中國機械工程學刊第四十四卷第六期第 573~581 頁(民國一百一十二年)
Journal of the Chinese Society of Mechanical Engineers, Vol.44, No.6, pp573~581 (2023)

An Innovative Approach to Nonlinear Guidance Law for Path Following of Unmanned Surface Vehicle

Osman Ünal***, Nuri Akkaş* and Gökhan Atalı**

Keywords：path following, nonlinear guidance law, simulated annealing.

ABSTRACT

This study proposes an innovative approach to nonlinear guidance law based path following algorithm. To the best of our knowledge, we are the first to integrate the simulated annealing optimization technique with nonlinear guidance law for path following of USVs (Unmanned Surface Vehicles). The purpose of this paper is to minimize cross tracking errors using this novel method. In contrast to previous studies related to nonlinear guidance law, the presented new algorithm uses circles with variable radius. The radius of adjustable circles is optimized using simulated annealing method in each time step. This study concludes that cross tracking errors are significantly reduced owing to optimized radius compared to conventional nonlinear guidance law.

INTRODUCTION

In recent years, unmanned surface vehicles have been widely used in military and commercial fields to perform crucial and dangerous missions such as marine rescue, resource exploration, maritime patrol, coastal surveillance, intelligent fishery, environmental monitoring and terrain mapping (Fu and Wang, 2022; Karimi and Lu, 2021; Li et al., 2022; Yao, 2021; Zhu et al., 2021; Chen et al., 2019).The design of a reliable and effective motion control is extremely substantial so that USVs can completely perform the above-mentioned critical duties without mistakes in the marine environment. The movement of the USVs is autonomously controlled with three different approaches in the literature: point stabilization, path following, and trajectory tracking (Wu et al., 2022; Zhang et al., 2023; Yuan et al., 2023). Point stabilization ensures that the ship remains stable with a certain margin of error in the desired location and desired course angle despite the disruptive forces such as strong wind, waves and currents. In trajectory tracking, the autonomous vehicle must arrive to the predefined point at the desired time. Therefore, time plays an important role in this approach (Chen and Lu, 2018). In path following algorithm, it is requested that USVs converge to the predefined path with the least cross tracking error (vertical distance of USV to the predefined path) and chase the predefined path without deviating until it reaches to the target point. Thus, the main objective of path following algorithm is to minimize cross tracking errors (Wang et al.,, 2023). One of the most commonly used algorithms in path following is nonlinear guidance law. It is based on the geometrical approach (Park et al., 2007; Cho et al., 2015; Manzano et al., 2021; Bejarano et al., 2022). Nonlinear guidance law generates a circle to determine the virtual target point. The closest to the target point from the two points formed by the intersection of the circle and the predefined path is selected as the virtual target point. USV tries to catch the predefined path by turning its course towards this virtual target point. The radius of the circle is not deterministic and it is defined by the user. If radius is quite large, USV does not catch the predefined path immediately and it causes huge cross tracking errors. If the radius is too small, the virtual target point cannot be determined because the circle and the predefined path do not intersect, or unstable navigation is observed, which again leads to large cross tracking errors. This un-deterministic parameter is selected as a single and fixed value in the literature (Mao et al., 2015; Niu et al., 2016; Hernández et al., 2020). However, it must be optimized in each time step according to position of USV to the predefined path, velocity and maximum turning rate of USV, number and position of waypoints, and initial course angle of USV. This study proposes the smart circle based nonlinear guidance

Paper Received March, 2023. Revised May, 2023. Accepted July, 2023. Author for Correspondence: Osman ÜNAL.

* *Associate Professor, Sakarya University of Applied Sciences, Department of Mechanical Engineering, Sakarya, Turkey.*

** *Associate Professor, Sakarya University of Applied Sciences, Department of Mechatronic Engineering, Sakarya, Turkey.*

*** *Graduate Student, Sakarya University of Applied Sciences, Department of Mechanical Engineering, Sakarya, Turkey.*

law (SCBNLGL) for path following of unmanned surface vehicle. Simulated annealing optimization technique is integrated to nonlinear guidance law in SCBNLGL. SCBNLGL determines optimal radius of circles for all navigational conditions of USV and it considerably decreases cross tracking errors compared to traditional nonlinear guidance law.

NONLINEAR GUIDANCE LAW

Nonlinear guidance law is a sub-branch of path following algorithm based on geometric approach. Nonlinear guidance law generates a circle whose center is the current position of USV. Two points arise from the intersection of the line (predefined path) and the circle. The closer of these two points to the target point is selected as the virtual target point. USV turns its course to this virtual point and proceeds to reach this virtual target point. Since the position of USV changes in each time step, the circle with the same radius is reproduced over and over again, and new virtual target points formed by the intersection of this circle with the predefined path are determined. USV tries to arrive the target point by following these virtual target points. Figure 1 shows schematic of nonlinear guidance law.

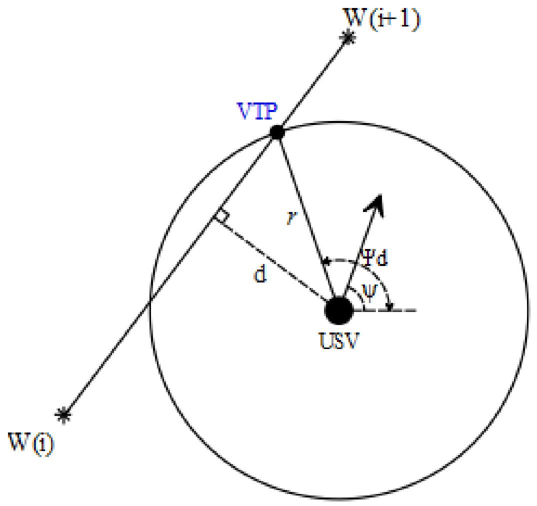

Fig. 1. Schematic of nonlinear guidance law.

In Fig. 1, W(i) and W(i+1) represent start point and target point, respectively. The line between start and target points is predefined path. USV and VTP mean unmanned surface vehicle and virtual target point. d, r, Ψ and Ψd symbolize cross tracking error, the radius of circle (un-deterministic parameter), initial course angle of USV and desired course angle of USV to reach virtual target point, respectively. The purpose of the nonlinear guidance law is that the USV precisely follows the predefined path and reaches the target point in the shortest time with the least cross tracking error. The USV's course angle is changed based on virtual target points. As a result, virtual target points should be carefully appointed so as to guarantee precise and reliable navigation. The position of the virtual target points directly depends on the radius of circles. Thus, the radius of circles (r) is the key parameter in nonlinear guidance law. Navigational accuracy primarily depends on it. It is assumed that the USV moves at a constant speed and there are no external forces, like as wind, waves, or other external sources. The kinematics of the two-dimensional nonlinear guidance law for USV can be described by following equation under these circumstances (Rhee et al., 2010):

$$\begin{aligned} p_x^{t+1} &= p_x^t + v\, cosd(\Psi)\, dt \\ p_y^{t+1} &= p_y^t + v\, sind(\Psi)\, dt \end{aligned} \qquad (1)$$

In Equation 1, v, dt and Ψ symbolize velocity of USV, time step size and course angle of USV. Superscriptions t and t+1 represent current time step and next time step. Subscriptions x and y indicate x and y coordinates. The position of USV is demonstrated by p. The following procedure summarizes the pseudo code of nonlinear guidance law:

Step 1: Define initial position of USV at x and y coordinates; $[p_x^t \; p_y^t]$.

Step 2: Define velocity, initial course angle and maximum turning rate of USV; v, Ψ_i and $\dot{\Psi}_{max}$.

Step 3: Determine predefined path using start and target points at x and y coordinates; $[W_{1_x} \; W_{1_y}]$ and $[W_{2_x} \; W_{2_y}]$

Step 4: Determine radius of circle (key parameter), *r*.

Step 5: Define time step size of numerical simulation, dt.

Step 6: Find the equation of predefined path (line) from start and target points using Equation 2.

$$y - W_{1_y} = \frac{W_{2_y} - W_{1_y}}{W_{2_x} - W_{1_x}} \left(x - W_{1_x}\right) \qquad (2)$$

Equation 3 shows the rearrangement of Eq. 2 to simplify the equation of predefined path:

$$y = \frac{W_{2_y} - W_{1_y}}{W_{2_x} - W_{1_x}}\, x - \frac{W_{2_y} - W_{1_y}}{W_{2_x} - W_{1_x}}\, W_{1_x} + W_{1_y} \qquad (3)$$

Step 7: Find the cross tracking error (*d*) using following Equation 4.

$$d = \frac{\left|\frac{W_{2_y} - W_{1_y}}{W_{2_x} - W_{1_x}} p_x^t - p_y^t - \frac{W_{2_y} - W_{1_y}}{W_{2_x} - W_{1_x}} W_{1_x} + W_{1_y}\right|}{\sqrt{\left(\frac{W_{2_y} - W_{1_y}}{W_{2_x} - W_{1_x}}\right)^2 + (-1)^2}} \qquad (4)$$

Step 8: Find the intersections of predefined path and the circle with radius r.

$$L_x = W_{2_x} - W_{1_x} \tag{5}$$

$$L_y = W_{2_y} - W_{1_y} \tag{6}$$

$$L_r = \sqrt{{L_x}^2 + {L_y}^2} \tag{7}$$

$$\begin{aligned} L_a &= \left(W_{1_x} - p_x^t\right)\left(W_{2_y} - p_y^t\right) \\ L_b &= \left(W_{2_x} - p_x^t\right)\left(W_{1_y} - p_y^t\right) \\ L &= L_a - L_b \end{aligned} \tag{8}$$

$$x_1 = \frac{L\, L_y + sgn(L_y)\, L_x \sqrt{r^2\, {L_r}^2 - L^2}}{{L_r}^2} \tag{9}$$

$$x_2 = \frac{L\, L_y - sgn(L_y)\, L_x \sqrt{r^2\, {L_r}^2 - L^2}}{{L_r}^2} \tag{10}$$

$$y_1 = \frac{-L\, L_x + |L_y| \sqrt{r^2\, {L_r}^2 - L^2}}{{L_r}^2} \tag{11}$$

$$y_2 = \frac{-L\, L_x - |L_y| \sqrt{r^2\, {L_r}^2 - L^2}}{{L_r}^2} \tag{12}$$

In Equation 9 and 10, the function of sgn:

$$\mathrm{sgn}(\mathrm{L_y}) = \begin{cases} -1, & \mathrm{L_y} < 0 \\ 1, & \mathrm{L_y} \geq 0 \end{cases} \tag{13}$$

As a result of the Equation 5 to 13, two intersection points are obtained: $q_1=[x_1\ y_1]$ and $q_2=[x_2\ y_2]$.

Step 9: Measure the distance of the intersection points to the target point.

$$d_1 = \sqrt{\left(W_{2_x} - x_1\right)^2 + \left(W_{2_y} - y_1\right)^2} \tag{14}$$

$$d_2 = \sqrt{\left(W_{2_x} - x_2\right)^2 + \left(W_{2_y} - y_2\right)^2} \tag{15}$$

Step 10: Determine the virtual target point (VTP).

$$\mathrm{VTP} = \begin{cases} \mathrm{q_1}, & \mathrm{d_1} < \mathrm{d_2} \\ \mathrm{q_2}, & \mathrm{d_1} \geq \mathrm{d_2} \end{cases} \tag{16}$$

Step 11: Calculate desired course angle (Ψ_d) using VTP and current position of USV.

$$\Psi_d = atan\left(\frac{VTP_y - p_y^t}{VTP_x - p_x^t}\right) \tag{17}$$

Step 12: Calculate the turning rate of USV ($\dot{\Psi}$) using desired course angle, previous course angle and time step size.

$$\dot{\Psi} = \left(\Psi_d - \Psi_p\right)/dt \tag{18}$$

Step 13: Check the turning rate. If the computed turning rate of USV ($\dot{\Psi}$) is larger than the maximum turning rate of USV ($\dot{\Psi}_{max}$), the desired course angle is updated.

$$\Psi_d = \dot{\Psi}_{max}\, dt + \Psi_p \tag{19}$$

Step 14: Determine the new location of USV in the next time step.

$$\begin{aligned} p_x^{t+1} &= p_x^t + cos(\Psi_d)\, v\, dt \\ p_y^{t+1} &= p_y^t + sin(\Psi_d)\, v\, dt \end{aligned} \tag{20}$$

If the radius of circle (r) is smaller than cross tracking error (d), the circle cannot intersect the predefined path. In that case, virtual target point is not determined and USV cannot follow the predefined path. Therefore, the radius of the circle must be at least the length of the cross tracking error in nonlinear guidance law.If the radius of circle equals to the cross tracking error or the radius of circle is very small (close to the cross tracking error), an offensive navigation may be observed (see Figure 2). In these cases, USV quickly reach to predefined path; however, it cannot follow predefined path accurately. It leaves the predefined path immediately after the first contact. It moves oscillating. The oscillatory navigation (unstable navigation) causes large cross tracking errors.If the radius of circle is too large compared to the cross tracking error, a lethargic navigation is obtained (see Figure 3). USV cannot catch the predefined path rapidly. The first contact with the predefined path takes a very long time. It leads to large cross tracking error again. In brief, the radius of circle must be optimized according to position of USV to predefined path, velocity and course angle of USV, number and location of waypoints, and maximum turning rate of USV. This study presents optimal radius of circles in order to obtain stable and accurate navigation for all navigational condition using simulated annealing optimization method.

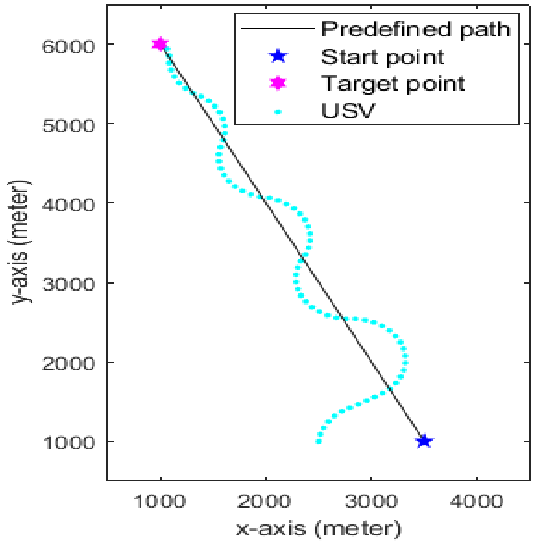

Fig. 2. An offensive navigation because of small radius (r).

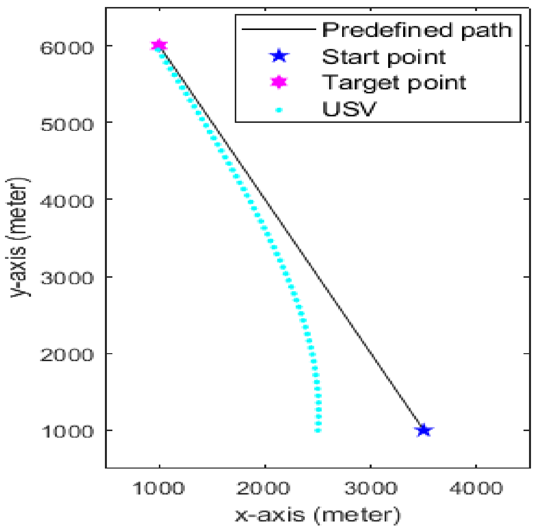

Fig. 3. A lethargic navigation due to large radius (r).

SMART CIRCLE BASED NONLINEAR GUIDANCE LAW

The optimal radius to obtain stable, accurate and precise navigation changes conditions to conditions. For example, optimal radius is not same for different positions of USV respect to predefined path. The optimal value also depends on maximum turning rate and velocity of USV, location of waypoints, and number of target points. It is almost impossible to adjust optimal radius using trial and error method for all navigational conditions, especially for multi-task operations. Therefore, this study proposes the use of simulated annealing optimization technique to automatically find optimal radius for all navigational conditions. Simulated annealing algorithm, introduced by Kirkpatrick et al. (1983), is one of the most effective heuristic techniques for solving global optimization problem (Ait-Saadi et. al., 2022). It can be used for any objective function (Venkatesan and Narayanswamy, 2003). Simulated annealing method is inspired by the real-life physical annealing process. Physical annealing is the process of heating the metal up to the annealing temperature so as to modify the material to the desired structure and then gradually cooling it down. Just as it is easy to change the structural shape of the material at high temperature, the optimization method gets rid of the local minimum barriers by searching for the global minimum value of the objective function in a very wide range with a high coefficient at the beginning. This coefficient, which is determined to be high at the beginning of the optimization, is gradually reduced under certain conditions, allowing the global minimum point to be determined precisely in the global minimum area (like as gradually lowering the temperature in physical annealing). Pattern search optimization can find the global minimum point of the objective function owing to this physical phenomenon. The pseudo code of the simulated annealing optimization technique integrated nonlinear guidance law, called as smart circle based nonlinear guidance law, is illustrated by following steps to clearly explain the proposed novel algorithm:

Step 1: Determine initial position of USV at x and y coordinates.

$\left[p_x^t \; p_y^t\right] = [2500 \; 1000]$

Step 2: Define velocity, initial course angle and maximum turning rate of USV.

$v = 9 \; meters/seconds$

$\Psi_i = 90 \; degrees$

$\dot{\Psi}_{max} = 1 \; degrees/seconds$

Step 3: Determine predefined paths at x and y coordinates.

$\left[W_{1_x} \; W_{1_y}\right] = [3500 \; 1000]$

$\left[W_{2_x} \; W_{2_y}\right] = [1000 \; 6000]$

$\left[W_{3_x} \; W_{3_y}\right] = [1000 \; 1000]$

Step 4: Define numerical simulation time step size.

$d_t = 10 \; second$

Step 5: Find the equation of first predefined path (line) from start point $\left(W_{1_x}, W_{1_y}\right)$ and first target points $\left(W_{2_x} \; W_{2_y}\right)$ using Equation 2.

$$y - 1000 = \frac{6000-1000}{1000-3500}(x - 3500) \tag{21}$$

Rearrange Equation 21 to simplify the equation of first predefined path:

$$y = -2x + 8000 \tag{22}$$

Step 6: Find the optimal radius of circle using simulated annealing optimization technique for next time step (t+1).
Step 6.1: Define parameters for simulated annealing algorithm.
-Objective function, f: nonlinear guidance law
-Input parameter of objective function, r_i: radius of circles
-Output parameter of objective function, $f(r_i)$: cross tracking errors.
-Choosing the search domain to find optimal radius: Lower and upper limits of radius are selected as the cross tracking error and twice cross tracking error in current time step.
[lb, ub]=[d, 2d]
-Defining constant value of simulated annealing (T). It is chosen T=100 as default value in Matlab environment.
-Determination of constant value reduction factor (c): c=0.8.
-Maximum number of iteration: k=100. If the number of iterations performed to find optimal radius is higher than 100, loop is broken and optimization is stopped.
-Termination factor (permissible error): ε=0.01. If $\varepsilon > |r_2 - r_1|$, loop is broken and optimization is stopped.
Step 6.2: Find the cross tracking error (*d*) using Equation 4 for i^th iteration in simulated annealing optimization.

$$d = 894.42\ meter \tag{23}$$

Initial radius within the lower and upper bounds for first iteration in simulated annealing is (i=1):
$r_i=r_1=(lb+ub)/2=$ 1341.63 meter
Step 6.3: Find the intersections of first predefined path and the circle with radius r_i (for first iteration r_1=1341.63 meter) using equations 5-12.

$$L_x = -2500 \tag{24}$$

$$L_y = 5000 \tag{25}$$

$$L_r = 5590.17 \tag{26}$$

$$\begin{aligned} L_a &= 5 \cdot 10^4 \\ L_b &= 0 \\ L &= 5 \cdot 10^4 \end{aligned} \tag{27}$$

$$x_1 = 2600.02 \tag{28}$$

$$x_2 = 3799.98 \tag{29}$$

$$y_1 = 2799.96 \tag{30}$$

$$y_2 = 400.04 \tag{31}$$

Two intersection points are obtained: $q_1=[x_1\ y_1]$ and $q_2=[x_2\ y_2]$.
Step 6.4: Calculate the distance of the intersection points using equations 14 and 15.

$$d_1 = 3577.75 \tag{32}$$

$$d_2 = 6260.95 \tag{33}$$

Step 6.5: d1<d2, so VTP=[2600.02 2799.96].
Step 6.6: Determine the desired course angle (Ψ_d) using equation 17.

$$\Psi_d = 86.82\ degrees \tag{34}$$

Step 6.7: Calculate the turning rate of USV ($\dot{\Psi}$) using equation 18.

$$\dot{\Psi} = 0.32\ degrees/seconds \tag{35}$$

Step 6.8: The computed turning rate of USV ($\dot{\Psi} = 0.32$) is not larger than the maximum turning rate of USV ($\dot{\Psi}_{max} = 1$). Thus there is no need to update desired course angle.
Step 6.9: Determine the new location of USV in the next time step using Equation 20.

$$\begin{aligned} p_x^{t+1} &= 2500.5 \\ p_y^{t+1} &= 1089.9 \end{aligned} \tag{36}$$

Step 6.10: Calculate the new cross tracking error (d_1) using Step 6.2 and note that the use of r_1=1341.63 meter leads to d_1=853.77. This is the end of first iteration in simulated annealing algorithm. $f(r_1)=d_1.f(1341.63)=853.77$.

After Step 6.10, simulated annealing optimization method generates new possible radius that gives lower cross tracking error using following sub-steps:

6.10.1: Choose the next radius (for second iteration, r_2=1156.65) in vicinity of r_1 using randomly distributed number within lower and upper limit, and then return back to Step 6.3 in order to calculate new cross tracking error (d_2) using new radius (r_2). Note the new values, f(r_2)=d_2. f(1156.65)=848.96.

Step 6.10.2: Check the internal energy (ΔE).

Step 6.10.2.1: If ΔE is lower than zero, accept new radius.

$\Delta E = f(r_2) - f(r_1) = 848.96 - 853.77 = -4.81$

ΔE is lower than zero, so go to Step 6.10.3.

Step 6.10.2.1: If ΔE is not lower than zero (suppose that r_2=1546.32 and d_2=859.63), this iteration is called as bad move but r_2 is not directly rejected. The metropolis criterion is used to decide whether to accept or reject this worse point (r_2):

T=100 (see Step 6.1 for constant value "T").

$\Delta E = f(r_2) - f(r_1) = 859.63 - 853.77 = 5.86$

The probability of accepting this worse point (r_2): p.

$$p = e^{\frac{-\Delta E}{T}} = e^{\frac{-5.86}{100}} = 0.94 \tag{37}$$

Generate a uniformly distributed random number (u) between 0 and 1. Suppose that *u* equals to 0.72. The probability of accepting the worse point (p) is greater than randomly selected number. Therefore, accept this worse point (r_2) and go to Step 6.10.3. Otherwise, reject this possible value and return back to Step 6.10.1 in order to generate new radius of circle. It is important note that simulated annealing searches the best minimum value in global area and gets rid of the local minimum barriers owing to Step 6.10.2.1. That's why it is called as a global optimization.

Step 6.10.3: Check the permissible error (ε). If ε is lower than 0.01, stop the iteration. Else, go to Step 6.10.4.

$\varepsilon = |r_2 - r_1| = |1156.65 - 1341.63| = 184.9800$

Step 6.10.4: This iteration is good move because internal energy is lower than zero. Thus, decreases the constant value (T=80).

This is the end of second iteration, so return back to Step 6.10.1 so as to find possible best radius of circle. This loop continues until the USV reaches the first target point. After the first task, same procedures are applied for the second target point. The USV's course angle is calculated by determining the radii that result in the least track errors owing to simulated annealing integrated nonlinear guidance law (SCBNLGL).

Increasing of internal energy (ΔE) leads to reduction of the probability of accepting the worse point (p). This avoids worse radius values resulting in greater cross tracking errors. The large constant value (T=100) at the beginning of the optimization (similarly in physical annealing, it is easy to change the physical structure of the material when the temperature is high) allows the global minimum value to be searched in a wider range and allows the algorithm to get rid of the local minimum barriers. Reducing the simulated annealing constant number (T) in each iteration (like as gradually decreasing the temperature in physical annealing) enables the simulated annealing optimization method to focus on the area where the global optimization value is located and precisely determine the global minimum.

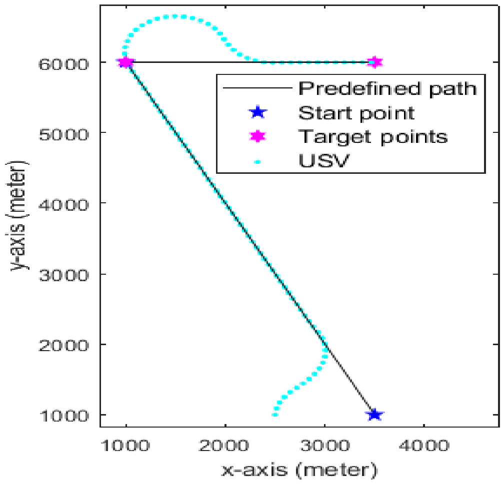

Fig. 4. Stable and optimal navigation of USV using SCBNLGL algorithm.

Figure 4 shows the stable and accurate navigation of USV using proposed SCBNLGD. It is important note that USV must contact target points in this multi-tasks numerical simulation. Secondly, maximum turning rate of USV is 1 degree per second. USV moves under this precondition. In Fig. 4, SCBNLGD algorithm generates unique circles that lead to minimum cross tracking errors in both first and second predefined paths. Thus, USV automatically adapts itself for multi-tasks operation without human-being intervention. USV reaches predefined paths quickly as well as stably. SCBNLGD algorithm reduces total cross tracking errors significantly compared to traditional nonlinear guidance law (see Fig. 2 and Fig. 3).

RESULTS AND DISCUSSIONS

Initial position of USV respect to first predefined path, location of target points, initial course angle of USV, velocity of USV, and maximum turning rate of USV directly affect the size of the optimum radius. In this section, these five important parameters are investigated and discussed to prove the adaptability of SCBNLGL for every navigational condition and all kinds of USV. Numerical results clearly show the superiority of SCBNLGL algorithm over conventional nonlinear guidance law.

Table 1. Comparison of SCNLGL with the classical

NLGL for different navigational conditions.

Cases		NLGL		Novel SCNLGL	
		Cross Track Errors (meter) for α=40m	Cross Track Errors (meter) for α=1000m	Cross Track Errors (meter)	Optim α (meter)
Initial position	[500,1500]	33630.0	37463.0	29008.2	167.1
	[1500,2500]	11984.0	14066.0	8918.8	169.5
	[4000,2000]	21571.0	24563.0	17583.4	173.7
	[3000,3000]	7744.2	5946.1	2962.0	180.7
Initial course	0 degs	48147.0	52155.0	44288.0	170.8
	90 degs	21571.0	24563.0	17583.4	173.7
	180 degs	17689.0	21096.0	13674.0	170.9
	270 degs	21601.0	22213.0	17568.0	173.4
velocity	7 m/s	23475.0	30693.0	21161.0	131.8
	9 m/s	21571.0	24563.0	17583.4	173.7
	11 m/s	21442.0	20646.0	15294.0	209.6
	13 m/s	25220.0	17932.0	13701.0	257.9
max turn rate	0.75 deg/s	31880.0	25670.0	19223.0	227.9
	1.00 deg/s	21571.0	24563.0	17583.4	173.7
	1.25 deg/s	18336.0	23876.0	16592.0	137.1
	1.50 deg/s	17073.0	23417.0	15925.0	119.4
multi-tasks	**1st path**	21571.0	24563.0	17583.4	173.7
	2nd path	34190.1	40192.6	9865.8	177.7
	3rd path	41122.1	42619.3	811.9	147.8

In Table 1, the radius of circle is determined as a function of alpha (α) parameter in order to simplify the calculation and reduce the computational cost. The radius of circles must be larger than cross tracking error in each time step to intersect circle and predefined path. The alpha value is the parameter that expresses how much value should be added to the cross tracking error in order to make a stable and effective navigation. The numerical simulations in this section were performed using three different methods: traditional nonlinear guidance law (NLGL) using fixed small alpha (α=40 meter, small circles), traditional nonlinear guidance law (NLGL) using fixed large alpha (α=1000 meter, large circles), smart circle based nonlinear guidance law (SCBNLGL) using simulated annealing optimization method (optimal circles). Cross tracking errors were calculated using each method for five different cases: four different positions of USV, four different initial course angles of USV, four different velocities of USV, four different maximum turning rates of USV and three different predefined paths. While position of USV, number and location of predefined paths, and initial course angle of USV refers to different navigational conditions, velocity and maximum turning rate of USV imply the kind of marine vessels. The objective of these numerical simulations is to illustrate the versatility of SCBNLGL for any navigational circumstance and any kinds of USV.

Initial position of USV influences the optimal radius of circle. Different initial positions of USV require unique and optimal radius to minimize cross tracking errors. The use of fixed radius is not convenient for effective navigation. Moreover, the size of radius must be greater than cross tracking error in current time step. If fixed radius that gives good navigation for previous initial position of USV may be lesser than cross tracking error for new initial position of USV. In this case, the circle and predefined path do not intersect and virtual target point does not determined. USV cannot follow the predefined path. Therefore, the optimum radius should be automatically determined by the optimization method considering the different initial positions of the USV. Four different initial positions of USV have been investigated to show the advantage of SCBNLGL. Proposed model has been compared to conventional nonlinear guidance law (NLGL) using small and large fixed value. It is observed that SCBNLGL adapts different initial position of USV and it generates unique value to obtain optimal circle. Table 1, shows cross tracking errors of classical NLGL and SCBNLGL. Optimal values for different initial positions of USV have been calculated by simulated annealing optimization method as 167.12, 169.55, 173.74 and 180.72. As the distance of the USV's initial position to the predefined path increases, smaller alpha values are used to determine the optimum radius. SCBNLGL reduces cross tracking errors considerably compared to conventional NLGL for different initial positions of USV.

Initial course angle of USV also affects the cross tracking errors. If the bow of the USV is towards the predefined path, the USV can easily reach the desired route with less cross tracking errors. The high speed USVs intend to unstable navigation. Therefore, the optimum alpha value for the marine vessel with a speed of 13 m/s in Table 1 has been calculated as approximately twice the optimum alpha value of the marine vessel with a speed of 7 m/s. Maximum maneuverability of the marine vessel is defined as maximum turning rate of USV. High maneuverability vessels require the smaller alpha value to obtain optimal radius of circle compared to low maneuverability USV.

Figure 5 indicates the numerical simulation of USV for the multi-tasks operation using NLGL and SCBNLGL. In Fig. 5, the initial position of USV is selected as [4000, 2000] for x and y coordinates, respectively. Target points are [1500, 6000], [5500, 4000] and [5500, 1000]. Velocity and maximum turning rate are chosen as 9 meter per second and 1 degree per second, respectively. NLGL using small alpha value leads to unstable navigation and larger one causes lazy navigation. Total cross tracking errors for small and large alpha values are calculated as 96883.2 meter and 107374.8 meter. Both of them give large cross tracking errors. On the other hand, SCBNLGL reaches to predefined path rapidly as well as stably. It uses unique and optimal alpha value (for 1st path α=173.7m, for 2nd path α=177.7m, for 3rd path α=147.8m) to minimize cross tracking errors. Total cross tracking error using SCBNLG is only 28261.1 meter. It is even smaller than one-third of cross tracking errors of the conventional nonlinear guidance law.

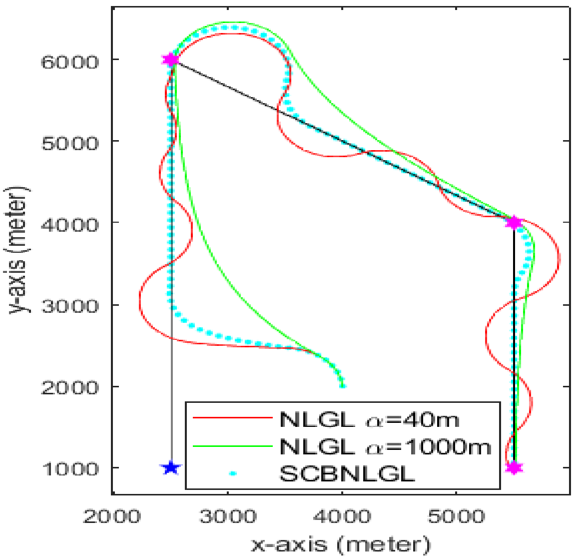

Fig.5. Comparison of traditional NLGL and proposed SCBNLGL for multi-tasks operation.

CONCLUSIONS

In this study, simulated annealing optimization method was integrated with nonlinear guidance law to determine optimal navigational conditions of USV. The proposed technique was called as smart circle based nonlinear guidance law (SCBNLGL). In contrast to classical nonlinear guidance law, SCBNLGL generates unique and adjustable circles to determine virtual target points. The use of fixed radius in conventional nonlinear guidance law is not appropriate for every navigational condition and all kinds of marine vessel. The optimal radius depends on initial position, maximum turning rate, velocity and initial course angle of USV, location of target points, and number of predefined paths. It is almost impossible to find the most suitable radius for each sailing situation and each type of vessel manually by trial and error method. SCBNLGL automatically determine optimal radius to find best virtual target points for all navigational conditions and all types of USV. Proposed model was tested in numerical simulations using different cases. It is concluded that SCBNLGL adapt all conditions and it significantly decreases cross tracking errors compared to traditional nonlinear guidance law.

REFERENCES

Ait-Saadi, A., Meraihi, Y., Soukane, A., Ramdane-Cherif, A., & Gabis, A. B., "A novel hybrid Chaotic Aquila Optimization algorithm with Simulated Annealing for Unmanned Aerial Vehicles path planning," *Computers and Electrical Engineering*, Vol.104, pp.108461 (2022).

Bejarano, G., Manzano, J. M., Salvador, J. R., & Limon, D., "Nonlinear model predictive control-based guidance law for path following of unmanned surface vehicles," *Ocean Engineering*, Vol.258, pp.111764 (2022).

Chen, Y. H., Chen, Y. Y., Huang, J. M., & Guo, N. Z., "Guidance Law Design of Unmanned Robot Boats Based on Vision Navigation", *J. CSME*, Vol.40, No.3, pp.248-255 (2019).

Chen, Y. Y., & Lu, C. T., "Nonlinear guidance law design of autonomous underwater vehicles," *J. CSME*, Vol.39, No.3, pp.319-328 (2022).

Cho, N., Kim, Y., & Park, S., "Three-dimensional nonlinear differential geometric path-following guidance law," *Journal of Guidance, Control, and Dynamics*, Vol.38, No. 12, pp.2366-2385 (2015).

Fu, M., & Wang, L., "Finite-time coordinated path following control of underactuated surface vehicles based on event-triggered mechanism," *Ocean Engineering*, Vol.246, pp.110530 (2022).

Hernández-Morales, L., Valeriano-Medina, Y., Hernández-Santana, L., & Mesa-Suarez, E., "Nonlinear guidance law algorithm applied to a small unmanned surface vehicle," *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, Vol.234, No.3, pp.623-633 (2020).

Karimi, H. R., & Lu, Y., "Guidance and control methodologies for marine vehicles: A survey," *Control Engineering Practice*, Vol.111, pp.104785 (2021).

Kirkpatrick, S., Gelatt Jr, C. D., & Vecchi, M. P., "Optimization by simulated annealing," *Science*, Vol.220, pp.671-680 (1983).

Li, J., Zhang, G., Shan, Q., & Zhang, W., "A novel cooperative design for USV-UAV systems: 3D mapping guidance and adaptive fuzzy control," *IEEE Transactions on Control of Network Systems*, pp.1-11 (2022).

Manzano, J. M., Salvador, J. R., Bejarano, G., & Limon, D., "Nonlinear model predictive control applied to robust guidance of autonomous surface vehicles," *In 2021 60th*

IEEE Conference on Decision and Control (CDC), pp. 5735-5740 (2021).

Mao, Y., Chen, Q., Hou, Z., & Zheng, G. (2015, July). An improved nonlinear guidance law for unmanned aerial vehicles path following. In 2015 34th Chinese control conference (CCC) (pp. 5271-5276). IEEE.

Niu, H., Lu, Y., Savvaris, A., & Tsourdos, A., "Efficient path following algorithm for unmanned surface vehicle," *In OCEANS 2016-Shanghai*, pp.1-7 (2016).

Park, S., Deyst, J., & How, J. P., "Performance and lyapunov stability of a nonlinear path following guidance method," *Journal of guidance, control, and dynamics*, Vol.30, No.6, pp.1718-1728 (2007).

Venkatesan, R., & Narayanswamy, R., "Application of genetic algorithm and simulated annealing for optimization of extrusion die ratio and die cone angle," *In Proceedings of the 12th International Scientific Conference on Achievements in Mechanical and Material Engineering, Gliwice, Zakopane*, pp. 1001-1006 (2003).

Wang, H., Tan, H., & Peng, Z., "Quantized Communications in Containment Maneuvering for Output Constrained Marine Surface Vehicles: Theory and Experiment," *IEEE Transactions on Industrial Electronics*, (2023).

Wu, J., Jin, Z., Liu, A., Yu, L., & Yang, F., "A hybrid deep-Q-network and model predictive control for point stabilization of visual servoing systems," *Control Engineering Practice*, Vol.128, pp.105314 (2022).

Yao, Q., "Dual-disturbance-observer-based robust finite-time trajectory tracking control for robotic surface vehicle under measurement uncertainties," *Ocean Engineering*, Vol.242, pp.110183 (2021).

Yuan, C., Shuai, C., Ma, J., & Fang, Y., "An efficient control allocation algorithm for over-actuated AUVs trajectory tracking with fault-tolerant control," *Ocean Engineering*, Vol.273, pp.113976 (2023).

Zhang, G., Wang, L., Li, J., & Zhang, W., "Improved LVS guidance and path-following control for unmanned sailboat robot with the minimum triggered setting," *Ocean Engineering*, Vol.272, pp.113860 (2023).

Zhu, C., Zeng, J., Huang, B., Su, Y., & Su, Z., "Saturated approximation-free prescribed performance trajectory tracking control for autonomous marine surface vehicle," *Ocean Engineering*, Vol.237, pp.109602 (2021).