An Improved IEHO Super-Twisting Sliding Mode Control Algorithm for Trajectory Tracking of a Mobile Robot

Ahmed OULTILIGH[1]*, Hassan AYAD[1], Abdeljalil EL KARI[1], Mostafa MJAHED[2], Nada EL GMILI[3], Ernő HORVÁTH[4], Claudiu POZNA[5]

[1] Applied Physics Department, Cadi Ayyad University, Marrakesh 40000, Morocco
a.oultiligh@gmail.com (*Corresponding author*), ayad.ha@gmail.com, jalil.elkari@gmail.com

[2] Mathematics and Systems Department, Royal School of Aeronautics, Marrakesh 40000, Morocco
mjahed.mostafa@gmail.com

[3] Electrical Engineering Department, University of Hassan II, Mohammedia 28806, Morocco
elgmilinada@gmail.com

[4] Vehicle Industry Research Center, Széchenyi István University of Győr, 1 Egyetem, Győr, Hungary
herno@ga.sze.hu

[5] Department of Automation, Transilvania University of Brasov, 29 Eroilor Boulevard, Brasov, Romania
claudiu.radu63@gmail.com

Abstract: In recent years, trajectory tracking of a mobile robot has been one of the most addressed problems in the specilized literature, as a mobile robot must have the ability to follow a trajectory, while also compensating various external and internal disturbances. This paper proposes an IEHO-STSM controller based on the super-twisting sliding mode for the path tracking of a mobile robot. First, a new improved IEHO algorithm has been developed and introduced, based on the EHO (Elephant Herding Optimization) metaheuristic algorithm. The developed algorithm consisted in improving the performance of the basic EHO such as convergence speed, exploration and exploitation capabilities. Then, based on a dynamic model of the mobile robot, a super-twisting sliding mode (STSM) controller was designed to guide the robot to the desired trajectory. Finally, the improved IEHO algorithm was applied for adjusting the parameters of the super-twisting sliding mode (STSM) controller. The analysis of the proposed IEHO algorithm has been done by comparing it with EHO, PSO (Particle Swarm Optimization) and GWO (Grey Wolf Optimizer) algorithms, by employing it in tuning the STSM. The simulation results show that the proposed IEHO-STSM can reach both high precision and high speed capability, by overcoming external and internal disturbances.

Keywords: EHO, GWO, IEHO, PSO, Sliding mode, Super twisting, IEHO-STSM.

1. Introduction

In the last two decades, the continued improvement in the performance of mobile robots has greatly expanded their application areas and they have come to be used in agriculture, industry, transportation, health care, building structure, and other sectors (Fabregas et al., 2019). Mobile robots are non-linear, time-varying, and highly coupled dynamic systems, among them, wheeled mobile robots (WMR). This kind of system is a dynamic device designed with non-holonomic constraints. The use of mobile robots in various fields of application increases the need for precise controller systems for the control of robot motion. Indeed, it is extremely difficult for a mobile robot to compensate for several disturbances and follow a pre-planned trajectory in different environments (Bessas et al., 2016; Matraji et al., 2018). This problem is the title of several research works, in which researchers propose different solutions based on the mobile robot's kinematics using different types of control, as well as the application of optimization methods to adjust and search the optimal parameters of such control to meet the requirements of precise control.

Trajectory tracking is one of the most difficult problems in mobile robotics. This problem consists in driving the robot along a predefined trajectory according to a pre-calculated speed in order to reach the robot's destination in a short time (Oultiligh et al., 2021). The non-holonomic characteristics, internal dynamics, and external disturbances of WMRs lead to different types of uncertainty that make it more challenging to propose a perfect tracking strategy, especially over long distances or in complex environments (Shijin & Udayakumar, 2017).

In the specialized literature, the developed controllers vary the linear and angular velocities of the robot to reduce the distance error which allows it to accurately follow the desired trajectory. In (Besseghieur et al., 2018; Obaid et al., 2016), researchers propose classical controllers based on the backstepping method by working on a non-holonomic wheeled mobile robot. The researchers Cui et al. (2015) combined an adaptive controller based on H_∞ with the backstepping technique to assure the stabilization and the convergence of the tracking error under bounded perturbations.

https://doi.org/10.24846/v33i1y202405

Among the methods used in the control of mobile robots, there is the sliding mode control (SMC). In the research literature, this controller is known for its robustness against external disturbances, and uncertainties in the dynamic model (Binh et al., 2017). Nikranjbar et al. (2018) presented an application of the artificial potential fields in adapting a sliding mode control of a mobile robot in a dynamic environment.

In (Qiu et al., 2019), the authors proposed a non-compulsive, non-singular sliding mode controller for mobile robots. The results illustrated that the stability of the controller was high with a small tracking error. Similarly, in (Mirzaeinejad, 2019), an optimization-based nonlinear controller applied to nonholonomic wheeled mobile robots was presented, and a sliding mode controller has been proposed (Goswami & Padhy, 2018) for trajectory tracking in the presence of disturbance and sensor measurement noise. Also, a backstepping global fast terminal sliding mode controller was designed in (Truong et al., 2021) for finite-time tracking control of non-holonomic systems. In the same context, the authors Chen et al. (2020) designed a second-order sliding mode controller (SOSM) to avoid the chattering problem of the classical sliding mode control (SMC). The controller was used to solve the path-following problem for four-wheeled electric vehicles, taking into account the modeling errors and the complexity of driving scenarios.

For the same reasons, an adaptative super-twisting sliding mode control has been developed (Liu et al., 2020). In the sliding mode control, a major drawback lies in the selection of parameters that leads to the use of large forces in the reaching law which is necessary to move the robots and causes, for example, large spending of energy. To avoid this problem, several solutions based on intelligence methods and algorithms have been suggested in the specialized literature. In (Vijay & Jena, 2017), a sliding mode control adapted by an artificial neuro-fuzzy system has been applied to improve the robustness of the controller against disturbances.

Another control based on SMC gain adaptation using artificial neural networks has been tested on a quadcopter with external disturbances in (Cibiraj & Varatharajan, 2017). An additional application of neural networks in continuous time has been developed in (Vázquez et al., 2016) for the path following of a two degrees of freedom (2-DOF) manipulator robot.

In (Aouf et al., 2017), the authors proposed an optimization of the gains of a PID controller using the particle swarm optimization (PSO) algorithm to reach the trajectory with a mobile robot. An adjustment of the PID controller gains for trajectory tracking of a quadcopter was presented in several works, using different optimization algorithms including, PSO-CS (particle swarm optimization-cuckoo search) (El Gmili et al., 2019), neural network, and fuzzy-logic (El Hamidi et al., 2019) and Genetic algorithm (Siti et al., 2019). Also, the authors Shirzadeh et al. (2021) developed a robust adaptive sliding mode neuro-fuzzy type 2 controller for trajectory tracking optimized by the ACO algorithm.

The authors Pozna et al. (2022) presented a hybridization between PSO and particle filter algorithms that can be applied for the tuning of several controller parameters which are applicable to trajectory tracking. In (Zhou et al., 2019), a sliding mode controller based on a hybrid grey wolf optimization was designed for robot trajectory tracking.

The aim of this paper is to introduce a sliding mode controller using the super-twisting algorithm to avoid the chattering phenomena. The proposed controller will be able to control a mobile robot by tracking a defined trajectory. During this work, an improvement of the basic algorithm EHO will be proposed. Then, the improved IEHO algorithm will increase the performance of EHO by enhancing its weaknesses, increasing its convergence speed, and its ability to find the optimal solution in a search space preventing any local minimum. After designing the controller based on the super-twisting sliding mode algorithm (STSM), the IEHO algorithm will be used in tuning the parameters of STSM. Indeed, intelligent tuning will ensure a better configuration of the STSM controller, and, consequently, a higher performance of the proposed IEHO-STSM controller for trajectory tracking of mobile robots.

The remainder of this paper is organized as follows. Section 2 introduces the elephant herding optimization algorithm (EHO). Section 3 describes the proposed improved IEHO algorithm. Section 4 presents the design of the super-twisting sliding mode control (STSM) for a unicycle mobile robot. Section 5, describes the adjustment process of the STSM controller by the proposed IEHO algorithm. Then, the results of the simulation are discussed in Section 6, and Section 7 concludes the paper.

2. Elephant Herding Optimization Algorithm

Elephant Herding Optimization (EHO), proposed by Wang et al. (2015) is an optimization algorithm that belongs to the group of swarm intelligence algorithms and its principle is based on the social behavior of elephants. Since 2015, EHO has been the subject of several works, including the research papers of El-Naggar et al. (2021), regarding the PID controller tuning, and of Li et al. (2020).

In general, the search for the optimal solution using optimization algorithms when solving such a problem requires two capabilities: the global search in which the algorithm explores the entire search space to avoid the local minimum, and the local search in which the algorithm searches for the optimal solution in small areas of the space. Elephant Herding Optimization is known for its ability to balance these two capabilities. Its behavior is presented as follows: a number k of clans compose a population of elephants, and each clan is guided by a leader elephant called the matriarch m_i. At each generation, a male elephant decides to leave its clan to live far from the other elephants (see Figure 1). The power of EHO lies in the representation of clans as a local search space, while the entire population represents a global search space. Thus, the elephants leaving their clans allow the algorithm to explore the search space. In terms of optimization, the fitness function has a better value presented by the matriarch and a worst value presented by the male elephants that move away from the clan. The procedure for implementing the EHO algorithm usually follows two steps:

- the clan updating operator;
- the separating operator.

2.1 The Clan Updating Operator

The clan updating is the first operator used in the EHO algorithm. Its principle is to update the position of each elephant $j(j=1,2,....,n)$ in the clan $c_i(i=1,2,....,k)$ using the following equation:

$$x_{new,c_i,j} = x_{c_i,j} + \alpha(m_i - x_{c_i,j})r \tag{1}$$

where $x_{new,c_i,j}$ and $x_{c_i,j}$ are the new and the old positions of the elephant j in the clan c_i, respectively; the matriarch m_i is the best solution in the clan c_i and r is a random number between 0 and 1.

According to the basic idea of EHO, equation (1) models the idea that each elephant follows its matriarch with an influencing factor $\alpha \in [0,1]$. The update of the position m_i is done using equation (2) when $x_{c_i,j}$ and m_i are two equal solutions:

$$x_{new,c_i,j} = \beta x_{center,c_i} \tag{2}$$

where $\beta \in [0,1]$ is an influencing factor; x_{center,c_i} the center of the clan c_i;$x_{new,c_i,j}$ is the new position of the matriarch. The x_{center,c_i} is calculated using equation (3) where n_{c_i} is the number of elephants in the clan c_i.

$$x_{center,c_i} = \frac{1}{n_{c_i}} \sum_{j=1}^{n_{c_i}} x_{c_i,j} \tag{3}$$

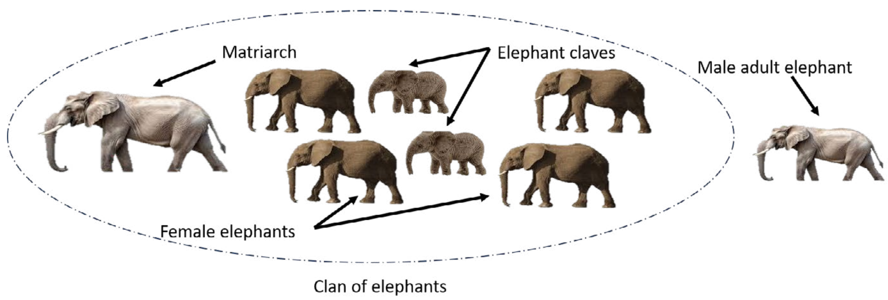


Figure 1. The social hierarchy of the EHO algorithm

2.2 The Separating Operator

The separating operator simulates the phenomenon created by the male elephants when they leave their clans. Indeed, at each generation, the male elephant or the solution with the worst fitness value x_{worst,c_i}, leaves its clan and gets replaced by another randomly generated elephant using equation (4).

$$x_{worst,c_i} = x_{\min} + (x_{\max} - x_{\min} + 1)r \quad (4)$$

where $r \in [0,1]$ is a uniform random distribution and $x_{\max}$ and $x_{\max}$ are the boundaries of the search space. The pseudocode of the EHO algorithm is shown in Table 1.

Table 1. The pseudocode of the EHO algorithm

EHO algorithm
Initialize the population
while (*t* < *MaxIt*)
Sort all the elephants by fitness
for all c_i **do**
for all elephants j in c_i **do**
Generate $x_{new,c_i,j}$ by Equation (1)
if $m_i = x_{c_i,j}$ **then** Generate $x_{new,c_i,j}$ by Equation (2)
end if; end for; end for;
for all c_i **do**
Replace x_{worst,c_i} in c_i by Equation (4)
end for
Evaluate the fitness
t=t+1
end while
Return the best solution m_i fou**nd.**

3. The Improved IEHO Algorithm

The objective of this section is to present the improved IEHO algorithm by developing the clan updating and separating operators of the EHO algorithm. The goal is to improve the exploration capacity and the convergence speed by proposing a good and reasonable updating process for the matriarch position.

The first improvement consists in updating the position of m_i according to the center of the clan x_{center,c_i}, its current position m_i, and the position of the best global solution m_g which presents the position of the best matriarch among all the matriarchs of the population. Equation (2) becomes:

$$x_{new,c_i,j} = m_i + \beta x_{center,c_i} + \Gamma(m_g - m_i) + R_0 r \quad (5)$$

$$\alpha = 2\frac{it}{MaxIt} \quad (6)$$

where $MaxIt$ is the iteration's maximum number and it is the current iteration.

One of the weaknesses of EHO is the replacement of the worst elephant leaving the clan. Indeed, replacing this solution with a random one can make the algorithm converge away from the optimum solution, which can directly influence its convergence speed. To control the procedure of the algorithm, it will be better to replace the worst solution with another most promising one.

The third improvement will be the changing of the replacement process of x_{worst,c_i} (equation (4)) by a new one described in equation (7). The update of x_{worst,c_i} is done by taking into consideration the position of the best global solution in all clans. Table 2 presents the pseudocode of the IEHO algorithm.

$$x_{worst,c_i} = x_{worst,c_i} + \Gamma(m_g - x_{worst,c_i}) + R_0 r \quad (7)$$

where x_{worst,c_i} is the solution with the worst fitness value in the clan; m_g the position of best matriarch among all the clans; $\Gamma \in [0,1]$ and R_0 are two influence factors; $r \in [0,1]$ is a uniform random distribution.

Table 2. The pseudocode of the IEHO algorithm

IEHO algorithm
Initialize the population
while (*t* < *MaxIt*)
Sort all the elephants by fitness
for all c_i **do**
for all elephants j in c_i **do**
Generate $x_{new,c_i,j}$ by Equation (1)
if $m_i = x_{c_i,j}$ **then** Generate $x_{new,c_i,j}$ by Equation (5)
end if; end for; end for;
for all c_i **do**
Replace x_{worst,c_i} in c_i by Equation (7)
end for
Evaluate the fitness
t=t+1
end while
Return m_g

4. Super-Twisting Sliding Mode Control Applied to a Unicycle Mobile Robot

4.1 Kinematic Control of a Unicycle Mobile Robot

The design of the unicycle mobile robot kinematic controller has been made based on the model shown in Figure 2. The kinematic model of the unicycle mobile robot is described as follows:

$$\dot{x}\begin{pmatrix}\dot{x}\\ \dot{y}\\ \dot{\psi}\end{pmatrix}=\begin{pmatrix}\cos\psi & -a\sin\psi\\ \sin\psi & a\cos\psi\\ 0 & 1\end{pmatrix}\begin{pmatrix}u\\ \omega\end{pmatrix} \tag{8}$$

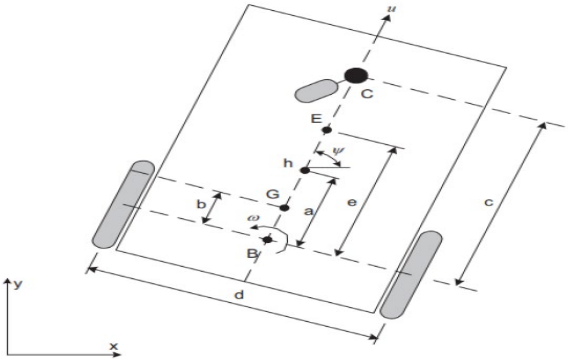


Figure 2. Unicycle mobile robot structure

Based on the work carried out by Martins et al. (2008), the proposed kinematic control law is defined as:

$$\begin{pmatrix}u_{ref}\\ \omega_{ref}\end{pmatrix}=\begin{pmatrix}\cos\psi & \sin\psi\\ -\frac{1}{a}\sin\psi & \frac{1}{a}\cos\psi\end{pmatrix} * \begin{pmatrix}\dot{x}_d+l_x\tanh(\frac{k_x}{l_x}\tilde{x})\\ \dot{y}_d+l_y\tanh(\frac{k_y}{l_y}\tilde{y})\end{pmatrix} \tag{9}$$

where u_{ref} and ω_{ref} are the references of the linear velocity and the angular velocity, respectively; a is the distance between the point of interest and the central point of the virtual axis linking the traction wheels (point B); $\tilde{x}=x_d-x$ and $k_x>0$ represent the position errors along the X and Y axis respectively; $k_x>0$ and $k_y>0$ are the controller gains; $l_y\in\mathbb{R}$ and $l_y\in\mathbb{R}$ are saturation constants; (x,y) and (x_d,y_d) are the current and desired coordinates of the robot position, respectively and $(\dot{x}_d,\dot{y}_d)$ its desired velocities. The objective of this type of controller is to provide linear and angular velocity references for the dynamic controller (Khai et al., 2020).

4.2 Super-Twisting Sliding Mode Controller Design

After choosing the kinematic control law u_{ref} and ω_{ref}, as in equation (9), it is necessary to ensure that the robot reaches those speeds. This can be done by accurately choosing the inputs of the τ_R and τ_L pairs based on the mobile robot's dynamics. The dynamic model of a non-holonomic wheeled unicycle robot can be presented based on the Lagrange formalism, according to the following equation:

$$\begin{pmatrix}\frac{1}{b} & 0\\ 0 & \frac{d}{2b}\end{pmatrix}\begin{pmatrix}u_1\\ u_2\end{pmatrix}=\begin{pmatrix}\frac{2I_\omega}{b^2}+m & 0\\ 0 & \frac{I_\omega d^2}{2b^2}+I\end{pmatrix}\begin{pmatrix}\dot{u}\\ \dot{\omega}\end{pmatrix}+\begin{pmatrix}0 & -m_c\omega\\ m_c\omega & 0\end{pmatrix}\begin{pmatrix}u\\ \omega\end{pmatrix} \tag{10}$$

with:

$$\begin{cases}u_1=\tau_R+\tau_L\\ u_2=\tau_R-\tau_L\end{cases} \tag{11}$$

where τ_R and τ_L are the input torque expressed in $(N.m)$ for the Right and the Left wheels respectively; m is the total mass of the robot; m_c is the total mass of the robot's platform; I is the total equivalent inertia of the system; b is the radius of the wheel; d is the width of the robot.

The objective of the present control algorithm is to minimize the trajectory tracking error and improve the stability of the robot using the super-twisting algorithm as a second-order sliding mode controller, known for its high robustness and reduction of the chattering phenomena compared to the first-order sliding mode control (Yang et al., 2020).

4.3 Selection of a Sliding Surface

The choice of the sliding surface presented in equation (12) is made based on the general formula of the sliding surface:

$$S(t)=\begin{pmatrix}S_1(t)\\ S_2(t)\end{pmatrix}=\begin{pmatrix}e_v+k_1\int e_v\\ e_\omega+k_1\int e_\omega\end{pmatrix} \tag{12}$$

where k_1 is a positive constant; e_v and e_ω are the errors between the speeds produced by the kinematic controller (desired speeds) and the real speeds of the robot and are presented as follows:

$$\begin{pmatrix} e_v \\ e_\omega \end{pmatrix} = \begin{pmatrix} u_{ref}(t) - u(t) \\ \omega_{ref}(t) - \omega(t) \end{pmatrix} \tag{13}$$

By deriving equation (12), one obtains:

$$\dot{s}(t) = \begin{pmatrix} \dot{s}_1(t) \\ \dot{s}_2(t) \end{pmatrix} = \begin{pmatrix} \dot{e}_v + k_1 e_v \\ \dot{e}_\omega + k_1 e_\omega \end{pmatrix} \tag{14}$$

4.4 Determination of a Control Law

The objective of this step is to find a robust control law to achieve the sliding mode in finite time. In the following, the super-twisting sliding mode algorithm is distinguished by remarkable robustness among various control algorithms (Li & Peng, 2021):

$$\dot{s} = -\lambda |S|^{1/2} sign(S) - k_2 \int sign(S) \tag{15}$$

Let the differential equation of the proposed system be:

$$\dot{x}(t) = f(x,t) + g(x,t)u(t) + d(t) \tag{16}$$

where (*f*,g) are non-linear functions; u is the input of the system or the control; *d* is a disturbance; *x* is the state variable of the system.

Considering that the present system is linear and free from any perturbation, equation (16) becomes:

$$\dot{x}(t) = g(x,t)u(t) \Rightarrow \begin{pmatrix} \dot{u} \\ \dot{\omega} \end{pmatrix} = g(x,t) \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \tag{17}$$

From equation (10), it can be found that:

$$g(x,t) = \begin{pmatrix} \dfrac{1}{b(\frac{2I_\omega}{b^2} + m)} & 0 \\ 0 & \dfrac{L}{b(\frac{I_\omega d^2}{2b^2} + I)} \end{pmatrix} \tag{18}$$

Then, one defines the functions as:

$$g_1 = \frac{1}{b(\frac{2I_\omega}{b^2} + m)}; g_2 = \frac{L}{b(\frac{I_\omega d^2}{2b^2} + I)} \tag{19}$$

Equation (17) becomes:

$$\begin{cases} \dot{u} = g_1 U_1 \\ \dot{\omega} = g_2 U_2 \end{cases} \tag{20}$$

To find the control law $U(t)$, equation (15) has been solved using equations (14) and (20) as follows:

$$\begin{aligned} \dot{s}(t) &= \begin{pmatrix} \dot{e}_v + k_1 e_v \\ \dot{e}_\omega + k_1 e_\omega \end{pmatrix} \\ &= \begin{pmatrix} \dot{u}_{ref} - g_1 U_1 + k_1 e_v \\ \dot{\omega}_{ref} - g_2 U_2 + k_1 e_\omega \end{pmatrix} \\ &= \begin{pmatrix} -\lambda |S|^{1/2} sign(S) - k_2 \int sign(S) \\ -\lambda |S|^{1/2} sign(S) - k_2 \int sign(S) \end{pmatrix} \end{aligned} \tag{21}$$

The control law obtained is presented in equation (22):

$$U = \begin{cases} U_1 = \dfrac{1}{g_1} \begin{bmatrix} \dot{u}_{ref} + k_1 e_v(t) \\ +\lambda |S|^{1/2} sign(S) - k_2 \int sign(S) \end{bmatrix} \\ U_2 = \dfrac{1}{g_2} \begin{bmatrix} \dot{\omega}_{ref} + k_1 e_\omega(t) \\ +\lambda |S|^{1/2} sign(S) - k_2 \int sign(S) \end{bmatrix} \end{cases} \tag{22}$$

where k_1, λ and k_2 are strictly positive constants.

4.5 Stability Analysis

Taking the time derivative of the Lyapunov function as $V_i = \frac{1}{2} S_i^2$ to ensure the finite-time convergence and the asymptotic stability of the sliding mode, the derivative of V_i with respect to time must be negatively defined. Using equations (13)-(14), the derivative of the surface *S* is written as follows:

$$\begin{cases} \dot{s}_1(t) = \dot{e}_v + k_1 e_v \\ \quad = \dot{u}_{ref} - \dot{u} + d_1 + k_1 e_v \\ \quad = \dot{u}_{ref} - g_1 U_1 + d_1 + k_1 e_v \\ \dot{s}_2(t) = \dot{e}_\omega + k_1 e_\omega \\ \quad = \dot{\omega}_{ref} - \dot{\omega} + d_2 + k_1 e_\omega \\ \quad = \dot{\omega}_{ref} - g_2 U_2 + d_2 + k_1 e_\omega \end{cases} \tag{23}$$

Substituting equation (22) into equation (23) one obtains:

$$\begin{cases} \dot{s}_1 = -\lambda |S|^{1/2} sign(S) - k_2 \int sign(S) + d_1 \\ \dot{s}_2 = -\lambda |S|^{1/2} sign(S) - k_2 \int sign(S) + d_2 \end{cases} \tag{24}$$

By using the above equations, the Lyapunov function V_i can be calculated as follows:

$$\begin{aligned} V_1 &= S_1\dot{S}_1 \\ &= S_1(-\lambda|S_1|^{1/2} sign(S_1) - k_2\int sign(S_1) + d_1) \quad (25) \\ &= -\lambda|S_1|^{3/2} - k_2|S_1| + d_1 S_1 \end{aligned}$$

$$\begin{aligned} V_2 &= S_2\dot{S}_2 \\ &= S_2(-\lambda|S_2|^{1/2} sign(S_2) - k_2\int sign(S_2) + d_2) \quad (26) \\ &= -\lambda|S_2|^{3/2} - k_2|S_2| + d_2 S_2 \end{aligned}$$

To ensure the convergence of the system to the sliding surface and its stability in approaching the origin asymptotically, V_i must be defined as negative. Therefore, the design parameters must be selected to meet the condition $V_i \le 0$. Suppose that d_i is bounded with γ. Then, $V_i \le 0$ is ensured by choosing $k_2 \le 0$.

5. Adaptative Super-Twisting Sliding Mode Control Using IEHO Algorithm

The controller based on the super-twisting sliding mode developed in the previous section (equation (22)) is used to keep the mobile robot on its trajectory. In fact, the correct operation of this controller requires a good choice of its parameters, which is not easy to achieve in a similar problem. In this work, the IEHO algorithm is used to adapt the proposed controller by determining the optimal parameters (k_1, λ and k_2) . The IEHO algorithm is implemented in a search space of dimension *D=3* (the three parameters k_1, λ and k_2) and for a maximum number of iterations equal to *60*. The fitness function optimized by IEHO must ensure robust, fast, and accurate responses. That is why the Integral Square Error (ISE) defined by (27) is used to evaluate the fitness function. According to several research works, ISE provides better performance when evaluating such a controller, compared to other performance indices such as; Absolute Temporal Integral Error (ATIE), Absolute Integral Error (AIE), and Square Temporal Integral Error (STIE).

$$ISE = \int_0^{+\infty} e^2(t)dt \quad (27)$$

The IEHO must minimize the fitness function expressed by :

$$f = ISE_x + ISE_y \quad (28)$$

where f is a vector that includes the integral square error of the robot position along $y(ISE_y)$ and $y(ISE_y)$, respectively.

The calculation of the fitness function is performed using the dynamic model of the simulated mobile robot for the period *t = 63s* to optimize the controller parameters, as shown in Figure 3.

6. Simulations and Results

In this section, simulations of the proposed controller have been carried out to demonstrate the effectiveness and the utility of tuning the parameters of the super-twisting sliding mode controller by means of the IEHO algorithm. The proposed IEHO-STSM adaptive controller is applied for a path based on scenarios. Simulations demonstrate the efficient use of IEHO in comparison with the employment of EHO, PSO, and GWO algorithms, by examining the trajectory tracking results of the various methods in the presence of external disturbances. All algorithms have been implemented in MATLAB/SIMULINK to search the optimal parameters for the STSM

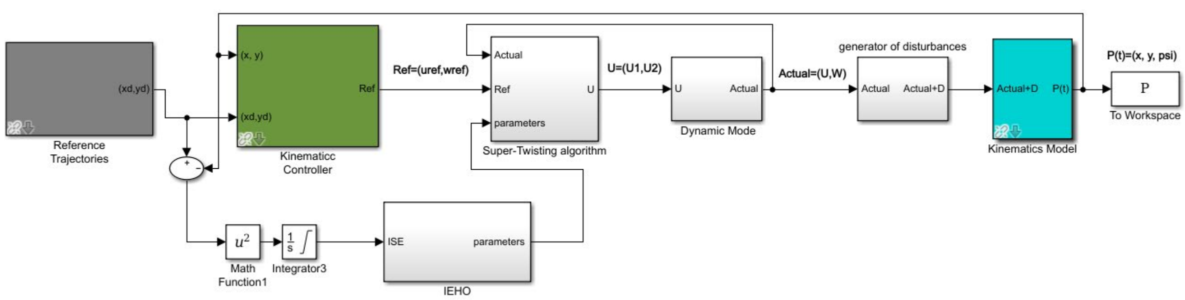


Figure 3. Complete control scheme for mobile robot

controller. The parameter values used for these algorithms are listed in Table 3.

Table 3. The values of the parameters for different algorithms

Algorithm	Iterations	Specific parameters
IEHO	*60*	Population Size = 400
		Number of clans = 50
		$\Gamma = 1.9;\ \beta = 0.4;\ R_0 = 2$
EHO	*60*	Population Size = 400
		Number of clans = 50
		$\alpha = 0.25;\ \beta = 0.4$
GWO	*60*	Number of search agents = 400
PSO	*60*	Number of particles = 400
		The confidence coefficients $C_1 = 0.9;\ C_2 = 1.4$
		The inertia weight $w = 0.9$

6.1 Reach a Fixed Point

The tests carried out in this subsection verify the ability of the proposed controller to enable the mobile robot to reach a fixed point in the presence of disturbances. The simulation tests evaluate the STSM controller adjusted by different algorithms using the values of the paraeters (k_1, λ and k_2) found by each algorithm (see Table 4).

Table 4. Values of the parameters k_1, λ and k_2 found by each algorithm

Algorithm	Specific parameters
IEHO	$k_1 = 23.740$
	$k_2 = 4.889e^{-04}$
	$\lambda = 16.843$
EHO	$k_1 = 4.759$
	$k_2 = 0.416$
	$\lambda = 3.791$
GWO	$k_1 = 4.062$
	$k_2 = 0.0037$
	$\lambda = 2.845$
PSO	$k_1 = 4.719$
	$k_2 = 0.0033$
	$\lambda = 2.829$

Figure 4 shows the response of the STSM controller adjusted by EHO, GWO, PSO, and IEHO algorithms. The results show that the parameters found by IEHO allow the STSM controller to reach the fixed point with zero error compared to the other algorithms. The IEHO-STSM ensures the stability of the mobile robot when it reaches the fixed point. The controller successfully eliminates all the external disturbances.

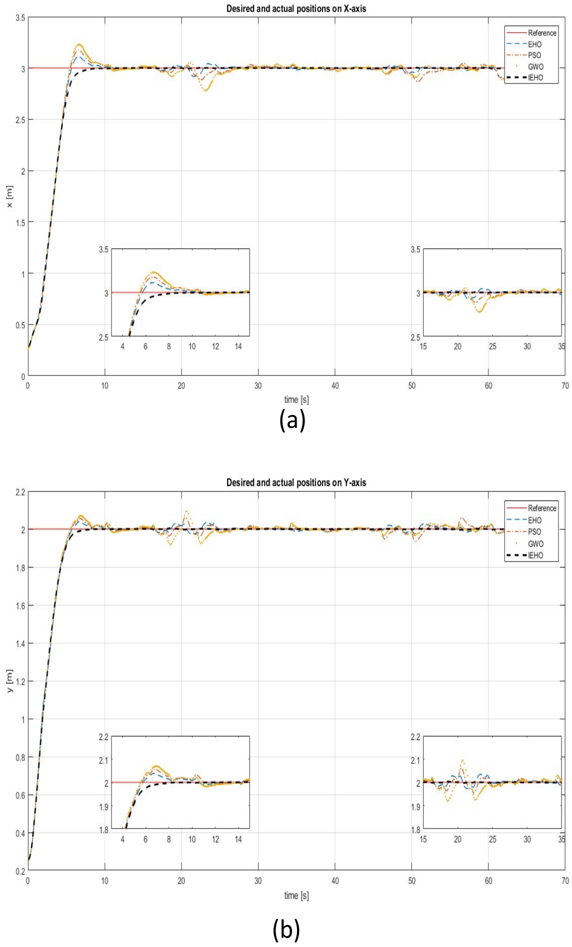


Figure 4. IEHO-STSM, EHO-STSM, PSO-STSM and GWO-STSM for mobile robot's reaching position: (a) x position; (b) y position

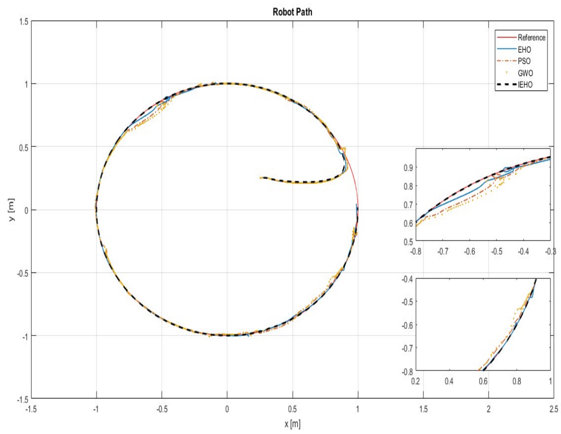
6.2 Trajectory Tracking

In this subsection, the simulations test the ability of the mobile robot to achieve and track a desired trajectory. To verify the stability of the proposed controller, external disturbances have been applied to the angular and linear velocities of the robot along its trajectory. Figure 5 and Figure 6 illustrate the desired trajectory tracking of the mobile robot when using different controllers obtained by each algorithm, as well as the distance error evolution for each type of trajectory.

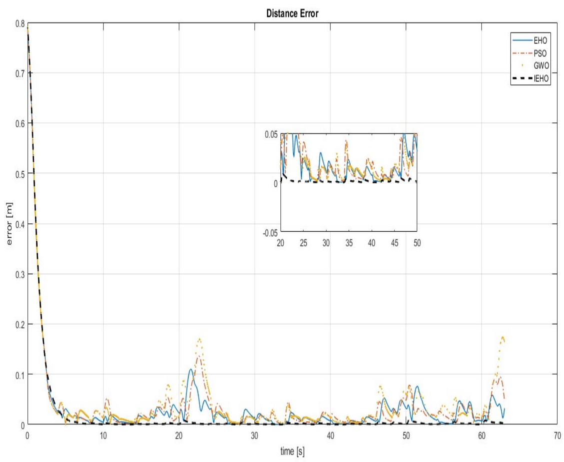
Figure 5(a) and Figure 6(a) illustrate the perfect tracking of the desired path. From those figures, it can be seen that the IEHO-STSM controller eliminates external disturbances along the trajectory, better than other controllers, namely

EHO-STSM, PSO-STSM, and GWO-STSM which have an error of +0.1 m, as response to disturbances.

However, the IEHO-STSM controller stays stable against all the disturbances and keeps the error between the robot's current and desired trajectory to the minimum (see Figure 5(b) and Figure 6-(b)). These results prove the superiority of IEHO-STSM in trajectory tracking. Indeed, the IEHO algorithm adjusts the STSM controller to have a large capacity to reject unknown external or internal perturbations over several iterations equal to 60. In addition, several tests have been established which demonstrated that other algorithms needed more iterations as well as more population size to find a result equivalent to the one found by IEHO. This proved the higher speed of convergence of IEHO when compared to those of other methods (see Figure 7).

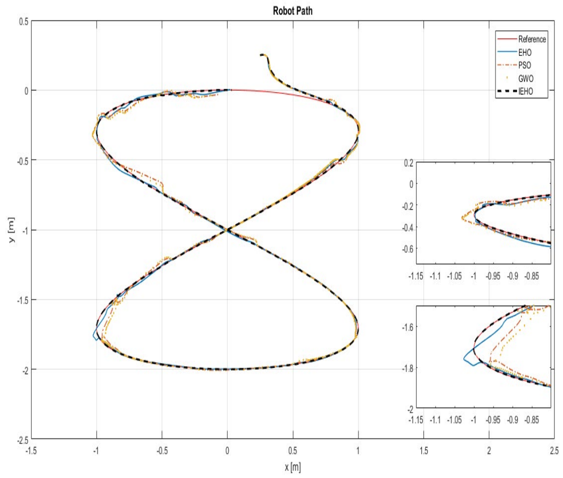


(a)

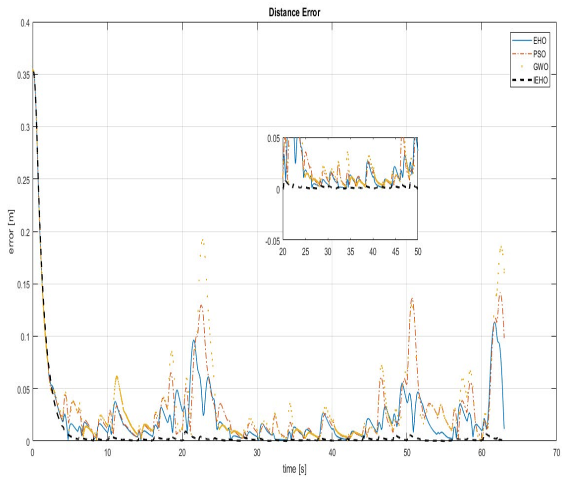


(b)

Figure 5. (a) Mobile robot's trajectory tracking using IEHO-STSM, EHO-STSM, PSO-STSM and GWO-STSM ; (b) The distance error evolution



(a)



(b)

Figure 6. (a) Mobile robot's trajectory tracking using IEHO-STSM, EHO-STSM, PSO-STSM and GWO-STSM; (b) The distance error evolution

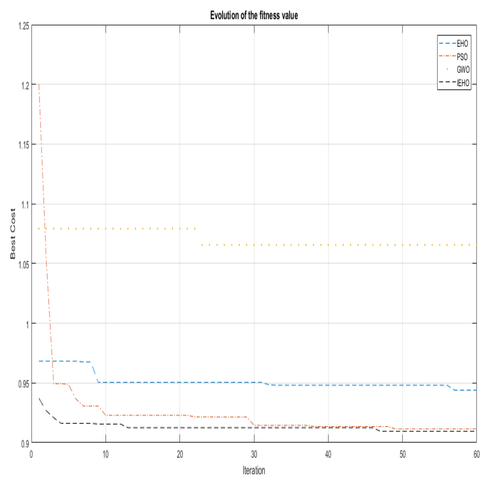


Figure 7. Evolution of the fitness value

Table 5. Table comparative based on ISE and IAE errors

Scenario	Performance	IEHO-STSM	EHO-STSM	GWO-STSM	PSO-STSM
Scenario 1	C_T (s)	**25.53**	67.28	27.63	51.84
	ISE_x	**0.4738**	0.4983	0.5343	0.5116
	ISE_y	**0.0438**	0.0578	0.0869	0.0663
	IAE_x	**1.1387**	1.8224	2.1743	1.9720
	IAE_y	**0.3790**	0.9699	1.2380	1.0907
Scenario 2	C_T (s)	**26.32**	56.03	36.54	41.83
	ISE_x	**0.0525**	0.0827	0.1537	0.1230
	ISE_y	**0.0600**	0.0748	0.1029	0.0829
	IAE_x	**0.4184**	1.1912	1.7587	1.5889
	IAE_y	**0.4364**	1.0046	1.3819	1.1588

In addition to the distance error, a computation of the integral square error (ISE) and the integral absolute error (IAE) was performed to compare IEHO-STSM with the other controllers. Table 5 shows the values of ISE, IAE errors, and the computation time (C_T) for every controller in each scenario. IEHO-STSM has minimal computation time and better performance in minimizing ISE and IAE errors, which confirms the superiority of the IEHO algorithm in finding the optimal solution over other algorithms. The improvement proposed in this work for IEHO has greatly developed the space exploration capability of the basic EHO algorithm, while the improved IEHO algorithm becomes faster and more precise in finding the global minimum, avoiding being trapped in a local minimum, as in the case of the the PSO and EHO algorithms.

7. Conclusion

To solve the trajectory tracking problem of a mobile robot, a sliding mode controller has been proposed. Indeed, in this work, a super-twisting algorithm controller has been developed based on the dynamic model of a unicycle mobile robot to reduce the trajectory error. In addition, an improved IEHO algorithm has been proposed to extend the optimization performances of the basic EHO algorithm. The testing of IEHO has been done by employing it in tuning the parameters of the STSM controller. Simulation results show that the proposed IEHO algorithm has a higher convergence speed and a greater space exploration capacity compared to EHO, which makes it a competitive algorithm for solving the optimization problem. Also, the same results prove that the proposed IEHO-STSM algorithm can rapidly approach the desired trajectory with higher precision tracking, and a great capacity to remove all types of disturbance, when comparing it to other methods.

REFERENCES

Aouf, A., Boussaid, L. & Sakly, A. (2017). A PSO algorithm applied to a PID controller for motion mobile robot in a complex dynamic environment. In: *2017 International Conference on Engineering & MIS (ICEMIS)*, 0*8-10 May 2017, Monastir, Tunisia*. New York, IEEE. pp. 1–7. doi: 10.1109/ICEMIS.2017.8273012.

Bessas, A., Benalia, A. & Boudjema, F. (2016) Integral sliding mode control for trajectory tracking of wheeled mobile robot in presence of uncertainties. *Journal of Control Science and Engineering*. 2016, 7915375. doi: 10.1155/2016/7915375.

Besseghieur, K. L., Trębiński, R., Kaczmarek, W. & Panasiuk, J. (2018) Trajectory tracking control for a nonholonomic mobile robot under ROS. *Journal of Physics: Conference Series*. 1016(1), 12008. doi: 10.1088/1742-6596/1016/1/012008.

Binh, N. T., Hung, N. M., Tung, N. A., Nam, D. P., & Long, N. T. (2017) Robust H-infinity backstepping control design of a wheeled inverted pendulum system. In: *2017 International Conference on System Science and Engineering (ICSSE)*, *21-23 July 2017 , Ho Chi Minh City, Vietnam*. New York, IEEE. pp. 444–449. doi: 10.1109/ICSSE.2017.8030914.

Chen, J., Shuai, Z., Zhang, H. & Zhao, W. (2020) Path following control of autonomous four-wheel-independent-drive electric vehicles via second-order sliding mode and nonlinear disturbance observer techniques. *IEEE Transactions on Industrial Electronics*. 68(3), 2460–2469. doi: 10.1109/TIE.2020.2973879.

Cibiraj, N. & Varatharajan, M. (2017) Chattering reduction in sliding mode control of quadcopters using neural networks. *Energy Procedia*. 117, 885–892. doi: 10.1016/j.egypro.2017.05.207.

Cui, R., Guo, J. & Mao, Z. (2015) Adaptive backstepping control of wheeled inverted pendulums models. *Nonlinear Dynamics*. 79, 501–511. https://doi.org/10.1007/s11071-014-1682-9

El Gmili, N., El Mjahed, M., El Kari, A. & Ayad, H. (2019) Particle Swarm Optimization and Cuckoo Search-Based Approaches for Quadrotor Control and Trajectory Tracking. *Applied Sciences*. 9(8), 1719. doi: 10.3390/app9081719.

El Hamidi, K., Mjahed, M., El Kari, A. & Ayad, H. (2019) Neural Network and Fuzzy-logic-based Self-tuning PID Control for Quadcopter Path Tracking. *Studies in Informatics and Control*. 28(4), 401–412. doi: 10.24846/v28i4y201904.

El-Naggar, M. F., Mosaad, M. I., Hasanien, H. M., AbdulFattah, T. A. & Bendary, A. F. (2021) Elephant herding algorithm-based optimal PI controller for LVRT enhancement of wind energy conversion systems. *Ain Shams Engineering Journal*. 12(1), 599–608. doi: 10.1016/j.asej.2020.07.013.

ElShaarawy, I. A., Houssein, E. H., Ismail, F. H. & Hassanien, A. E. (2019) An exploration-enhanced elephant herding optimization. *Engineering Computations*. 36(9), 3029-3046. doi: 10.1108/EC-09-2018-0424.

Fabregas, E., Farias, G., Aranda-Escolástico, E. Garcia, G., Chaos, D., Dormido-Canto, S. & Bencomo, S. D. (2019) Simulation and experimental results of a new control strategy for point stabilization of nonholonomic mobile robots. *IEEE Transactions on Industrial Electronics*. 67(8), 6679–6687. doi: 10.1109/TIE.2019.2935976.

Goswami, N. K. & Padhy, P. K. (2018) Sliding mode controller design for trajectory tracking of a non-holonomic mobile robot with disturbance. *Computers & Electrical Engineering*. 72, 307–323. doi: 10.1016/j.compeleceng.2018.09.021.

Khai, T. Q., Ryoo, Y.-J., Gill, W.-R. & Im, D.-Y. (2020) Design of kinematic controller based on parameter tuning by fuzzy inference system for trajectory tracking of differential-drive mobile robot. *International Journal of Fuzzy Systems*. 22, 1972–1978. doi: 10.1007/s40815-020-00842-9,

Li, W., Wang, G.-G. & Alavi, A. H. (2020) Learning-based elephant herding optimization algorithm for solving numerical optimization problems. *Knowledge-Based Systems*. 195, 105675. doi: 10.1016/j.knosys.2020.105675.

Li, Y.-R. & Peng, C.-C. (2021) Super-twisting sliding mode control law design for attitude tracking task of a spacecraft via reaction wheels. *Mathematical Problems in Engineering*. 2021(8), 1–13. doi: 10.1155/2021/6644033.

Liu, H., Nie, J., Sun, J. & Tian, X. (2020) Adaptive super-twisting sliding mode control for mobile robots based on high-gain observers. *Journal of Control Science and Engineering*. 2020. 4048507. doi: 10.1155/2020/4048507.

Martins, F. N., Celeste, W. C., Carelli, R., Sarcinelli-Filho, M. & Bastos-Filho, T. F. (2008) An adaptive dynamic controller for autonomous mobile robot trajectory tracking. *Control Engineering Practice*. 16(11), 1354–1363. doi: 10.1016/j.conengprac.2008.03.004.

Matraji, I., Al-Durra, A., Haryono, A., Al-Wahedi, K. & Abou-Khousa, M. (2018) Trajectory tracking control of skid-steered mobile robot based on adaptive second order sliding mode control. *Control Engineering Practice*. 72(4), 167–176. doi: 10.1016/j.conengprac.2017.11.009.

Mirzaeinejad, H. (2019) Optimization-based nonlinear control laws with increased robustness for trajectory tracking of non-holonomic wheeled mobile robots. *Transportation Research Part C: Emerging Technologies*. 101, 1–17. doi: 10.1016/j.trc.2019.02.003.

Nikranjbar, A., Haidari, M. & Atai, A. A. (2018) Adaptive sliding mode tracking control of mobile robot in dynamic environment using artificial potential fields. *Journal of Computer & Robotics*. 11(1), 1–14.

Obaid, M. A. M., Husain, A. R. & Al-Kubati, A. A. M. (2016) Robust backstepping tracking control of mobile robot based on nonlinear disturbance observer. *International Journal of Electrical and Computer Engineering*. 6(2), 901-908. doi: 10.11591/ijece.v6i2.pp901-908.

Oultiligh, A., Ayad, H., El Kari, A., Mjahed, M. & El Gmili, N. (2021) A Hybrid PSO-GWO Algorithm for Robot Path Planning in Uncertain Environments. *International Review of Automatic Control (IREACO).* 14(6), 360. doi: 10.15866/ireaco.v14i6.20045.

Pozna, C., Precup, R.-E., Horváth, E. & Petriu, E. M. (2022) Hybrid particle filter–particle swarm optimization algorithm and application to fuzzy controlled servo systems. *IEEE Transactions on Fuzzy Systems*. 30(10), 4286–4297. doi: 10.1109/TFUZZ.2022.3146986.

Qiu, Y., Li, B., Shi, W. & Zhang, X. (2019) Visual servo tracking of wheeled mobile robots with unknown extrinsic parameters. *IEEE Transactions on Industrial Electronics*. 66(11), 8600–8609. doi: 10.1109/TIE.2019.2891407.

Shijin, C. S. & Udayakumar, K. (2017) Speed control of wheeled mobile robots using PID with dynamic and kinematic modelling. In: *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, *17-18 March 2017, Coimbatore, India*. New York, IEEE. pp. 1–7. doi: 10.1109/ICIIECS.2017.8275962.

Shirzadeh, M., Amirkhani, A., Tork, N. & Taghavifar, H. (2021) Trajectory tracking of a quadrotor using a robust adaptive type-2 fuzzy neural controller optimized by cuckoo algorithm. *ISA Transactions*. 114, 171–190. doi: 10.1016/j.isatra.2020.12.047.

Siti, I., Mjahed, M., Ayad, H. & El Kari, A. (2019) New Trajectory Tracking Approach for a Quadcopter Using Genetic Algorithm and Reference Model Methods. *Applied Sciences*. 9(9), 1780. doi: 10.3390/app9091780.

Truong, T. N., Vo, A. T. & Kang, H.-J. (2021) A backstepping global fast terminal sliding mode control for trajectory tracking control of industrial robotic manipulators. *IEEE Access*. 9, 31921–31931. doi: 10.1109/ACCESS.2021.3060115.

Vázquez, L. A., Jurado, F., Castaneda, C. E. & Santibanez, V. (2016) Real-time decentralized neural control via backstepping for a robotic arm powered by industrial servomotors. *IEEE Transactions on Neural Networks and Learning Systems*. 29(2), 419–426. doi: 10.1109/TNNLS.2016.2628038.

Vijay, M. & Jena, D. (2017) PSO based neuro fuzzy sliding mode control for a robot manipulator. *Journal of Electrical Systems and Information Technology*. 4(1), 243–256. doi: 10.1016/j.jesit.2016.08.006.

Wang, G.-G., Deb, S., & Coelho, L. dos S. (2015) Elephant herding optimization. In: *2015 3rd International Symposium on Computational and Business Intelligence (ISCBI)*, *07-09 December 2015, Bali, Indonesia*. New York, IEEE. pp. 1–5. doi: 10.1109/ISCBI.2015.8.

Yang, F., Zhang, K. & Yu, L. (2020) Adaptive super-twisting algorithm-based nonsingular terminal sliding mode guidance law. *Journal of Control Science and Engineering*. 2020, 1058347. doi: 10.1155/2020/1058347.

Zhou, Z., Wang, C., Zhu, Z., Wang, Y. & Yang, D. (2019) Sliding mode control based on a hybrid grey-wolf-optimized extreme learning machine for robot manipulators. *Optik*. 185, 364–380. doi: 10.1016/j.ijleo.2019.01.105.