International Journal of Ocean System Engineering 2(3) (2012) 139-150
http://dx.doi.org/10.5574/IJOSE.2012.2.3.139

# An Improved Guidance Algorithm for Smooth Transition at Way-Points in 3D Space for Autonomous Underwater Vehicles

**Saravanakumar Subramanian[1*] and Asokan Thondiyath[1]**

[1] *Robotics Laboratory, Department of Engineering Design, Indian Institute of Technology Madras, Chennai 600036, India*

(Manuscript Received June 22 2012; Revised July 25, 2012; Accepted August 15, 2012)

## Abstract

This paper presents an improved guidance algorithm for autonomous underwater vehicles (AUV) in 3D space for generating smoother vehicle turn during the course change at the way-points. The way-point guidance by the line-of-sight (LOS) method has been modified for correcting the reference angles to achieve minimal calculation and smoother transition at the way-points. The algorithm has two phases in which the first phase brings the vehicle to converge to a distance threshold point on the line segment connecting the first two way-points and the next phase generates an angular path with smoother transition at the way-points. Then the desired angles are calculated from the reference and correction angles. The path points are regularly parameterized in the spherical coordinates and mapped to the Cartesian coordinates. The proposed algorithm is found to be simple and can be used for real time implementation. The details of the algorithm and simulation results are presented.

**Keywords:** Autonomous underwater vehicle (AUV), Guidance, Path planning, Line-of-sight, 3D space

## 1. Introduction

In recent years, the need of underwater robotic vehicles (URV) has increased rapidly due to the advances in oceanic technologies. There are many kinds of underwater robotic vehicles. Remotely operated vehicles (ROV) and Autonomous Underwater Vehicles (AUV) are widely used for applications such as underwater cable and pipeline monitoring, seabed and deep sea exploration, monitoring volcanoes, tsunami detection, and intervention tasks. ROV is operated by an operator from a remote location and controlled through a tether from a mother ship whereas the AUV is untethered and controlled by onboard computers which has in-built intelligent program to perform the mission. Between the two types, AUV requires lower operating cost, avoids operator fatigue, and reduces survey time and chances of cable entanglement. Due to these advantages, the demand for AUVs for underwater mission is increasing rapidly [1], [2]. The level of autonomy of an AUV depends on the guidance, navigation and control (GNC) systems. The efficiency of these three systems is important to an autonomous vehicle's mission success and utility. However any malfunction or inaccuracy in one of the systems decreases the efficiency of the other systems.

The main objective of the AUV guidance system is to receive target related information from the navigation system and generate the references for the vehicle control system so that the vehicle can move through a set of way-points as per the given sequence [3], [4]. Hence the guidance system has a major role in bringing autonomy to the AUV, and the effectiveness of the AUV is mainly limited by the precision and accuracy of the guidance schemes. In the guidance system, the ability to move the vehicle along a desired geometric path is very critical for many AUV applications. The path that will be

[*] Corresponding author. Tel.: +91-9381498414, Fax.: +91-44-22574732, E-mail address: srsaravanakumar@gmail.com

generated during maneuvering is usually specified in terms of the way-points using the Cartesian coordinates $(x,y,z) \in R^3$. Basically, the guidance problem is divided into two types as geometric task or time invariant path following and dynamic task or time variant trajectory tracking [5], [6]. Guidance system also includes sophisticated features like obstacle avoidance, minimum time navigation, fuel optimization and weather routing [3], [4]. Path following drives the vehicle to converge to and follow a desired spatial path without strict temporal specifications. Trajectory tracking is needed when the vehicle tracks a time parameterized reference path. These two concepts have been long standing guidance problems that are still under active research. Over the last two decades, only very few guidance algorithms have been proposed for the AUVs in 3D space [7]-[9]. Several guidance laws such as way-point guidance by LOS, vision based guidance, Lyapunav based guidance, guidance using chemical signals and magnetometers, proportional navigation guidance, and electromagnetic guidance are being used for developing the guidance strategies [5]. Way-point guidance by LOS is one of the most widely used methods for AUVs due to its simplicity and computational advantages. However, it has a major drawback of undesirable control energy consumption due to transients during course change at the way-points [5], [16], [17].

Smooth transition between two consecutive way-points can be obtained in 2D space by using straight lines and circular arcs [10]. In this method, if a sharp U-turn is required during a course change at way-points, the path generated will be far away from the way-point and hence misses it. Most of the algorithms proposed in 2D space for AUV guidance use only the current way-point for generating a smooth path. Also, they are complex as they use coordinate transformations and include dynamics of the vehicle [11]-[15]. A simple technique to avoid the missed way-point and the overshoot issues in planar case was proposed by Bakaric et al. [16]. This method generates a smooth way-point transition by considering the next way-point while calculating the heading correction for the current way-point. In this method, the vehicle calculates the heading correction far away from the current way-point, though it is not essentially needed. This increases the computation of desired heading for the vehicle to achieve a smooth path. Also the initial way-point and the vehicle's starting position are considered to be the same, which may not be true in real-time situations. To solve these drawbacks, an algorithm was proposed for planar case by fixing distance threshold points at both sides of the way-points in [18]. This method minimizes the computation of the heading correction but it is assumed that the AUV will move in the horizontal plane only, though the vehicle can move in 3D space in the real scenario.

In this paper we propose an improved guidance algorithm for path planning which incorporates the 3D motion capability of the AUV and generates a 3D path. A distance threshold point in 3D space is fixed in the direction of the way-points. The vehicle can calculate the course angles correction between the distance threshold points in the spherical coordinates and the position information are mapped to 3D Cartesian coordinates. Initially the vehicle is commanded to converge to the first distance threshold point if its starting position is different from initial way-point and then it is moved by generating a smoother transition at the way-points until it reaches the goal point. The smoother vehicle turn is achieved by calculating an auxiliary point at both sides of the way-points and finding the direction of turn. As the angular position correction is done at both sides of the way-points, a smoother turn in 3D path is achieved in between the distance threshold points and the way-points. Only minimal calculation is needed for correcting the two angles in the spherical coordinates which determines the smoother transition at the way-points.

The remaining part of the paper is organized as follows. In Section 2, the 3D path planning method is presented by improving the basic planar way-point guidance by the LOS method. Section 3 describes the simulation results for 3D path planning of an AUV using the improved algorithm and they are compared with the basic LOS method. Finally, the conclusions and future work are given in Section 4.

## 2. Improved Way-Point Guidance by Line-of-Sight Method

Way-point guidance by the line-of-sight (LOS) is intuitive and widely used in the application for the path following of underwater vehicles. It is also called as three point guidance scheme. In the basic

LOS algorithm, the reference angles are calculated only with respect to the current way-point, not with the next way-point. As the vehicle is always aiming the current way-point, it cannot turn smoothly once it approaches the current way-point. This causes sudden change of the course angle which affects the stern and rudder plane responses to move suddenly to the maximum position. This results in undesirable energy consumption due to the transients between two consecutive way-points [16], [17]. In order to achieve smoother turn at the way-points, the way-point guidance by the LOS is modified by making some corrections on the reference angles that are determined from the basic LOS guidance law. The LOS in 3D space can be defined by two angular variables as:

$$\psi_{ri} = arctan\left(\frac{y_{i+1} - y_v}{x_{i+1} - x_v}\right) \quad (1)$$

$$\theta_{ri} = arctan\left(\frac{z_{i+1} - z_v}{\sqrt{(x_{i+1} - x_v)^2 + (y_{i+1} - y_v)^2}}\right), \quad (2)$$

where $(x_{i,}y_i, z_i\,(i:1..n))$ are the coordinates of the given set of way-points. $\psi_r$ and $\theta_r$ are denoted as the vehicle's heading (azimuth) and path (elevation) angles. These angles are computed by using the four quadrant tangent function, *atan2*. When the vehicle lies within the sphere of acceptance with a radius, $\rho_o$ around the way-point, i.e. if the vehicle location $(x_v, y_v, z_v)$ satisfies eq. (3), then the next way-point can be selected.

$$dP_{Ci} = \sqrt{\left((x_i - x_v)^2 + (y_i - y_v)^2 + (z_i - z_v)^2\right)} \le \rho_0 \quad (3)$$

Here $dP_{Ci}$ is the distance between the vehicle position and the current way-point. The sphere of acceptance $(\rho_0)$ is taken as 2L, where L is the length of the vehicle.

The various angles used for calculating the correction angles are shown in Fig. 1. The complete algorithm is divided into two phases. In the first phase, distance threshold points are fixed on an imaginary straight line segment that is connecting two consecutive way-points. Then the vehicle's path from its initial (arbitrary) position is brought to coincide with the line segment connecting the first two way-points before reaching the distance threshold point in this line. In the second phase, auxiliary points are fixed on the imaginary straight line segment and a path from the current position of the vehicle to the current way-point is generated with smoother turn at the way-points.

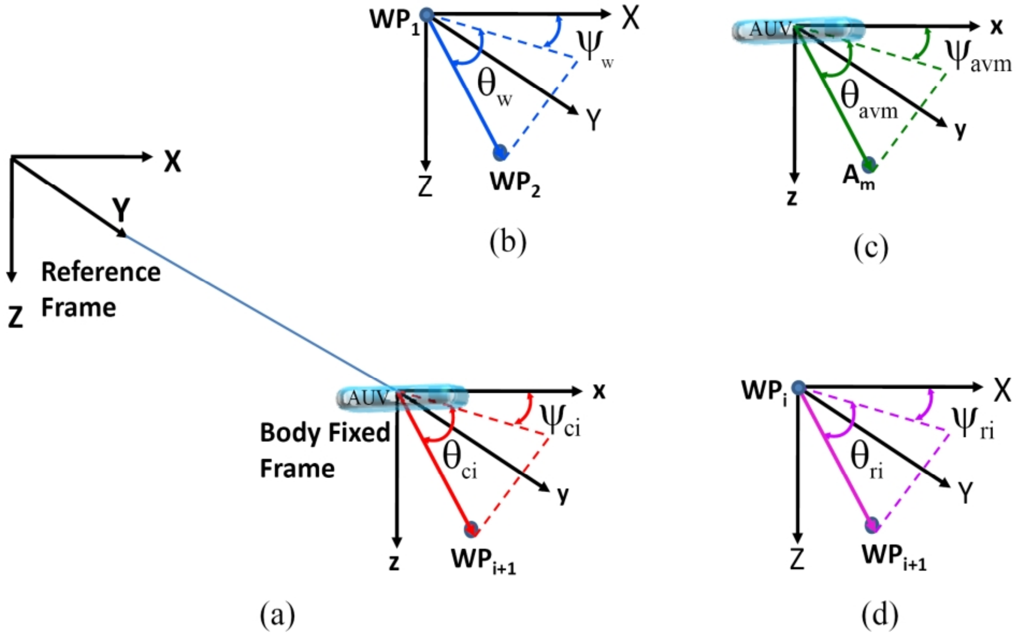

Fig. 1: Calculation of different angles

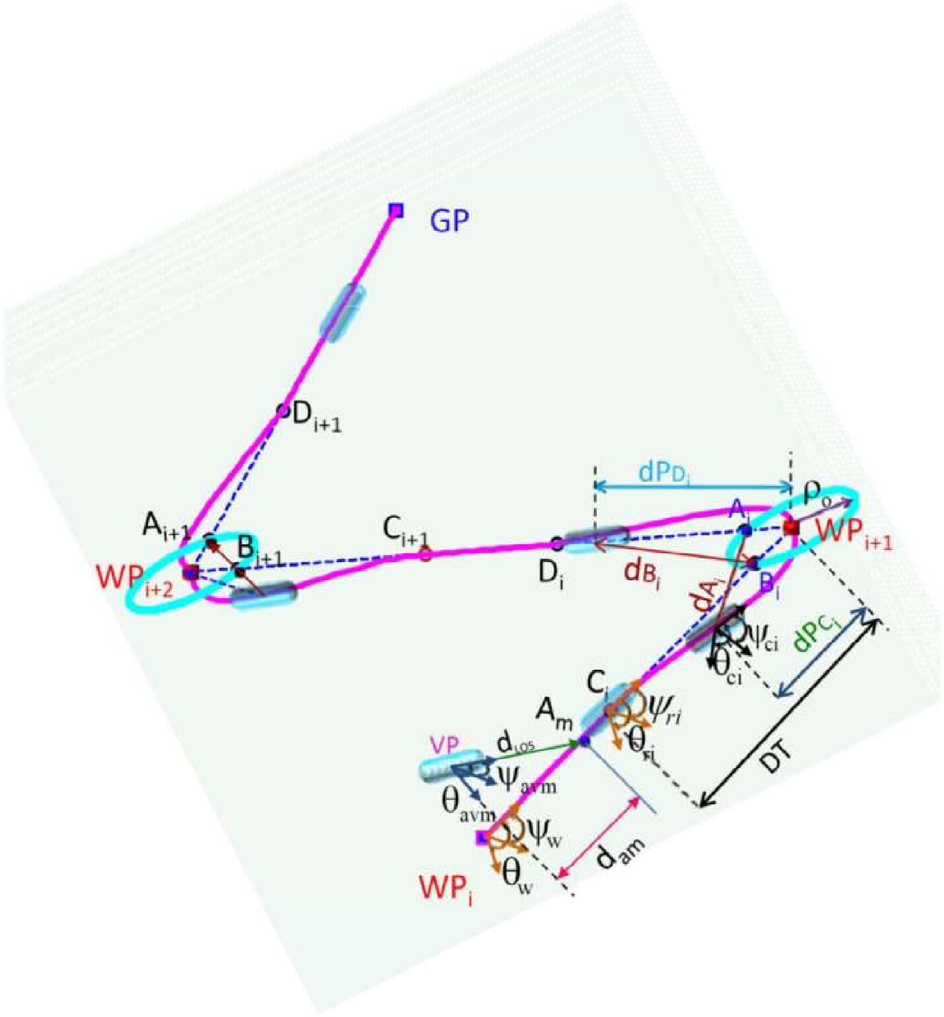

Fig. 2: Improved LOS guidance for smooth way-point transition in 3D space.

The methodology for the improved LOS guidance in 3D space is shown in Fig. 2 and it can be explained as follows. Let $VP(x_v, y_v, z_v)$ is the current position of the vehicle and $GP(x_g, y_g, z_g)$ is the goal point and $WP_i(x_i, y_i, z_i\ (i:1..n))$ is the given set of way-points. The initial position of the vehicle could be at any location in the vicinity of the way-points. It is assumed that these data are available in the navigation system and the guidance system receives these data at regular interval. The proposed algorithm initially receives the first three way-points (n=3) along with $VP$ and $GP$. Let $WP_i$, $WP_{i+1}$ and $WP_{i+2}$ are the previous, current and next way-points respectively. The successive way-point after the next way-point is updated when the vehicle reaches the current way-point. A distance threshold point $C_i\,(x_i, y_i, z_i)$ is located on the line segment connecting the previous and current way-points. Similarly another distance threshold point $D_i\,(x_i, y_i, z_i)$ is located on the line segment connecting the current and next way-points. To locate the threshold points, a threshold distance (DT) is defined and is taken as 10 times the length of the vehicle, L. Though this distance is an adjustable constant, 10L is sufficient even if any sharp U-turn is needed. The distance threshold points can then be defined as

$$\begin{aligned} Cx_i &= x_{i+1} - DT\cos(\theta_{ri})\cos(\psi_{ri}) \\ Cy_i &= y_{i+1} - DT\cos(\theta_{ri})\sin(\psi_{ri}) \\ Cz_i &= z_{i+1} - DT\sin(\theta_{ri}) \end{aligned} \tag{4}$$

$$\begin{aligned} Dx_i &= x_{i+1} + DT\cos(\theta_{ri})\cos(\psi_{ri}) \\ Dy_i &= y_{i+1} + DT\cos(\theta_{ri})\sin(\psi_{ri}), \\ Dz_i &= z_{i+1} + DT\sin(\theta_{ri}) \end{aligned} \tag{5}$$

where $(Cx_i, Cy_i, Cz_i)$ and $(Dx_i, Dy_i, Dz_i)$ are the coordinates of the distance threshold points $C_i$ and $D_i$. In the first phase, a LOS vector $(d_{LOS})$ is created from the vehicle's initial position to a point ("aiming point, A$_m$") on the current way-point line segment using along track distance $(d_r)$ and cross track distances $(d_{c1}, d_{c2})$. The aiming point distance, $d_{am}$ is calculated by adding 'k' meters to the along track distance. The initial way-point $WP_1$ is used to generate the LOS path for the vehicle to move, as if the vehicle was at $WP_1$ initially. In this way, the need to specify the initial position of the vehicle as $WP_1$ is eliminated. Even if the current vehicle position is unknown to the path planner, the guidance algorithm can bring the vehicle to the desired path. The LOS vector defines the desired angles in the spherical coordinates by calculating the along track distance. Then the position of the vehicle is calculated using homogeneous transformation in the spherical coordinates [8]. The aiming point distance and the sphere of acceptance $(\rho_0)$ are taken as 3L and 2L respectively. This distance and sphere of acceptance can be varied in order to generate different path. The desired azimuth and elevation angles between the first two way-points can be given as:

$$\psi_w = arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{6}$$

$$\theta_w = arctan\left(\frac{z_2 - z_1}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}\right), \tag{7}$$

where $(\theta_w)$ varies from +35° to -35°. The along track distance and the cross track distances can be obtained as:

$$\begin{bmatrix} d_r \\ d_{c1} \\ d_{c2} \\ 1 \end{bmatrix} = \begin{bmatrix} cos(\theta_w)cos(\psi_w) & -sin(\theta_w)cos(\psi_w) & -cos(\theta_w)sin(\psi_w) & x_{wp1} \\ cos(\theta_w)sin(\psi_w) & -sin(\theta_w)sin(\psi_w) & cos(\theta_w)cos(\psi_w) & y_{wp1} \\ sin(\theta_w) & cos(\theta_w) & 0 & z_{wp1} \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x_{vm} \\ y_{vm} \\ z_{vm} \\ 1 \end{bmatrix} = (M_{wp})^{-1} \begin{bmatrix} x_{vm} \\ y_{vm} \\ z_{vm} \\ 1 \end{bmatrix} \tag{8}$$

where $[x_{wp1}, y_{wp1}, z_{wp1}]$ denotes the coordinates of $WP_1$, $(x_{vm}, y_{vm}, z_{vm} (m:1..n)$ represents the vehicle's current position until it reaches the first distance threshold point and $M_{wp}$ denotes the homogeneous transformation matrix in the spherical coordinates. The inverse homogeneous transformation matrix is used to map $WP_1$ from global frame to local frame. Then the aiming point, A$_m$ $(x_{avm}, y_{avm}, z_{avm})$ is calculated by adding k meters (k=3L) to the along track distance and transforming the local coordinates to the global coordinates as:

$$\begin{bmatrix} x_{avm} \\ y_{avm} \\ z_{avm} \\ 1 \end{bmatrix} = (M_{wp}) \begin{bmatrix} d_r + k \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{9}$$

The desired heading and path angles are obtained as:

$$\psi_{avm} = atan2\left(\frac{y_{avm} - y_{vm}}{x_{avm} - x_{vm}}\right) \tag{10}$$

$$\theta_{avm} = atan2\left(\frac{z_{avm} - z_{vm}}{\sqrt{(x_{avm} - x_{vm})^2 + (y_{avm} - y_{vm})^2}}\right) \tag{11}$$

In the next phase, the vehicle is moving towards the current way-point from the first distance threshold point $C_1$ by considering the next way-point in order to achieve smoother transition at the way-points. In this method, the vehicle moves in an angular path between $C_1$ and $GP$. The desired angles are calculated by adding some correction values with the reference angles and a smooth turn is achieved at the way-points.

In order to achieve this, two auxiliary points $A(x_{Ai}, y_{Ai}, z_{Ai})$ and $B(x_{Bi}, y_{Bi}, z_{Bi})$ are defined at a radial distance $\rho_0$ from the current way-point on the LOS as shown in Fig. 2. The coordinates of the auxiliary point $A_i$ is computed from the planar equations described by Bakaric et al. [16] as

$$x_{Ai} = x_{i+1} + \rho_o \frac{sgn(Dx_i - x_{i+1})}{\sqrt{1 + \left( \frac{(Dy_i - y_{i+1})^2 + (Dz_i - z_i)^2}{(Dx_i - x_i)^2} \right)}}$$
$$y_{Ai} = y_{i+1} + \rho_o \frac{sgn(Dy_i - y_{i+1})}{\sqrt{1 + \left( \frac{(Dx_i - x_i)^2 + (Dz_i - z_i)^2}{(Dy_i - y_i)^2} \right)}} \qquad (12)$$
$$z_{Ai} = z_{i+1} + \rho_o \frac{sgn(Dz_i - z_{i+1})}{\sqrt{1 + \left( \frac{(Dy_i - y_{i+1})^2 + (Dx_i - x_i)^2}{(Dz_i - z_{i+1})^2} \right)}}$$

Similarly, the coordinates of the auxiliary point $B_i$ can be given as:

$$x_{Bi} = x_{i+1} + \rho_o \frac{sgn(Cx_i - x_{i+1})}{\sqrt{1 + \left( \frac{(Cy_i - y_{i+1})^2 + (Cz_i - z_{i+1})^2}{(Cx_i - x_{i+1})^2} \right)}}$$
$$y_{Bi} = y_{i+1} + \rho_o \frac{sgn(Cy_i - y_{i+1})}{\sqrt{1 + \left( \frac{(Cx_i - x_{i+1})^2 + (Cz_i - z_{i+1})^2}{(Cy - y_{i+1})^2} \right)}} \qquad (13)$$
$$z_{Bi} = z_{i+1} + \rho_o \frac{sgn(Cz_i - z_{i+1})}{\sqrt{1 + \left( \frac{(Cy_i - y_{i+1})^2 + (Cx_i - x_{i+1})^2}{(Cz_i - z_{i+1})^2} \right)}}$$

The distances between the vehicle position and the auxiliary points $A_i, B_i$ can be calculated as,

$$d_{Ai} = \sqrt{(x_{Ai} - x_v)^2 + (y_{Ai} - y_v)^2 + (z_{Ai} - z_v)^2}$$
$$d_{Bi} = \sqrt{(x_{Bi} - x_v)^2 + (y_{Bi} - y_v)^2 + (z_{Bi} - z_v)^2} \qquad (14)$$

Similarly, the distance '$dP_{Di}$'from the vehicle to the current way-point when the vehicle is in the distance threshold region can be calculated using eq. 3. The normalized difference between the auxiliary and the current way-point distance from the vehicle can be calculated as

$$\varepsilon_{Ai} = \frac{d_{Ai} - dP_{C_i}}{\rho_o}$$
$$\varepsilon_{Bi} = \frac{d_{Bi} - dP_{D_i}}{\rho_o} \qquad (15)$$

If the next way-point lies in the direction of the current way-point, then the normalized difference factor becomes 1, and no angle correction is required. Hence the vehicle follows the reference angles. Depending on the actual location of the next way-point, this factor assumes a value between -1 to 1. Depending on the position of the next way-point, vehicle needs to calculate the correction angles $\psi_{ci}$ and $\theta_{ci}$. The sign of the correction angles need to be calculated to determine the turning direction of the vehicle. This sign depends on the spherical coordinate angles between the previous and current way-points and the current and next way-points. The sign of the correction angles $\psi_{sci}$ and $\theta_{sci}$ between $C_i$ and $WP_i$ can be determined as

$$\psi_{sci} = sgn(\psi_{ri} - \psi_{ri+1})$$
$$\theta_{sci} = sgn(\theta_{ri} - \theta_{ri+1}) \qquad (16)$$

Similarly, the sign of the angles correction between $WP_i$ and $D_i$ can be determined as

$$\psi_{sci} = sgn(\psi_{ri+1} - \psi_{ri})$$
$$\theta_{sci} = sgn(\theta_{ri+1} - \theta_{ri}) \qquad (17)$$

In order to have a smooth variation in correction angles, it is necessary to gradually change the angles from zero at the distance threshold point to a maximum value at a predefined distance and then bring it to zero at the way-point.

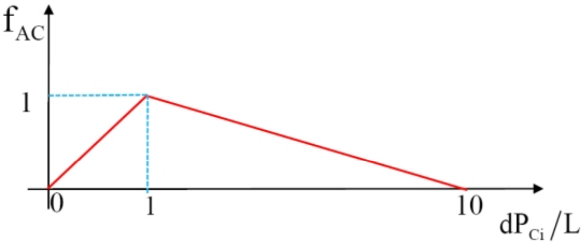

Fig. 3: Distance factor for the auxiliary point Ai

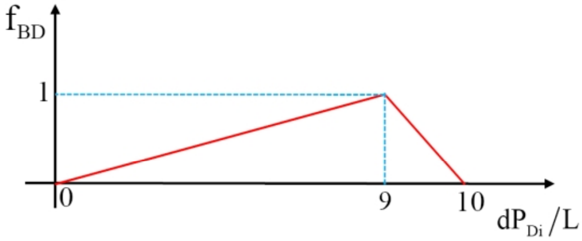

Fig. 4: Distance factor for the auxiliary point B

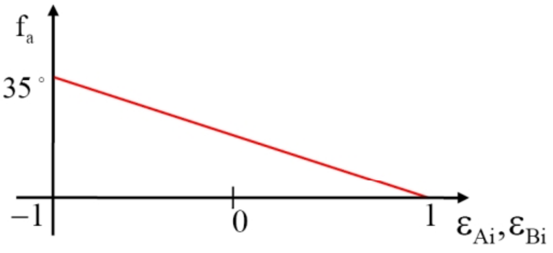

Fig. 5: Angle factors for calculating angles correction.

This gradual change is implemented through two variables i.e. distance factor and angle factor. The distance factors $f_{AC}(dP_{Ci})$ and $f_{BD}(dP_{Di})$ represent the gradual change of the angles with respect to the distances $dP_{Ci}$ and $dP_{Di}$ respectively. The angle factors $f_a(\varepsilon_{Ai})$ and $f_a(\varepsilon_{Bi})$ represent the angle change considering the maximum turning limit of the vehicle with respect to $\varepsilon_{Ai}$ and $\varepsilon_{Bi}$ respectively. These factors are obtained using the linear curve fitting as explained using Figs. 3 to 5 [18].

The angles correction $(\psi_{ci})$ and $(\theta_{ci})$ between $C_i$ and $WP_{i+1}$ can be calculated as

$$\begin{aligned}\psi_{ci} &= (\psi_{sc}).f_{AC}(dP_{Ci}).f_a(\varepsilon_{A})\\ \theta_{ci} &= (\theta_{sc}).f_{AC}(dP_{Ci}).f_a(\varepsilon_{A})\end{aligned} \tag{18}$$

Similarly, the angles correction between $WP_{i+1}$ and $D_i$ can be calculated as

$$\begin{aligned}\psi_{ci} &= (\psi_{sc}).f_{BD}(dP_{Di}).f_a(\varepsilon_{B})\\ \theta_{ci} &= (\theta_{sc}).f_{BD}(dP_{Di}).f_a(\varepsilon_{B})\end{aligned} \tag{19}$$

Finally the desired angles $(\psi_{di}\ and\ \theta_{di})$ are calculated as [16]

$$\begin{aligned}\psi_{di} &= \psi_{ri} + \psi_{ci}\\ \theta_{di} &= \theta_{ri} + \theta_{ci}\end{aligned} \tag{20}$$

The mapping from spherical coordinates to three dimensional Cartesian coordinates is given by

$$\begin{aligned}x_{v+1} &= x_v + r\cos(\theta_{di})\cos(\psi_{di})\\ y_{v+1} &= y_v + r\cos(\theta_{di})\sin(\psi_{di}),\\ z_{v+1} &= z_v + r\sin(\theta_{di})\end{aligned} \tag{21}$$

where $r$ is the radial distance used to parameterize the distance between two points in 3D space. The path points are generated in the Cartesian coordinates and the vehicle is commanded by using these points.

Here $(x_{v,}y_v, z_v)$ is the current position of the vehicle and $(x_{v+1,}y_{v+1}, z_{v+1})$ is the next position of the vehicle. The correction angles are calculated between $C_i$ and the current way-point as well as the current way-point and $D_i$. Thus the algorithm provides a smooth path in 3D space on both sides of the current way-point. It is a simple and elegant algorithm without complex coordinate transformations and takes into account the initial position, the vehicle size and the speed in calculating the path. The number of calculations required for finding the correction heading and path angles are reduced by introducing the distance threshold points. By defining the distance factors and angle factors in a better way and fixing the auxiliary points at both sides of the way-points, the smoothness of the angular path is improved. The flow chart in Fig. 6 shows the various steps involved in generating the smooth path for the vehicle using the modified algorithm.

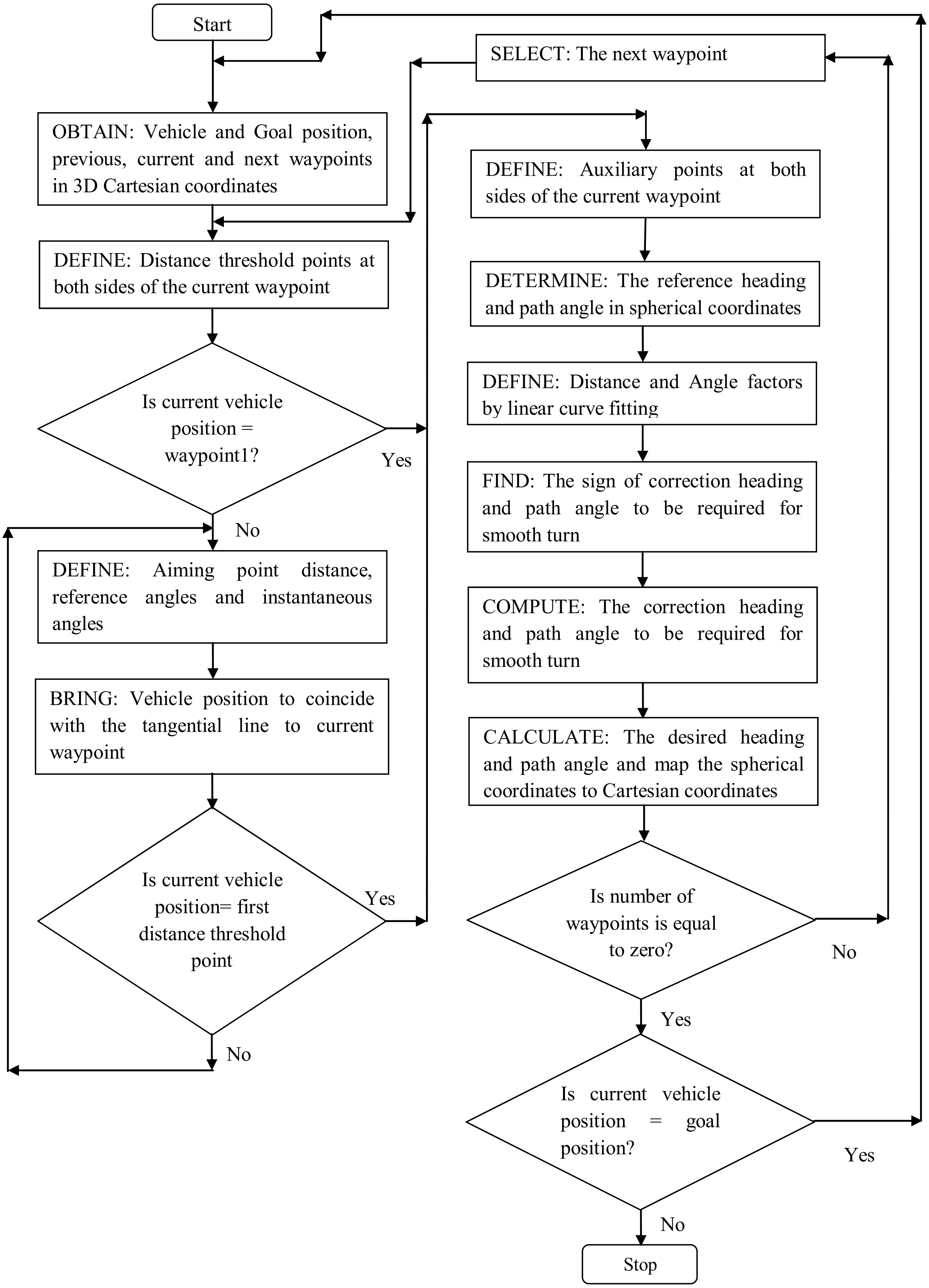
Start
OBTAIN: Vehicle and Goal position, previous, current and next waypoints in 3D Cartesian coordinates
DEFINE: Distance threshold points at both sides of the current waypoint
Is current vehicle position = waypoint1?
Yes
No
DEFINE: Aiming point distance, reference angles and instantaneous angles
BRING: Vehicle position to coincide with the tangential line to current waypoint
Is current vehicle position= first distance threshold point
Yes
No
SELECT: The next waypoint
DEFINE: Auxiliary points at both sides of the current waypoint
DETERMINE: The reference heading and path angle in spherical coordinates
DEFINE: Distance and Angle factors by linear curve fitting
FIND: The sign of correction heading and path angle to be required for smooth turn
COMPUTE: The correction heading and path angle to be required for smooth turn
CALCULATE: The desired heading and path angle and map the spherical coordinates to Cartesian coordinates
Is number of waypoints is equal to zero?
No
Yes
Is current vehicle position = goal position?
Yes
No
Stop

Fig. 6: Flowchart for the improved guidance algorithm

## 3. Simulation Results

To verify the performance of the proposed guidance algorithm, the numerical simulations are performed. The following assumptions are made for the simulation:

- The vehicle is considered as a flat-fish shape AUV of length, L=4.5m
- The forward speed of the vehicle is fixed to a constant value, $u_d = 2$m/s
- The geometric path is regularly parameterized with a radial distance (r) of 2m
- An angular rate of $1.6^o$/s is used for calculating the maximum heading and path angles change in order to determine the angle factors
- The desired path between two consecutive way-points and the time of arrival at the way-point are not defined
- The vehicle cannot move sideways
- The aiming point distance and sphere of acceptance are taken as 13.5m and 9m respectively. These values can be adjusted for getting different path profile

Simulations are carried out for four different cases.

- Case-1: Vehicle Position - (10,0,2), way-points-(0,0,0), (75,75,0) and goal point-(150,0,0)
- Case-2: Vehicle Position - (10,0,2), way-points-(0,0,0),(75,0,75), and goal point-(150,0,0)
- Case-3: Vehicle Position - (20,50,0), way-points-(0,0,0), (75,75,75), (150,150,150), (250,250,250),(300,300,300), and goal point-(500,500,500)
- Case-4: Vehicle Position - (150,-20,0), way-points-(150,-20,0), (0,0,50), (350,0,50), (200,50,50), (350,100,50), (0,100,50), and goal point-(150,120,0)

In Case-1, the way-points are considered in such a way that the vehicle can move only in the horizontal plane after reaching the first distance threshold point, $C_1$. Hence the z-coordinate is taken as zero for the way-points and the goal point. The vehicle's initial position is randomly located, different from the $WP_1$ in the 3D space, in order to check whether the vehicle converges to $C_1$. The results of the simulation are shown in Figs. 7 to 10. Fig. 7 shows that the algorithm generates a smooth path in both horizontal and vertical planes until the current position of the vehicle converges to $C_1$. Figs. 8 to 10 show the path generated in the three planes as the vehicle move from the initial position to the target position through the way-point.

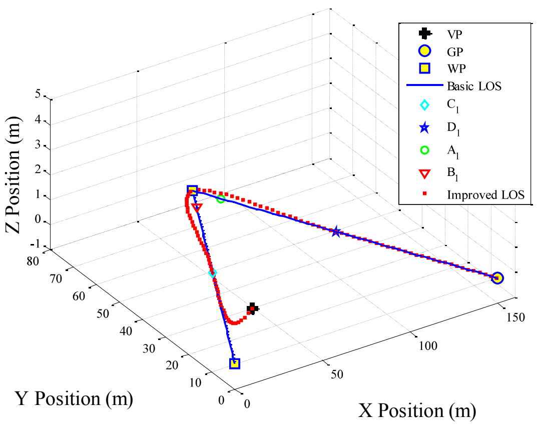

Fig. 7: 3D path generated by improved guidance algorithm for Case-1

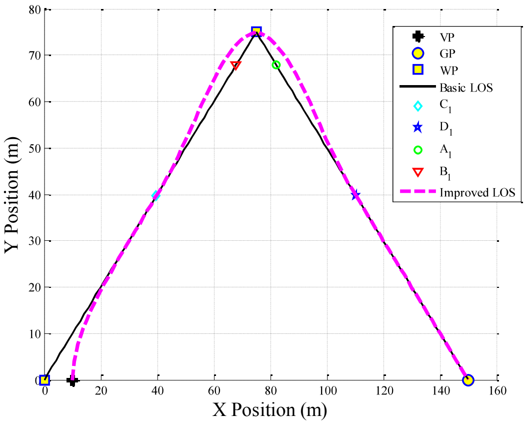

Fig. 8: Path generated in XY plane by improved guidance algorithm for Case-1

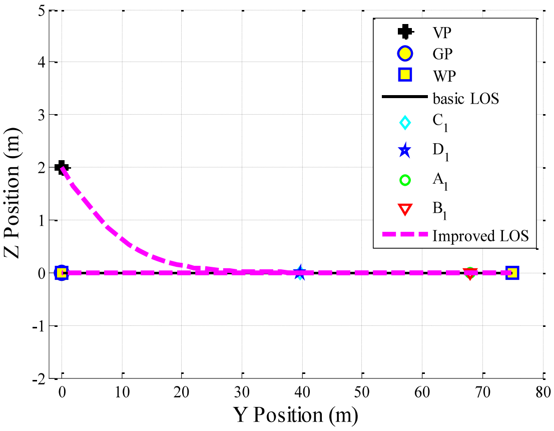

Fig. 9: Path generated in YZ plane by improved guidance algorithm for Case-1

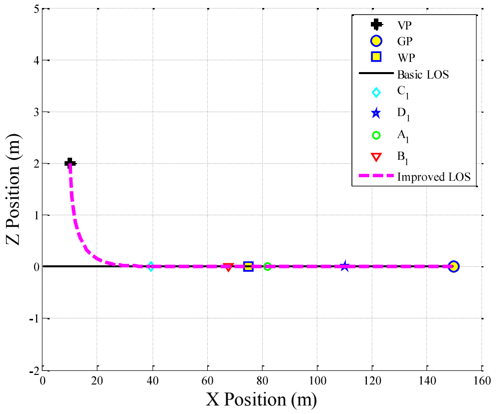

Fig. 10: Path generated in XZ plane by improved guidance algorithm for Case-1

In order to illustrate the smooth transition in the vertical plane, y and z coordinates of the way-points are interchanged in Case 2, while keeping the same initial and goal positions as in the previous case. The simulation results are shown in Fig. 11. From the above simulation results, it can be seen that the algorithm is capable of finding a smooth path between the way-points in 3D space, irrespective of the initial position.

In order to analyze the performance of the algorithm in other situations, multiple way-points are considered in Case-3. They are located in the same direction from the initial way-point to the goal point. As in the previous cases, the initial position of the vehicle is located away from $WP_1$. The simulation results are shown in Fig. 12. It can be seen that the correction angles near the way-points are zero since the way-points lie on a straight line path. Hence the desired angles are equal to the reference angles and the vehicle is moved in a straight line until it reaches the goal point.

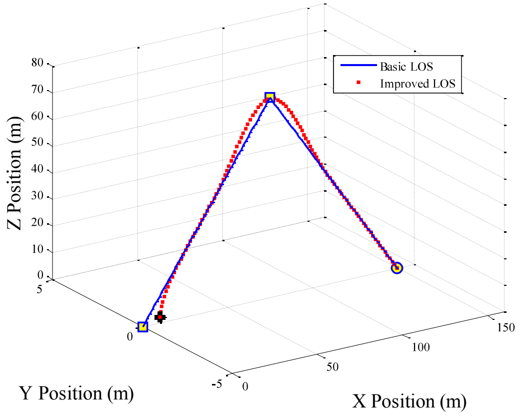

Fig. 11: 3D path generated by improved guidance algorithm for Case-2

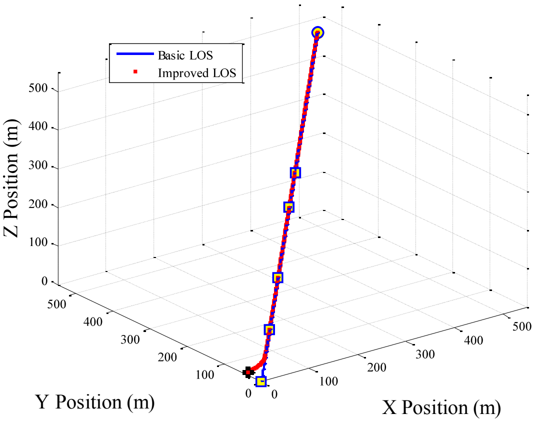

Fig. 12: 3D Path generated in XYZ plane by improved guidance algorithm for Case-3

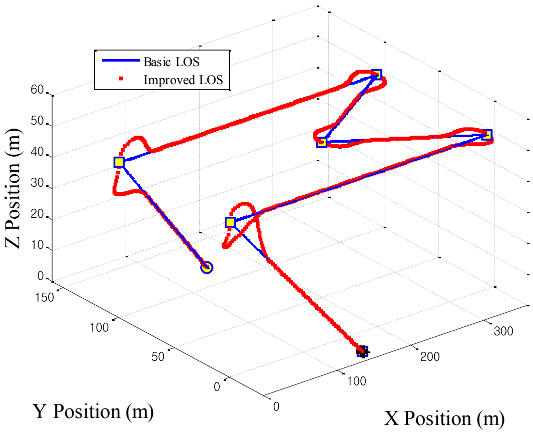

Fig. 13: 3D Path generated in XYZ plane by improved guidance algorithm for Case-4

Multiple way-points at random locations are considered in Case-4. The way-points are located in such a way that the vehicle needs to move in 3D path continuously to reach the goal point. The 3D path generated for this case is shown in Fig. 13. The result shows that the algorithm is capable of generating a smooth path from the initial position to the final position through the way-points. The above simulation results confirm that the modified algorithm is capable of generating paths that avoid sudden course changes at the way-points. It can also be observed that the vehicle computes the orientation correction only between the distance threshold points and the way-points. This actually reduces the computational load as well as the distance travelled.

The minimal calculation required for generating a smoother transition at the way-points by the proposed method can be checked by comparing with the method proposed by Bakaric et al. [16]. The number of iterations performed for calculating the heading correction in the forward path in a planar case is considered in both methods. The starting point (0,0), the way-point (50,50) and the goal point (100,00) are given as inputs to both methods and the paths generated by the methods are shown in Fig. 14. If the distance threshold point is not fixed in the forward path, the heading correction is calculated far away from the current way-point and it requires 40 iterations to reach the way-point. In the case of the proposed algorithm, the heading correction requires only 29 iterations as the correction is done between the distance threshold points and the way-points. If multiple way-points are taken and the heading correction for both sides of the way-points is needed to be calculated, then the number of iterations is greatly reduced by using the proposed algorithm. It can also be seen that the path generated by our proposed method is much smoother than the other methods as shown in Fig. 14.

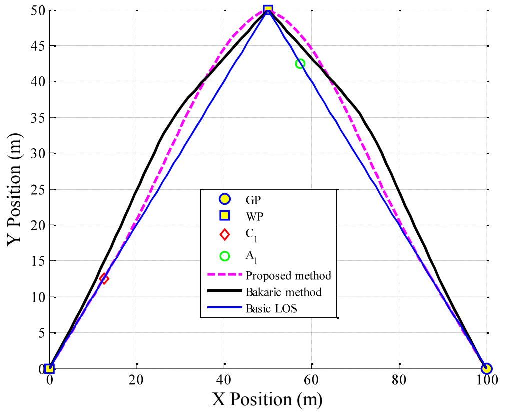

Fig. 14: Checking minimal calculation of angle correction

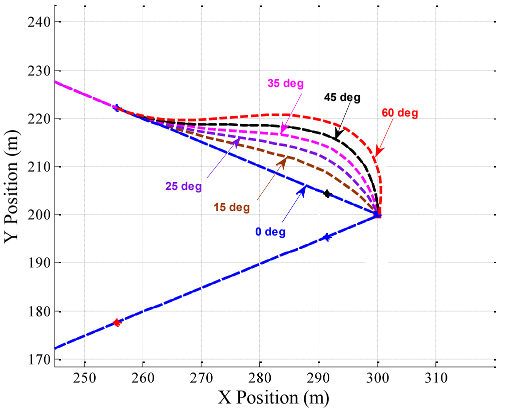

Fig. 15: Determining the ideal value for turning angle to calculate the angle factor

The maximum turning angle required for the distance of 10L with the given velocity and the heading rate is theoretically calculated as 36$^o$. Hence an ideal value of 35$^o$ is chosen for determining the angle factors. In order to ensure the smooth turn at the way-points, the algorithm is simulated for the forward path turn with different turning angles as shown in Fig. 15. It has been observed that the turning angle more than 35$^o$ degree generates more turn and not gradually changes near the distance threshold points. The turning angle less than the ideal value produces a minimum turn which is not desirable near the way-points. Hence a typical value of 35$^o$ is chosen for the smooth turn. The same value is taken for the reversed path also.

## 4. Conclusions and Future Work

A simple algorithm for time invariant path planning for the AUV is developed in 3D space. The simulation results show that the algorithm generates a smooth path between two consecutive way-points irrespective of the initial position of the vehicle. Also, the transient issues at the way-points are eliminated. It is also shown that even if a sharp U-turn is needed, it is possible to achieve a better and smooth transition. Since the vehicle starts computing the angle correction between two distance threshold points and the way-points, only minimal computation is required in order to get smoother turn at the way-points. This algorithm can be used in on-line as well as off-line mode of operation. In on-line mode, it can be used in outer loop, so that the output from this algorithm can be used for inner loop to generate a trajectory by the trajectory tracking applications. In the off-line mode, this algorithm can be used to eliminate the problem of changing the way-points by the mission planner in order to avoid a sharp U-turn. Even if the mission planner fixes some arbitrary way-points, the vehicle can go through the way-points. In both modes, the algorithm can be integrated with other guidance schemes such as obstacle avoidance. In order to implement the developed 3D way-point guidance algorithm for the real-time applications, hardware in the loop (HIL) simulations will be carried out to validate the algorithm. The results will be presented in the near future.

## References

[1] F. Repoulias, and E. Papadopoulos, *Planar Trajectory Planning and Tracking Control Design for Underactuated AUVs*, J. of Ocean Eng., 34 (1990) 1650-1667.

[2] T. W. Kim, and J. Yuh, *Development of a Real-Time Control Architecture for a Semi-Autonomous Underwater Vehicle for Intervention Missions*, J. of Control Engineering Practice, 12 (2004) 1521-1530.

[3] T. I. Fossen, *Guidance and Control of Ocean Vehicle*, John Wiley and Sons Ltd., Chichester, England (1994).

[4] T. I. Fossen, *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*, Marine Cybernetics AS, Trondheim, Norway (2002).

[5] W. Naeem, R. Sutton, S. M Ahmad, and R. S. Burns, *A Review of Guidance Laws Applicable to Unmanned Underwater Vehicles*, J. of Navigation, 56 (1) (2003) 15-29.

[6] R. Skjetne, *The Manoeuvering Problem, Ph.D. Thesis, Norwegian University of Science and Technology*, Trondheim, Norway (2005).

[7] M. Breivik, and T. I. Fossen, *Principles of Guidance-Based Path Following in 2D and 3D. 44th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC'05)*, Seville, Spain, (2005) 627-634.

[8] O. Engelhardsten, *3D AUV Collision Avoidance, M.S. Thesis, Norwegian University of Science and Technology*, Trondheim, Norway (2005).

[9] P. Encarnacao, and A. Pascoal, *3D Path Following for Autonomous Underwater Vehicle. IEEE Conference on Decision and Control, Sydney*, NSW, Australia, (2000) 2977-2982.

[10]T. I. Fossen, M. Breivik, and R. Skjetne, *Line-Of-Sight Path Following of Underactuated Marine Craft*, IFAC Manoeuvering and Control of Marine Craft (MCMC'03), Girona, Spain, (2003).

[11]X. Xiang, L. Lapierre, B. Jouvencel, and O. Parodi, *Coordinated Path Following Control of Multiple Nonholonomic Vehicles*, IEEE Oceans 09 (OCEANS'09), Bremen, (2009) 1-7.

[12]S. R. Oh, and J. Sun, *Path following of underactuated marine surface vessels using line of sight based model predictive control*, J. of Ocean Engineering, 37(2-3) 289-295.

[13]M. Aicardi, G. Casaline, G. Indiveri, A. Aguiar, P. Encarnacao, and A. Pascoal, *A Planar Path Following Controller for Underactuated Marine Vehicles*, IEEE Mediterranean Conference on Control and Automation (MED '01), Dubrovnik, Croatia, (2001).

[14]R. Skjetene, U. Jorgensen, and A. R. Teel, *Line of sight path following along regularly parameterised curves solved as a generic manoeuvering problem*, IEEE Conference on Decision and Control and European Control Conference (CDC-ECC'11), Orlando, USA, (2011) 2467-2474.

[15]L. Moreira, T.I. Fossen, and G. Guedes Soares, *Path Following Control System for a Tanker Ship Model*, J. of. Ocean Engineering, 34 (2007) 2074–2085.

[16]V. Bakaric, Z. Vukic, and R. Antonic, *Improved Basic Planar Algorithm of Vehicle Guidance through Way-Points by the Line-Of-Sight*, IEEE 1st International Symposium on Control, Communications and Signal Processing, (2004) 541-544.

[17]D. J. Yeo, *Design of AUV Tracking System Using the Sliding Mode Control and The Optimal Control Theory*, M.S. Thesis, Department of Naval Architecture and Ocean Engineering, Seoul National University, Korea, (1999).

[18]S. Saravanakumar, and T. Asokan, *Way-Point Guidance-based Planar Path Following and Obstacle Avoidance of Autonomous Underwater Vehicle*, 8th International Conference on Informatics in control, Automation and Robotics (ICINCO '11), Noordwijkerhout, The Netherlands, (2011) 191-198.