Autonomous Docking between a Mobile Subsea Docking Station and an AUV while in Motion

Nikolas Dahn*, Christopher Gaudig†, Tim Lehr†
Niklas Pech†, Peter Kampmann†, Leif Christensen*
Frank Kirchner*

***Abstract*—The launch and recovery of an autonomous underwater vehicle (AUV) is a critical phase for the operation of these vessels and one of the most likely points of failure. One of the major risk factors are weather conditions, which may even prevent the process altogether. We address this issue with a novel launch and recovery system (LARS) comprised of a highly maneuverable AUV-like docking station tethered to a supply vessel. While this allows the mitigation of unfavorable weather conditions, it requires a higher degree of autonomous capabilities. In this paper, we present a novel method for facilitating physical docking underwater between two autonomous underwater vehicles, while both of them are in motion and with tight tolerances. This is achieved by blending between control laws based on a custom distance metric. Our approach is validated in simulation and physical trials.**

I. Introduction

With the recent push towards sustainability came the construction of thousands of new subsea installations like pipelines, foundation structures or cables, significantly increasing the demand for underwater inspection and maintenance. Due to safety concerns and the availability of human industrial divers, these tasks lend themselves naturally to remotely operated vehicles (ROVs) and autonomous underwater vehicles (AUVs) which require less or no human supervision at all.

However, an AUV's autonomy is always limited by its installed battery and ability to recharge it. While deploying a subsea-resident AUV [1] with its own docking stations and command servers (e.g. in an offshore wind farm) is possible, adding this kind of infrastructure to thousands of kilometers of pipelines is unfeasible. AUVs can thus only be used for short missions or when accompanied by a surface vessel, significantly increasing the costs through crew and equipment.

Instead of the continued deployment of a "mothership", we propose to use a highly mobile AUV similar to [2] to "catch" and recharge the inspection AUV underwater while in motion. This approach comes with some benefits: (1) deployment is no longer bound to a specific location, (2) the inspection AUV has less requirements on maneuverability, and (3) docking is decoupled from the sea state, i.e. wind and waves. Deploying the docking station by ship also means that the AUV does not depend on any fixed infrastructure as all power supply and communication facilities can be carried by the supply vessel.

* Robotics Innovation Center
German Research Center for Artificial Intelligence
Bremen, Germany
ORCID: 0000-0003-2262-2989, 0000-0003-2290-2163
0000-0002-1713-9784

† ROSENXT Creation Center GmbH
Bremen, Germany
ORCID: 0009-0001-0349-0979, 0009-0006-9090-4311
0009-0000-3103-9993, 0000-0002-1943-2466

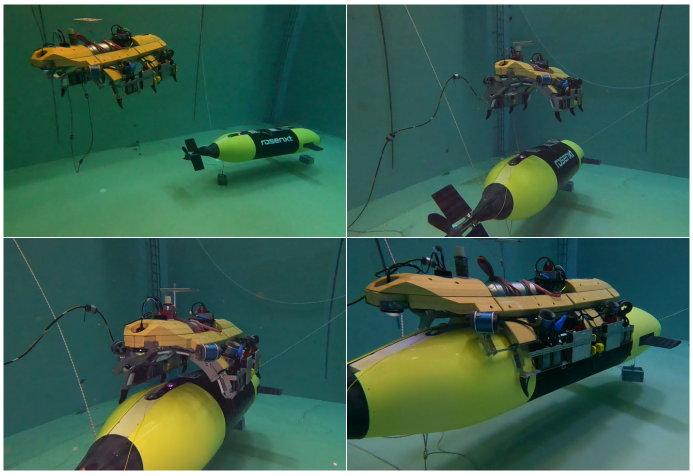


Fig. 1: A mobile docking station docking to an AUV. Video available at https://youtu.be/hwrcJu0mtdk.

In this paper, we consider a scenario where a target AUV (TAUV) has to be recharged by an AUV acting as a mobile docking station (DS). The TAUV is designed for maximum energy efficiency and thus needs to stay in motion in order to remain stable. Our approach is validated both in an underwater simulation as well as in a controlled *real-world* environment.

II. State of the Art

Docking in motion belongs to the broader spectrum of rendezvous maneuvers and can be categorized based on the vehicles' cooperativeness and maneuverability. For example, air-to-air refueling is fully cooperative and symmetrically underactuated, while an aerial drone evading a homing missile is antagonistic with a maneuverability advantage for the drone. Accordingly, we are looking at a cooperative scenario with a mobility advantage for the docking station. The scenario's difficulty is further described by environmental disturbances, like currents and waves in the surrounding medium.

In [3], the problem of a drone landing on a moving boat is treated as an optimization problem. Using a model predictive control (MPC) algorithm with variable horizon, the authors derive a set of objective functions and constraints to find

the optimal rendezvous trajectory for the aerial vehicle. They present results from a simulation as well as real-world tests with a drone and a virtual boat, with a landing corridor of $\pm 0.5m$. However, our scenario is specifically targeted at *docking* in motion and thus requires much higher precision.

A related scenario is presented in [4], where up to 25 drones coordinate with a moving truck to decide on optimal landing times before their batteries run out. The drones are subject to perturbations such as higher-than-estimated power consumption or delays in individual missions. While less concerned about the actual rendezvous procedure, coordination between vehicles plays a major role in maintaining cooperative conditions.

A rendezvous-and-docking algorithm with multiple AUVs was investigated in [5], albeit in a planar case with simplified vehicle models. MPC is used to take additional constraints into account, like staying within range of a mission goal. The docking process itself and the effects on the resulting trajectory are not considered.

For the case of passive underwater docking stations, the *EurEx-LUNa* project [6], [7] has explored methods for autonomous docking under ice including practical trials, with the goal of one day deploying such a vessel on Jupiter's ice moon Europa. Although a layer of ice isolates the vessel from the sea state and thus allows for a simpler approach, robustness is of high importance to their work, since any failure can be potentially catastrophic, especially in the targeted use case of extraterrestrial exploration.

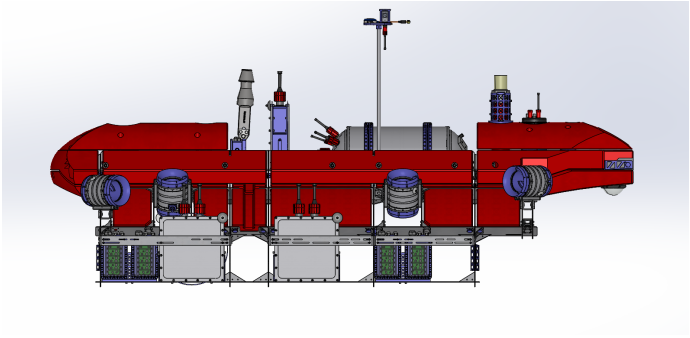
AI-based docking approaches have also been explored for underwater scenarios. For example, in [8] the authors present a novel reward function for deep reinforcement learning to approach a passive docking station with high reliability, albeit in simulation.

Overall, we find that the prior research is not directly applicable to our use case: the practical and physical docking of two AUVs while in motion.

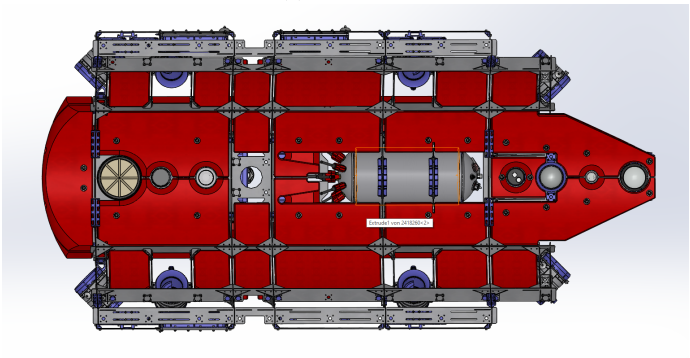
III. Scenario Specification

In contrast to many docking stations for underwater applications developed and presented in section II, the DS presented here is intended to be deployed from a surface vessel (SV) and then used to retrieve a target AUV (TAUV) out of the water. These situations form the most critical part of any AUV operation, as crashes of the TAUV into the SV structure are likely and can render the AUV inoperable, especially in rough sea states. While it is possible to work around weather conditions when launching the AUV, the conditions during retrieval after a long endurance mission of several days or even weeks are rarely predictable. The docking robot used here was designed with this scenario in mind. In order to achieve a robust connection between an SV and a survey AUV, we intend to initiate the docking process far below the water level, free of any weather-induced disturbances.

Furthermore, we designed the DS to be as agile as possible in order to capture the TAUV. By using such a design, we enable the survey AUV to be as energy efficient as possible,



(a) Side view



(b) Top view

Fig. 2: The developed docking station (DS)

requiring only limited dynamic responsiveness in terms of direction adaptation. The DS on the other hand has a virtually unlimited power source as it is tethered to and supplied by the SV. Using a DS with a large maneuverability advantage compared to the TAUV will not just make the docking process faster, but also more reliable by enabling fine and controlled adjustments.

These design considerations have been incorporated into the concept phases of the system development [9]. The resulting docking station is shown in Figure 2.

The DS comprises eight thrusters in total, allowing for propulsion vectors in all six degrees of freedom. Waypoint navigation for the DS is realized using a global navigation satellite system (GNSS) receiver, a fiber-optic gyro inertial measurement unit (FOG IMU), doppler velocity log (DVL), and ultra short baseline (USBL) transponder. The onboard camera and lighting system can be used for close-range tracking of the TAUV. Further communication devices are an optical communication unit and a close-range high-bandwith data transmission unit that works by modulating an inductively coupled magnetic field.

The TAUV used in this scenario is depicted in Fig. 3. The for the docking process relevant navigation and communication devices are matching those of the DS.

To integrate with the existing software ecosystem already used on the TAUV and DS (as well as other AUVs), the

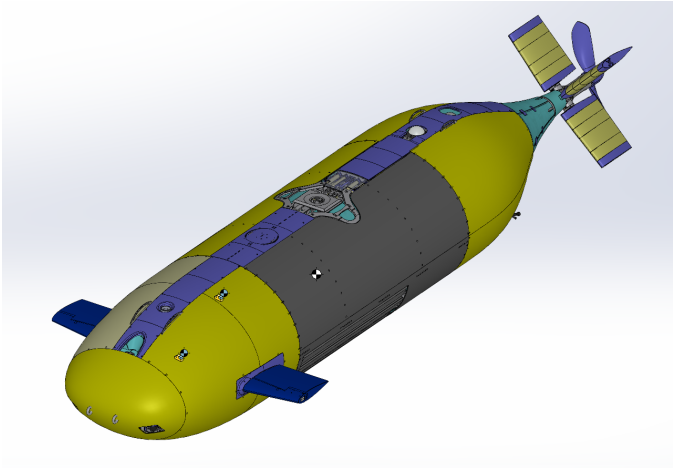


Fig. 3: Sketch of the target AUV (TAUV) used for docking trials

newly developed components for the docking process adhere to predefined interfaces and implement a state machine to reflect whether a module is stopped, initialized, configured, connected, paused or running [10].

High-level mission planning is based on the PLEXIL language [11]. Our mission specifications are designed such that both AUVs are able to handle successful and failed docking attempts, e.g. by communicating their state to the SV and waiting for new commands from a human operator, or (depending on the mission configuration) by autonomously restarting the docking process. Other error-handling behaviors like letting the TAUV proceed towards a safe loitering position can be implemented as well.

IV. Methodology

Our approach to enable docking in motion comprises two main components: (1) a docking protocol, which coordinates the behavior between the vessels, and (2) the actual docking or homing behavior. The docking protocol depends on an established communication channel, e.g. acoustic modems (often integrated with USBL positioning transceivers), optical communication, WiFi or Ethernet. The docking behavior depends on odometry estimates of the TAUV, which are provided by an unscented Kalman filter integrating multiple sensors and communicated information.

A. Docking Protocol

Since docking requires the coordination between three vessels (DS, TAUV, SV), we defined a protocol that allows each participant to agree on (or reject) the modalities before the actual docking process starts. The protocol is implemented in the form of three state machines, one for each participant, that interact with each other by exchanging messages. The protocol is shown in fig. 4 and has four distinct phases: negotiation (1-3 in the figure), approach (4-5), docked, and disengage (6).

Although not shown in the figure, all messages are forwarded to the SV which can then confirm the requests to

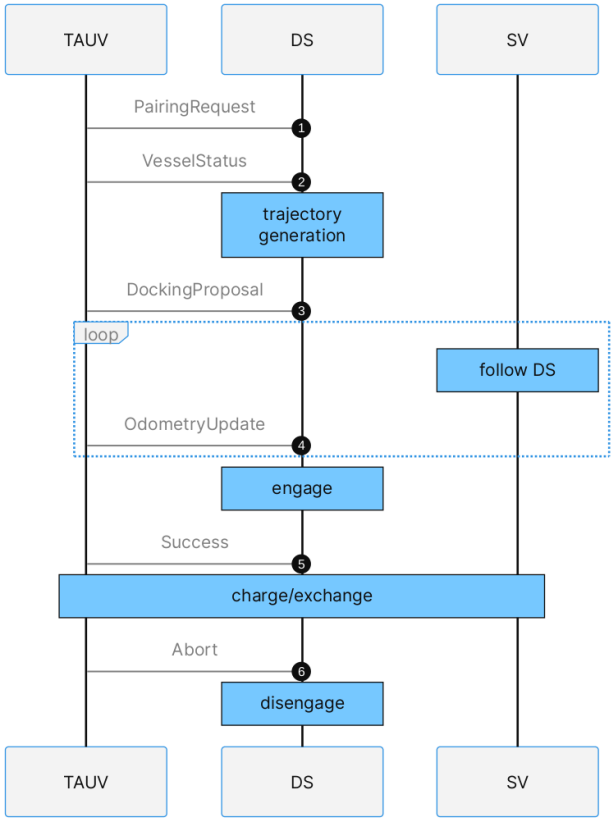


Fig. 4: Our docking protocol. For brevity, communication with the SV is not shown; assume that every message is forwarded to and approved by the SV.

continue the docking procedure. This allows the SV to provide additional restrictions, e.g. to take navigation capabilities, political borders and the tether between DS and SV into account, as well as enabling human involvement. In case of errors or unacceptable conditions, any participant may cancel the docking process in any state, which shortcuts into the *disengage* sequence. Obviously, this level of coordination depends on an established communication channel. While generally not a strict requirement, it allows for a more sophisticated and robust docking procedure and is a rather unintrusive requirement for cooperative scenarios.

B. Communication

To enable coordination between DS, TAUV and SV, and to improve pose estimates during the actual docking process, various communication channels are available between the vessels. They differ primarily in their range, data rate and availability. This is why the main channel in use changes with the different phases of docking.

With a range of several kilometers, an Ultra-Short Baseline (USBL) acoustic positioning system can be used in particular when the vehicles are still so far apart that other communication channels are not yet available. The USBL system consists of acoustic modems installed on the SV, DS, and TAUV and can be used to exchange small data packets alongside the

positioning data. Communication via USBL has the disadvantage of low bandwidth and high latency. Noise, reflections and acoustic interference can adversely affect transmission.

When the DS and the TAUV have come within approximately 30 to 100 m of each other, and the DS is located above the TAUV, BlueComm beacons on the DS and TAUV may provide an optical data link. This connection offers a moderate to high bandwidth with low latency, so that more data can be exchanged between the systems than with the USBL. The optical connection can be affected by turbidity and optical interference.

Since the DS is tethered to the SV, the tether can be used for high bandwidth, very low latency data exchange. While the docking process is not yet completed, the data rates and the availability of acoustic and optical links are the limiting factors of communication between the TAUV and DS.

Both the DS and the TAUV have a Blue Logic Subsea "USB" system. One of the main objectives of the docking is to align both Subsea "USB" systems to provide a wireless, inductive data and power connection between the DS and TAUV. With a bandwidth of up to 100 MBit/s and low latency, this is the fastest communication channel to the TAUV. Log files can be downloaded from the TAUV and new missions can be uploaded while docked. The Subsea "USB" devices can transfer up to 2 kW of power to the TAUV while docked so as to recharge the TAUV battery.

The available communication modalities are unified by an abstraction layer that transparently selects the best (highest bandwidth) channel available. To simplify the exchange of data, especially when only communication channels with low data rates are available, protocol and telemetry packets are compressed before sending.

Our scenario benefits from the fact that the vessels are expected to get closer to each other as time passes. Distance has an inverse relationship with many key channel parameters like latency, bandwidth and signal-to-noise (SNR) ratio. In addition, in underwater settings, higher bandwidth channels like optical and radio communication are only available at short distances. The quality of available communication channels therefore improves the further the docking process advances, allowing for higher update frequencies of e.g. telemetry data for odometry estimates.

C. Odometry Filter

To achieve docking with tight tolerances, the odometry of the TAUV must be known as accurately as possible. In our scenario, the DS has access to several estimates of the TAUV's current pose and motion:

- Telemetry messages from the TAUV (geographic pose, velocity)
- USBL localization (relative pose, geographic pose)
- Visual markers (relative pose)
- Planned course, speed and orientation of TAUV

These sources are subject to various limitations. The telemetry messages and USBL measurements rely on the acoustic channel, which introduces high latency and is particularly unreliable. The visual markers (selected based on tests in [12]) only work at very close distances and depend on the water's turbidity. The reliability of the desired course, speed, and orientation of the TAUV depend on how accurately it is actually able to follow them.

To arrive at reliable estimates, we employ an Unscented Kalman Filter (UKF) to fuse these data sources. The state of the filter includes the pose and velocities in a local cartesian coordinate system and is initialized with the desired course, velocity, and orientation. Geographic poses and velocities from telemetry messages are projected into this local coordinate system in advance.

D. Docking Behavior

Part of the docking protocol is negotiating a trajectory on which the docking should happen. It is limited by the maneuverability of the docking station (DS), target AUV (TAUV) and supply vessel (SV) as well as geopolitical boundaries and possibly other user parameters (e.g. staying close to the mission site). The DS first collects all restrictions and calculates the available area from them. It then uses an iterative search algorithm to find a straight line long enough for docking. If it cannot find one, it instead layers large circles on top of each other to achieve the desired trajectory length.

The resulting path is transmitted to the TAUV. While the TAUV is expected to adhere to the docking trajectory, the DS uses a control algorithm which is independent of the negotiated path and instead relies on frequent updates of the TAUV's odometry. At this point, the SV has already agreed to the negotiated modalities and is thus expected to follow the tethered DS as necessary.

In general, docking can only be successful if the DS approaches the TAUV gently *and* from the correct direction, as well as with sufficiently tight tolerances. We solve this by letting the DS approach the TAUV through a sequence of waypoints. These waypoints connect into line segments or *waylines*. Each waypoint i is associated with a threshold τ_i, and a control law $\mathcal{C}_i$ that generates acceleration commands to steer the DS towards the waypoint as shown in fig. 5. This allows us to use different control laws and parameters for different phases of the approach. To achieve smooth movement, we use the distance of the DS from the closest wayline to blend between control laws according to Algorithm 1.

Algorithm 1 Control law blending based on waylines

$d := \min(P_{DS}, \overline{p_i p_{i+1}}), i = 0, 1, 2, ...$ ▷ distance
$w = d/\tau_i$ ▷ weight
$\gamma_k = \mathcal{C}_k(P_{DS}, p_k), \quad k = i, i+1$ ▷ law outputs
$\Gamma = w * \gamma_i + (1 - w) * \gamma_{i+1}$ ▷ final command

Using way*lines* instead of way*points* has the important benefit that the employed laws cannot create an equilibrium where the DS would get stuck between two attractors. Since

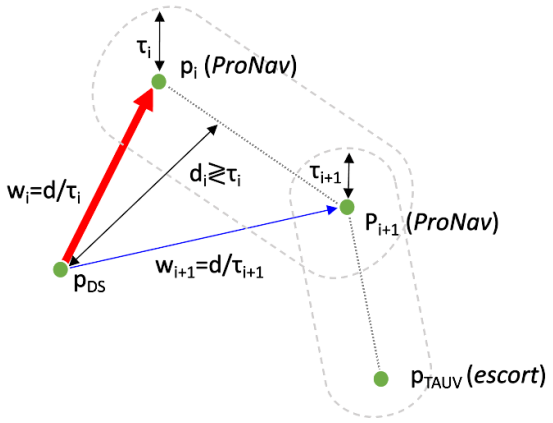


Fig. 5: Blending of control law outputs based on wayline distances.

blending is based on the distance to the *line* connecting the laws' reference points, this factor will still change even when both laws would create an equilibrium, as they will still pull the DS towards the wayline and thus shift the weights' ratio. This also means that once the DS' location coincides with the wayline, the second waypoint's control law will be in full control and drag the DS along the wayline, until it enters the threshold of the next wayline. We generate steering commands using three user-parameterized control laws:

- Zero effort miss (ZEM, long distance, predictive),
- Proportional-integral-derivative control (PID, close range, reactive), and
- Model-predictive control (MPC, close range, predictive).

1) Far Distance (ZEM): For steering the DS towards the TAUV at longer (non-docking) distances, we implemented a control law based on Proportional Navigation (ProNav) [13], which is often used in homing missiles; however, related methods are also used in maritime and aviation applications to e.g. avoid collisions.

ProNav is based on the understanding that a vehicle is on a collision course when the line of sight (LOS) unit vector $\mathcal{L}$ towards the target remains constant over time. As a result, changes in the LOS can be translated into acceleration commands to counteract any relative motion away from the collision course. Delays and noise in the measurement of the target's movement can be compensated by applying a factor > 1 (the navigation factor, typically in the range of 2 to 5) to the resulting accelerations, essentially overcompensating for deviations. Additional terms can be added in order to account for maneuvers and predictions. The particular flavor used in our implementation is called zero effort miss (ZEM) and tries to minimize the miss distance at a certain point of time in the future.

$$\begin{aligned} z &= \mathcal{L} + v_{rel} * t_{tgo} \\ z_\perp &= \mathcal{L} - dot(\mathcal{L}, z) * z \\ \gamma_{ZEM} &= N_{nav} * z_\perp / t_{tgo} \end{aligned} \tag{1}$$

The basic form of ZEM is shown in equation 1. The miss distance z, that is the distance by which the pursuer would miss without corrections (i.e. zero effort), is calculated from the LOS $\mathcal{L}$, relative velocity v_{rel}, and estimated time-to-go t_{tgo}. The steering command γ_{ZEM}, which is supposed to compensate the miss distance, is then calculated as an acceleration *perpendicular* to the ZEM and towards the LOS ($z_\perp$). The navigation factor for overcompensation is denoted as N_{nav}.

2) Close Distance (PID): While ProNav is excellent for hitting a target based on its current velocity vector, in our use case, a collision is undesirable; instead, at close range we blend into an "escort behavior" which tries to mimic the target's movements while moving along the LOS towards the control law's reference point (the final point will of course be the actual docking interface).

$$\gamma_{PID} = \dot{\vec{v}}_{rel} * N_{nav} + d_f * \vec{\mathcal{L}}_R \tag{2}$$

As shown in equation 2, the steering command γ_{PID} is equal to the relative acceleration $\dot{\vec{v}}_{rel}$ between the DS and the TAUV multiplied by a navigation factor N_{nav} which has the same purpose as in ProNav above. We add an additional acceleration term towards the reference point $\vec{\mathcal{L}}_R$, which is subject to a PID controller that calculates d_f based on the distance error.

Even though this approach offers little in the way of predictive maneuvering, successful docking can be achieved for three reasons: 1. channel latency decreases with the distance between DS and TAUV, allowing to react faster to deviations; 2. more accurate odometry estimates become available at close distance; and 3. the maneuverability (i.e. acceleration advantage) of the DS is high compared to the TAUV.

3) Close Distance (MPC): During physical trials, we found that the above PID controller sometimes struggled to follow the waylines. This is because our guidance laws generate acceleration commands; however, our PID controller is not predictive, meaning that it will generate small accelerations even when close to the wayline, causing the DS to overshoot its target. Although docking could still succeed given enough time, we decided that a more robust approach with stronger predictive capabilities would be desirable.

To this end, we implemented a model-predictive control (MPC) controller as an alternative close-range docking behavior. The controller predicts future poses of both the DS and TAUV by extrapolating their current dynamics S. The cost J of each state is then the weighted average of distance error e_{dist} and relative velocity error e_{vrel} as shown in equation 3.

$$\begin{aligned} J &= f_{dist} * e_{dist} + f_{vrel} * e_{vrel} \\ \gamma_{MPC} &= \min_J(S_{DS}, S_{TAUV}) \end{aligned} \tag{3}$$

f_{dist} and f_{vrel} are user-defined parameters and were set empirically. We then employ scipy's sequential least squares

programming algorithm to minimize the cost over the extrapolated state field. The selected steering command γ_{MPC} is the first command of the discovered minimal cost state sequence.

V. Simulation

Docking trials at sea involve a considerable amount of logistics. A simulation environment is therefore a useful tool to evaluate both mission execution and algorithm performance well ahead of time. Multi-robot interaction, communication and even adverse environmental influences such as water currents can be explored, albeit with limited physics realism.

A. Setup

As all software components involved in the docking process are implemented as ROS [14] nodes, the simulation environment of choice is Gazebo [15], extended by features such as hydrodynamics provided by the UUV Simulator [16]. The 3D models used in our simulation are derived from the CAD models used for the manufacturing of the real-world AUVs, including virtual counterparts of thrusters, rudders and fins. The simulation models are shown in figure 6.

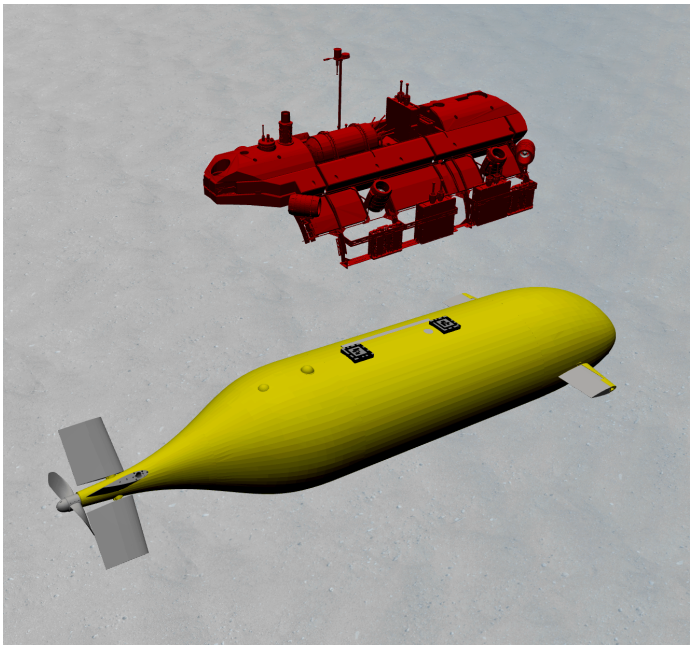


Fig. 6: The TAUV (bottom) and DS (top) in the Gazebo simulation environment.

In a real-world scenario, each platform (DS, TAUV, SV) would run its own ROS core, isolating its own set of ROS nodes from the other vehicles, with exceptions only in the form of communication between select nodes required for the negotiation of the docking process and the exchange of small telemetry data packets. To replicate this setup of three sparsely interconnected ROS cores, a multi-master implementation transferring only selected topics between the participants is used [17]. All messages used in our docking protocol (see IV-A) are transmitted between the three vehicles by means of the multi-ROS-core topic interlink. Notably, this still uses the communication abstraction layer discussed in section IV-B for message exchange between the DS and TAUV. Since the DS and SV would be tethered, their communication is more direct.

With this setup, our docking trials in the simulation environment are somewhat simplified: although all subsystems like route planning, navigation, drivers, etc. are present, the communication between vehicles is free of delays and latencies. The world-fixed WGS-84 coordinates of both the TAUV and the DS are also exactly known. A transmission of the vessel frames via telemetry messages is thus not required for simulated docking. While we expect these to become relevant factors in future physical trials, we assume their contribution to be binary in that our docking approach cannot succeed if we do not have a reliable communication channel or good pose estimates.

B. Scenarios

We start each simulation by placing the DS roughly 30 m away from the TAUV (offset on all axes). The TAUV has three waypoints attached: (1, ZEM) 4m above and slightly behind the TAUV, (2, PID) 2 m above the docking interface, and (3, PID) inside the docking interface. The MPC controller was not fully evaluated yet at the time of writing and will be reported on in the future.

We evaluate our docking algorithm for the following cases:

- Constant velocity: the TAUV is driving straight and with constant speed.
- Varying forward speed: the TAUV is driving straight, but its speed oscillates.
- Curved trajectory: the TAUV moves with a constant turn rate.
- Water currents: the TAUV is trying to drive straight but experiences lateral forces.

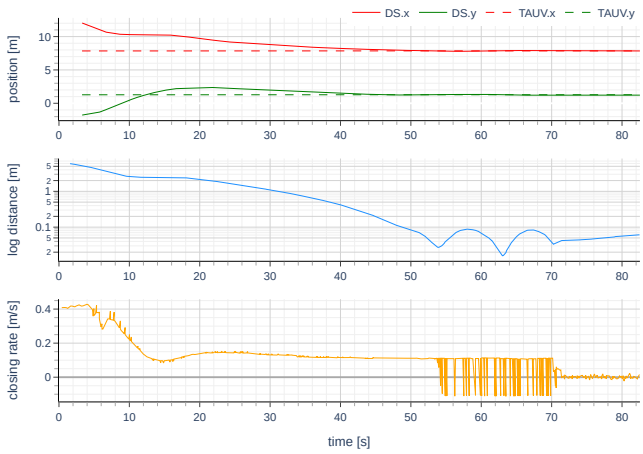


Fig. 7: Result of a docking experiment in simulation where the TAUV moves forward in a straight line with constant speed.

1) Constant Velocity: For our baseline we give the TAUV a simple mission: *drive in a straight line at a given water*

depth with constant forward speed. The results in figure 7 show that the DS initially requires time to assume its course of pursuit provided by the ZEM guidance law. It then closes on the TAUV with a constant rate of around 0.3 m/s. Once the guidance law mixture transitions into our PID escort mode, the closing rate shows a much more dynamic pattern. This is due to the various controlled axes of the DS interacting while matching both orientation and speed of the TAUV. We expect that further controller tuning would reduce these oscillations. However, the simulation does not capture the real-world dynamics of our vehicles with sufficient accuracy. Since our distance metric shows that the DS stays well within the required tolerances, we forwent fine-tuning our simulation parameters to perfection.

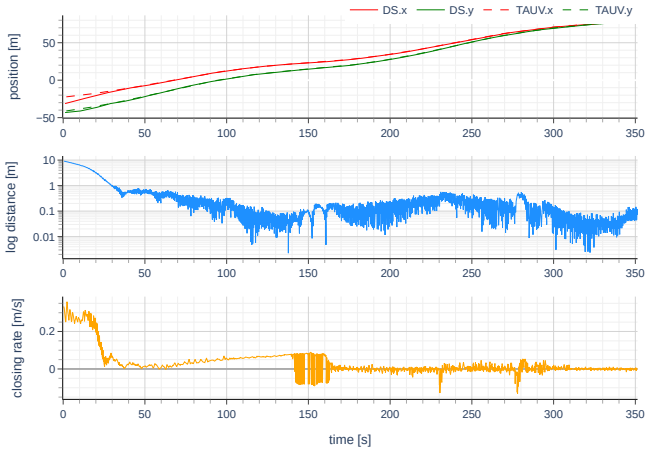


Fig. 8: Result of a simulation run where the TAUV forward speed varies between 0.2 m/s and 0.7 m/s.

2) Varying Forward Speed: For this simulation scenario the TAUV forward speed was set to an initial 0.5 m/s and then changed in steps of $\pm$ 0.02 m/s ranging from 0.2 m/s to 0.7 m/s. Figure 8 clearly shows that the approach algorithm requires additional time to adapt to the changing TAUV dynamics, but can eventually still dock with the TAUV successfully.

3) Curved TAUV trajectory: In this simulation run, the TAUV is not following a straight path, but its tail rudders are set for maximum torque on the yaw axis. This causes the TAUV to follow a curved trajectory, resulting in a roughly circular path.

Figure 9 shows the result of this simulation scenario: especially during the final approach in the escort guidance law, the distance decrease between DS and TAUV is much noisier. More corrections of the DS' linear and angular velocities are required to match the TAUV's motion, the PID controllers proportional and integral components oscillate more than in the straight path scenario.

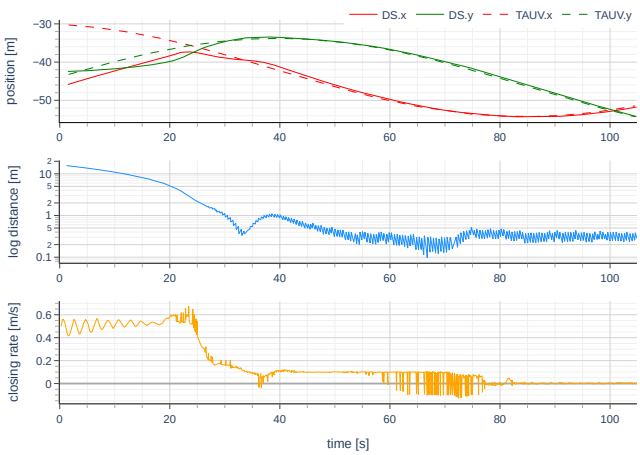


Fig. 9: Result of a simulation run where the TAUV trajectory is not straight, but curved towards port.

4) Lateral water currents: To evaluate the applicability of our docking approach also under adverse environment conditions, a simulation scenario with lateral water currents was used.

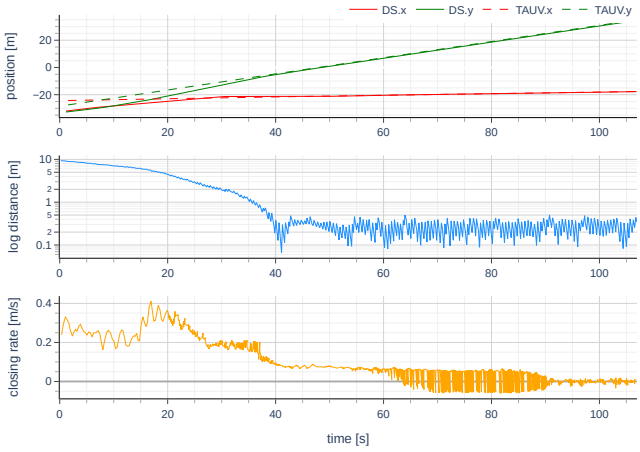


Fig. 10: Result of a simulation run where both TAUV and DS are affected by lateral water currents.

Both the TAUV and the DS experience a lateral current of 0.5 m/s. While the TAUV still tries to traverse its straight-line trajectory, it is laterally offset by the water current. This results in constant yaw angle adjustments. While the DS experiences the same water current, its shape produces a lateral force different from the one the TAUV experiences. Also, the DS is able to move along its sway-axis without requiring yaw angle adjustments.

Figure 10 shows that the closing rate varies over time much stronger than in the previous scenarios. Both the ZEM and the escort guidance laws produce more oscillations in their control output than in the case of no external water current disturbance. However, docking can still be successfully achieved under these conditions.

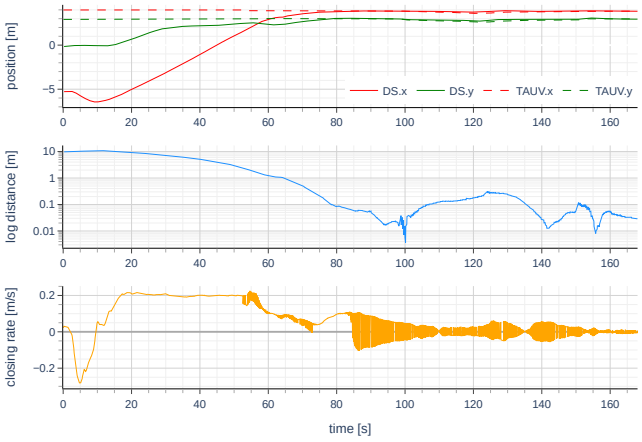


Fig. 11: Data recorded in our real-world tests.

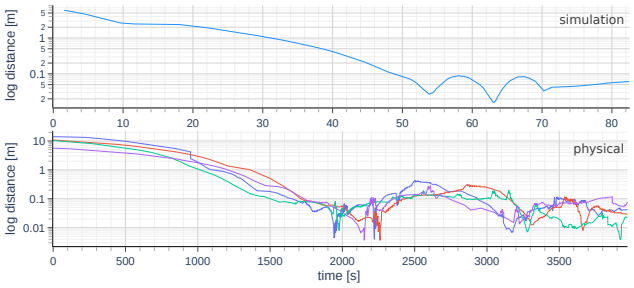


Fig. 12: Comparison between simulated (top) and physical (bottom) trials with stationary TAUV.

VI. Physical Trials

We conducted multiple physical tests over the course of two weeks at the DFKI Robotics Innovation Center, Germany using the mobile DS and TAUV described in section III. All experiments were conducted using the ZEM and escort PID laws, with a waypoint setup reflecting that of the simulation. Due to the size of the vehicles, the experiments could only be executed with the TAUV resting; additional tests in open waters are scheduled for later this year. Some snapshots from our experiments can be seen in Fig. 1. A summary video is available at https://youtu.be/hwrcJu0mtdk.

During our tests we ran more than 30 successful docking maneuvers fully automated. Fig. 11 shows a plot from a typical approach. Crucially, we note that the absolute distance from the final goal point is less than 5 cm, which for our AUVs is required for the coupling mechanism to engage.

Fig. 12 shows a side-by-side comparison of the simulated logarithmic distance between the DS and the TAUV in the stationary case and a selection of recordings from the physical trials. Note that for this plot the horizontal axis of the physical trials is in sample steps, which was done to align the trials' recordings. We note that the overall shape and progression appear to be similar, with an initial far-range approach followed by two hills representing close-range navigation patterns. Furthermore, the physical trials are more drawn out and experience several sharp spikes. We attribute this to interactions of the thrusters with the environment not covered in the simulation, as well as different controller parameters.

We recorded data for 18 of our trials, 13 of which we consider to be successful based on the achieved steady state distance, indicating a success rate of around 72%. The successful runs achieved a mean distance of 0.066 cm with a standard deviation of 0.04 cm. However, to protect our AUVs during these first trials, we had attached about 3 cm of foam to the underside of the DS, preventing full engagement and adding to the distance. The simulation also showed that the performance is very sensitive to the PID controller parameters. We therefore assume that our physical setup is able to achieve a similar steady state as the simulation of ≤ 5 cm. This will be validated in future physical trials, including open sea tests.

VII. Conclusion & Outlook

Retrieval of AUVs is a challenging and critical element in any open waters mission, especially in rough sea states. To circumvent this issue, we present a highly maneuverable docking station that is able to "catch" and engage the target AUV while it is in motion. Our approach consists of two parts: (1) a custom protocol to negotiate docking modalities, which is made possible by varying means of communication (e.g. acoustic). And (2) a homing algorithm that blends navigation commands from two guidance laws to achieve a smooth and controlled approach based on filtered pose estimates and a custom distance metric. We validated our approach for various simulated scenarios (AUV stationary, in motion, turning, lateral currents) and in physical trials (AUV stationary), and achieved reliable couplings with a deviation of less than 5 cm. Additional trials in open waters are planned for later this year.

References

[1] J. Albiez, S. Joyeux, C. Gaudig, J. Hilljegerdes, S. Kroffke, C. Schoo, S. Arnold, G. Mimoso, P. Alcantara, R. Saback, J. Britto, D. Cesar, G. Neves, T. Watanabe, P. Merz Paranhos, M. Reis, and F. Kirchnery, "FlatFish - a compact subsea-resident inspection AUV," in *OCEANS 2015 - MTS/IEEE Washington*, IEEE, Oct. 2015.

[2] L. Christensen, J. Hilljegerdes, M. Zipper, A. Kolesnikov, B. Hulsen, C. E. S. Koch, M. Hildebrandt, and L. C. Danter, "The hydrobatic dual-arm intervention AUV Cuttlefish," in *OCEANS 2022, Hampton Roads*, IEEE, Oct. 2022.

[3] L. Persson and B. Wahlberg, "Variable prediction horizon control for cooperative landing on moving target," in *2021 IEEE Aerospace Conference (50100)*, IEEE, Mar. 2021.

[4] C. Liang, X. Luo, X. Chen, and B. Han, "Route planning of truck and multi-drone rendezvous with available time window constraints of drones," *Science China Technological Sciences*, vol. 65, pp. 2190—-2204, Aug. 2022.

[5] M. C. Nielsen, T. A. Johansen, and M. Blanke, "Cooperative rendezvous and docking for underwater robots using model predictive control and dual decomposition," in *2018 European Control Conference (ECC)*, IEEE, June 2018.

[6] T. Creutz, B. Wehbe, S. Arnold, and M. Hildebrandt, "Towards robust autonomous underwater docking for long-term under-ice exploration," in *OCEANS 2023 - Limerick*, IEEE, June 2023.

[7] M. Hildebrandt, L. Christensen, and F. Kirchner, "Combining cameras, magnetometers and machine-learning into a close-range localization system for docking and homing," in *OCEANS 2017 - Anchorage*, pp. 1–6, 2017.

[8] M. Patil, B. Wehbe, and M. Valdenegro-Toro, "Deep reinforcement learning for continuous docking control of autonomous underwater vehicles: A benchmarking study," in *OCEANS 2021: San Diego – Porto*, IEEE, Sept. 2021.

[9] B. Blum-Thomas, I. Cebulla, and S. Ritz, "Shape design of a towed subsurface docking station for enhanced auv docking maneuvers under hydrodynamic considerations." unpublished.

[10] P. Kampmann, C. Gaudig, F. Nauert, M. Fritsche, and T. Johannink, "A software architecture for resilient long term autonomous missions of AUVs," in *2022 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, pp. 1–6, IEEE, 2022.

[11] V. Verma, T. Estlin, A. Jonsson, C. Pasareanu, R. Simmons, and K. Tso, "Plan execution interchange language (plexil) for executable plans and command sequences," in *International symposium on artificial intelligence, robotics and automation in space (iSAIRAS)*, 01 2005.

[12] D. B. dos Santos Cesar, C. Gaudig, M. Fritsche, M. A. dos Reis, and F. Kirchner, "An evaluation of artificial fiducial markers in underwater environments," in *OCEANS 2015 - Genova*, pp. 1–6, 2015.

[13] N. F. Palumbo, R. A. Blauwkamp, and J. M. Lloyd, "Basic principles of homing guidance," *Johns Hopkins APL Technical Digest*, vol. 29, no. 1, pp. 25–41, 2010.

[14] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Workshops at the IEEE International Conference on Robotics and Automation*, 2009.

[15] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (Sendai, Japan), pp. 2149–2154, Sep 2004.

[16] M. M. M. Manhães, S. A. Scherer, M. Voss, L. R. Douat, and T. Rauschenbach, "UUV simulator: A Gazebo-based package for underwater intervention and multi-robot simulation," in *OCEANS 2016 MTS/IEEE Monterey*, IEEE, sep 2016.

[17] A. Tiderko, F. Hoeller, and T. Röhling, "The ROS multimaster extension for simplified deployment of multi-robot systems," *Robot Operating System (ROS) The Complete Reference (Volume 1)*, pp. 629–650, 2016.