Journal of Robotics and Control (JRC)
Volume 2, Issue 5, September 2021
ISSN: 2715-5072 DOI: 10.18196/jrc.25112

Assessment of Collision Avoidance Strategies for an Underwater Transportation System

Faheem Ur Rehman[1], Enrico Anderlini[2], Giles Thomas[3]
[1,2,3] Department of Mechanical Engineering, University College London, United Kingdom
Email: [1]faheem.rehman.11@ucl.ac.uk, [2] e.anderlini@ucl.ac.uk, [3]giles.thomas@ucl.ac.uk
Corresponding Author: Faheem Ur Rehman

***Abstract*—Transportation using multiple autonomous vehicles with detection avoidance capability is useful for military applications. It is important for such systems to avoid collisions with underwater obstacles in an effective way, while keeping track of the target location. In this paper, sensor-based and path-planning methods of external collision avoidance were investigated for an underwater transportation system. In particular, sensor-based wall-following and hard-switching collision avoidance strategies and an offline RRT* path-planning method was implemented on the simulation model of the transportation system of four Hovering Autonomous Underwater Vehicles (HAUVs). Time-domain motion simulations were performed with each method and their ability to avoid obstacles was compared. The hard-switching method resulted in high yaw moments which caused the vehicle to travel towards the goal by a longer distance. Conversely, in the wall-following method, the yaw moment was kept to zero. Moreover, the wall-following method was found to be better than the hard-switching method in terms of time and power efficiency. The comparison between the offline RRT* path-planning and wall-following methods showed that the fuel efficiency of the former is higher whilst its time efficiency is poorer. The major drawback of RRT* is that it can only avoid the previously known obstacles. In future, offline RRT* and wall following can be blended for a better solution. The outcome of this paper provides guidance for the selection of the most appropriate method for collision avoidance for an underwater transportation system.**

***Keywords*—Autonomous Underwater Vehicle (AUV); underwater transportation; multi vehicles; sensor-based methods; path-planning method; PID controller**

I. INTRODUCTION

Autonomous Underwater Vehicles (AUVs) have been widely used for ocean exploration, military and industrial applications [1]–[5]. Due to the demand to increase the time and cost efficiency of underwater tasks, there is a growing interest in applications involving multiple AUVs [6][7]. A larger number of vehicles provides the advantages of fault tolerance, robustness and flexibility [8]. Additionally, multiple simple vehicles can achieve the same task as a more complex, larger and more expensive system.

The novel approach of transportation using multiple AUVs is important for the military sector, as it allows payloads to be moved covertly. Moreover, high precision can be achieved in path following, as the wave effects are negligible, which is important for commercial underwater installations such as oil and gas extraction rigs [9]. However, an obstacle avoidance strategy is critical to enable the successful delivery of underwater transportation,

There are two ways to apply an external collision avoidance strategy. One way is to implement a sensor-based method which activates when the obstacle is detected by the sensors on the vehicle [10]. The other solution is to use a path-planning method which helps to identify the safe path to be followed [11][12][13]. The path-planning could be offline or online. Offline path-planning is carried out before the start of the mission and the generated trajectory can only avoid the previously known obstacles. Conversely, online planning produces an incremental trajectory that allows the reaction to changes in the environment, moving obstacles and errors during motion [14].

A wide range of research has been dedicated to developing collision avoidance strategies using robust controllers [15], [16] [17]. Formation control and switching methods have been used to avoid internal collision between a group of vehicles. For instance, in [18], the virtual structure formation control strategy and the artificial potential field (APF) were integrated to avoid internal collision between a group of unicycle models. In [19], the hard-switching strategy was employed to prevent internal collision. In [20], the switching method was validated by performing experiments on toy model cars for collision avoidance while tracking the desired path. For external collision avoidance with obstacles, the wall-following and path-planning methods are mostly used in the literature. For instance, the wall-following strategy was used for indoor mobile robots in [10] to avoid collision with the obstacles. In [21], the APF-based path-planning method was used along with the real-time feature extraction algorithm to generate a collision-free path for a mobile robot. In [12], rapidly exploring Rapidly-exploring Random Tree Star (RRT*) was used to get the desired path and avoid obstacles for the cooperative aerial transportation system. In addition to RRT*, Dynamic Movement Primitives (DMPs) were used to avoid the unknown obstacles. DMPs modify the trajectory based on the virtual structure formation control strategy.

In the maritime industry, only the path-planning methods can be found for external collision avoidance in the literature. For instance, in [22][13][11], the fast-marching-based path-planning algorithm was used for an Unmanned Surface Vessel (USV) to avoid the collision in a dynamic environment. In [23], Voronoi diagrams and Dijkstras' algorithm were used to obtain a collision-free path for a Remotely Operated Vehicle (ROV). The strategy was also experimentally validated. In [24], the RRT algorithm was

implemented on AUVs and merged with Manoeuvre Automation (MA) to smooth the path.

However, no research has been carried out on the implementation of different types of external collision avoidance strategies on underwater systems, especially for the novel transportation system of multiple AUVs. Therefore, this paper contributes by exploring collision avoidance methods for autonomous underwater transportation systems. Both sensor-based and path-planning strategies are introduced for an underwater transportation system for the first time. The merits and shortcomings of each method in comparison with other methods will provide guidance for the selection of an appropriate collision avoidance strategy.

The rest of the paper is organised as follows. Section **Error! Reference source not found.** describes the underwater transportation system for which the collision avoidance methods are explored. Section **Error! Reference source not found.** explains the dynamic model of the transportation system. In Section III.A, the sensor-based methods are developed i.e. the hard-switching and wall-following techniques; they are implemented on the transportation system encountering a single obstacle in Section 5. In Section 6, the wall-following and RRT* path-planning methods are implemented on the transportation system encountering four obstacles. Finally, concluding remarks are given in Section 7.

II. SYSTEM DESCRIPTION AND MODELLING

The collision avoidance methods were implemented on a novel transportation system of four Minerva Hovering Autonomous Underwater Vehicles (HAUVs), which are connected to a cubic payload via solid links, as shown in Fig. 1. The system was developed in [25] for the safe transportation of a payload underwater.

Each vehicle in the system was assumed to have one obstacle detection sensor installed. The distance from the centre of the system to each sensor is 3.2m. This includes the length of the portion of the cubic payload (0.5m), the solid link (1m) and the length of the vehicle (1.52m).

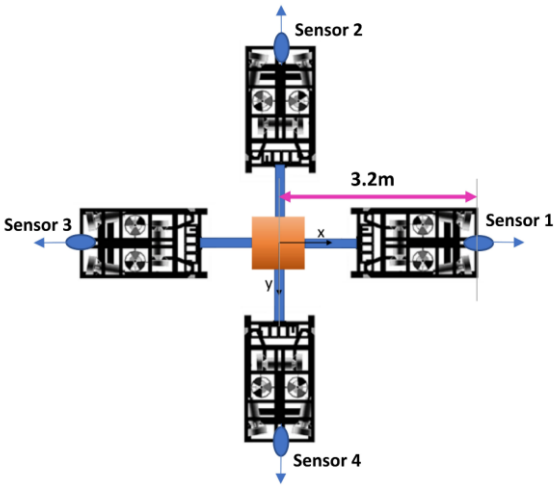


Fig. 1. Underwater transportation system of four HAUVs connected to a cubic payload via solid links

A dynamic model was first developed for the above-mentioned transportation system to analyse the obstacle collision avoidance. The following assumptions were made:

- The whole system is a rigid body of constant mass. This is a valid assumption as all the part bodies are made of rigid material and are fastened rigidly.
- The mass distribution does not change during motion. This is also valid as the HAUVs and their connections to the payload. remain the same as they were at the start of the mission.
- The surface waves are ignored as the system is operating at a depth where wave effects are negligible.
- The sea current effects are ignored to ease the analysis and to get a better preliminary comparison between different types of collision avoidance strategies.
- The limitations of the sensors installed on the system such as sensor noise are ignored.
- The water density remains constant.

The dynamic model consists of the kinematics and kinetics. Kinematics describes the motion of the body without accounting for the forces and moments which are taken up by the kinetics [26]. Fossen's approach [27] was used to develop the dynamic model according to which the position and orientation are taken in the Earth-fixed frame (EFF) whereas, the velocities, forces and moments are expressed in the Body-fixed frame (BFF). The vectorial representation of the dynamic model is given as [27]

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}\boldsymbol{v}. \tag{1}$$

$$\boldsymbol{M}\dot{\boldsymbol{v}} + \boldsymbol{C}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau}. \tag{2}$$

$\boldsymbol{\eta}$ is the vector of position and orientation of the system, given as

$$\boldsymbol{\eta} = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \psi \end{bmatrix}. \tag{3}$$

$\boldsymbol{v}$ is the velocity vector in the BFF, written as

$$\boldsymbol{v} = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix}. \tag{4}$$

$\boldsymbol{J}$ is the transformation matrix to transform $\boldsymbol{v}$ from BFF to EFF. $\boldsymbol{M}$ is the mass matrix which is the sum of the rigid body mass ($\boldsymbol{M_{RB}}$) and added mass ($\boldsymbol{M_A}$) matrices. $\boldsymbol{C}(\boldsymbol{v})$ is the Coriolis matrix which takes into account the rotational effect of BFF about the EFF. This consists of the rigid body Coriolis matrix $\boldsymbol{C_{RB}}(\boldsymbol{v})$ and added mass Coriolis matrix $\boldsymbol{C_A}(\boldsymbol{v})$. $\boldsymbol{D}(\boldsymbol{v})$ is the damping matrix which consists of the linear and quadratic damping terms. $\boldsymbol{g}(\boldsymbol{\eta})$ is the vector of hydrostatic forces and moments.

The transformation matrix $\boldsymbol{J}$ is written as

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{T_{3\times3}} & \boldsymbol{O_{3\times3}} \\ \boldsymbol{O_{3\times3}} & \boldsymbol{R_{3\times3}} \end{bmatrix}. \tag{5}$$

Where $\boldsymbol{T}$ is the transformation matrix to transform the translational terms and $\boldsymbol{R}$ to transform the rotational terms, given as

$$\boldsymbol{T} = \begin{bmatrix} \cos\psi\cos\theta & \begin{pmatrix}-\sin\psi\cos\phi \\ +\cos\psi\sin\theta\sin\phi\end{pmatrix} & \begin{pmatrix}\sin\psi\sin\phi \\ +\cos\psi\sin\theta\cos\phi\end{pmatrix} \\ \sin\psi\cos\theta & \begin{pmatrix}\cos\psi\cos\phi \\ +\sin\psi\sin\theta\sin\phi\end{pmatrix} & \begin{pmatrix}-\cos\psi\sin\phi \\ +\sin\psi\sin\theta\cos\phi\end{pmatrix} \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix}. \quad (6)$$

$$\boldsymbol{R} = \begin{bmatrix} 1 & \tan\theta\sin\phi & \tan\theta\cos\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix}. \quad (7)$$

$\boldsymbol{\tau}$ is the thrust vector which can be written as the product of thrust allocation matrix ($\boldsymbol{T_a}$) and vector of thrust forces applied by the thrusters on the system ($\boldsymbol{f}$), given as

$$\boldsymbol{\tau} = \boldsymbol{T_a f}. \quad (8)$$

In this paper, the collision avoidance of the transportation system was analysed in the horizontal plane. Therefore, the dynamic model was reduced by considering only (x, y, ψ) coordinates.

The model parameters of the transportation system were acquired from [25], where the parameters of the Minerva HAUV were taken from [28] and then modified to get their effect about the centre of the combined body, i.e. the centre of the payload [29]. For the manipulators and payload, the hydrodynamic parameters were calculated using a semi-empirical approach.

III. COLLISION AVOIDANCE METHODS

The sensor-based and path-planning methods are developed in this section. Due to the non-availability of any previous data regarding the implementation of collision avoidance strategies on the system under study, several tests were initially performed to verify the outcomes.

A. Sensor-Based Collision Avoidance

Two sensor-based collision avoidance methods were developed for the transportation system: the hard-switching and wall-following strategies.

In both methods, the obstacles are detected by the vehicles' sensors. Different sources of underwater sensing can be used. The acoustic source is good for long-ranges but they are not efficient at short-range due to low bit rates [30]. In contrast, the Electromagnetic (EM) waves in the Radio Frequency (RF) range provide the highest bit rates for short ranges, but they are highly attenuated in water over long ranges [31]. Due to the requirement of short-range sensing, the infrared proximity sensors [32] were assumed to be used on the transportation system.

1) Hard-Switching

The strategy for the hard-switching method is shown in Fig. 2. Two controllers are applied: the go-to-goal and avoid-obstacle controllers. The system moves towards the goal using the go-to-goal controller until the norm of the difference between the obstacle position ($\boldsymbol{\eta_{obs}}$) and sensor position ($\boldsymbol{\eta_{si}}$) is less than or equal to the maximum allocated distance around the obstacle (d_s) at which the avoid-obstacle controller gets activated. The thrust vectors are decided accordingly. For instance, when the condition for go-to-goal is met, the go-to-goal thrust vector ($\boldsymbol{\tau_{gtg}}$) is applied and when the avoid obstacle controller is activated, the avoid-obstacle thrust vector ($\boldsymbol{\tau_{ao}}$) is implemented, given as

$$\boldsymbol{\tau} = \begin{cases} \boldsymbol{\tau_{gtg}} & ||\boldsymbol{\eta_{obs}} - \boldsymbol{\eta_{si}}|| > d_s \\ \boldsymbol{\tau_{ao}} & ||\boldsymbol{\eta_{obs}} - \boldsymbol{\eta_{si}}|| \leq d_s \end{cases} \quad (9)$$

a) Go-to-Goal Controller

The control scheme is applied on the difference between the desired ($\boldsymbol{\eta_{dgtg}}$) and actual position ($\boldsymbol{\eta}$) of the system. The outcome is given as desired thrust vector in EFF ($\boldsymbol{\tau_{egtg}}$), which is multiplied by the inverse of transformation matrix ($\boldsymbol{J}$) to get the effect in the BFF ($\boldsymbol{\tau_{dgtg}}$). $\boldsymbol{\tau_{dgtg}}$ is multiplied by the inverse of the thrust allocation matrix ($\boldsymbol{T_a}$) to get the desired force vector ($\boldsymbol{f_{d_{gtg}}}$). A saturation limit is applied on the thrust forces to get the actual thrust force vector ($\boldsymbol{f_{gtg}}$). This is then multiplied by $\boldsymbol{T_a}$ to get the actual thrust vector ($\boldsymbol{\tau_{gtg}}$) which is given as control input to the dynamic model of the system so that the actual position of the system ($\boldsymbol{\eta}$) can be obtained. The process continues until the desired goal location is achieved or the controller switches to the avoid-obstacle controller.

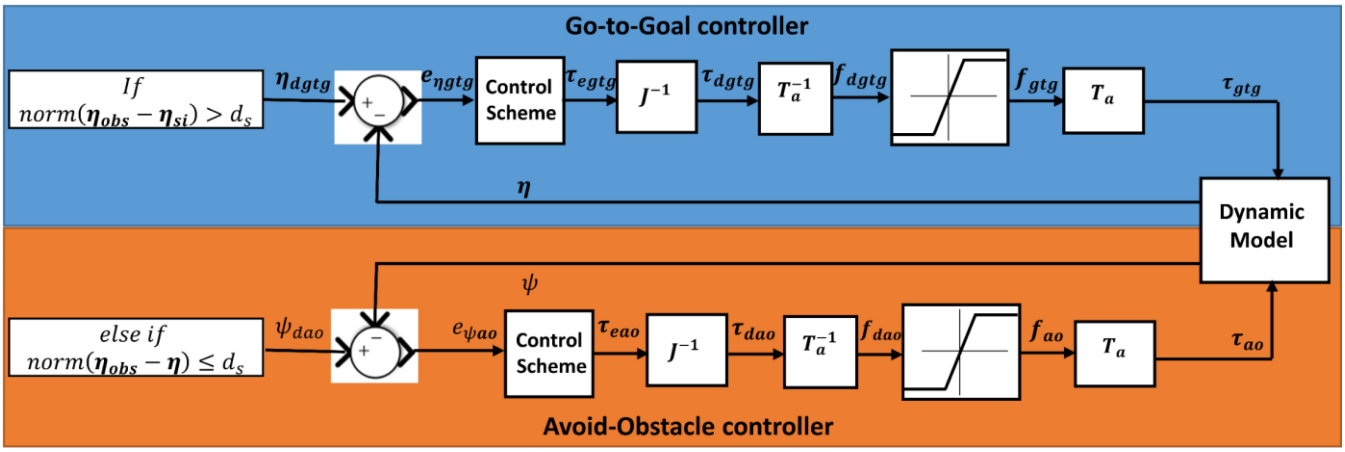


Fig. 2. Hard-Switching strategy

b) Avoid-Obstacle Controller

The avoid-obstacle controller turns the system clockwise by 90° when the obstacle is detected by the sensor on the vehicle. Therefore, the desired heading angle in the avoid-obstacle condition becomes

$$\psi_{dao} = \psi + \frac{\pi}{2}. \tag{10}$$

The control scheme is applied on the difference between the desired heading angle (ψ_{dao}) and the actual heading angle (ψ) to get the desired thrust moment ($\tau_{\psi dao}$). The desired thrust vector to avoid the obstacle in EFF becomes

$$\boldsymbol{\tau_{eao}} = [0 \;\; 0 \;\; \tau_{\psi dao}]^t. \tag{11}$$

$\boldsymbol{\tau_{eao}}$ is then multiplied by the inverse of $\boldsymbol{J}$ to get the desired thrust vector in BFF ($\boldsymbol{\tau_{d_{ao}}}$). This is then multiplied by the inverse of $\boldsymbol{T_a}$ to get the desired thrust force vector ($\boldsymbol{f_{d_{ao}}}$). The saturation limit is then applied to get the actual thrust force vector ($\boldsymbol{f_{gtg}}$) which is multiplied by $\boldsymbol{T_a}$ to get the actual thrust vector ($\boldsymbol{\tau_{ao}}$). $\boldsymbol{\tau_{ao}}$ is then given as input to the dynamic model of the system which provides ψ. The process continues until the system is out of the danger zone and the go-to-goal controller takes over.

2) Wall-Following

A wall-following strategy is developed in two phases for the go-to-goal and avoid-obstacle behaviours: the planning phase and the tracking phase. In the first phase, the desired velocity vector is achieved by applying a linear control law. In the second phase, the control scheme is applied to get the desired control input.

The complete wall-following strategy is shown in Fig. 3.

a) Go-to-Goal

In the first phase, a linear feedback controller is applied on the difference between the goal position ($\boldsymbol{\eta_{goal}}$) and the actual position of the system ($\boldsymbol{\eta}$). This provides the desired velocity vector in EFF ($\dot{\boldsymbol{\eta}}_{\boldsymbol{gtg}}$), given as

$$\dot{\boldsymbol{\eta}}_{\boldsymbol{gtg}} = \boldsymbol{K_{gtg}}\left(\boldsymbol{\eta_{goal}} - \boldsymbol{\eta}\right) = \boldsymbol{K_{gtg}}\boldsymbol{e_{\eta gtg}}. \tag{12}$$

$\boldsymbol{K_{gtg}}$ is the go-to-goal gain matrix. To check the asymptotic stability, the error dynamics is taken as

$$\dot{\boldsymbol{e}}_{\boldsymbol{\eta}} = -\dot{\boldsymbol{\eta}} = -\boldsymbol{K_{gtg}}\boldsymbol{e_{\eta gtg}}. \tag{13}$$

For the positive $\boldsymbol{K_{gtg}}$, the eigenvalues are all positive. Therefore, the error between $\boldsymbol{\eta_{goal}}$ and $\boldsymbol{\eta}$ asymptotically goes to zero. This ensures the closed-loop stability of the system.

In the next phase, the desired velocity vector in BFF ($\boldsymbol{v_{dgtg}}$) is obtained by multiplying $\dot{\boldsymbol{\eta}}_{\boldsymbol{gtg}}$ with the inverse of $\boldsymbol{J}$. The desired heading angle (ψ_{dgtg}) is worked out by taking the tangent inverse of the ratio between the sway velocity (v) and surge velocity (u). The control scheme is then applied to get the desired motion response.

b) Collision Avoidance

In the first phase, the desired avoid-obstacle velocity vector ($\dot{\boldsymbol{\eta}}_{\boldsymbol{ao}}$) is calculated by multiplying a scalar avoid-obstacle gain matrix ($\boldsymbol{K_{ao}}$) to the difference between the obstacle position ($\boldsymbol{\eta_{obs}}$) and the actual position of the system ($\boldsymbol{\eta}$).

$$\dot{\boldsymbol{\eta}}_{\boldsymbol{ao}} = \boldsymbol{K_{ao}}(\boldsymbol{\eta} - \boldsymbol{\eta_{obs}}) = \boldsymbol{K_{ao}}\boldsymbol{e_{\eta ao}}. \tag{14}$$

$$\dot{\boldsymbol{e}}_{\boldsymbol{\eta ao}} = \dot{\boldsymbol{\eta}} = \boldsymbol{K_{ao}}\boldsymbol{e_{\eta ao}}. \tag{15}$$

$\boldsymbol{K_{ao}}$ is positive definite; therefore, the eigenvalues are positive and the system is unstable. This is because the system is moving away from the obstacle. If there is no other controller, the system will keep on moving away from the obstacle to infinity. However, when the system is out of the obstacle influential area, the go-to-goal controller takes over. Therefore, this choice of the control input for the obstacle collision avoidance is acceptable.

Using $\dot{\boldsymbol{\eta}}_{\boldsymbol{ao}}$ mentioned in equation (14), the system can get trapped in the local minima, i.e. the go-to-goal and the avoid-obstacle velocity vectors could come directly opposite to each other. This is quite unlikely in reality, as a small amount of noise can avoid the system ending up at the local minima [33]. However, to develop a safer algorithm, the avoid-obstacle velocity vector can be rotated by ±90°. This way the system can follow the obstacle either clockwise or anticlockwise.

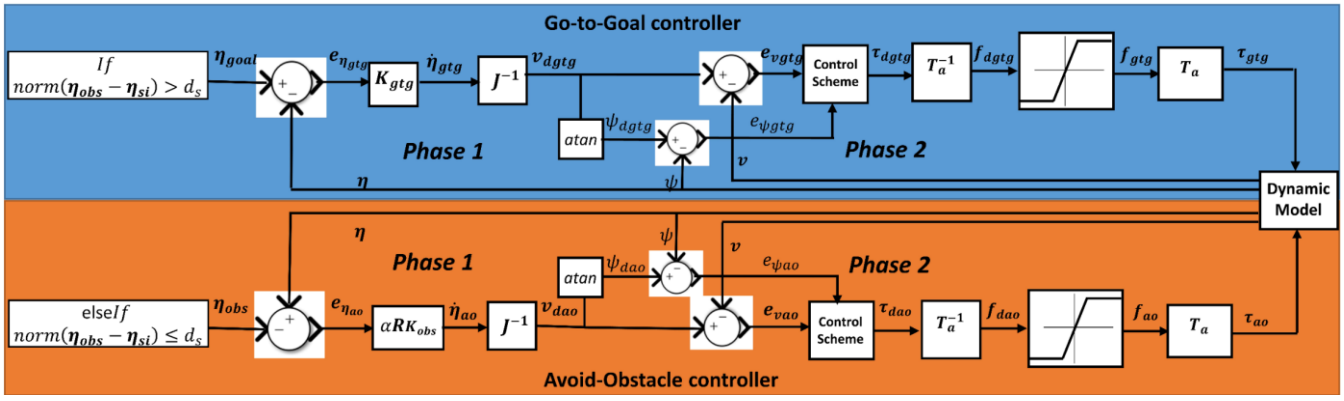


Fig. 3. Wall-Following strategy

$$\dot{\boldsymbol{\eta}}_{\boldsymbol{ao}} = \alpha \boldsymbol{R}\boldsymbol{K_{ao}}\boldsymbol{e_{obs}}. \tag{16}$$

α is the scaling factor which must be carefully selected in the control system design, $\boldsymbol{R}$ is the transformation matrix

which can transform the velocity vector by clockwise or anti clockwise.

For clockwise,

$$\boldsymbol{R} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \quad (17)$$

And for counter-clockwise,

$$\boldsymbol{R} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}. \quad (18)$$

To decide which direction to transform the avoid-obstacle velocity vector, the dot product of the velocity vector in each clockwise ($\dot{\boldsymbol{\eta}}_{\boldsymbol{fw,c}}$) and anticlockwise ($\dot{\boldsymbol{\eta}}_{\boldsymbol{fw,cc}}$) are taken with the go-to-goal velocity vector ($\dot{\boldsymbol{\eta}}_{\boldsymbol{gtg}}$). If the angle between the go-to-goal and any of the clockwise or anticlockwise velocity vectors is less than 90°, its dot product will be positive and the distance between the system and goal will be lower. Therefore, it is desired to follow the obstacle in that direction. Moreover, it is required to specify when the system should stop following the wall. This is obtained by taking the dot product of the avoid-obstacle velocity vector ($\dot{\boldsymbol{\eta}}_{\boldsymbol{ao}}$) and the go-to-goal velocity vector ($\dot{\boldsymbol{\eta}}_{\boldsymbol{gtg}}$). If the dot product is greater than zero then the go-to-goal controller gets activated. The flow chart of the algorithm is shown in Fig. *4*.

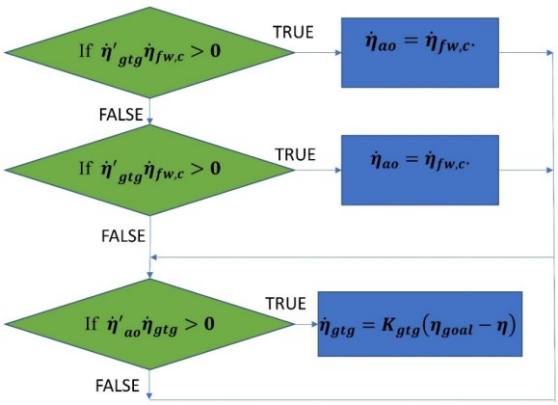


Fig. 4. Flow chart of the algorithm which is generated for the wall-following method of obstacle avoidance to decide on which direction to follow the obstacle and when to stop following

In the next phase, the desired avoid-obstacle velocity vector in BFF ($\boldsymbol{v}_{\boldsymbol{dao}}$) and the heading angle (ψ_{dao}) are worked out in the same way as for the go-to-goal controller. Finally, the control scheme is applied to obtain the desired motion response which avoids the collision with the obstacle.

3) RRT Path-Planning*

The path-planning method aims to select a collision-free path but at the same time cost-optimal [34]. In simple words, the path should be free of obstacles and the shortest possible. Here, the focus is on the Sampling-Based Path-Planning (SBP) algorithms. To improve exploration and enable the discovery of global optima for complex problems, the SBP algorithms apply randomised sampling in the search space [35]. The Rapidly exploring Random Tree (RRT) was developed based on the SBP algorithm in [36], showing satisfactory performance for problems with twelve degrees of freedom. However, the path generated with RRT was not optimal. Therefore, a modified form of RRT was developed known as RRT* [37]. The path quality was improved with RRT* as it introduces the features of tree rewiring and precise neighbour search. However, the optimal solutions were obtained with RRT* at the cost of higher execution time and slower path convergence rate [34]. This problem was solved by introducing RRT* Smart which improves the convergence rate of RRT* [38]. In this paper, the RRT* algorithm was used in MATLAB to get the desired waypoints.

A trajectory is generated using the waypoints provided by the path-planning method. In trajectory generation, the path between waypoints is called a segment which is the best possible route between the two adjacent waypoints. For n waypoints, there are $n-1$ segments. The most implemented of the path trajectories are minimum jerk trajectory and minimum snap trajectory [9], [39]–[41].

In this paper, the minimum snap trajectory was generated which defines the segment by a seventh order polynomial, given as [9] [41] [42].

$$p_i(t) = \alpha_{i0} + \alpha_{i1}\frac{t - S_{i-1}}{T_i} + \alpha_{i2}\left(\frac{t - S_{i-1}}{T_i}\right)^2 + \alpha_{i3}\left(\frac{t - S_{i-1}}{T_i}\right)^3 + \cdots + \alpha_{i7}\left(\frac{t - S_{i-1}}{T_i}\right)^7. \quad (19)$$

Where i is the number of waypoints, $p_i(t)$ is the polynomial of the path between two waypoints, T_i is the time step and S_i is the total time from the start till the end of each segment.

To solve for $p_i(t)$, the coefficients α_{ij} are required to be solved. There are a total of $8n$ coefficients which require $8n$ constraints to get a solution. These constraints are obtained considering 1) the trajectory passes through the waypoints, 2) velocity, acceleration and jerk are zero at the start and end of the trajectory, 3) velocity, acceleration and 3rd to 6th order derivatives of the polynomial are continuous [41] [42].

IV. RESULTS AND DISCUSSION

The methods of external collision avoidance were implemented on the transportation system which was described in Section II. First, the hard-switching and wall-following methods were implemented and the results are discussed and compared. This is followed by the implementation of wall-following and path-planning methods and their results are compared.

A. Implementation of the Hard-Switching and Wall-Following Methods

The hard-switching and wall-following methods were first implemented on the transportation system encountering a single circular obstacle positioned at (50m, 40m). The effective area of the obstacle was considered as a sphere with a radius of 20m. The sensor's detection range was assumed to be 5m. This results in a safe distance of 25m from the centre of the obstacle at which the avoid-obstacle controller gets activated. The goal location was considered at (100m,100m).

1) Hard-Switching

The hard-switching method was first implemented on the transportation system encountering the obstacle. The control

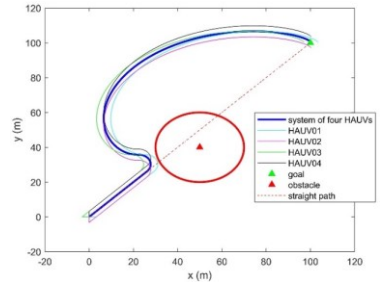
scheme was selected to be the PID controller. The PID gains (i.e. Proportional (K_P), Integral (K_I) and Differential (K_D)) at which a suitable motion response was obtained are shown in TABLE 1. The decrease in these gains would have implications. For instance, if the proportional gain of avoid-obstacle controller is reduced, the system will penetrate the danger zone and could even collide with the obstacle. Conversely, if the proportional gain of the go-to-go controller is reduced, the motion response gets slower. On the other hand, any further increase in the gains would not improve the motion response.

TABLE 1. PID GAINS FOR THE HARD-SWITCHING METHOD FOR THE TRANSPORTATION SYSTEM ENCOUNTERING A SINGLE OBSTACLE

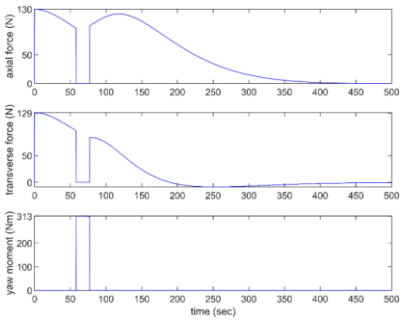
	K_P	K_I	K_D
Avoid obstacle	200	0	1
Go-to-goal	1.3	0	0

As can be seen in Fig. 5(a), the transportation system avoids the obstacle. The system was moving towards the goal until one of the sensors on the system detected the obstacle. At this point, the avoid-obstacle controller was activated which turned the system by 90º.

Fig. 5(b) shows the thrust response of the system. An initial axial and transverse thrust force of 130N and 129N are applied respectively to accelerate the system towards the goal location. The forces then start decreasing and reduce to zero at the point where the controller to avoid the obstacle gets activated. At this point, a yaw moment of 313Nm is applied to turn the system around the circular obstacle. Due to the high yaw moment, the system took a longer distance around the circular obstacle to reach the goal location.



(a) Displacement of the system in the horizontal plane



(b) Thrust and torque applied by the controller

Fig. 5. Response of the system encountering a single obstacle with the hard-switching method for obstacle avoidance

2) *Wall-Following*

The wall-following method was then applied on the transportation system. As the system of HAUVs is over-actuated, the heading angle was kept to zero throughout by applying a separate controller. This cannot be achieved in the hard-switching method where the avoid-obstacle controller only relies on the turning of the system.

PID controllers were used for which the PID gains are shown in TABLE 2.

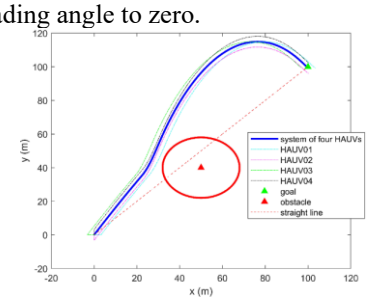
TABLE 2. PID GAINS FOR THE WALL-FOLLOWING METHOD FOR THE TRANSPORTATION SYSTEM ENCOUNTERING A SINGLE OBSTACLE

	Go-to-goal	Avoid-obstacle
K_{Px}	3	3
K_{Ix}	0	0
K_{Dx}	5	1
K_{Py}	5	6
K_{Iy}	0	0
K_{Dy}	10	5
$K_{P\psi}$	5	2
$K_{I\psi}$	0	0
$K_{D\psi}$	1	1

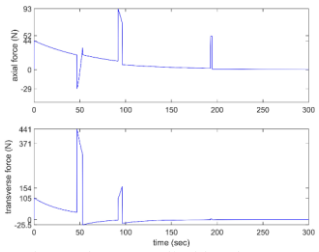
The scaling factor α was set to 3, and the diagonal terms of the wall-following gain matrices $\boldsymbol{K_{ao}}$ and $\boldsymbol{K_{gtg}}$ to 1 and 0.5 respectively.

As can be seen in Fig. 6(a), the transportation system avoided the circular obstacle. The system started moving towards the goal until one of the sensors on the system detects the circular obstacle, at which point the system began to follow the obstacle's boundary clockwise. Once clear of the obstacle, the system moved back to approaching the goal.

In Fig. 6(b), thrust forces of 150N and 250N are initially applied in surge and sway, respectively, to accelerate the system towards the goal. When one of the sensors on the system detects the obstacle, the axial thrust force first decreases to -54N to decelerate the system and then increases to 187N to increase its pace around the obstacle. Similarly, the transverse thrust force increases to 487N to accelerate around the obstacle. The forces then decrease to zero gradually as the system approaches the goal location. No yaw moment is observed as a separate controller was applied to keep the heading angle to zero.



(a) Displacement of the system in the horizontal plane



(b) Thrust applied by the controller

Fig. 6. Response of the system encountering a single obstacle with the wall-following method for obstacle avoidance

3) Comparison between Hard-Switching and Wall-Following

The comparison between the hard-switching and wall-following methods has proven that wall-following is better for the underwater transportation system due to the following:

1. The resultant distance travelled by the system in the wall-following while avoiding the obstacle was slightly lower compared to the hard-switching, as shown in TABLE 3.
2. The time taken by the system with the wall-following method is significantly lower than for the hard-switching scheme. The wall-following approach took around 350secs to reach the goal location, whereas the hard-switching controller took around 500secs, as shown in TABLE 3. Therefore, the wall-following method is more time-efficient.
3. The yaw moment which was produced to avoid the obstacle was very high, i.e. 313Nm, in the hard-switching approach, as shown in TABLE 3. This resulted in high yaw angles which would cause the system to significantly steer far from its course. This could not be avoided, as the avoid-obstacle controller completely relies on the yaw moment for turning the system. Conversely, no yaw moment was applied by the wall-following method, as no yaw moment was necessary to avoid the obstacle. Instead, the combination of surge and sway thrust forces would achieve the desired objective by exploiting the over-actuated nature of the HAUVs. This solution results in a smoother response of the transportation system.

TABLE 3. TRAJECTORY CHARACTERISTICS OF THE SYSTEM WITH THE HARD-SWITCHING AND WALL-FOLLOWING METHODS

	Hard-Switching	Wall-Following
Resultant distance (m)	141.8	141.2
Net time taken (secs)	500	350
Yaw moment applied (Nm)	313	0

4. Higher PID gains were required in the hard-switching method, to which the system is highly sensitive. A slight decrease in the go-to-goal controller gain or a slight increase in the avoid-obstacle controller gain would move the system unstably away from the goal, resulting in mission failure. On the other hand, a slight decrease in the avoid-obstacle controller gain or a slight increase in the go-to-goal controller gain would result in the system penetrating in the obstacle danger zone. Conversely, low PID gains were sufficient to accomplish the transportation task with the wall-following method. Additionally, the system is not sensitive to the controller's grains with the wall-following method
5. From the Root Mean Square (RMS) values in TABLE 4, the net power consumption with the hard swithing method is higher than the wall-following method. This is mainly due to the very high power required to turn the system to avoid the obstacle with the hard-switching method.

TABLE 4. *RMS* OF THE POWER CONSUMED WITH THE HARD-SWITCHING AND WALL-FOLLOWING METHODS WHEN THE TRANSPORTATION SYSTEM ENCOUNTERS A SINGLE OBSTACLE

	Hard-Switching	Wall-Following
RMS_X(Watt)	41.08	29.84
RMS_Y(Watt)	21.22	57.50
RMS_M(Watt)	88.98	0
Net (Watt)	151.28	87.34

B. Implementation of the Wall-Following and Path-Planning Methods

A new, more complex case study is used to compare the performance of the selected wall-following method with the RRT* path-planning strategy for an obstacle avoidance problem. Both approaches were implemented on the transportation system when encountering four circular obstacles of radius 1m each, placed at the positions described in TABLE 5. The detection range of the sensors was assumed to be 1m. The goal was located at (22m, 35m).

TABLE 5. POSITION OF THE OBSTACLES

Position	$(x_{obs}(m), y_{obs}(m))$
obstacle 1	(10,12)
obstacle 2	(20,20)
obstacle 3	(10,33)
obstacle 4	(20,30)

1) Wall-Following

The wall-following method was applied using the PID gains shown in TABLE 6.

TABLE 6. PID GAINS FOR THE WALL-FOLLOWING METHOD FOR THE TRANSPORTATION SYSTEM ENCOUNTERING FOUR OBSTACLES

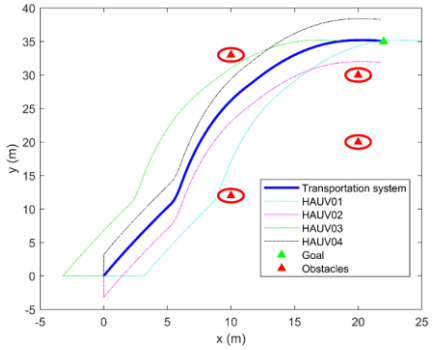
	Go-to-goal controller	Avoid-obstacle controller
K_{Px}	20	20
K_{Ix}	0	0
K_{Dx}	50	5
K_{Py}	30	100
K_{Iy}	0	0
K_{Dy}	100	5
$K_{P\psi}$	5	5
$K_{I\psi}$	0	0
$K_{D\psi}$	0	0

The diagonal gain terms of the wall-following method were set to 0.1 and 1 for $\boldsymbol{K_{gtg}}$ and $\boldsymbol{K_{ao}}$, respectively. The scaling factor α was set to 1 throughout.

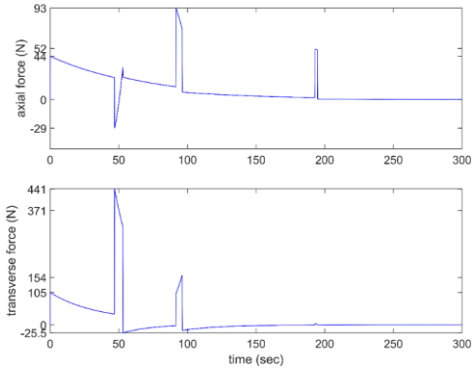
As can be seen in Figure 7(a), the transportation system reached the goal location while avoiding the obstacles.

In Figure 7(b), the thrust forces in both surge and sway initially increase to accelerate the system towards goal. As the first obstacle is detected, a negative axial thrust force of -29N and a positive transverse thrust force of 441N are applied by the system to prevent collision with the obstacle. Subsequently, the axial and transverse thrust forces of 93N and 154N, respectively, are applied to avoid the third obstacle. Finally, an axial thrust force of 52N and a transverse thrust force of 4N are applied to pass by the fourth obstacle,

before the forces drop to zero when the system reaches the goal location.



(a) Displacement of the system in the horizontal plane



(b) Thrust applied by the controller

Figure 7: Response of the system encountering four obstacles with the wall-following method

2) RRT Path-Planning*

Path-planning methods were also studied to generate a safe trajectory avoiding the obstacles. In this regard, the RRT* planner was used to generate a safe path for the transportation system while avoiding the four obstacles, which were positioned as shown in TABLE 5.

The RRT* method searches the map around the system and obtains the shortest route, whilst avoiding the obstacles. This method assumes the moving system to be a point mass; therefore, the obstacles need to be inflated.

In this study, each obstacle was inflated to 4.2m radius to account for the size of the transportation system and the obstacle. The obtained path is shown in Figure 8.

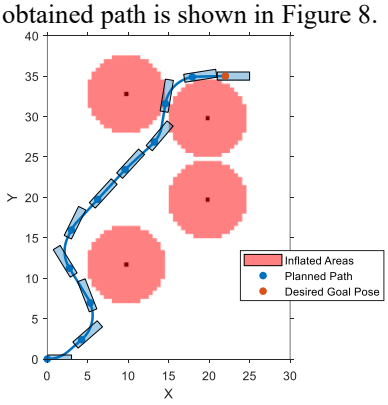


Figure 8: Path generation using RRT* encountering four obstacles

The waypoints that were obtained from RRT* generated path can be seen in TABLE 7. In turn, these waypoints were used to develop a minimum snap trajectory. PID controllers were then applied to follow the desired trajectory.

TABLE 7. WAYPOINTS OBTAINED FROM THE RRT* PATH-PLANNING ALGORITHM FOR THE SYSTEM ENCOUNTERING FOUR OBSTACLES

Waypoints	Position (m)
waypoint 1	(0,0)
waypoint 2	(4.2,2.4)
waypoint 3	(5.3,6.9)
waypoint 4	(2.7,11.2)
waypoint 5	(3,16)
waypoint 6	(6.2,19.7)
waypoint 7	(9.6,23.4)
waypoint 8	(13.2,26.8)
waypoint 9	(14.6,31.6)
waypoint 10	(17.9,34.9)
waypoint 11	(22,35)

The waypoints were used to generate a minimum snap trajectory [41]. The minimum snap trajectory joins the waypoints with a seventh order polynomial segments. The motion response depends on the segment time: a higher segment time improves the precision of the motion response but at the cost of a longer completion time and the application of thrusters at lower and inefficient thrust ranges. Therefore, a compromise needs to be found when selecting the segments' duration.

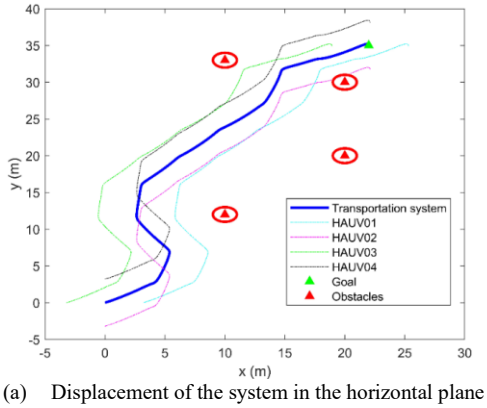
PID controllers were used in the control system design for the transportation system to follow the trajectory, as PID control was found to be effective for underwater payload transportation [43]. The PID gains which were used for the trajectory tracking are shown in TABLE 8. The higher derivative gains were required to avoid overshoot at the end of each segment of the trajectory.

TABLE 8: PID GAINS FOR TRAJECTORY FOLLOWING FOR THE TRANSPORTATION SYSTEM ENCOUNTERING FOUR OBSTACLES

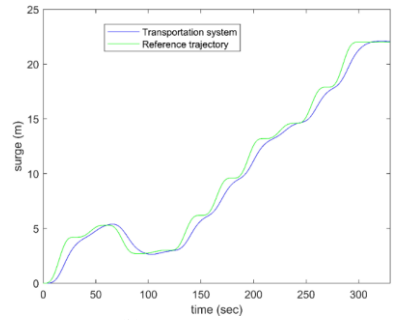
	K_P	K_I	K_D
x	10	0	80
y	20	0	40
ψ	5	0	1

As can be seen in Fig. 9(a), (b) and (c), the trajectory is followed as desired. The Root Mean Square Error ($RMSE$) between the actual and desired motion responses is low as displayed in TABLE 9, which indicates high trajectory tracking precision.

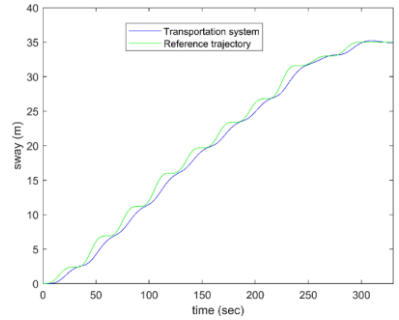
Fig. 9(d) shows the thrust forces applied by the controller to enable the transportation system to follow the desired trajectory in surge and sway. The forces are continuously changing within the reasonable thrust ranges to ensure precise tracking of the trajectory. The yaw moment is zero as the heading angle was separately controlled to be zero.



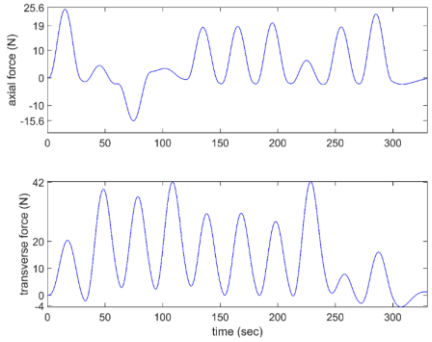
(a) Displacement of the system in the horizontal plane



(b) Surge response



(c) Sway response



(d) Thrust applied by the controller

Fig. 9. Response of the system encountering four obstacles for the trajectory following method

TABLE 9. *RMSE* BETWEEN THE ACTUAL AND DESIRED TRAJECTORY FOR THE SYSTEM ENCOUNTERING FOUR OBSTACLES

$RMSE_x$(m)	0.7
$RMSE_y$(m)	0.9

3) Comparison between Wall-Following and Path-Planning

Both the wall-following and RRT* path-planning methods have some advantages and disadvantages for collision avoidance, as mentioned below:

1. The wall-following method is based on sensor detection. Although the ultrasonic sensors have limitations as discussed earlier, the close-range detection of the obstacle's position can be relied on. This provides the advantage of online detection and avoidance of the obstacles. On the other hand, the RRT* path-planning method operates off-line, thus generating the path based on the existing previous knowledge of the obstacles in the underwater environment. Therefore, any obstacle that is unidentified before the operation would not be avoided. Similarly, moving obstacles cannot be detected, thus causing a serious collision risk.
2. The controllers were applied to the transportation system to follow the trajectory which was generated based on the waypoints obtained from the RRT* path planner. However, the response of the transportation system could deviate from the planned path, as the trajectory segments were independently created by the trajectory generator and the control inputs would have time lags and errors. Therefore, the trajectory-following method could result in an uncontrolled collision between the transportation system and an obstacle if the deviations from the path are substantial. On the other hand, the wall-following approach does not suffer from this issue, as the dynamics of the transportation system are directly controlled.
3. The time taken by the system to reach the goal location was higher for path-planning than the wall-following method. This is because the duration of each segment of the trajectory is selected to ensure sufficiently high tracking accuracy.
4. A comparison of the *RMS* of powers of the two methods is shown in TABLE 10. It can be seen that the power consumption for the wall-following method is higher than the path-planning approach. This is because, for the wall-following method, the forces applied by the thrusters to accelerate the transportation system towards the goal were abruptly increased or decreased significantly when the transportation system interacted with the obstacles due to the high controller gains. On the other hand, the thrust forces were smoothly applied during each segment with the path-planning method.

TABLE 10: *RMS* OF THE POWER CONSUMED WITH THE WALL-FOLLOWING AND PATH-PLANNING METHODS FOR THE TRANSPORTATION SYSTEM ENCOUNTERING FOUR OBSTACLES

	Wall-Following	Path-Planning
RMS_X(Watt)	1.64	1.35
RMS_Y(Watt)	5.77	3.25
Net (Watt)	7.41	4.6

V. CONCLUSIONS

In this paper, several collision avoidance methods were implemented on an underwater transportation system of four HAUVs to determine the most suitable approach. Firstly, sensor-based methods were analysed in detail for the case study of the transportation system encountering a single obstacle. In this study, the wall-following method presented higher time and fuel efficiency than the hard-switching method. Additionally, the hard-switching method caused higher yaw moments and required higher PID gains that resulted in an undesirable stronger system response and higher sensitivity to the gains selection. From this analysis, the wall-following method was selected to be compared against the offline RRT* path-planning scheme. This analysis showed that the former ensures obstacle avoidance, whereas the latter does not guarantee obstacle avoidance for unknown or moving objects due to its off-line planning nature. Additionally, the RRT* path-planning scheme combined with minimum-snap trajectory generation may cause an unrealistic trajectory that may not be followed accurately by the underwater transportation system, as the system's dynamics and physical limits are not considered in the trajectory generation step. Furthermore, the actual distance travelled and the time taken to reach the goal location was

higher for the path-planning than the wall-following method. However, the power consumption of the RRT* path-planning approach is lower than the wall-following solution.

One recommendation for future work would be to blend the offline RRT* path-planning and wall-following methods to obtain a complete strategy which would be able to avoid both stationary and moving obstacles. Moreover, a comparison will be required between the recommended strategy and online path-planning methods to decide on the best approach.

REFERENCES

[1] R. Wernli, "Low Cost UUV's for Military Applications: Is the Technology Ready?," San Diego, 2000.

[2] K. Alam, T. Ray, and S. G. Anavatti, "Design and construction of an autonomous underwater vehicle," Neurocomputing, vol. 142, pp. 16–29, 2014.

[3] N. M. Puzai, A. F. Ayob, and M. R. Arshad, "A Review on Recent Advancements in Unmanned Underwater Vehicle Design," J. Ocean. Mech. Aerosp. - Sci. Eng., vol. 31, pp. 1–8, 2016.

[4] W. H. Wang, X. Q. Chen, A. Marburg, J. G. Chase, and C. E. Hann, "Design of Low-Cost Unmanned Underwater Vehicle for Shallow Waters," in IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, 2008, pp. 204–209.

[5] C. D. Williams, "AUV systems research at the NRC-IOT: An update," Int. Symp. Underw. Technol., pp. 59–73, 2004.

[6] D. J. Stilwell and B. E. Bishop, "Platoons of Underwater Vehicles," IEEE Control Syst. Mag., pp. 45–52, 2000.

[7] X. Xiang, B. Jouvencel, and O. Parodi, "Coordinated formation control of multiple autonomous underwater vehicles for pipeline inspection," Int. J. Adv. Robot. Syst., vol. 7, no. 1, pp. 75–84, 2010.

[8] J. Ghommam, H. Mehrjerdi, M. Saad, and F. Mnif, "Formation path following control of unicycle-type mobile robots," Rob. Auton. Syst., vol. 58, no. 5, pp. 727–736, 2010.

[9] F. U. Rehman, G. Thomas, and E. Anderlini, "Centralized Control System Design for Underwater Transportation using two Hovering Autonomous Underwater Vehicles (HAUVs)," in IFAC-PapersOnLine, 2019, vol. 52, no. 11, pp. 13–18.

[10] K. Al-Mutib, F. Abdessemed, M. Faisal, H. Ramdane, M. Alsulaiman, and M. Bencherif, "Obstacle Avoidance Using Wall-Following Strategy for Indoor Mobile Robots," in 2nd IEEE International Symposium on Robotics and Manufacturing Automation (ROMA), 2016.

[11] Y. Liu and R. Bucknall, "Path planning algorithm for unmanned surface vehicle formations in a practical maritime environment," Ocean Eng., vol. 97, pp. 126–144, 2015.

[12] H. Lee, H. Kim, and H. J. Kim, "Planning and Control for Collision-Free Cooperative Aerial Transportation," IEEE Trans. Autom. Sci. Eng., vol. 15, no. 1, pp. 189–201, 2018.

[13] R. Song, Y. Liu, and R. Bucknall, "A multi-layered fast marching method for unmanned surface vehicle path planning in a time-variant maritime environment," Ocean Eng., vol. 129, pp. 301–317, 2017.

[14] Z. Shiller, "Off-Line and On-Line Trajectory Planning," in Motion and Operation Planning of Robotics Systems, G. Carbone and F. Gomez-Bravo, Eds. Springer, Cham, 2015, pp. 29–62.

[15] E. Lalish and K. A. Morgansen, "Decentralized reactive collision avoidance for multivehicle systems," in IEEE Conference on Decision and Control, 2008, pp. 1218–1224.

[16] J. L. Fernández, R. Sanz, J. A. Benayas, and A. R. Diéguez, "Improving collision avoidance for mobile robots in partially known environments: The beam curvature method," Rob. Auton. Syst., vol. 46, pp. 205–219, 2004.

[17] Y. Yoon, J. Shin, H. J. Kim, Y. Park, and S. Sastry, "Model-predictive active steering and obstacle avoidance for autonomous ground vehicles," Control Eng. Pract., vol. 17, no. 7, pp. 741–750, 2009.

[18] Q. Li and Z. P. Jiang, "Formation tracking control of unicycle teams with collision avoidance," in 47th IEEE Conference on Decision and Control, 2008, pp. 496–501.

[19] N. Isoda, K. Kogiso, and T. Asai, "Switching strategies of collision avoidance and tracking control for vehicles based on non-cooperative game and model predictive control," 22nd IEEE Int. Symp. Intell. Control, pp. 178–183, 2007.

[20] K. Kogiso, M. Nogochi, K. Hatada, N. Kida, N. Hirade, and K. Sugimoto, "Experimental Validation of Switching Strategy for Tracking Control with Collision Avoidance in Non-Cooperative Situation Using Toy Model Cars," SICE J. Control. Meas. Syst. Integr., vol. 3, no. 4, pp. 229–236, 2010.

[21] T. Weerakoon, "Artificial Potential Field and Feature Extraction Method for Mobile Robot Path Planning in Structured Environments," Kyushu Institute of Technology, 2016.

[22] R. Song, W. Liu, Y. Liu, and R. Bucknall, "A two-layered fast marching path planning algorithm for an unmanned surface vehicle operating in a dynamic environment," Ocean. - Genova, pp. 1–8, 2015.

[23] O. Grefstad and I. Schjolberg, "Navigation and collision avoidance of underwater vehicles using sonar data," in 2018 IEEE/OES Autonomous Underwater Vehicle Workshop, 2018, pp. 1–6.

[24] C. S. Tan, "A Collision Avoidance System for Autonomous Underwater Vehicles," University of Plymouth, 2006.

[25] F. U. Rehman, E. Anderlini, and G. Thomas, "The Impact of Sea Current on Underwater Transportation using Four AUVs," in International Conference on Autonomous Ships, 2020, pp. 49–58.

[26] J. S. Beggs, Kinematics. Taylors & Francis, 1983.

[27] T. I. Fossen, Handbook of Marine Craft Hydrodynamics and Motion Control, 1st Ed. Sussex: John Wiley & sons, 2011.

[28] S. M. Mo, "Development of a Simulation Platform for ROV systems," Marine Technology Master Thesis, Norwegian University of Science and Technology, 2015.

[29] F. U. Rehman, G. Thomas, and E. Anderlini, "Development of a Simulation Platform for Underwater Transportation using two Hovering Autonomous Underwater Vehicles (HAUVs)," in Proceedings of the 6th International Conference of Control, Dynamic Systems, and Robotics (CDSR'19), 2019, pp. 1–8.

[30] N. Farr, "Underwater acoustic/optical communications and data connectivity," Woods Hole Oceanographic Institution (WHOI). 2018.

[31] J. Lloret, S. Sendra, M. Ardid, and J. Rodrigues, "Underwater Wireless Sensor Communications in the 2.4 GHz ISM Frequency Band," Sensors (Basel), vol. 12, no. 4, pp. 4237–4264, 2012.

[32] M. Li, S. Guo, J. Guo, H. Hirata, and H. Ishihara, "Development of a biomimetic underwater microrobot for a father – son robot system," Microsyst. Technol., vol. 23, no. 4, pp. 849–861, 2017.

[33] M. Egerstedt, "the control of mobile robots," 2018. [Online]. Available: https://www.coursera.org/learn/mobile-robot.

[34] I. Noreen, A. Khan, and Z. Habib, "A Comparison of RRT, RRT* and RRT*-Smart Path Planning Algorithms," Int. J. Comput. Sci. Netw. Secur., vol. 16, no. 10, pp. 20–27, 2016.

[35] S. Karaman and E. Frazzoli, "Incremental sampling-based algorithms for optimal motion planning," Robot. Sci. Syst., vol. 6, pp. 267–274, 2011.

[36] S. LaValle, "Rapidly-Exploring Random Trees: A New Tool for Path Planning," 1998.

[37] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," Int. J. Rob. Res., vol. 30, no. 7, pp. 846–894, 2011.

[38] J. Nasir et al., "RRT*-SMART: A rapid convergence implementation of RRT*," Int. J. Adv. Robot. Syst., vol. 10, 2013.

[39] T. Elmokadem, M. Zribi, and K. Youcef-Toumi, "Terminal sliding mode control for the trajectory tracking of underactuated Autonomous Underwater Vehicles," Ocean Eng., vol. 129, pp. 613–625, 2017.

[40] D. H. Salunkhe, S. Sharma, S. A. Topno, C. Darapaneni, A. Kankane, and S. V. Shah, "Design, Trajectory Generation and Control of Quadrotor Research Platform," in International Conference on Robotics and Automation for Humanitarian Applications, 2016, pp. 1–7.

[41] D. Mellinger and V. Kumar, "Minimum Snap Trajectory Generation and Control for Quadrotors," pp. 2520–2525, 2011.

[42] D. Mellinger, "Trajectory Generation and Control for Quadrotors," Mechanical Engineering and Applied Mechanics PhD Thesis, University of Pennsylvania, 2012.

[43] E. Anderlini, G. G. Parker, and G. Thomas, "Control of a ROV carrying an object," Ocean Eng., vol. 165, no. March, pp. 307–318, 2018.