

Experimental Validation of the Generation of Direct and Quadratic Reference Currents by Combining the Ant Colony Optimization Algorithm and Sliding Mode Control in PMSM using the Process PIL

Adil Najem ^{a,1,*}, Ahmed Moutabir ^{a,2}, Abderrahmane Ouchatti ^{a,3}, Mohammed El Haissouf ^{b,4}

^aG.E.I.T.I.I.L laboratory, Ain chok Faculty of Sciences Casablanca, Hassan 2nd University, Morocco

^bDepartment of Applied Physics, MISI Laboratory, FST Settati, University Hassan 1st Settati, Morocco

¹adilnajem2@gmail.com; ²a.moutabir65@gmail.com; ³ouchatti_a@yahoo.fr; ⁴elhaissouf@gmail.com

* Corresponding Author

ARTICLE INFO

Article history

Received January 08, 2024

Revised February 21, 2024

Accepted March 08, 2024

Keywords

Sliding Mode Control;

PMSM;

Processor in the Loop;

Ant Colony Optimization;

ABSTRACT

This article aims to enhance the control efficiency of the Permanent Magnet Synchronous Motor (PMSM) by generating optimal reference currents i_{dref} and i_{qref} using Ant Colony Optimization (ACO), while ensuring a minimal absorbed current condition to reduce energy consumption and optimize PMSM performance. The ACO algorithm is chosen for its ability to find global solutions and robustness in complex environments, while Sliding Mode Control (SMC) provides advantages in terms of robustness against disturbances and the ability to maintain the system in a desired state. The implementation of the processor-in-the-loop (PIL) technique using MATLAB software with code composer and the LAUNCHXL-F28069M board enables the controller to be implemented in real hardware (LAUNCHXL-F28069M) to test the simulation environment (inverter and PMSM). Our results demonstrate the efficiency of ACO compared to the analytical method (AM) in terms of response time and minimizing absorbed current for different load values. Artificial intelligence (AI) has successfully and efficiently addressed the non-linearity between torque and reference currents, thus reducing energy consumption. This has allowed for the optimization of PMSM performance in a straightforward and efficient manner.

This is an open-access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



1. Introduction

The PMSM plays a central role in modern industrial applications due to its numerous advantages compared to other types of motors [1]-[3]. The PMSM offers high efficiency, high power density, and a significant torque-to-inertia ratio for high power applications. This type of motor is utilized in various applications such as electric cars, automation, and renewable energy systems [4]-[5].

Field-Oriented Control (FOC) involves converting three-phase alternating quantities (current and voltage) into a biphasic reference system called d-q, where both components, current and voltage, are independently controlled. This control consists of two loops often arranged in cascade: an external speed loop and an internal current loop. This structure is more resistant to disturbances and responds better to setpoints [6]. Compared to the Direct Torque Control (DTC) method, which

focuses directly on the electromagnetic torque of the motor, FOC allows for swift operation in all four quadrants and reduces torque and current ripple [7], [8].

Given the non-linearity of the PMSM, using linear control technology to achieve perfect control of the PMSM becomes challenging due to non-linearity, sensitivity to disturbances, and the high torque of the PMSM. Therefore, various non-linear control methods have been proposed, such as predictive control [9], [10], backstepping [11], [12], adaptive control [13], robust control [14], [15], synergistic control [16], [17], and SMC [18], [19]. All these approaches are designed to control nonlinear systems, and their positive aspects include:

- **Robustness:** Ensuring system robustness against parameter variations and external disturbances.
- **Adaptability:** Adjusting controller parameters based on changes in the system.
- **Performance:** Optimizing performance to track reference values, reject disturbances, and minimize errors.

One of the main advantages of SMC compared to other mentioned control methods is its robustness against disturbances and uncertainties in the model. Additionally, the use of sliding surfaces guides the system to a desired state independently of disturbances, maintaining reasonable performance even in the presence of unpredictable or unmodeled disturbances, which is often challenging with other approaches. One of the reasons guiding us to choose SMC is that SMC control can often be implemented relatively simply. Moreover, SMC generally aims to bring the system onto a specifically fast trajectory, offering rapid response times. In summary, SMC is preferred in situations where robustness against disturbances is crucial, and simplicity of implementation is important [20]-[22].

Control of a nonlinear system is also enhanced by AI approaches, such as Reinforcement Learning (RL) [23]-[25]. RL improves PMSM control without using controllers with highly complex mathematical equations. This approach doesn't require knowledge of the mathematical model of the system to be controlled. [26] In this article, RL replaces Proportional-Integral (PI) control in the internal current loop (*id* and *iq*) to achieve robust control. RL interacts with the environment, sending actions to the system. If the action is good, the RL agent receives a reward; otherwise, a penalty is given. Fuzzy logic [27], neuro-fuzzy control [28], Particle Swarm Optimization (PSO) [29], [30], and Genetic Algorithm (GA) [31], [32] are also used. A combination of these algorithms with the classic PI controller has made the control of a nonlinear system robust against disturbances, uncertainties, and nonlinearities within the system.

Control can also be improved by heuristic algorithms like the ACO algorithm, inspired by the social behavior of ants laying pheromones on the ground to find an optimal path. ACO uses a similar principle to solve optimization problems [33], [34]. In this article, the SMC acts on the speed and current loops by canceling the error between reference and measured values. The speed loop sends the reference electromagnetic torque to the ACO algorithm to generate reference currents (*i_{dref}* and *i_{qref}*) for the current loop, optimizing PMSM control.

In the literature, ACO has effectively optimized the optimal parameters of controllers for stable and precise output. Various articles demonstrate the combination of ACO and SMC. For example, in [35], SMC is combined with a conventional PI controller whose parameters are determined by ACO. Similar ideas are applied in [36] with FOC and SMC, and in [37] and [38], where ACO optimizes SMC parameters. [39] shows the efficiency of the ACO_SMC application compared to the GA_SMC approach. [40] This paper initiated the control of the PMSM using SMC with a non-zero direct reference current and also the addition of an equivalent control to reduce chattering, this paper aims is to generate direct and quadratic reference currents from the reference torque while minimizing the absorbed current using an analytical method.

Increasingly, research focuses on minimizing losses during the operation of an electrical drive, such as Joule losses caused by absorbed current, to save electrical energy. For certain applications,

the design criterion is based on minimal loss to achieve high efficiency and energy autonomy, as seen in electric cars or autonomous aircraft. Notations and symbols can be seen in Table 1.

The objective of this work is to generate optimal reference currents using the ACO algorithm without relying on mathematical calculations to optimize PMSM performance and reduce energy consumption. To develop our article, the paper is organized as follows: Method is presented in Section 2, Section 3 gives the results and Section 4 concludes.

Table 1. Notations and symbols

ω	Rotor speed
ω_{ref}	Rotor speed referential
C_M, C_r	Electromagnetic and load torques
v_d, v_q	Direct and quadratic stator voltages
i_d, i_q	Direct and quadratic stator currents
T_d, T_q	Direct and Quadratic time constants
PIL	Processor in the Loop
CCS	Code Composer Studio
R_s	Stator resistance
L_d, L_q	Direct and quadratic inductances
J, f	Inertia and coefficient of friction
$ITAE$	Integral of Time and Absolute Error
ψ_m	Permanent magnet flux
p	Number of pairs of magnetic poles

2. Method

2.1. PMSM Model

The diagram in Fig. 1 illustrates the different phases of PMSM modeling.

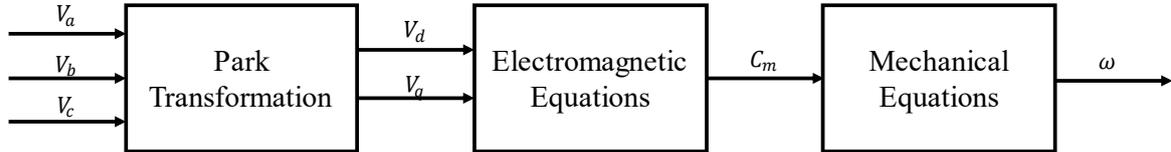


Fig. 1. PMSM model diagram

To control PMSM, it is very useful to develop a mathematical model to accurately describe its behavior. The model is described in a rotating d-q reference frame, with the state representation containing two state variables, namely the direct and quadrature stator currents (i_d and i_q), and a third variable which is the angular velocity ω .

$$\begin{cases} \frac{di_d}{dt} = -\frac{R_s}{L_d} i_d + \frac{L_q}{L_d} p\omega i_q + \frac{v_d}{L_d} \\ \frac{di_q}{dt} = -\frac{R_s}{L_q} i_q - \frac{L_d}{L_q} p\omega i_d - \frac{\psi_m}{L_q} p\omega + \frac{v_q}{L_q} \\ \frac{d\theta_e}{dt} = p\omega \\ \frac{d\omega}{dt} = -\frac{f}{J} \omega + \frac{C_M}{J} - \frac{C_r}{J} \end{cases} \quad (1)$$

We pose

$$\frac{1}{T_d} = \frac{R_s}{L_d}; \quad \frac{1}{T_q} = \frac{R_s}{L_q}; \quad \frac{1}{T_m} = \frac{f}{J} \quad (2)$$

Its electromagnetic state modulus is given by equation (3):

$$\frac{d}{dt} \begin{bmatrix} i_d \\ i_q \end{bmatrix} = \begin{bmatrix} -\frac{1}{T_d} & p\omega \frac{L_q}{L_d} \\ -p\omega \frac{L_d}{L_q} & -\frac{1}{T_q} \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} - p\omega \psi_m \frac{1}{L_q} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} \begin{bmatrix} v_d \\ v_q \end{bmatrix} \quad (3)$$

The mechanical equation of the PMSM is:

$$\frac{d\omega}{dt} = -\frac{1}{T_m} \omega + \frac{C_M}{J} - \frac{C_r}{J} \quad (4)$$

The electromagnetic torque is expressed as follows:

$$C_M = \frac{3p}{2} (\psi_m i_q + (L_d - L_q) i_d i_q) \quad (5)$$

Its equation of state equation (6) is deduced from equations (3), (4) and (5).

$$\frac{d}{dt} \begin{bmatrix} \omega \\ i_d \\ i_q \end{bmatrix} = \begin{bmatrix} -\frac{1}{T_m} \omega + \frac{3p}{2} (\psi_m i_q + (L_d - L_q) i_d i_q) - \frac{C_r}{J} \\ -\frac{1}{T_d} i_d + p\omega \frac{L_q}{L_d} i_q \\ -\frac{1}{T_q} i_q - p\omega \frac{L_d}{L_q} i_d - p\omega \psi_m \frac{1}{L_q} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} \begin{bmatrix} v_d \\ v_q \end{bmatrix} \quad (6)$$

2.2. Sliding Mode Control of the Permanent Magnet Synchronous Motor

SMC is insensitive to external disturbances and variations in internal parameters, but it is essential to reduce chattering. We will replace a sign function with a hyperbolic tangent function to reduce chattering. The use of the sign function leads to certain consequences such as abrupt transitions, i.e., rapid oscillations around the reference value, and poor quality of control signals and absorbed currents. Since the sign function is abrupt as shown in Fig. 2, it can make the system sensitive to disturbances. A justification using MATLAB is provided in the simulation section.

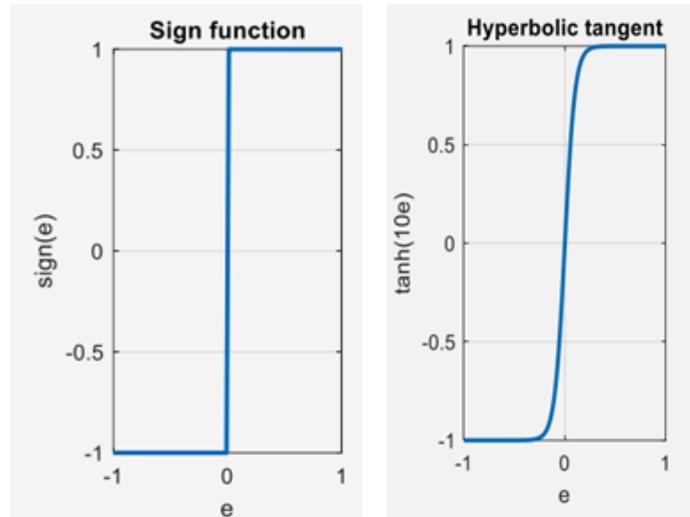


Fig. 2. Sign function and hyperbolic tangent

Fig. 3 contains two SMC loops, a speed loop and a current loop cascaded with a generator of the referential currents i_{dref} and i_{qref} , which is controlled by ACO algorithm.

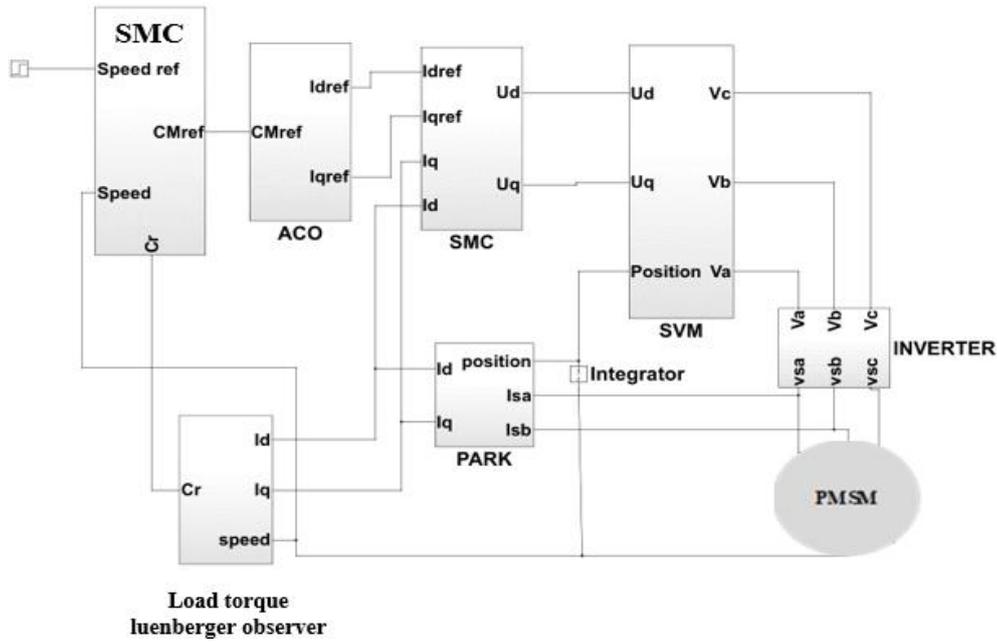


Fig. 3. Control system block diagram

We propose to divide the control synthesis into two stages, the first having slow dynamics (speed) and the other fast dynamics (current) in order to simplify the calculation of the control law.

2.2.1. Speed Loop

We are currently in the first synthesis stage, which is speed control aimed at determining the control law C_{Mref} . Firstly, we define the error e_v between the speed ω and the reference speed ω_{ref}

$$e_v = \omega - \omega_{ref} \quad (7)$$

The sliding mode surface is governed by the following equation:

$$s = e_v = \omega - \omega_{ref} \quad (8)$$

Its derivative is expressed by equation (9):

$$\dot{s} = \dot{e}_v = \dot{\omega} - \dot{\omega}_{ref} \quad (9)$$

We introduce the term of control.

$$\dot{s} = \dot{e}_v = -K_1 \text{sign}(e_v) \quad (10)$$

We take the positive Lyapunov candidate function V_1 , its derivative is negative for K_1 greater than 0, which satisfies the stability condition.

$$V_1 = \frac{1}{2} e_v^2 > 0 \rightarrow \dot{V}_1 = -K_1 e_v \text{sign}(e_v) < 0 \quad (11)$$

We then conclude the control law, from equations (9), (10) and (12) we can derive C_{Mref}

$$\dot{\omega} = -\frac{1}{T_m} \omega + \frac{C_M}{J} - \frac{C_r}{J} \quad (12)$$

$$C_{Mref} = J(-K_1 \text{sign}(e_v) + \frac{\omega}{T_m} + \frac{C_r}{J} + \dot{\omega}_{ref}) \quad (13)$$

We pose

$$G = \frac{1}{J}; F = -\frac{\omega}{T_m} - \frac{C_r}{J} \quad (14)$$

Then the control law is expressed by equation (15):

$$C_{Mref} = G^{-1}(-K_1 \text{sign}(e_v) - F + \dot{\omega}_{ref}) \quad (15)$$

By applying the integration of equation (10), we deduced the response time, which is expressed by equation (16).

$$t_{rv} = \frac{|e_v(0)|}{K_1} \quad (16)$$

From a speed response time $t_{rv}=100\text{ms}$ that we set, we were able to calculate the speed controller parameter, however, for an initial speed of 100 rad/s, the control parameter K_1 is calculated as follows:

$$K_1 = \frac{|e_v(0)|}{trv} = 1000 \quad (17)$$

2.2.2. Current Loop

We used the current loop to determine the control law U. This current loop is faster than the speed loop. We pose.

$$X_i = \begin{bmatrix} i_d \\ i_q \end{bmatrix}; X_{iref} = \begin{bmatrix} i_{dref} \\ i_{qref} \end{bmatrix}; U = \begin{bmatrix} v_d \\ v_q \end{bmatrix} \quad (18)$$

With current error

$$E_i = X_i - X_{iref} \quad (19)$$

The equation that expresses the electromagnetic modulus of state is:

$$\dot{X}_i = \begin{bmatrix} -\frac{1}{T_d} i_d + p\omega \frac{L_q}{L_d} i_q \\ -p\omega \frac{L_d}{L_q} i_d - \frac{1}{T_q} i_q - p\omega \psi_m \frac{1}{L_q} \end{bmatrix} + \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} U \quad (20)$$

We pose

$$A = \begin{bmatrix} -\frac{1}{T_d} i_d + p\omega \frac{L_q}{L_d} i_q \\ -p\omega \frac{L_d}{L_q} i_d - \frac{1}{T_q} i_q - p\omega \psi_m \frac{1}{L_q} \end{bmatrix} \quad (21)$$

$$B = \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} \quad (22)$$

Equation (20) becomes:

$$\dot{X}_i = A + BU \quad (23)$$

The derivative of the current error:

$$\dot{E}_i = \dot{X}_i - \dot{X}_{iref} \quad (24)$$

We introduce the term of control.

$$\dot{E}_i = A + BU - \dot{X}_{iref} = -K_2 \text{sign}(E_i) \quad (25)$$

With

$$K_2 = \begin{bmatrix} K_d & 0 \\ 0 & K_q \end{bmatrix} ; \text{sign}(E_i) = \begin{bmatrix} \text{sign}(i_d - i_{dref}) \\ \text{sign}(i_q - i_{qref}) \end{bmatrix} \quad (26)$$

The Lyapunov candidate function ensures stability when its derivative is negative, so all the coefficients of the K_2 matrix must be positive.

$$V_2 = \frac{1}{2} E_i^T E_i > 0 \rightarrow \dot{V}_2 = -E_i^T K_2 \text{sign}(E_i) < 0 \quad (27)$$

The control law is given by equation (28):

$$U = B^{-1}(-K_2 \text{sign}(E_i) - A + \dot{X}_{iref}) \quad (28)$$

Using a response time of 10ms, which is less than the speed, and a current of 10A, we obtained the coefficients K_d and K_q :

$$K_d = K_q = \frac{|\Delta i_d|}{trc} = \frac{|\Delta i_q|}{trc} \quad (29)$$

The parameters K_1, K_d, K_q are deduced based on a desired settling time. However, for very fast control, the parameters (K_1, K_d, K_q) have larger values, which results in an increase in chattering.

2.3. Ant Colony Optimization

2.3.1. Description

ACO is a computer technique based on the behavior of ants during the search for food, the communication between ants is done in an indirect way by means of a chemical substance called pheromone [41]. The pheromone is used to show the shortest route between the food source and their nest, and along this route, the pheromone is increasingly deposited on the shortest route to the food, and the pheromone is vaporized to prevent other ants from taking another route [42].

This technique is used to have global optimizations of complex problems. Dorigo was the first to develop the ACO algorithm, since then, this technique has been improved over time, the ACO can be combined with other methods so that it can be improved [43]-[45]. The probability that an ant k located at node i will choose to go to another node in the network is given by the following relation:

$$P_{ij}^k = \begin{cases} \frac{(\tau_{ij}^k)^\alpha (\eta_{ij}^k)^\beta}{\sum_{l \in N_i^k} (\tau_{il}^k)^\alpha (\eta_{il}^k)^\beta} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases} \quad (30)$$

τ_{ij}^k Presents pheromone levels, denominator is a summation to account for all possible cases, N_i^k the set of possible trails for an ant k when in node i . α , β are parameters that influence the pheromone evaporation process and the behavior of ants, such as α aims to control the importance of the amount of pheromone by ants when they choose their next step. β controls the importance of the

heuristic (information about the problem) when ants choose the next step. A high value of α gives more weight to the quantity of pheromone, and a high value of β gives more weight to heuristic information, as well as for low value of α means that ants are less influenced by pheromone and are likely to explore new paths.

Similarly, a low value of β means that ants rely more on information. The choice of α , β depends on the speed at which we want the ants to converge to an optimal solution, and the two values attempt to balance between exploration and exploitation. η_{ij}^k represents heuristic information (visibility) [46]. ρ Represents the percentage of pheromone vaporization (evaporate rate).

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} \text{ avec } 0 \leq \rho \leq 1 \quad (31)$$

Pheromone levels are updated when ants have just crossed the path.

$$\tau_y \leftarrow \tau_y + \sum_{k=1}^m \Delta\tau_y^k \text{ avec } \Delta\tau_{ij}^k = \frac{1}{C^k} \quad (32)$$

C^k is associated with the reward of ant k for choosing this path. We used ACO to have β_1 and β_2 which give optimal performance and minimal error [47].

$$ITAE = \int_0^t t|E|dt \quad (33)$$

For an ant k in a population N , the coefficients β_1 and β_2 for an optimal solution are in the interval $[lb, ub]$ that we divide at steps, the smaller the steps the more optimal the solution, but the calculation time becomes longer. ζ represents a parameter introduced to reinforce pheromone when ants approach optimal solutions [48].

2.3.2. Procedure

[49] Step 1: Initialize the pheromone τ , all discrete values have the same pheromone values.
Step 2: Calculate the probability.

$$p_j^k = \frac{\tau_j}{\sum_{j=1}^m \tau_j} \quad (34)$$

Step 3: Find the cumulative probability ranges associated with different discrete values according to their probability. Step 4: N random numbers each included in the interval $(0,1)$, one for each ant. Step 5: Pheromone update: after determining β_1 and β_2 for each ant, the best value is calculated f_{best} and also the bad value f_{worst} , then the pheromone must be added for the best value.

$$\tau_j^{new} = \tau_j^{old} + \sum_k \Delta\tau_j^{(k)} \quad (35)$$

For values with poor results, we have to decrease the pheromone

$$\tau_j^{new} = (1 - \rho)\tau_j^{old} \quad (36)$$

We repeat this procedure for several iterations.

2.4. Estimating the Resistive Torque

[50] In order to have a more rigorous control, it is mandatory to introduce the load torque, however, this torque is not measurable, so the design of a load torque observer is necessary to send the observed values back to the controller, the variables are selected as follows:

$$x = \begin{bmatrix} \omega \\ C_r \end{bmatrix}; U = C_M; y = \omega \quad (37)$$

The general form of state space:

$$\begin{cases} \dot{x} = Ax + BU \\ y = Cx + DU \end{cases} \quad (38)$$

Where A, B, C, D are matrices that depend on the PMSM parameters.

$$\begin{cases} \begin{bmatrix} \dot{\omega} \\ \dot{C}_r \end{bmatrix} = \begin{bmatrix} -f & -1 \\ J & J \end{bmatrix} \begin{bmatrix} \omega \\ C_r \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} C_M \\ y = [1 \quad 0] \begin{bmatrix} \omega \\ C_r \end{bmatrix} \end{cases} \quad (39)$$

We follow equation (40) for the observer design:

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + BU + L(y - \hat{y}) \\ \hat{y} = C\hat{x} \end{cases} \quad (40)$$

With

$$L = \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} \quad (41)$$

Replacing (39) in (40) gives (42):

$$\begin{cases} \dot{\hat{\omega}} = \frac{1}{J}(C_M - \hat{C}_r - f\hat{\omega} + JL_1(\omega - \hat{\omega})) \\ \dot{\hat{C}}_r = L_2(\omega - \hat{\omega}) \end{cases} \quad (42)$$

We have taken the eigenvalues of the matrix $\begin{bmatrix} -\frac{f}{J} - L_1 & -\frac{1}{J} \\ -L_2 & 0 \end{bmatrix}$ less than 0, so that the load torque is close to the actual value:

$$\left[\lambda I - \begin{bmatrix} -\frac{f}{J} - L_1 & -\frac{1}{J} \\ -L_2 & 0 \end{bmatrix} \right] = 0 \quad (43)$$

$$\lambda^2 + \left(\frac{f}{J} + L_1\right)\lambda - \frac{L_2}{J} = 0 \quad (44)$$

We take α_1 and α_2 two negative eigenvalues satisfy the following equation:

$$\lambda^2 - (\alpha_1 + \alpha_2)\lambda + \alpha_1\alpha_2 = 0 \quad (45)$$

From equations (44) and (45) we deduce L_1 and L_2 .

$$L_1 = -\left(\alpha_1 + \alpha_2 + \frac{f}{J}\right) \quad (46)$$

$$L_2 = -(\alpha_1\alpha_2J) \quad (47)$$

The coefficients α_1 and α_2 are adjusted so that the value of the load speed and torque are close to the actual value.

2.5. Generating References Currents

There are several criteria for generating the referential currents i_{dref} and i_{qref} , the simplest being to cancel the i_{dref} current to make the referential electromagnetic torque proportional to i_{qref} . Another example of a criterion cited in this article [40] is the generation of the two currents i_{dref} and i_{qref} based on the criterion of minimizing the absorbed current. The authors defined the relationships that link i_{dref} and i_{qref} to the referential electromagnetic torque while respecting the criterion cited (the minimum current). The AM method used in this article consists of replacing the torque expression with a simplified expression (48).

$$\gamma = y(1 - x) \quad (48)$$

With

$$\gamma = C_{Mref} \frac{2(L_q - L_d)}{3p\psi_M^2} ; x = \frac{L_q - L_d}{\psi_M} i_d ; y = \frac{L_q - L_d}{\psi_M} i_q \quad (49)$$

From the derivation operation, we determine the relationships between the referential currents and the torque to obtain the minimum absorbed current.

$$z^2 = x^2 + y^2 \quad (50)$$

$$\frac{dz^2}{dx} = 0 \rightarrow \gamma^2 = -x(1 - x)^3 \quad (51)$$

$$\frac{dz^2}{dy} = 0 \rightarrow \gamma = \frac{y}{2}(1 + \sqrt{1 + 4y^2}) \quad (52)$$

So, we can derive the approximate expressions (53) and (54) from the instruction *polyfit* of *Matlab*.

$$i_{dref} = -0.006C_{Mref}^2 \quad (53)$$

$$i_{qref} = 0.4C_{Mref} \quad (54)$$

For our paper, we will respect the same generation criterion (the minimum current), using the ACO algorithm. Equation (5) shows that the electromagnetic torque is not proportional to i_{dref} and i_{qref} . Our idea is to determine the two gains, which allow us to give for each referential electromagnetic torque a current i_{dref} and i_{qref} , which translates into two equations (55) and (56):

$$i_{dref} = -\beta_1 C_{Mref} \quad (55)$$

$$i_{qref} = \beta_2 C_{Mref} \quad (56)$$

ACO algorithm :

- Initialize the number of ants.
- Define the search space containing the possible solutions [a, b], which will be as large as possible; for our article we have chosen it between 0 and 1000.
- We have taken a step h of 0.001 to obtain an optimal solution, although the calculation time is long.
- Initialize all discretized values to the same pheromone value τ which is 1.
- Calculate the probability for each discrete variable (34).

- On the basis of the probabilities of different discrete values, find the cumulative probability range associated with them.
- Generate N random numbers from 0 to 1, using this value to determine the discrete value included in the search space $[a, b]$.
- Replace the discrete values in the system to be controlled in order to calculate the ITAE error, in our case the absolute peak value of the absorbed current must be compared with the value 0, in order to verify the imposed condition of having a minimum absorbed current, we must also compare the speed of the motor with the reference speed 100rad/s.
- Determine the best value f_{best} and the worst value f_{worst} for each ant, strengthen the pheromone for the best value (35) and weaken the pheromone for the worst value (36).
- The algorithm ends after a number of iterations.

Fig. 4 illustrates the strategy of the ACO algorithm for determining the coefficients β_1 and β_2 in order to achieve an optimal solution. Indeed, ACO tests all possible combinations of β_1, β_2 and selects the best combination. Translates the algorithm shown in Fig. 5.

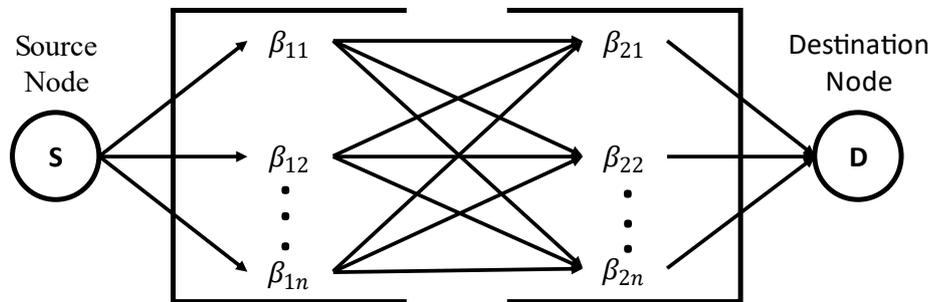


Fig. 4. Graphical of ACO tuning coefficients

After training in MATLAB, we were able to obtain both gains while meeting the minimum current criterion.

$$\beta_1 = 0.102 \quad (55)$$

$$\beta_2 = 0.445 \quad (56)$$

2.6. Simulation and Discussion

In this section, we focus on the Software-in-the-Loop (SIL) technique where the control operation runs on MATLAB. This is a preliminary step before moving to the Processor-in-the-Loop (PIL) technique. The goal of this part is to test our controller before transferring it to real hardware (LaunchX F28069M) while checking chattering as well as speed estimation and load torque.

We set the reference speed to 100 rad/s, and the load torque varies from 0 to 10 Nm at $t=0.25s$. This part shows the behavior of the SMC controller associated with the ACO algorithm, generating reference currents while respecting the condition of having minimal absorbed current, and comparing ACO with AM.

2.6.1. Reduced Chattering

At the beginning, we will verify chattering using the sign function in the SMC control law. Fig. 6 and Fig. 7 reveal the oscillations of speed and torque around the reference. Fig. 8, Fig. 9, and Fig. 10 show the poor quality of control signals and absorbed currents. Consequently, there is a need to implement solutions to reduce chattering. To address this chattering issue, we will replace the sign function with hyperbolic tangent. The figures show a significant reduction in chattering, along with

precise and stable signals. Fig. 11 and Fig. 12 show the reduction of ripples; indeed, the speed and torque stabilize around the operating point.

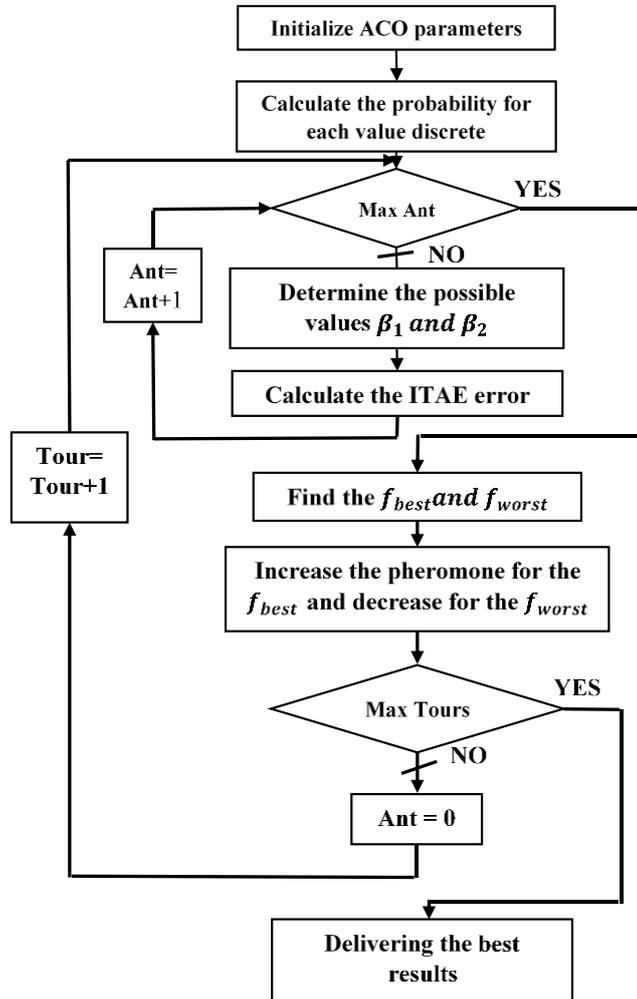


Fig. 5. Flowchart for ACO algorithm

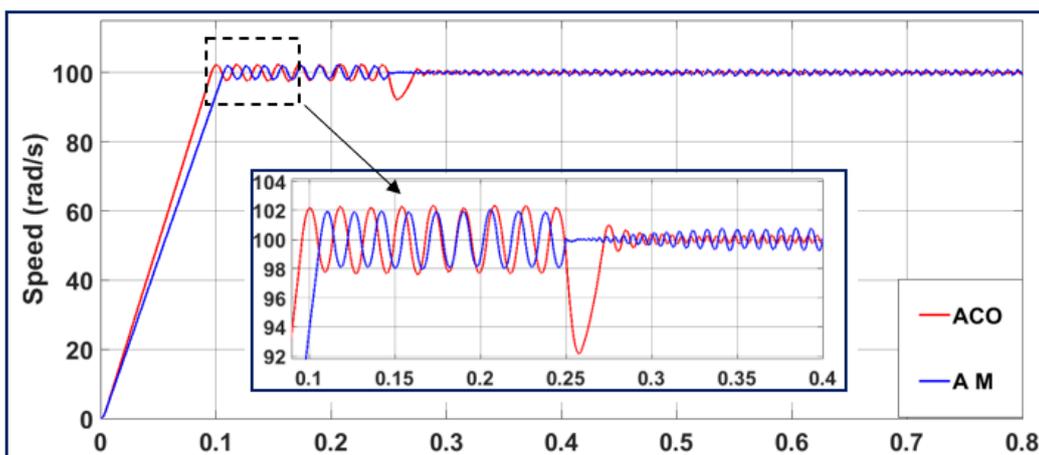


Fig. 6. Speed response by MATLAB/Simulink

The Fig. 13, Fig. 14 and Fig. 15 illustrates the quality of control signals for normal operation, including currents that do not contain fluctuations that could disturb the control of the PMSM.

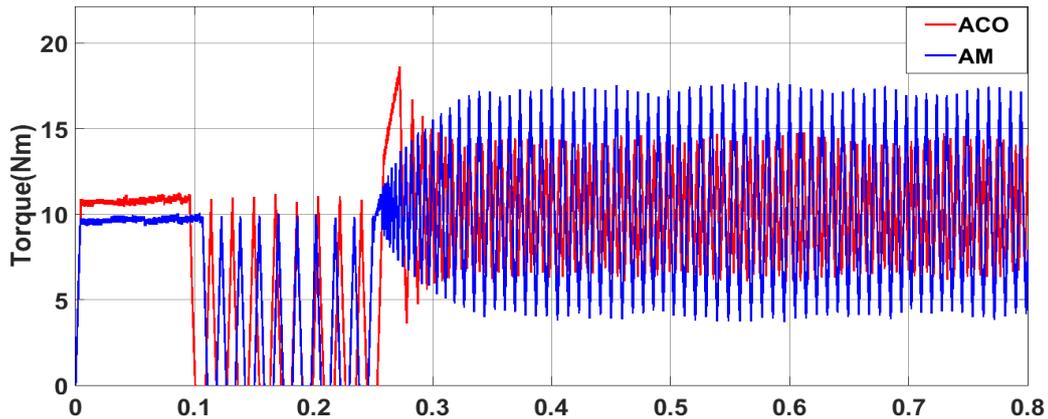


Fig. 7. Torque response by MATLAB/Simulink

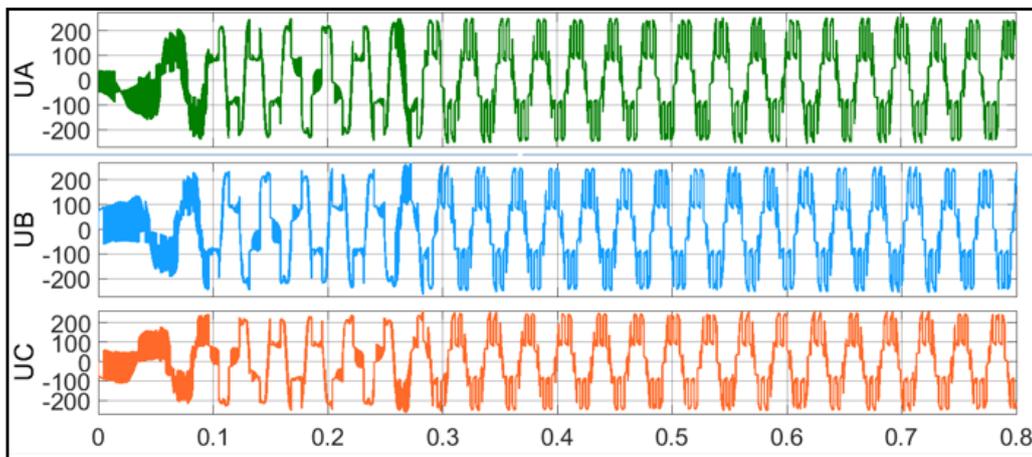


Fig. 8. Control signals response by MATLAB/Simulink

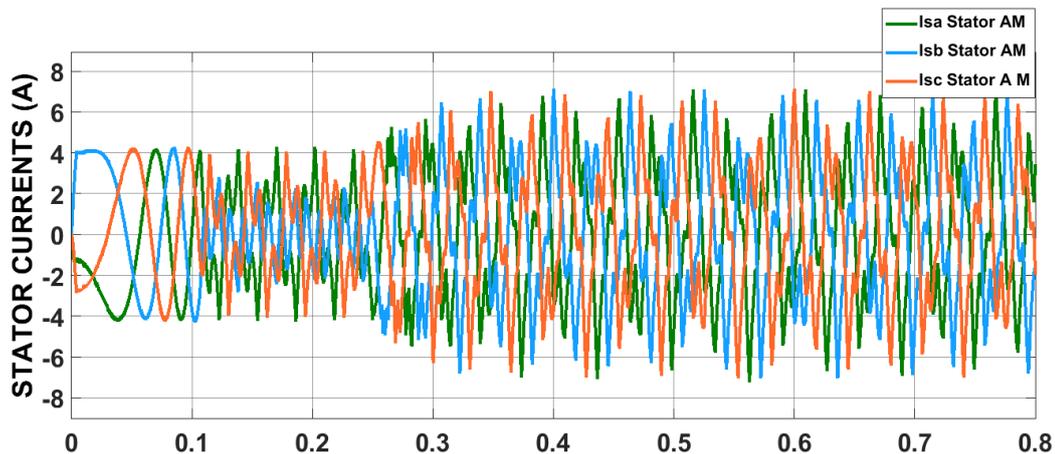


Fig. 9. Stators currents response by MATLAB/Simulink

2.6.2. Speed and Torque Estimator

The purpose of the estimator is to provide accurate estimates of the motor's speed and load torque. It should be able to quickly track changes in speed and torque. To closely examine our estimator, we will compare the actual motor speed with the estimated speed, and similarly for the torque.

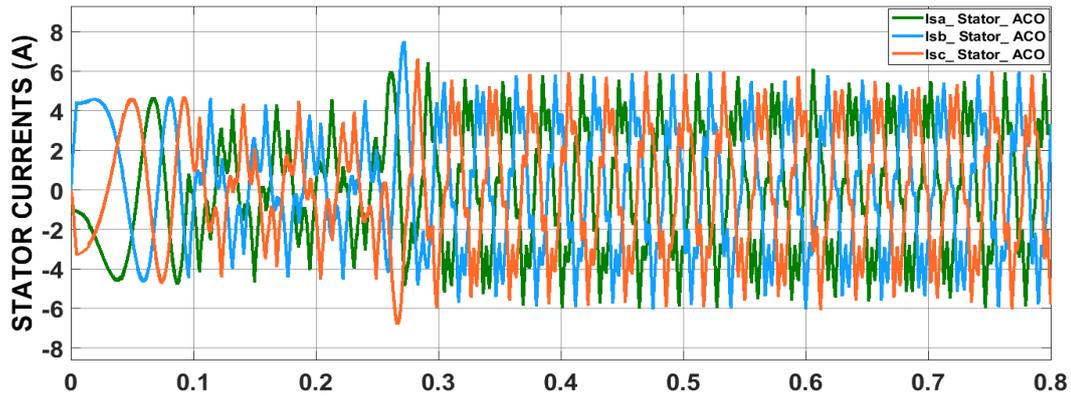


Fig. 10. Stators currents response by MATLAB/Simulink

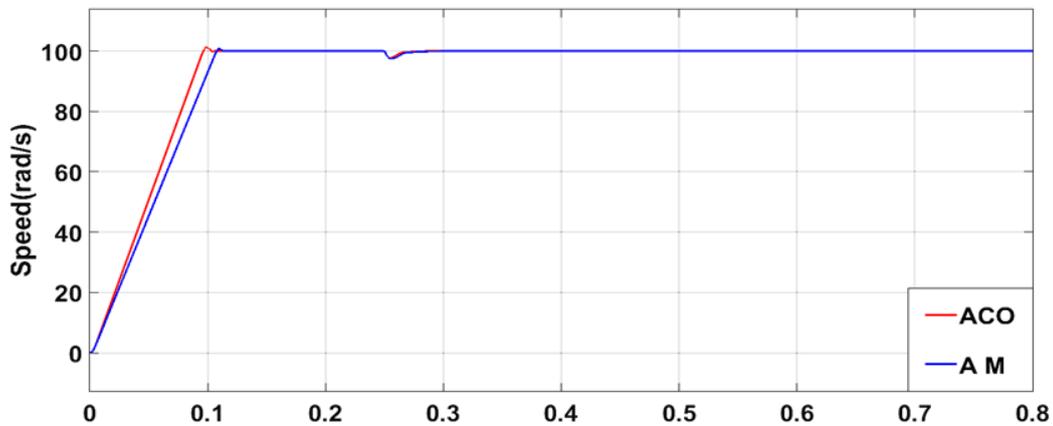


Fig. 11. Speed response by MATLAB/Simulink

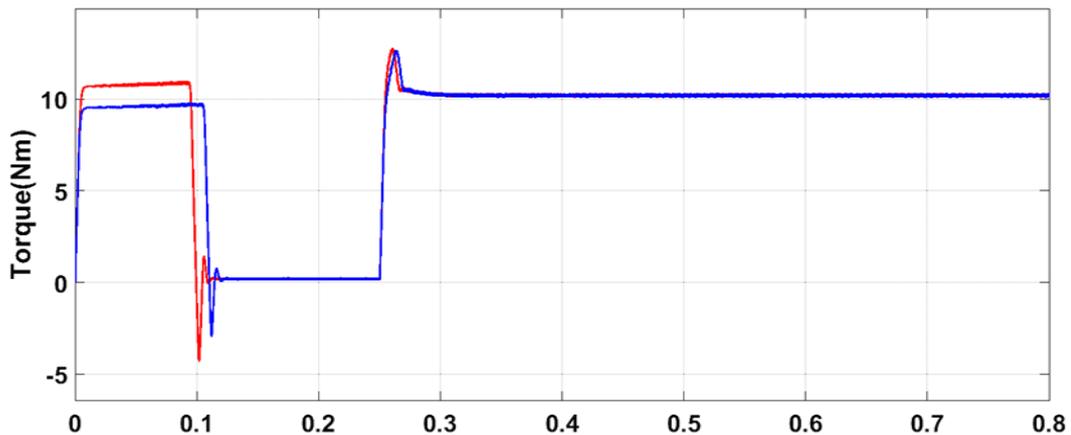


Fig. 12. Torque response by MATLAB/Simulink

The Fig. 16 and Fig. 17 demonstrate the accurate estimation of speed and torque. The actual values match the estimated values except during transition point. This is achieved through eigenvalue tuning using a trial-and-error method.

The speed and torque estimator provides accurate information that is then utilized by the control regulator to adjust commands and optimize the performance of the PMSM. By using an estimator in a control system instead of a speed sensor, which can be costly to install and maintain, the system costs and complexity can be reduced while still providing acceptable performance. Sometimes, a combination of a speed sensor and an estimator can be employed to enhance the reliability and robustness of the system.

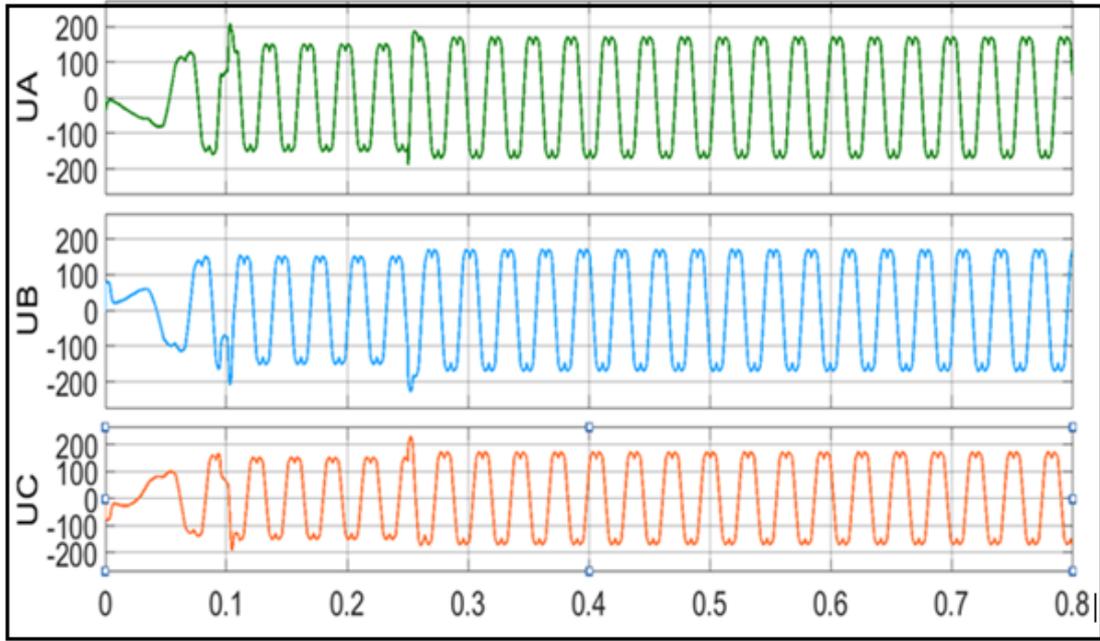


Fig. 13. Control signals response by MATLAB/Simulink

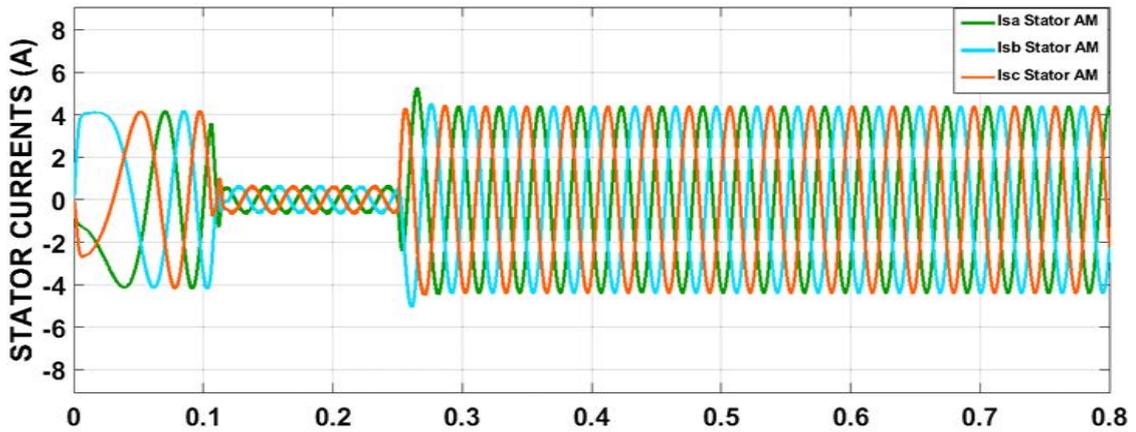


Fig. 14. Stators currents response by MATLAB/Simulink

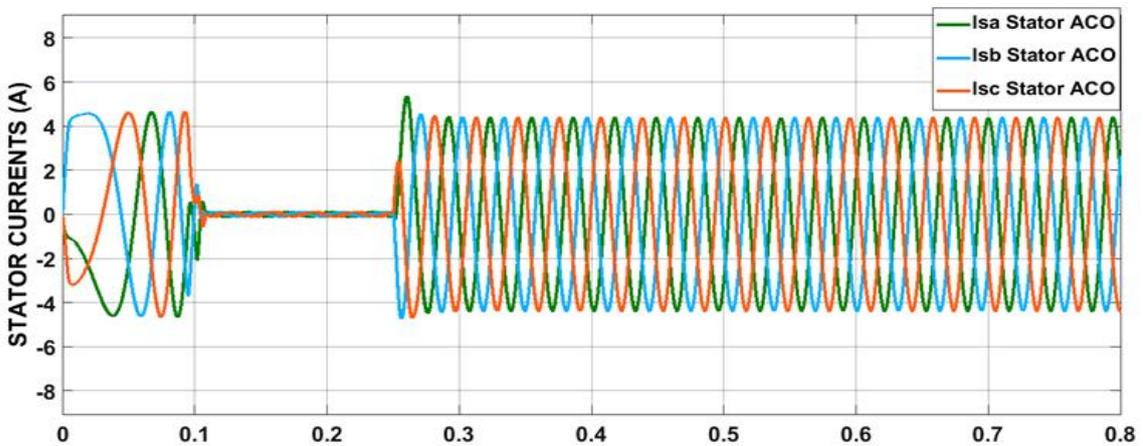


Fig. 15. Stators currents response by MATLAB/Simulink

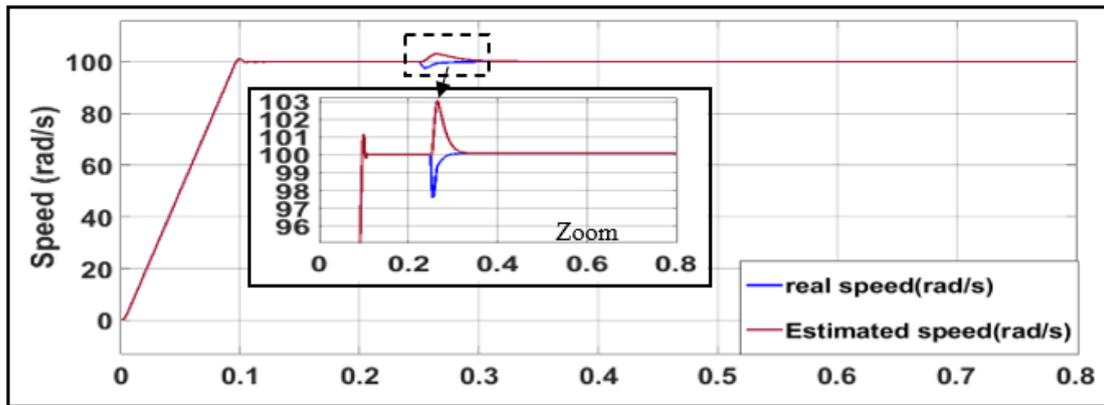


Fig. 16. Speed estimation by MATLAB/Simulink

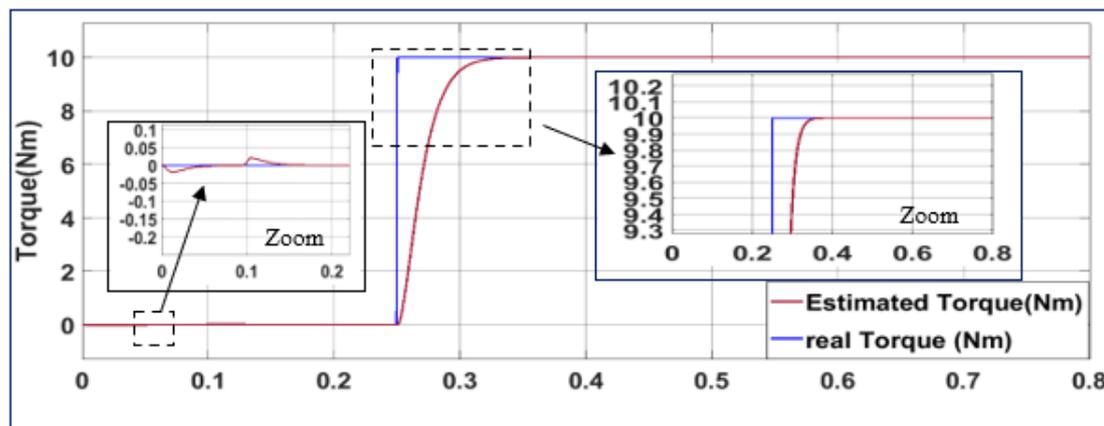


Fig. 17. Torque estimation by MATLAB/Simulink

3. Results and Discussion

We started with the SIL process, which requires execution in a simulated environment (MATLAB), as well as configuring simulation parameters such as initial conditions, a sampling time of $100\mu\text{s}$, and simulation inputs to attempt simulation under real conditions. Once the simulation is validated, we automatically generate Simulink code for real hardware such as the DSP LaunchX F28069M designed by Texas Instruments. DSP is a platform based on the C28x architecture, providing a combination of computing power and integrated peripherals. It is equipped with connectors that facilitate integration into various systems.

The next step is the PIL process, a development methodology that allows testing the controller on real hardware while leveraging the Simulink environment (inverter and PMSM) Fig. 18.

The PIL process is implemented with the DSP LaunchX F28069M through the following steps:

- Configure the development environment, including the installation of software provided by Texas Instruments, such as Code Composer Studio (CCS).
- Select the hardware configuration panel and then the development board, choosing options and configuring SCI mode externally (serial communication) Fig. 19.

The serial communication between the PC and DSP was done via a USB port with a fixed transmission speed of 921600 bps. For smoother communication, a MATLAB script is added, as shown in the Fig. 20. Modulation and sampling frequencies are set at 10 kHz. $f_{PWM} = f_{SMP} = 10\text{KHz} \rightarrow T_{SMP} = 100\mu\text{s}$

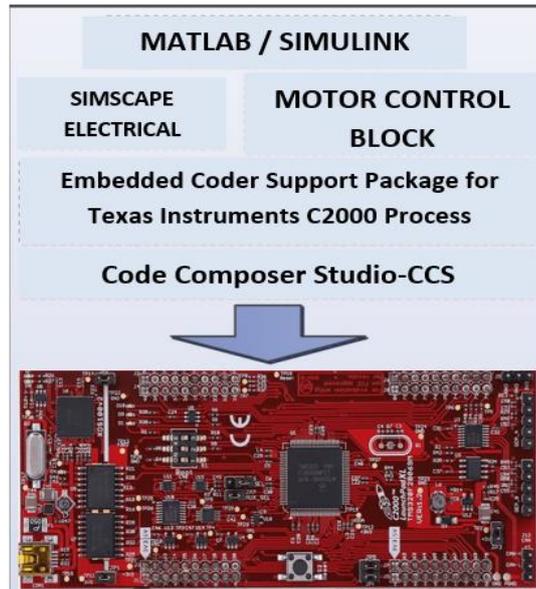


Fig. 18. Architecture for controlling the PMSM-PIL application

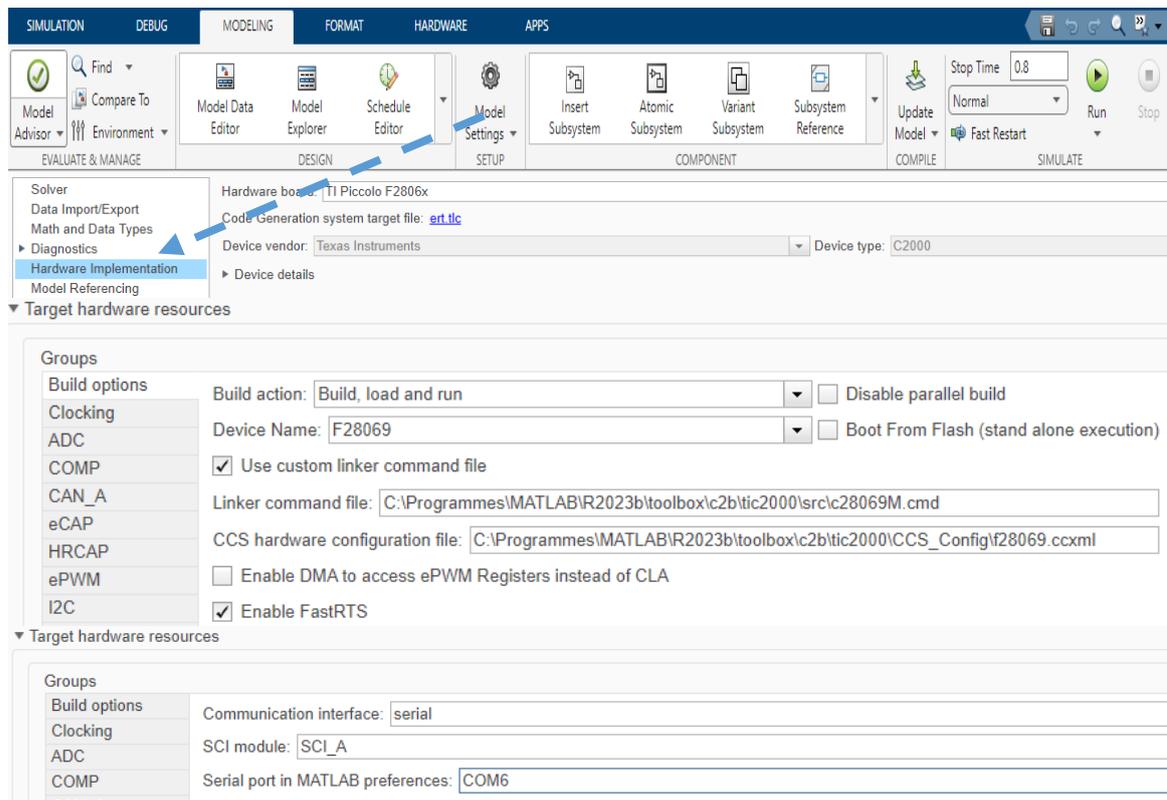


Fig. 19. Setting parameters for the PIL stage of the PMSM control system

```
setpref('MathWorks_Embedded_IDE_Link_PIL_Preferences', 'COMPort', 'COM6');
setpref('MathWorks_Embedded_IDE_Link_PIL_Preferences', 'BaudRate', 921600);
setpref('MathWorks_Embedded_IDE_Link_PIL_Preferences', 'enableserial', true);
```

Fig. 20. Serial communication parameters in MATLAB

Through the scheme described in the Fig. 21 and for a speed reference of 100 rad/s and a load torque ranging from 0 to 10 Nm at 0.25s, as well as the PMSM motor with characteristics mentioned in Table 2, we can begin the PIL process.

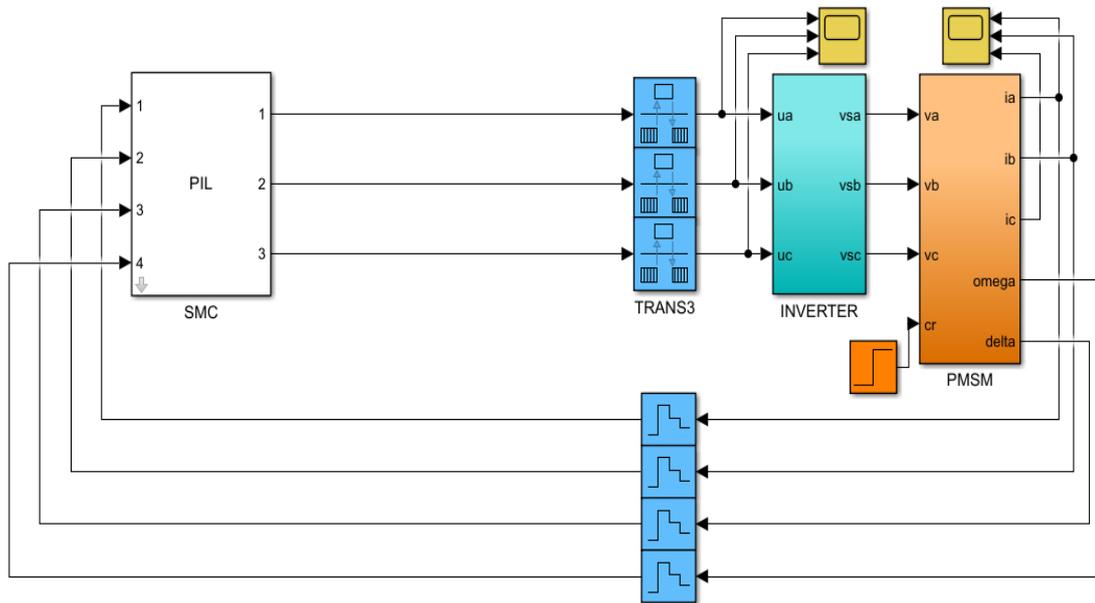


Fig. 21. Processor-in-Loop (PIL) implementation Set-up

Table 2. Machine parameters

Parameter	Value
Rated output power	1500 W
Rated load torque	10 Nm
Rated speed	1500 rpm
Rated voltage	220/380V
Resistance R_s	2.5 Ω
Direct inductance L_d	0.025H
Quadratic inductance L_q	0.075H
Moment of inertia J	0.01 kg. m ²
Coefficient of friction f	0.002 N.m. s/rad
Time constants T_d	0.01 s
Time constants T_q	0.03 s
Mechanic time constant T_m	5 s

Fig. 22 showing the speed evolution using the ACO algorithm and the AM method as described in [40]. Both approaches meet the 100 ms speed loop response time, exhibit good precision, and have low overshoot, less than 1.2% for both methods, as confirmed by Table 3.

In practical terms, it is crucial to have both low overshoot and a shorter response time to ensure that the system reaches a stable state without excessive oscillations. This is essential for maintaining stability and precision. A quick response enables the system to adapt rapidly to real-time setpoint changes, ensuring optimal performance. The combination of a shorter response time and low overshoot contributes to better precision and reduces vibrations in mechanical systems, ultimately enhancing system longevity.

Fig. 23 illustrates the three phases: during start-up, the difference between the electromagnetic torques for the two approaches is almost 1Nm. The torques remain constant because the speed variations are entirely constant during this start-up phase. After the end of the start-up phase, the oscillation is less undulating (reduced chattering). This is due to the hyperbolic tangent function in the sliding mode control, which replaces the sign function. Indeed, chattering is a phenomenon that

must be eliminated as it leads to additional losses in the motor, such as losses due to Foucault currents and hysteresis losses. A reduction in these losses results in a decrease in electrical energy consumption and improved performance. Chattering introduces disturbances in the operation, causing the output to oscillate around its reference, reducing its ability to effectively follow the setpoint. Another drawback sometimes problematic in certain applications introduced by chattering is the noise and vibrations in the motor. In the second and third phases, the load torque increases from 0 to 10Nm, the two approaches are similar.

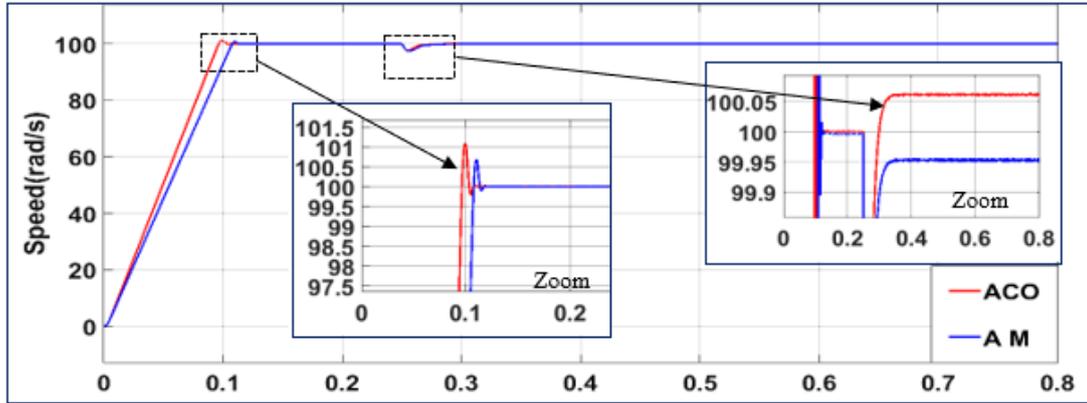


Fig. 22. Speed response by PIL

Table 3. Comparative analysis using ACO and AM

Method	Time and Peak Overshoot			Settling Time ts (sec)
	Rise Time tr (sec)	Peak overshoot MP (%)	Peak Time tp (sec)	
AM	0.084	0.7	0.11	0.1
ACO	0.075	1.1	0.1	0.094

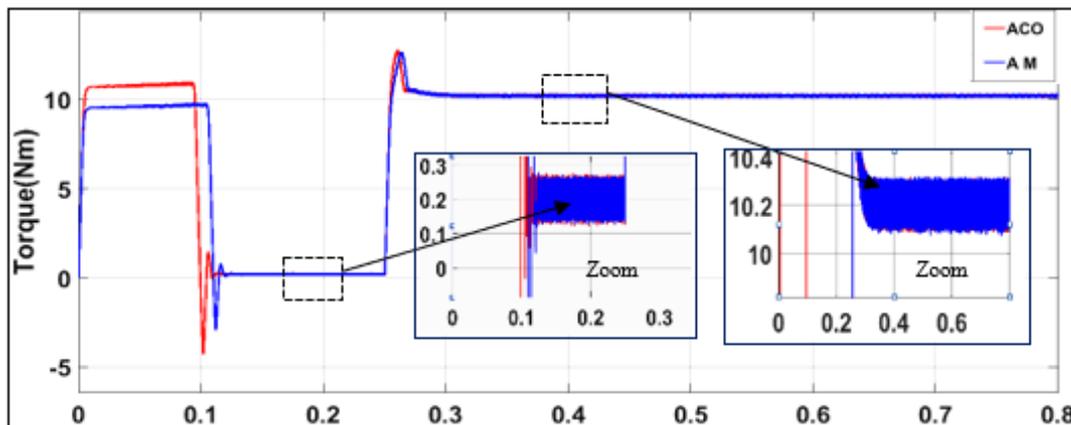


Fig. 23. Torque response by PIL

According to equation (5), the torque is not proportional to the direct and quadratic currents, which makes control less easy than in the case where the direct current is zero. Both approaches (ACO and AM) have made the torque proportional to the quadratic current, since the current and torque curves have the same shapes (Fig. 23 and Fig. 24). Fig. 25 and Fig. 26 show the stator currents for the two approaches, during the start-up phase the peak values of the currents for the ACO algorithm are between 4.5 and 4.6A, these values are very close to the values for the AM method, which vary between 4.2 and 4.4A.

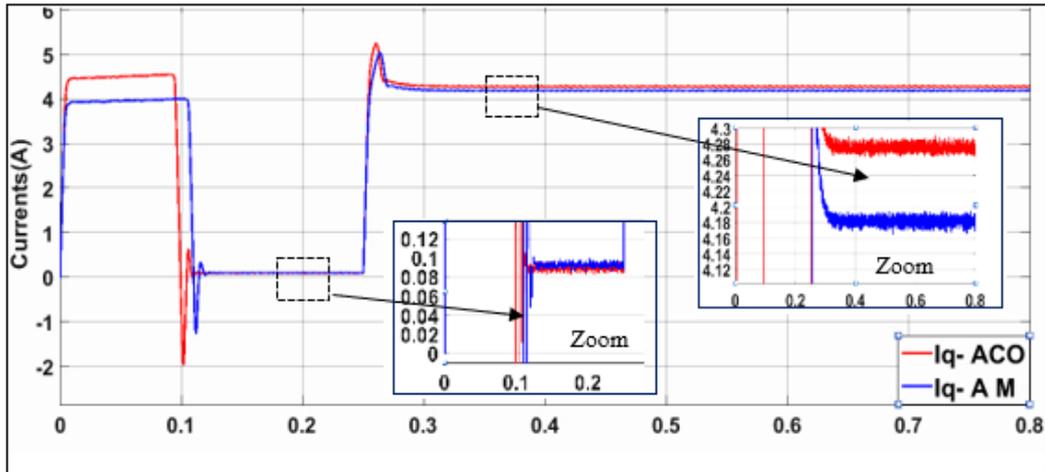


Fig. 24. Iq- current response by PIL

For no-load operation, ACO has proven to be more efficient than AM, as the maximum current values for ACO are lower than those for AM. This is because ACO is an algorithm that seeks to find the best solution in a complex search space, it can dynamically adapt to a changing environment, and operating conditions can vary according to load. In the case of no-load operation, the load torque is zero, making the optimization environment less complex. The ACO was able to better optimize absorbed currents, this is down to the choice of α and β , which must balance between exploration and operation. We chose $\alpha=1$ to reinforce the pheromone, meaning ants don't explore other paths, and $\beta=0$ to be cautious of solutions. Adjusting α and β is a difficult task. After the load torque was increased from 0 to 10Nm, the currents are similar for both ACO and AM.

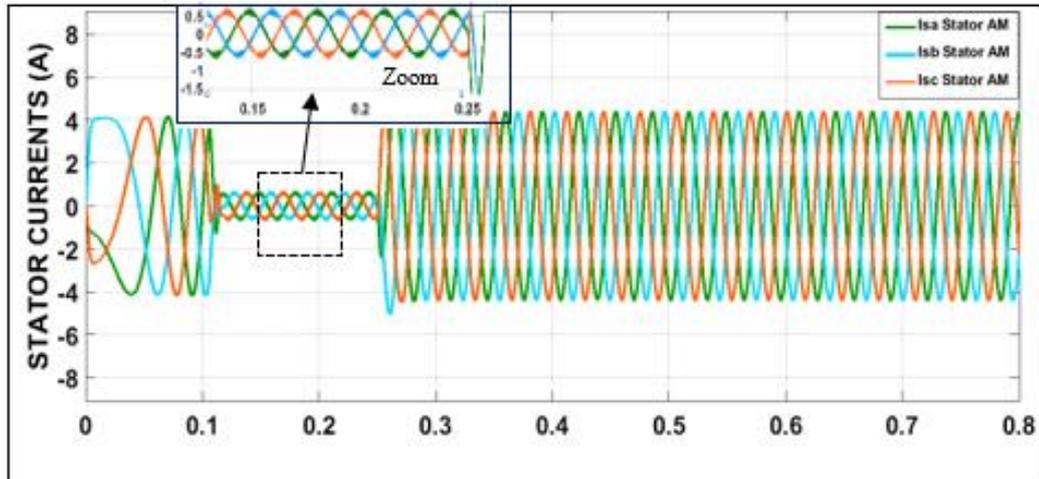


Fig. 25. Stators currents response by PIL of AM

To test the ACO algorithm solution, we took different speed reference values such as 50rad/s and -100rad/s at $t=0.35s$ and also a load torque varying between 5 and 10Nm at $t=0.25s$. Fig. 27 shows the evolution of the speed using the ACO algorithm and also using the analytical method to reach a speed of 50rad/s with a load torque of 5Nm. The speed loop response time is 50 ms, divided by 2, because the initial speed is 50rad/s and the sliding mode parameters have not changed. At $t=0.25s$, the torque has increased to 10Nm, and both approaches have followed the reference identically, for both speed and electromagnetic torque. For $t=0.35s$, the speed reference is -100 rad/s, the motor speed decreases to catch up with the reference, both methods followed the speed and torque

reference in a similar way as also shown in Fig. 28, at start-up the torque is not constant, since the speed variations are not as constant.

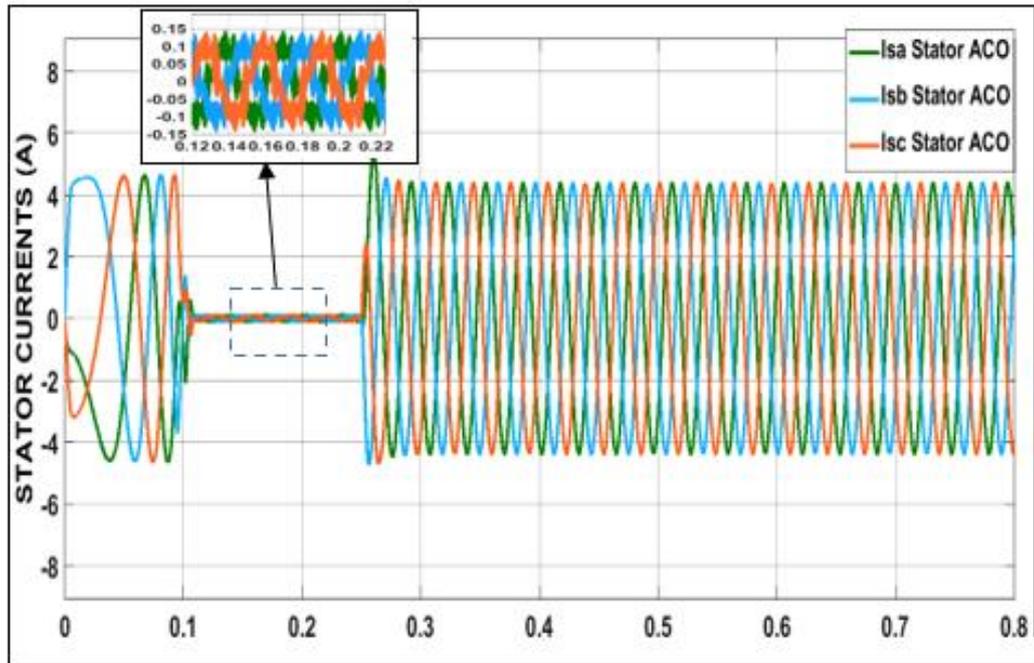


Fig. 26. Stators currents response by PIL of ACO

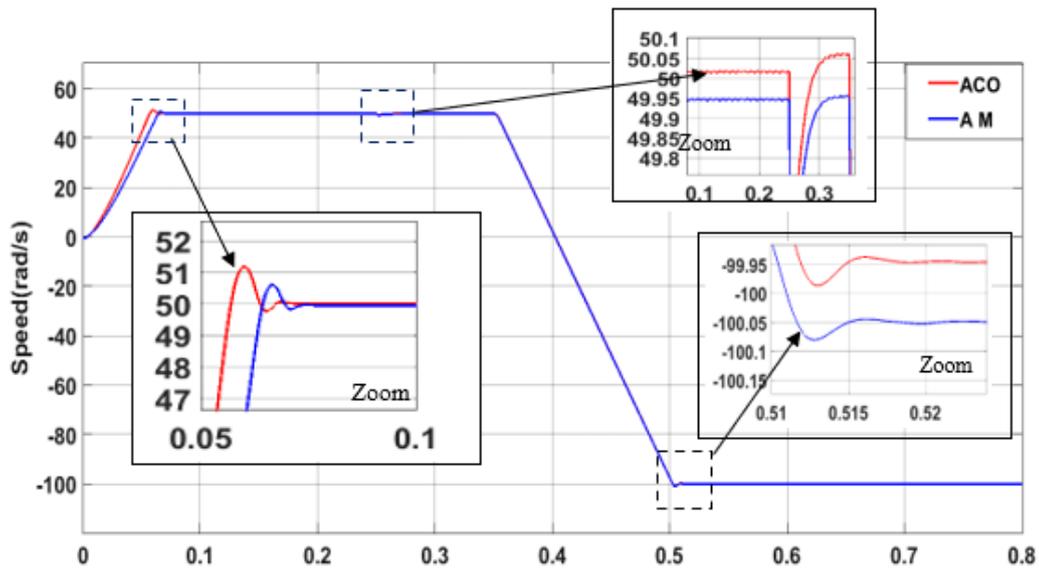


Fig. 27. Speed response by PIL

The relationship between the C_{Mref} couple and the reference currents (i_{dref} and i_{qref}) is not linear; therefore, the simplest solution to optimize control is to have a zero direct reference current, resulting in a couple proportional to the quadratic current. The motor resembles a direct current machine, and this is referred to as artificial decoupling.

In this article, ACO was able to make the couple proportional to the quadratic current, as shown in Fig. 29, and also to the direct current. In other words, for each C_{Mref} , there is a known i_{dref} in a simpler way to modify the magnetic field strength produced by the motor's permanent magnets. The

same applies to i_q , which regulates the output torque based on application requirements. This can be developed through an algorithm to enhance control.

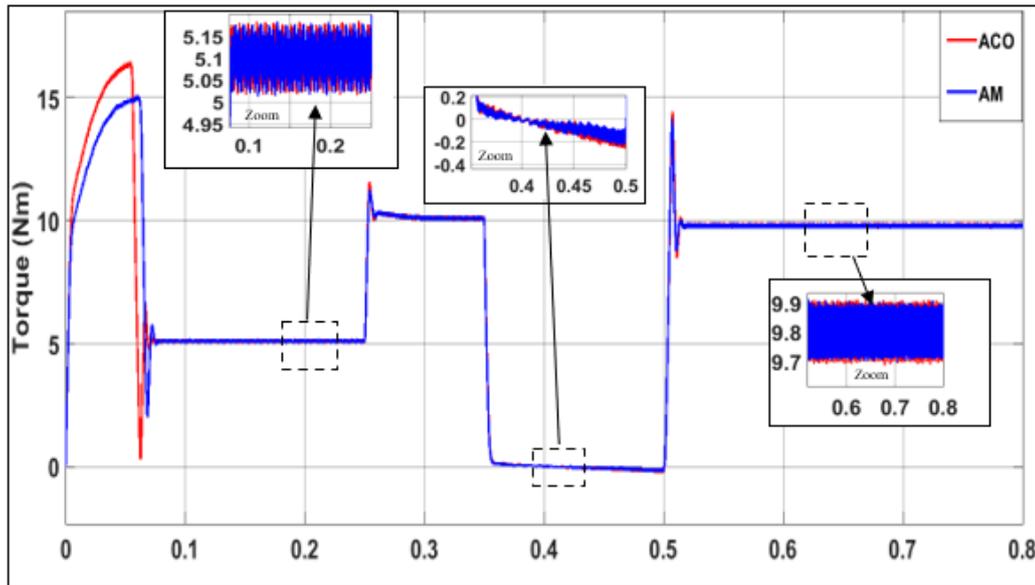


Fig. 28. Torque response by PIL

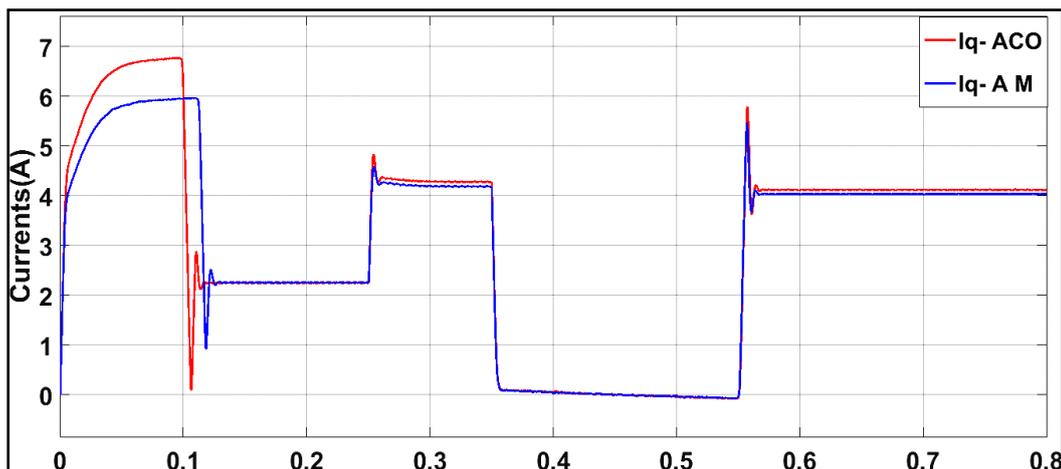


Fig. 29. Iq- current response by PIL

In Fig. 30 and Fig. 31, the stator currents are identical for both approaches, except when the reference speed is changed from 50rad/s to -100rad/s. During this phase, control by the ACO algorithm generated lower absorbed currents than AM. ACO meets the criterion of generating direct and quadratic referential currents (minimum current) more efficiently than the analytical method.

In this section, we will assess the robustness of two approaches, ACO_SMC and AM_SMC, in a scenario where the engine parameters have increased by 50%, such as $L_d = 0.0375H$, $R_s = 3.75 \Omega$, $L_q = 0.1125H$, $f = 0.003 N.m.s/rad$, $J = 0.015 kg.m^2$, to closely resemble practical conditions and demonstrate the effectiveness of AI combined with SMC. Maintaining a consistent reference speed of 100 rad/s, the load torque varies from 0 to 10 Nm, with the addition of random noise ranging between -0.5 and 0.5 Nm.

Fig. 32 and Fig. 33 demonstrate that the AM_SMC approach could not track speed and torque references between instants 0.1 and 0.18s. Additionally, the speed and torque had very high values that could deteriorate the PMSM. In contrast, the combination of the two approaches, SMC and ACO,

showed efficiency and robustness in the face of disturbances and uncertainties. Table 4 reinforces the efficiency with statistics showing the speed of the ACO_SMC approach and a low increase in overshoot and a decrease in response time.

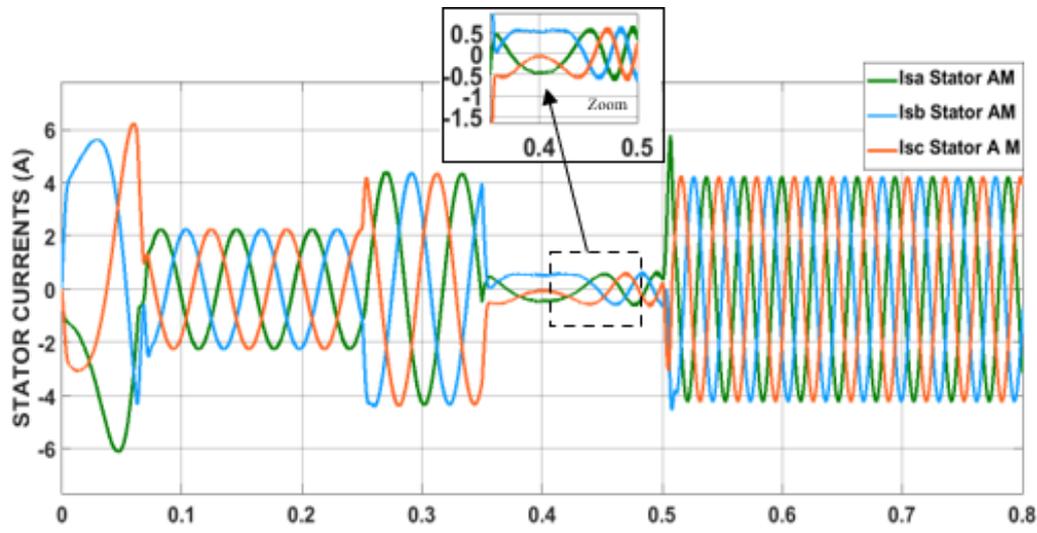


Fig. 30. Stators currents response by PIL of AM

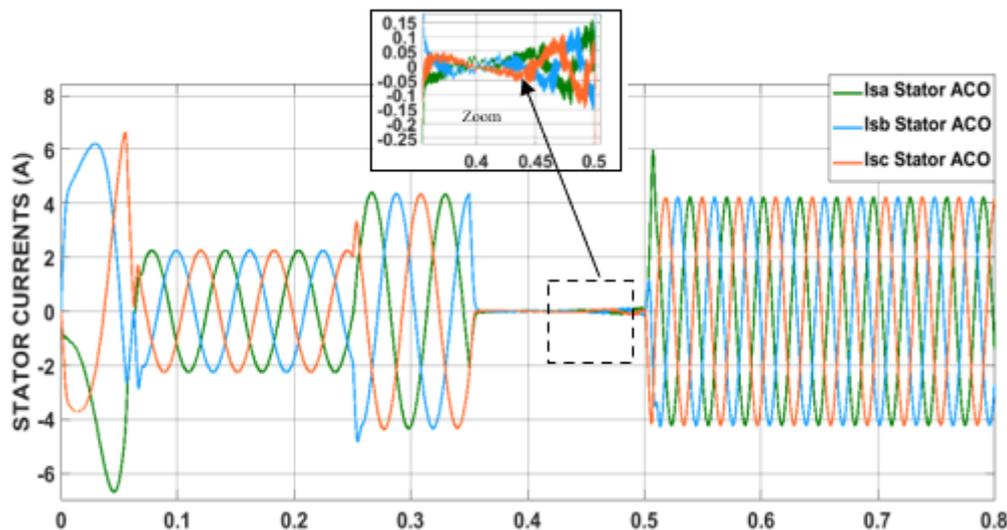


Fig. 31. Stators currents response by PIL of ACO

The PMSM experienced a current peak that exceeded ten times the nominal current value when controlled by AM_SMC, as shown in Fig. 34. This has a negative effect on the motor. On the other hand, the current absorbed using ACO_SMC did not change even if the parameters underwent a 50% increase, and noise was added to the load torque, as illustrated in Fig. 35.

In summary, the AM approach generally involves using mathematical equations to solve a problem or models to find an exact solution. AM can be effective for well-defined problems but not for complex problems, unlike AI, which has demonstrated its simplicity and effectiveness in finding solutions to certain complex problems.

The ACO_SMC approach can be generalized to a wider range of operating conditions; however, ACO must be trained to have more relevant data on the new solution. The most challenging part of the ACO algorithm is the choice of α and β to achieve better performance. We are considering using meta-reinforcement learning, which is an extension of RL. In meta-reinforcement learning, the agent

aims to learn how to perform a task more efficiently and possibly other similar tasks. This means that the agent adjusts its learning algorithm based on past experiences to enhance its ability to adapt to new environments. The mode of experimental work is shown in Fig. 36.

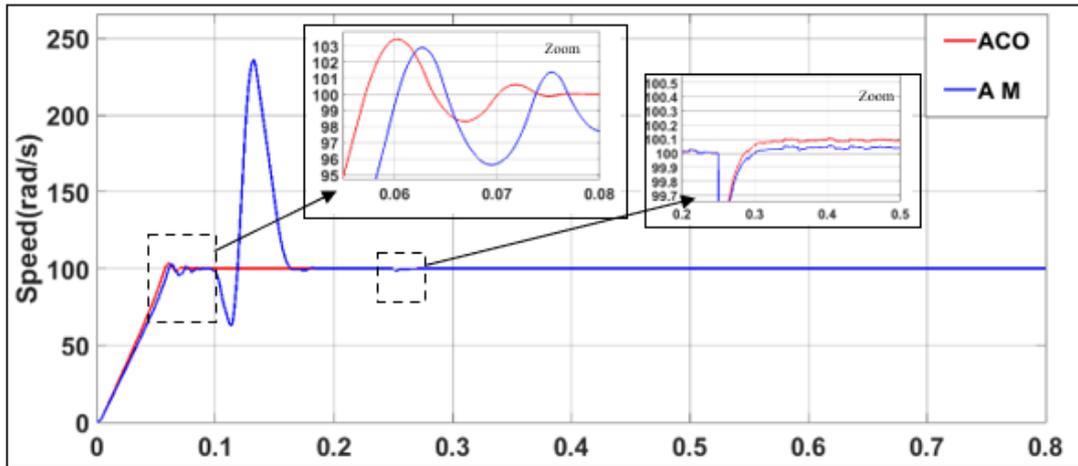


Fig. 32. Speed response by PIL

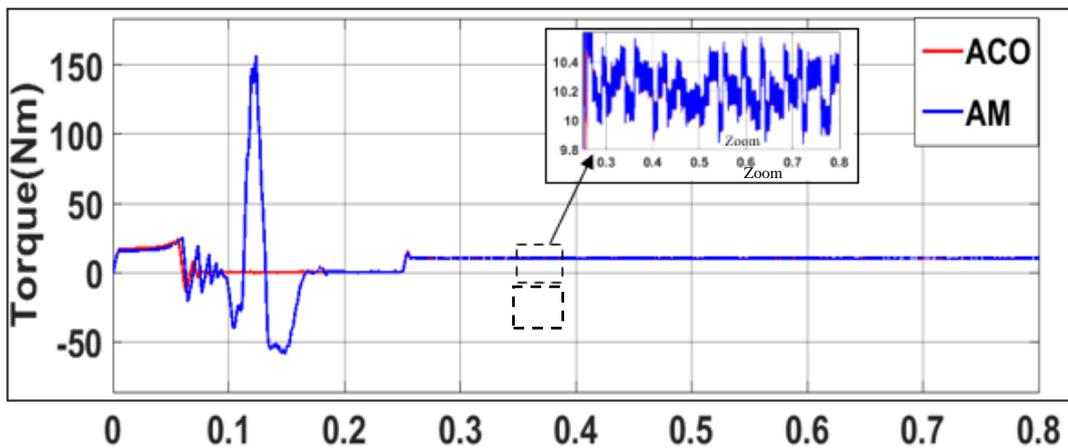


Fig. 33. Torque response by PIL

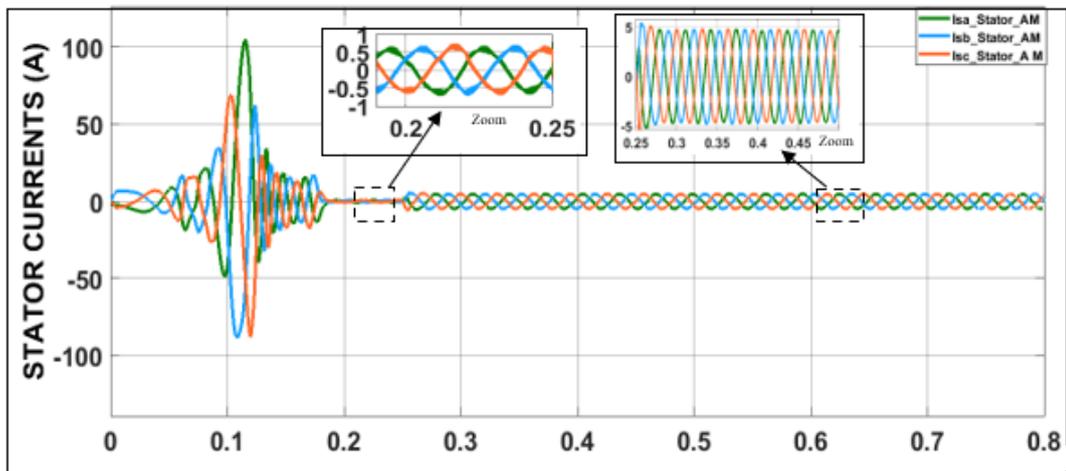
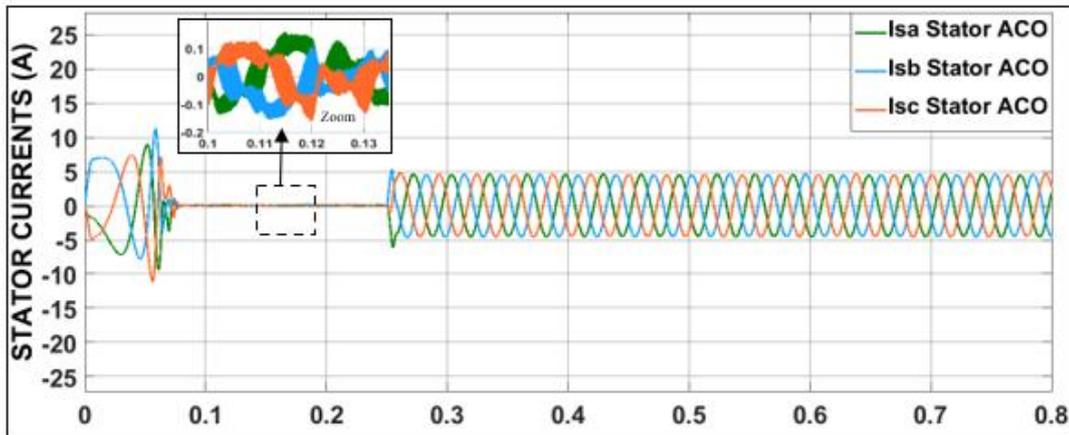
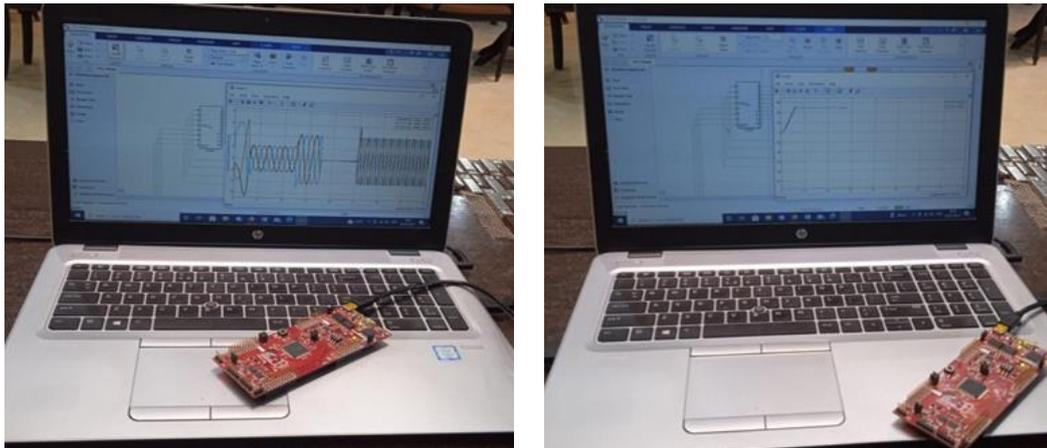


Fig. 34. Stators currents response by PIL of AM

Table 4. Analysis Using ACO

Method	Time and Peak Overshoot			
	Rise Time tr (s)	Peak overshoot MP (%)	Peak Time tp (s)	Settling Time ts (s)
ACO	0.0442	3.4	0.06	0.056

**Fig. 35.** Stators currents response by PIL of ACO**Fig. 36.** Experimental work illustration

4. Conclusion

This study demonstrated the effectiveness of the ACO algorithm in optimizing the control of the PMSM, particularly in managing its nonlinearities, such as the nonlinearity between the reference torque and the reference currents, without relying on mathematical formulas or a defined model. While AM imposed simplified expressions to solve the nonlinearity between torque and reference currents, which is challenging for a complex or undefined system to find the most appropriate equation, ACO's optimization principle is based on experiments verifying combinations to achieve the best solutions.

Both approaches showed low overshoots, with 1.1% for ACO and 0.7% for AM. Regarding response time, AM was 0.1s, while ACO was 0.094s. ACO garnered more interest compared to AM when the PMSM parameters experienced a 50% increase and additional noise was added to the load torque. ACO demonstrated a small overshoot increase (2.3%), a shorter response time (0.056s), and minimal absorbed currents. In contrast, AM revealed current spikes exceeding ten times the nominal

current, negatively impacting the PMSM. Minimizing losses is crucial for efficient drive system performance. ACO successfully addressed this challenge for various scenarios, providing shorter response times, lower overshoot, and minimal absorbed currents. Despite AI's simplicity and effectiveness, it has several limitations, including:

- The dimensionality of the search space to find the optimal solution, especially if the space is very large.
- The algorithm may converge to local optima rather than global optima.
- Sensitivity to hyperparameters for some algorithms, requiring expertise to obtain optimal results.
- Some algorithms require significant computational resources.
- AI may produce solutions that are difficult to interpret.

Moving on to the points that also have influences on the performance optimization of the PMSM:

- Estimating the load torque enables better PMSM control, leading to several performance benefits, such as more precise speed and position control, energy consumption minimization through control algorithms, reduction of vibration and noise by adjusting motor control based on estimated load torque, and increased motor lifespan through precise load torque management to reduce wear.
- Electrical parameters (resistances, inductances, electrical time constants) play a crucial role in optimizing PMSM performance, influencing the motor's dynamic response. Precise estimations are essential for effective control and optimal performance in terms of speed, torque, and precision. The same applies to mechanical parameters (load inertia, friction coefficient...), which prevent undesirable vibrations and oscillations when well-estimated.

This research has broad applications, including electric vehicles and autonomous aircraft, with potential benefits in electric vehicle energy efficiency and overall PMSM performance improvement, offering faster acceleration and responsive control. ACO successfully optimized the parameters of the reference current generator efficiently and simply. However, we aim to enhance ACO to achieve higher performance in highly diverse and complex situations. We are considering employing meta-reinforcement learning to optimize α and β .

Authors' contributions: Adil Najem: Conceptualization, methodology, experimentation. Ahmed Moutabir: Writing, preparation of the original version. Abderrahmane Ouchatt: Supervision. Mohammed El Haissouf: Experimentation.

Funding: No external funding was allocated to this research.

Acknowledgments: We express our gratitude to the members and leaders of the "Laboratory G.E.I.T.I.I.L" for their insightful comments and clear-sighted recommendations.

Conflicts of interest: The authors certify the absence of known conflicting financial interests or personal relationships that could have influenced their work presented in this article.

References

- [1] A. Najem, A. Moutabir, M. Rafik and A. Ouchatti, "Comparative Study of PMSM Control Using Reinforcement Learning and PID Control," *2023 3rd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, pp. 1-5, 2023, <https://doi.org/10.1109/IRASET57153.2023.10153024>.
- [2] R. Thike and P. Pillay, "Mathematical Model of an Interior PMSM With Aligned Magnet and Reluctance Torques," *IEEE Transactions on Transportation Electrification*, vol. 6, no. 2, pp. 647-658, 2020, <https://doi.org/10.1109/TTE.2020.2991369>.

-
- [3] P. Alkorta, O. Barambones, J. A. Cortajarena, I. Martija, and F. J. Maseda, "Effective Position Control for a Three-Phase Motor," *Electronics*, vol. 9, no. 2, p. 241, 2020, <https://doi.org/10.3390/electronics9020241>.
- [4] C. Candelo-Zuluaga, J. -R. Riba and A. Garcia, "PMSM Parameter Estimation for Sensorless FOC Based on Differential Power Factor," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-12, 2021, <https://doi.org/10.1109/TIM.2021.3096861>.
- [5] A. Oubelaid *et al.*, "Intelligent Speed Control and Performance Investigation of a Vector Controlled Electric Vehicle Considering Driving Cycles," *Electronics*, vol. 11, no. 13, p. 1925, 2022, <https://doi.org/10.3390/electronics11131925>.
- [6] V. Biyani, J. R. T. E. T. A, S. S. V. S and P. K. P, "Comparative Study of Different Control Strategies in Permanent Magnet Synchronous Motor Drives," *2021 IEEE 5th International Conference on Condition Assessment Techniques in Electrical Systems (CATCON)*, pp. 275-281, 2021, <https://doi.org/10.1109/CATCON52335.2021.9670516>.
- [7] X. Chi *et al.*, "A ripple suppression of sensorless FOC of PMSM electrical drive system based on MRAS," *Results in Engineering*, vol. 20, p. 101427, 2023, <https://doi.org/10.1016/j.rineng.2023.101427>.
- [8] L. Zhang, X. Zhu, R. Cui and S. Han, "A Generalized Open-Circuit Fault-Tolerant Control Strategy for FOC and DTC of Five-Phase Fault-Tolerant Permanent-Magnet Motor," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 8, pp. 7825-7836, 2022, <https://doi.org/10.1109/TIE.2021.3106012>.
- [9] F. Xiao, Z. Chen, Y. Chen, and H. Liu, "A finite control set model predictive direct speed controller for PMSM application with improved parameter robustness," *International Journal of Electrical Power & Energy Systems*, vol. 143, p. 108509, 2022, <https://doi.org/10.1016/j.ijepes.2022.108509>.
- [10] Y. Zhang, H. Jiang and H. Yang, "Model Predictive Control of PMSM Drives Based on General Discrete Space Vector Modulation," *IEEE Transactions on Energy Conversion*, vol. 36, no. 2, pp. 1300-1307, 2021, <https://doi.org/10.1109/TEC.2020.3036082>.
- [11] M. El haissouf, M. Elharoussi, and A. Ba-razzouk, "Backstepping control strategy for induction motor with rotor flux and load torque observers," *ITM Web Conference*, vol. 48, p. 04002, 2022, <https://doi.org/10.1051/itmconf/20224804002>.
- [12] J. Zhang, S. Wang, P. Zhou, L. Zhao, and S. Li, "Novel prescribed performance-tangent barrier Lyapunov function for neural adaptive control of the chaotic PMSM system by backstepping," *International Journal of Electrical Power & Energy Systems*, vol. 121, p. 105991, 2020, <https://doi.org/10.1016/j.ijepes.2020.105991>.
- [13] J. Wu, J. Zhang, B. Nie, Y. Liu and X. He, "Adaptive Control of PMSM Servo System for Steering-by-Wire System With Disturbances Observation," *IEEE Transactions on Transportation Electrification*, vol. 8, no. 2, pp. 2015-2028, 2022, <https://doi.org/10.1109/TTE.2021.3128429>.
- [14] X. Liu and Q. Zhang, "Robust Current Predictive Control-Based Equivalent Input Disturbance Approach for PMSM Drive," *Electronics*, vol. 8, no. 9, p. 1034, 2019, <https://doi.org/10.3390/electronics8091034>.
- [15] S. Shirmohammadi and A. Ferrero, "Camera as the instrument: the rising trend of vision based measurement," *IEEE Instrumentation & Measurement Magazine*, vol. 17, no. 3, pp. 41-47, 2014, <https://doi.org/10.1109/MIM.2014.6825388>.
- [16] L. Xiao, L. Zhang, F. Gao, and J. Qian, "Robust fault-tolerant synergetic control for dual three-phase PMSM drives considering speed sensor fault," *IEEE Access*, vol. 8, pp. 78912-78922, 2020, <https://doi.org/10.1109/MIM.2014.6825388>.
- [17] M. Nicola and C. I. Nicola, "Sensorless Fractional Order Control of PMSM Based on Synergetic and Sliding Mode Controllers," *Electronics*, vol. 9, no. 9, p. 1494, 2020, <https://doi.org/10.3390/electronics9091494>.
- [18] G. Yao, Y. Cheng, Z. Wang, and Y. Xiao, "Study on a Second-Order Adaptive Sliding-Mode Observer Control Algorithm for the Sensorless Permanent Magnet Synchronous Motor," *Processes*, vol. 11, no. 6, p. 1636, 2023, <https://doi.org/10.3390/pr11061636>.
- [19] F. M. Zaihidee, S. Mekhilef, and M. Mubin, "Robust Speed Control of PMSM Using Sliding Mode Control (SMC)—A Review," *Energies*, vol. 12, no. 9, p. 1669, 2019,
-

<https://doi.org/10.3390/en12091669>.

- [20] D. Nicolis, F. Allevi and P. Rocco, "Operational Space Model Predictive Sliding Mode Control for Redundant Manipulators," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1348-1355, 2020, <https://doi.org/10.1109/TRO.2020.2974092>.
- [21] L. Wu, J. Liu, S. Vazquez and S. K. Mazumder, "Sliding Mode Control in Power Converters and Drives: A Review," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 3, pp. 392-406, 2022, <https://doi.org/10.1109/JAS.2021.1004380>.
- [22] S. Wang, J. Na and Q. Chen, "Adaptive Predefined Performance Sliding Mode Control of Motor Driving Systems With Disturbances," *IEEE Transactions on Energy Conversion*, vol. 36, no. 3, pp. 1931-1939, 2021, <https://doi.org/10.1109/TEC.2020.3038010>.
- [23] M. Nicola and C. -I. Nicola, "Improvement of PMSM Control Using Reinforcement Learning Deep Deterministic Policy Gradient Agent," *2021 21st International Symposium on Power Electronics (Ee)*, pp. 1-6, 2021, <https://doi.org/10.1109/Ee53374.2021.9628371>.
- [24] R. Morimoto, S. Nishikawa, R. Niiyama and Y. Kuniyoshi, "Model-Free Reinforcement Learning with Ensemble for a Soft Continuum Robot Arm," *2021 IEEE 4th International Conference on Soft Robotics (RoboSoft)*, pp. 141-148, 2021, <https://doi.org/10.1109/RoboSoft51838.2021.9479340>.
- [25] M. Tian, K. Wang, H. Lv, and W. Shi, "Reinforcement learning control method of torque stability of three-phase permanent magnet synchronous motor," *Journal of Physics: Conference Series*, vol. 2183, no. 1, p. 012024, 2022, <https://doi.org/10.1088/1742-6596/2183/1/012024>.
- [26] F. Yin, X. Yuan, Z. Ma, and X. Xu, "Vector Control of PMSM Using TD3 Reinforcement Learning Algorithm," *Algorithms*, vol. 16, no. 9, p. 404, 2023, <https://doi.org/10.3390/a16090404>.
- [27] G. Jegadeeswari, G. Ezhilarasi, V. Pramila, J. Vijayanand and B. Kirubadurai, "Performance Comparison of Dynamic Responses & Speed Control of Switched Reluctance Motor Using PI & Fuzzy logic Controller," *2023 9th International Conference on Electrical Energy Systems (ICEES)*, pp. 674-680, 2023, <https://doi.org/10.1109/ICEES57979.2023.10110237>.
- [28] M. Nicola and C. -I. Nicola, "Improved Performance of Sensorless Control for PMSM Based on Neuro-fuzzy Speed Controller," *2021 9th International Conference on Modern Power Systems (MPS)*, pp. 1-6, 2021, <https://doi.org/10.1109/MPS52805.2021.9492586>.
- [29] Y. Hu, Y. Zhang and D. Gong, "Multiobjective Particle Swarm Optimization for Feature Selection With Fuzzy Cost," *IEEE Transactions on Cybernetics*, vol. 51, no. 2, pp. 874-888, 2021, <https://doi.org/10.1109/TCYB.2020.3015756>.
- [30] J. M. Ahn, J. C. Son, and D. K. Lim, "Optimal Design of Outer-Rotor Surface Mounted Permanent Magnet Synchronous Motor for Cogging Torque Reduction Using Territory Particle Swarm Optimization," *Journal of Electrical Engineering & Technology*, vol. 16, no. 1, pp. 429-436, 2021, <https://doi.org/10.1007/s42835-020-00599-z>.
- [31] J. Song, Y. -K. Wang, Y. Niu, H. -K. Lam, S. He and H. Liu, "Periodic Event-Triggered Terminal Sliding Mode Speed Control for Networked PMSM System: A GA-Optimized Extended State Observer Approach," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 5, pp. 4153-4164, 2022, <https://doi.org/10.1109/TMECH.2022.3148541>.
- [32] M. Nicola and C. -I. Nicola, "Tuning of PI Speed Controller for PMSM Control System Using Computational Intelligence," *2021 21st International Symposium on Power Electronics (Ee)*, pp. 1-6, 2021, <https://doi.org/10.1109/Ee53374.2021.9628297>.
- [33] Y. Wang, L. Wang, G. Chen, Z. Cai, Y. Zhou, and L. Xing, "An Improved Ant Colony Optimization algorithm to the Periodic Vehicle Routing Problem with Time Window and Service Choice," *Swarm and Evolutionary Computation*, vol. 55, p. 100675, 2020, <https://doi.org/10.1016/j.swevo.2020.100675>.
- [34] L. Zhang, J. Bai, and J. Wu, "Speed Sensor-Less Control System of Surface-Mounted Permanent Magnet Synchronous Motor Based on Adaptive Feedback Gain Supertwisting Sliding Mode Observer," *Journal of Sensors*, vol. 2021, 2021, <https://doi.org/10.1155/2021/8301359>.
- [35] B. P. Ganthia, R. Pradhan, R. Sahu, and A. K. Pati, "Artificial Ant Colony Optimized Direct Torque Control of Mathematically Modeled Induction Motor Drive Using PI and Sliding Mode Controller,"

- Recent Advances in Power Electronics and Drives: Select Proceedings of EPREC 2020*, vol. 707, pp. 389–408, 2021, https://doi.org/10.1007/978-981-15-8586-9_35.
- [36] A. L. Boloix, “Design and Implementation of Advanced Control Strategies for a Permanent Magnet Synchronous Motor: A Comparative Study between FOC and SMC,” *UPCommons*, 2023, <http://hdl.handle.net/2117/400158>.
- [37] K. Jammousi, M. Bouzguenda, Y. Dhieb, M. Ghariani and M. Yaich, “Gain optimization of sliding mode speed control for DC motor,” *2020 6th IEEE International Energy Conference (ENERGYCon)*, pp. 159-163, 2020, <https://doi.org/10.1109/ENERGYCon48941.2020.9236508>.
- [38] L. J. Mpanza and J. O. Pedro, “Nature-Inspired Optimization Algorithms for Sliding Mode Control Parameters Tuning for Autonomous Quadrotor,” *2019 IEEE Conference on Control Technology and Applications (CCTA)*, pp. 1087-1092, 2019, <https://doi.org/10.1109/CCTA.2019.8920474>.
- [39] A. Mughees and I. Ahmad, “Multi-Optimization of Novel Conditioned Adaptive Barrier Function Integral Terminal SMC for Trajectory Tracking of a Quadcopter System,” *IEEE Access*, vol. 11, pp. 88359-88377, 2023, <https://doi.org/10.1109/ACCESS.2023.3304760>.
- [40] M. E. Haissouf, M. Elharoussi, A. Ba-Razzouk, “DSP in the Loop Implementation of an Enhanced Sliding Mode Control for Permanent Magnet Synchronous Motor,” *International Review of Electrical Engineering*, vol. 18, no. 4, pp. 291-298, 2023, <https://doi.org/10.15866/iree.v18i4.23399>.
- [41] S. Ebadinezhad, “DEACO: Adopting dynamic evaporation strategy to enhance ACO algorithm for the traveling salesman problem,” *Engineering Applications of Artificial Intelligence*, vol. 92, p. 103649, 2020, <https://doi.org/10.1016/j.engappai.2020.103649>.
- [42] S. Mahfoud, A. Derouich, A. Iqbal, and N. El Ouanjli, “ANT-colony optimization-direct torque control for a doubly fed induction motor : An experimental validation,” *Energy Reports*, vol. 8, pp. 81-98, 2022, <https://doi.org/10.1016/j.egy.2021.11.239>.
- [43] S. Mahfoud, A. Derouich, N. El Ouanjli, N. V. Quynh, and M. A. Mossa, “A New Hybrid Ant Colony Optimization Based PID of the Direct Torque Control for a Doubly Fed Induction Motor,” *World Electronic Vehicle Journal*, vol. 13, no. 5, p. 78, 2022, <https://doi.org/10.3390/wevj13050078>.
- [44] F. Al-Amyal, L. Számel, and M. Hamouda, “An enhanced direct instantaneous torque control of switched reluctance motor drives using ant colony optimization,” *Ain Shams Engineering Journal*, vol. 14, no. 5, p. 101967, 2023, <https://doi.org/10.1016/j.asej.2022.101967>.
- [45] Y. K. Poudel and P. Bhandari, “Control of the BLDC Motor Using Ant Colony Optimization Algorithm for Tuning PID Parameters,” *Archives of Advanced Engineering Science*, pp. 1–12, 2023, <https://doi.org/10.47852/bonviewAAES32021184>.
- [46] N. Gupta, M. Kaur, R. Gupta, “Ant colony optimization based optimal tuning of Fractional Order (FO) PID controller for controlling the speed of a DC motor,” *Journal of Engineering Research*, vol. 11, no. 3, pp. 100-111, 2022, <https://doi.org/10.36909/jer.15251>.
- [47] A. Nayyar and R. Singh, “IEEMARP- a novel energy efficient multipath routing protocol based on ant Colony optimization (ACO) for dynamic sensor networks,” *Multimedia Tools and Application*, vol. 79, pp. 35221–35252, 2020, <https://doi.org/10.1007/s11042-019-7627-z>.
- [48] C. Miao, G. Chen, C. Yan, and Y. Wu, “Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm,” *Computers & Industrial Engineering*, vol. 156, p. 107230, 2021, <https://doi.org/10.1016/j.cie.2021.107230>.
- [49] H. Albalawi, S. Zaid, M. El-Shimy, A. M. Kassem, “Ant Colony Optimized Controller for Fast Direct Torque Control of Induction Motor,” *Sustainability*, vol. 15, no. 4, p. 3740, 2023, <https://doi.org/10.3390/su15043740>.
- [50] C. Zhu, Q. Tu, C. Jiang, M. Pan, H. Huang, and Z. Tu, “Global fast terminal sliding mode control strategy for permanent magnet synchronous motor based on load torque Luenberger observer,” *IEICE Electronics Express*, vol. 18, no. 19, pp. 1-6, 2021, <https://doi.org/10.1587/elex.18.20210348>.