2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Autonomous Docking Using LiDAR-Based Tracking and Adaptive Pose Selection: Closed-Loop Sea Trials*

August J. Fors,Simon J. N. Lexau,Edmund F. Brekke,Miguel Hinostroza,Anastasios M. Lekkas,Morten Breivik

***Abstract*—This paper presents a fully integrated autonomous docking system validated through closed-loop sea trials on the milliAmpere1 research ferry operating in a live maritime harbour with moving vessels. Real harbour environments require continuous situational awareness and adaptive decision-making under dynamic traffic conditions. The proposed architecture combines cartographic land masking, LiDAR-based clustering, probabilistic multi-target tracking (JIPDA), dynamic footprint estimation, adaptive docking pose selection, and real-time path replanning within a finite state machine framework. Rather than introducing new algorithms, the contribution lies in system-level integration and operational validation of a complete perception-to-control pipeline under realistic maritime constraints. The system is demonstrated in multiple closed-loop experiments including collision avoidance and adaptive docking with moving obstacles. Results highlight both performance characteristics and practical deployment considerations, including runtime behaviour, sensor limitations, and integration trade-offs. The work provides empirical evidence that robust autonomous docking in dynamic harbour environments can be achieved through carefully engineered integration of established methods.**

***Index Terms*—Autonomous docking, LiDAR-based tracking, collision avoidance, field experiment, closed-loop control, marine robotics.**

I. Introduction

Autonomous surface vessels represent a transformative technology for maritime transportation, driven by the need to address crew shortages, improve safety, and optimize operational efficiency. Recent focus on autonomous ships has sparked intensive research into dock-to-dock operations, with automated docking being particularly challenging. The complexity of harbour environments, with their mix of static infrastructure and dynamic obstacles, demands sophisticated situational awareness systems.

Current autonomous docking approaches typically rely on predefined dock geometries, visual markers, or simplified environmental representations. The Docking Characteristic Index (DCI) analysis by Lexau et al. [1] demonstrates that current closed-loop autonomous docking systems have notable limitations in dynamic obstacle handling, obstacle position estimation, and sensor integration. These limitations motivate the need for more comprehensive SITAW systems capable of real-time environmental adaptation.

Fig. 1: The milliAmpere1 ferry.

Recent work has demonstrated LiDAR-based docking in controlled scenarios [2] and tracking-based collision avoidance in open waters [3]. However, work showing integrated SITAW systems capable of both adaptive docking pose selection and real-time collision avoidance in cluttered harbour environments have not been reported to the authors' knowledge. This work demonstrates the first integrated system combining real-time obstacle tracking, adaptive spatial reasoning, and closed-loop control for comprehensive autonomous harbour operations validated in complex maritime conditions.

The presented system employs LiDAR-based Situational Awereness (SITAW) system for closed-loop docking and near-shore manoeuvring. The approach combines precise land masking, real-time obstacle tracking, and dynamic assessment of available docking spaces to enable autonomous vessels to identify free docking areas and determine safe docking poses in real-time as harbour conditions change.

The proposed system was validated using real-world data from the milliAmpere1 (MA1) research ferry ([4], [5]) and experimentally verified in closed-loop operation with a control system, demonstrating effectiveness for both collision avoidance and docking. The main contributions are:

- **Closed-loop validation of a full-stack autonomous docking system** on an operational ferry in a live harbour

All authors are affiliated with the Centre for Research-based Innovation (SFI) AutoShip: Autonomous Ships for Safe and Sustainable Operations, Department of Engineering Cybernetics - Norwegian University of Science and Technology (NTNU), NO-7491 Trondheim, Norway. forsaugust@gmail.com, {simon.lexau, edmund.brekke, miguel.hinostroza, anastasios.lekkas, morten.breivik}@ntnu.no

* This work was supported by the Research Council of Norway under Project 309230 (SFI AutoShip).

environment with moving vessels. To our knowledge, this represents the first integrated LiDAR-centric perception-to-control pipeline demonstrated under real maritime traffic conditions.

- **A system-level architecture for dynamic harbour docking**, combining land masking using authoritative cartographic data, probabilistic multi-target tracking, dynamic footprint extension, adaptive docking pose selection, and real-time path replanning.
- **Adaptive dockability assessment under dynamic obstacles**, where available docking regions are computed online based on tracked obstacle footprints and vessel geometry.
- **Operational insights from real-world deployment**, including practical limitations of LiDAR resolution, tracking stability, runtime constraints, and software integration challenges in safety-critical maritime systems.

Unlike algorithm-centric contributions that focus on isolated perception or planning modules, this work emphasizes system-level integration and operational validation. The novelty lies not in proposing new clustering, tracking, or planning algorithms in isolation, but in demonstrating how established methods can be composed into a robust, closed-loop docking architecture validated under real maritime traffic conditions.

II. Background

A. Sensor Technologies for Maritime SITAW

Maritime situational awareness systems rely on various sensor modalities, each with distinct advantages and limitations. Radar has proven effective for long-range detection and tracking in open waters ([6], [7]), but suffers from poor resolution at short ranges typical in harbour environments ([8]). Camera-based systems provide rich visual information and have benefited from advances in computer vision ([9]–[11]), but performance degrades significantly in adverse weather conditions and requires additional processing for range estimation ([12]). LiDAR sensors offer advantages for harbour operations, providing accurate range measurements from centimeters to hundreds of meters while maintaining effectiveness in fog and darkness. However, LiDAR data requires sophisticated processing to distinguish between different object types, as the sensor provides geometric rather than semantic information.

B. Detection and Tracking Methods

The detection stage forms the foundation of any tracking system. For range-bearing sensors like LiDAR and radar, clustering algorithms are the standard approach for grouping sensor returns into detections ([13], [14]). While convolutional neural networks have been explored for LiDAR-based maritime detection ([15], [16]), real-world effectiveness remains inconclusive compared to traditional clustering methods. Multi-target tracking in maritime environments presents unique challenges due to complex

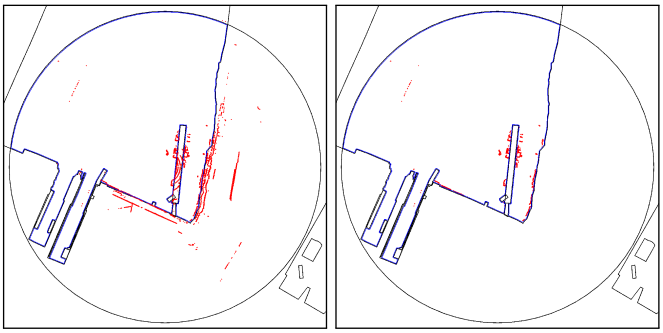

Fig. 2: Unfiltered (left) and filtered (right) point clouds in OpenCV image.

motion patterns and measurement uncertainties. The Joint Integrated Probabilistic Data Association (JIPDA) tracker [17] has demonstrated reliable performance in maritime applications using both LiDAR and radar data ([13], [18]). Current state-of-the-art approaches include Poisson Multi-Bernoulli Mixture (PMBM) trackers ([3], [19]), though these require greater computational resources.

C. Autonomous Docking Approaches

Existing autonomous docking methods can be categorized into several approaches. Marker-based systems rely on visual or electronic beacons to identify predefined docking locations ([20]–[22]). Geometry-based methods detect dock structures using sensor data, such as LiDAR line detection or stereo vision with stixel representations ([23], [24]). Map-based approaches combine sensor data with prior environmental knowledge to identify manoeuvrable areas ([2]). However, these methods typically assume static environments and struggle with dynamic obstacles such as other vessels entering or leaving docking areas. Few existing systems can adaptively update available docking spaces based on real-time obstacle tracking.

D. Closed-Loop Maritime Systems

Several research efforts have demonstrated SITAW systems operating in closed-loop with vessel control. Hem et al. validated their tracking system for collision avoidance in open waters ([3]), while Martinsen et al. developed a control-focused approach using LiDAR and maps for docking ([2]). The Roboat project represents a successful implementation of autonomous water taxis with integrated collision avoidance and control ([25], [26]). However, comprehensive systems that handle both collision avoidance and adaptive docking in dynamic harbour environments remain limited.

III. Methodology

This section describes the tracking pipeline, adaptive pose selection, and motion control systems that were developed for the COLAV and docking experiments.

A. Land Masking

The first step of the tracking pipeline is land masking. For this step, we used a map of the experiment site at Nyhavna basin in Trondheim, Norway, provided by the Norwegian

Algorithm 1 Land Masking Algorithm

```
input_pc
output_pc ← empty list
image ← OpenCV image of specified size
Draw circle around the vessel in image
Draw map contours in image
Dilate image
contours ← find_contours(image)
con_idx ← 0
for idx, contour in contours do
    if vessel_pos in contour then
        con_idx ← idx        ▷ The index of the smallest contour that contains the MA2 position
    end if
end for
inside_contours ← list(con_idx)
for idx, contour in contours do
    if point_on contour inside contours[con_idx] then
        append idx to inside_contours    ▷ All contours inside MA1's sea contour, normally floating docks.
    end if
end for
for point in input_pc do
    pixel ← point_to_pixel(point)
    if pixel in contours[con_idx] then    ▷ Check if point is in main contour
        for idx in inside_contours do
            if pixel not in contours[idx] then
                point ← pixel_to_point(pixel)
                append point to output_pc
            end if
        end for
    end if
end for
return output_pc
```

Mapping Authorities. The map provides detailed information of the contours of the harbour, including floating docks. In order to scrutinize this information for land masking, we convert the map to an OpenCV image, which allows us to readily extract the contours of the harbour environment ([27]). The intermediate results can be seen in Figure 2, where the map contours are black-, the LiDAR points red- and the OpenCV contours are blue lines. The LiDAR points that fall within land contours are subsequently removed. The full algorithm is described in Algorithm 1. With the chosen resolution of the OpenCV image, the method enables land masking on a decimeter-level of precision. The resolution can be increased, however, this was not deemed necessary for the purpose of the experiments presented here.

A challenge of relying on maps for maritime control systems, is the varying water levels with tides. In our scenario, we are mostly concerned with docking to a floating dock, and performing COLAV in safe distance to any varying shore line.

B. Target Tracking System

The proposed tracking framework uses the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) clustering scheme to generate detections from the land masked LiDAR points ([28]). By following this approach, we seek to minimize the amount of false tracks. The detections are passed into a Joint Integrated Probabilistic Data Association (JIPDA) tracker that estimates the position and velocity. The outputs from JIPDA are commonly referred to as tracks, or obstacles. Furthermore, the footprints of the tracks can be obtained by taking the convex hull of the clusters, which yields a rough estimate of the obstacles' extent ([19]).

For dynamic obstacles, tracks with a speed greater than a tunable threshold had their footprints extended in the direction of their velocity vectors. The extended footprint represents the predicted swept area that the obstacle could occupy over the next 20 seconds based on its current trajectory. This 20-second horizon provides adequate safety margins for typical harbour manoeuvring speeds. An example of a dynamic obstacle with elongated footprint can be seen in Figure 3.

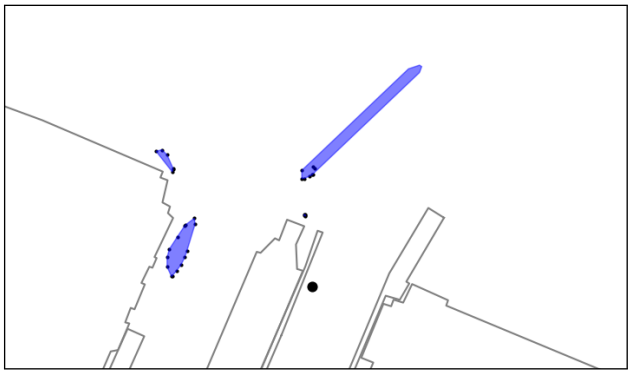

Fig. 3: Elongated footprint.

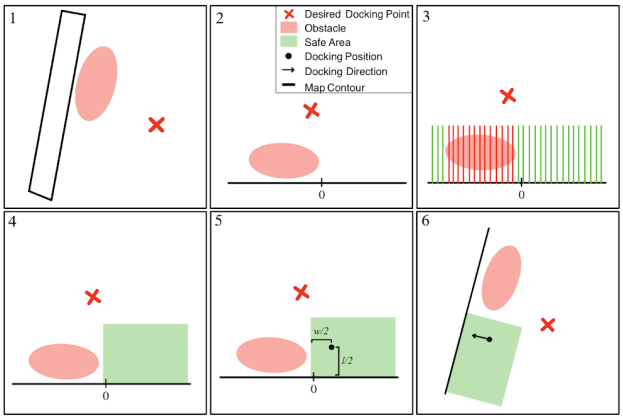

Fig. 4: Method for finding safe area and docking pose.

C. Adaptive Docking Pose Selection

Given a rough docking point from the operator, the adaptive docking pose system identifies safe, convex dock regions and computes a precise docking pose when the vessel enters sensor range. The method is illustrated in Figure 4 and described as follows:

- Step 1 - When the vessel is within a tunable radius of the desired docking point (marked as the red "X"), the system identifies nearby obstacles.

- Step 2 - The system finds the map contour closest to the desired docking point and transforms coordinates to the contour reference frame.
- Step 3 - The system generates lines at regular increments perpendicular to the map contour, each with the length of MA1. Areas where lines intersect with obstacles are marked as unsafe.
- Step 4 - The system identifies safe regions and filters out any region smaller than MA1's width plus safety margin.
- Step 5 - The system selects the docking position in the closest safe area, accounting for MA1's dimensions. $l/2$ and $w/2$ represent half the length and half the width of MA1, respectively.
- Step 6 - The system transforms back to the world frame and sets the heading angle perpendicular to the map contour.

Land masking enables the tracker to maintain confident tracks on docked vessels, as LiDAR points from these vessels will not be clustered together with the dock structure. With tracks on the obstacles, the system responds quickly when vessels undock or otherwise obstruct the planned route.

D. Motion Control System

The motion control system consists of a path planner, a guidance law, a reference filter, a low-level control system, and thrust allocation. The control- and thrust allocation systems were previously developed by [29].

The path planning system employs A* search on a uniform grid to generate an initial collision-free path through obstacles from the current position of the vessel, to any desired point in the basin. The raw waypoint sequence is smoothed using cubic B-splines with tangent-continuous concatenation, where each curve segment maintains derivative continuity at junction points. An iterative segment-merging approach tests increasingly long waypoint jumps to minimize path complexity while respecting safety constraints. The smoother incorporates intermediate control points and safety buffering around obstacles to ensure collision avoidance. This hybrid approach balances computational efficiency with smooth, navigable trajectories.

The adaptive line-of-sight (ALOS) guidance law, proposed by Fossen in [30], was used for path following in this work. Furthermore, a nonlinear reference filter was employed to generate smooth, continuous, reference signals to the control system. A dynamic positioning (DP) system was used for low-level control.

IV. Experimental Setup

A. Research Vessel

The research vessel is the milliAmpere1 stainless steel autonomous ferry prototype, see Fig. 1. From 2017 it has been used as testing platform at the Norwegian University of Science and Technology (NTNU). The main characteristics of MA1 are given in Table I.

TABLE I: Technical specifications for milliAmpere1.

MA1	**Particulars**
Length (LOA)	5.0 m
Beam	2.8 m
Draught	0.2 m
Total draught (w/thrusters)	0.6 m
Air draught	3.3 m
Light weight	1.8 tons
Max passengers	6
Propulsion	4 azimuth thrusters (1.6 kW each)
Operation speed	3 knots
Max speed	5 knots
Energy	Electric, 24V DC system
Batteries	Lead-Acid VRL, 24 kWh
Navigation sensors	RTK GNSS-compass, IMU
SITAW sensors	IR/EO cameras, X-band radar, anemometer, LiDAR

Fig. 5: Obstacle vessel used in the closed loop experiments. The vessel was augmented to compensate for limited 16-ray LiDAR resolution.

B. Closed Loop Experiments

The experiments were designed with the primary goal of testing the perception system as part of a real-time control system. A Buster XL vessel, see Figure 5, was used as an obstacle vessel. To overcome limitations with the 16-beam LiDAR sensor, which has fewer vertical scan lines than current-generation sensors, the vessel's height and visible area were increased in a pragmatic manner.

The closed-loop experiments are illustrated in Figure 6. Experiment 0 was included to verify that the integrated system worked properly. In experiment 1, the target tracker is evaluated in a standard crossing collision avoidance scenario. In accordance with COLREGS, the research vessel is the give-way vessel as the obstacle vessel approaches from the starboard side. Experiments 2 and 3 are pure docking experiments. Experiment 2 evaluates the docking pose selection scheme. The final experiment extends experiment 2 by involving a dynamic obstacle to assess the scheme's adaptability.

V. Closed Loop Implementation

The system architecture is based on a finite state machine (FSM) that integrates the target tracker, adaptive docking pose scheme, path planner, and motion control system with the necessary logic for collision avoidance and state switching. Since the research vessel relies on the

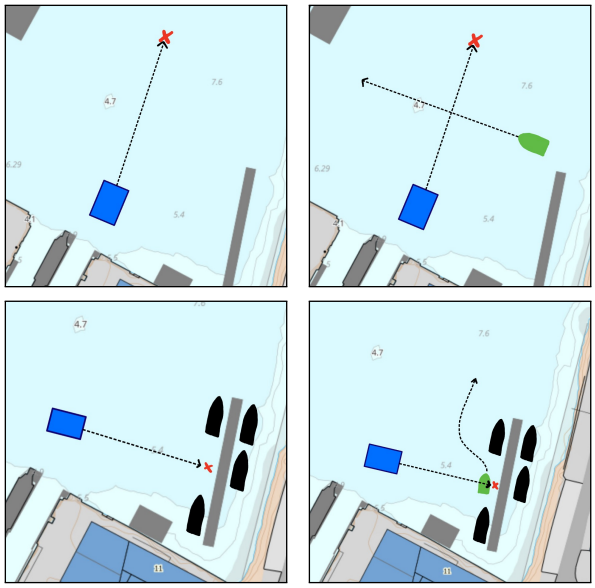

Fig. 6: Closed loop tests, from top left to bottom right, 0, 1, 2 and 3.

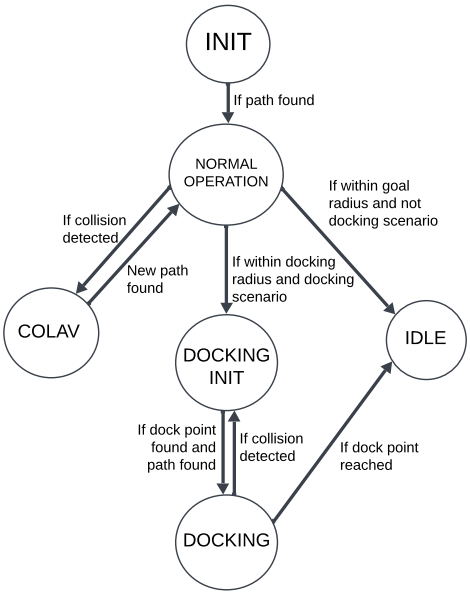

Fig. 7: State machine diagram.

Robot Operating System (ROS) for internal and external communications, the integrated system was implemented using ROS nodes with topics and service calls for message passing between components.

A. Finite State Machine

The workflow of the integrated system is represented by the Unified Modeling Language (UML) state diagram in Figure 7. All experiments start at the *INIT* state, transitioning to *NORMAL OPERATION* as soon as the path finder outputs a safe and valid path. The *IDLE* state is the end state, which is reached when the experiment has concluded. In this state, the DP system is used for station-keeping. If a potential future collision is detected while in *NORMAL OPERATION*, the FSM will transition to *COLAV*. If the current test is a docking experiment, the FSM will transition from *NORMAL OPERATION* to *DOCKING INIT* when the vessel is within a certain radius of the initial docking point. In the *DOCKING INIT* state, first the final docking pose and the safe convex regions will be computed. Next, the path planner finds a safe, smooth path from the vessel's current position to the final docking pose. If the final dock point and path are valid, the FSM will transition to *DOCKING*. If the path becomes obstructed during *DOCKING*, the FSM will transition back to *DOCKING INIT*. Upon reaching the final dock pose, the FSM transitions to *IDLE*.

As such, experiment 0 will have a predictable state flow: *INIT* → *NORMAL OPERATION* → *IDLE*.

The state flow of experiment 1 will look something like: *INIT* → *NORMAL OPERATION* → *COLAV* → *NORMAL OPERATION* → *IDLE*.

Furthermore, we expect the flow of experiment 2 to be: *INIT* → *NORMAL OPERATION* → *DOCKING INIT* → *DOCKING* → *IDLE*.

Finally, experiment 3 should produce the following state transition flow:

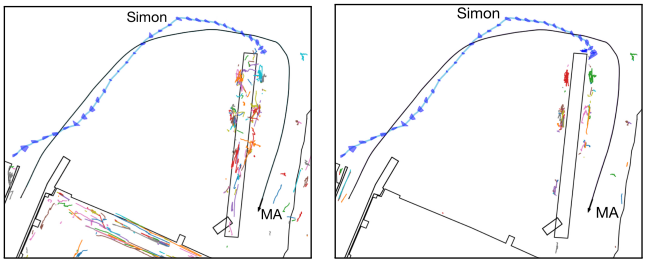

Fig. 8: Open-loop tracking results. Unfiltered (left) vs. filtered (right), showing improved obstacle detection.

INIT → *NORMAL OPERATION* → *DOCKING INIT* → *DOCKING* → *DOCKING INIT* → *DOCKING* → *IDLE*.

VI. Results

A. Open-Loop Experiments

The following results demonstrate the effectiveness of the proposed land masking scheme using data collected during open-loop experiments with the MA1 ferry in Trondheim harbour. Figure 8 shows tracking results with and without land masking applied to LiDAR data. The filtered case produces substantially fewer false tracks and clearly identifies docked vessels near the floating dock, while the unfiltered case generates numerous spurious tracks from dock infrastructure and static objects.

Without ground truth measurements for tracked obstacles, evaluation relied on qualitative analysis of system behaviour, which effectively demonstrated system capabilities in realistic maritime scenarios.

Figure 9 demonstrates the impact of land masking on docking space identification. Without proper filtering, the system incorrectly clusters docked vessels with dock infrastructure, preventing accurate identification of available docking areas. The filtered case clearly separates obstacles

from infrastructure, enabling precise assessment of safe docking regions.

B. Closed-Loop Experiments

The closed-loop experiments demonstrate the integrated system's performance in real maritime conditions. In all figures, the black dot and line represent MA1's position and trajectory, the red line shows the planned path, blue polygons indicate tracked obstacle footprints, and the red "x" marks the target location.

All experiments were held under calm weather conditions with little wind and clear visibility.

Experiment 0, intended as a basic integration test without obstacles, encountered an unexpected vessel that triggered the collision avoidance logic. Figure 10 shows the system successfully replanning the path as the obstacle's footprint intersects the original trajectory. The obstacle appears to cross the planned path because the collision avoidance procedure activates and replans while the obstacle passes through the area. Post-experiment analysis revealed a software bug that caused approximately 10-second delays in path replanning due to unnecessary plot generation processes.

Experiment 1 demonstrates the system's response to a deliberate crossing scenario. Figure 11 shows successful path replanning as the obstacle vessel crosses from left to right and back. The curved trajectory of MA1 results from the aforementioned software bug, which caused the system to withhold control inputs during plot generation, leading to vessel drift until new paths were computed.

The docking capability is demonstrated in Experiment 2, shown in Figure 12. Following correction of the software bug identified in earlier experiments, the system initially plans a path to the desired docking point, then calculates available docking area and determines a safe pose before planning the final approach path. Figure 12 shows that the vessel follows the initially planned path, calculates docking pose and the corresponding path and is able to reach the planned position.

Experiment 3 extends the docking scenario with dynamic obstacles. Figure 13 demonstrates the system's adaptability, initially planning for the original pose but successfully recalculating a different docking location when the obstacle creates a conflict. The results show the system initially calculating the docking pose to the available spot at the south end of the dock. Immediately after planning the path the system detects a collision due to the instability of a track giving it a false velocity leading to the extension of the footprint. This makes the system find a new docking position and at this time the obstacle vessel has moved along and a closed docking position is available. The system then plans a path to this position and executes the manoeuvrer.

C. Runtime

The SITAW system operates with two main computational components: continuous perception and tracking, and event-driven planning algorithms. The main perception loop, indicated by track publication frequency, operates at an average of 2.6 Hz. Event-driven components include path planning during collision avoidance and docking area calculation during docking initialization. Due to the software bug identified in experiments 0 and 1, where unnecessary plot generation caused significant delays, runtime analysis from these experiments is not representative. Simulation studies indicate the path planner operates efficiently with runtimes between 0.1 to 0.5 seconds, and in the experiments the time from the system entering path planning to exiting, was on average 1.2 seconds.

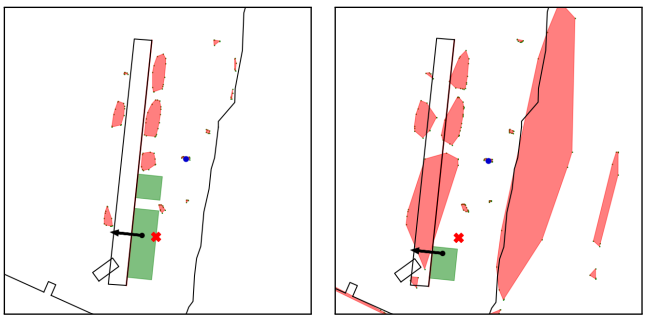

Fig. 9: Docking region identification: filtered (left) vs. unfiltered (right) showing accurate area detection.

VII. Discussion

The closed-loop experiments successfully validated the integrated perception and control system for autonomous harbor operations. However, the real-world implementation revealed important considerations regarding sensor limitations, system performance characteristics, and control integration for future autonomous maritime systems.

A. Hardware Constraints

The MA1 ferry's sensor configuration presents several limitations. The LiDAR's 16-laser vertical resolution occasionally produces unstable detections when horizontal planes misalign with obstacle hulls, affecting both detection reliability and footprint estimation accuracy. This limitation particularly impacts velocity estimation for dynamic obstacles, leading to noisy footprint extensions that can complicate docking pose selection in the presence of moving vessels.

B. System Performance

The perception loop operates at 2.6 Hz, sufficient for experimental scenarios but limiting potential tracking performance compared to higher update rates. During path replanning, the system transitions to IDLE state where dynamic positioning maintains vessel position through station keeping. The current implementation does not replan optimal paths after obstacles clear, instead maintaining collision avoidance paths until destination - adequate for demonstration but representing future areas of improvement.

C. Control System Integration

The adaptive docking pose selection system generates safe, convex docking regions that provide valuable spatial information for autonomous docking operations. However,

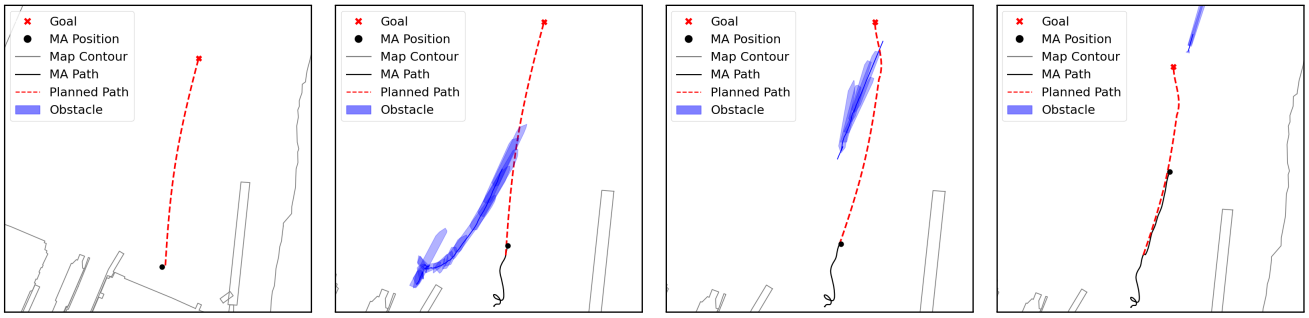

Fig. 10: Evolution of tracks, path and position in experiment 0. While intended as a basic system integration test, an unexpected vessel entry triggered collision avoidance, demonstrating real-world system capabilities.

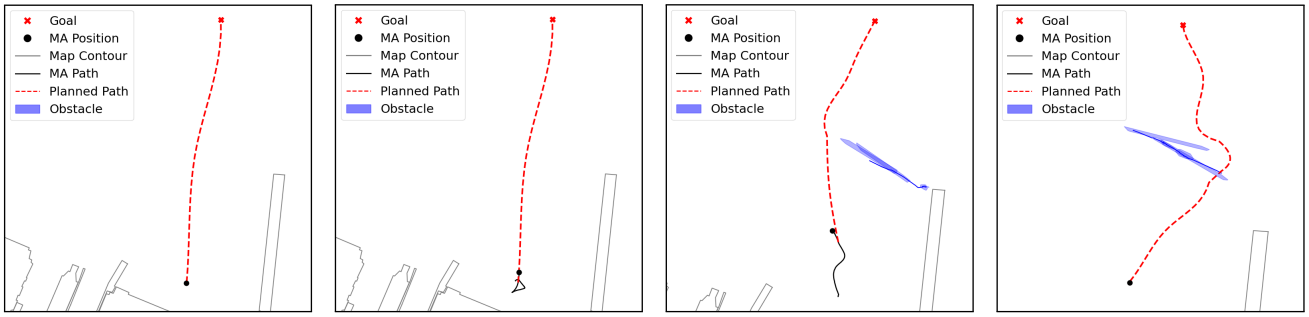

Fig. 11: Evolution of tracks, path and position in experiment 1.

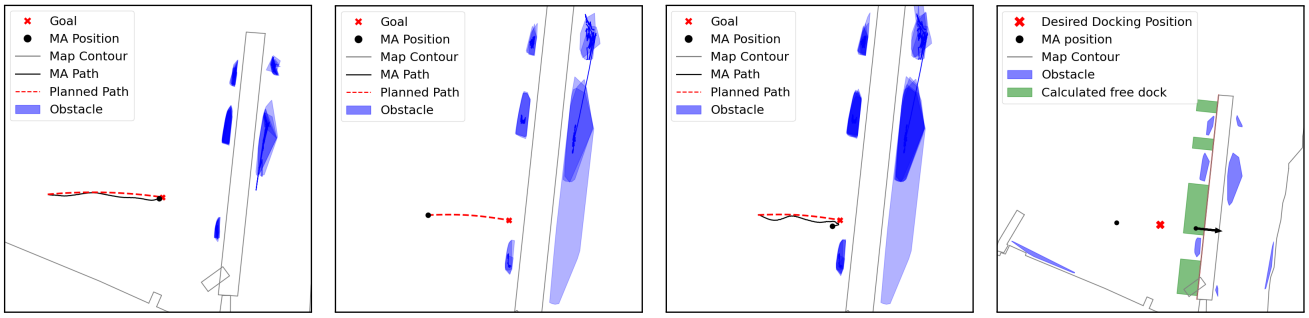

Fig. 12: Evolution of tracks, path and position in experiment 2. The rightmost figure depicts the calculated dockable areas.

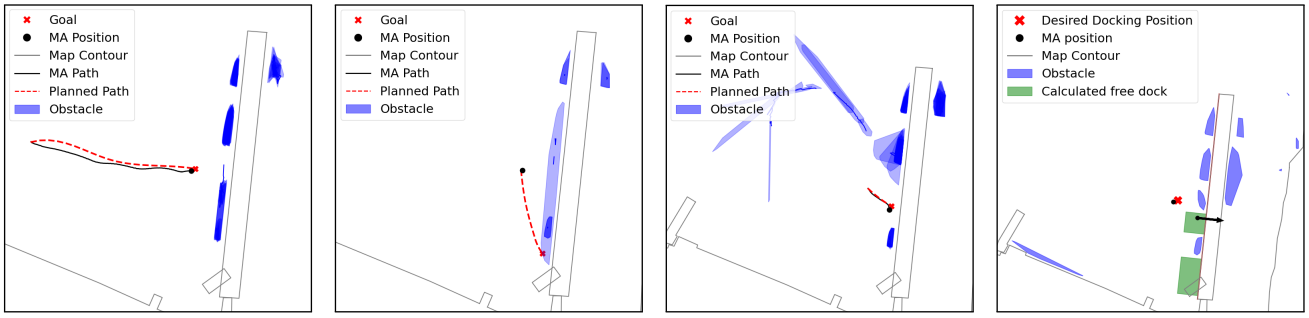

Fig. 13: Evolution of tracks, path and position in experiment 3. The path is recalculated between the second and third figures. The rightmost figure depicts the final calculated dockable areas.

these safe, convex docking regions were not directly integrated with the control system in this implementation. Future work could incorporate these regions as spatial constraints for Model Predictive Control systems, potentially improving docking precision and safety.

VIII. Conclusion

This paper presents an integrated perception, tracking, and control system for autonomous surface vessels operating in near-shore environments, focusing on docking and collision

avoidance. The system combines LiDAR sensing with precise land masking to filter irrelevant data points, enabling reliable detection and tracking of obstacles. This allows the system to reliable perform collision avoidance and dynamically assess available docking areas. Previous autonomous docking systems, such as reviewed by Lexau et al. ([1]), typically assume static harbour conditions and use predefined docking poses. The proposed system, however, adapts in real-time to environmental changes and obstacle movement. Validation using real-world data from the milliAmpere1 research ferry and closed-loop experiments demonstrates effectiveness for both collision avoidance and autonomous docking in dynamic conditions.

The primary contribution of this work is empirical validation of a complete autonomous docking pipeline in a live maritime harbour environment under dynamic traffic conditions. These results provide a foundation for future development of robust autonomous maritime operations in complex near-shore environments.

REFERENCES

[1] S. Lexau, M. Breivik, and A. Lekkas, "Automated docking for marine surface vessels - a survey," *IEEE Access*, vol. 11, pp. 132 324–132 367, 01 2023.

[2] A. B. Martinsen, G. Bitar, A. M. Lekkas, and S. Gros, "Optimization-based automatic docking and berthing of asvs using exteroceptive sensors: Theory and experiments," *IEEE Access*, vol. 8, pp. 204 974–204 986, 2020.

[3] A. H. Gullikstad, M. Baerveldt, and E. F. Brekke, "Pmbm filtering with fusion of target-provided and exteroceptive measurements: Applications to maritime point and extended object tracking," *IEEE Access*, vol. 12, pp. 55 404–55 423, 2024.

[4] E. F. Brekke, E. Eide, B. H. Eriksen, E. F. Wilthil, M. Breivik, E. Skjellaug, Ø. K. Helgesen, A. M. Lekkas, A. B. Martinsen, E. H. Thyri, T. Torben, E. Veitch, O. A. Alsos, and T. A. Johansen, "milliampere: An autonomous ferry prototype," *Journal of Physics: Conference Series*, vol. 2311, no. 1, p. 012029, jul 2022.

[5] M. A. Hinostroza, E. Eide, E. F. Brekke, M. Breivik, A. M. Lekkas, R. Skjetne, T. Håland Bryne, A. Gusev, and A. Gudahl Tufte, "Milliampere1 autonomous ferry prototype: Hardware and software," in *International Conference on Offshore Mechanics and Arctic Engineering*, vol. 88926. American Society of Mechanical Engineers, 2025, p. V003T06A051.

[6] E. F. Brekke, A. H. Gullikstad, and L. C. N. Tokle, "Multitarget tracking with multiple models and visibility: Derivation and verification on maritime radar data," *IEEE Journal of Oceanic Engineering*, vol. 46, no. 4, pp. 1272–1287, Jul. 2021.

[7] E. F. Wilthil, A. L. Flåten, and E. F. Brekke, "A target tracking system for asv collision avoidance based on the pdaf," *Sensing and Control for Autonomous Vehicles*, vol. 474, pp. 269–288, May 2017.

[8] Ø. K. Helgesen, K. Vasstein, E. F. Brekke, and A. Stahl, "Heterogeneous multi-sensor tracking for an autonomous surface vehicle in a littoral environment," *Ocean Engineering*, vol. 252, p. 111168, May 2022.

[9] R. Yoneyama and Y. Dake, "Vision-based maritime object detection covering far and tiny obstacles," *IFAC-PapersOnLine*, vol. 55, no. 31, pp. 210–215, 2022.

[10] L. Niu, Y. Fan, T. Liu, and Q. Han, "Momt: A maritime real-time visual multiobject tracking algorithm based on unmanned surface vehicles," *IEEE Sensors Journal*, vol. 24, no. 21, pp. 35 429–35 447, 2024.

[11] H. Park, S. Ham, T. Kim, and D. An, "Object recognition and tracking in moving videos for maritime autonomous surface ships," *Journal of Marine Science and Engineering*, vol. 10, p. 841, 06 2022.

[12] X. Chen, X. Xu, Y. Yang, Y. Huang, J. Chen, and Y. Yan, "Visual ship tracking via a hybrid kernelized correlation filter and anomaly cleansing framework," *Applied Ocean Research*, vol. 106, p. 102455, Jan. 2021.

[13] E. F. Brekke, A. H. Gullikstad, and L. C. N. Tokle, "The vimmjipda: Hybrid state formulation and verification on maritime radar benchmark data," *Global Oceans*, pp. 1–5, Oct. 2020.

[14] H. Hilmarsen, N. Dalhaug, T. A. Nygård, E. F. Brekke, R. Mester, and A. Stahl, "Maritime tracking-by-detection with object mask depth retrieval through stereo vision and lidar*," in *2024 27th International Conference on Information Fusion (FUSION)*, 2024, pp. 1–8.

[15] Y. Xie, C. Nanlal, and Y. Liu, "Reliable lidar-based ship detection and tracking for autonomous surface vehicles in busy maritime environments," *Ocean Engineering*, vol. 312, p. 119288, Nov. 2024.

[16] 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, Sep. 2022.

[17] D. Musicki and R. Evans, "Joint integrated probabilistic data association - jipda," in *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997)*, vol. 2, 2002, pp. 1120–1125.

[18] H. Hilmarsen, "Maritime tracking-by-detection using camera and lidar," Master's thesis, Norwegian University of Science and Technology, Jun. 2024.

[19] M. Baerveldt, A. H. Gullikstad, and E. F. Brekke, "Comparing multiple extended object tracking with point based multi object tracking for lidar in a maritime context," *Journal of Physics: Conference Series*, vol. 2618, no. 1, 2023.

[20] L. Digerud, "Utilizing vision-based techniques to aid unmanned cargo ships in the auto-docking scenario," Master's thesis, Norwegian University of Science and Technology, 2022.

[21] S. Kim, H. Jo, H. Kim, and J. Park, "Development of an autonomous docking system for autonomous surface vehicles based on symbol recognition," *Ocean Engineering*, vol. 283, p. 114753, 2023.

[22] Ø. Volden, A. Stahl, and T. I. Fossen, "Development and experimental validation of visual-inertial navigation for auto-docking of unmanned surface vehicles," *IEEE Access*, vol. 11, pp. 45 688–45 710, 2023.

[23] H. Wang, Y. Yin, Q. Jing, Z. Cao, Z. Shao, and D. Guo, "Berthing assistance system for autonomous surface vehicles based on 3d lidar," *Ocean Engineering*, vol. 291, Jan. 2024.

[24] T. A. Nygård, N. Dalhaug, R. Mester, E. F. Brekke, and A. Stahl, "Stereo camera-based free space estimation for docking in urban waters," *Modeling, Identification and Control*, vol. 45, no. 2, pp. 51–63, 2024.

[25] W. Wang, T. Shan, P. Leoni, D. Fernández-Gutiérrez, D. Meyers, C. Ratti, and D. Rus, "Roboat ii: A novel autonomous surface vessel for urban environments," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 1740–1747.

[26] W. Wang, D. Fernández-Gutiérrez, R. Doornbusch, J. Jordan, T. Shan, P. Leoni, N. Hagemann, J. K. Schiphorst, F. Duarte, C. Ratti, and D. Rus, "Roboat iii: An autonomous surface vessel for urbantransportation," *Journal of Field Robotics*, vol. 40, no. 8, pp. 1996–2009, Aug. 2023.

[27] OpenCV, *OpenCV Documentation*, Nov. 2024, https://docs.opencv.org/3.4/index.html.

[28] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the second International Conference on Knowledge Discovery and Data mining (KDD-96)*, 08 1996, pp. 226–231.

[29] C. Fruzzetti, M. Martelli, A. Lekkas, R. Skjetne, and M. Breivik, "Model-based motion control design for the milliampere1 prototype ferry," in *2024 European Control Conference (ECC)*. IEEE, 2024, pp. 3636–3643.

[30] T. I. Fossen, "An adaptive line-of-sight (alos) guidance law for path following of aircraft and marine craft," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 6, pp. 2887–2894, 2023.