IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 30, 2021, accepted May 16, 2021, date of publication May 26, 2021, date of current version June 22, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3083883

Docking of Non-Holonomic AUVs in Presence of Ocean Currents: A Comparative Survey

JOAN ESTEBA, PATRYK CIEŚLAK, NARCÍS PALOMERAS, (Member, IEEE), AND PERE RIDAO, (Member, IEEE)

Computer Vision and Robotics Research Institute (VICOROB), University of Girona, 17003 Girona, Spain

Corresponding author: Joan Esteba (joan.esteba@udg.edu)

This work was supported by the European Union's Horizon 2020 Research and Innovation Program through the ATLANTIS "The Atlantic Testing Platform for Maritime Robotics: New Frontiers for Inspection and Maintenance of Offshore Energy Infrastructures" Project under Grant 871571.

ABSTRACT This paper presents a comparative study of docking algorithms intended for non-holonomic autonomous underwater vehicles, docking in funnel-shaped docking stations, operating under the influence of ocean currents. While descriptive surveys have been already reported in the literature, our goal is to compare the most relevant algorithms through realistic Monte Carlo simulations to provide an insight into their performance. To this aim, a new numerical performance indicator is proposed, which, based on the geometry of the manoeuvre, is able to characterize a successful or unsuccessful docking, providing a metric for comparison. The experimental study is carried out using hardware-in-the-loop simulation by means of the Stonefish simulator, including the dynamic/hydrodynamic model of the Sparus II AUV, models of all internal and external sensors, and the collision geometry representing the docking station.

INDEX TERMS Docking, AUV, ocean currents, non-holonomic.

I. INTRODUCTION

In recent years, underwater robotic technologies have been used in several economic sectors, like oil and gas industry, offshore wind energy generation, scientific research, etc. Nowadays, it is common to use Remotely Operated Vehicles (ROV) to perform tasks such as manipulation and deployment of structures [1], visual inspection and non-destructive testing, core sampling, etc. Moreover, Autonomous Underwater Vehicles (AUV) are being consolidated in this field and starting to be considered a mature technology [2], [3]. This fact encourages the development of complementary technologies to solve some of their current limitations: limited communications bandwidth when the vehicle is submerged (i.e., when relying only on acoustic channels) and highly limited autonomy, resulting from the weight of batteries. These are two of the main concerns that AUVs have to address when facing missions that require persistent autonomy. To mitigate these issues, a concept has been proposed by several researchers, to endow AUVs with the capacity to persistently operate over an area: the use of docking stations (DS). A DS provides the AUV with a protected environment, allows it to recharge its batteries, and can include high-bandwidth communication channels, to transfer huge amounts of data. In the literature, several examples of DS systems can be found ([4]–[9]). Each DS concept was tailored to a specific AUV and used its own perception and docking strategy. Several descriptive surveys about docking can be already found in the literature: [10]–[12].

In a previous work, [13] the University of Girona developed a prototype of a DS consisting of a fixed funnel equipped with an acoustic transponder and a set of light beacons. The experience obtained while validating this design showed us that the control strategy used to dock was insufficient to deal with severe ocean currents. Furthermore, reliance on a vision system presented problems in very turbid water scenarios. Building from this previous experience, this article compares several docking algorithms, already established in the literature, and how they behave in the presence of ocean currents. All methods have been adapted to use a non-holonomic AUV, a fixed funnel DS, and an ultra-short baseline (USBL) system to localize the DS with respect to the AUV. Extensive tests, based on dynamic simulations [14], have been performed to find the methods' strengths and weaknesses.

The paper is organized as follows: Section II explores the state of the art of docking system designs and docking

The associate editor coordinating the review of this manuscript and approving it for publication was Tao Liu.

algorithms developed for non-holonomic AUV, able to deal with ocean currents. Section III describes selected methods and how they have been adapted to the proposed setup. Section IV presents the tools used to develop and evaluate the experiments, and Section V introduces the simulation setup. Obtained results are shown in Section VI, discussed in Section VII, and the conclusions are presented in Section VIII.

II. STATE OF THE ART

A thorough review of the literature on autonomous docking was carried out, focusing especially on those methods that take into account the presence of ocean currents. Starting with the references of one of the most comprehensive surveys in the field [11], all cited articles that meet our criteria were reviewed and added to the list. This exercise was repeated for all the articles in the list that were not previously reviewed. After several iterations, more than one hundred eighty publications were reviewed.

Docking systems can be classified from many different points of view. One possible classification is to differentiate holonomic and non-holonomic robots. A holonomic AUV can control all its degrees of freedom, which makes it easier to complete the docking maneuver even in the presence of ocean currents. Examples of docking of holonomic robots include systems based only on acoustics [15], or approaches that combine acoustics with vision [16], [17]. In [18] a system where the AUV reaches another vehicle using vision is presented, while [19] presents a system where an AUV docks on a submarine, with a mechanical system guided by acoustic, electromagnetic, and optical sensors. In [20] and [21] solutions based on sonar and vision technology are presented. [22] introduces an adaptive DS which is automatically leveled to maintain horizontal orientation.

The non-holonomic AUVs have a limitation on their control that makes the docking maneuver more challenging, especially in presence of ocean currents. The docking concepts for non-holonomic robots can be classified considering the capture mechanism used: pole docking, landing docking, net docking, and funnel docking.

The pole docking system consists of a vertical pole where the AUV attaches. It allows the vehicle to reach the DS from any direction (see Fig. 1), simplifying the way to deal with ocean currents. To allow the AUV to attach to the pole, some modifications to the AUV are required, for example, adding a mechanical V-shaped structure on its front. One of the drawbacks of pole systems is that they usually offer neither protection nor connection (i.e., neither power nor data link) between the AUV and the DS. Therefore, some other mechanism is needed to deliver this functionality. Examples of the pole docking concept are presented in [7], and in [23] where the acoustics (i.e., USBL system) are used to detect the DS position, in [24] where vision (i.e., camera system using markers) is used to detect the DS, or in [25] where both acoustics and vision are used to localize the DS (i.e., USBL system combined with a camera). There are also commercial applications for pole docking like the one reported in [26].

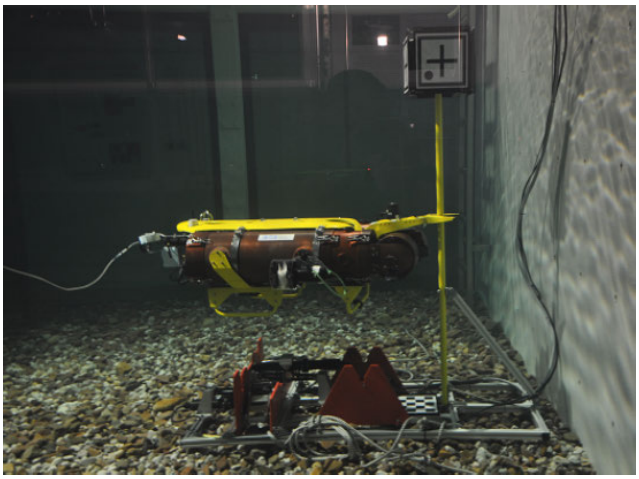


FIGURE 1. Example of pole docking [24].

FIGURE 2. Example of landing docking [28].

The concept of landing-based docking is similar to the concept of a jet landing on an aircraft carrier. It consists of a mechanical structure where the AUV lands and is attached with a mechanical appendix installed on the AUV. An example of this system appears in [27]. A similar concept is presented in [28] (see Fig. 2) where the AUV has a T-shaped appendage that docks to a V-shaped structure. The problem with this system is that it forces the AUV to approach the DS from a specific direction, which can be problematic in the presence of ocean currents.

The net docking system is presented in [29]. This concept is designed to be a launch and recovery system but is not intended for permanent deployment (see Fig. 3). It consists of a net with an acoustic sensor located in its center, the AUV attaches to the net using a hook added to its nose. Like pole docking, the AUV can choose the direction of the approach, but with the advantage of having the optimal sensor placement (i.e., a USBL can be placed exactly at the position where the AUV should arrive).

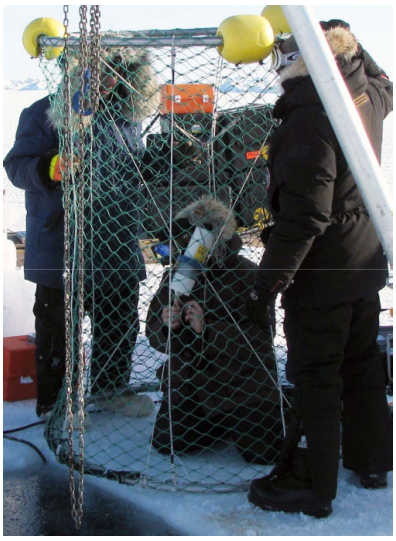


FIGURE 3. Example of net docking [29].

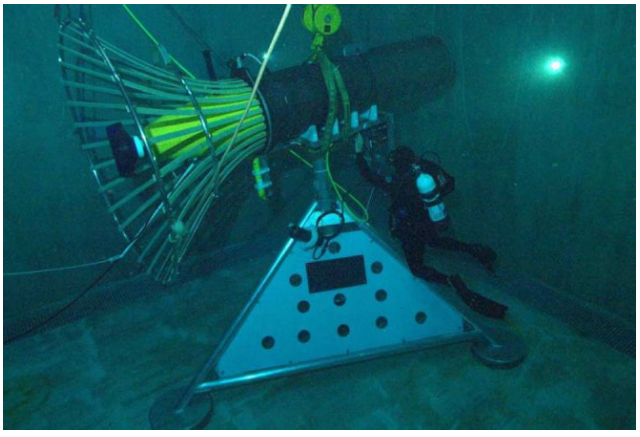


FIGURE 4. Example of funnel docking [4].

The funnel docking system is probably the most popular one (see Fig. 4). The mechanical structure of the DS has a shape of a funnel to help the AUV to enter it. The main advantages of this concept are that it allows long-term deployment (i.e., it can offer protection, charging, and data transmission), and there is no need to add mechanical parts to the AUV; for these reasons, this article is focused on funnel-shaped docking systems. Examples where funnel docking is used are reported in [4], [8], [9], [30], and [31]. Like the landing docking system, this configuration has a potential problem with ocean currents because the AUV is required to achieve a particular orientation to enter the DS.

Several strategies to deal with ocean currents with a non-holonomic AUV, when using a funnel-shaped DS, have been published. In [4] the AUV assumes a crab angle to compensate for the ocean currents. In [5], a DS able to control the funnel orientation is presented, to improve the crab angle solution. In [32] a fuzzy controller is applied to perform the docking maneuver. In [31], the AUV heads towards the DS with a crab angle and when it almost reaches the DS, it suddenly changes the heading to be parallel to the DS axis. In [33] and [34] it is proposed to follow a path with some offset, with respect to the DS axis, and use the ocean current to correct this error, in order to enter parallel to the DS.

III. METHODOLOGIES

Seven methods, described in the literature, have been implemented and compared in this article. A unified setup has been defined, including a funnel-shaped DS, a non-holonomic AUV, and a USBL to detect the DS position with respect to the AUV. The methods have been slightly adapted when necessary to make them compatible with the proposed setup. All the methods generate only a heading reference and a surge velocity reference for the AUV, assuming that a low-level controller will transform them into thruster setpoints [35]. The problem is tackled as a 2D problem (i.e., xy plane) assuming that the z component (i.e., depth or altitude with respect to the bottom) is known and can be independently controlled for an AUV, like the one proposed in Section IV-A1.

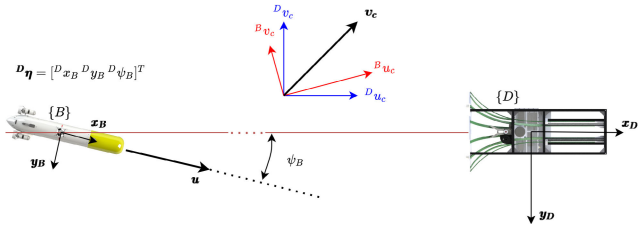


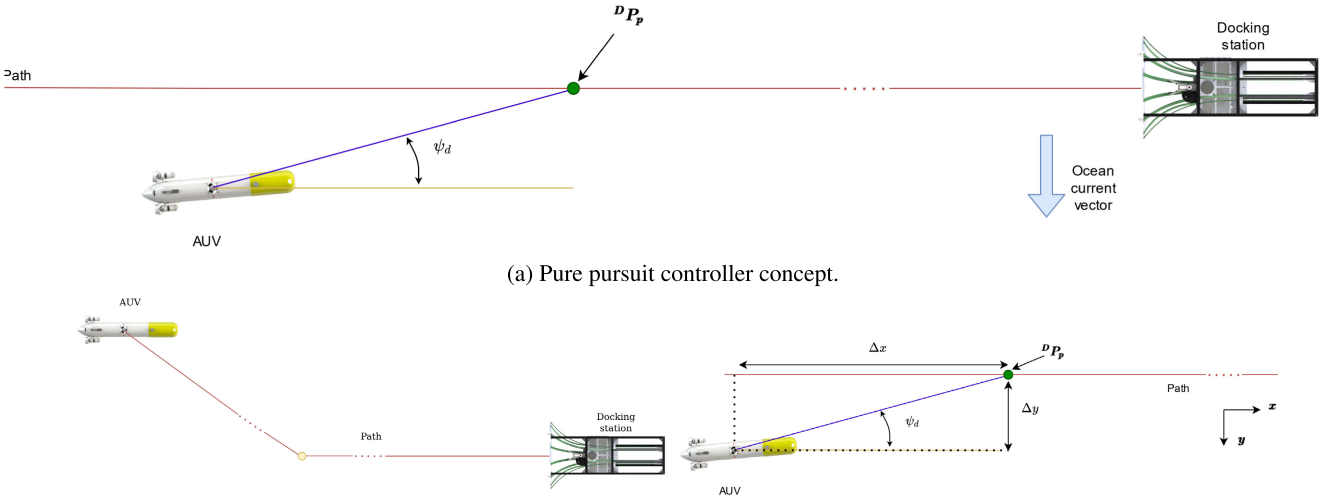
FIGURE 5. Basic variables representation.

In the following subsection, the studied docking algorithms are described.

In Fig. 5 the basic variables involved in the process are described. Two different reference frames are presented, the $\{D\}$ frame located at the position of the DS, and the $\{B\}$ frame at the position of the AUV. The ocean current vector ($\boldsymbol{\nu_c}$) is represented for both systems. In the $\{D\}$ frame the pose of the DS is represented as $^{\boldsymbol{D}}\boldsymbol{P_D} = [^{D}x_D\ ^{D}y_D\ ^{D}z_D]^T = [0\ 0\ 0]^T$. The AUV is represented in the $\{D\}$ frame as three components: $^{\boldsymbol{D}}\boldsymbol{\eta} = [^{D}x_B\ ^{D}y_B\ ^{D}\psi_B]^T$. The simulated robot, a Sparus II AUV, has direct control on the desired surge velocity (i.e., velocity with respect to the ground: u) and on the vehicle heading with respect to the $\{D\}$ frame (i.e., $^{D}\psi_B$).

A. PURE PURSUIT CONTROLLER

This method is based on the pure pursuit controller [36]. Despite having been used by several authors [5], [28], [37] it is not designed to deal with ocean currents. However, it has been included in this article as a baseline to which the others are compared. The method is based on following a linear path, centered on the DS (see Fig. 6). At each time step, the AUV computes the heading reference ψ_d that moves the AUV from its current position ($^{\boldsymbol{D}}\boldsymbol{\eta}$) to reach a look-ahead point $^{\boldsymbol{D}}\boldsymbol{P_p}$, on the path, in front of the robot. The look-ahead distance Δx

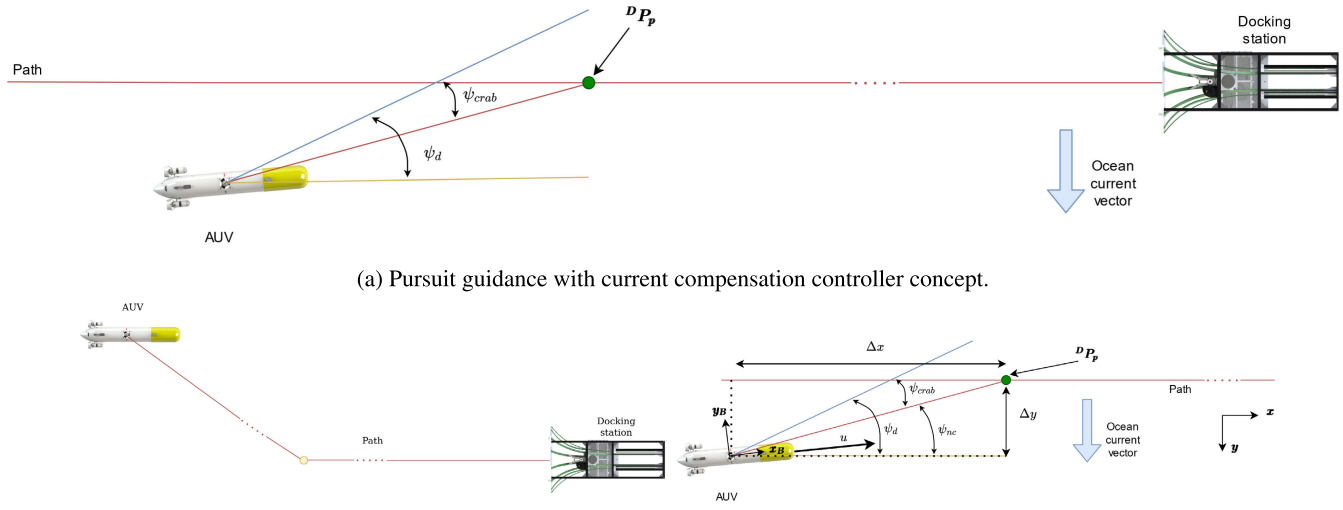


(a) Pure pursuit controller concept.

(b) Pure pursuit controller path concept.

(c) Pure pursuit controller heading concept.

FIGURE 6. Pure pursuit controller representation.



(a) Pursuit guidance with current compensation controller concept.

(b) Pursuit guidance with current compensation controller path concept.

(c) Pursuit guidance with current compensation controller heading concept.

FIGURE 7. Pursuit guidance with current compensation controller representation.

is defined on the x axis of the $\{D\}$ frame and must be tuned:

$$^{D}\boldsymbol{P_p} = [^{D}x_p \; 0]^T, \tag{1}$$

$$\Delta x = {}^{D}x_p - {}^{D}x_B, \quad \Delta y = {}^{D}y_B. \tag{2}$$

The desired heading of the AUV to reach $^{D}\boldsymbol{P_p}$ is calculated following (see Fig. 6c):

$$\psi_d = \text{atan2}(\Delta y, \; \Delta x). \tag{3}$$

The AUV surge velocity (u) is considered constant and set to the desired docking velocity (u_{dock}). The pure pursuit method guarantees the AUV docking velocity but not its heading.

B. PURSUIT GUIDANCE WITH CURRENT COMPENSATION CONTROLLER

The *Pursuit guidance with current compensation controller* (PGCC controller) is based on the idea of applying a correction of the desired heading of the AUV (i.e., introducing a crab angle) in order to compensate for the ocean currents. This concept has been presented by different authors: [5], [38]. The method (see Fig. 7) computes the reference heading for the AUV, adding a corrective crab angle to the default heading that matches the path bearing.

The desired heading and the linear velocity of the AUV are calculated to compensate for the ocean currents. First, the ocean current vector is estimated with respect to the robot's local frame $\{B\}$. Then, the through-water surge

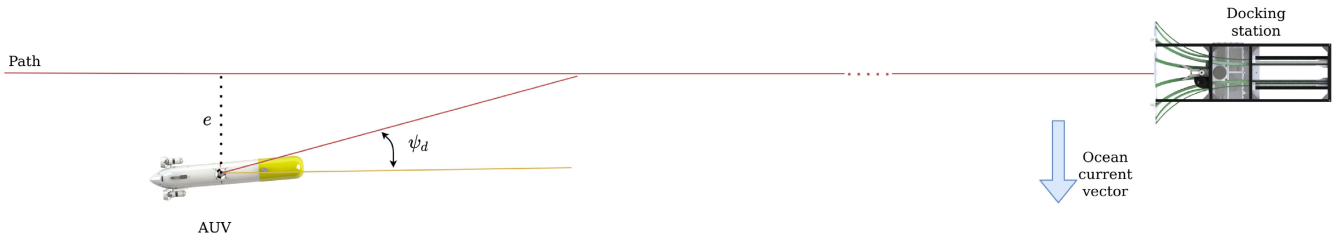


FIGURE 8. Cross-track controller representation.

velocity of the AUV is calculated:

$$^{B}u_{B,w} = u_{dock} - ^{B}u_c. \tag{4}$$

Knowing the velocity with respect to the water $^{B}u_{B,w}$ and knowing the y component of the ocean current on the $\{B\}$ frame $^{B}v_c$, the crab angle ψ_{crab} can be calculated as follows:

$$\psi_{crab} = \text{atan2}(-^{B}v_c, \ ^{B}u_{B,w}), \tag{5}$$

limiting the maximum angle to 90°. As in the previous method, the controller tries to reach a point $^{D}\boldsymbol{P_p}$ in front of the vehicle, computed using (1). To reach it, a heading is defined, without considering the ocean currents:

$$\psi_{nc} = \text{atan2}(\Delta y, \ \Delta x). \tag{6}$$

The desired heading ψ_d is defined as the sum of both ψ_{crab} and ψ_{nc}, see Fig. 7c:

$$\psi_d = \psi_{nc} + \psi_{crab}. \tag{7}$$

The AUV surge velocity u is considered constant and set to the desired docking velocity u_{dock}. As well as the previous method, this method guarantees the AUV docking velocity but not the heading.

C. CROSS-TRACK CONTROLLER

The strategy of this method is based on applying a correction to the AUV heading, in order to compensate for the cross-track error, that appears due to the lateral current. This method is used in [4], [39].

A path co-linear with the center-line of the DS is defined and the cross-track error e is obtained as the difference between the y component of the path (that is zero) and the position of the AUV (see Fig. 8):

$$e = ^{D}y_B. \tag{8}$$

The desired heading ψ_d is obtained with a PID control law, acting over e, which uses an integral anti-windup limit, see [40]. As in the previous methods, u is considered constant and set to u_{dock}. This method guarantees the AUV docking velocity but not the heading.

D. FUZZY CONTROLLER

This method is based on [32] and [41]. It calculates the heading and the linear velocity of the robot, following the fuzzy rules presented in [41] (Fig. 9). It calculates the associated weights of the fuzzy rules according to Fig. 9b and the memberships functions presented in Fig. 9c and Fig. 9d to obtain the heading (Table 1) and the linear velocity of the AUV (Table 2). For the experiment set in this paper, not all the sections are used. To better understand which ones are used, see the green sections represented in Fig. 9b and the set of the experiments presented in Section. V.

Like the previous methods, this method guarantees the AUV docking velocity but not the heading.

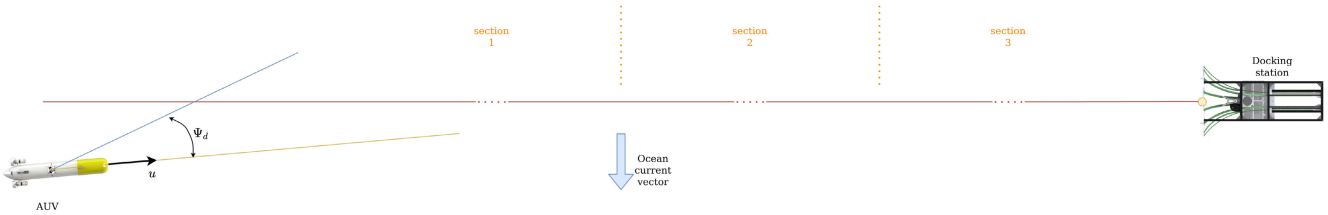
E. TOUCHDOWN ALIGNMENT CONTROLLER

The method showed in [31] is used to fix one of the possible problems that appear when using the pure pursuit, the cross-track, the PGCC, or the fuzzy controllers, that is reaching the DS without being parallel to it. In this method, the AUV executes a cross-track controller but when it reaches a position just in front of the DS the AUV heading is abruptly changed to align it with the DS before *touchdown*.[1] In the original article direct control of the AUV rudder is used for this purpose. However, because the Sparus II AUV does not have a rudder, in this article the $^{D}\psi_B$ is suddenly modified to emulate this behavior.

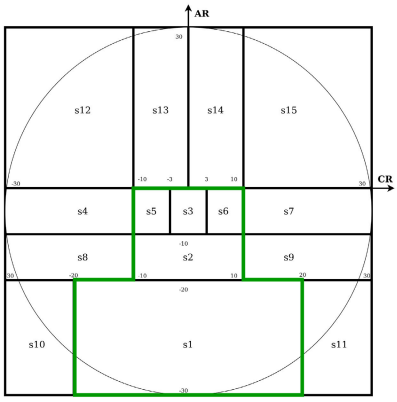
The method follows two steps (see Fig. 10):

- The AUV follows the DS center-line path using a cross-track controller. A conventional PID with an integral anti-windup is used to compute the crab angle β that reduces the cross-track error to zero (as in the cross-track controller).
- Before touchdown, the crab angle must be eliminated. According to the maximum yaw rate (r_{max}), β, and u_{dock}; the distance with respect to the DS (d_f), at which the AUV heading must be abruptly changed, must be computed. The original authors propose to use a cosine curve described by (10) to change the reference heading during the final alignment (see Fig. 10b).

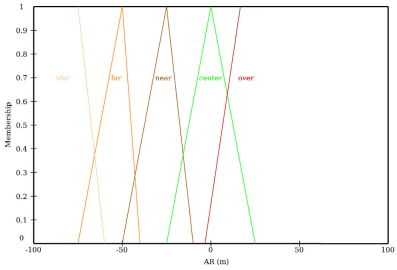
[1] According to Park, *touchdown* means the contact of the AUV and the docking station. The word comes from airplane flight operation.



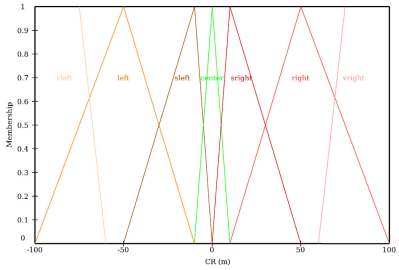
(a) Fuzzy controller concept.



(b) Fuzzy controller sections.



(c) Fuzzy controller membership AR.



(d) Fuzzy controller membership CR.

FIGURE 9. Fuzzy controller representation.

TABLE 1. Fuzzy controller ψ_d related to the section.

Heading reference related to the section		
$\psi_1 = -8\,CR$	$\psi_6 = -90$	$\psi_{11} = -1.5\,AR - 125$
$\psi_2 = -12\,CR$	$\psi_7 = -175$	$\psi_{12} = 0.4\,CR - 160$
$\psi_3 = -15\,CR$	$\psi_8 = 0.4\,CR + 150$	$\psi_{13} = -60$
$\psi_4 = 175\,CR$	$\psi_9 = 0.4\,CR - 150$	$\psi_{14} = 0.4\,CR + 160$
$\psi_5 = 90$	$\psi_{10} = 1.5\,AR - 125$	$\psi_{15} = 60$

TABLE 2. Fuzzy controller u_d related to the section (u_{dock} corresponds to docking entrance velocity).

Linear velocity reference related to the section		
$u_1 = u_{dock}$	$u_6 = 0.85\,u_{dock}$	$u_{11} = 0.7\,u_{dock}$
$u_2 = u_{dock}$	$u_7 = 0.7\,u_{dock}$	$u_{12} = 0.7\,u_{dock}$
$u_3 = u_{dock}$	$u_8 = 0.7\,u_{dock}$	$u_{13} = 0.7\,u_{dock}$
$u_4 = 0.7\,u_{dock}$	$u_9 = 0.7\,u_{dock}$	$u_{14} = 0.7\,u_{dock}$
$u_5 = 0.85\,u_{dock}$	$u_{10} = 0.7\,u_{dock}$	$u_{15} = 0.7\,u_{dock}$

The distance between the AUV and the DS is defined as d, keeping in mind that the position of the DS is the origin of the $\{D\}$ frame:

$$d = -{}^{D}x_B. \tag{9}$$

The desired heading of the AUV in the last part of the maneuver is defined as a cosinusoidal function of the distance:

$$\psi_d = -\frac{\beta}{2}\cos\left(\pi\frac{d}{d_f}\right) + \frac{\beta}{2}. \tag{10}$$

Then, given the cosinusoidal heading the minimum distance ($d_{f,min}$) required to perform the maneuver is given by:

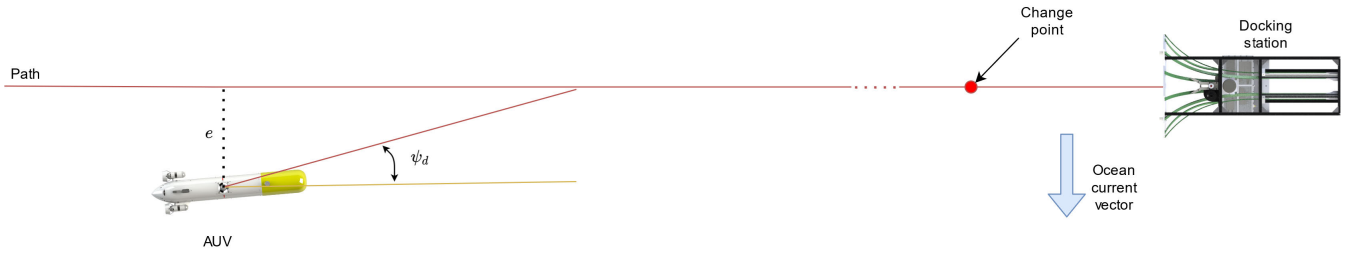
$$d_{f,min} = \left|\frac{\beta}{2}\frac{\pi}{r_{max}}u_{dock}\right|. \tag{11}$$

To have some safety margin the distance where the change point is set is defined as:

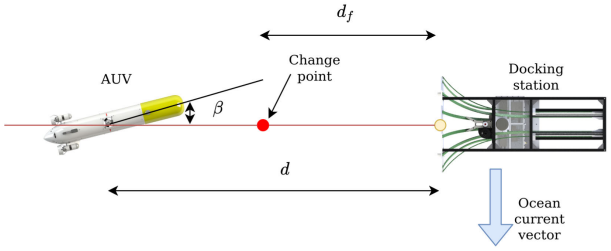
$$d_f = k\,d_{f,min}. \tag{12}$$

where k corresponds to a gain.

This method allows controlling both the AUV docking heading and forward velocity.

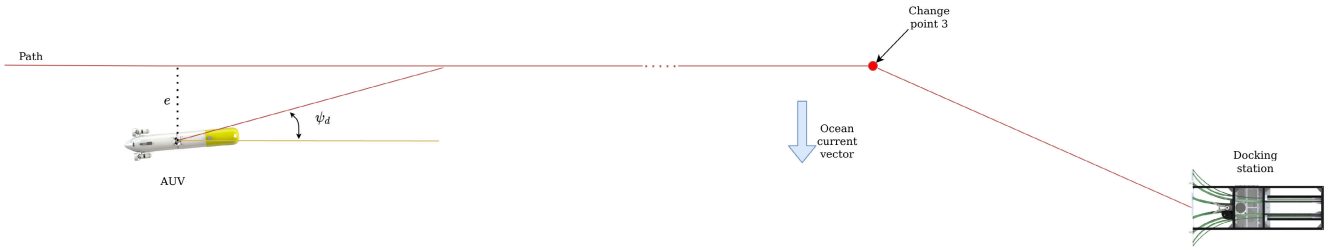


(a) Touchdown alignment controller concept.

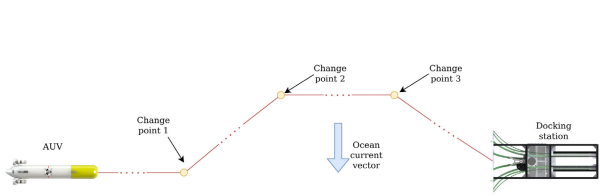


(b) Touchdown alignment controller final maneuver concept.

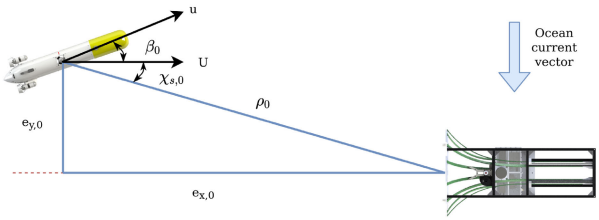
FIGURE 10. **Touchdown alignment controller representation.**



(a) Sideslip controller concept.



(b) Sideslip controller path concept.



(c) Sideslip controller final maneuver concept.

FIGURE 11. **Sideslip controller representation.**

F. SIDESLIP CONTROLLER

Park *et al.* [33] propose an alternative methodology to deal with the problem of reaching the DS with the desired angle. In this method, the authors create a path parallel to the DS axis at some distance from the docking station. Once the AUV reaches a specific point, a maneuver to move the robot from this path to the DS center-line, while keeping the vehicle parallel to the DS, is performed.

This methodology consists of three steps (see Fig. 11):

- First the AUV approaches the DS using the cross-track controller to compensate for the ocean current.
- When the necessary crab angle (β), to compensate the ocean currents, is achieved (*change point 1*), the vehicle accelerates keeping the same heading (i.e., ${}^{D}\psi_B = \beta$) until the cross-track error is around 3 m (*change point 2*) on the side from where the current is coming (see Fig. 11b).
- The vehicle follows the path parallel to the DS axis until it is around 20 m in front of the DS (*change point 3*). At this moment the AUV heading reference ψ_d is gradually changed from β to the DS angle (${}^{D}\psi_D = 0$) that is the desired heading, that the vehicle has to

achieve when it reaches the DS, following (13) to (21), see Fig. 11c.

$$e_x = \rho\, cos(\chi_s), \quad e_{x,0} = \rho\, cos(\chi_{s,0}) \tag{13}$$
$$e_y = \rho\, sin(\chi_s), \quad e_{y,0} = \rho\, sin(\chi_{s,0}) \tag{14}$$
$$0 \leq w_1 \leq 1, \quad 0 \leq w_2 \leq 1,\ 0 \leq w_3 \leq 1 \tag{15}$$
$$w_1 = e_x\ /max(e_x,\ e_{x,0}) \tag{16}$$
$$w_2 = e_y\ /max(e_y,\ e_{y,0}) \tag{17}$$
$$w_3 = \rho\ /max(\rho,\ \rho_0) \tag{18}$$
$$\alpha = (1 - w_2)\,\beta_0 + w_2\chi_s \tag{19}$$
$$\gamma = (1 - w_1)\,{}^D\psi_D + w_2\beta_0 \tag{20}$$
$$\psi_d = (1 - w_3)\alpha + w_3\gamma \tag{21}$$

where ρ is the distance between the DS and the AUV, e_x and e_y are the x and y component of the distance between the DS and the AUV, χ_s is the angle between the DS axis and the AUV, ρ_0, $e_{x,0}$, $e_{y,0}$, $\chi_{s,0}$ are the values of the variables explained before when the AUV starts the final approaching maneuver. This method allows controlling both the AUV docking heading and forward velocity.

G. SLIDING PATH CONTROLLER

Sans-Muntadas *et al.* [34] presents a strategy that, like Park *et al.* [33], requires planning a path. This method follows the following steps (see Fig. 12):

- The AUV approaches the DS following a path parallel to its center-line with an offset of a few meters on the side of the current (approaching path, see Fig. 12a). An Integral Line-Of-Sight (ILOS) controller is used to regulate the cross-track error to zero.
- When a certain distance from the DS is reached, the AUV switches to the sliding path. In the sliding path a control rule is set to compensate the offset of the approaching path in the y component of the $\{D\}$ frame, using the ocean currents, and aligning the AUV with the DS.

The ILOS controller [42] uses an integral function that controls the heading of the AUV, in order to minimize the cross-track error (see Fig. 12c), which is represented by the following rules:

$$\psi_d = \text{atan2}\,(y\ + \sigma\ y_{int},\ \Delta)\,, \tag{22}$$
$$\dot{y}_{int} = \frac{\Delta\ y}{(y + \sigma\ y_{int})^2 + \Delta^2}. \tag{23}$$

where y is the distance between ${}^D y_B$ and the approaching path, y_{int} is the integral factor, σ is a gain, and Δ is a look-ahead distance (as in the Pure pursuit controller). During this approaching phase the AUV has a crab angle to compensate for the ocean currents. The distance between the center-line and the approaching path (y_a) is computed as follows (see Fig. 12a):

$$\tan(\chi_s) = {}^D v_c / u_{dock}, \tag{24}$$
$$y_a = -l\,\sin(\chi_s). \tag{25}$$

where ${}^D v_c$ is the lateral current, and l is the distance to the DS where the sliding path controller will be enabled.

The sliding path is a straight line that leads to the entrance of the DS. Two control laws are used to follow this path. On one side the ILOS controller drives the cross-track error to zero but does not guarantee the AUV heading to be the same as ${}^D\psi_D$. On the other side, a Speed Regulated Guidance (SRG) controller forces the AUV to be parallel to the DS and tries to adjust the AUV position with respect to the sliding path, controlling only its surge velocity, see Fig. 12d:

$$\psi_d = 0, \tag{26}$$
$$S_c = u_m \frac{2}{\pi} \tan(k_u e), \tag{27}$$
$$u_{srg} = u_{dock} - {}^D u_c - S_c. \tag{28}$$

where u_m is the maximum velocity that the AUV could reach, k_u is a gain, and ${}^D u_c$ is the ocean current velocity on the x axis in the $\{D\}$ frame. A hybrid framework is used during the sliding path to decide which controller, the ILOS or the SRG, is active. Basically, when the AUV cross-track error is above a threshold, the ILOS controller is activated. When the cross-track error is below another threshold, the SRG is enabled instead. Some hysteresis is added when defining the thresholds to avoid chattering due to sensor noise (see Fig. 12b). In order to define the different zones, first, the line that represents the sliding path is calculated:

$$a = \frac{{}^D y_D - {}^D y_{cp2}}{{}^D x_D - {}^D x_{cp2}}, \tag{29}$$
$$b = {}^D y_D - a\,{}^D x_D, \tag{30}$$

where $\boldsymbol{{}^D P_{cp2}}$ is the position of the *change point 2* in the $\{D\}$ frame (see Fig. 11b). The SRG guidance zone is defined as:

$$a\,{}^D x_B + b - \phi_{dock}/2 \leq {}^D y_B \leq a\,{}^D x_B + b + \phi_{dock}/2, \tag{31}$$

where ϕ_{dock} is the external diameter of the funnel. The zone where the SRG control is changed to ILOS control is defined as the union of:

$${}^D y_B > a\,{}^D x_B + b + \phi_{dock}/2, \tag{32}$$

and

$${}^D y_B < a\,{}^D x_B + b - \phi_{dock}/2. \tag{33}$$

The ILOS guidance zone is defined as the union of:

$${}^D y_B \geq a\,{}^D x_B + b + \phi_{dock}/4, \tag{34}$$

and

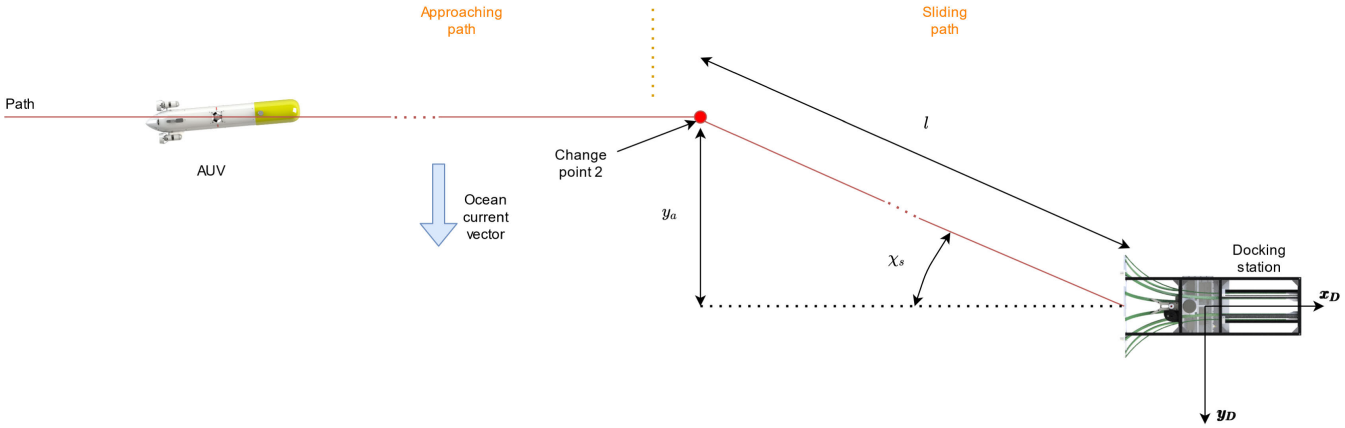
$${}^D y_B \leq a\,{}^D x_B + b - \phi_{dock}/4. \tag{35}$$

Finally, the zone where the ILOS control is changed to SRG control is represented as:

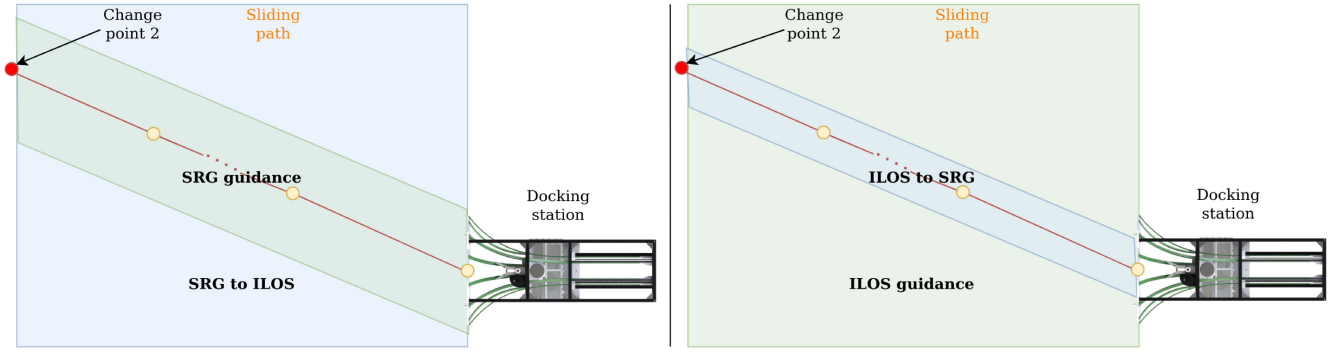
$$a\,{}^D x_B + b - \phi_{dock}/4 < {}^D y_B < a\,{}^D x_B + b + \phi_{dock}/4. \tag{36}$$

If the SRG control is applied, when the AUV reaches the DS, the rules of the control can be defined as:

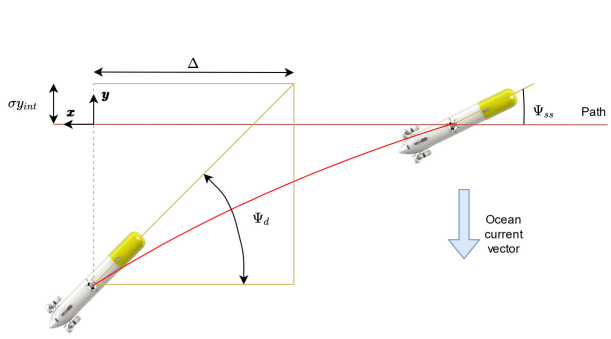
$$\psi_d = 0, \tag{37}$$
$$u_d = u_{srg}. \tag{38}$$



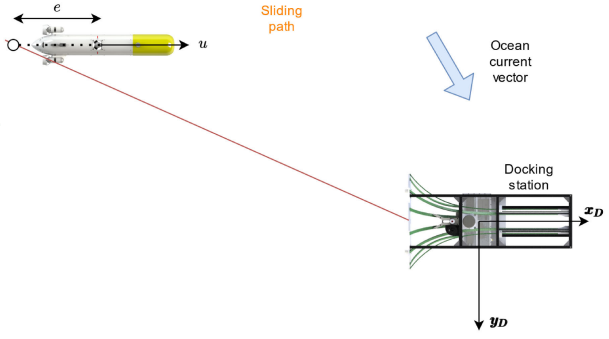
(a) Sliding path concept.



(b) Sliding path control zones.



(c) Sliding path ILOS concept.



(d) Sliding path SRG concept.

FIGURE 12. **Sliding path representation.**

Note that the u_{srg} defined in (28) is with respect to the water, while the one defined in (38) is with respect to the ground. This method allows for controlling the heading of the AUV but not the docking velocity.

IV. DESIGN TOOLS

This section presents the setup (i.e., AUV, DS, and localization system) used for the comparative analysis; as well as the Stonefish dynamic simulator used to carry out all the tests; and a new metric, proposed in this article, to evaluate the quality of a docking maneuver.

A. DOCKING SETUP

This section introduces the AUV, the DS, and the USBL system that have been simulated.

1) SPARUS II

The Sparus II is an AUV developed primarily for seabed surveys and offshore structure inspection by the University of Girona, and recently commercialized by Iqua Robotics SL. [43], [44]. It combines the classical concept of a torpedo-shaped vehicle with hovering capabilities (see Fig. 13). To allow for the integration of equipment, that depends on the

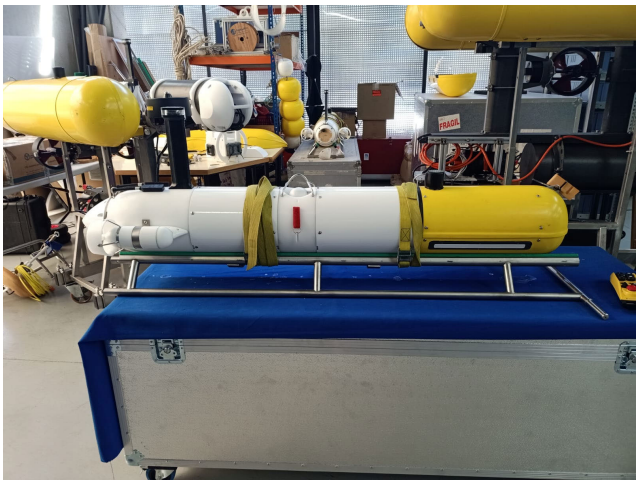


FIGURE 13. Photography of the sparus II.

application, the robot has a fully configurable payload area. Its software architecture is based on COLA2 [35] which is utilising the ROS open-source middleware [45]. The main specifications and features of the Sparus II can be found in Table 3. The shape of the AUV hull is optimised for navigation at medium/high velocities. The vehicle can be controlled in surge, heave, and yaw degrees of freedom independently by means of three thrusters (one vertical and two horizontal). It can reach a maximum velocity in surge of 3 kn. The vehicle is rated for up to 200 m depth. Its navigation suite includes an Inertial Measurement Unit (IMU), a Doppler Velocity Log (DVL), a GPS, and, optionally, a USBL. The latter can be used as a beacon to localize the AUV from the surface or, as inverted-USBL, to localize a target underwater, here the DS equipped with an acoustic beacon.

A Kalman filter estimates the position and velocity of the vehicle using the information from the sensors. The filter is first initialized at the surface, with the GPS, but it is also possible to apply position updates using the USBL when the vehicle is submerged. The control system is divided into two parts: a high-level controller and a low-level controller. In this article, a high-level controller for each of the docking methods, described in Section III, has been defined. The low-level controller, based on a cascade of PID and an open-loop thruster model, takes the desired velocities or heading set-points, defined by the high-level controller, and generates the thrusters' set-points.

The depth of the vehicle is controlled independently by the vertical thruster. This reason justifies that the study presented in this paper could have been simplified to the *xy* plane.

2) DOCKING STATION

The DS that has been simulated in all of the experiments was developed by the University of Girona [13], see Fig. 14. This DS was tailored to the Sparus II AUV and it has been designed to be as small and lightweight as possible, for easy deployment, recovery, and transportation. The structure

TABLE 3. Sparus II specifications.

Sparus II specifications	
Lenght	1.6 m
Hull diameter	230 mm
Max width	460 mm
Weight in air	52 kg
Maximum depth	200 m
Energy	1.4 kWh Li-Ion battery
Endurance	8 - 10 h
Max surge velocity	3 kn
Propulsion system	3 thrusters (magnetic coupling)
DoFs	Surge, heave, pitch, roll and yaw
Structure	Modular aluminium and acetal hull
Navigation	DVL, IMU, pressure sensor and GPS
Payload area	8 l or 7 kg in air
Payload interface	Ethernet, RS-232 / regulated 12 V and 24 V
Comunications	WiFi, XBee, GSM/3G, acoustic modem
Safety	Emergency primary battery, independent pinger tracking system, flasher light, USBL and acoustic modem

is divided into two parts: the base and the docking funnel (see Fig. 15). It can handle translation misalignment of up to 40 cm and heading misalignment of up to 30°. The rails are made of flexible POM (polyoxymethylene) to absorb collisions.

3) VEHICLE-DS LOCALIZATION

An inverted-USBL (i.e., a transceiver located on the AUV and a transponder on the DS) has been simulated to obtain the position of the DS from the AUV. To simplify the visualization of all of the experiments, the DS position has been defined at the origin of the $\{D\}$ frame.

B. STONEFISH SIMULATOR

The Stonefish simulator [14] is an advanced open-source simulation tool, designed for marine robotics. It can simulate

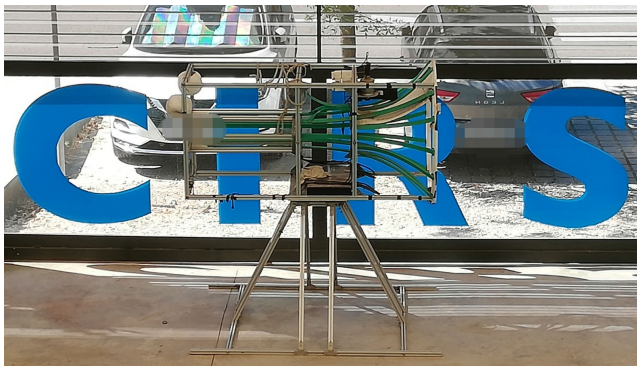


FIGURE 14. Photography of the docking station.

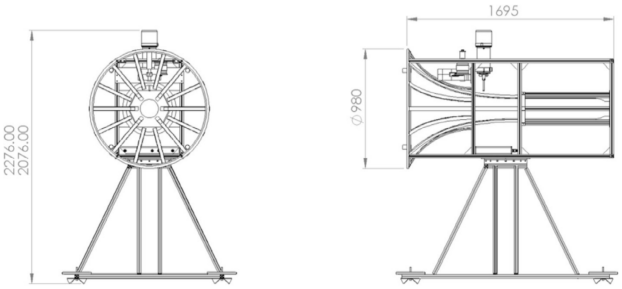


FIGURE 15. Main dimensions of the docking station (in mm).

multiple underwater and surface robots, working in a virtual ocean environment. The simulator is able to compute the full dynamics and hydrodynamics of underwater and surface robots, simulate the operation of thrusters and sensors, and to emulate the acoustic communication devices. The simulated sensors cover the full range of devices found in marine robots, e.g., DVL, pressure sensor, USBL, and all types of sonars. Moreover, it delivers the possibility to fully define an environment using geometric models or heightmap-based terrain. It can simulate ocean currents and Fast-Fourier transform (FFT) based waves. Finally, realistic rendering of the ocean surface and underwater environment, including light absorption and scattering, enables accurate simulation of underwater cameras.

The Stonefish simulator was used to recreate the docking scenario, including the complete models of the Sparus II AUV and the DS, situated at the bottom of a virtual ocean. In Fig. 16 the simulator window is presented, showing the visualisation of the docking scenario during one of the simulations.

The docking algorithms presented in Section III have been implemented as high-level controllers that run within the COLA2 robot architecture [35]. The simulator communicates with the COLA2 architecture, through a ROS-based interface, and it replaces the real vehicle in a hardware-in-the-loop (HIL) configuration. In this way all of the developments carried out using the simulator can be directly tested on the real vehicle. The parameters of all of the devices, including noise characteristics, were set up according to the manufacturers' specifications. Moreover, different types of underwater ocean

FIGURE 16. Stonefish simulation example.

current models can be defined in Stonefish. Here, a uniform current velocity field was used, with different current direction and velocity for different experiments. The simulator can also compute rigid collisions between the AUV and the DS, allowing estimation of the behaviour of the system during the entrance.

In order to realistically recreate the behavior of a physical USBL, the Stonefish uses equations reported in [46] and sets the different parameters using real data from the EvoLogics 18/34 USBL. This allows us to account for the equipment's measurement errors and noise characteristics.

The DVL characteristics match those of the Teledyne Marine WHN 600, which is the model used by the AUV Sparus II. The DVL readings are used by the AUV's navigation system to estimate its position and velocity, relative to the ground, as well as to estimate the direction and magnitude of ocean currents.

C. ENTRANCE QUALITY ANALYSIS

To evaluate the docking methods, a new technique to measure the quality of the entrance to the DS is proposed here. A geometrical analysis to obtain a representative value of the quality of the entrance is presented, based on a collision analysis where the momentum lost in the entrance maneuver is estimated.

1) COLLISION ANALYSIS

The collision analysis estimates the momentum lost on the entrance due to the collisions and friction between the robot and the DS (m_l). To do that, it compares the momentum of the AUV crossing the first section of the DS (m_{s1}) with the momentum of the AUV crossing the second section (m_{s2}), see Fig. 17. These quantities can be obtained using the velocity of the AUV at the moment of crossing each of the sections $|\vec{v_1}|$, $|\vec{v_2}|$ and the mass of the vehicle M:

$$m_{s1} = M|\vec{v_1}|, \quad m_{s2} = M|\vec{v_2}|. \tag{39}$$

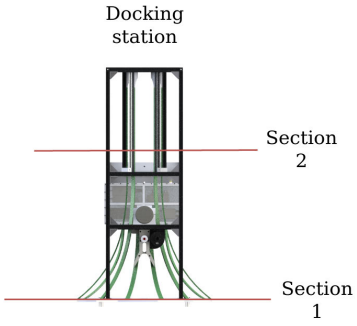


FIGURE 17. Collision analysis sections representation.

The change in momentum is mainly explained by the action of the thrusters (m_T) and the energy lost in the collisions and friction between both sections:

$$m_{s2} = m_{s1} + m_T - m_l. \tag{40}$$

The software architecture of the Sparus II allows us to record the force that the thrusters are generating at each moment in time (F_t). Integrating this force over the period between the sections, the change of momentum (impulse) caused by thrusters can be obtained:

$$m_T = \int_{t_1}^{t_2} F_t dt. \tag{41}$$

Then, the change of momentum related to the collisions and friction can be calculated as:

$$m_l = m_T + m_{s1} - m_{s2}. \tag{42}$$

This analysis has a few drawbacks: the maneuver needs to be performed before it is evaluated, the method cannot distinguish whether the AUV enters the DS or does not, and it does not provide a normalized value. For these reasons the geometrical analysis, presented in the next subsection, was developed.

2) GEOMETRICAL ANALYSIS

Let us simplify the system representing the DS with a triangle, where one side represents the DS entrance section and the opposite vertex the DS center (see Fig. 18). When the AUV reaches the *entrance section*, the position and the heading of the AUV $^{D}\boldsymbol{\eta_e}$ are registered. To simplify the nomenclature $^{D}\psi_{B,e}$ will be called α. With this diagram, an optimal entrance angle (α_o) for each entrance position can be approximated:

$$\alpha_o = \text{atan2}(^{D}y_{B,e}, {}^{D}x_{B,e}). \tag{43}$$

The heading of the AUV at the entrance and the optimal entrance angle can be compared as:

$$e_\alpha = \alpha - \alpha_o. \tag{44}$$

It is necessary to take into account on which side of the center-line of the DS the AUV is located, when colliding with

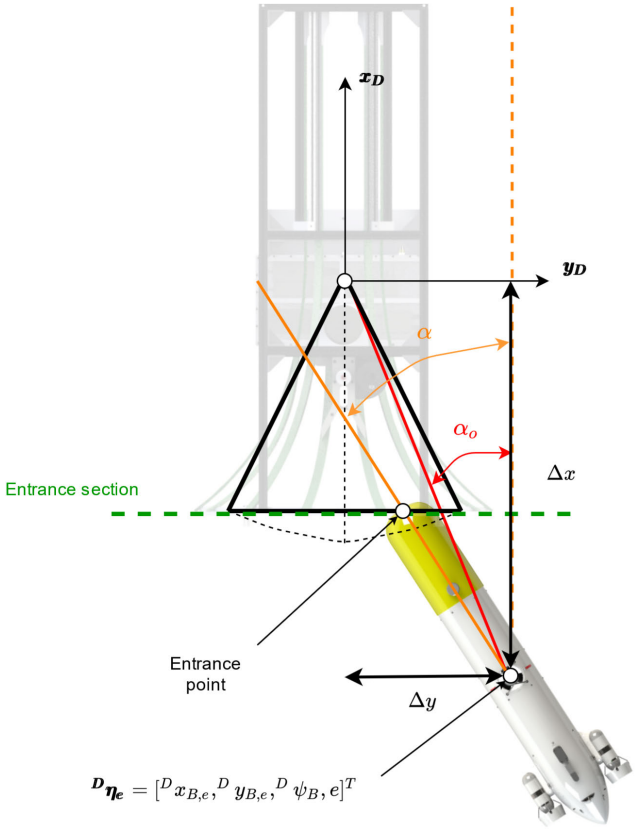


FIGURE 18. Geometrical analysis concept.

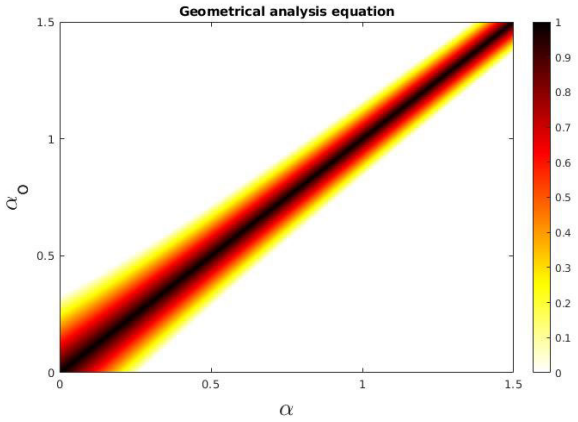
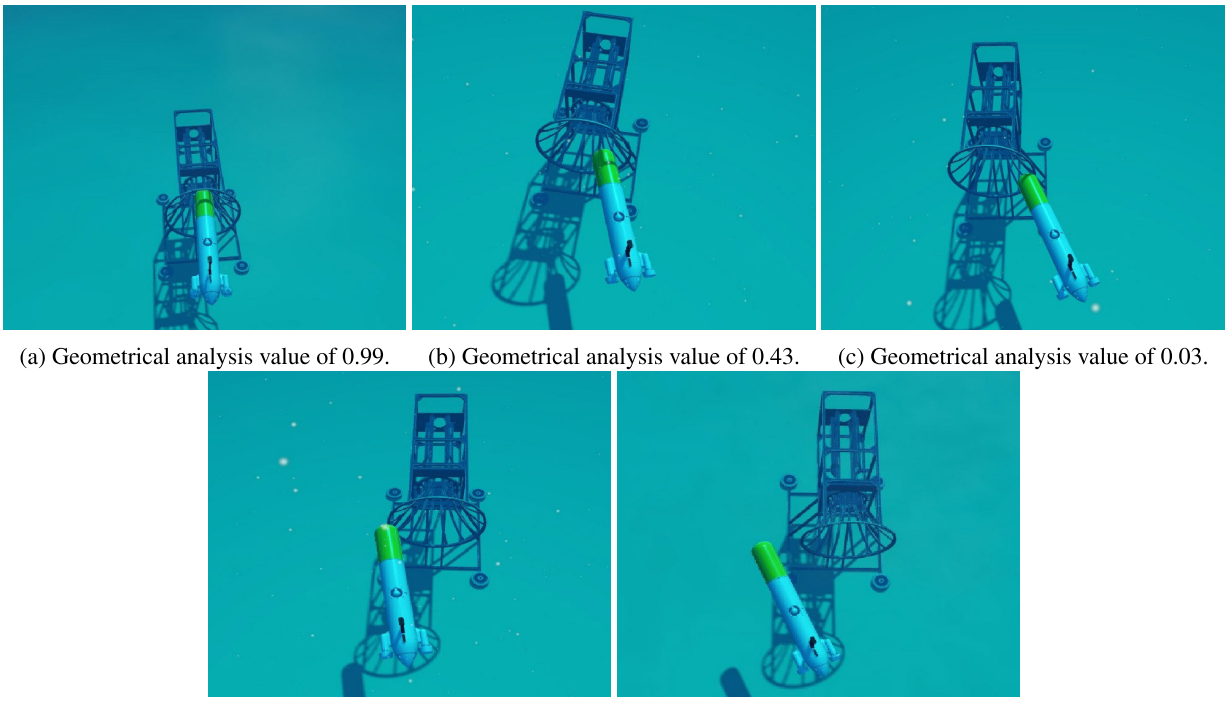


FIGURE 19. Graphical representation of the geometrical analysis equation. The value of g is represented by the 'hot' color map.

the funnel, because the same e_α value will result in different behaviours, e.g., when the vehicle approaches the funnel on its right side and $e_\alpha > 0$, the docking maneuver will be performed more smoothly and the lost momentum will be lower, than if it was approaching on the left side. To consider this fact, a weight w is assigned, following these criteria:

$$w = \begin{cases} w_1, & \text{sgn}(\Delta y) = \text{sgn}(e_\alpha) \\ w_2, & \text{otherwise.} \end{cases} \tag{45}$$



(a) Geometrical analysis value of 0.99. (b) Geometrical analysis value of 0.43. (c) Geometrical analysis value of 0.03.

(d) Geometrical analysis value of -0.01. (e) Geometrical analysis value of -1.88.

FIGURE 20. **Geometrical analysis examples.**

Regarding the presented concept, the geometrical analysis can be expressed with the formula:

$$g = 1 - w|e_\alpha|(1 + |\alpha_o|). \tag{46}$$

It should be noticed that this formulation consists of two terms. The first one evaluates the entrance angle against the optimal entrance angle. The second one considers the value of the optimal entrance angle itself to reward entrances in the center of the DS. These behaviours can be seen in Fig. 19. The combination of these two terms was used, in an attempt to reflect the results obtained in the collision analysis in the cases in which the AUV enters the DS. Moreover, the equation (46) was designed to have a value of $g = 1$ if the docking maneuver is ideal, a value of $0 < g \leq 1$ if the AUV enters the DS, and a value $g \leq 0$ if the AUV does not enter the DS, see Fig. 20. The weights in (45) were evaluated as $w_1 = 1$ and $w_2 = 3.05$, based on the collision analysis performed on simulated data, assuming the aforementioned properties of (46). To summarize, the value of the geometrical analysis g, allows evaluation of the quality of the docking maneuver and at the same time it retains the connection with the physical phenomena accompanying the entrance (collision and friction). It provides a normalized value that allows an intuitive understanding of the docking process. Moreover, it can be used to compute the set of preferred docking parameters (pairs of the offset from the center-line and the heading), i.e., the region of attraction of a particular docking algorithm.

V. COMPARISON BENCHMARKS

The docking algorithms presented in Section III were tested using the setup introduced in Section IV, at three levels of increasing complexity. In order to average the results, at each level, a set of fifty simulations have been made, for each specific ocean current velocity vector ($\boldsymbol{\nu_c}$), considered to lie on the xy plane of $\{D\}$ and sampled from a square set ranging from (-0.4 m/s, -0.4 m/s) to (0.4 m/s, 0.4 m/s), with a step of 0.1 m/s on each axis. This resulted in more than eighty five thousand simulations executed. In each simulation, the AUV starts 300 m in front of the DS, to ensure sufficient space to perform the docking maneuver (${}^{\boldsymbol{D}}\boldsymbol{\eta} = [-300, 0, 0]^T$).

A. LEVEL 1: PERFECT MEASUREMENTS, 1.0 m/S

At this level, the AUV knows exactly the position of the DS in the world frame as well as its own position and the velocity of the current ($\boldsymbol{\nu_c}$). The desired docking velocity (u_{dock}) is set to 1 m/s to guarantee that it is significantly higher than the ocean current velocity. Some of the reviewed papers assume the same conditions ([31], [33], [34]). The objective of this level is to determine the ocean current conditions which these methods can cope with, when perfect measurements about the environment are available. It is important to remember that Sparus II dynamics and hydrodynamics are fully simulated and the vehicle control is affected by them.

B. LEVEL 2: NOISY MEASUREMENTS, 1.0 m/S

At this level, the AUV detects the position of the DS using a USBL, modeled as explained in Section IV. Moreover, a DVL

is used to estimate the ocean current velocity. The desired docking velocity is set to 1 m/s. The objective of this level is to determine how inaccuracies in both the localization and ocean current estimation affect the methods reviewed.

C. LEVEL 3: NOISY MESUREMENTS, 0.3 m/S

Because colliding with the DS at 1 m/s is not recommendable for a vehicle like the Sparus II AUV, the authors wanted to evaluate how the studied docking algorithms behave when the docking velocity was much lower, here around 0.3 m/s. This condition means that it is possible to experiment with an ocean current faster than the docking velocity, here up to 0.55 m/s. The same localization and current estimation system as in Level 2 are used.

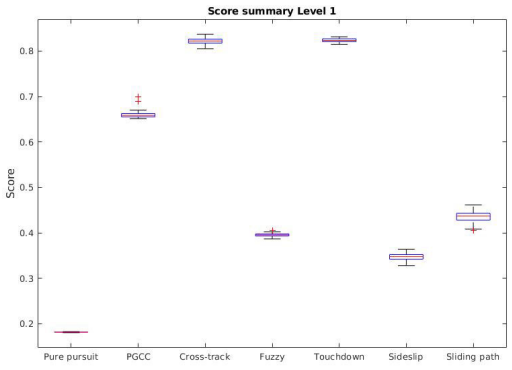
VI. RESULTS

This section presents the results of the performed simulations. An intuitive graphical representation of the behavior of each method was created, in a form of 3D plots depicting the mean geometrical value ($\bar{g}$) for each $\boldsymbol{\nu_c}$. Therefore, the 2D current velocity is represented on the xy plane, while the z axis represents $\bar{g}$, computed based on averaging fifty experiments. To further improve the readability, the *Hot* color map was used, where the black color denotes a perfect entrance while the white color a failed docking, and red and yellow being intermediate points. A single indicator, named 'score', can be obtained based on this representation, by integrating $\bar{g}$ over the set of $\boldsymbol{\nu_c}$ for which $\bar{g} \geq 0$. This volume is then normalized by comparing it with the maximum volume, given by a theoretical perfect score distribution where $\bar{g} = 1$ for all tested values of $\boldsymbol{\nu_c}$. The 2D plots presented in the Annex help to understand the behaviour of the methods in detail.

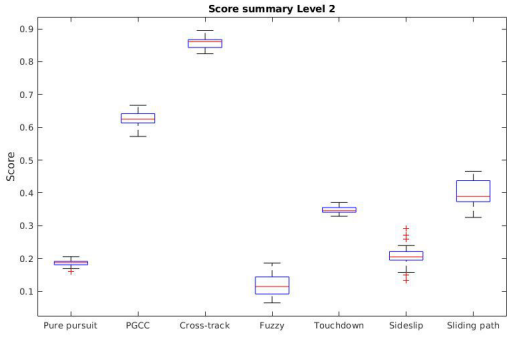
Figures 22 and 23 show the results obtained at Level 1 and Table 4 shows the average value and the standard deviation for the score obtained for each method.

The results of Level 2 can be analysed following Fig. 24, which shows the results in 3D, for more detail, Fig. 25 shows the results in the plane with the color map and Table 5 shows the geometrical volume mean and standard deviation of the different simulations.

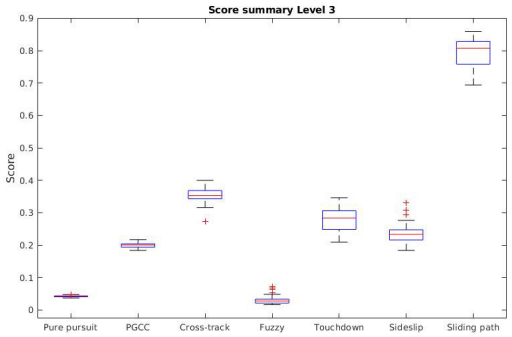
Finally, related to Level 3, in order to have a first picture of the results, see Fig. 26; for more details see Fig. 27; and for the quality number associated with each method see Table 6. As a summary, a comparison of all of the methods for each level is shown in Fig. 21.



(a) Score summary for the Level 1.



(b) Score summary for the Level 2.



(c) Score summary for the Level 3.

FIGURE 21. Score summary for each level.

VII. DISCUSSION

In this section, the results are discussed method by method.

A. PURE PURSUIT CONTROLLER

The pure pursuit controller is not designed to deal with ocean currents. It is presented here as a baseline to compare with the other methods. As already seen in [37], this controller cannot deal with lateral currents. It is not able to ensure successful docking at any level, if the lateral component of the current velocity is higher than 0.1 m/s.

B. PGCC CONTROLLER

The Pure Guidance with Current Compensation controller generally delivers good results. For the first two levels, it can deal with practically all the ocean current conditions, reaching a score of 0.660 at Level 1, and a score of 0.624 at Level 2. A little decline in performance is noticed moving from Level 1 to Level 2, due to the sensors' measurement noise. The Level 3 is significantly affected by the reduction of the docking velocity, resulting in a low score of 0.148, due to low values of $\bar{g}$ when $u_{dock} \leq {}^{D}u_c$. For example, in the case of $u_{dock} = {}^{D}u_c$ the AUV velocity with respect to the water has to be zero. Therefore, the only way to reduce the cross-track error is to set the desired heading $\psi_d = 90°$. However, this

TABLE 4. Score results for the perfect measurements simulations.

Method	Score	
	mean	std
Pure pursuit controller	0.181	0.001
PGCC controller	0.660	0.005
Cross-track controller	0.822	0.007
Fuzzy controller	0.395	0.005
Touchdown alignment controller	0.824	0.004
Sideslip controller	0.347	0.009
Sliding path controller	0.436	0.013

TABLE 5. Score results for the noisy measurements simulations with a docking velocity of 1 m/s.

Method	Score	
	mean	std
Pure pursuit controller	0.186	0.009
PGCC controller	0.624	0.026
Cross-track controller	0.858	0.015
Fuzzy controller	0.119	0.033
Touchdown alignment controller	0.348	0.010
Sideslip controller	0.207	0.033
Sliding path controller	0.402	0.039

does not allow the vehicle to enter the DS. As a result, it is not possible to simultaneously eliminate the cross-track error and complete the docking.

C. CROSS-TRACK CONTROLLER

The cross-track controller generally performs well, with the best score of 0.858 at Level 2 and also a very good result at Level 1 (0.822). It exhibits the same problems as the PGCC

TABLE 6. Score results for the noisy measurements simulations with a docking velocity of 0.3 m/s.

Method	Score	
	mean	std
Pure pursuit controller	0.148	0.002
PGCC controller	0.224	0.008
Cross-track controller	0.403	0.019
Fuzzy controller	0.128	0.014
Touchdown alignment controller	0.345	0.022
Sideslip controller	0.311	0.021
Sliding path controller	0.790	0.025

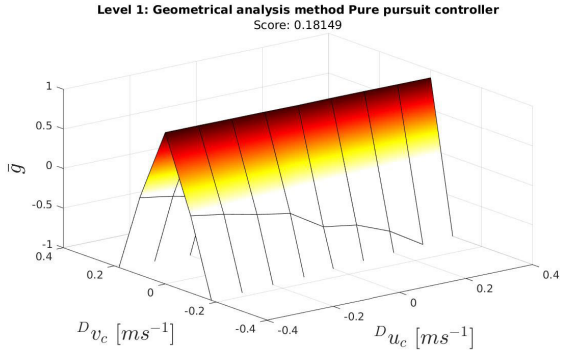
controller, when $u_{dock} \leq {}^{D}u_c$. However, it still scores an acceptable value of 0.403 at Level 3.

The improvement observed between Level 1 and Level 2, occurring despite the addition of noise in the sensors, can be explained by the way the geometrical analysis works. The geometrical analysis tries to evaluate not only if the AUV enters the DS but also how it enters. When the cross-track controller, or the PGCC controller, is working perfectly, the AUV enters the DS right through its center with the necessary crab angle to counteract the lateral component of the ocean current velocity. This crab angle is penalized by the geometrical analysis if the vehicle is perfectly in the center, but it is less penalized if there is some cross-track error, as shown in Fig. 18. When noisy measurements are used, the controller is unable to reduce the cross-track error to zero, and therefore a slight improvement in the score appears.

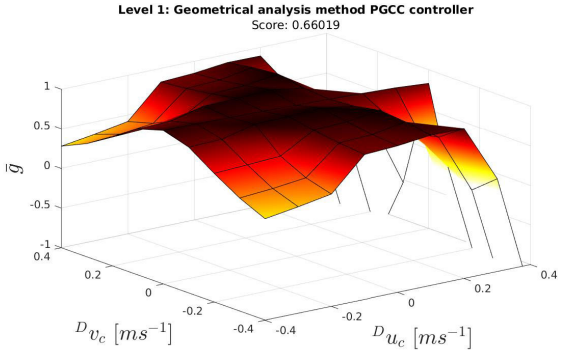
D. FUZZY CONTROLLER

The fuzzy controller achieves acceptable results at Level 1 (0.395), however, its performance at Level 2 and Level 3 is very low (0.119 and 0.128 respectively).

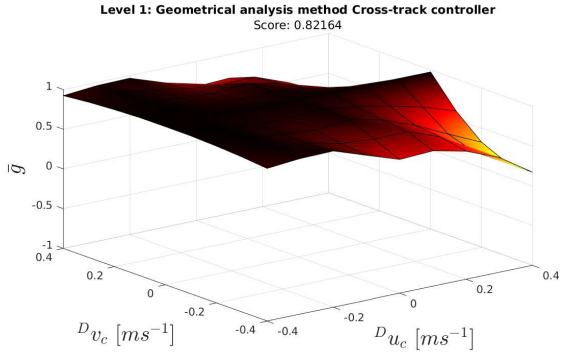
This controller has different particularities. A parameter-varying proportional controller corrects the heading, in order to reduce the cross-track error (CR, following Fig. 9). The proportional gain varies according to the fuzzy rules, which take into account the cross-track error (gain-scheduling). Due to the lack of an integral part in the control law, it cannot regulate the cross-track error to zero in the steady state. In order to deal with the ocean currents, it requires a non-zero cross-track error. Therefore, in the final phase of the approach, a large proportional gain is required to increase



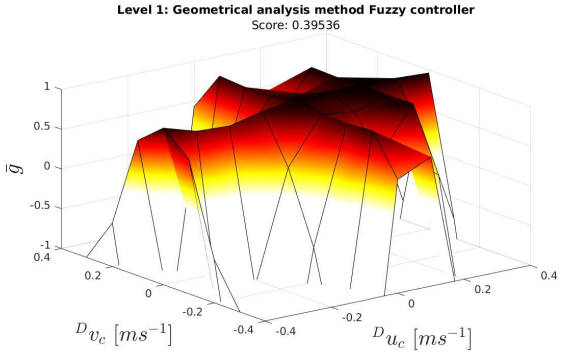
(a) 3D geometrical analysis, Level 1, Pure pursuit controller.



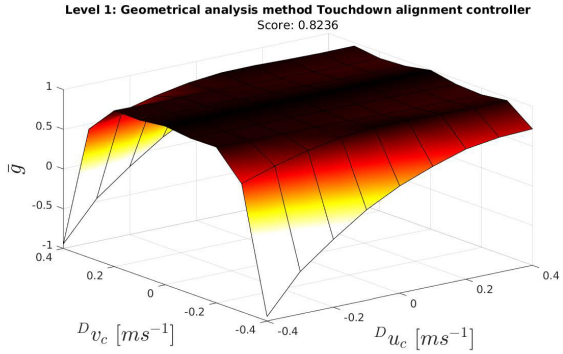
(b) 3D geometrical analysis, Level 1, PGCC controller.



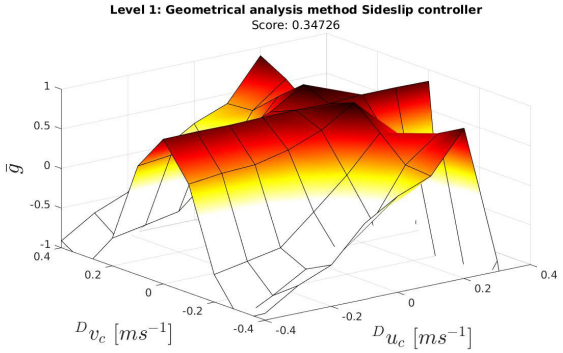
(c) 3D geometrical analysis, Level 1, Cross-track controller.



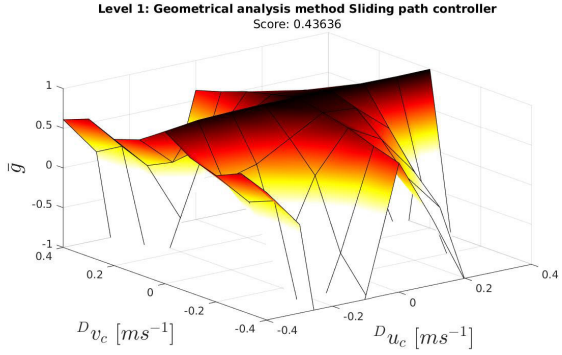
(d) 3D geometrical analysis, Level 1, Fuzzy controller.



(e) 3D geometrical analysis, Level 1, Touchdown alignment controller.

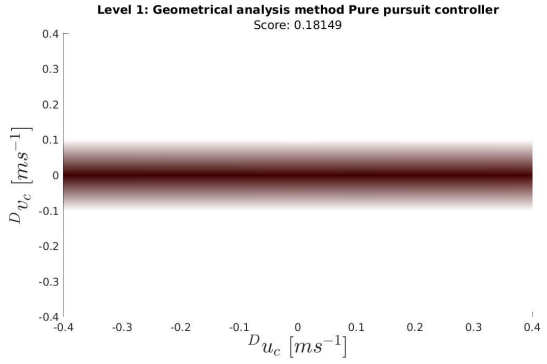


(f) 3D geometrical analysis, Level 1, Sideslip controller.

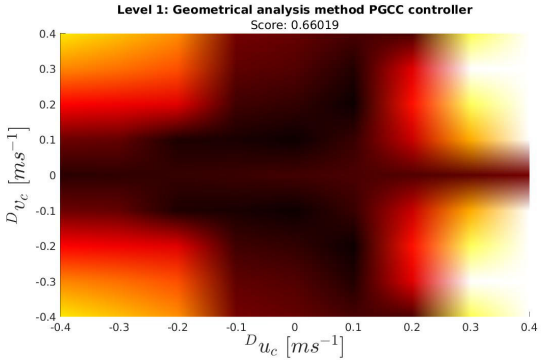


(g) 3D geometrical analysis, Level 1, Sliding path controller.

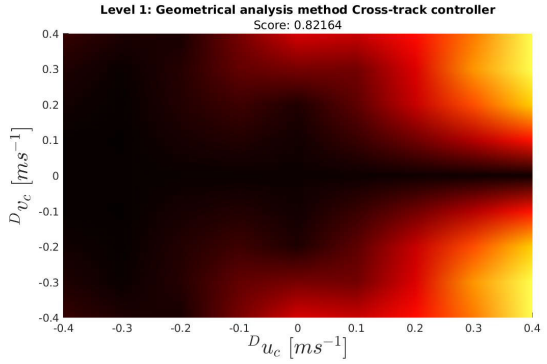
FIGURE 22. 3D representation of the geometrical analysis for the perfect measurements simulations.



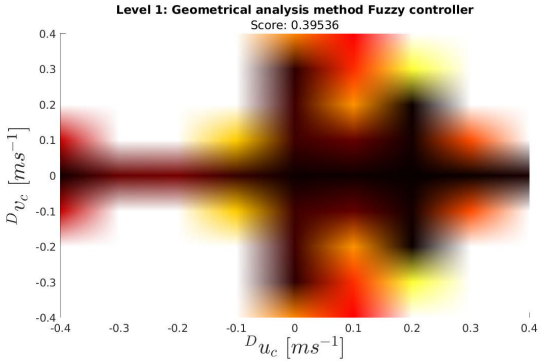
(a) 2D geometrical analysis, Level 1, Pure pursuit controller.



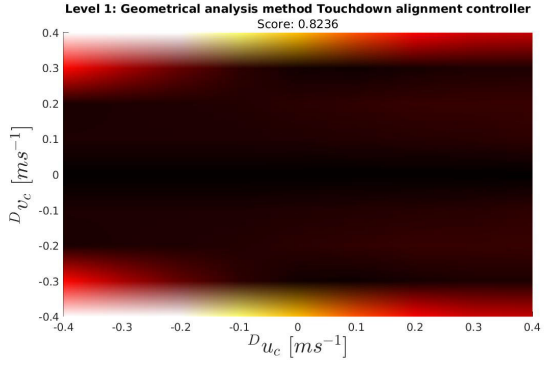
(b) 2D geometrical analysis, Level 1, PGCC controller.



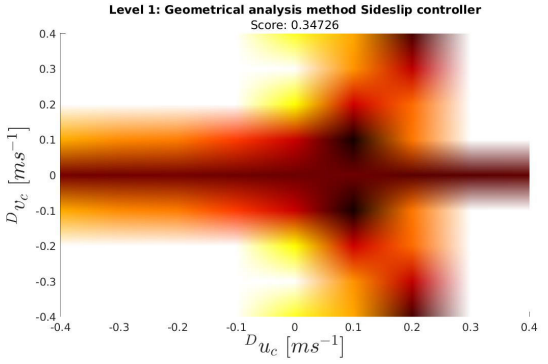
(c) 2D geometrical analysis, Level 1, Cross-track controller.



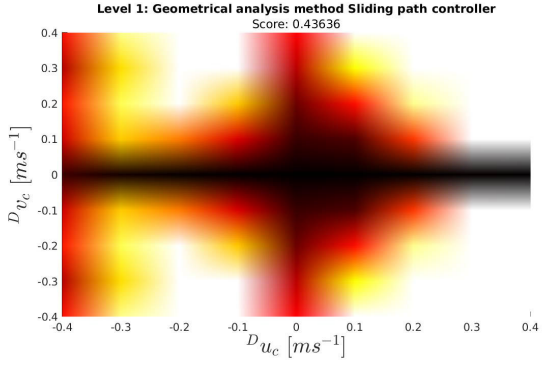
(d) 2D geometrical analysis, Level 1, Fuzzy controller.



(e) 2D geometrical analysis, Level 1, Touchdown alignment controller.

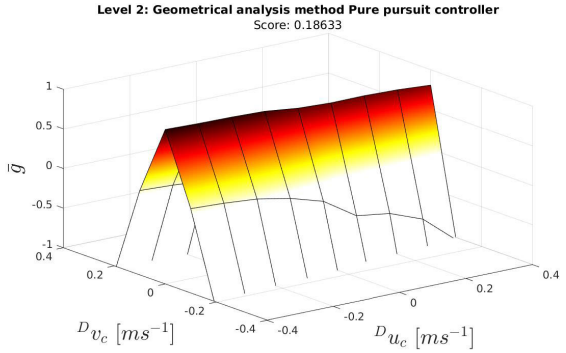


(f) 2D geometrical analysis, Level 1, Sideslip controller.

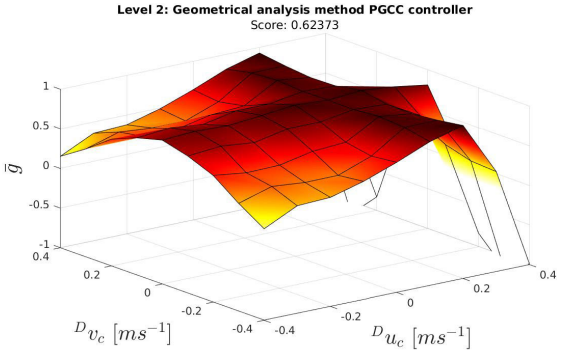


(g) 2D geometrical analysis, Level 1, Sliding path controller.

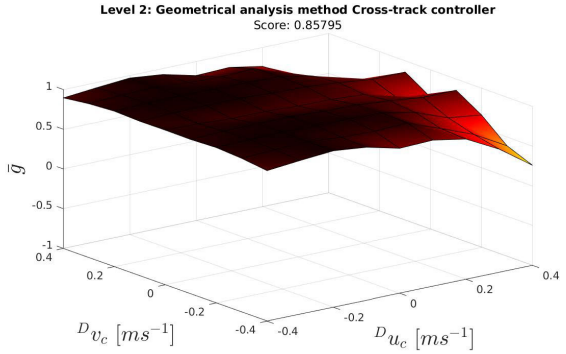
FIGURE 23. **2D representation of the geometrical analysis for the perfect measurements simulations. The value of $\bar{g}$ is represented by the 'hot' color map, according to Fig. 19.**



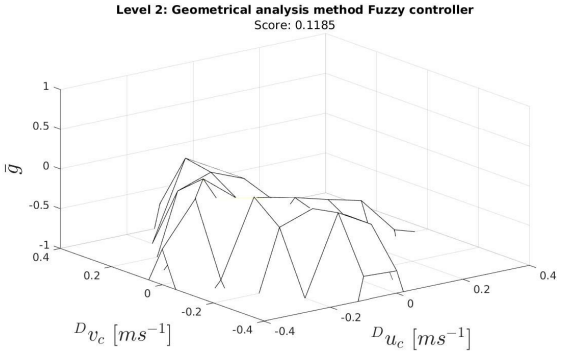
(a) 3D geometrical analysis, Level 2, Pure pursuit controller.



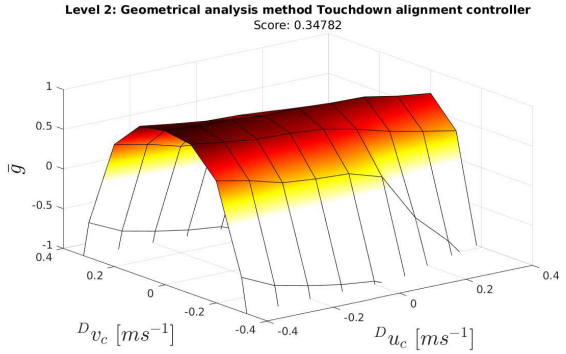
(b) 3D geometrical analysis, Level 2, PGCC controller.



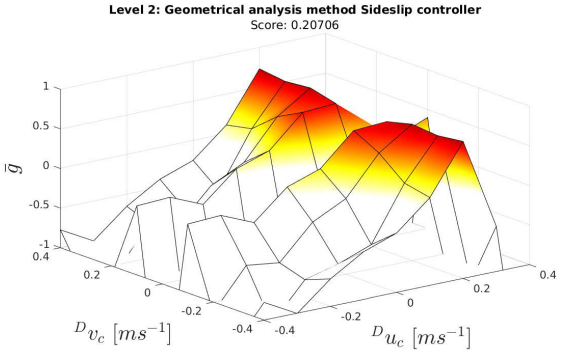
(c) 3D geometrical analysis, Level 2, Cross-track controller.



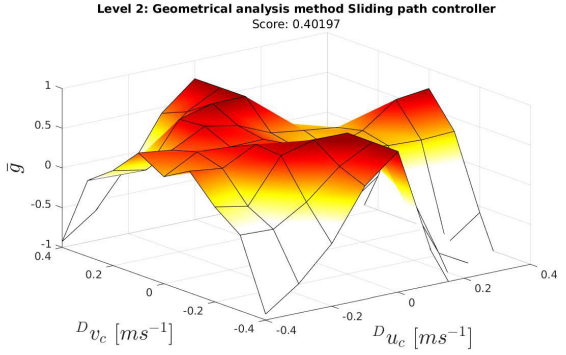
(d) 3D geometrical analysis, Level 2, Fuzzy controller.



(e) 3D geometrical analysis, Level 2, Touchdown alignment controller.

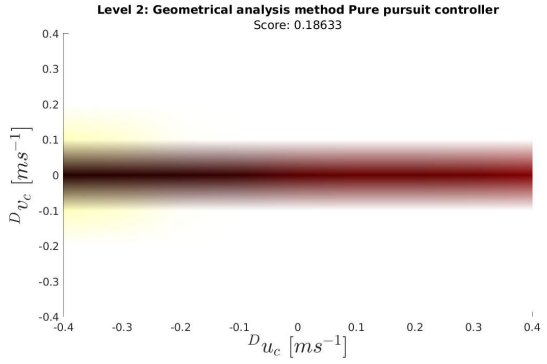


(f) 3D geometrical analysis, Level 2, Sideslip controller.

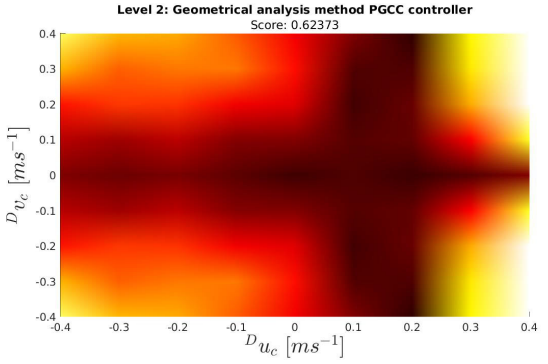


(g) 3D geometrical analysis, Level 2, Sliding path controller.

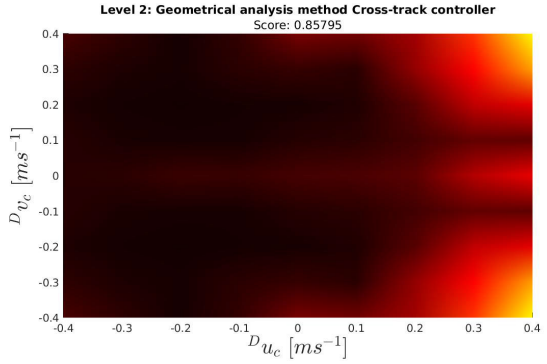
FIGURE 24. 3D representation of the geometrical analysis for the noisy measurements simulations with a docking velocity of 1 m/s.



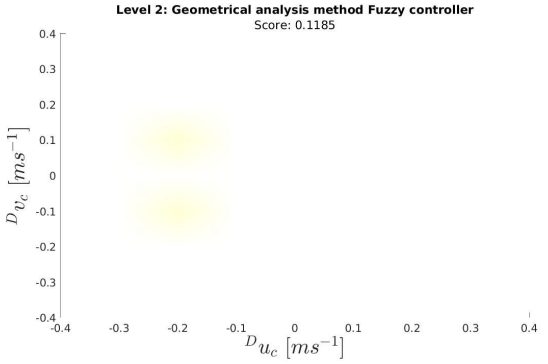
(a) 2D geometrical analysis, Level 2, Pure pursuit controller.



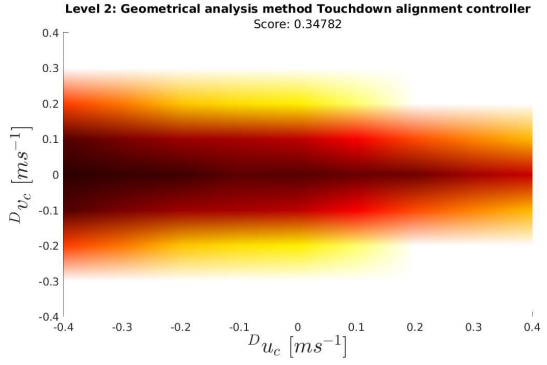
(b) 2D geometrical analysis, Level 2, PGCC controller.



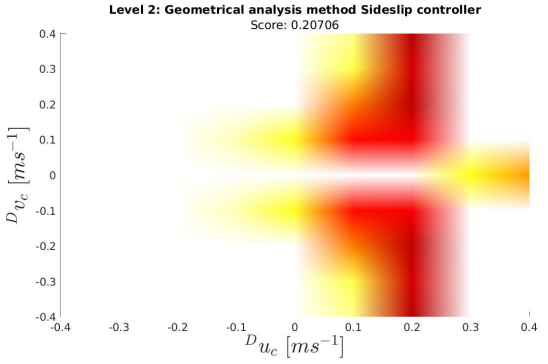
(c) 2D geometrical analysis, Level 2, Cross-track controller.



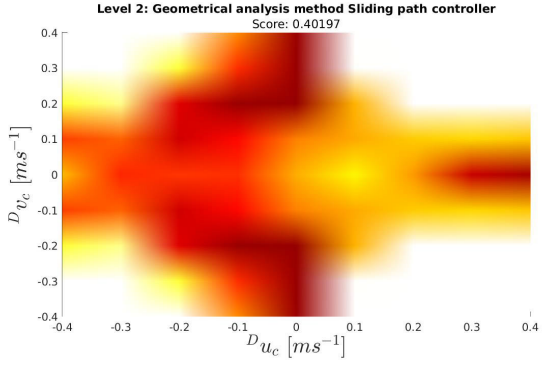
(d) 2D geometrical analysis, Level 2, Fuzzy controller.



(e) 2D geometrical analysis, Level 2, Touchdown alignment controller.

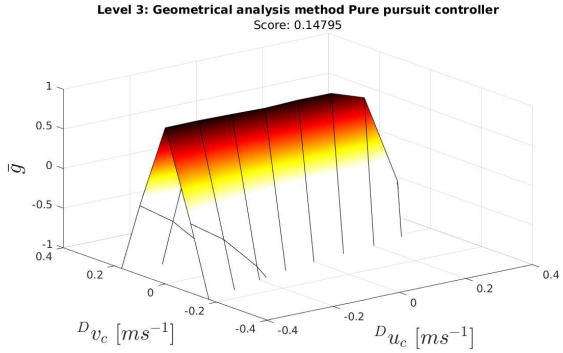


(f) 2D geometrical analysis, Level 2, Sideslip controller.

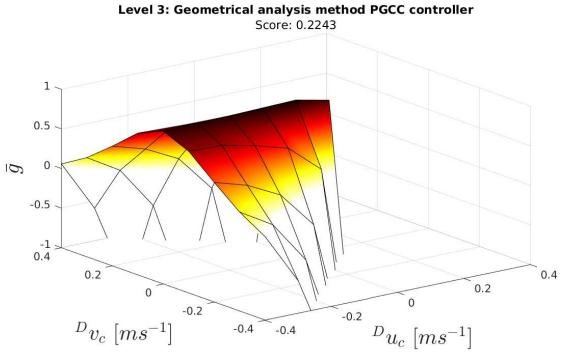


(g) 2D geometrical analysis, Level 2, Sliding path controller.

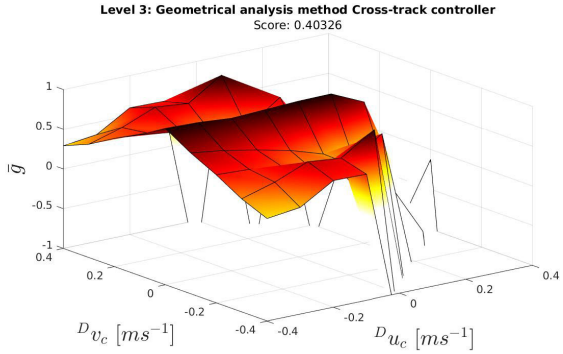
FIGURE 25. 2D representation of the geometrical analysis for the noisy measurements simulations with a docking velocity of 1 m/s. The value of $\bar{g}$ is represented by the 'hot' color map, according to Fig. 19.



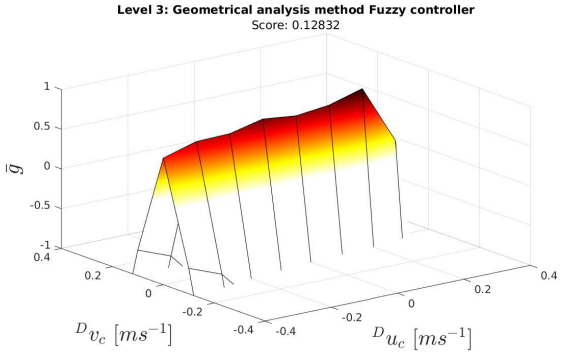
(a) 3D geometrical analysis, Level 3, Pure pursuit controller.



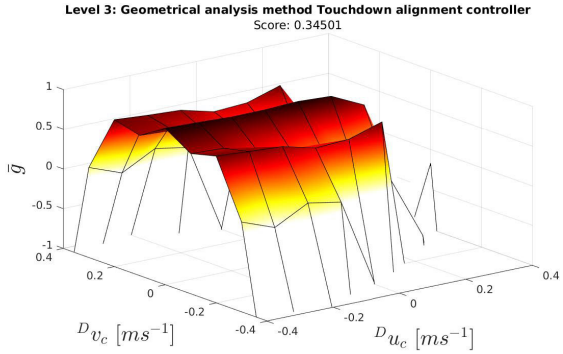
(b) 3D geometrical analysis, Level 3, PGCC controller.



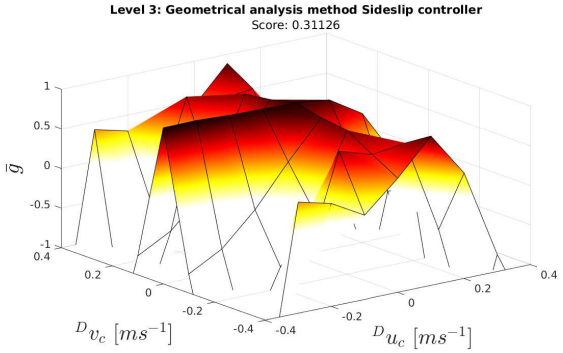
(c) 3D geometrical analysis, Level 3, Cross-track controller.



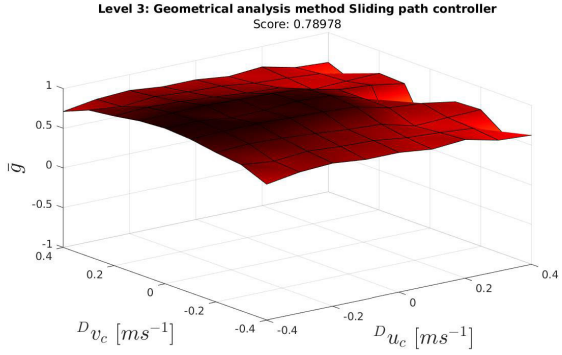
(d) 3D geometrical analysis, Level 3, Fuzzy controller.



(e) 3D geometrical analysis, Level 3, Touchdown alignment controller.

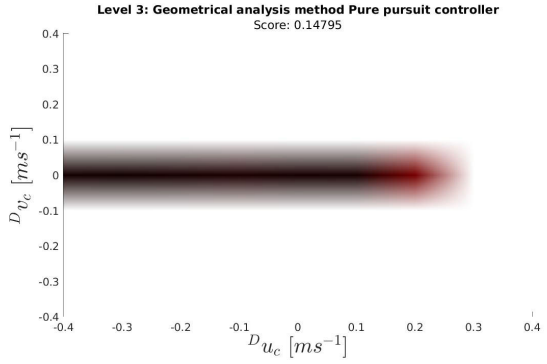


(f) 3D geometrical analysis, Level 3, Sideslip controller.

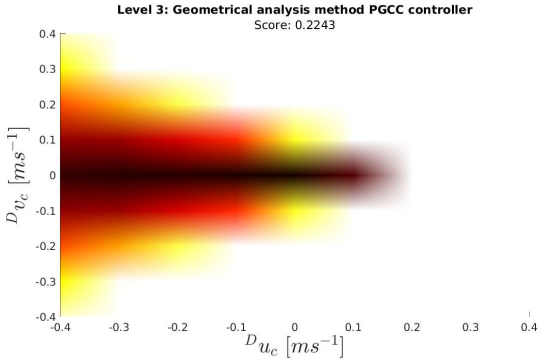


(g) 3D geometrical analysis, Level 3, Sliding path controller.

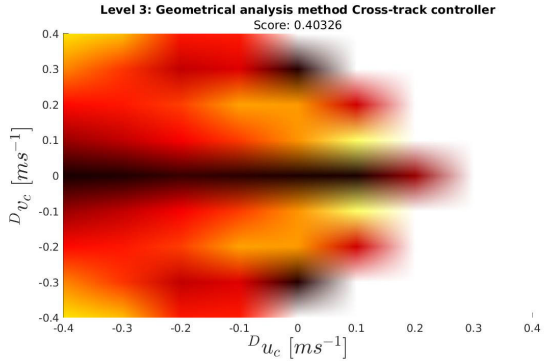
FIGURE 26. 3D representation of the geometrical analysis for the noisy measurements simulations with a docking velocity of 0.3 m/s.



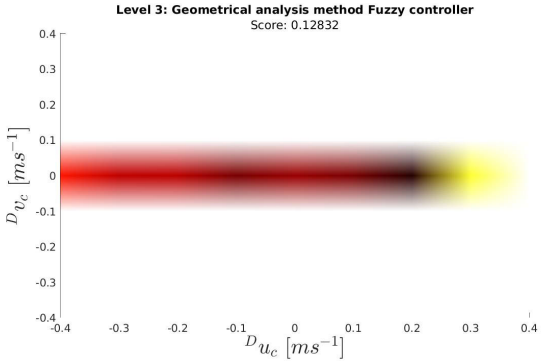
(a) 2D geometrical analysis, Level 3, Pure pursuit controller.



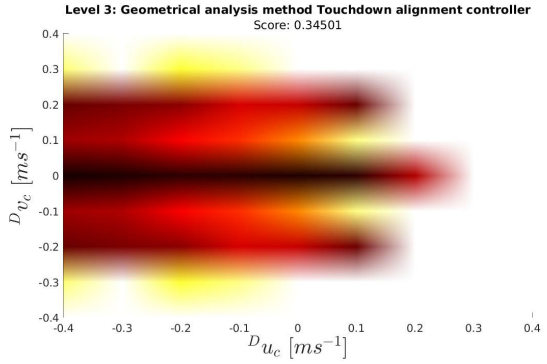
(b) 2D geometrical analysis, Level 3, PGCC controller.



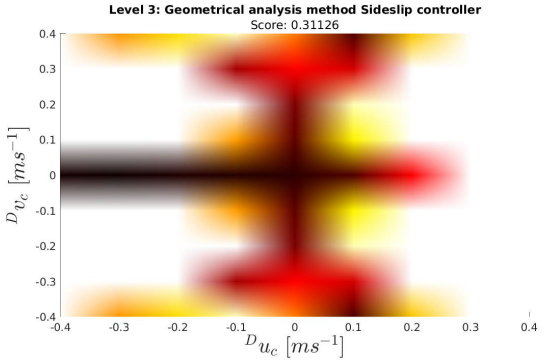
(c) 2D geometrical analysis, Level 3, Cross-track controller.



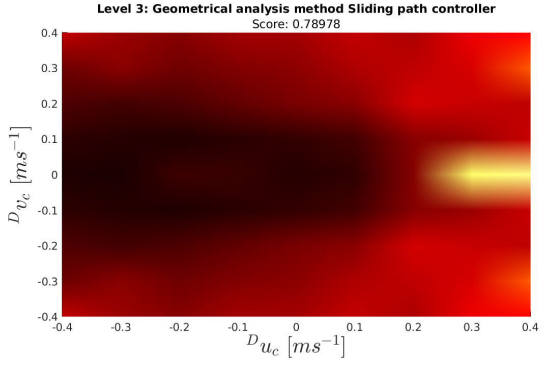
(d) 2D geometrical analysis, Level 3, Fuzzy controller.



(e) 2D geometrical analysis, Level 3, Touchdown alignment controller.



(f) 2D geometrical analysis, Level 3, Sideslip controller.



(g) 2D geometrical analysis, Level 3, Sliding path controller.

FIGURE 27. 2D representation of the geometrical analysis for the noisy measurements simulations with a docking velocity of 0.3 m/s. The value of $\bar{g}$ is represented by the 'hot' color map, according to Fig. 19.

the sensitivity of the controller. For this reason, when the noise in the sensors' measurements is taken into account, the system is noticeably affected, failing to enter the DS in the majority of the cases at Level 2. If the parameters recommended by the original authors - gains according to Table 1 and membership functions according to Fig. 9 - are used, the score at Level 1 improves from 0.395 to 0.599, and the vehicle enters in practically all of the scenarios. However, when the noise in the DS localization is taken into account, the results do not show relevant improvements.

E. TOUCHDOWN ALIGNMENT CONTROLLER

The Touchdown alignment controller achieves excellent results at Level 1 (0.824) but only acceptable results at Levels 2 and 3 (0.348 and 0.345 respectively).

The Touchdown alignment controller is a cross-track controller with a final heading correction. In Level 1, where no noise is present in the localization of the DS, it achieves the best results (by adapting the heading), to enter in a smooth way. However, when the DS location is uncertain, it shows worse scores than the Cross-track controller. This results from the fact that the heading correction depends directly on the relative position between the AUV and the DS.

F. SIDESLIP CONTROLLER

The Sideslip controller achieves average results. At Level 1 it reaches a score of 0.347 (with especially good results in case of longitudinal ocean currents), at Level 2 its score is 0.207, while at Level 3 the algorithm scores 0.311.

This method tries to correct the heading of the AUV in a smooth way, during the final part of the path. The results show that it is significantly affected by the low level controllers of the Sparus II, not being able to accomplish the maneuver in several cases, and showing better results when the docking velocity is low. The control algorithm is also notably affected by the noisy localization of the DS, since the heading depends on it directly. This fact can be appreciated by comparing the scores between Level 1 and Level 2.

G. SLIDING PATH CONTROLLER

Using the parameters proposed in the original paper, the performance obtained with this method is acceptable at Level 1 and 2 (0.436 and 0.402 respectively), but surprisingly, it is excellent at Level 3 (0.790), where it performs best.

At high docking velocities (approximately 1 m/s) the sliding path controller exhibits good results for medium and low ocean currents, but not for high currents. Overall, the system appears robust to noisy sensor measurements. The observed errors come from the ILOS controller, which does not seem to have enough space to stabilize the vehicle within the sliding path section, in cases of fast ocean currents.

At Level 3, this method has the same problem as the PGCC controller and the Cross-track controllers, when the longitudinal ocean current velocity component is close to, or greater than, the docking velocity. However, in this method the AUV follows a trajectory parallel to the center-line of the DS, rather than one coincident with it, and when the AUV reaches the sliding path section, the previous cross-track error is automatically eliminated. This happens due to the shape of the sliding path trajectory itself. This feature, coupled with the fact that at Level 3 the docking velocity is equal to 0.3 m/s, gives the vehicle more time to correct the final sliding path using the ILOS controller, making it work with all ocean current configurations at this level.

If the longitude of the sliding path is increased from 20 m (the distance used in the original paper) to 50 m, the score obtained at Level 1 increases up to 0.523, resulting in the AUV docking in practically all ocean current conditions. This improvement can be understood because the ILOS controller has more time to correct the position of the AUV, in the extreme cases.

VIII. CONCLUSION

This paper has compared seven docking methods in the presence of ocean currents, using the same setup: a funnel-shaped DS and a Sparus II AUV. These methods were selected after an exhaustive literature survey. They are: the Pure pursuit controller, the Pure Guidance with Current Compensation controller, the Cross-track controller, the Fuzzy controller, the Touchdown alignment controller, the Sideslip controller, and the Sliding path controller. Three scenarios have been simulated. In the first one (called Level 1), the controller knew the exact position of the DS and the ocean current velocity vector; also, the docking velocity was set to 1 m/s. In the second scenario (called Level 2), the controller used a realistic USBL model to estimate the position of the DS and a DVL model to estimate the ocean current velocity vector; the docking velocity was also set to 1 m/s. Finally, Level 3 was performed following the Level 2 conditions, but with a docking velocity of 0.3 m/s. Moreover, a new metric has been developed to be able to quantify the quality of the entrance of the AUV into the DS.

In the analysis of Level 1, it has been determined that the method that has the best performance is the Touchdown controller. This controller utilizes a Cross-track controller to minimize the cross-track error, correcting the effects of the ocean currents by applying a crab angle, and in the final moment it corrects the crab angle to dock in alignment with the DS.

At Level 2 the method that shows the best performance is the Cross-track controller.

At Level 3, the best performing method was the Sliding path controller. In this scenario, when the ocean current component parallel to the DS is directed towards it, and the current velocity is similar to the docking velocity, the AUV does not have the capacity to maintain the docking velocity, while effectively minimizing the cross-track error. The reason why the Sliding path controller outperforms the other methods is due to its final maneuver, where the AUV is able

to compensate for the cross-track error that occurred in the approaching path. This level reveals an interesting problem to study, that is, how to minimize the cross-track error when the ocean current is favourable and close to or larger than the docking velocity. In future work, a new algorithm will be developed and tested, to take into account the ocean currents that push a non-holonomic AUV towards the DS.

ABBREVIATIONS

AUV	Autonomous underwater vehicle
ROV	Remotely operated vehicle
DS	Docking station
USBL	Ultra short baseline
PGCC	Pursuit guidance with current compensation
PID	Proportional-integral-derivative controller
ILOS	Integral line of sight
SRG	Speed regulated guidance
COLA2	Component orientated layer-based architecture for autonomy
ROS	Robot operating system
IMU	Inertial measurement unit
DVL	Doppler velocity log
GPS	Global positioning system
POM	Polyoxymethylene
FFT	Fast Fourier transform
HIL	Hardware-in-the-loop
3D	Three dimensions
2D	Two dimensions

NOTATION

$^{D}\boldsymbol{P_D} = [^{D}x_D\ ^{D}y_D\ ^{D}\psi_D]^T$:	Docking station position in the $\{D\}$ frame (see Fig. 5)
$^{D}\boldsymbol{\eta} = [^{D}x_B\ ^{D}y_B\ ^{D}\psi_B]^T$:	AUV position in the $\{D\}$ frame (see Fig. 5)
$^{D}\boldsymbol{P_p} = [^{D}x_p\ ^{D}y_p]^T$:	Point position in the $\{D\}$ frame
$^{D}\boldsymbol{P_{cp2}} = [^{D}x_{cp2}\ ^{D}y_{cp2}]^T$:	Change point 2 position in the $\{D\}$ frame
Δx:	Distance in the x axis of the $\{D\}$ frame
Δy:	Distance in the y axis of the $\{D\}$ frame
u:	Surge velocity of the AUV
u_{dock}:	Surge velocity of the AUV set to dock
u_d:	Desired surge velocity of the AUV
$^{B}u_{B,w}$:	Surge velocity of the AUV with respect to the water
$\boldsymbol{\nu_c} \in \mathbb{R}^2$:	Ocean current velocity
$^{D}\boldsymbol{\nu_c} = [^{D}u_c\ ^{D}v_c]^T$:	Ocean current velocity in the $\{D\}$ frame
$^{B}\boldsymbol{\nu_c} = [^{B}u_c\ ^{B}v_c]^T$:	Ocean current velocity in the $\{B\}$ frame
ψ_d:	Desired yaw angle of the AUV
ψ_{crab}:	Crab angle of the AUV to compensate the ocean currents
ψ_{nc}:	Heading angle of the AUV without ocean currents
e:	Distance error in the Cross-track controller
β:	Heading of the AUV when it starts the final maneuver
r_{max}:	Maximum yaw rate of the AUV
d:	Distance between the AUV and the DS in the x axis of the $\{D\}$ frame used in the Touchdown alignment controller
d_f:	Distance in the x axis of the $\{D\}$ frame where the touchdown maneuver starts
$d_{f,min}$:	Minimum distance in the x axis of the $\{D\}$ frame where the touchdown maneuver starts
ρ:	Distance between the AUV and the DS in the Sideslip controller
e_x:	Distance on the x axis of the $\{D\}$ frame between the AUV and the DS in the Sideslip controller
e_y:	Distance on the y axis of the $\{D\}$ frame between the AUV and the DS in the Sideslip controller
χ_s:	Angle between the AUV and the DS, see Fig. 11c
ρ_0:	Distance between the AUV and the DS at the moment when the final maneuver starts in the Sideslip controller, see Fig. 11c
$e_{x,0}$:	Distance on the x axis of the $\{D\}$ frame between the AUV and the DS at the moment when the final maneuver starts in the Sideslip controller, see Fig. 11c
$e_{y,0}$:	Distance on the y axis of the $\{D\}$ frame between the AUV and the DS at the moment when the final maneuver starts in the Sideslip controller, see Fig. 11c
$\chi_{s,0}$:	Angle between the AUV and the DS in the moment when the final maneuver starts, see Fig. 11c
σ:	A gain
Δ:	Look-ahead distance
y_a:	Distance between the center-line and the approaching path in the Sliding path controller, see Fig. 12a

l: Length of the sliding path
k_u: A gain
ϕ_{dock}: External diameter of the funnel of the DS
m_l: Momentum lost on the entrance due to the collisions and friction between the AUV and the DS
m_T: Change of momentum generated by the thrusters between the first and the second section, see Fig. 17
m_{s1}: Momentum of the AUV when crossing the first section, see Fig. 17
m_{s2}: Momentum of the AUV when crossing the second section, see Fig. 17
$|\vec{v_1}|$: Velocity of the AUV when crossing the first section, see Fig. 17
$|\vec{v_2}|$: Velocity of the AUV when crossing the second section, see Fig. 17
M: Mass of the AUV
F_t: Force that the thrusters are producing in one instance
t_1: Time when the AUV crosses the first section, see Fig. 17
t_2: Time when the AUV crosses the second section, see Fig. 17
α: Heading of the AUV in the $\{D\}$ frame when it crosses the entrance section, see Fig. 18
α_o: Optimal heading of the AUV in the $\{D\}$ frame when it crosses the entrance section, see Fig. 18
e_α: Difference between α and α_o
w: A weight
g: Geometrical analysis value

REFERENCES

[1] L. L. Whitcomb, "Underwater robotics: Out of the research laboratory and into the field," in *Proc. Millennium Conf., IEEE Int. Conf. Robot. Automat. Symposia*, vol. 1, Apr. 2000, pp. 709–716.
[2] J. W. Nicholson and A. J. Healey, "The present state of autonomous underwater vehicle (AUV) applications and technologies," *Mar. Technol. Soc. J.*, vol. 47, no. 5, pp. 5–6, 2013.
[3] D. Ribas, N. Palomeras, P. Ridao, M. Carreras, and A. Mallios, "Girona 500 AUV: From survey to intervention," *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 1, pp. 46–53, Feb. 2012.
[4] R. S. McEwen, B. W. Hobson, L. McBride, and J. G. Bellingham, "Docking control system for a 54-cm-Diameter (21-in) AUV," *IEEE J. Ocean. Eng.*, vol. 33, no. 4, pp. 550–562, Oct. 2008.
[5] C. Yang, S. Peng, S. Fan, S. Zhang, P. Wang, and Y. Chen, "Study on docking guidance algorithm for hybrid underwater glider in currents," *Ocean Eng.*, vol. 125, pp. 170–181, Oct. 2016.
[6] T. Kawasaki, T. Fukasawa, T. Noguchi, and M. Baino, "Development of AUV 'Marine Bird' with underwater docking and recharging system," in *Proc. 3rd Int. Workshop Sci. Use Submarine Cables Related Technol. (SSC)*, 2003, pp. 166–170.
[7] H. Singh, J. G. Bellingham, F. Hover, S. Lemer, B. A. Moran, K. von der Heydt, and D. Yoerger, "Docking for an autonomous ocean sampling network," *IEEE J. Ocean. Eng.*, vol. 26, no. 4, pp. 498–514, Oct. 2001.
[8] R. Stokey, B. Allen, T. Austin, R. Goldsborough, N. Forrester, M. Purcell, and C. von Alt, "Enabling technologies for REMUS docking: An integral component of an autonomous ocean-sampling network," *IEEE J. Ocean. Eng.*, vol. 26, no. 4, pp. 487–497, Oct. 2001.
[9] M. D. Feezor, F. Y. Sorrell, P. R. Blankinship, and J. G. Bellingham, "Autonomous underwater vehicle homing/docking via electromagnetic guidance," *IEEE J. Ocean. Eng.*, vol. 26, no. 4, pp. 515–521, Oct. 2001.
[10] T. Podder, M. Sibenac, and J. Bellingham, "AUV docking system for sustainable science missions," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Apr. 2004, pp. 4478–4485.
[11] J. G. Bellingham, "Autonomous underwater vehicle docking," in *Springer Handbook of Ocean Engineering*. Cham, Switzerland: Springer, 2016, ch. 16, pp. 387–406.
[12] A. M. Yazdani, K. Sammut, O. Yakimenko, and A. Lammas, "A survey of underwater docking guidance systems," *Robot. Auto. Syst.*, vol. 124, Feb. 2020, Art. no. 103382.
[13] N. Palomeras, G. Vallicrosa, A. Mallios, J. Bosch, E. Vidal, N. Hurtos, M. Carreras, and P. Ridao, "AUV homing and docking for remote operations," *Ocean Eng.*, vol. 154, pp. 106–120, Apr. 2018.
[14] P. Cieslak, "Stonefish: An advanced open-source simulation tool designed for marine robotics, with a ROS interface," in *Proc. OCEANS Marseille*, 2019, pp. 1–6.
[15] J. C. Evans, K. M. Keller, J. S. Smith, P. Marty, and V. Rigaud, "Docking techniques and evaluation trials of the SWIMMER AUV: An autonomous deployment AUV for work-class ROVs," in *Proc. Oceans Conf. Rec. (IEEE)*, vol. 1, 2001, pp. 520–528.
[16] J. Evans, P. Redmond, C. Plakas, K. Hamilton, and D. M. Lane, "Autonomous docking for Intervention-AUVs using sonar and video-based real-time 3D pose estimation," in *Proc. Oceans, Celebrating Past. Teaming Toward Future*, 2003, pp. 2201–2210.
[17] T. Matsuda, T. Maki, K. Masuda, and T. Sakamaki, "Resident autonomous underwater vehicle: Underwater system for prolonged and continuous monitoring based at a seafloor station," *Robot. Auto. Syst.*, vol. 120, Oct. 2019, Art. no. 103231.
[18] Z. Yan, D. Xu, T. Chen, J. Zhou, S. Wei, and Y. Wang, "Modeling, strategy and control of UUV for autonomous underwater docking recovery to moving platform," in *Proc. 36th Chin. Control Conf. (CCC)*, Jul. 2017, pp. 4807–4812.
[19] G. D. Watt, A. R. Roy, J. Currie, C. B. Gillis, J. Giesbrecht, G. J. Heard, M. Birsan, M. L. Seto, J. A. Carretero, R. Dubay, and T. L. Jeans, "A concept for docking a UUV with a slowly moving submarine under waves," *IEEE J. Ocean. Eng.*, vol. 41, no. 2, pp. 471–498, Apr. 2016.
[20] L. Brignone, M. Perrier, and C. Viala, "A fully autonomous docking strategy for intervention AUVs," in *Proc. OCEANS Eur.*, Jun. 2007, pp. 1–6.
[21] S. Krupiński, F. Maurelli, G. Grenon, and Y. Petillot, "Investigation of autonomous docking strategies for robotic operation on intervention panels," in *Proc. OCEANS*, 2008, pp. 1–10.
[22] J. Wallen and Z. Song, "Development of an adaptive docking station for resident underwater vehicles," in *Proc. OCEANS-Marseille*, Jun. 2019, pp. 1–7.
[23] E. I. Sarda and M. R. Dhanak, "Launch and recovery of an autonomous underwater vehicle from a station-keeping unmanned surface vehicle," *IEEE J. Ocean. Eng.*, vol. 44, no. 2, pp. 290–299, Apr. 2019.
[24] M. Wirtz, M. Hildebrandt, and C. Gaudig, "Design and test of a robust docking system for hovering AUVs," in *Proc. Oceans*, Oct. 2012, pp. 1–6.
[25] P. W. Kimball, E. B. Clark, M. Scully, K. Richmond, C. Flesher, L. E. Lindzey, J. Harman, K. Huffstutler, J. Lawrence, S. Lelievre, J. Moor, B. Pease, V. Siegel, L. Winslow, D. D. Blankenship, P. Doran, S. Kim, B. E. Schmidt, and W. C. Stone, "The ARTEMIS under-ice AUV docking system," *J. Field Robot.*, vol. 35, no. 2, pp. 299–308, Mar. 2018.
[26] Kawasaki Heavy Industries. *Close-Range Subsea Pipeline Inspection by Autonomous Underwater Vehicle (AUV)*. Accessed: Oct. 10, 2020. https://global.kawasaki.com/en/corp/newsroom/news/detail/?f=20200715_8265
[27] T. Kawasaki, T. Noguchi, T. Fukasawa, S. Hayashi, Y. Shibata, T. Iimori, N. Okaya, K. Fukui, and M. Kinoshita, "'Marine Bird', a new experimental AUV—Results of docking and electric power supply tests in sea trials," in *Proc. Ocean MTS/IEEE Techno-Ocean, Bridges Across Oceans Conf.*, vol. 3, Nov. 2004, pp. 1738–1744.
[28] B. R. Page and N. Mahmoudian, "AUV docking and recovery with USV: An experimental study," in *Proc. OCEANS-Marseille*, Jun. 2019, pp. 1–5.
[29] A. Kukulya, A. Plueddemann, T. Austin, R. Stokey, M. Purcell, B. Allen, R. Littlefield, L. Freitag, P. Koski, E. Gallimore, J. Kemp, K. Newhall, and J. Pietro, "Under-ice operations with a REMUS-100 AUV in the arctic," in *Proc. IEEE/OES Auto. Underwater Vehicles*, Sep. 2010, pp. 1–8.
[30] B. Allen, T. Austin, N. Forrester, R. Goldsborough, A. Kukulya, G. Packard, M. Purcell, and R. Stokey, "Autonomous docking demonstrations with enhanced REMUS technology," in *Proc. OCEANS*, Sep. 2006, pp. 1–6.
[31] J.-Y. Park, B.-H. Jun, P.-M. Lee, J.-H. Oh, and Y.-K. Lim, "Underwater docking approach of an under-actuated AUV in the presence of constant ocean current," *IFAC Proc. Volumes*, vol. 43, no. 20, pp. 5–10, Sep. 2010.

[32] K. Teo, E. An, and P.-P.-J. Beaujean, "A robust fuzzy autonomous underwater vehicle (AUV) docking approach for unknown current disturbances," *IEEE J. Ocean. Eng.*, vol. 37, no. 2, pp. 143–155, Apr. 2012.
[33] J.-Y. Park, B.-H. Jun, K. Kim, P.-M. Lee, J.-H. Oh, and Y.-K. Lim, "Improvement of vision guided underwater docking for small AUV ISiMI," in *Proc. OCEANS*, Oct. 2009, pp. 1–5.
[34] A. Sans-Muntadas, K. Y. Pettersen, E. Brekke, and V. F. Henriksen, "A hybrid approach to underwater docking of AUVs with cross-current," in *Proc. OCEANS MTS/IEEE Monterey*, Sep. 2016, pp. 1–7.
[35] N. Palomeras, A. El-Fakdi, M. Carreras, and P. Ridao, "COLA2: A control architecture for AUVs," *IEEE J. Ocean. Eng.*, vol. 37, no. 4, pp. 695–716, Oct. 2012.
[36] R. C. Craig, "Implementation of the pure pursuit path tracking algorithm," Tech. Rep. [Online]. Available: https://apps.dtic.mil/sti/citations/ADA255524
[37] N. Hurtos, A. Mallios, N. Palomeras, J. Bosch, G. Vallicrosa, E. Vidal, D. Ribas, N. Gracias, M. Carreras, and P. Ridao, "LOON-DOCK: AUV homing and docking for high-bandwidth data transmission," in *Proc. OCEANS Aberdeen*, Jun. 2017, pp. 1–7.
[38] J.-Y. Park, B.-H. Jun, P.-M. Lee, Y.-K. Lim, and J.-H. Oh, "Docking problem and guidance laws considering drift for an underactuated AUV," in *Proc. OCEANS IEEE-Spain*, Jun. 2011, pp. 1–7.
[39] B. W. Hobson, R. S. McEwen, J. Erickson, T. Hoover, L. McBride, F. Shane, and J. G. Bellingham, "The development and ocean testing of an AUV docking station for a 21' AUV," in *Proc. OCEANS*, Sep. 2007, pp. 1–6.
[40] T. I. Fossen, "Marine control systems," *J. Guid. Control Dyn.*, vol. 28, no. 9, pp. 1689–1699, 2002.
[41] K. Teo, B. Goh, and O. K. Chai, "Fuzzy docking guidance using augmented navigation system on an AUV," *IEEE J. Ocean. Eng.*, vol. 40, no. 2, pp. 349–361, Apr. 2015.
[42] W. Caharija, K. Y. Pettersen, M. Bibuli, P. Calado, E. Zereik, J. Braga, J. T. Gravdahl, A. J. Sorensen, M. Milovanovic, and G. Bruzzone, "Integral line-of-sight guidance and control of underactuated marine vehicles: Theory, simulations, and experiments," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 5, pp. 1623–1642, Sep. 2016.
[43] M. Carreras, C. Candela, D. Ribas, A. Mallios, L. L. Magí, E. Vidal, N. Palomeras, and P. Ridao, "Sparus II, design of a lightweight hovering AUV," in *Proc. 5th Int. Workshop Marine Technol. (Martech)*, 2013, pp. 152–155.
[44] M. Carreras, J. D. Hernandez, E. Vidal, N. Palomeras, D. Ribas, and P. Ridao, "Sparus II AUV—A hovering vehicle for seabed inspection," *IEEE J. Ocean. Eng.*, vol. 43, no. 2, pp. 344–355, Apr. 2018.
[45] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "ROS: An open-source robot operating system," in *Proc. ICRA Workshop Open Source Softw.*, 2009.
[46] J. Tong, X. Xu, L. Hou, Y. Li, J. Wang, and L. Zhang, "An ultra-short baseline positioning model based on rotating array & reusing elements and its error analysis," *Sensors*, vol. 19, no. 20, p. 4373, Oct. 2019.

JOAN ESTEBA was born in Girona, Spain, in 1994. He received the bachelor's degree in industrial technologies engineering and the M.Sc. degree in industrial engineering from the University of Girona (UdG), in 2016 and 2018, respectively, where he is currently pursuing the Ph.D. degree with the Computer Vision and Robotics Research Institute (VICOROB). He has been working for several years developing vision and robotic technologies in the automobile industry and leading different projects. He has been leading some development projects in the food industry. He is a Registered Professional Engineer in Catalonia. His research interest includes docking systems for autonomous underwater robotic systems.

PATRYK CIEŚLAK was born in Krakow, Poland, in 1986. He received the Ph.D. degree from the Department of Robotics and Mechatronics, AGH University of Science and Technology, Kraków, in 2016. During his Ph.D. studies, he was involved in several projects focused around mechatronic design and control design in mobile robotics and manipulator systems. In his Ph.D. thesis, he tackled a difficult problem of designing a new balancing mono-wheel robot, from the mechanical structure, through model-based control, and to the software and firmware architecture. Since 2017, he has been working with the Underwater Vision and Robotics Lab (CIRS), University of Girona, Spain. He is currently a Postdoctoral Researcher with the Computer Vision and Robotics Research Institute (VICOROB), University of Girona. He started his work at CIRS as a Marie Skłodowska-Curie Postdoctoral Fellow, after being awarded a grant for a project aimed at developing control algorithms to enable compliant control for an I-AUV performing an autonomous non-destructive testing operation. During this project, he visited Heriot-Watt University, Edinburgh, U.K., where he worked at the Ocean Systems Laboratory (OSL), for a period of two months. After the successful completion of the grant-supported project, he continued his research in underwater manipulation, tackling tasks related to obstacle avoidance, motion planning, and cooperative manipulation utilising two I-AUVs. He is continuing his research at CIRS, working on underwater intervention for inspection, maintenance, and repair (IMR) operations in offshore wind farms. He is the coauthor of a commercial rehabilitation robot called Prodrobot, which is a stationary lower limbs exoskeleton, used in the learning and improvement of natural gait patterns in children. He is the author of an advanced open-source robot simulator called Stonefish, designed for marine robotics, which simulator is currently used in all underwater related research at CIRS, other institutions around the world, and can be found on Github.

NARCÍS PALOMERAS (Member, IEEE) was born in Vilafant, Spain, in 1981. He received the Ph.D. degree from the University of Girona, Spain, in 2011. He is currently a Postdoctoral Researcher with the Computer Vision and Robotics Research Institute (VICOROB), University of Girona. He is the Coordinator of a Joint Degree Erasmus Mundus on Intelligent Field Robotic Systems. He has participated in several research projects, all related to underwater robotics, both national and European, including TRIDENT, PANDORA, MORPH, MERBOTS, LOONDOCK, TWINBOT, 3DAUV, and ATLANTIS, as well as in different European competitions for AUVs, such as SAUC-E and ERL. His research interests include underwater robotics in topics, like planning, exploration, intelligent control architectures, mission control, and localization.

PERE RIDAO (Member, IEEE) was born in Girona, Spain, in 1969. He received the Ph.D. degree in industrial engineering from the University of Girona, Spain, in 2001. He has directed nine Ph.D. theses (with four more currently under direction) and 14 M.S. theses. He is currently the Director of the Computer Vision and Robotics Research Institute (VICOROB), the Head of the Underwater Robotics Research Center (CIRS), and an Associate Professor with the Department of Computer Engineering, University of Girona. He is the coauthor of four licenses and one Spanish/European patent and the author of more than 100 publications. He is a co-founder of Iqua Robotics S.L.—a spin-off company. Since 1997, he has been participating in 24 research projects (15 European and nine National). His research interests include designing and developing autonomous underwater vehicles for 3-D mapping and intervention. He has served as the Chair for the IFAC's Technical Committee on Marine Systems.