical objects that are detected at short notice limit the approach. To reduce calculation times, obstacles are approximated as circles, which represents a simplification in terms of the realistic mapping of structures. Finally, unknown dynamical obstacles can currently only be taken into account to a limited extent, as the approach only adapts the initial trajectories or paths in the sense of traffic management and cannot generate fundamentally new trajectories.

In future research, we plan to expand the perception system by integrating camera inputs that provide semantic information, as well as refining radar integration for more reliable long-distance object detection. We also intend to explore new training strategies for the deep learning-based detection module, particularly approaches leveraging unlabeled data sets and additional sensing modalities. Additionally, we aim to improve the detector's inference time such that object tracking can be further investigated. Im-

plementing factor graph-based tracking for multi-modal object detection stands as another key objective, as it promises increased reliability and adaptability under diverse operational conditions.

Although the methods have been tested in reality, they rely on adequate sensor information, especially GNSS position. Without this, automated operation is currently not possible, and the nautical officer has to take over. Consequently, future research will focus on resilience to system faults. In addition to developing alternative navigation procedures, strategies for pre-planning of maneuvers should be adapted, and vehicle guidance and control should be more conservative. The use of a resilient GNC system is intended to compensate for the impact of uncertain navigation information. Based on this, emergency strategies should be developed to keep MASS within a perpetually safe state.

Finally, because GNSS signals are susceptible to jamming and spoofing, we aim to develop a more robust navigation solution that uses GNSS only when reliable, while incorporating camera, LiDAR, and radar odometry for local positioning. Moving toward a factor graph-based framework will enable a flexible structure, global batch optimization, and the possibility of correcting historical data. We will also investigate control strategies suited to GNSS-denied environments, where localization remains uncertain, along with methods to enhance computational efficiency across the system.

## 7 Conclusion

This paper presented the current work of the GN2plus research project as an integrated approach for networked multi-vessel automated maneuvering, combining robust navigation, advanced environment perception, and centralized collision-avoiding trajectory generation. Tests were conducted under real conditions in the port of Rostock using vessels of a relevant size and considering nautical requirements. The results confirm the feasibility of safe and efficient operations in restricted port environments, paving the way for wider adoption of autonomous maritime systems.

## Acknowledgments

This work was supported by the German Federal Ministry for Economic Affairs and Climate Action within project GN2plus under grant 50NA2304 and by the Federal Ministry of Transport within project ModularShipAssist under grant 19H22009C. The basis for the support is a decision of the German Bundestag. Special thanks go to the crew of the research vessel Deneb and the responsible staff of the Federal Maritime and Hydrographic Agency for their great support, the inspiring discussions, and the valuable advice during the experimental trials.

## References

- [1] Eurostat. (2024) Freight transport statistics - modal split. Statistical Office of the European Union (Eurostat). Accessed: 2025-04-29. [Online]. Available: [https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Freight\\_transport\\_statistics\\_-\\_modal\\_split](https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Freight_transport_statistics_-_modal_split)
- [2] United Nations Conference on Trade and Development, “World investment report 2024: Investing in sustainable development,” 2024, accessed: 02.04.2025. [Online]. Available: <https://unctad.org/system/files/official-document/rmt2024.en.pdf>
- [3] Central Commission for the Navigation of the Rhine (CCNR), “Labour market report 2024,” [https://inland-navigation-market.org/wp-content/uploads/2024/03/Livret\\_EN\\_A5\\_Labour-market-2024\\_web\\_bd.pdf](https://inland-navigation-market.org/wp-content/uploads/2024/03/Livret_EN_A5_Labour-market-2024_web_bd.pdf), 2024, accessed on 20 June 2025.
- [4] *Convention on the International Regulations for Preventing Collisions at Sea (COLREGs)*, 1972nd ed., International Maritime Organization (IMO), London, UK, 1972, rules valid as of 15. Juli 1977. [Online]. Available: <https://www.imo.org/en/About/Conventions/Pages/COLREG.aspx>
- [5] S. Thombre, Z. Zhao, H. Ramm-Schmidt, J. M. Vallet García, T. Malkamäki, S. Nikolskiy, T. Hammarberg, H. Nuortie, M. Z. H. Bhuiyan, S. Särkkä, and V. V. Lehtola, “Sensors and ai techniques for situational awareness in autonomous ships: A review,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 64–83, 2022.
- [6] C.-H. Zhou, H.-C. Ku, and S.-H. Lee, “Ship detection of unmanned surface vehicle based on yolov8,” in *IEEE 13th Global Conference on Consumer Electronics (GCCE)*, 2024.
- [7] G. Jocher and contributors, “YOLOv8: Ultralytics yolov8 in pytorch,” <https://github.com/ultralytics/ultralytics>, 2023, accessed: 2025-06-02.

- [8] H. Xu, Y. Yu, J. He, C. Pang, and X. Zhang, “Multi-stage fusion object detection with marine radar and camera in complex maritime context,” *IEEE Transactions on Instrumentation and Measurement*, pp. 1–1, 2025.
- [9] T. Clunie, M. DeFilippo, M. Sacarny, and P. Robinette, “Development of a perception system for an autonomous surface vehicle using monocular camera, lidar, and marine radar,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 14 112–14 119.
- [10] M. Akdağ, P. Solnør, and T. A. Johansen, “Collaborative collision avoidance for Maritime Autonomous Surface Ships: A review,” *Ocean Engineering*, vol. 250, p. 110920, Apr. 2022.
- [11] W. Xie, L. Gang, M. Zhang, T. Liu, and Z. Lan, “Optimizing Multi-Vessel Collision Avoidance Decision Making for Autonomous Surface Vessels: A COLREGs-Compliant Deep Reinforcement Learning Approach,” *Journal of Marine Science and Engineering*, vol. 12, no. 3, pp. 1–29, Feb. 2024.
- [12] K. Aylward, R. Weber, Y. Man, M. Lundh, and S. N. MacKinnon, ““Are You Planning to Follow Your Route?” The Effect of Route Exchange on Decision Making, Trust, and Safety,” *Journal of Marine Science and Engineering*, vol. 8, no. 4, pp. 1–19, Apr. 2020, publisher: MDPI AG.
- [13] Ø. Rødseth, H. Nordahl, L. A. Wennesberg, B. Myhre, and S. Petersen, “Operational Design Domain for Cars Versus Operational Envelope for Ships: Handling Human Capabilities and Fallbacks,” in *Proceedings of the 31st European Safety and Reliability Conference*, 01 2021, pp. 2022–2029.
- [14] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, 2011.
- [15] M. Kurowski, S. Roy, J.-J. Gehrt, R. Damerius, C. Büskens, D. Abel, and T. Jeinsch, “Multi-vehicle Guidance, Navigation and Control towards Autonomous Ship Maneuvering in Confined Waters,” in *Proceedings of the European Control Conference (ECC19)*, Naples, Italy, 2019, pp. 2559–2564.
- [16] R. Zweigel, J. Gehrt, S. Liu, S. Roy, C. Büskens, M. Kurowski, T. Jeinsch, A. Schubert, M. Gluch, O. Simanski, E. Pairet-Garcia, F. Siemer, and D. Abel, “Optimal maneuvering and control of co-operative vehicles as case study for maritime applications within harbors,” in *2019 18th European Control Conference (ECC)*, 2019, pp. 3022–3027.
- [17] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, “CasADi: a software framework for nonlinear optimization and optimal control,” *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, Mar. 2019.
- [18] C. Rethfeldt, A. U. Schubert, R. Damerius, M. Kurowski, and T. Jeinsch, “System Approach for Highly Automated Manoeuvring with Research Vessel DENEb,” *IFAC-PapersOnLine*, vol. 54, no. 16, pp. 153–160, 2021.
- [19] 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)*, 2020, pp. 1–6.
- [20] 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*. the Institute of Navigation, 2021, pp. 4116–4127.
- [21] S. Liu, “Integrity determination and enhancement of multi-sensor navigation systems in maritime applications,” Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, Aachen, 2023.
- [22] J. Lin, P. Diekmann, C.-E. Framing, R. Zweigel, and D. Abel, “Maritime environment perception based on deep learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 15 487–15 497, 2022.
- [23] 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 *2021 European Control Conference (ECC)*, 2021, pp. 2239–2244.

- [24] 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,” *Journal of Physics: Conference Series*, vol. 1357, p. 012019, 10 2019.
- [25] A. U. Schubert, M. Kurowski, R. Damerius, S. Fischer, M. Gluch, M. Baldauf, and T. Jeinsch, “From Manoeuvre Assistance to Manoeuvre Automation,” *Journal of Physics: Conference Series*, vol. 1357, no. 1, p. 012006, Oct. 2019.
- [26] M. Nitsch, J.-J. Gehrt, R. Zweigel, and D. Abel, “Tightly Coupled INS/GNSS Navigation Filter for the Automation of a River Ferry,” *IFAC-PapersOnLine*, vol. 54, no. 16, pp. 139–145, Jan. 2021.
- [27] K. E. Fjortoft and T. Haugen, “Integrated Maritime Autonomous Transport Systems (IMAT),” *Journal of Physics: Conference Series*, vol. 1357, no. 1, p. 012034, Oct. 2019, publisher: IOP Publishing.
- [28] M. Kurowski, A. Haghani, P. Koschorrek, and T. Jeinsch, “Guidance, Navigation and Control of Unmanned Surface Vehicles,” *at – Automatisierungstechnik*, vol. 63, no. 5, pp. 355–367, 2015, publisher: DeGruyter Oldenbourg.
- [29] A. U. Schubert, M. Kurowski, R. Damerius, S. Fischer, M. Gluch, M. Baldauf, and T. Jeinsch, “From manoeuvre assistance to manoeuvre automation,” *Journal of Physics: Conference Series*, vol. 1357, no. 1, p. 012006, oct 2019.
- [30] E. Rentzow, M. Kurowski, H. Korte, and T. Jeinsch, “Navigation System Integration and Evaluation for Precise Underwater Operations,” in *Proceedings of the 2023 DGON Inertial Sensors and Systems (ISS)*, 2023, pp. 1–14.
- [31] R. Damerius and T. Jeinsch, “Real-Time Path Planning for Fully Actuated Autonomous Surface Vehicles,” in *Proceedings of the 30th Mediterranean Conference on Control and Automation (MED)*. Vouliagmeni, Greece: IEEE, Jun. 2022, pp. 508–513.
- [32] M. Kurowski, A. Schubert, and T. Jeinsch, “Automated Maneuvering in Confined Waters using Parameter Space Model and Model-based Control,” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14 495–14 500, 2020.
- [33] T. Hahn, R. Damerius, C. Rethfeldt, A. U. Schubert, M. Kurowski, and T. Jeinsch, “Automated maneuvering using model-based control as key to autonomous shipping,” *at – Automatisierungstechnik*, vol. 70, no. 5, pp. 456–468, 2022, publisher: DeGruyter Oldenbourg.
- [34] M. Nitsch, “Navigation of a miniaturized autonomous underwater vehicle exploring waters under ice,” Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, Aachen, 2024, veröffentlicht auf dem Publikationsserver der RWTH Aachen University; Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2024.
- [35] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, “Pointpillars: Fast encoders for object detection from point clouds,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 689–12 697.
- [36] M. Contributors, “MMDetection3D: OpenMMLab next-generation platform for general 3D object detection,” <https://github.com/open-mmlab/mmdetection3d>, 2020.
- [37] Y. Yan, Y. Mao, and B. Li, “Second: Sparsely embedded convolutional detection,” *Sensors*, vol. 18, no. 10, 2018.
- [38] J. Lin, P. Diekmann, C. Framing, R. Zweigel, and D. Abel, “Dataset for lidar-based maritime perception,” 2022, dataset.
- [39] M. Hahner, D. Dai, A. Liniger, and L. Van Gool, “Quantifying data augmentation for lidar based 3d object detection,” *arXiv preprint arXiv:2004.01643*, 2020.
- [40] P. A. Thomas, J. Barr, B. Balaji, and K. White, “An open source framework for tracking and state estimation (‘stone soup’),” in *Defense + Security*, 2017. [Online]. Available: <https://api.semanticscholar.org/CorpusID:67294683>

- [41] Y. Dodge, Ed., *Mahalanobis Distance*. New York, NY: Springer New York, 2008, pp. 325–326.
- [42] H. W. Kuhn, “The Hungarian Method for the Assignment Problem,” *Naval Research Logistics Quarterly*, vol. 2, no. 1–2, pp. 83–97, March 1955.
- [43] H. Moravec and A. Elfes, “High resolution maps from wide angle sonar,” in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 116–121.
- [44] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. Cambridge, Mass.: MIT Press, 2005.
- [45] W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3d surface construction algorithm,” *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, p. 163–169, Aug. 1987.
- [46] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors, “scikit-image: image processing in python,” *PeerJ*, vol. 2, p. e453, 2014.
- [47] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, I. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. . Contributors, “Scipy 1.0: Fundamental algorithms for scientific computing in python,” *Nature Methods*, vol. 17, pp. 261–272, 2020.
- [48] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, “The quickhull algorithm for convex hulls,” *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469–483, 1996.
- [49] Federal Maritime and Hydrographic Agency (BSH), “Ordinance on the Navigation of Maritime Waterways (SeeSchStrO),” 2023. [Online]. Available: [https://www.bsh.de/DE/PUBLIKATIONEN/\\_Anlagen/Downloads/Nautik\\_und\\_Schifffahrt/Seehandbuecher\\_ueberregional/20005-SeeSchStrO.pdf?\\_\\_blob=publicationFile&v=2](https://www.bsh.de/DE/PUBLIKATIONEN/_Anlagen/Downloads/Nautik_und_Schifffahrt/Seehandbuecher_ueberregional/20005-SeeSchStrO.pdf?__blob=publicationFile&v=2)
- [50] —, “Data sheet of survey, wreck search and research vessel Deneb,” 2023. [Online]. Available: [https://www.bsh.de/DE/Das\\_BSH/BSH-Schiffe/\\_Anlagen/Downloads/Datenblatt\\_DENEb.pdf?\\_\\_blob=publicationFile&v=5](https://www.bsh.de/DE/Das_BSH/BSH-Schiffe/_Anlagen/Downloads/Datenblatt_DENEb.pdf?__blob=publicationFile&v=5)
- [51] J. R. Marx, R. Damerius, and T. Jeinsch, “Linearized Model Predictive Control with Offset-freeness for Trajectory Tracking on Inland Vessels,” in *2023 31st Mediterranean Conference on Control and Automation (MED)*. Limassol, Cyprus: IEEE, Jun. 2023, pp. 692–697.
- [52] M. Kurowski, J. Thal, R. Damerius, H. Korte, and T. Jeinsch, “Automated Survey in Very Shallow Water using an Unmanned Surface Vehicle,” *IFAC-PapersOnLine*, vol. 52, no. 21, pp. 146–151, 2019.
- [53] R. Damerius, T. Hahn, I. Karez, A. Schubert, B. Kolewe, and T. Jeinsch, “Guidance, Navigation and Control of Couplable Unmanned Surface Vehicles,” in *OCEANS 2024 - Singapore*. Singapore, Singapore: IEEE, Apr. 2024, pp. 1–8.
- [54] L. N. Smith and N. Topin, “Super-convergence: very fast training of neural networks using large learning rates,” in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications*, T. Pham, Ed., vol. 11006, International Society for Optics and Photonics. SPIE, 2019, p. 1100612.
- [55] OpenStreetMap contributors, “Planet dump retrieved from <https://planet.osm.org>,” <https://www.openstreetmap.org>, 2017.