Optimal Second Order Sliding Control for the Robust Tracking of a 2-Degree-of-Freedom Helicopter System based on Metaheuristics and Artificial Neural Networks

Raghda JOUIROU*, Wafa BOUKADIDA, Anouar BENAMOR

Laboratory of Automatic Signal and Image Processing, National Engineering School of Monastir, University of Monastir, 5019, Tunisia

joui.raghda@gmail.com (**Corresponding author*), wafaboukadida91@gmail.com, benamor_anouar@yahoo.com

Abstract: This paper presents a novel formulation for a Second-order Sliding Controller (SOSMC) for the trajectory tracking of a 2-degree-of-freedom (DoF) helicopter system based on the combination of a robust Second-order Discrete Sliding Mode Controller (SDOSMC) and a Linear Quadratic Regulator (LQR). The combination of these two approaches guarantees the effectiveness of the SOSMC in the face of uncertainties and disturbances. A new approach based on the resolution of the Sylvester equation is proposed in order to design a sliding surface that would ensure the optimal performance of the Sliding Mode Control (SMC). The performance of the SMC heavily depends on the selection of the sliding surface. The reformulation of the control problem is treated as a multi-objective optimization problem, and a new objective function based on dynamic aggregation is proposed for this purpose. The main contribution of this article lies in using the set of solutions provided by metaheuristics to leverage the capabilities of Deep Learning (DL) and to predict the optimal solution in terms of performance. The proposed control methodology is applied to control the pitch and yaw axes of the Quanser helicopter system. The simulation results, along with a comparative analysis were included to support the importance and efficacy of the suggested control technique.

Keywords: Second-order Discrete Sliding Mode Controller (SDOSMC), Linear Quadratic Regulator (LQR), Multi-Objective Optimization (MOO), Deep Learning (DL).

1. Introduction

Due to the extensive use of Digital Signal Processing (DSP) and digital computers in controller design, much effort has been devoted to investigating discrete-time sliding mode control. Sliding mode control has proven to be a nonlinear control technique that combines good robustness and efficiency for incompletely modeled or uncertain systems. This approach stems from the theory of variable structure control and has given rise to multiple studies for more than sixty years (Utkin, Guldner & Shi, 1999; Niu, Ho & Wang, 2010). This method is widely known as a robust control method and offers significant robustness against disturbances and uncertainties. Sliding mode control is favored for the simple calculations it involves and its excellent efficiency.

This approach includes two steps. First, a slip surface is determined to bring the system's state trajectory towards that surface and make it commute using a logic of appropriate switching around it until the equilibrium point, resulting in the phenomenon of sliding. In the second phase, the control law must be created so that any state external to the slip surface reaches the surface in a limited amount of time and stays there (Utkin, 1978; Hassen, Laamiri & Messaoud, 2021).

Due to their agility, speed, and ability to land and take off from various terrains, helicopters have a wide range of applications in land, sea, and air activities, which has led to extensive research to improve their control (Boukadida et al., 2019). However, the large nonlinearities, model errors, and inter-axis coupling of these systems make them difficult to control.

For more than twenty years, researchers have used the Quanser Aero 2-DoF helicopter as a testbed to experimentally validate different control strategies and analyze their effectiveness. One notable approach is the design of an optimal regulator based on a combination of SMC and LQR, which guarantees accurate trajectory tracking of the Quanser Aero system (Nuthi & Subbarao, 2015). Additionally, a new control scheme that uses an appropriate mix of adaptive control methods and LQR + Integrator (LQR-I) was published in (Niu, Ho & Wang, 2010) and has shown effective stability and tracking of an input.

However, there are some drawbacks to these studies. One major issue is the choice of the penalty matrices Q and R, which poses a significant problem for the LQR technique (Tsai et al., 2013; Boukadida et al., 2019).The quadratic cost index J, composed of these two matrices, is minimized by the LQR controller. However, determining these matrices is not a simple task and can significantly affect the controller's

https://doi.org/10.24846/v32i4y202307

performance. Each different value of Q and R will ultimately result in a different system response. For example, choosing a higher value of R may stabilize the helicopter with less energy, but the time response will be longer. Similarly, higher Q values reflect faster convergence of the system to the origin, but higher energies must be taken into account.

To address these issues and strike a speed-energy balance, a multi-objective optimization problem is suggested. This approach simultaneously considers a multi-objective optimization (MOO) problem while optimizing all individual objectives. Metaheuristics such as Moth Flame Optimization (MFO) Algorithm, Gray Wolf Optimization (GWO) Algorithm, Particle Swarms Optimization (PSO), Genetic Algorithms (GAs), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), and Whale Optimization (WO) algorithms can be used to tune the Q and R parameters. In this work, the GA and WO methods are used to complete the LQR's weighting matrices, while also improving system performance and making a speed-energy trade-off.

It is important to select the values of these two weighting matrices carefully so that the exploration can quickly identify regions of the search space that comprise high-quality points, without wasting too much time exploiting less promising regions.

Computer vision enables machines to analyze, process, and understand one or more pieces of data provided by the system. This program can be considered to process visual data through patterns based on geometry, physics, biology, statistics, and learning theory. Computer vision is also recognized as an initiative to automate and integrate a wide range of processes and models for AI-based visual perception. Artificial intelligence (AI) refers to a group of technologies that allow robots to carry out tasks and solve issues typically handled by people. Sometimes, AI activities are relatively simple for humans, such as planning a robot's movements, picking up objects, or driving a car (He et al., 2016).

The ambition to create machines capable of imitating human intelligence was born in the 1950s. Artificial intelligence seeks to create specific machines capable of making decisions, simplifying, or even replacing human intervention (Nadimpalli, 2017; Cui et al., 2022; Yin, Niu & Liu, 2017).

In the framework of the proposed strategy, the reliance will be on the results of metaheuristic algorithms previously obtained in the context of optimization and global minimum achievement. Additionally, a neural network was developed that plays a pivotal role in this process. This neural network is harnessed for deep learning, utilizing the insights garnered from the metaheuristic algorithms. The neural network undergoes training to predict the best solution, thereby validating its optimality. It is important to emphasize that in this approach, deep learning is leveraged for the neural networks. Since the primary objective of this research was to determine the ideal LQR configuration, specifically the optimal values for the weighting matrices Q and R, it was crucial to strike the best possible balance between tracking precision and control effort.

The remainder of this paper is organized as follows. Section 2 provides an overview of the Quanser Aero 2-DoF Helicopter system. Section 3 is dedicated to problem formulation, laying the foundation for the subsequent developments. In Section 4, a novel method for constructing an optimal controller is presented, which combines mode control and the LQR technique. Additionally, a new approach to designing the optimal sliding surface based on the solution of the Sylvester equation is presented. Section 5 focuses on optimization techniques for controller parameters, with two distinct parts: multi-objective optimization, where genetic algorithms are discussed, and the latter delving into Multi-Layer Perceptrons. Section 6 is dedicated to validation, and it sets forth and analyzes simulation results from a comparative perspective to validate the proposed approach. Finally, the conclusion is given in Section 7.

2. The Quanser Aero 2-DoF Helicopter Description

The Quanser 2-DoF helicopter, shown in Figure 1, consists of a helicopter model mounted on a fixed base with 2 drive propellers, each consisting of 2 DC motors. The front propeller controls the altitude of the helicopter, causing it to rotate around the pitch axis, while the rear propeller controls the lateral movement of the helicopter around the yaw axis. High-resolution encoders are

used to measure the pitch and yaw angles. As it is illustrated in the Quanser Aero free-body diagram in Figure 2, the helicopter has two degrees of freedom: rotation about the pitch axis, denoted by θ, and movement along the yaw axis, represented by the angle ψ. The objective of controlling the input voltage on the DC motor is to regulate the pitch, while the system is forced to follow the reference trajectory by the yaw angle. The parameters of the helicopter system are defined in Table 1.

Figure 1. Quanser 2-DoF Helicopter (Boukadida et al., 2019)

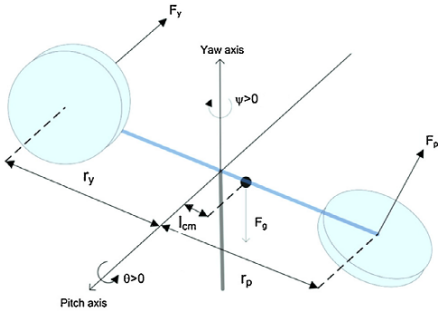

Figure 2. Simple free-body diagram of the2-DoF Helicopter (Boukadida et al., 2019)

Table 1. Helicopter system parameters (Boukadida et al., 2019)

Symbol	Description	Value	Unit
J_p	Moment of inertia of the pitch axis	0.0384	Kg · m²
J_y	Moment of inertia of the yaw axis	0.0432	Kg · m²
B_p	Equivalent viscous damping of the pitch axis	0.800	N/V
B_y	Equivalent viscous damping of the yaw axis	0.318	N/V
K_{pp}	Pitch motor thrust torque	0.204	N · m/V
K_{yy}	Yaw motor thrust torque	0.072	N · m/V
K_{py}	Yaw-pitch thrust torque	0.0068	N · m/V
K_{yp}	Pitch-yaw thrust torque	0.0219	N · m/V
m	Total mass of the helicopter	1.3872	Kg
l	Distance between center of mass and pitch axis	0.186	m
g	Gravitational acceleration	9.81	m/s²

Using the Euler-Lagrange formula, the following is the helicopter system:

$$\begin{cases} \dfrac{d^2\theta(t)}{dt^2} = \dfrac{K_{pp}V_{mp} + K_{py}V_{my} - B_p\dot{\theta}(t) + \Gamma(t)}{J_p + ml^2} \\ \dfrac{d^2\psi(t)}{dt^2} = \dfrac{K_{yp}V_{mp} + K_{yy}V_{my} - B_y\dot{\psi}(t) + N(t)}{J_y + m[cos(\theta(t))]^2 l^2} \end{cases} \quad (1)$$

where:

$$\begin{cases} \Gamma(t) = m[\dot{\psi}(t)]^2 sin(\theta(t))l^2 cos(\theta(t)) + mgcos(\theta(t))l \\ N(t) = 2m\dot{\psi}(t)sin(\theta(t))l^2 cos(\theta(t))\dot{\theta}(t) \end{cases} \quad (2)$$

$\theta(t), \dot{\theta}(t), \psi(t), \dot{\psi}(t), F_p, F_y, V_{mp}$ and V_{my} represent the pitch angle, the pitch angular velocity, yaw angle, yaw angular velocity, pitch thrust force, yaw thrust force, pitch and yaw motor control input voltages, respectively. The dynamics of the system should be modeled as a linear system in order to create a state regulator based on the LQR strategy. The given system's nonlinear model is then linearized around the origin by adding the values $\theta = 0, \dot{\theta} = 0, \psi = 0$ and $\dot{\psi} = 0$. The resulting differential equation is linearized as follows:

$$\begin{cases} \dfrac{d^2\theta(t)}{dt^2} = \dfrac{K_{pp}V_{mp} + K_{py}V_{my} - B_p\dot{\theta}(t)}{J_p + ml^2} \\ \dfrac{d^2\psi(t)}{dt^2} = \dfrac{K_{yp}V_{mp} + K_{yy}V_{my} - B_y\dot{\psi}(t)}{J_y + ml^2} \end{cases} \quad (3)$$

The system is described with four states: θ := Pitch angle, $\dot{\theta}$:= Pitch angular velocity, ψ := Yaw angle, and $\dot{\psi}$:= Yaw angular velocity.

The following are the linearized coupled state and system output equations:

$$\begin{cases} \dot{x}_1 = x_3 \\ \dot{x}_2 = x_4 \\ \dot{x}_3 = 29.29cos(x_1) - 2.36(u_1 + d_1) + 0.79(u_2 + d_2) \\ \dot{x}_4 = -3.49x_4 + 0.24(u_1 + d_1) + 0.79(u_2 + d_2) \\ y_1 = x_1 \\ y_2 = x_2 \end{cases} \quad (4)$$

It is assumed that the state vector is represented as $x = [x_1, x_2, x_3, x_4]^T = [\theta, \psi, \dot{\theta}, \dot{\psi}]^T$

The output vector is $y = [\theta, \psi]^T$

$d_1 = \sin(t), d_2 = 1.1\cos(t)$ are delivered into the system as disturbance voltage signals.

3. Problem Formulation

In case a continuous linear system is used, it is described by the following differential equation:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(k) + f(t) \\ y(t) = Dx(t) \end{cases} \quad (5)$$

where:

$x \in \mathbb{R}^n$ is the state vector and $y \in$ ^{p} is the output vector. $u \in \mathbb{R}^m$ is the control input. A, B and D are real well-known constant matrices of dimensions $n \times n, n \times m$ and $p \times n.$ $f(t)$ represents the parametric uncertainties on the model and the external disturbances assumed to be unknown but bounded. However, the implementation of this theory in discrete mode raised great interest when the researchers focused on the use of digital computers DSP. Discrete-time uncertain systems are described by:

$$\begin{cases} x(k+1) = \Phi x(k) + \Gamma u(k) + D_p(k) \\ y(k) = Dx(k) \end{cases} \quad (6)$$

where:

$x(k), y(k)$ and $u(k)$ are the state vector, the output vector and the control input of the above-mentioned dimensions n, m and p. Φ, Γ and D are real known constant matrices and the vector $D_p(k)$ is the external disturbance, which is presumed to be unidentified and norm-bounded, impacting the system, which satisfy the following assumptions:

Assumption 1:

The two matrices $(\Phi,\ D)$ are observables and the two matrices (Φ, Γ) are controllables.

Assumption 2:

$d(k) \in \mathbb{R}_m$ meets the corresponding criteria (Drazenovic, 1969):

$$D_p(k) = \Gamma d(k) \quad (7)$$

The second supposition states that (2) can be written as:

$$x(k+1) = \Phi x(k) + \Gamma(u(k) + d(k)) \quad (8)$$

The disruptions are unknown and meet the criteria below:

$$\| d(k) \| < \rho_{max} \quad (9)$$

The main goal of this work is to solve tracking problems with parameter uncertainty, convergence towards the desired state of the reference model and stability of the control loop can be ensured. The aim of this work is to fulfill the following conditions: to create a second-order sliding mode controller and ensure that the reference model's trajectory is tracked. A brand-new methodical approach is suggested for creating sliding surfaces. In addition to ensuring steady operation, this approach also controls the dynamic reaction of the system by managing its parameters.

The reference model is given by Zhang et al. (2018):

$$\begin{cases} x_r(\mathrm{k}+1) = A_r x_r(\mathrm{k}) + B_r\ \mathrm{r(k)} \\ y_r(\mathrm{k}) = D_r\ x_r(\mathrm{k}) \end{cases} \quad (10)$$

where:

$x_r(k) \in \mathbb{R}^n$, $y_r(k) \in \mathbb{R}^p$ represent the reference model's state and output vectors.

$A_r \in \mathbb{R}^{n \times n}$, $B_r \in \mathbb{R}^{n \times k}$ and $D_r \in \mathbb{R}^{p \times n}$ are known constant matrices. $r(k) \in \mathbb{R}^k$ is the reference input.

A tracking error is described as:

$$e(k) = y(k) - y_m(k) \quad (11)$$

4. Optimal Second-Order Sliding Mode Control

The objective is to determine a control law that guarantees the continuation of the trajectory of the model in (10). The idea is to transform the tracking problem into a stabilization problem.

$$\begin{cases} z(k) = x(k) - \varsigma x_r(k) - \upsilon r(k) \\ e(k) = y(k) - y_r(k) \end{cases} \quad (12)$$

where $z(k)$ is the new auxiliary state, and ς and υ are two matrices to be determined.

Theorem: If there exist $\varsigma \in \mathbb{R}^{n \times n}$, $\upsilon \in \mathbb{R}^{n \times n}$, $\mathrm{Y} \in \mathbb{R}^{m \times n}$ and $\Omega \in \mathbb{R}^{m \times n}$ satisfying:

$$\begin{cases} \Phi\varsigma - \varsigma A_r = -\Gamma \mathrm{Y} \\ (\Phi - I_n)\upsilon - \varsigma B_r = -\Gamma\Omega \\ D\varsigma = D_r \\ D\upsilon = 0 \end{cases} \quad (13)$$

Then the system's new dynamics is described by:

$$\begin{cases} z(k+1) = \Phi z(k) + \Gamma(u(k) - \mathrm{Y}x_r(k) - \Omega r(k) + d(k)) \\ E(k) = Dz(k) \end{cases} \quad (14)$$

Additionally, the different matrices are the solutions of the following equation:

$$\begin{bmatrix} vec(\dot{o}) \\ vec(\Upsilon) \end{bmatrix} = \begin{bmatrix} I_{nr} \otimes \Phi - A_r^T \otimes I_n & I_{nr} \otimes \Gamma \\ I_n \otimes D & 0 \end{bmatrix}^+ \begin{bmatrix} 0 \\ vec(D_r^T) \end{bmatrix}$$
$$\begin{bmatrix} \upsilon \\ \Omega \end{bmatrix} = \begin{bmatrix} \Phi - I_n & \Gamma \\ D & 0 \end{bmatrix}^{-1} \begin{bmatrix} \dot{o} B_r \\ 0 \end{bmatrix} \quad (15)$$

where $vec(x)$ is the vector obtained by arranging all the columns of x in one vector and $\otimes$ is the Kronecker product of the matrices.

Proof: To begin with, it is assumed that the reference input changes gradually over a brief sampling period. Moreover, based on expressions (8), (10), (13) and (14), the following is obtained:

$$\begin{aligned} &z(k+1) = x(k+1) - \varsigma x_r(k+1) - \upsilon r(k+1) \\ &= \Phi x(k) + \Gamma(u(k) + d(k)) - \varsigma A_r x_r(k) - \varsigma B_r r(k) \\ &- \upsilon r(k+1) \\ &= \Phi x(k) + \Gamma(u(k) + d(k)) - \varsigma A_r x_r(k) - \varsigma B_r r(k) \\ &- \upsilon r(k) \\ &= \Phi(x(k) - \varsigma x_r(k) - \upsilon r(k)) + \Phi \varsigma x_r(k) + \Phi \upsilon r(k) \\ &+ \Gamma(u(k) + d(k)) - \varsigma A_r x_r(k) - \varsigma B_r r(k) - \upsilon r(k) \\ &= \Phi z(k) + (\Phi \varsigma - \varsigma A_r) x_r(k) + (\Phi \upsilon - \varsigma B_r - \upsilon) r(k) \\ &+ \Gamma(u(k) + d(k)) \\ &= \Phi z(k) + \Gamma(u(k) - \Upsilon x_r(k) - \Omega r(k) + d(k)) \end{aligned} \quad (16)$$

On the other hand, according to the equations (6), (10), (12) and (13), one obtains:

$$\begin{aligned} &e(k) = y(k) - y_r(k) \\ &= Dx(k) - D_r x_r(k) \\ &= Dx(k) - D\varsigma x_r(k) \\ &= D(z(k) + \varsigma x_r(k) + \upsilon r(k)) - D\varsigma x_r(k) \\ &= Dz(k) \end{aligned}$$

To determine the matrices $\varsigma, \upsilon, \Upsilon$ and Ω the first and third equations of (13) are rewritten in matrix form as follows:

$$\begin{bmatrix} \Phi & \Gamma \\ D & 0 \end{bmatrix} \begin{bmatrix} \varsigma \\ \Upsilon \end{bmatrix} = \begin{bmatrix} \varsigma A_r \\ D_r \end{bmatrix} \quad (17)$$

The matrices ς and Υ are the solutions of equation (17) as they are mentioned in (Pai, 2014). This expression can, in fact, be rewritten as follows:

$$\Delta\Sigma = \Lambda \quad (18)$$

where: $\Delta = \begin{bmatrix} I_{nr} \otimes \Phi - A_r^T \otimes I_n & I_{nr} \otimes \Gamma \\ I_n \otimes D & 0 \end{bmatrix}^+$,

$\Sigma = \begin{bmatrix} vec(\dot{o}) \\ vec(\Upsilon) \end{bmatrix}$, and $\Lambda = \begin{bmatrix} 0 \\ vec(D_r^T) \end{bmatrix}$

The solutions of expression (18) exist only if $rank(\Delta\Sigma) = rank(\Delta)$ (Pai, 2014). One of these solutions is given by:

$$\Sigma = \Delta^+ \Lambda \quad (19)$$

where Δ^+ is the pseudo-inverse of Δ. Similarly, the second and fourth equations of (13) are described in matrix form as follows:

$$\begin{bmatrix} \Phi - I_n & \Gamma \\ D & 0 \end{bmatrix} \begin{bmatrix} \upsilon \\ \Omega \end{bmatrix} = \begin{bmatrix} \varsigma B_r \\ 0 \end{bmatrix} \quad (20)$$

It is assumed that:

$$\begin{bmatrix} \Psi_{11} & \Psi_{22} \\ \Psi_{21} & \Psi_{22} \end{bmatrix} = \begin{bmatrix} \Phi - I_n & \Gamma \\ D & 0 \end{bmatrix}^{-1} \quad (21)$$

It follows from (20) that:

$$\begin{bmatrix} \upsilon \\ \Omega \end{bmatrix} = \begin{bmatrix} \Psi_{11} & \Psi_{22} \\ \Psi_{21} & \Psi_{22} \end{bmatrix} \begin{bmatrix} \varsigma B_r \\ 0 \end{bmatrix} \quad (22)$$

After creating the related matrices of the reference model, a new optimal control law is suggested to satisfy the trajectory tracking:

$$\begin{cases} u(k) = v(k) + \Upsilon x_r(k) + \Omega r(k) \\ v(k) = u(k) - (\Upsilon x_r(k) + \Omega r(k)) \end{cases} \quad (23)$$

By replacing equation (23) in system (14), the following is obtained:

$$z(k+1) = \Phi z(k) + \Gamma(v(k) + d(k)) \quad (24)$$

4.1 Optimal Control Law for a Nominal System

To choose the best controller, it is necessary to minimize the performance index J, calculated as follows:

$$J = \sum_{k=0}^{\infty} (z^T(k) Q z(k) + u_{op}^T(k) R u_{op}(k)) \quad (25)$$

Q and R are two square matrices with *n* and *m* as their respective dimensions. The Hamiltonian is expressed in (Das & Mahanta, 2014) to determine the control that minimizes the quadratic index J:

$$\begin{aligned} &H(k) = z^T(k) Q z(k) + u^T(k) R u(k) \\ &+ p^T(k+1) z(k+1) \end{aligned} \quad (26)$$

P represents a symmetric positive matrix that solves the Riccati equation, which is given by:

$$P = Q + \Phi^T P\Phi - \Phi P\Gamma\left(R + \Gamma^T P\Gamma\right)^{-1}\Gamma^T P\Phi \quad (27)$$

The optimal control for reducing the quadratic index is, based on $H(k)$:

$$\begin{aligned} Uop(k) &= -\left(R + \Gamma^T P\Gamma\right)^{-1}\Gamma^T P\Phi z(k) \\ &= -Kz(k) \end{aligned} \quad (28)$$

4.2 Optimal Sliding Surface

The objective of the control is to ensure the stability of the system despite the uncertainties. The ideal sliding state is achieved by the existence of a finite time (tf) such that the solution of the system satisfies s(k) = 0 for all t ≥ tf (Zhao, Wu & Ma, 2013).

The following equation expresses the first-order sliding surface s(k):

$$s(k) = Cz(k) \quad (29)$$

with:

$$C = \begin{bmatrix} \overline{C} & I \end{bmatrix} \quad (30)$$

The sliding surface function is defined as follows:

$$\begin{aligned} s(k+1) &= (\overline{C}\overline{\Phi}_{11} + \overline{\Phi}_{21})z_1(k) \\ &+ (\overline{C}\overline{\Phi}_{12} + \overline{\Phi}_{22})z_2(k) + \Gamma_2 u(k) \end{aligned} \quad (31)$$

Once the optimal sliding speed has been attained, the cancellation of the sliding surface leads to:

$$z_2(k) = -\overline{C}z_1(k) \quad (32)$$

The equivalent sliding mode controller is inferred from s(k +1) = 0:

$$\begin{aligned} u_{eq}(k) &= -\Gamma_2^{-1}((\overline{C}\overline{\Phi}_{11} + \overline{\Phi}_{21})z_1(k) \\ &+ (\overline{C}\overline{\Phi}_{12} + \overline{\Phi}_{22})z_2(k)) \end{aligned} \quad (33)$$

Based on expressions (29) and (30) the stabilizing control law is described as follows:

$$\begin{aligned} u_{eq}(k) &= -\Gamma_2^{-1}(\overline{C}\overline{\Phi}_{11} - \overline{C}\overline{\Phi}_{12}\overline{C} + \overline{\Phi}_{21} \\ &- \overline{\Phi}_{22}\overline{C})z_1(k)) \end{aligned} \quad (34)$$

In order to find the parameters of the sliding surface that generate the optimal control given by equation (28), the following is assumed:

$$u_{eq}(k) = Uop(k) \quad (35)$$

Taking into account the state transformation $z(k) = Tx(k)$ and based on equation (32), the new expression for the optimal control in (28) can be expressed as follows:

$$\begin{aligned} Uop(k) &= -K_1 T^T z(k) = -Kz(k) \\ &= -[K_1 K_2]\begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} = (K_1 - K_2\overline{C})\, z_1(k) \end{aligned} \quad (36)$$

According to equations (34) and (35), the new sliding surface is calculated by solving the following equation:

$$\overline{C}\overline{\Phi}_{12}\overline{C} + \left(\overline{\Phi}_{22} - \Gamma_2 K_2\right)\overline{C} - \overline{C}\overline{\Phi}_{11} + (\Gamma_2 K_1 - \overline{\Phi}_{21}) = 0 \quad (37)$$

This equation, with the unknown term $\overline{C}$ is known as the Sylvester equation.

The system below describes the new optimal sliding surface:

$$\begin{cases} s(k) = \begin{bmatrix} \overline{C} \\ I_{m\times m} \end{bmatrix}^T \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} - \begin{bmatrix} \overline{C} \\ I_{m\times m} \end{bmatrix}^T \begin{bmatrix} z_{10}(k) \\ z_{20}(k) \end{bmatrix} + h(k) \\ h(k) = h(k-1) - C\left(\overline{\Phi} - \overline{\Gamma}K - I\right)z(k-1) \end{cases} \quad (38)$$

with: $z_0 = \begin{bmatrix} z_{10} \\ z_{20} \end{bmatrix}$

K is the gain matrix, z_0 is an initial condition, and $h(0)$ is a null vector of order m.

5. Optimization of Control Parameters

After establishing the sliding surface structure and determining its matrix , the optimal control law is designed. In order to eliminate the convergence phase during which the system is sensitive to external disturbances, the control structure can be modified as follows:

$$\begin{aligned} u(k) &= -Kz(k) - \left(C\overline{\Gamma}\right)^{-1}[\Psi T_e \mathrm{s(k)} \\ &+ \left(\mathrm{G}T_e + \rho_{max}\left\|C\overline{\Gamma}\right\|\right) sign(s(k))] \end{aligned} \quad (39)$$

where Ψ and G are two positive diagonal matrices satisfying: $0 \le \Psi T_e < 1$ and $\mathrm{G}T_e > 0$.

The resulting control depends on the gain matrix K, which in turn is dependent on matrices Q and R. In order to anticipate the parameters leading to the synthesis of an optimal sliding mode control, genetic algorithms were employed as a multi-objective method for establishing a database for neural network learning. This makes it possible to predict an optimal solution for the design of

a control law that provides a balance between various objectives related to trajectory tracking.

5.1 Multi-objective Optimization

To solve this problem, heuristics is used to find the best solution to the problem, or at least, the least bad solution. The main idea behind heuristics is to explore the space of solutions in an attempt to converge to an optimal solution.

While sliding mode control is important, choosing the Q and R matrices for the LQR method (Raghda, Anouar & Wafa, 2022) can be challenging, as there is a trade-off between energy and speed that must be considered. The implementation of the switching signal requires the determination of the matrices Ψ and G (the discontinuous control gain). In this paper, this problem was reformulated as a multi-objective optimization. Multi-objective optimization problems are known as the most common processes in engineering, where multiple minimized apps are resolved. A compromise was reached in (Ebrahim et al., 2021). The Genetic Algorithm (GA) optimization was used because it can solve difficult problems using heuristics and meta-heuristics.

The definition of the objective function according to Boukadida et al. (2019) is:

$$f\left(x(t)\right) = w_1(t)\,ISE + w_2(t)\,log_{10}(Tv) \qquad (40)$$

The total variation of the control law is denoted by Tv. ISE is the Integrated Squared Error. w_1 and w_2 are the weights which are defined as:

$$w_1(t) = |\sin(\frac{2\Pi t}{F})|\ ,\ w_2(t) = 1 - w_1(t) \qquad (41)$$

F is the frequency controlling the rate of change for the weights.

5.1.1 Genetic Algorithms (GAs)

Genetic algorithms use Darwin's theory of the evolution of species. It is based on three principles: the principle of adaptation, the principle of variation and the principle of inheritance (Thengade & Dondal, 2012). There are three evolutionary operators in genetic algorithms:

Selection: Choosing the most appropriate individual. Crossover: Mixing by copying the particularities of selected individuals.

Mutation: random modification of the characteristics of an individual.

Figure 3 illustrates the genetic algorithm and its operational flow. A genetic algorithm begins with an initial population, evaluates the quality of each individual with respect to an objective, selects individuals based on their fitness, reproduces them through crossover, introduces diversity through mutation, and replaces the least fit individuals with new generations. This process is repeated until a satisfactory solution is achieved.

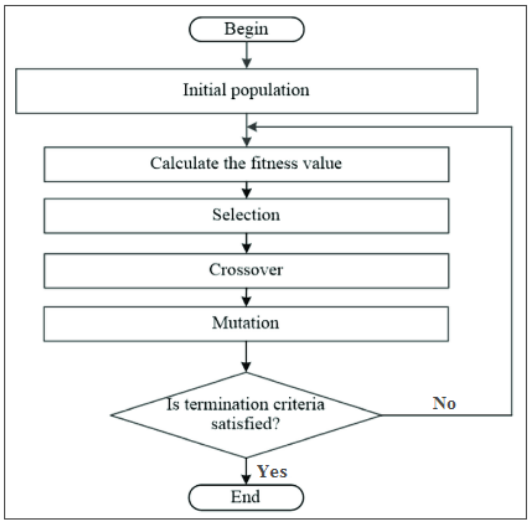

Figure 3. Procedure of GA (Albadr et al., 2020)

5.2 Multi-Layer Perceptrons (MLPs)

Multilayer perceptrons (MLPs) belong to the class of feedforward artificial neural networks. A MLP consists of at least three layers of nodes: an input layer, a hidden layer, and an output layer. Apart from input nodes, each node is a neuron that uses a non-linear activation function (Gerardin et al., 2009). MLPs use a supervised learning method called backpropagation to adjust their weights during training. The multiple layers and non-linear activations differentiate MLPs from linear perceptrons. MLPs are often called simple neural networks, especially when they have only one hidden layer (Figure 4).

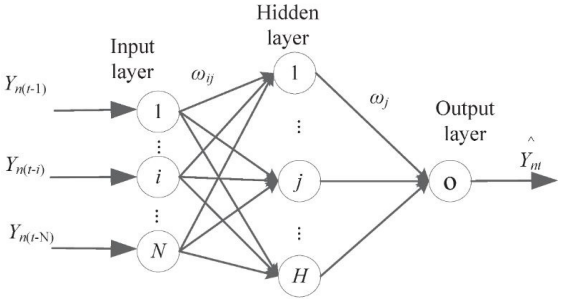

Figure 4. The Multilayer Perceptron (Wang et al., 2022)

The proposed network uses the outputs obtained by genetic algorithms as input data. This enables the development of a robust deep learning system capable of accurately predicting the two matrices Q and R, as well as the matrices Ψ and G. It is important to emphasize that neural networks are the essential tool in the deep learning process, allowing for the full exploitation of the benefits of this advanced approach in artificial intelligence.

Further on, this network assigns biases and weights to the different neurons in the network based on the input data.

This article focuses on regression, which involves predicting the most suitable values for Q, R, Ψ and G. The input data is represented by $X_{(i)}$, and then the corresponding values $Y_{(i)}$ are predicted. The networks are commonly trained using a traditional feed-forward backpropagation learning algorithm that minimizes the Mean Squared Error (MSE) of the training data. The input data for training the network is a database of observations that includes different values of Q, R, Ψ and G obtained through metaheuristic algorithms.

Table 2 illustrates the obtained experimental results and how the parameterization data was used in the training method. The parameters used in the regression are the learning rate (LR), number of epochs, number of hidden layers, number of neurons per hidden layer, and best validation performance (BVP).

In relation to Table 2, a multilayer network was used consisting of an input layer with 280 neurons and ten hidden layers, each composed of eight neurons with Purelin transfer functions.

Table 2. Experimental results obtained for the regression parameters

LR	No. of Epochs	No. of Hidden Layers (HL)	No. of neurons per HL	BVP
0.001	100	2	4	3.3864e-09
0.001	1000	2	4	6.9408e-13
0.001	1000	4	8	3.9724e-14
0.001	1000	10	8	1.676e-18
0.001	100	10	16	5.7537e-15
0.001	1000	4	16	6.9758e-12

6. Validation

The Quanser 2-DoF helicopter model is mounted on a fixed base with two propellers driven by two DC motors. It allows control of pitch and yaw. The corresponding workstations with two different controllers, namely SDOSMC and (SDOSMC+GA+DL) were simulated using MATLAB to show the effectiveness of the combination of SDOSMC, LQR, genetic algorithms and deep learning. Figure 5 illustrates the proposed methodology for predicting an optimal solution for the control law design. It is composed of 3 blocks. The GA block's role is to provide a set of data for Q, R, Ψ and G that serves as inputs for the NN network. A DL-based learning process is then conducted to predict best values for Q, R, Ψ and G for the design of the optimal control law.

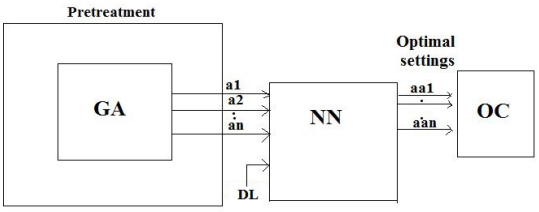

Figure 5. Diagram illustrating the proposed approach

Further on, the performance of the traditional optimal controller and that of the proposed deep learning-based optimal controller were compared. Figures 6 and 7 illustrate the evolution of two angles, θ and ψ. From these plots, it is evident that each output converges rapidly towards its reference, ensuring precise trajectory tracking for the proposed controller. It is clear that the robustness of the proposed approach makes the effect of disturbances negligible.

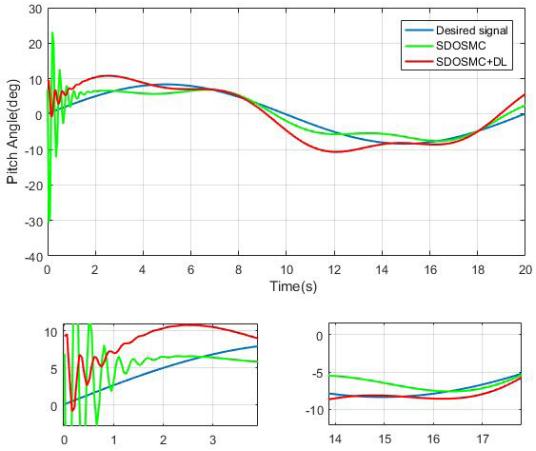

Figure 6. Evolution of the pitch angle for the two employed optimal controllers

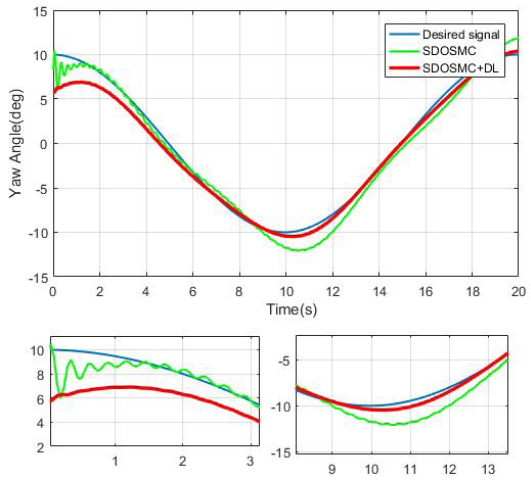

Figure 7. Evolution of the yaw angle for the two employed optimal controllers

Figures 8 and 9 schematically represent the pitch and yaw motor control. Figure 9 features a smooth curve in steady system state, indicating the absence of the chattering phenomenon.

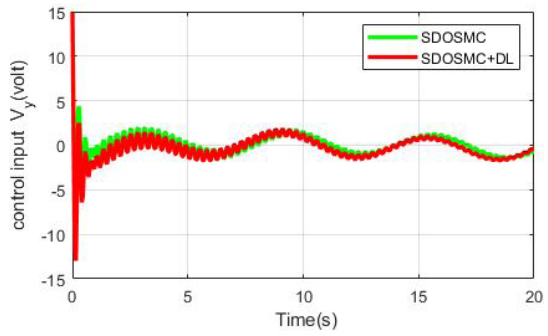

Figure 8. Evolution of the yaw motor control input voltage (Vy) for the two employed optimal controllers

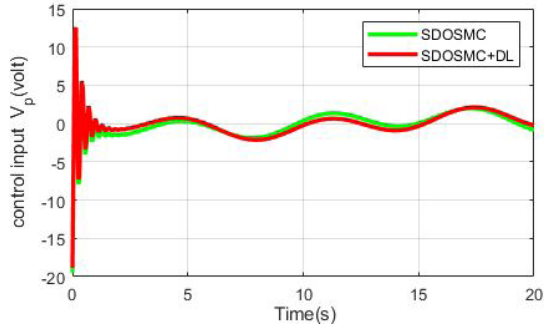

Figure 9. Evolution of the pitch motor control input voltage (V_p) for the two employed optimal controllers

These figures demonstrate that combining deep learning with GA and SDOSMC results in a faster convergence of the pitch and yaw angles to the desired signal.

In both cases, the employed approaches completely rejected the perturbations, and the related trajectories remained within a confined range of motion.

Table 3 below conclusively confirms the superior effectiveness of the proposed deep learning-based optimal control approach in comparison with that of the traditional optimal controller. The obtained results highlight a faster convergence, a higher trajectory tracking precision, and an enhanced robustness against disturbances. These findings further bolster the standing of the proposed method as a more effective and innovative solution for resolving tracking problems with parameter uncertainty.

The control architecture of the Quanser Aero helicopter seems to benefit from combining the LQR approach with intelligent methods, as this results in a fast convergence, the reduction of chattering, and a low oscillation amplitude. The simulation results demonstrate the effectiveness of this approach and the superiority of the proposed controller.

Table 3. Experimental results obtained for the two employed controllers

	SDOSMC	SDOSMC + DL
Convergence Time for (θ) (s)	≈ 2.2	≈ 1.1
Convergence Time for (ψ) (s)	≈ 10	≈ 0.2
Mean Error for (θ)	0.05	0.03
Mean Error for (ψ)	3.9×10^{-4}	3.7×10^{-4}
Chattering (θ)	++	–
Chattering (ψ)	++	–

7. Conclusion

This paper presents a new approach for designing a robust controller with the purpose of ensuring the reliable trajectory tracking of the pitch and yaw angles for a 2-degree-of-freedom helicopter. This article makes three key contributions.

The first is the development of an ideal sliding surface using a process that guarantees both the stability of the closed-loop system and the effectiveness of the control law. The second contribution involves designing a sliding mode controller coupled with an optimal global robust control system based on the LQR technique. The third contribution concerns the optimal controller configuration, and this is accomplished by employing a combination of intelligent techniques, including metaheuristics and deep learning, aimed at reducing the "chattering" phenomenon. The obtained experimental results demonstrate that this innovative approach can significantly enhance the closed-loop system's performance.

REFERENCES

Albadr, M. A., Tiun, S., Ayob, M. & Al-Dhief, F. (2020) Genetic Algorithm Based on Natural Selection Theory for Optimization Problems. *Symmetry.* 12(11), 1758. doi: 10.3390/sym12111758.

Boukadida, W., Benamor, A., Messaoud, H. & Siarry, P. (2019) Multi-objective design of optimal higher order sliding mode control for robust tracking of 2-DoF helicopter system based on metaheuristics. *Aerospace Science and Technology*. 91, 442-455. doi: 10.1016/j.ast.2019.05.037.

Cui, P., Gao, C., Jing, W. & An, R. (2022) Fault-Tolerant Control of Hypersonic Vehicle Using Neural Network and Sliding Mode. *International Journal of Aerospace Engineering.* 2022(6). doi: 10.1155:2022/1637305.

Das, M. & Mahanta, C. (2014) Optimal second order sliding mode control for linear uncertain systems. *ISA Transactions*. 53(6), 1807-1815. doi: 10.1016/j.isatra.2014.08.010.

Drazenovic, B. (1969) The invariance conditions in variable structure systems. *Automatica*. 5(3), 287-295.

Ebrahim, M. A., Ahmed, M. N., Ramadan, H. S., Becherif, M. & Zhao, J. (2021) Optimal metaheuristic-based sliding mode control of VSC-HVDC transmission systems. *Mathematics and Computers in Simulation.* 179, 178-193. doi: 10.1016/j.matcom.2020.08.009.

Gerardin, E., Chetelat, G., Chupin, M., Cuingnet, R., Desgranges, B., Kim, H.-S., Niethammer, M., Dubois, B., Lehericy, S., Garnero, L., Eustache, F. & Colliot, O. (2009) Multidimensional classification of hippocampal shape features discriminates Alzheimer's disease and mild cognitive impairment from normal aging. *Neuroimage*. 47(4), 1476-1486. doi: 10.1016/j.neuroimage.2009.05.036.

Hassen, M. D., Laamiri, I. & Messaoud, H. (2021) Exoskeleton Dynamic Modeling and Identification with Load and Temperature-Dependent Friction Model. In: *2021 IEEE 2nd International Conference on Signal, Control and Communication (SCC)*, *20-22 December 2021, Tunis, Tunisia*. IEEE. pp. 14-19.

He, K., Zhang, X., Ren, S. & Sun, J. (2016) Deep residual learning for image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 27-20 June 2016, Las Vegas, NV, USA.* IEEE. pp. 770-778.

Nadimpalli, M. (2017) Artificial Intelligence Risks and Benefits. *International Journal of Innovative Research in Science, Engineering and Technology*. 6(6), 1-6.

Nuthi, P. & Subbarao, K. (2015) Experimental verification of linear and adaptive control techniques for a two degrees-of-freedom helicopter. *Journal of Dynamic Systems, Measurement and Control.* 137(6), 064501. doi: 10.1115/1.4029273.

Utkin, V., Guldner, J. & Shi, J. (1999) *Sliding mode control in electromechanical systems*. London, Taylor & Francis. Niu, Y., Ho, D. W. C. & Wang, Z. (2010) Improved sliding mode control for discrete-time systems via reaching law. *IET Control Theory and Applications*. 4(11), 2245-2251. doi: 10.1049/iet-cta.2009.0296.

Pai, M.-C. (2014) Discrete-time sliding mode control for robust tracking and model following of systems with state and input delays. *Nonlinear Dynamics.* 76, 1769-1779. doi: 10.1007/s11071-014-1245-0.

Raghda, J., Anouar, B. & Wafa, B. (2022) Optimal sliding mode control for a pendulum system. In: *2022 International Conference on Control, Automation and Diagnosis (ICCAD)*, *13-15 July 2022, Lisbon, Portugal*. IEEE. pp. 1-6.

Thengade, A. & Dondal, R (2012) Genetic Algorithm - Survey Paper. *MPGI National Multi Conference 2012 (MPGINMC-2012),* 7-8 April 2012, Nanded, India. pp.25-29.

Tsai, S.-J., Huo, C.-L., Yang, Y.-K., Sun, T.-Y. (2013) Variable feedback gain control design based on particle swarm optimizer for automatic fighter tracking problems. *Applied Soft Computing*. 13(1), 58-75. doi: 10.1016/j.asoc.2012.07.032.

Utkin, V. I. (1978) *Sliding modes and their application in variable structure* systems. Moscow, Mir.

Wang, A., Li, Y., Yao, Z., Zhong, C., Xue, B., & Guo, Z. (2022). A Novel Hybrid Model for the Prediction and Classification of Rolling Bearing Condition. *Applied Sciences*, *12*(8), 3854. doi: 10.3390/app12083854 .

Yin, Y., Niu, H. & Liu, X. (2017) Adaptive Neural Network Sliding Mode Control for Quad Tilt Rotor Aircraft. *Mathematical Problems in Engineering.* 2017, 7104708. doi: 10.1155/2017/7104708.

Zhao, Y.-X., Wu, T. & Ma, Y. (2013) A Double Power Reaching Law of Sliding Mode Control Based on Neural Network. *Mathematical Problems in Engineering.* 2013, 1-9. doi: 10.1155/2013/408272. Zhang, J., Yang, X. & Yang, L. (2018) Virtual-command-based model reference adaptive control for abrupt structurally damaged aircraft. *Aerospace Science and Technology*. 78, 452-460. doi: 10.1016/j.ast.2018.04.043.