



Development of Path-Finding Controller Design for Hovercraft Model via Neural Network Technique and Meta-Heuristic Algorithms

Sura Muhi Hussein^{1*} Ahmed Sabah Al-Araji¹

¹Computer Engineering Department, University of Technology –Iraq, Baghdad, Iraq

* Corresponding author's Email: 120126@uotechnology.edu.iq

Abstract: The objective of the navigation process is to determine the most efficient route for the hovercraft and regulate its movement along that route without any oscillation. The primary goal of this work is to determine the most efficient routes for a hovercraft operating in a global environment. The proposed approach, called Artificial Bee Colony Self Perception Particle Swarm Optimization (ABC-SPPSO), is utilized to accomplish this objective. The main benefit of utilizing the ABC-SPPSO algorithm is its ability to design the most efficient route while preventing collisions with stationary obstructions. In addition, a suggested controller that consists of a Feedforward Numerical Inverse Dynamic Controller (FNIDC) and a Feedback Neural Network Radial Basis Function (FNNRBF) control technique with the Grey Wolf Optimization (GWO) algorithm will be utilized to guide the hovercraft along predetermined paths. This suggested controller is designed to regulate the nonlinear dynamics of the hovercraft system in order to efficiently and rapidly generate the forces exerted by the starboard and portboard fans. These forces are utilized to control the orientation and position of the hovercraft. Moreover, the utilization of the suggested controller effectively reduces the differences between the desired and the actual positions in both the X-axis and the Y-axis. Additionally, the controller nearly eliminates any deviation in orientation and ensures a stable response without any oscillation. Specifically, the controller ensures that the hovercraft will promptly and accurately adhere to its intended trajectories. Ultimately, we verify the accuracy of the numerical simulation outcomes of the suggested control approach by contrasting them with those of other controllers, specifically in relation to the highest level of error improvement in the X-position and the Y-position. In particular, when comparing the proposed controller to the Improved Quasi-Velocities (IQV) controller, the results show that the suggested controller decreases the error rate of tracking on the X-position by 22% and enhances the tracking error rate on the Y-position by 14.8%. Furthermore, the suggested controller was evaluated against the terminal sliding mode controller (TSMC), and the results of the comparison indicate that the proposed controller enhances the error rate of tracking on the X-position by 50.7% and on the Y-position by 64.5%. Additionally, the suggested controller was evaluated against the nonlinear cascade controller, and the comparative analysis demonstrates that the suggested controller enhances the X-position and the Y-position tracking error rates by 25.9% and by 33%, respectively. Finally, from a comparative study with the neural network-based adaptive dynamic inversion controller, the proposed controller enhances the error rate of tracking on the X-position by 51% and on the Y-position by 42.9%.

Keywords: Artificial bee colony self-perception particle swarm optimization, Path finding, Hovercraft model feedforward numerical inverse dynamic controller, Radial basis function neural network, Trajectory tracking.

1. Introduction

The design of the hovercraft is based on a basic ship equation that exhibits nonlinearity and under-actuation. In this context, the utilization of high-performance vehicles featuring distinctive material architectures and navigation principles has witnessed a notable rise in several domains, such as

icebreakers, maritime transportation, military transportation, civilian survival, rescue operations, and scientific research recreational activities during the 21st century [1]. Hence, while contemplating the utilization of hovercrafts, they can navigate both water and land, provided that there are impediments of reasonable size, all while maintaining a state of hovering on a cushion of high-pressure air [2, 3].

When dealing with hovercraft navigation challenges, it is crucial to address three primary concerns: safety, accuracy, and efficiency. The primary considerations are ensuring a path free from collisions and accurately following the designated route. Efficiency refers to the ability of an algorithm to avoid unnecessary repetition and excessive processes, which is a futile use of time and energy [4]. Of all these issues, it might be argued that path finding is the most crucial aspect of vehicle navigation. Particularly, it is described as determining a geometrical path that must be the quickest and most obstacle-free route from the start to the destination states [5-6]. Furthermore, the hovercraft necessitates a control system to modify the force of each fan in order to achieve the desired position. Although several trajectory-tracking control methods are available for monitoring the hovercraft system, the primary objective is to operate the system in a cost-effective and efficient manner while maintaining the controller's robustness, stability, and reliability. As a result, numerous strategies have been proposed to address problems with trajectory-tracking and guarantee that the hovercraft adheres to the set path without faltering. For instance, in [7], the authors proposed a nonlinear controller for trajectory-tracking of the hovercraft system utilizing two controllers. The first controller is the position controller that depends on the inverse kinematics of the hovercraft model, and the second one is the forces selection technique that depends on nine cases of the forces based on if-then control actions with four control gains selected by the trial-and-error method to stabilize the control law. However, these values of the parameters and the nine cases of the forces lead to generating errors in the x-axis and the y-axis of the hovercraft during motion because the forces selection did not cover all the regions of the hovercraft platform movement in the environments. In [8], the researchers explained the direct neural network-based adaptive control structure that compensates for unknown hovercraft nonlinearities in a feedback linearizing control framework based on the line-of-sight guidance law with five control gains. These gains are obtained by the trial-and-error method to stabilize the system based on the Lyapunov function. However, these values of the control law parameters lead to significant errors in the position and orientation of the hovercraft and considerable oscillations in the kinematic velocities' control action. In [9], the author proposed a nonlinear controller based on quasi-velocities using the terminal sliding mode control algorithm to track two desired paths with minimum tracking errors of the hovercraft model.

However, the limitation of this work is that there are twelve parameters of the control law initialized using the trial-and-error method to stabilize the system based on the Lyapunov method. Nevertheless, these values cause a significant error in the starting of the hovercraft motion in the x-axis and the y-axis. Moreover, there is a high fluctuation in the response of the control action. Moreover, in [10], the authors employed a nonlinear control strategy for an under-actuated hovercraft using the derivative-free nonlinear Kalman filter to estimate the state and smooth out disturbances, which improved the maneuvering and accuracy of trajectory tracking. Nonetheless, the issue of this work is the linearization of the model that increased the problem of the localization and autonomous navigation of the hovercraft. The researchers in [11] introduced an effective control technique for hovercraft navigation utilizing a compensated back-stepping method with a finite-time extended state observer and the line-of-sight guidance law to ensure safety. However, the control law has twenty-eight parameters that were found by the trial-and-error method. Therefore, there is a small tracking error in the position and orientation of the hovercraft, and the response of the two control actions has oscillation. On the other hand, the authors in [12] designed a trajectory-tracking controller for a hovercraft platform with unmeasured linear velocity and subject to time-varying observers based on the Lyapunov method with eight control gains. However, these parameters were selected using trial-and-error method. Therefore, the stabilization of the control law was poor with a limited range of operation. In addition, the work in [13] presented a model-based approach to the nonlinear tracking control for both the horizontal and vertical dynamics of an under-actuated hovercraft vehicle with feedback accurate velocity signals and an observer-based sensor fusion using acceleration measurements and data from an optical flux sensor. Nonetheless, the stabilizing feedback control law was insufficient because the model of the hovercraft was linearized. On the other hand, the researchers in [14] developed fuzzy-based three PID controllers with nine control gains for non-holonomic hovercrafts to handle dynamic restrictions in path-planning and obstacle avoidance with the aid of IoT sensors and digital image processing. Nevertheless, the drawback of this controller is the limited number of the used scenarios with slow time response to generate the nine control parameters. In [15], the authors addressed the problem of the path following control for an under-actuated hovercraft in the presence of external disturbances and uncertain

parameters using the back-stepping method with ten parameters of the control law to steer the hovercraft to a neighborhood of the desired path and obtain global practical stability. However, the issue of this work is that the gains were obtained by the trial-and-error method utilizing a very small search space. Consequently, these values of the control law parameters lead to significant errors in the position and orientation of the hovercraft. Moreover, the authors in [16] proposed to design and experimentally validate a nonlinear controller for a hovercraft model based on the Lyapunov function actuated through thrust force and rudder angle and subject to static and linear velocity drag forces. In addition, an online estimation algorithm continuously estimates the drag parameters of the hovercraft to improve the performance over different cases. However, the convergence is considerably slow in finding the parameters because the chosen oval trajectory was not rich enough in lateral movements for the hovercraft, and the adaptation of these parameters was slow.

The problem definition of this work is divided into two stages. The first stage involves developing an optimal or nearly optimal desired path equation for the hovercraft platform to solve three problems of the path planning and achieve the requirements in terms of obstacle-free navigation, minimizing the distance to the target, and establishing the smoothest path for two scenarios of the hovercraft in a complex environment. The second stage involves developing the trajectory-tracking controller of the hovercraft that solves three problems of the trajectory-tracking and achieves the following requirements: the hovercraft must follow a predetermined path accurately without sliding and with minimal errors in position and orientation tracking. In particular, the main scientific contributions of this work include addressing the problem statement utilizing a hybrid method called artificial bee colony self-perception particle swarm optimization (ABC-SPPSO) to generate an optimal or nearly optimal smooth desired path equation for a hovercraft. This path should have the shortest distance and avoid collisions in a global environment, and the controller should generate smooth and optimal control action values of the two fans through numerical simulation. These objectives will be achieved by employing a suggested feedforward numerical inverse dynamic controller and a feedback neural network radial basis function control technique with the grey wolf optimization algorithm to find and tune the optimal or near-optimal control gain parameters. This control strategy obtains the fast and optimal value of the

fans' forces control actions that will thrust the hovercraft model by quickly tracking the desired path and stabilizing the motion.

This paper is structured as follows: Section 2 demonstrates an under-actuated hovercraft system. Section 3 outlines the path-finding controller technique. Section 4 provides the simulation results, and Section 5 presents the conclusions of this research.

2. Under-actuated hovercraft system

The term "under-actuated" pertains to hovercrafts that exhibit non-holonomic constraints in their motion, as the number of independent actuators is less than the number of degrees of freedom (DOF). The level of complexity in resolving issues related to under-actuated hovercrafts is greatly influenced by the specific configuration of the vehicle [14]. This paper focuses on the hovercraft system illustrated in Fig. 1. The hovercraft is pushed by an air propeller and supported by a skirt, which can be thought of as a cushion of air held within a flexible structure. Hovercraft propulsion is achieved using two electric motor-driven propellers, which facilitate forward and lateral maneuvering [17].

The hovercraft is a multi-input multi-output system that incorporates two input forces, namely the starboard and the portboard of two fans. Additionally, it has three outputs associated with the global positions of x , y , and ψ . The comprehension of hovercraft motion necessitates the elucidation of both kinematic and dynamic equations pertaining to the hovercraft. The hovercraft's kinematic and dynamic equations of motion can be expressed as follows [18, 19]:

$$\dot{x}_s = u_s \cos \psi - v_s \sin \psi \quad (1)$$

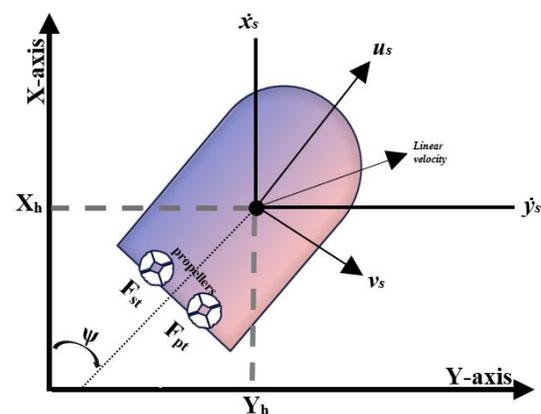


Figure. 1 The hovercraft system.

Table 1. Parameters definition of the equations for the hovercraft model.

Variables	Meaning	Units of variable
u_s, v_s	Surge and Sway speeds	m/s
\dot{x}_s, \dot{y}_s	Speeds in the directions of x and y coordinates	m/s
ψ	The orientation of the hovercraft	rad
r_s	Angular velocity	rad/s
d_{vi}, d_{ro}	The coefficient of viscous and rotational friction	(kg/s), (kg.m/s)
F_{st}, F_{pt}	The starboard, and portboard fans forces	N
L	Forces' arm	m
J	Rotational inertia	kg.m ²
M	Mass of the hovercraft	kg
\dot{r}_s	Angular acceleration	rad/s ²

$$J \dot{r}_s + d_{ro} r_s = L(F_{st} - F_{pt}) \tag{6}$$

The definitions of the variables in the hovercraft equations are presented in Table 1.

3. Path finding control strategy design

The proposed path finding control strategy for the hovercraft, which is shown in Fig. 2, consists of two slices: the first slice is to solve the path finding problem utilizing a hybrid method called artificial bee colony self-perception particle swarm optimization (ABC-SPPSO) by generating an optimal or nearly optimal smooth desired path equation for the hovercraft.

This path should have the shortest distance and also avoid collisions in a global environment. The second slice is to generate smooth and optimal values for the control actions of the two fans of the hovercraft model.

This is achieved by employing a suggested feedforward numerical inverse dynamic controller and a feedback neural network radial basis function control technique with the grey wolf optimization algorithm, which is used to find and tune the optimal or near-optimal control gain parameters. This control strategy obtains the fast and optimal value of the fans' forces control actions that will thrust the hovercraft model by quickly tracking the desired path and stabilizing the motion.

$$\dot{y}_s = u_s \sin \psi + v_s \cos \psi \tag{2}$$

$$\dot{\psi} = r_s \tag{3}$$

$$M \dot{u}_s - M v_s r_s + d_{vi} u_s = F_{st} + F_{pt} \tag{4}$$

$$M \dot{v}_s + M u_s r_s + d_{vi} v_s = 0 \tag{5}$$

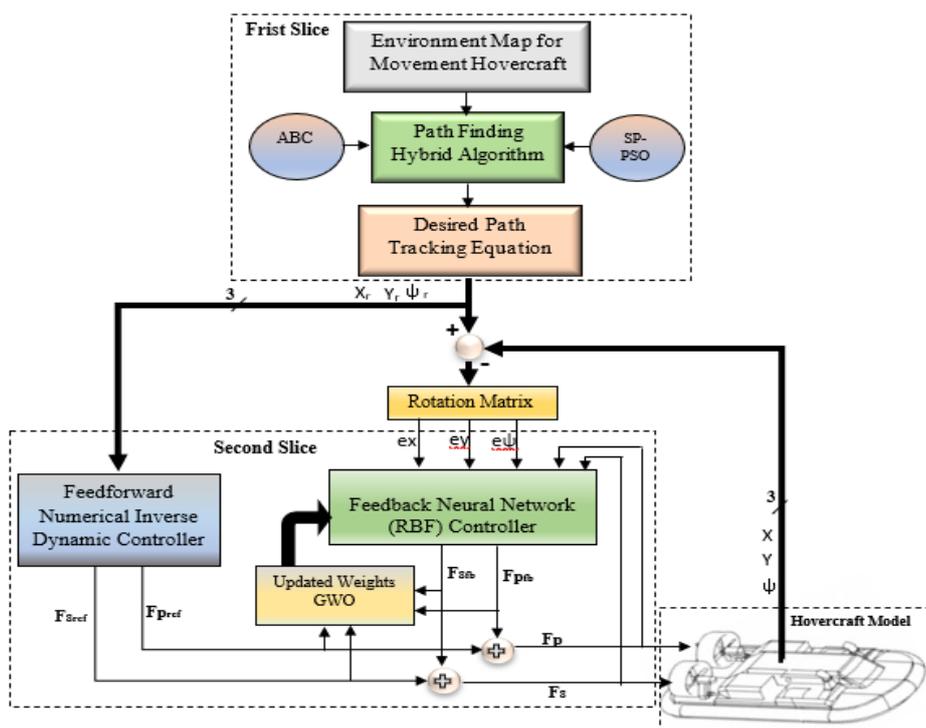


Figure. 2 The proposed path finding control strategy design for the hovercraft model

3.1 Hybrid path finding methodology

The work presents a hybrid path finding algorithm for hovercraft navigation, with a specific emphasis on the following three goals: obstacle-free navigation; minimizing the distance to the target; and establishing the smoothest path.

This approach utilizes the artificial bee colony self-perception particle swarm optimization method (ABC-SPPSO), which is adopted from [19]. The ABC algorithm, which was formulated by Karaboga et al. in 2005, is a computational approach that draws inspiration from the behavioral patterns observed in honey bee colonies [20]. In particular, this algorithm comprises three key components: food sources, employed and unemployed bees (referred to as spectator and scout), and two primary behavioral patterns: nectar source recruitment and abandonment. The algorithm employs recruit, onlooker, and scout bees in order to optimize the trajectory of the hovercraft. The sequential phases of this approach are illustrated in Eq. (7), Eq. (8), Eq. (9) and Eq. (10) [19, 20].

$$n_{i,j} = a_{i,j} + r_{i,j}(a_{i,j} - a_{k,j}) \quad (7)$$

$$f(a_i) = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (8)$$

$$ft_i = \begin{cases} \frac{1}{1+f(a_i)}, & \text{if } f(a_i) \geq 0 \\ 1 + |a_i|, & \text{if } f(a_i) < 0 \end{cases} \quad (9)$$

$$Pi(i) = \frac{ft_i}{\sum ft_i} \quad (10)$$

The variables' definitions for the equations used in this approach (ABC) are displayed in Table 2.

However, the ABC algorithm faces issues with accuracy and convergence speed, and it is effective for exploration but not for exploiting. On the other hand, Particle Swarm Optimisation (PSO) is a population-based stochastic optimisation method that draws inspiration from the phenomenon of fish crowding together and the behavior of bird flocks. It imitates a group of particles searching for the most

Table 2. Variables meaning of ABC equations

Variables	Meaning
$a_{i,j}$	Solution in the swarm
$r_{i,j}$	Random variable between [0,1]
$n_{i,j}$	Potential solution closer to its initial one
ft_i	The solution' fitness
$Pi(i)$	Probability of the solution

Table 3. Variables definition of the SP-PSO parameters

Variables	Meaning
w_i^k	Inertia weight
v_i^k	Velocity of the i^{th} particle at the k^{th} iteration
P_{SR}	Perception factor of the self-cognizance
P_{SP}	Perception factor of the social cognizance
c_1, c_2	Constants of acceleration, where (c_1+c_2) is less than 4
r_1, r_2	Random variable of interval (0,1)
L_{best}^k	The best personal location
G_{best}^k	The best global particles' position
x_i^k	The current position (or solution) of the i^{th} particle at the k^{th} iteration.

efficient solution. The PSO technique iteratively updates the particle swarm, where each particle represents a potential solution [21, 22]. Nevertheless, the conventional PSO lacks the population variety, and it easily falls into local minima. One proposed technique aims to enhance the PSO algorithm by

incorporating particle self-perception. This modification enhances the system's evolutionary process, leading to improved outcomes within a reduced time. Humans are capable of self-regulation, and group learning is where this idea originates from. As the algorithm progresses, particles can update their locations and velocities according to Eq. (11) and Eq. (12) [19], which allows them to move towards search regions that are more advantageous.

$$v_i^{k+1} = w_i^k v_i^k + P_{SR} c_1 r_1 (L_{best}^k - x_i^k) + P_{SP} c_2 r_2 (G_{best}^k - x_i^k) \quad (11)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (12)$$

Table 3 displays the symbols' definition of the equations depicting the relationship between velocity and location.

The ABC algorithm exhibits a sophisticated exploration capability and necessitates only a minimal number of control parameters. Consequently, it guarantees the identification of a path from the starting location to the desired destination by guiding the hovercraft towards the target while evading obstacles until it successfully arrives at the required place. Particles can undergo various alterations through self-perception to enable rapid identification and clever exploitation [19].

The following equations are utilized to obtain the proposed reference control actions of starboard and portboard forces in Eq. (18) and Eq. (19) by solving Eq. (4), Eq. (5), and Eq. (6), as follows:

$$F_{Sref} = 0.5 \left(M\dot{u}_s - Mv_s r_s + \left(\frac{J}{L}\right) \dot{r}_s \right) \quad (18)$$

$$F_{Pref} = 0.5 \left(M\dot{u}_s - Mv_s r_s - \left(\frac{J}{L}\right) \dot{r}_s \right) \quad (19)$$

3.2.1. Feedback neural network radial basis function (RBF) controller design

The feedback controller plays a critical role in maintaining the trajectory tracking error of the hovercraft system when its location deviates from the reference path during the transient state. Furthermore, this controller identifies the most efficient fan forces that minimize the feedback position error. The controller receives inputs in the form of feedback from the starboard and portboard forces of the hovercraft, together with configuration errors denoted as ex and ey , as well as the $e\psi$, which represents the error between the desired and the actual routes of the hovercraft. The calculation of this error can be performed using the rotation matrix in Eq. (20) [23].

$$\begin{bmatrix} ex \\ ey \\ e\psi \end{bmatrix} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y_r - y \\ \psi_r - \psi \end{bmatrix} \quad (20)$$

The feedback control mechanism employed in this work is based on the neural network radial basis function (NNRBF). In this regard, RBF networks differ from other neural networks in their ability to provide a universal approximation, and they exhibit a higher learning speed. An RBF network can be classified as a feedforward neural network [22,24]. The proposed approach integrates the principles of conventional neural networks and radial basis functions, resulting in robust capabilities for approximation and efficient training. The structure of the proposed neural network controller is illustrated in Fig. 4. The proposed controller architecture comprises two networks for starboard and portboard forces, which are subsequently combined to generate feedback forces. The architectural design of the network comprises three unique layers, specifically the input layer, the hidden layer, and the output layer [25].

The input layer receives input data and transmits it to the hidden layer. It consists of four input

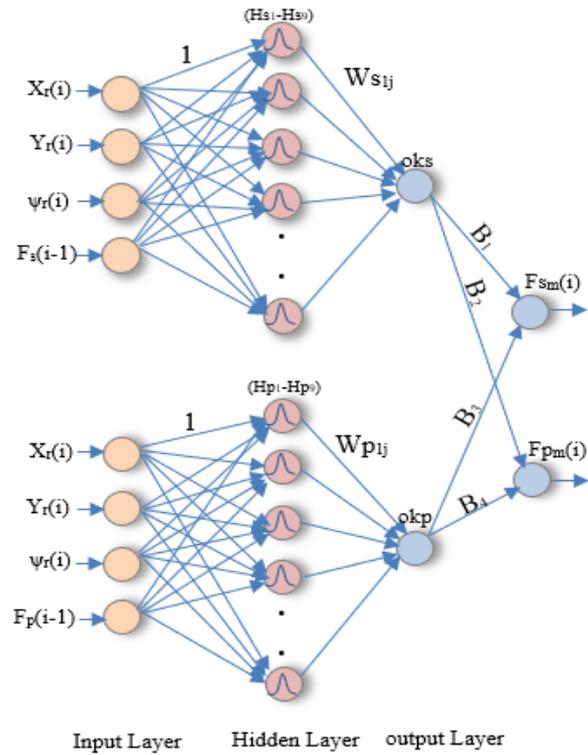


Figure. 4 The proposed RBF neural network controller design

neurons for each network to receive the current location, orientation, and prior inputs of starboard and portboard forces $G(i)=(Xr(i), Yr(i), \psi_r(i), Fs(i-1), Fp(i-1))$, as illustrated in Fig. 4. There are no weights in this layer. The hidden layer is where the computation occurs, and it consists of nine neurons Hs_j and Hp_j for each network with nonlinear activation functions. The RBF uses the Gaussian basis function as an activation function, as shown in Eq. (21) and Eq. (22) [22-25].

$$H_{Sj} = ae^{-\left(\frac{\sum(G_i-c)^2}{r^2}\right)} \quad (21)$$

$$H_{Pj} = ae^{-\left(\frac{\sum(G_i-c)^2}{r^2}\right)} \quad (22)$$

where c is the center of the geometric shape of the Gaussian functions of the neurons, a and r are the maximum amplitude and the width of the geometric shape of the Gaussian functions of neurons, respectively, and G_i represents the network inputs (position and feedback forces).

The output layer is made up of a single linear neuron for each network. This neuron is responsible for computing the weighted sum oks and okp of their inputs, as expressed in Eq. (23) and Eq. (24) [26].

$$oks = \sum_{j=1}^9 Hs_j \times Ws_{1j} \quad (23)$$

$$okp = \sum_{j=1}^9 Hp_j \times Wp_{1j} \quad (24)$$

The variables Ws and Wp denote the weights between the hidden and the output layers. Then, oks and okp are composed for representing the proposed feedback forces control action (Fs_m and Fp_m), as given in Eq. (25) and Eq. (26).

$$Fs_m(i) = oks \times B_1 + okp \times B_3 \quad (25)$$

$$Fp_m(i) = oks \times B_2 + okp \times B_4 \quad (26)$$

where $B_1, B_2, B_3,$ and B_4 are variable weights.

The control gain parameters of the feedback NNRF controller can be found and tuned by the off-line GWO algorithm in order to obtain the optimal or near-optimal forces' control actions (Fs_m and Fp_m) for the hovercraft model to keep the position and the orientation of the hovercraft in the desired path equation.

The Grey Wolf Optimization (GWO) meta-heuristic optimization algorithm was introduced by Mirjalili et al. in 2014 [27]. This algorithm draws inspiration from the distinctive social behaviour and leadership hierarchy observed in grey wolves. Grey wolves have four types: alpha (α), beta (β), and delta (δ). The alpha wolves set hunting, sleeping, and waking times, in addition to group management and hunting choices. The beta wolves assist the alphas in decision-making and can replace them. The deltas are the last to eat and hunt, following alpha and beta wolves. A delta wolf is needed to prevent group turmoil [27, 28].

The GWO strategy consists of three primary stages: searching for the prey, encircling the prey, and attacking the prey. The optimization process in this method is guided by the $\alpha, \beta,$ and δ . α represents the best solution, β represents the second best solution, and δ represents the third best option. These three wolves are followed by the remaining wolves [28].

The algorithm commences by the initialization of $\alpha, \beta,$ and δ , followed by the randomization of the weight population. Subsequently, the algorithm proceeds to compute the mean square error and the fitness of the current weights based on Eq. (27) [23].

$$\left. \begin{aligned} MSE &= 0.5 \times \sum ((Fs_{ref(i)} - Fs_{m(i)})^2 \\ &+ Fp_{ref(i)} - Fp_{m(i)})^2) \\ fitness &= \frac{1}{1+MSE} \end{aligned} \right\} \quad (27)$$

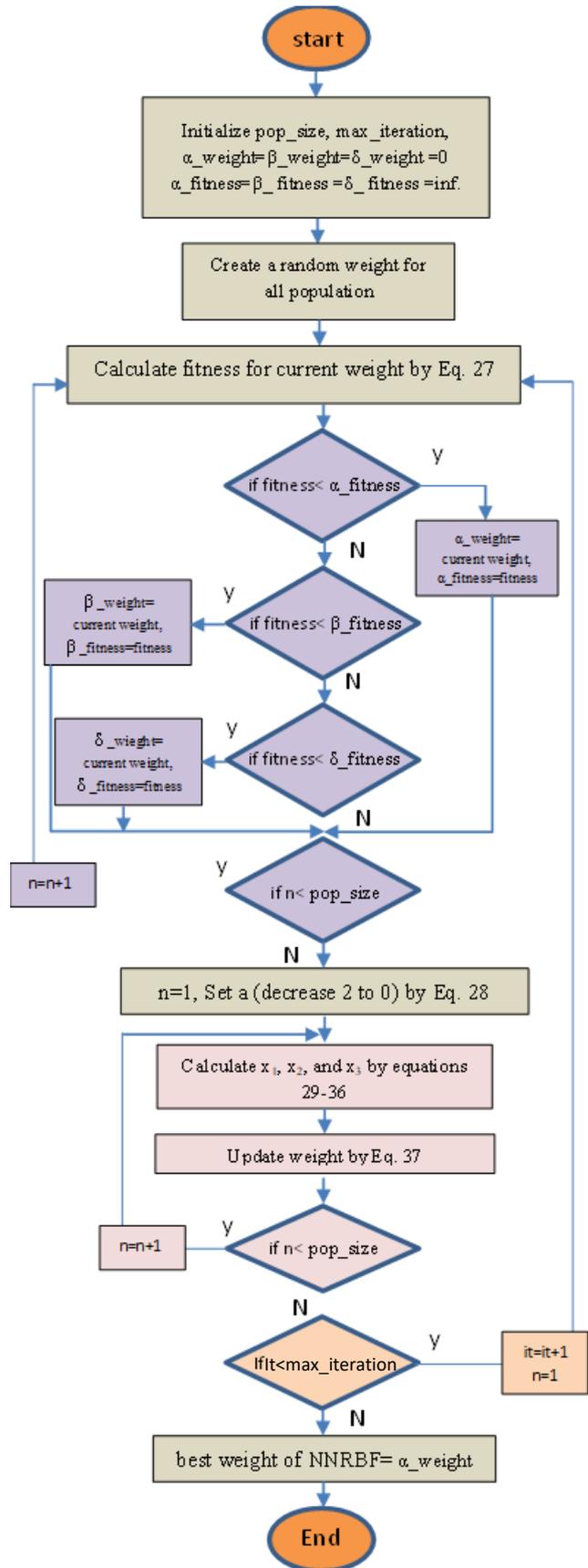


Figure. 5 Flowchart for tuning the weights of NNRF based on the GWO algorithm

The values of α_weight , β_weight , and δ_weight can be determined based on the estimated fitness of the weights. For the purpose of updating the weight for all populations, it is necessary to calculate the values of x_1 , x_2 , and x_3 for each population using the following equations [29]:

$$a = 2 - 1\left(\frac{2}{\text{No.of iterations}}\right) \quad (28)$$

$$A = 2a \times r_1 - a \quad (29)$$

$$c = 2 \times r_2 \quad (30)$$

$$D_\alpha = |c \times \alpha_weight - weight| \quad (31)$$

$$x_1 = \alpha_weight - A \times D_\alpha \quad (32)$$

$$D_\beta = |c \times \beta_weight - weight| \quad (33)$$

$$x_2 = \beta_weight - A \times D_\beta \quad (34)$$

$$D_\delta = |c \times \delta_weight - weight| \quad (35)$$

$$x_3 = \delta_weight - A \times D_\delta \quad (36)$$

Where the value of variable a decreases linearly from 2 to 0 with iterations.

It is employed to approach the solution range. The variables r_1 and r_2 represent random vectors within the range of (0,1).

According to Eq. (37), each weight can be updated as follows:

$$weight = \frac{x_1+x_2+x_3}{3} \quad (37)$$

The variables x_1 , x_2 , and x_3 denote the varying degrees of deviation of the current weight from the alpha (best), beta (second best), and delta (third best) values, respectively. Upon completion of all iterations, the optimal solution is denoted as α_weight [29].

The flowchart illustrating the method is depicted in Fig. 5 for tuning the parameters of the NNRBF controller.

4. Simulation results

In order to optimize the navigation process of the hovercraft operating within a workspace measuring [500 x 500] cm and containing stationary obstacles, as seen in Fig. 6, the initial phase involves generating a desired path equation that successfully travels toward the designated target while avoiding

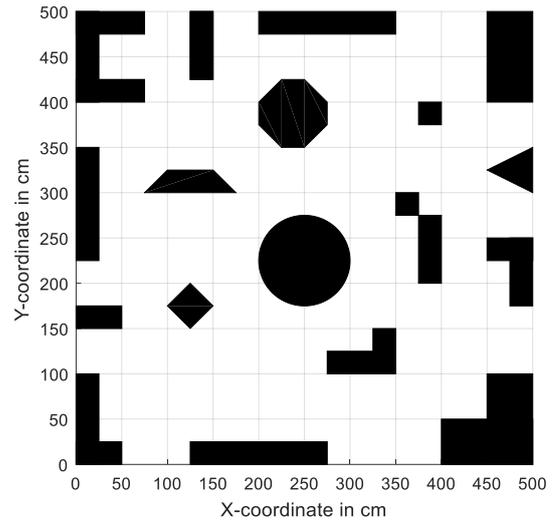


Figure. 6 The proposed environment

Table 4. Hovercraft specifications [18]

Specifications	Values
Width	35.6 cm
Height	18.1 cm
Depth	25.4 cm
Mass	5.15 kg
Viscous coefficient	4.5 kg/sec
Rotational friction	0.41 kg m/sec
The starboard and portboard fans' forces	5.5 N
Rotational inertia	0.047 kg m ²
Forces' Arm	0.123 m

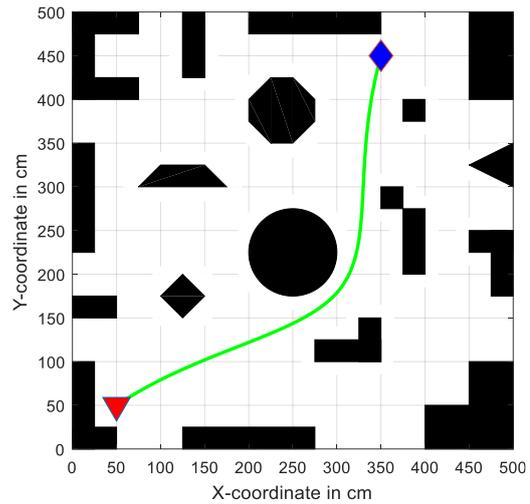
any potential impediments as much as possible to ensure a minimum distance to the target achieving the smoothest path.

The second phase entails the identification of an appropriate control scheme for the dynamic hovercraft model. The model is an under-actuated system, which is illustrated in Fig. 1, comprising two inputs, namely the forces exerted by the starboard and the portboard fans, and three outputs that indicate the position (x and y) and orientation. Consequently, it can be classified as a multi-input multi-output (MIMO) system. The model demonstrates pronounced nonlinearity and time-varying behaviour. This paper focuses on the hovercraft specifications shown in Table 4.

The MATLAB 2022a package was utilized, incorporating computer hardware characteristics, including an Intel Core i5-5200U processor with 8.00 GB of RAM and a CPU operating at a frequency of 2.20GHz. The hybrid ABC-SPPSO algorithm was implemented to satisfy the path finding criteria.

Table 5. The proposed parameters of ABC-SPPSO.

Symbols	Value
Scout bees numbers	10
Selected bees numbers	5
Recruit bees numbers equals the population size of SPPSO	40
Particle numbers	5
The fittest bees numbers	5
W	0.7
N_{pop}	40
Particle position (x,y)	5
c_1, c_2	1.37, 1.37
r_1, r_2	Random values between (0,1)
It_{Max}	50



(a)

Table 6. Hovercraft path length and iteration number for two cases

Case study	Best path length	No. of Iterations
Case 1	555.36cm	11
Case 2	390.46cm	10

The proposed hybrid ABC-SPPSO algorithm can determine the shortest path for the hovercraft in the given environment. The proposed parameters of the hybrid ABC-SPPSO algorithm are demonstrated in Table 5.

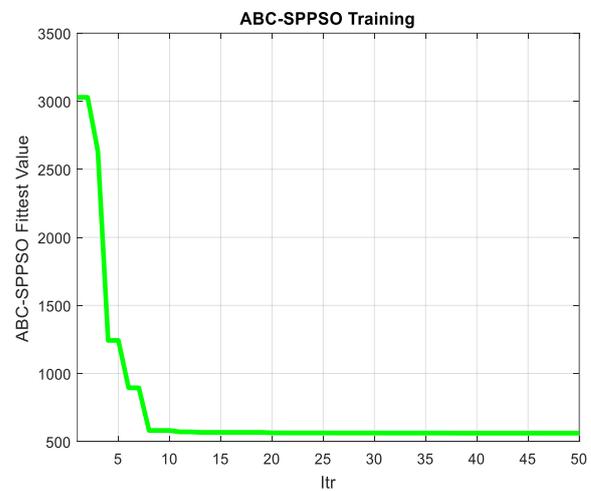
This work considers two scenarios involving a hovercraft. In the first scenario, the initial position of the hovercraft is indicated by a red triangle at coordinates (50, 50) cm. The blue diamond, which represents the goal point, is situated at coordinates (350, 450) cm. Fig. 7 represents the optimal or the near-optimal path with the minimum cost function for case 1.

In case 2, the initial position of the hovercraft is at the coordinates (50, 200) cm, as indicated by the red triangle. The target position, denoted by a blue diamond, is situated at coordinates (300, 450) cm.

Therefore, the proposed hybrid ABC-SPPSO algorithm is applied to find the shortest path for the hovercraft in the given environment. Fig. 8 represents the optimal or the near-optimal path with the minimum cost function for case 2.

Table 6 displays the shortest path for each hovercraft case, along with the corresponding number of iterations required to obtain this solution.

The optimal desired paths for the hovercraft in the two scenarios, following the implementation of the ABC-SPPSO method, are represented by the intended paths of Eq. (38) and Eq. (39) using the fitting function.



(b)

Figure. 7 Case 1 using ABC-SPPSO method for path 1: (a) path finding and (b) cost function

$$Y_r(X_r) = 1.4305 \times 10^{-11} \times X_r^6 - 1.3195 \times 10^{-8} \times X_r^5 + 4.6824 \times 10^{-6} \times X_r^4 - 0.0008 \times X_r^3 + 0.067 \times X_r^2 - 1.9213 \times X_r + 40 \quad (38)$$

$$Y_r(X_r) = 3.2290 \times 10^{-11} X_r^6 - 3.1256 \times 10^{-8} \times X_r^5 + 1.2014 \times 10^{-5} \times X_r^4 - 0.002326 \times X_r^3 + 0.23663 \times X_r^2 - 11.193756 \times X_r + 394.64 \quad (39)$$

These paths are used to train the proposed path finding controller, as shown in Fig. 2. The proposed feedback NNRBF controller, depicted in Fig. 4, is trained to perform the function of a numerical inverse dynamic trajectory tracking controller for the hovercraft. Therefore, we generate the reference starboard and portboard fan forces using the proposed Eq. (18) and Eq. (19).

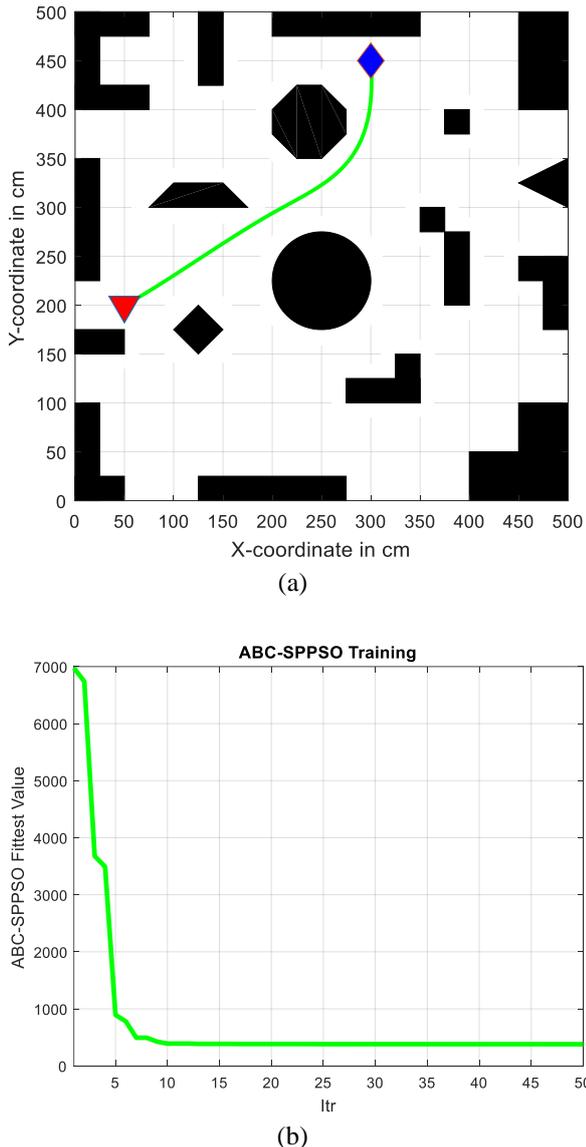


Figure. 8 Case 2 using ABC-SPPSO method for path 2: (a) path finding and (b) cost function.

The best-proposed values of the GWO algorithm in terms of the agent number are 50 and the maximum number of iterations is 100. After 100 iterations of the GWO technique, the suggested controller successfully generated starboard and portboard fan forces that matched the hovercraft's reference forces, as shown in Fig. 9 (a) and (b).

Subsequently, in order to ascertain the absence of the over-learning issue in the proposed feedback NNRBF controller, a test was conducted using different data to evaluate the performance of the controller model in generating the forces exerted by the starboard and portboard fans for the hovercraft. This evaluation is conducted without encountering the over-learning problem, as depicted in Fig. 10 (a) and (b).

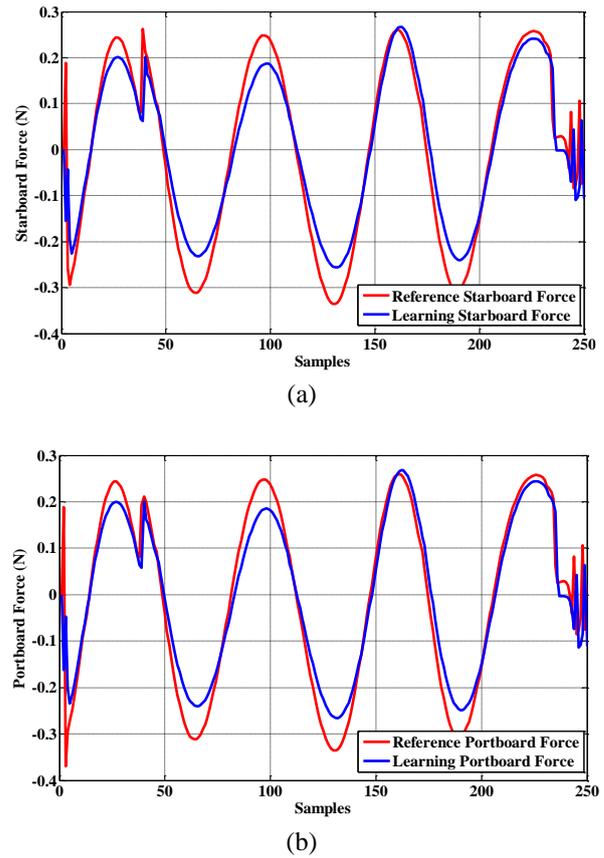


Figure. 9 Learning process result fans' forces: (a) starboard and (b) portboard

Fig. 11 shows the convergence curve performance of implementing the GWO method in the training process. Fig. 12 depicts the controller's learning performance response for the hovercraft with two fan forces.

The response is dependent on the learning data, which is the optimal reference path for the hovercraft. This decision is made by considering Eq. (27) as well as the numerical inverse dynamic equations that generate the reference forces.

Based on Eq. (27), the MSE reaches a tiny value of less than 0.001, as shown in Fig. 12, and there are no learning concerns, such as overfitting or overlearning.

Fig. 13 depicts the proposed NNRBF controller's learning accuracy using Eq. (40), which reaches 99.98% at 90 iterations.

$$Accuracy = \left(1 - \left(\frac{MSE}{100}\right)\right) \times 10 \quad (40)$$

The controller that has been proposed is now prepared to trace various types of desired pathways.

In order to evaluate the efficacy of the proposed NNRBF controller in trajectory tracking for

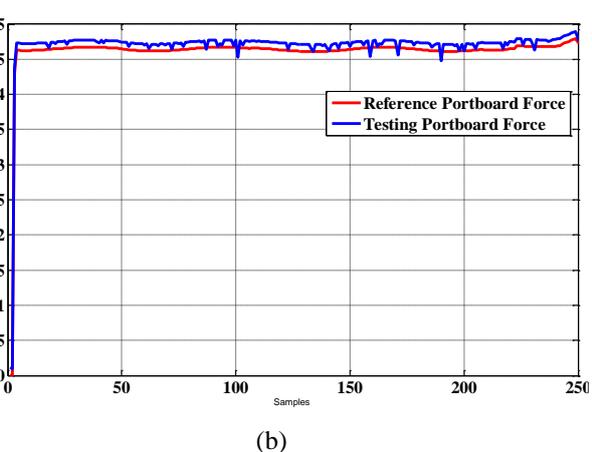
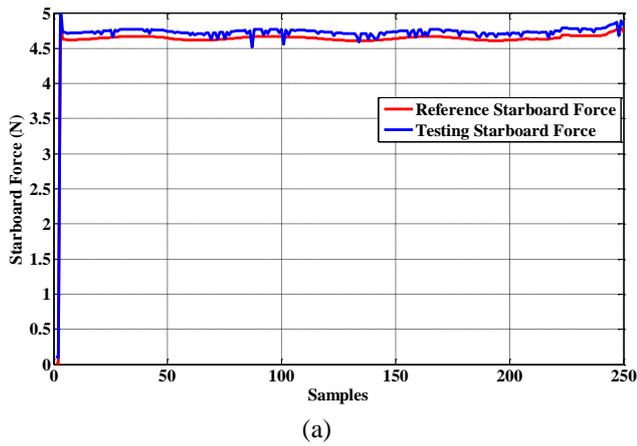


Figure. 10 Fans' forces of the testing process: (a) starboard and (b) portboard

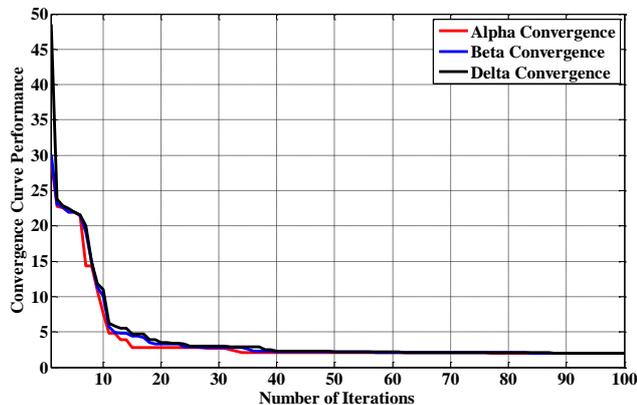


Figure. 11 Convergence curve performance of implementing GWO

hovercraft scenarios, Fig. 14 presents a two-dimensional simulation of the intended route for case 1, as determined by Eq. (38) and the actual output of the dynamic hovercraft model. The observed performance of the hovercraft exhibits rapidity and stability throughout a total of 300 samples, where the sampling time is taken to be 0.1 seconds. Fig. 15 (a) shows the output response of

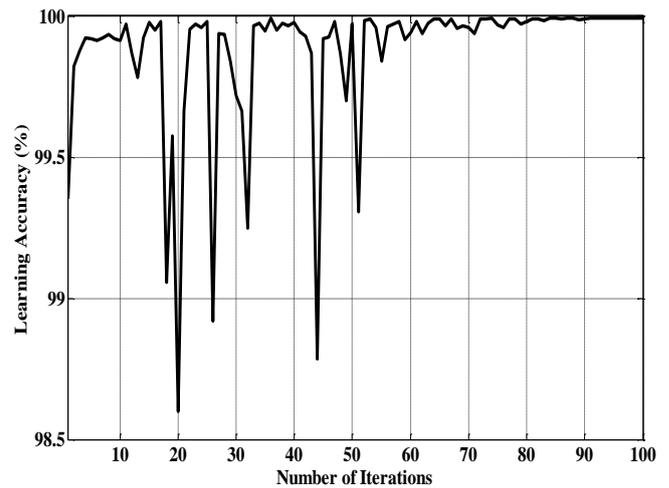


Figure. 12 Learning performance response of the proposed controller

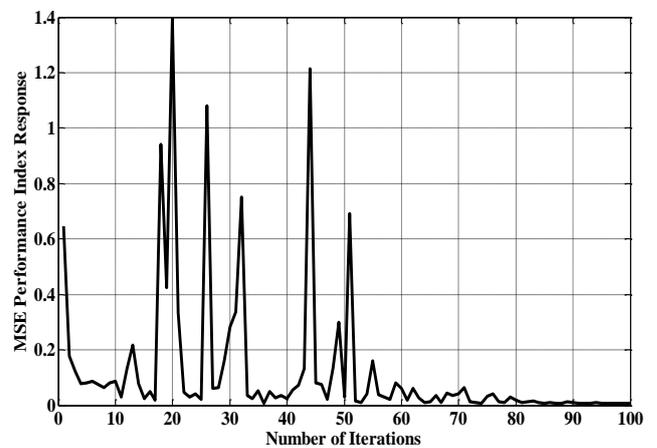


Figure. 13 Learning accuracy of the proposed controller

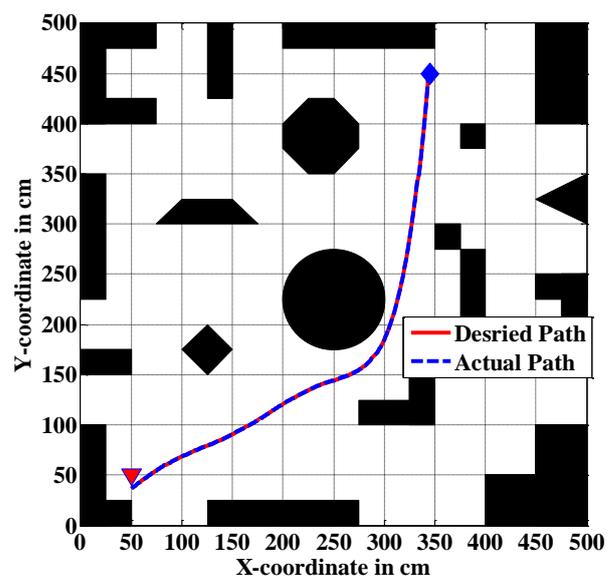


Figure. 14 Simulation result of the actual and the desired paths of the hovercraft for case 1

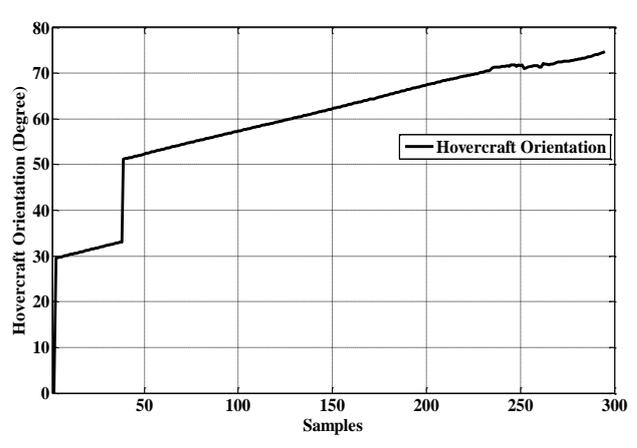
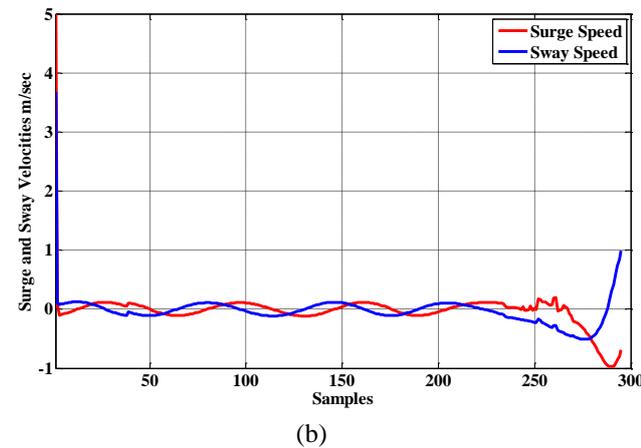
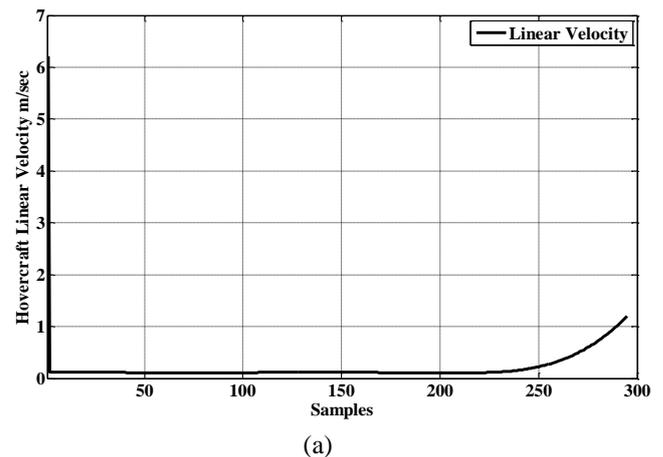
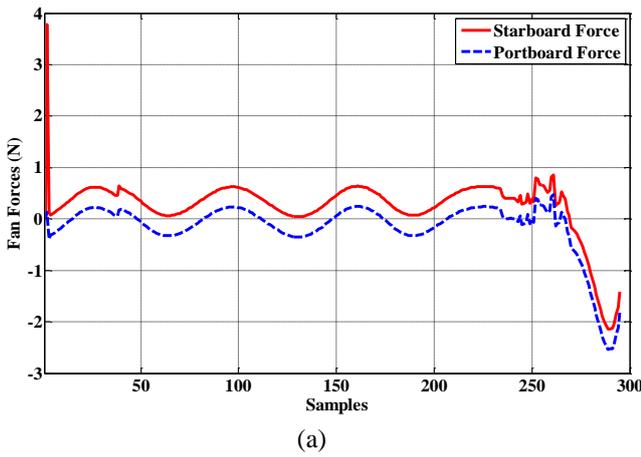


Figure. 15 Simulation result: (a) starboard and portboard forces of the fans and (b) surge and sway speeds

Figure. 16 Simulation result: (a) linear velocity and (b) hovercraft orientation

the proposed path-finding controller, demonstrating quick and smooth control of the hovercraft's starboard and portboard forces.

On the other hand, Fig. 15 (b) illustrates the surge and sway speeds. There is no saturation state, allowing for successful path tracking. Fig. 16 (a) and (b) demonstrate the best response of the hovercraft's linear velocity and orientation for case 1.

For case 2, the hovercraft's actual output and the desired path, which are identical and stable for 250 samples, are shown in Fig. 17.

The suggested path-finding controller generates rapid and smooth control responses for the hovercraft's starboard and portboard fans, as shown in Fig. 18 (a). On the other hand, Fig. 18 (b) shows the sway and surge speeds. The best response of the hovercraft's linear velocity and orientation for case 2 is shown in Fig. 19 (a) and (b).

In order to validate the efficacy of this proposed controller scheme in trajectory tracking for the hovercraft, we conducted a comparative analysis between the numerical simulation results of the proposed controller and those of alternative controllers. This analysis focused on measuring the

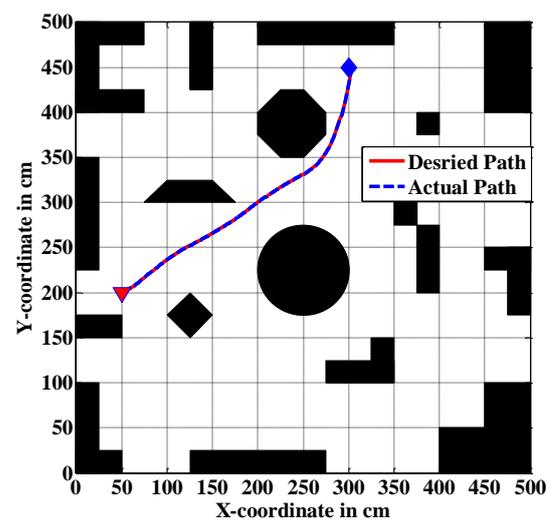


Figure. 17 Simulation result of the actual and the desired paths of the hovercraft for case 2

maximum improvement achieved in the hovercraft's tracking error in both the X-axis and the Y-axis positions. Initially, the suggested approach was

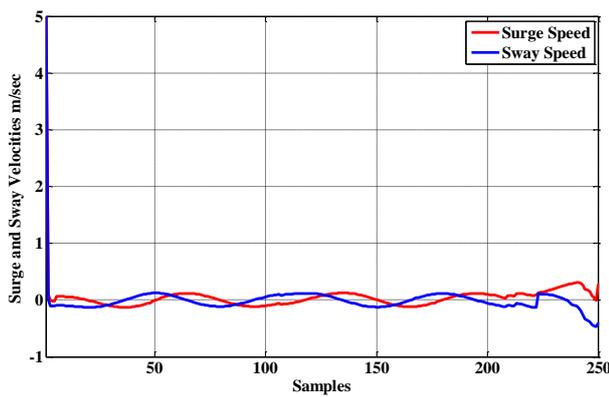
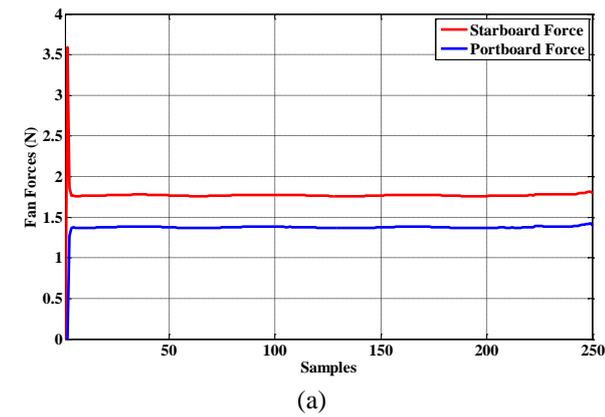


Figure. 18 Simulation result: (a) starboard and portboard forces of the fans and (b) surge and sway speeds

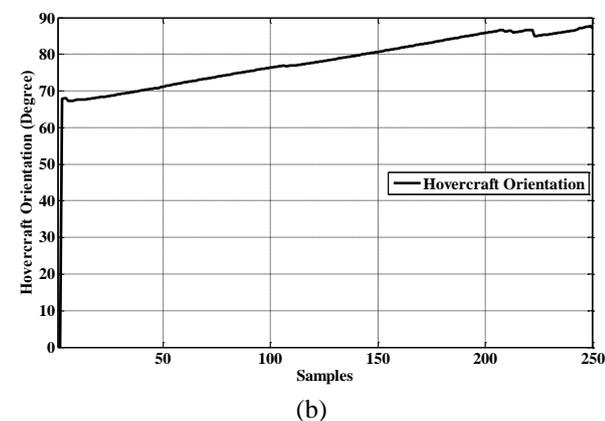
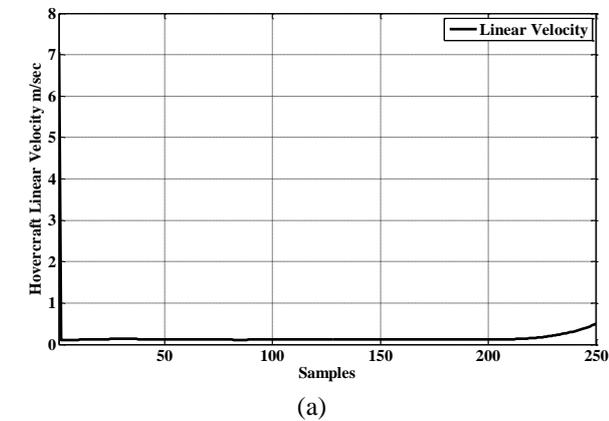


Figure. 19 Simulation result: (a) linear velocity and (b) hovercraft orientation

compared to the study conducted in [9], which introduced an improved quasi-velocity (IQV) controller for controlling the hovercraft system to follow two specified pathways.

Subsequently, terminal sliding mode control (TSMC) was employed for the same objective. In

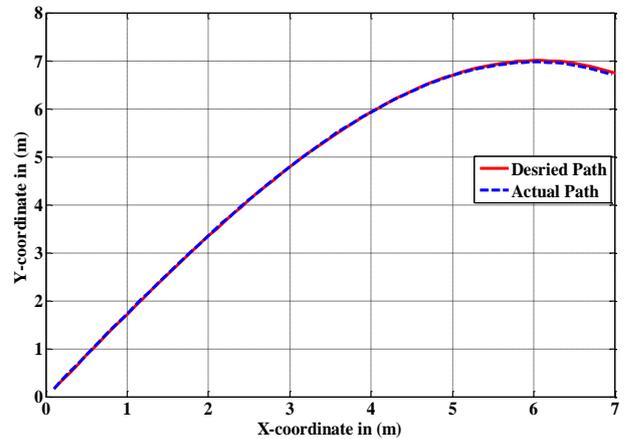


Figure. 20 Simulation result of the actual and the desired paths of the hovercraft for path1 of [9]

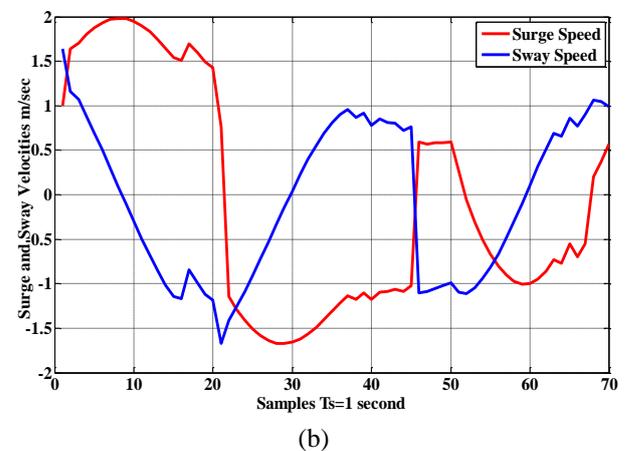
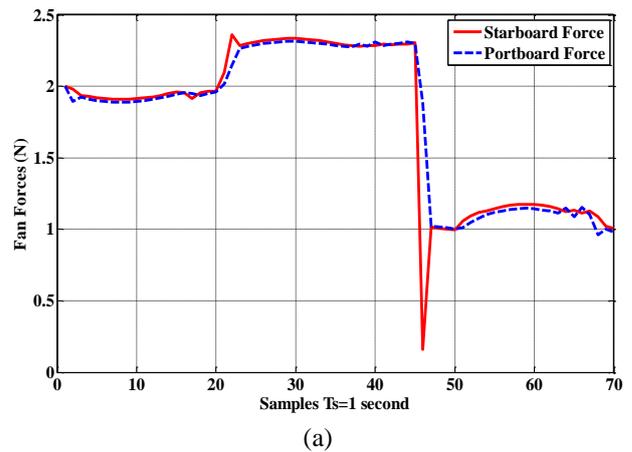


Figure. 21 Simulation result: (a) starboard and portboard forces and (b) surge and sway speeds

their study, the researchers in [9] conducted experiments in a fixed environment that included a workspace measuring [600 x 700] cm. The reference path equation for path1 and path2 was derived from the fitting function and is represented by Eq. (41) and Eq. (42), respectively.

$$Y_r(X_r) = -0.01543 \times X_r^3 - 0.00389 \times X_r^2 + 1.74804 \times X_r - 0.01083 \quad (41)$$

$$Y_r(X_r) = 0.03928 \times X_r^4 - 0.30311 \times X_r^3 + 0.14653 \times X_r^2 + 2.72180 \times X_r - 0.00350 \quad (42)$$

Following that, we applied the reference path equation that was obtained to the proposed path finding controller depicted in Fig. 2. Fig. 20 illustrates the discrepancy between the intended trajectory and the trajectory produced by the dynamic hovercraft model for path 1 based on Eq. (41).

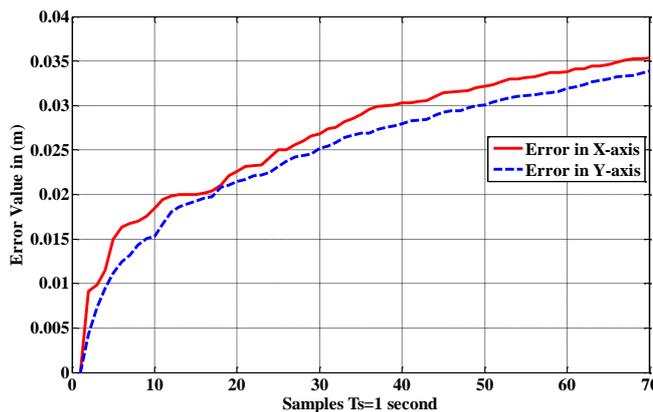


Figure. 22 Tracking error in x position and in y position for path1 of [9]

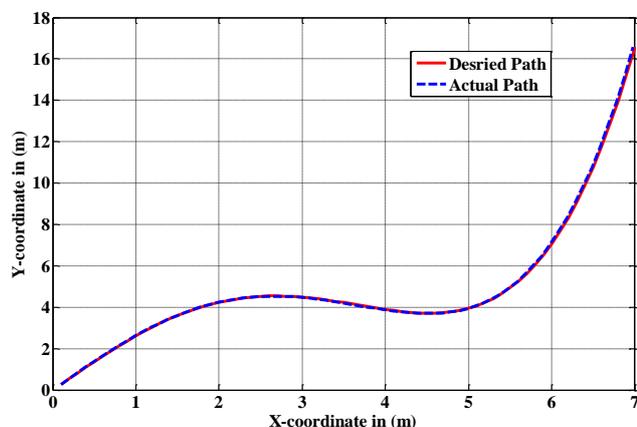


Figure. 23 Simulation result of the actual and the desired paths of the hovercraft for path 2 of [9]

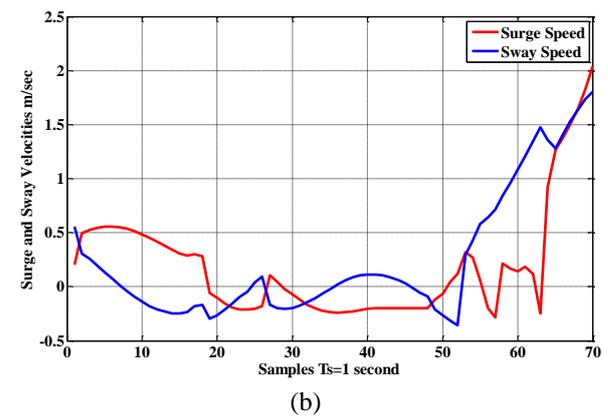
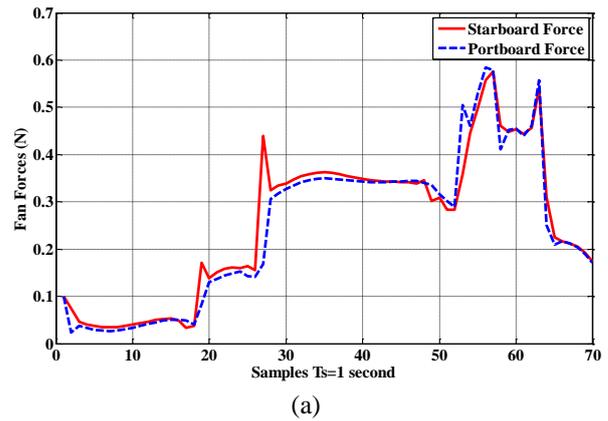


Figure. 24 Simulation result: (a) starboard and portboard forces and (b) surge and sway speeds

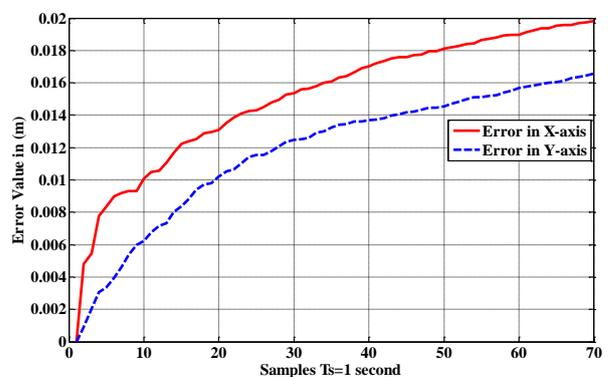


Figure. 25 Tracking error in x position and in y position for path 2 of [9]

The suggested controller generates rapid and smooth control responses for the hovercraft's starboard and portboard fans, as shown in Fig. 21 (a), whereas, Fig. 21 (b) shows the sway and surge speeds.

The hovercraft's current performance is characterized by high speed and the absence of oscillation throughout a span of 70 samples. A representation of the summing error in the x-position and in the y-position can be found in Fig.

22. The position error is increased after sample 50 because the proposed controller was trained for the desired map that has a size of 500×500 cm. Thus, after sample 50, the size of the map is greater than the desired map.

Fig. 23 illustrates the difference between the reference path and the actual path of the hovercraft for path 2 based on Eq. (42). As shown in Fig. 24 (a), the suggested controller produces a quick and smooth control action for the fans on the starboard and portboard of the hovercraft. The sway and surge speeds are shown in Fig. 24 (b).

The extent of the summing error in the x-position and the y-position are illustrated in Fig. 25 for path2 that was taken from [9]. The position error is increased after sample 55 because the proposed controller was trained using the desired map that has a size of 500×500 cm. Therefore, after sample 55, the size of the map is greater than the desired map.

As can be seen in Tables 7 and 8, the results of the simulations conducted in this particular instance demonstrate that the proposed controller generates a smaller tracking error in comparison to both the IQV controller and the TSMC controller in [9]. The reasons for the enhancement of our proposed controller over the two controllers in [9] are that the proposed path finding controller was trained off-line using the intelligent optimization algorithm of the GWO algorithm in order to find the optimal control gain parameters of the proposed NNRF controller, and it was tuned on-line in order to generate the optimal or near-optimal starboard and portboard forces' control action that moves the hovercraft in the desired path equation with a minimum tracking error in the x-axis and in the y-axis. However, the controller in [9] has limitations in the values of the control parameters.

Table 7. Tracking error result of the comparison with IQV

Axis tracking error (cm)	IQV [9]	The proposed controller	Enhancement (%)
Error in x-position in path 1	4.5	3.51	22%
Error in y-position in path 1	3.98	3.39	14.8%
Error in x-position in path 2	7.2	2.0	72.2%
Error in y-position in path 2	3.71	1.65	55.5%

Table 8. Tracking error result of the comparison with TSMC

Axis tracking error (cm)	TSMC [9]	The proposed controller	Enhancement (%)
Error in x-position in path 1	7.13	3.51	50.7%
Error in y-position in path 1	9.55	3.39	64.5%
Error in x-position in path 2	2.6	2.0	23%
Error in y-position in path 2	4.87	1.65	66.1%

Specifically, there are twelve parameters of the control law and they are initialized by the trial-and-error method. They do not use any intelligent algorithm but depend on the experience of the authors in order to stabilize the system based on the Lyapunov method. These values lead to an error in the start of the hovercraft motion in the x- and the y-axes, and there is a high fluctuation in the response of the control action, which leads to a tracking error in the position of the hovercraft during the motion.

Furthermore, we conducted a comparison between the suggested methodology and the study conducted in [7], which addresses the nonlinear control problem by treating it as a cascade control problem. This study employs a stationary setting with a workspace measuring [400×400] cm for a circular trajectory (path 1) and [600×400] cm for a complex trajectory (path 2).

The fitting function was used to obtain the reference path equations for paths 1 and 2 of reference [7], which are represented by Eq. (43) and Eq. (44), respectively.

$$\left. \begin{aligned} X_r(i) &= 2 \times \sin(i/10) \\ Y_r(i) &= 2 \times \cos(i/10) \end{aligned} \right\} \quad (43)$$

$$\left. \begin{aligned} X_r(i) &= -3 \times \sin(4 \times \pi \times i/300) \\ Y_r(i) &= 2 \times \sin(4 \times \pi \times i/200) \end{aligned} \right\} \quad (44)$$

After that, we utilized the reference paths equation that was automatically developed on the proposed controller depicted in Fig. 2.

Fig. 26 illustrates the differences between the intended trajectory and the real trajectory produced by the dynamic hovercraft model for path 1.

Fig. 27 (a) shows that the suggested path-finding controller generates quickly and smoothly the two

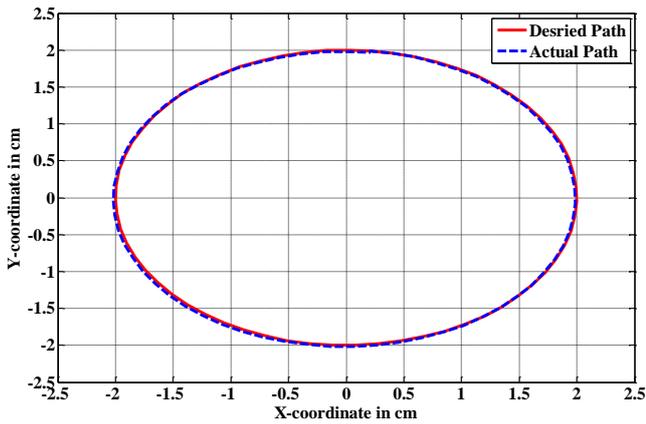
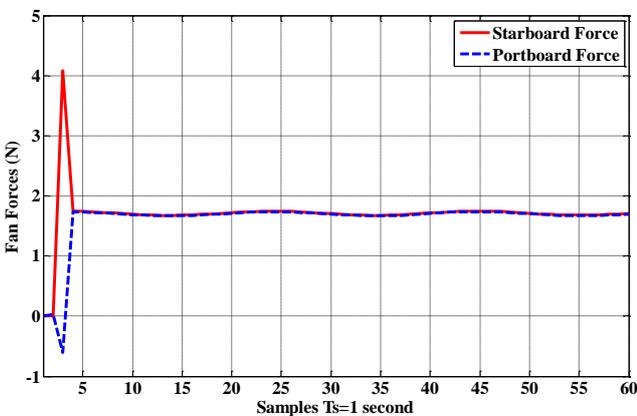
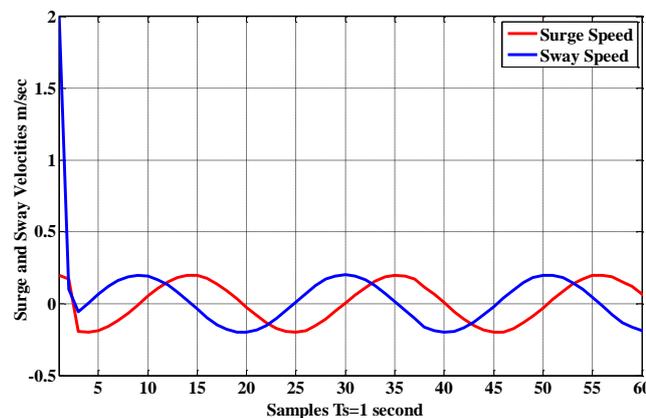


Figure. 26 Simulation result of the actual and the desired paths of the hovercraft for path1 of [7]



(a)



(b)

Figure. 27 Simulation result: (a) starboard and portboard forces and (b) surge and sway speeds

control signals of the starboard and the portboard forces for the fans on the hovercraft. Fig. 27 (b) shows the sway and surge speeds. The hovercraft's performance is characterized by high speed and smooth operation, with no oscillation observed during the sampling process. Fig. 28 depicts the error in the x and the y positions.

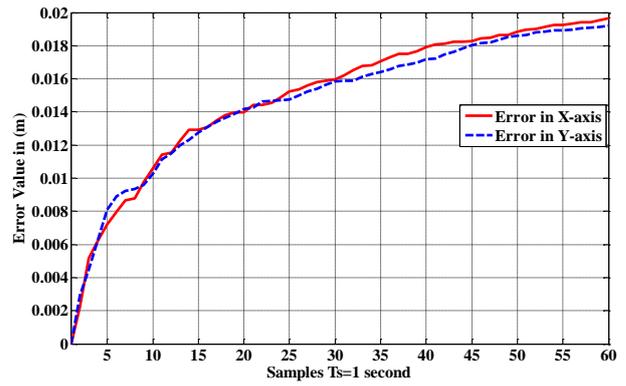


Figure. 28 Tracking error in x position and y positions for path1 of [7]

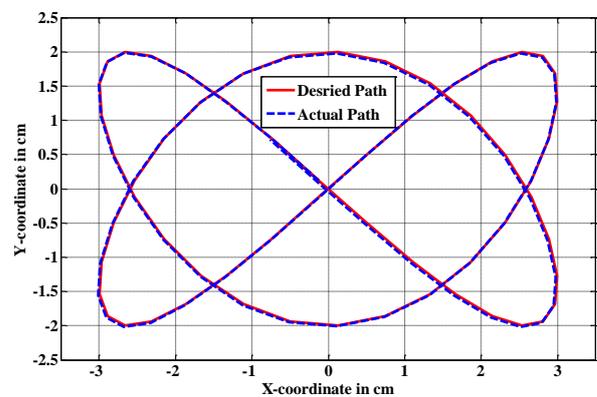


Figure. 29 Simulation result of the actual and the desired paths of the hovercraft for path 2 of [7]

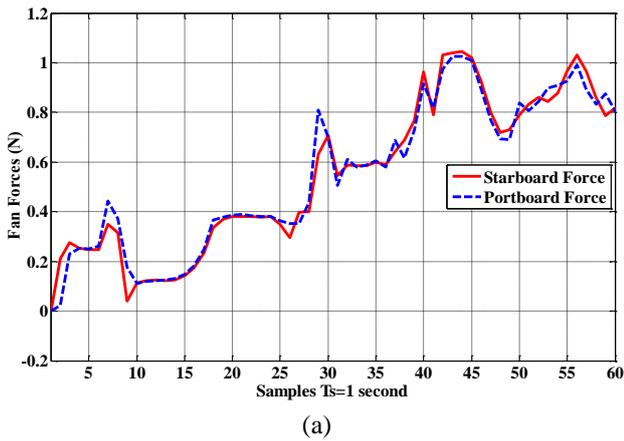
Fig. 29 shows the discrepancy between the reference path and the actual path for the hovercraft on path 2 that was taken from [7].

Fig. 30 (a) shows that the suggested controller controls the starboard and portboard forces of the fans on the hovercraft quickly and smoothly. Fig. 30 (b) shows the sway and surge speeds. Fig. 31 shows the magnitude of the x- and y-axes errors.

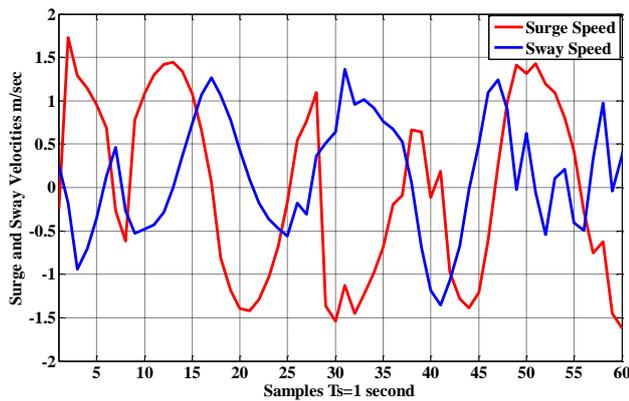
The results of the simulations conducted in this particular instance, as shown in Table 9, indicate that the suggested controller produces a lower tracking error compared to that of the nonlinear controller in [7] for the two paths.

The nonlinear controller in [7] consists of two controllers, the first is the position controller that depends on the inverse kinematics of the hovercraft model and the second is the force selection technique that depends on nine cases of the forces based on if-then control actions with four control gains selected using the trail-and-error method to stabilize the control law.

However, these values of the parameters are not the optimal values and the nine cases of the forces selection did not cover all the region of the



(a)



(b)

Figure. 30 Simulation result: (a) starboard and portboard forces and (b) surge and sway speeds

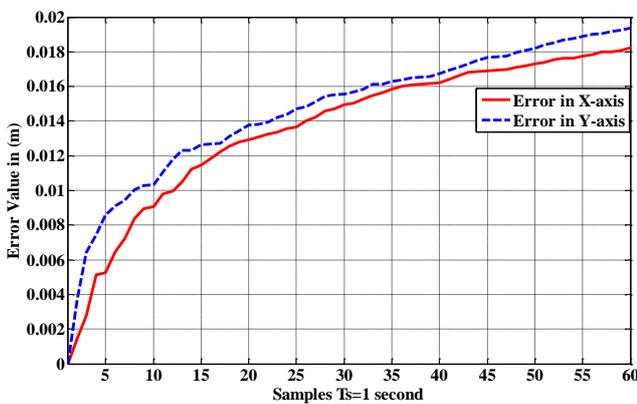


Figure. 31 Tracking error in x position and y position for path2 of [7]

hovercraft platform movement in the environments. Therefore, there are errors in the x-axis and the y-axis of the hovercraft during motion. In contrast, the proposed path-finding controller was trained off-line using the intelligent optimization algorithm of the GWO algorithm to find the optimal control gain parameters of the proposed NNRBF controller. It was then tuned on-line to generate the optimal or the

Table 9. Tracking error result of the comparison

Axis tracking error (cm)	Nonlinear controller in [7]	The proposed controller	Enhancement (%)
Error in x-position in path 1	2.58	1.91	25.9%
Error in y-position in path 1	2.81	1.88	33%
Error in x-position in path 2	7.73	1.82	76.4%
Error in y-position in path 2	8.53	1.92	77.5%

near-optimal starboard and portboard forces control action that moves the hovercraft in the desired path equation with the least amount of tracking error in the x- and y-axes. These are the reasons that our proposed controller is superior to the controller in [7].

Thirdly, we performed a comprehensive analysis comparing the proposed methodology with the research undertaken in [8], which focuses on guiding and controlling algorithms for an autonomous hovercraft. The methods utilize line-of sight guidance and neural network-based adaptive dynamic inversion control to accurately follow waypoints. This work employs a stationary setting with a workspace measuring [700×450] cm. Fig. 32 shows the difference between the reference path and the actual path for the hovercraft.

Fig. 33 (a) shows that the suggested path-finding controller controls the starboard and portboard forces of the fans on the hovercraft quickly and smoothly. In addition, Fig. 33(b) shows the sway and surge speeds and Fig. 34 shows the

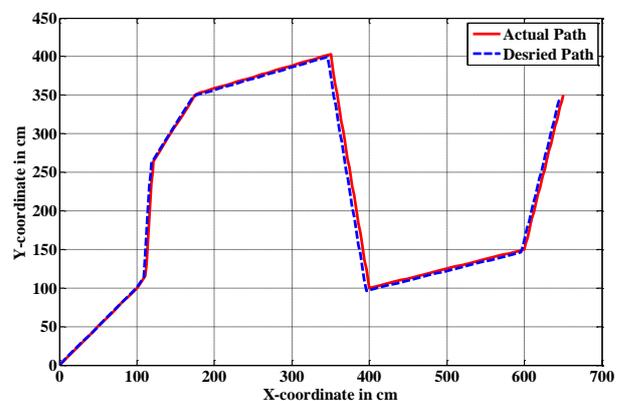


Figure. 32 Simulation result of the actual and the desired paths of the hovercraft of [8]

Table 10. Tracking error result of the comparison

Axis tracking error (cm)	The controller in [8]	The proposed controller	Enhancement (%)
Error in x-position	9.2	4.5	51%
Error in y-position	7.3	4.2	42.4%

trajectory tracking of the x- and the y-axes errors during the hovercraft motion. The results of the simulations conducted in this particular instance, as shown in Table 10, indicate that the suggested controller produces a lower tracking error compared to that of the controller in [8].

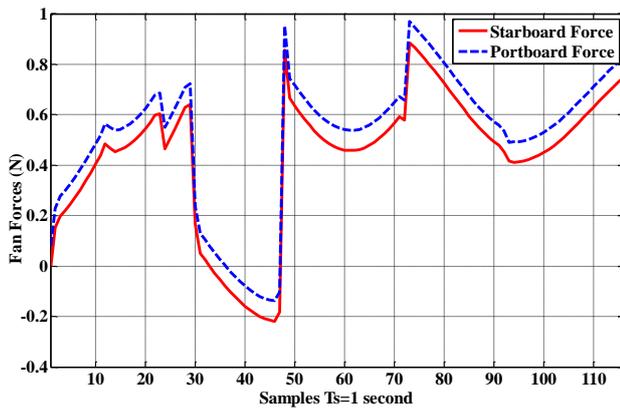
The reason that our proposed controller is better than the controller in [8] is that it was trained offline using the intelligent optimization algorithm of the GWO algorithm to determine the ideal control gain parameters of the proposed NNRF controller. It was then tuned online to produce the best or the near-best starboard and portboard forces control action that moves the hovercraft in the desired path equation with the least amount of tracking error in the x- and y-axes.

In contrast, the controller in [8] has limitations in the values of the control parameters. In particular, there are five parameters of the control law and they are initialized by the trial-and-error method. They do not use any intelligent algorithm and depend on the experience of the authors to stabilize the system based on the Lyapunov method. These values lead to an error in each segment of the desired path of the hovercraft motion in the x- and the y-axes, and there is a high oscillation in the response of the kinematic velocities' control action, leading to a high tracking error in the position of the hovercraft during motion.

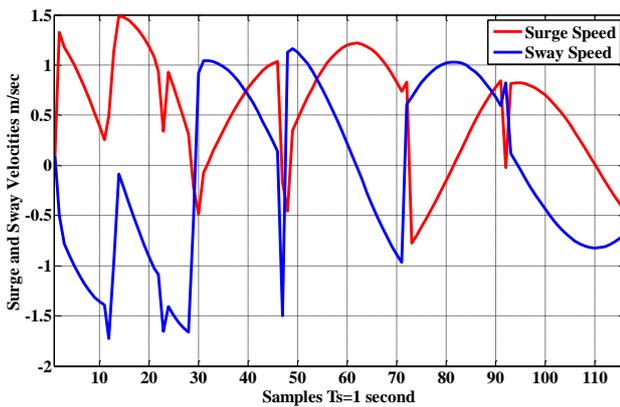
5. Conclusions

This research presented a design of the path-finding controller for trajectory tracking of the hovercraft system. The controller is based on a feedforward numerical inverse dynamic controller and a feedback neural network radial basis function controller with the GWO algorithm to find and tune the optimal or near-optimal control gain parameters. The purpose of this design is to follow the intended path equations of the hovercraft system. The pathways were constructed using the hybrid artificial bee colony self-perception particle swarm optimization algorithm, which is designed to discover the shortest path while avoiding collisions.

Specifically, this research aimed to address two issues encountered throughout the navigation procedure. The initial step involved generating an ideal or nearly optimal intended trajectory that fulfils three criteria: unhindered navigation, minimizing the distance to the goal, and establishing the most seamless path for a hovercraft operating in a global environment. The path planning problem has been resolved with the suggested ABC-SPPSO technique.



(a)



(b)

Figure. 33 Simulation result: (a) starboard and portboard forces and (b) surge and sway speeds

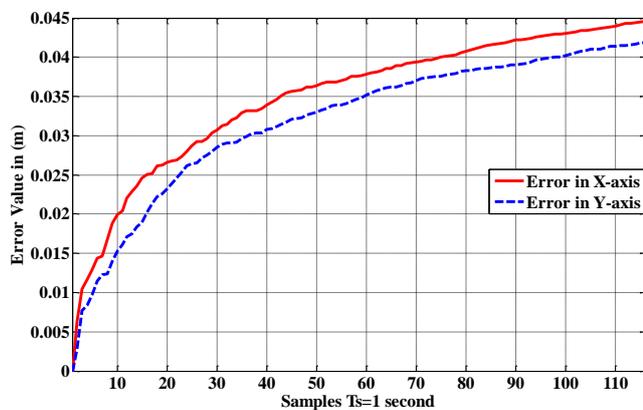


Figure. 34 Tracking error in x position and y position for the path of [8]

The subsequent issue in our research pertains to the development of a motion controller for the trajectory tracking of the hovercraft. Our objective is to guarantee that the hovercraft adheres to the predetermined path with minimal slipping and errors in position and orientation. To accomplish this objective, we have proposed a neural network radial basis function controller that effectively reduces tracking errors along the X-axis and the Y-axis. Furthermore, the error in the orientation is nearly negligible. The suggested controller accurately determines and delivers the optimal and seamless magnitudes of the forces exerted by the starboard and portboard fans, based on the intelligent controller using the GWO algorithm that generated the optimal control actions for the dynamic hovercraft platform model.

Consequently, the hovercrafts were highly proficient at accurately following the intended trajectory and reaching the target destination without any oscillation. In order to assess the efficacy of the suggested controller, a comparative analysis has been carried out to evaluate the maximum tracking errors in both the X-position and the Y-position. This analysis takes into account the work of other researchers who have employed different types of controllers.

In addition, the recommended controller was assessed in comparison to the IQV and TSMC controllers in [9]. The results of the comparison demonstrated that the proposed controller improves the tracking error rate on the X-position by 22% and on the Y-position by 14.8% in comparison with the results of the IQV controller in [9]. Compared to the results of the TSMC controller in [9], the proposed controller improves the tracking error rate on the X-position by 50.7% and on the Y-position by 64.5% because the controller in [9] has limitations in the values of the twelve control parameters of the control law that were initialized by the trial-and-error method. Moreover, the authors in [9] did not use any intelligent algorithm to find the optimal value for these parameters to stabilize the hovercraft system and to reduce the error in the start of the hovercraft motion in the x- and the y-axes.

Furthermore, the recommended controller was compared to the nonlinear cascade controller in [7], and the analysis showed that the suggested controller improves the tracking error rates for the X-position and the Y-position by 25.9% and 33%, respectively. In particular, the nonlinear controller in [7] has four parameters. However, these parameters' values are not the optimal values and the forces selection of the controller has only nine cases that did not cover all the region of the hovercraft

platform movement in the environments. Therefore, there are errors in the x-axis and the y-axis of the hovercraft during motion. Ultimately, the proposed controller was assessed in comparison to the neural network-based adaptive dynamic inversion controller in [8]. It improves the tracking error rate for the X-position by 51% and for the Y-position by 42.4% because the controller in [8] has limitations in the values of the control parameters that were initialized by the trial-and-error method, without the utilization of an intelligent algorithm. Particularly, the authors relied on their experience to stabilize the system.

For future endeavours, we recommend the practical implementation of the experimental work on the path finding algorithm and the proposed control method for the hovercraft system based on the development of an FPGA board.

Conflicts of Interest

The authors assert that they have no conflict of interest.

Author Contributions

Sura Muhi Hussein and Ahmed Sabah Al-Araji enhanced and developed a trajectory tracking controller for controlling the hovercraft model. Sura Muhi Hussein described the suggested neural network controller that relies on the radial basis function. Ahmed Sabah Al-Araji provided a detailed explanation of the kinematics and dynamics of the hovercraft. Furthermore, Sura Muhi Hussein and Ahmed Sabah Al-Araji analyzed the outcomes of the simulation in this research.

References

- [1] F. Rehman, and A. Mahmood, "Adaptive Sliding Mode-based Full-State Stabilization Control of an Under-Actuated Hovercraft", *International Journal of Dynamics and Control*, Vol.12, No.1, pp.1512-1521, 2024.
- [2] W. Retnaraj, S. Sengupta, T. Q. Dinh, and J. J. Chong, "Modeling and Feedback Linearization of an Electric Hovercraft for Path Tracking", In: *Proc. of International Conf. On Mechatronics Technology (ICMT)*, Kaohsiung, Taiwan, pp.1-6, 2022.
- [3] M. S. Pavăl, A. Popescu, T. Popescu, D. Zahariea, and D. E. Husaru, "Numerical Study on the Movement of Air Inside the Inner Cavity of a Hovercraft Model", *IOP Conference Series: Materials Science and Engineering*, Vol.444, No.8, pp.1-10, 2018.

- [4] N. Niazy, A. El-Sawy, and M. Gadallah, "Solving Capacitated Vehicle Routing Problem Using Chicken Swarm Optimization with Genetic Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol.13, No.5, pp.502-513, 2020, doi: 10.22266/ijies2020.1031.44.
- [5] A. A. Rasheed, A. S. Al-Araji, and M. N. Abdullah, "Static and Dynamic Path Planning Algorithms Design for a Wheeled Mobile Robot Based on a Hybrid Technique", *International Journal of Intelligent Engineering and Systems*, Vol.15, No.4, pp.167-181, 2022, doi: 10.22266/ijies2022.0831.16.
- [6] A. A. Kareem, B. K. Oleiwi, and M. J. Mohamed, "Planning the Optimal 3D Quadcopter Trajectory Using a Delivery System-Based Hybrid Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol.16, No.2, pp.427-439, 2023, doi: 10.22266/ijies2023.0430.34.
- [7] D. Chaos, D. Moreno-Salinas, R. Muñoz-Mansilla, and J. Aranda, "Nonlinear Control for Trajectory Tracking of a Nonholonomic RC-Hovercraft with Discrete Inputs", *Mathematical Problems in Engineering*, Vol.2013, No.1, pp.1-16, 2013.
- [8] K. Kim, Y. K. Lee, S. Oh, D. Moroniti, D. Mavris, G. J. Vachtsevanos, N. Papamarkos, and G. Georgoulas, "Guidance Navigation and Control of an Unmanned Hovercraft", In: *Proc. of International Conf. on Control and Automation, Platania, Greece*, pp.380-387, 2013.
- [9] P. Herman, "Controller for an Asymmetric Under-actuated Hovercraft in Terms of Quasi-Velocities", *Applied Sciences*, Vol.13, No.8, pp.1-20, 2023.
- [10] G. Rigatos, and G. Raffo, "Nonlinear Control of the Under-Actuated Hovercraft using the Derivative-Free Nonlinear Kalman Filter", In: *Proc. of International Conf. on Computational Intelligence (UKCI)*, Bradford, UK, pp.1-7, 2014.
- [11] M. Fu, and Q. Wang, "Safety-Guaranteed, Robust, Nonlinear, Path-Following Control of the Under-Actuated Hovercraft Based on FTESO", *Journal of Marine Science and Engineering*, Vol.11, No.6, pp.1-21, 2023.
- [12] W. Xie, D. Cabecinhas, R. Cunha, and C. Silvestre, "Global Practical Tracking for a Hovercraft with Unmeasured Linear Velocity and Disturbances", *IFAC-Papers OnLine*, Vol.53, No.2, pp.8959-8964, 2020.
- [13] L. Pröhl, and H. Aschemann, "Nonlinear Observer-Based Control of an Under-Actuated Hovercraft Vehicle", *IFAC-Papers OnLine*, Vol.53, No.2, pp.8973-8978, 2020.
- [14] Z. A. Ali, and B. Jabeen, "Prototyping Non-holonomic Hovercraft for Path Planning and Obstacle Avoidance", *Sir Syed University Research Journal of Engineering & Technology*, Vol.9, No.1, pp.7-12, 2019.
- [15] D. Lu, W. Xie, D. Cabecinhas, R. Cunha, and C. Silvestre, "Path Following Controller Design for an Under-actuated Hovercraft with External Disturbances", In: *Proc. of International Conf. On Control, Automation and Systems (ICCAS)*, Jeju, Korea, pp.76-81, 2019.
- [16] D. Cabecinhas, P. Batista, P. Oliveira, and C. Silvestre, "Hovercraft Control With Dynamic Parameters Identification", *IEEE Transactions on Control Systems Technology*, Vol.26, No.3, pp.785-796, 2017.
- [17] H. Karami, and R. Ghasemi, "Adaptive Neural Observer-Based Nonsingular Super-Twisting Terminal Sliding-Mode Controller Design for a Class of Hovercraft Nonlinear Systems", *Journal of Marine Science and Application*, Vol.20, No.2, pp.325-332, 2021.
- [18] A. P. Aguiar, L. Cremean, and J. P. Hespanha, "Position Tracking for a Nonlinear Under-Actuated Hovercraft: Design and Experimental Results", In: *Proc. of International Conf. on Decision and Control*, Maui, USA, pp.3858-3863, 2003.
- [19] S. M. Hussein, and A. S. Al-Araji, "Enhancement of a Path-Finding Algorithm for the Hovercraft System Based on Intelligent Hybrid Stochastic Methods", *International Journal of Intelligent Engineering and Systems*, Vol.17, No.2, pp.346-364, 2024, doi: 10.22266/ijies2022.0630.14.
- [20] K. E. Dagher, and M. N. Abdullah, "Airborne Computer System Based Collision-Free Flight Path Finding Strategy Design for Drone Model", *International Journal of Intelligent Engineering and Systems*, Vol.14, No.6, pp.234-248, 2021, doi: 10.22266/ijies2021.1231.22.
- [21] A. S. Alaraji, A. K. Ahmed, and M. K. Hamzah, "Development of a Path Planning Algorithms and Controller Design for Mobile Robot", In: *Proc. of International Conf. on Electrical Engineering, (SCEE 2018)*, Baghdad, Iraq, pp.72-77, 2018.
- [22] A. S. Al-Araji, K. E. Dagher, and M. N. Abdullah, "Development of Intelligent Control Strategy for an Anesthesia System based on

- Radial Basis Function Neural Network Like PID Controller”, *International Journal of Intelligent Engineering and Systems*, Vol.17, No.3, pp.622-634, 2024, doi: 10.22266/ijies2024.0630.48.
- [23] A. S. Alaraji, "Development of Kinematic Path-Tracking Controller Design for Real Mobile Robot via Back-Stepping Slice Genetic Robust Algorithm Technique", *Arabian Journal for Science and Engineering*, Vol.39, No.12, pp.8825-8835, 2014.
- [24] M. Cao, X. Zhou, and Y. Ju, "Robot Motion Planning Based on Improved RRT Algorithm and RBF Neural Network Sliding”, *IEEE Access*, Vol.11, No.1, pp.592-600, 2023.
- [25] S. Malaysha, M. Awad, and R. Hadrob, "Classification and Prediction of Low-Density Lipoprotein Cholesterol LDL-C in The Palestinian Patients Using Machine Learning Techniques”, *International Journal of Intelligent Engineering and Systems*, Vol.15, No.1, pp.453-463, 2022, doi: 10.22266/ijies2022.0228.41.
- [26] A. S. alaraji, M. F. Abbod, and H. S. Alraweshidy, "Design of a Neural Predictive Controller for Nonholonomic Mobile Robot based on Posture Identifier", In: *Proc. of International Conf. on Intelligent Systems and Control, (ISC 2011)*, Cambridge, United Kingdom, pp.198-207, 2011.
- [27] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer”, *Advances in Engineering Software*, Vol.69, No.1, pp.46-61, 2014.
- [28] Y. Li, X. Lin, and J. Liu, "An Improved Grey Wolf Optimization Algorithm to Solve Engineering Problems”, *Sustainability*, Vol.13, No.6, pp.1-23, 2021.
- [29] K. E. Dagher, and J. Hagege, "Enhancement of the Blood Glucose Level for Diabetic Patients based on an Adaptive Auto-Tuned PID Controller via Meta-Heuristic Methods”, *International Journal of Intelligent Engineering and Systems*, Vol.17, No.3, pp.250-261, 2024, doi: 10.22266/ijies2024.0630.21.