An Hierarchical Architecture for Docking Autonomous Surface Vehicles

Pedro Leite, Renato Silva, Aníbal Matos and Andry Maykol Pinto

Centre for Robotics and Autonomous Systems of INESC TEC

Faculty of Engineering, University of Porto

Email: {pedro.nuno, renato.j.silva}@inesctec.pt, {anibal, andry.pinto}@fe.up.pt

Abstract—Autonomous Surface Vehicles (ASVs) provide the ideal platform to further explore the many opportunities in the cargo shipping industry, by making it more profitable and safer. This paper presents an architecture for the autonomous docking operation, formed by two stages: a maneuver module and, a situational awareness system to detect a mooring facility where an ASV can safely dock. Information retrieved from a 3D LIDAR, IMU and GPS are combined to extract the geometric features of the floating platform and to estimate the relative positioning and orientation of the moor to the ASV. Then, the maneuver module plans a trajectory to a specific position and guarantees that the ASV will not collide with the mooring facility. The approach presented in this paper was validated in distinct environmental and weather conditions such as tidal waves and wind. The results demonstrate the ability of the proposed architecture for detecting the docking platform and safely conduct the navigation towards it, achieving errors up to 0.107 m in position and 6.58° in orientation.

I. INTRODUCTION

The application of autonomous vehicles in the maritime industry is still very new, yet its growth is imminent as projects for fully scaled vessels are starting to become a reality [1]. As Autonomous Surface Vehicles (ASVs) continue to prove their reliability in simple and specific tasks where human labor could potentially be endangered, more emphasis is being put onto these ships as the future of cargo shipping.

To secure regulatory approval, as well as industry support and public acceptance, ASVs need to be at least as safe as the existing vessels [2]. Currently one of the operations that involve the most risks is certainly the docking one. A failure in this process can lead to a great amount of monetary losses by damaging the vessel, the physical infrastructure of the port or even the support crew [3]. Making this operation autonomous would greatly reduce these risks.

The autonomous docking operation must be capable of overcoming the challenges that harsh weather and environmental conditions pose. To accomplish such, the vessel needs a robust system that makes it aware of its surroundings and ables it to identify the docking platform and possible hazards. When in close proximity to the mooring facilities, the ship must be controlled in order to follow a given path that leads to a safe docking maneuver, avoiding damages to both the vessel and the physical infrastructure.

This paper introduces an integrated architecture that aims to solve the presented challenges. It can be divided into two stages: a Situational Awareness (SA) system and a Maneuver module. The first stage combines the data retrieved from a 3D LIDAR, an IMU and a GPS to map the vessel's surroundings and recognize the dock. As this stage targets the identification of expected geometrical shapes, there is no need to modify the dock or surrounding waters by adding beacons or transponders. The maneuver module is the complementary part of the SA system, taking into consideration the reconstruction of the scenario obtained by this system, the route from origin to destination and other navigational aspects (e.g. specific actions to perform in each waypoint) to generate a suitable path. A key part of this operation is ensuring that the planned trajectory is a safe one, through a viability check procedure.

The contributions of this paper include:

- An integrated architecture for the docking operation, formed by a situational awareness system and a maneuvering module;
- A SA system that takes an hierarchical approach towards template matching and fuses information from multiple sensors to identify the dock and calculate its relative pose to the ASV;
- A maneuver module that ensures the navigation of the ASV towards the berth, supervising the operation through a viability check heuristic;
- An extensive qualitative and quantitative evaluation under realistic environmental constraints, wind and three sets of waves composed by an amplitude, period and direction.

The article is organized as follows: section II presents a brief review of the existing approaches for the docking operation. Section III shows the concept of the introduced architecture, detailing both the maneuver module and the SA system. Once presented the architecture, section IV shows the results obtained in a realistic simulation environment, capturing distinct scene conditions. Finally, section V presents the most important conclusions of this research and the future work.

II. RELATED WORK

In the research about autonomous vehicles very few propose and detail a complete architecture for the docking operation. From those that do, the SA system is either based on knowledge and models, or it relies on data from LIDARs and/or cameras. As for trajectory planning, most of the solutions

resort to the well known A* algorithm. The work presented by Christopher Ash *et al.* [4] combines data from multiple LIDARs, which is fused together and then flattened into an occupancy grid, where it is treated as a binary image. The geometric features of the dock are then computed, with data from a camera, used to define a Region of Interest (ROI). The A* algorithm then generates a path for the operation. Joohyun Woo *et al.* [5] utilizes learning based recognition algorithms to identify the dock, projecting the sensors' information into a grid map. Relevant features from the represented objects are computed and a Support Vector Machine is used for shape classification. The usage of the LIDAR is associated with some uncertainty when it comes to its robustness in harsh conditions such as the maritime ones [6]. Despite these concerns, perception systems solely based on LIDAR have been proposed and demonstrated, e.g. [7]. One of the possible approaches is to project the 3D point cloud into a plane and process the data as an image with computer vision algorithms. A different method is proposed by Joel M. Esposito *et al.* [8], using as input the 3D point cloud returned by the LIDAR and a target cloud (representative of a known dock model) generated *a priori*. A rough first alignment is made and then the Iterative Closest Point (ICP) algorithm is used to estimate a transformation that fully aligns the correspondences in both clouds. Model and knowledge based approaches are also presented by the likes of Qi Chen *et al.* [9] and Zhu Bing *et al.* [10], utilizing satellite provided images to identify a possible harbor location with computer vision algorithms.

III. ARCHITECTURE FOR THE DOCKING OPERATION

An example scenario of the data captured by the LIDAR of the docking platform is shown in figure 1, the reflections caused by the challenging conditions faced in a maritime environment hinder the recognition process. The architecture proposed in this paper is presented in figure 2, this diagram is representative of the interactions between the integrating modules. Its goal is to handle these environmental constraints, enabling the system to function in multiple scenarios, controlling the movement of the vessel in close proximity to physical infrastructures without damages.
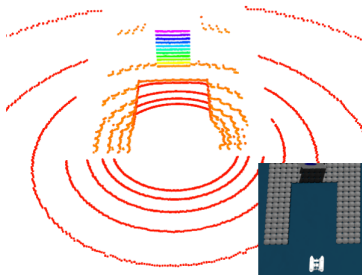

Figure 1: Example of the data gathered by the LIDAR and its respective simulated scenario.

The dynamics that occur inside each module are also depicted in figure 2. Information from the sensors is gathered and used by both subsystems that integrate the SA module, making use of it to estimate a target point for the docking. Once received, the target is validated inside the maneuver module, generating linear and angular velocities that are then converted into specific velocities (V_1,V_2) for each thruster and fed to the vessel's differential drive.
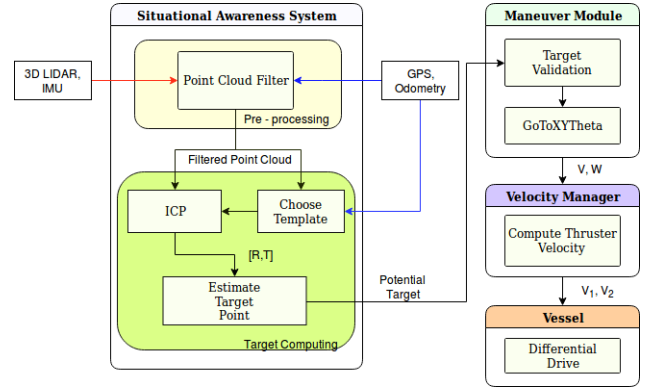

Figure 2: Proposed Architecture for the Docking Operation.

A. Situational Awareness System

For the docking operation to be successful, making the vessel aware of its surroundings is a vital part of the process. This fact makes the SA system the most important part of the presented architecture. A division into two subsystems can be noticed in figure 2, a pre-processing and a target computing node integrate this system.

The data obtained from the 3D LIDAR is susceptible to the environmental conditions, the constant bobbing of the ASV caused by the tidal waves and wind can result in a very noisy point cloud, figure 1 for instance. For the SA module to work properly in a maritime scenario it must be able to interpret and analyze the reliability of the sensor information according to different environmental conditions at all moments. A simple but effective approach is to consider a ROI that can be adapted to the local conditions of the scenario. To mitigate the effects caused by the environment, this region is adjusted dynamically by using GPS and IMU data. The sensor data is then used to estimate a target point.

The distance from the dock affects the information gathered by the LIDAR, as the ASV gets closer to the floating platform, more reliable and precise data can be collected. For this reason, increasingly more complex templates should be used for matching by the registration method. This hierarchical approach can be observed in figure 3, the degree of detail can be extended to *n* levels, each correspondent model more complex than the previous. The Iterative Closest Point (ICP) algorithm is used as the registration method. Given an input point cloud and a target one, this algorithm estimates a transformation matrix (composed by a Rotation and a Translation) $[R, T]$ that aligns the correspondences between both [11].

An example of application for the proposed approach, with 3 levels, is presented in figure 4 (only the two extremes are
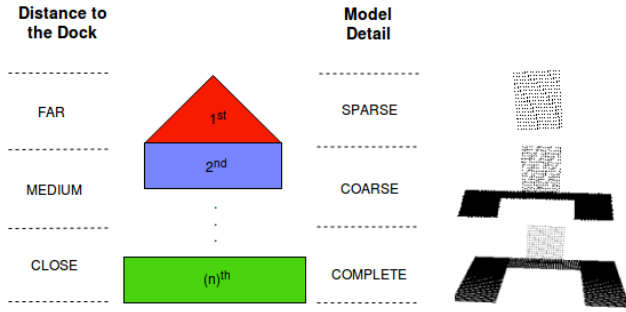
Figure 3: Representation of the Hierarchical Approach to Template Matching.

shown). Represented in black are the templates, the green and red point clouds are the input and output, respectively. When the ASV is rather far away from the docking platform, see figure 4.c), the most relevant part of the 3D scenario reconstructed by the SA module relies upon the upper structure of the dock. Once it gets closer, figure 4.a), a much more complete and reliable point cloud will be gathered, and thus, a template with more detail will be considered during the registration process. The transformations from the initial stage, see figures 4.a),c), to the final one, represented by figures 4.b) and 4.d) respectively, are returned via the $[R, T]$ matrices previously presented.
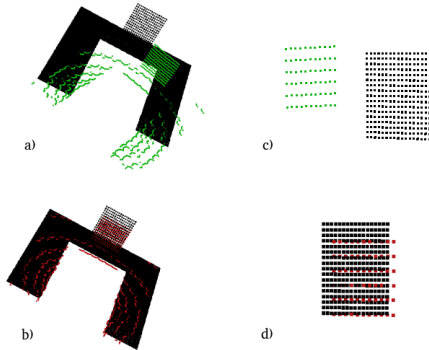

Figure 4: Example of application for the proposed hierarchical approach with ICP. a), c) Close and Far view templates (black), respectively, and input point cloud (green); b), d) Clouds are aligned with the targets (red) after the iteration.

To estimate a target point from this information, first, with basis on previously gathered knowledge of the platform's geometrical features (height and width of the berthing area) and the GPS data, Universal Transverse Mercator (UTM) coordinates of the dock, an optimal docking point is calculated and used as the reference for the algorithm. Take this target point as a position and orientation relative to the mooring facility where the ASV should stop. The $[R, T]$ matrix obtained from the ICP is representative of the relative pose between the dock and the ASV, as shown in figure 4.

As the reference point is known in UTM coordinates,

equations 1 and 2 are used to obtain these coordinates in reference to the ASV's current position.

$$P_{target} = R^{-1} \cdot P_{ref} - T \qquad (1)$$

$$\begin{bmatrix} X_{target} \\ Y_{target} \end{bmatrix} = \begin{bmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{bmatrix} \cdot \begin{bmatrix} X_{ref} \\ Y_{ref} \end{bmatrix} - \begin{bmatrix} t_x \\ t_y \end{bmatrix} \qquad (2)$$

where P_{target} is the target point after the transformation, P_{ref} is the point taken as reference (in UTM coordinates) and R, T are the Rotation and Translation matrices returned by the ICP, respectively.

This potential target point is then processed by the maneuver module, which is responsible for deciding whether this is a valid point or not.

B. Maneuver Module

This module is designed to act as a complementary system towards the SA algorithm, making the connection between it and the velocity manager. It is tasked with ensuring the safe navigation of the vessel. Once a new potential target point for the docking is received, the heuristic presented in figure 5 is used to check if the given target is valid or if it proves unsafe, potentially causing harm towards the vessel and/or mooring facilities. Each time the ICP converges, a fitness score is also returned. This score is calculated by summing the squared distances from source to target, being indicative of the quality of the convergence. For this heuristic, the score is used as the first metric of evaluation. If the target point, correspondent to the latest convergence of the ICP, has an associated fitness score higher than a threshold defined *a priori* (by observation of several iterations the value 0.095 is used as threshold), it is considered a bad target point. When this occurs, the given X and Y coordinates of the target are ignored, the ASV stays on its current location and uses the target's orientation to rotate under its own axis. As it aligns with the docking platform, the LIDAR will be able to provide better data which will consequently generate a more viable target point.

The navigation control is based on the center of mass of the ASV, if the threshold of the first metric is respected, the 3D scenario is represented (a *KdTree* being an efficient method to do so) and a radius that takes into consideration the dimensions of the vessel as well as a safe margin distance towards the platform is defined. This aims to ensure that the given coordinates of the potential target do not lay inside of the input point cloud. If there are any points within radius, the target is considered a bad point and its position is discarded, following principle previously described. Otherwise, it is accepted as valid target point and both the position and orientation are kept and used to generate the adequate path.

In order to navigate the vehicle towards its desired berthing location, a control module was implemented by taking into consideration the direction and distance to the desired target point. This path is discretized in small (free) sections and, once it is reached, the final orientation is corrected. The

position error, utilized as a norm of the X and Y coordinates, was given a tolerance of $|0.2|m$, while the final θ error was given $|8.6|°$. This module is also responsible for computing linear and angular velocities, feeding them to the ASV's velocity manager. These take into consideration the kinematic constraints of the vessel, making sure that the maximum velocities are respected.
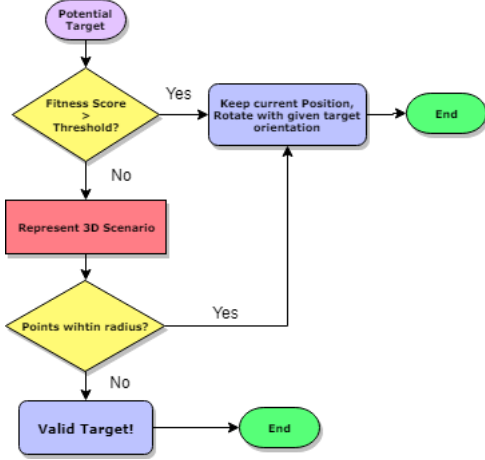

Figure 5: Flowchart representative of the target validation heuristic.

IV. PRACTICAL RESULTS

A set of experiments were conducted as part of this research. These aim to analyze and understand the behavior of the proposed architecture. Testing was carried out on a Gazebo 3D simulation representative of a maritime scenario, with both tidal waves and wind as environmental constraints. This simulation is the one provided by the RobotX Maritime Challenge. The model used in said simulation, was the Zarco ASV [12]. For these tests, it is assumed that the ASV departs from an initial point that lays inside a catch zone, this being the general area where the LIDAR is able to obtain any information from the docking platform. The position and orientation of the docking platform are also taken as known from the start. The metrics used to evaluate each iteration were both the position and orientation errors of the final pose of the ASV when compared to the reference point.

A. Minimal Environmental Constraints

The first set of tests were performed with minimal environmental effects, the wind was kept constant with $(\vec{V_x}, \vec{V_y}) = (1,1)$ m/s and three sets of waves were used in testing:
1) An amplitude of 0.06 m, a period of 12.6 seconds and a direction of $(\vec{x}, \vec{y}) = (-1,0)$;
2) An amplitude of 0.04 m, a period of 3.7 seconds and a direction of $(\vec{x}, \vec{y}) = (-0.7,0.7)$;
3) An amplitude of 0.03 m, a period of 6.3 seconds and a direction of $(\vec{x}, \vec{y}) = (0.7,0.7)$;

The results are presented in figure 6. This graphic is representative of the evolution of the ASV's position within three trajectories with different starting points. Trajectory 1 starting from the left, trajectory 2 in front of the platform and finally, starting from the right is trajectory 3. Dashed lines are representative of the docking platform's walls and the circle of the optimal target point that serves as a reference (with the given $|0.2|m$ of tolerance). The final position of the ASV is marked at the end of each trajectory with a cross.
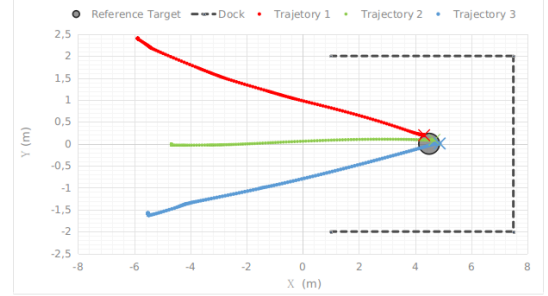

Figure 6: Evolution of the ASV's position with minimal environmental constraints.

Even if minimal, the effects of the waves and wind, can still be noticed in both trajectories 2 and 3, represented with the green and the blue colors, respectively. Ideally, the trajectory that starts right in front of the dock would just have to suffer a translation towards the reference point, since it is already aligned. However, a slight variation of the Y coordinate can be observed. The reasoning behind this, is a possible wave that shifted the vessel from the optimal path, since the induced error is not enough to trigger a response from the algorithm, this error is not corrected. Trajectory 3, also suffers from the effects of the maritime environment, right at the start, as the ASV is adjusting its orientation towards the computed line, some minimal drag from the waves can be seen. It is quickly made obsolete once the ASV finishes its alignment and starts moving. From the set of tests run in these conditions, an average position error of $0.1068\,\text{m}$ and a $6.6°$ error in the orientation were obtained, both within tolerance. If given a tighter tolerance these errors could potentially be reduced even more.

The performance of the SA system was also tested for these environmental constraints. The target point used as a reference is the point $(X, Y) = (4.7, 0)$ and a trajectory like the one presented in red (trajectory 1) in figure 6 was used as sample for this test. It is to be expected that the targets generated from data collected from a further away point of view would be rather inaccurate and converge to the reference as the ASV approaches the docking platform, figure 7 illustrates such behavior.

The evolution of the positional error correspondent to the estimated targets, is rather inaccurate and has an high variability during the first 60 points. This can be considered the period of time when the data gathered by the LIDAR is not viable enough to generate a precise target. As the ASV approaches the platform, better quality data is retrieved from the sensors and more viable points are estimated, as proven by the latter
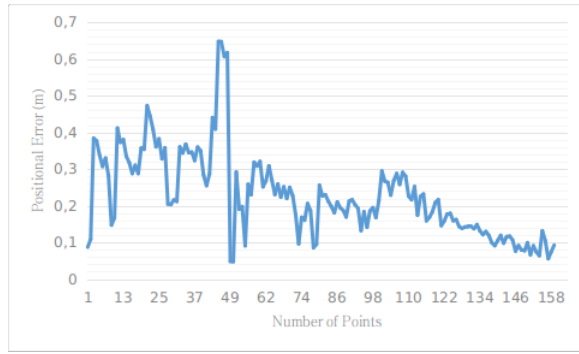
Figure 7: Evolution of the positional error for the estimated targets, with minimal environmental constraints.
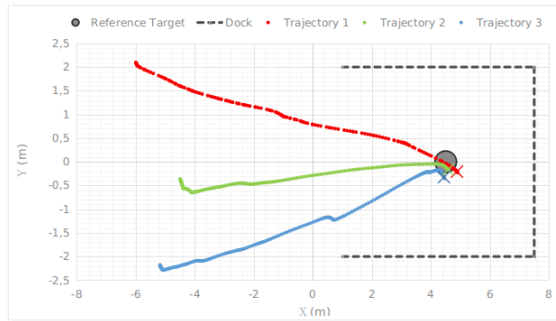

Figure 8: Evolution of the ASV's position with harsh environmental constraints.

set of points as these are much more accurate and converge towards the given reference.

B. Harsh Environmental Constraints

The second set of tests were carried out in harsh environmental conditions, with waves three times higher in amplitude. The wind condition was kept constant. Results from said testing are presented in figure 8.

From the initial instants of each trajectory, a clear influence of the waves and wind can be seen, dragging the vessel along the negative Y axis. During the first path (represented in red), the ASV takes advantage of the the momentum generated from these constraints and reaches the target much quicker, this can be seen as this trajectory has a visible lower number of points. However, the extra momentum ends up causing a small overshoot in the vessel's final position. The path that begins aligned with the dock, trajectory 2, has a clear deviation in its position every time it stops to rotate and adjust its orientation, this drag is compensated quickly once the desired θ is achieved. When the ASV reaches its goal and stops to readjust its final orientation, a visible drag starts to occur once again, as it does not exceed the tolerance limit, no action is taken. Finally, the trajectory represented in blue starts in the worse possible scenario, as the waves are countering the desired action. It shows a slower response and, reflecting the behavior of the path in green, when it stops to adjust its orientation, a visible drag occurs that leads to a positional error

in permanent state. The tests performed in these conditions had an average position error of $0.2652\,\mathrm{m}$ and a $9.1°$ error in the orientation, slightly overshooting the given tolerance. The results obtained from testing the SA system in these harsh conditions can be observed in figure 9. As the algorithm relies on the data gathered from the LIDAR, it is to be expected that these conditions could hinder the efficiency of the system.
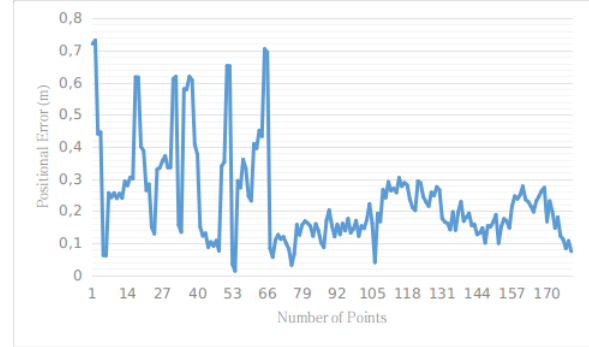

Figure 9: Evolution of the positional error for the estimated targets, with harsh environmental constraints.

Comparing the results from figure 9 with the ones obtained in figure 7, a much higher variability can be observed during the initial period. This is obviously caused by the stronger conditions. A later convergence towards the desired target point can also be noticed, as the graph from figure 9 has a larger amount of points. The reasoning behind this, is that the ASV has more difficulty in achieving the desired rotation since the waves keep interfering with the process, knocking it out of the desired path and shifting its orientation. Once the ASV's alignment is enough to get viable information from the docking platform, the estimated target point starts converging towards the desired reference. The heuristic presented for the target validation can be seen in action in figure 10. Whenever the estimated target is rejected, the ASV's current position is taken as the target, these points mirror the trajectories presented in figures 6 and 8. The later convergence of the trajectory affected by harsh conditions (depicted in orange) can be seen as points get rejected within a closer distance to the dock (approximately $1\,\mathrm{m}$), while the other trajectory accepts the target points as reliable $2\,\mathrm{m}$ away from the platform. Excluding some outliers, the trajectory affected by minimal constraints is more accurate along the course. The variability induced by the stronger waves can be observed as the accepted points are much more distributed for this trajectory.

A summary of the obtained results is presented in table I. As expected, the accuracy of the algorithm is considerably affected by the environmental constraints, notably in the positional as the error roughly doubles from one scenario to the other. A slight increase in the orientation error is also observable. From comparing both scenarios, we can easily conclude that the stronger waves and wind also hinder the quality of the estimated target point, as the latter scenario takes longer to converge towards the expected reference.
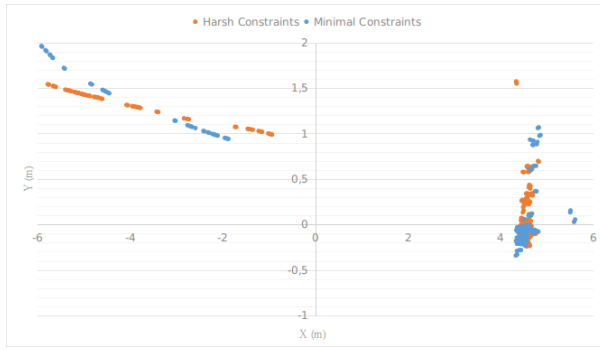
Figure 10: Comparison of the distribution of the estimated target points with both environmental constraints.

Table I: Summarized comparison between scenarios.

Scenario	Positional Error (m)	Orientation Error (°)	Rejected Targets (%)
Minimal Constraints	0.1068	6.6	18.09
Harsh Constraints	0.2652	9.1	33.81

Table II presents further comparison between models for both the tested scenarios. The standard deviation of the distance for the far model is higher with both environmental constraints, as expected. This is due to the quality of the data being much more reliable when within close proximity with the dock. When subjected to harsh constraints the number of adjustments made by the ASV is larger, meaning that the amount of rejected target points is also higher (see table I), due to the fact that the stronger constraints hinder the needed adjustments.

Videos representative of the experiments are available in[1,2].

Table II: Comparison between the Far and Close Model for the studied scenarios.

Scenario		Far Model	Close Model
Minimal Constraints	Distance Error (m)	0.1270	0.0642
	# Adjustments	13	11
Harsh Constraints	Distance Error (m)	0.1929	0.0786
	# Adjustments	56	14

V. CONCLUSIONS AND FUTURE WORK

This research presents an hierarchical architecture for the docking operation of ASVs, formed by a maneuver module and a SA system. Information from multiple sensors, such as a 3D LIDAR, IMU and GPS are gathered and fused together by the SA system. A pre-processing and a target computing node are integrated in said system. The practical results demonstrated the viability of this approach, as the loading of different templates shows a great convergence of the registration method. To safely conduct the navigation towards the estimated target point, a maneuver module was developed as part of the proposed architecture. Validation of the target and generation of the adequate linear and angular velocities are the main tasks of this module.

The proposed architecture has been tested within two sets of experiments, a set with minimal environmental effects, and a set with harsh ones. Evaluation of each set was based on the positional and orientational errors of the final point reached by the ASV when in comparison with the reference point previously calculated. More emphasis was given to the positional aspect of the trajectory since this is the key aspect that needs to be ensured to guarantee a safe navigation. Both the average error results are satisfactory, achieving errors up to $0.107\,\mathrm{m}$ in position and $6.6°$ in orientation. A slight deviation from the given tolerances is to be noted when harsh maritime conditions are into play.

Future work involves integrating into the proposed architecture a couple of modules responsible for computing the orientation and position of the docking platform as well as generating a suitable path to enter the catch zone. Finally, the current architecture will also be adapted for the undocking operation.

VI. ACKNOWLEDGMENT

REFERENCES

[1] J. Walker, "Autonomous Ships Timeline - Comparing Rolls-Royce, Kongsberg, Yara and More," 2017.

[2] D. M. Authority, "Analysis of Regulatory Barriers to the use of Autonomous Ships Final Report," tech. rep., 2017.

[3] Health and S. Authority, "Hazards in Port and Dock Operations," tech. rep., 2015.

[4] C. M. Ash, N. D. Bloom, J. D. Brown, A. N. Butka, S. P. Cronin, G. C. Delp, R. T. Goring, C. J. Hockley, Z. R. Joswick, D. Lodato, N. R. Middlebrooks, B. F. Mathews, J. L. Pletz, J. S. Romney, M. A. Schoener, N. C. Schultz, A. M. Thompson, D. J. Thompson, J. K. Wyble, T. A. Zuercher, C. F. Reinholtz, E. J. Coyle, B. K. Butka, and P. N. Currier, "Design of the Minion Research Platform for the 2016 Maritime RobotX Challenge," 2016.

[5] J. Woo, I. Seo, J. Lee, J. Park, A. Park, M. Kim, Y. Jung, J. Park, R. You, H. Choi, and N. Kim, "ASV : MACS," pp. 1–7.

[6] R. Royce, "AAWA Position Paper: Remote and Autonomous Ships - The next steps," 2016.

[7] R. Halterman and M. Bruch, "Velodyne HDL64E LIDAR for Unmanned Surface Vehicle Obstacle Detection,"

[8] J. M. Esposito and M. Graves, "An algorithm to identify docking locations for autonomous surface vessels from 3-D LiDAR scans," *IEEE Conference on Technologies for Practical Robot Applications, TePRA,* 2014.

[9] "Harbor detection of remote sensing images based on model," *Proceedings of the 2010 2nd International Conference on Future Computer and Communication, ICFCC 2010,* vol. 1, pp. 322–325, 2010.

[10] Z. Bing, L. Jinzong, and C. Aijun, "Knowledge Based Recognition of harbor target,"

[11] J. Sprickerhof, "PCL :: Registration," tech. rep., 2012.

[12] "Ocean Systems group website, https://oceansys.fe.up.pt/?section=tech."

[1]https://tinyurl.com/gazebosimulation

[2]https://tinyurl.com/multilevelapproach