Science and Technology for Energy Transition **80**, 33 (2025)

https://doi.org/10.2516/stet/2025010

Innovative Strategies and Technologies for Sustainable Renewable Energy and Low-Carbon Development
Mohan Lal Kolhe and Yang Miao (Guest Editors)

Available online at:
stet-review.org

REGULAR ARTICLE OPEN ACCESS

Online intelligent parameter and speed estimation of permanent magnet synchronous motors using bacterial foraging optimization

Mohammed M. Alrashed[1], Mohamed F. Elnaggar[1], Aymen Flah[2,3,4,5,*], and Claude Ziad El-Bayeh[6]

[1] Department of Electrical Engineering, College of Engineering, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia
[2] University of Gabes, National Engineering School of Gabes, 6072, Tunisia
[3] Centre for Research Impact & Outcome, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, 140401, Punjab, India
[4] Applied Science Research Center, Applied Science Private University, Amman, 11931, Jordan
[5] Jadara University Research Center, Jadara University, Irbid, Jordan
[6] College of Engineering and Technology, University of Doha for Science and Technology, Doha, Qatar

Received: 31 January 2025 / Accepted: 27 February 2025

Abstract. Accurate estimation of the parameters and speed of Permanent Magnet Synchronous Motors (PMSMs) is crucial for achieving optimal performance in control applications. Traditional methods, such as the Model Reference Adaptive System (MRAS) rely on manually tuned Proportional-Integral (PI) controllers, leading to suboptimal results due to fixed tuning parameters that do not adapt to varying operating conditions. This limitation affects the precision of parameter identification, leading to potential inefficiencies in motor control. This paper proposes an intelligent online estimation method that leverages Popov hyperstability theory and the Bacterial Foraging Optimization (BFO) algorithm to address this issue. The proposed approach simultaneously estimates three key PMSM parameters – stator resistance, inductance, and permanent magnet flux – along with the actual motor speed. Unlike conventional methods, an online BFO-based tuning algorithm is integrated into the MRAS framework, allowing adaptive and optimal adjustment of controller parameters in real time. Extensive practical evaluations demonstrate that the proposed method significantly improves estimation accuracy and adaptability compared to traditional approaches. The results confirm its effectiveness in enhancing PMSM control performance, making it a promising solution for high-precision motor applications. Experimental results demonstrate a 12% improvement in estimation precision compared to traditional manual tuning methods.

Keywords: PMSM, MRAS, Parameters estimation, Bacterial foraging optimization, Speed variation.

Abbreviation

Parameter

i_d	Direct axis stator current
i_q	Quadrature axis stator current
v_d	Direct axis stator voltage
v_q	Quadrature axis stator voltage
R_s	Armature resistance
L_d	Direct axis stator self-inductance
L_q	Quadrature axis stator self-inductance
λ_m	Permanent magnet flux linkage

Variable/Term

$\theta_i\ (j,\ k,\ l)$	Position of the ith bacterial in the jth and kth chemotactic and reproduction step and the lth elimination step
Δ_s	Swim direction
C	Swim size
$J\ (i,\ j,\ k,\ l)$	Represents the minimum value of fitness calculated in the chemotactic step
$C\ (i) = C\ (i) + d_1$	Bacteria positions are updated using a specific equation during the swim condition
$C\ (i) = C\ (i) - d_2$	Bacteria positions are updated using a specific equation during the tumbling condition
$d_1,\ d_2$	Random positive values less than one, are used in the position update equations

* Corresponding author: aymen.flah@enig.u-gabes.tn

1 Introduction

1.1 Research background

Permanent Magnet Synchronous Motors (PMSMs) have made significant attention in applications requiring large-speed operation. This is largely due to their exceptional advantages, including high efficiency, low inertia, a high torque-to-current ratio, a high power factor, and minimal maintenance requirements. Additionally, their compact size and lighter weight make PMSMs preferred over induction motors for high-performance applications [1].

Industries such as transportation and electric vehicles increasingly rely on PMSMs, particularly for their ability to operate effectively in high-speed regions, facilitated by field-weakening techniques [2]. Researchers have also focused on reducing the overall cost of PMSM speed control systems in these applications, especially by eliminating the need for mechanical sensors. Numerous sensorless techniques for speed estimation have been proposed) alongside modifications to control strategies like Direct Torque Control (DTC) to enhance performance and optimize control loops [3].

One of the key advantages of PMSMs is their capability to operate efficiently at high speeds. This feature has been the subject of extensive research and innovation [4]. These efforts underscore the PMSM's utility and growing prominence in advanced applications requiring superior performance and reliability.

1.2 Literature review

AC motor control method, which is based on Field-Oriented Control (FOC) topology, enables precise regulation of motor speed or torque by decomposing the motor flux into two components: direct and quadrature) [5]. This approach relies on two key current controllers to generate the required stator voltages for these components. A significant challenge lies in regulating the quadrature current component, which requires a reference current for the quadrature stator to initiate operation.

In this method, the quadrature component is directly tied to the electromagnetic torque reference, which is determined by an alternative controller exploiting speed data. The regulator calculates the needed torque component once the actual and reference speeds are obtained. However, the primary difficulty arises from accurately determining the actual speed and identifying the rotor position, which is critical for effective operation [6].

The accurate identification and stability of PMSM parameters heavily influences the effectiveness of vector control strategies. However, a significant challenge arises due to the variation of these parameters over time, caused by factors such as temperature fluctuations, mechanical or electrical faults, vibrations, or environmental conditions like dust in electric vehicles. These parameter variations directly impact the drive performance, the precision of identification techniques) control loops [7] and algorithms like field weakening [8].

Several PMSM parameter identification methods have been proposed to ensure high accuracy in parameter tracking to address this issue. For example, Bolognani introduced a recursive parameter identification method for estimating PMSM parameters [9]. Additionally, adaptive identification models using predictive current control techniques have been explored) and adaptive algorithms have been applied in various contexts [10]. Techniques leveraging Lyapunov and Popov stability criteria have also been developed, with Popov's criterion offering robust convergence [11]. Zhu *et al.* [12] employed the Kalman filter to estimate parameters like winding resistance and magnet flux in brushless AC motors. The Model Reference Adaptive System (MRAS) has emerged as a promising approach for simultaneous parameter observation in PMSMs. By leveraging the Popov stability criterion, MRAS adjusts the error between the estimator and the actual model to estimate unknown parameters. The system uses two models: the reference model and the adjustable model, with the latter relying on a Proportional-Integral (PI) adapting mechanism. However, a consistent challenge in the literature has been the absence of a robust method for tuning these PI mechanism parameters [13]. Approaches such as recurrent neural networks or fuzzy logic have been employed to replace PI regulators and mitigate tuning issues. These methods, however, demand either extensive training data (in the case of neural networks) or clearly defined boundaries (for fuzzy logic systems) [14].

The challenges associated with vector control strategies and PMSM parameter estimation are further exacerbated when considering traditional motor control methods that rely on rotary encoders or resolvers. These devices are widely used in conventional systems to provide precise rotor position information, enabling optimal control performance. While they ensure high accuracy in ideal conditions, real-world applications reveal several drawbacks. Issues such as high costs, oversized designs, and compatibility limitations with machine sizes arise due to the complex wiring schemes, sensitive encoder placement, and susceptibility to external factors like dust, vibration, and temperature fluctuations. These limitations ultimately affect the robustness and reliability of PMSM drive systems. Building upon the previously discussed parameter estimation challenges and advancements in intelligent techniques, such as the Bacterial Foraging Optimization (BFO) algorithm, an alternative solution emerges in the form of software-based speed encoders. Recent research, combined with advancements in processor technologies and computational performance, suggests that software speed encoders can effectively replace hardware encoders. This approach addresses the cost and durability issues associated with physical encoders. It integrates seamlessly with advanced parameter estimation methods like MRAS, where BFO optimization enhances the tuning of sensitive parameters. By leveraging these software-based solutions, the PMSM control system can achieve high performance while overcoming traditional limitations. Integrating intelligent algorithms such as BFO further ensures that the system remains adaptive and robust against parameter variations and environmental challenges, ultimately improving modern motor control strategies' overall efficiency and reliability [15].

Various sensorless methods for rotor position and speed estimation have been explored in the literature, including approaches using artificial intelligence) observers based on sliding mode) and MRAS. The effectiveness of these software-based speed encoders has been demonstrated in the referenced works under specific operating conditions and within limited speed ranges. However, their strength has primarily been validated within certain speed zones, often excluding the high-speed range.

1.3 Research motivation and problem statement

The control of PMSMs relies heavily on accurate speed and rotor position estimation, which is crucial for achieving high performance in industrial and automotive applications. Traditional control strategies, such as FOC require precise knowledge of motor parameters to regulate torque and speed effectively. However, a significant challenge arises in accurately determining these parameters, as they tend to vary over time due to temperature fluctuations, mechanical and electrical faults, vibrations, and environmental conditions. These variations can significantly affect the stability and efficiency of PMSM drives, leading to degraded performance in real-world operating conditions [16].

One widely adopted solution for speed and position estimation is using rotary encoders or resolvers. These hardware-based sensors provide precise real-time data, ensuring optimal control performance. However, despite their accuracy, they introduce several practical limitations. The high cost of these devices, coupled with their complex wiring requirements and sensitivity to external disturbances such as dust, vibrations, and extreme temperatures, makes them less suitable for many industrial applications. Furthermore, their integration into compact or cost-sensitive motor systems is often impractical, highlighting the need for alternative estimation methods.

To address these challenges, sensorless estimation techniques have been explored in the literature, including artificial intelligence-based approaches, sliding mode observers, and MRAS estimators. Among these, MRAS has emerged as a promising solution for PMSM parameters and speed estimation due to its reliance on the Popov stability criterion, which ensures convergence and robustness. However, a critical issue in MRAS implementation is the tuning of the PI controller, which is essential for adapting the adjustable model to match the reference model. Traditional manual tuning methods are inefficient, as they fail to account for the nonlinear behavior of PMSMs, leading to suboptimal performance in dynamic operating conditions) [17].

Another major challenge arises in high-speed motor operation, where PMSMs can reach speeds up to 200% of their rated value, particularly in field-weakening mode. In such conditions, crucial parameters like stator resistance and magnet flux undergo significant variations, directly impacting the mathematical models used for sensorless speed estimation. This variation can cause instability, reduced estimation accuracy, and ultimately compromise the reliability of the control system. Given that high-speed operation is essential in applications such as electric vehicles, robotics, and industrial automation, addressing these parameter variations is vital to ensuring consistent performance and stability across a wide speed range.

The limitations of hardware-based encoders and the challenges of parameter variations and PI tuning in MRAS-based estimation emphasize the need for an advanced, adaptive solution. Recent developments in intelligent optimization algorithms, such as the BFO algorithm, provide a potential means to overcome these issues. By dynamically tuning the MRAS PI controller, optimization techniques can enhance the adaptability of PMSM control systems, improving their performance in both normal and high-speed operating conditions. This approach motivates the development of a robust, software-based speed encoder that integrates MRAS with BFO optimization to provide a cost-effective, accurate, and reliable alternative to traditional sensor-based methods.

1.4 Contributions

The contributions of this work can be summarized in this ensemble of points as follows:

- Development of a software-based speed encoder as an alternative to traditional rotary encoders, reducing cost, complexity, and susceptibility to external disturbances.
- Enhancement of MRAS-based parameter and speed estimation by integrating the BFO algorithm, enabling dynamic and adaptive tuning of the PI controller.
- Improvement in estimation accuracy and robustness by addressing PMSM parameter variations due to temperature fluctuations, mechanical faults, and environmental conditions, ensuring stable operation across a wide range of speeds.
- Ensuring high-speed operational stability by refining sensorless speed estimation techniques to account for variations in key parameters, such as magnet flux and stator resistance, particularly in field-weakening mode.
- Reduction in computational complexity compared to traditional neural networks and fuzzy logic-based tuning approaches, making the proposed method more practical for real-time motor control applications.
- Validation through extensive practical evaluations, demonstrating a 12% improvement in estimation precision compared to traditional manual tuning methods while maintaining stability in dynamic operating conditions.

1.5 Paper organization

The paper is structured into six sections. Following the introduction, Section 2 presents the mathematical model of the main electrical machine. Section 3 discusses the FOC strategy, outlining its typology and implementation. Section 4 provides a mathematical analysis of MRAS estimators, followed by a detailed explanation of the selected optimization algorithm, BFO. Section 5 presents the simulation and experimental results, demonstrating the implementation of this complex architecture on control hardware. Finally, the conclusion summarizes the paper's

key findings' and provides insights for future research directions.

2 PMSM model

The dynamic simulation model of this machine, assuming negligible effects from saturation and hysteresis iron losses, can be set as a nonlinear differential equation. These mathematical models establish relationships between the stator currents and voltages, as is in equation (1). This model links the rotor field to the mechanical parameters, including torque, speed, and angular position, equation (2) [18–20]. The mechanical motor equation is (3).

.

$$\begin{cases} v_{\mathrm{d}} = R_{\mathrm{s}} i_{\mathrm{d}} + L_{\mathrm{d}} \dfrac{\mathrm{d}i_{\mathrm{d}}}{\mathrm{d}t} - \omega L_{\mathrm{q}} i_{\mathrm{q}} \\ v_{\mathrm{q}} = R_{\mathrm{s}} i_{\mathrm{q}} + L_{\mathrm{q}} \dfrac{\mathrm{d}i_{q}}{\mathrm{d}t} + \omega L_{\mathrm{d}} i_{\mathrm{d}} + \omega \lambda_{\mathrm{m}} \end{cases} \tag{1}$$

$$\begin{cases} T_{\mathrm{e}} = \left(\dfrac{3}{2}\right)\left(\dfrac{P}{2}\right)(\lambda_{\mathrm{d}} i_{\mathrm{q}} - \lambda_{\mathrm{q}} i_{\mathrm{d}}) \\ \lambda_{\mathrm{d}} = L_{\mathrm{d}} i_{\mathrm{d}} + \lambda_{\mathrm{m}} \; and \; \lambda_{\mathrm{q}} = L_{\mathrm{q}} i_{\mathrm{q}} \end{cases} \tag{2}$$

$$(T_{\mathrm{e}} - T_{\mathrm{l}}) = \left(\frac{P}{2}\right)\left(J\frac{\mathrm{d}\omega}{\mathrm{d}t} + f\omega\right). \tag{3}$$

With θ and ω are the rotor angular position and speed, respectively, f is the friction coefficient, J is the moment of rotor inertia, T_{e} is the developed electromagnetic torque, and T_{l} is the load torque.

The voltage source inverter (VSI) is described with the IGBT state (S_i, $i = a, b, c$) and the DC bus voltage U_{DC}. The generated voltage vector is given in equation (4).

$$\vec{V}_s = \sqrt{\frac{2}{3}} U_{\mathrm{DC}}\left(S_a + S_b \mathrm{e}^{i2\pi/3} + S_c \mathrm{e}^{i4\pi/3}\right). \tag{4}$$

3 Selected control topology

Since the 1970s, various vector control strategies have been developed, including rotor-flux-oriented, stator-flux-oriented, and magnetizing-flux-oriented approaches. These strategies have proven highly effective for controlling synchronous motors in adjustable-speed drive applications. They excel in handling rapidly changing loads across a wide speed range, including high-speed operation enabled by the field-weakening strategy.

Some sources try to select DTC, FOC, or Scalar control due to several advantages outlined in Table 1. While DTC offers better performance in certain aspects, such as fast dynamic response and high control loop precision, it also introduces significant ripple in current and voltage signals. These ripples negatively impact the accuracy of estimators that rely on stator current and voltage measurements. Therefore, considering the need for a sensorless control approach and the challenge of torque ripple in DTC, FOC was the most suitable control topology for this induction machine.

The fundamental concept is to control the electromagnetic torque in a manner analogous to that of a separately excited DC machine. The three-phase stator currents are measured and transformed into the d-q frame using Clarke and Park transformations to achieve this. A speed sensor or observer evaluates the speed error by comparing the actual motor speed with the target speed. The speed controller generates a torque command, determining the quadrature stator current (i_{q}) as the control input. Under nominal conditions, the direct stator current (i_{d}) is set to zero. However, i_{d} is adjusted to a negative value in high-speed regions as part of the field-weakening strategy [22, 23]. This adjustment modifies the electromagnetic torque equation, as expressed in equation (5).

$$T_{\mathrm{e}} = \left(\frac{3}{2}\right)\left(\frac{P}{2}\right)(\lambda_{\mathrm{m}} i_{\mathrm{q}}). \tag{5}$$

The outputs of the stator current controllers are the direct and quadrature stator voltages. These voltages are then processed using the Clarke transformation to generate the reference voltages required for the Pulse With Modulation (PWM) block, which drives the stator terminals. The overall structure of the field-oriented vector control system is illustrated in Figure 1.

4 MRAS adaptation for PMSM parameters and speed estimation

4.1 MRAS for PMSM parameters

The MRAS estimator is designed to determine the parameters of a PMSM using the principles of Popov stability theory. This estimator relies solely on real-time measurements of current, voltage, and rotor speed to accurately estimate stator resistance, stator inductance, and rotor flux linkage simultaneously. The estimator offers enhanced response speed and reduced computational time due to the inclusion of both proportional and integral components. By selecting the *d*-axis and *q*-axis stator current components as state variables, the PMSM state system can be expressed as follows in equation (6). The adjustable parameter state system is given by equation (7). We note $\frac{1}{\tau} = \frac{L_{\mathrm{s}}}{R_{\mathrm{s}}}$.

$$\dot{X} = \begin{bmatrix} -\tau & \omega \\ -\omega & -\tau \end{bmatrix} \begin{bmatrix} i_{\mathrm{d}} & i_{\mathrm{q}} \end{bmatrix}^T + \begin{bmatrix} \frac{1}{L_{\mathrm{s}}} & 0 \\ 0 & \frac{1}{L_{\mathrm{s}}} \end{bmatrix} \begin{bmatrix} u_{\mathrm{d}} & u_{\mathrm{q}} \end{bmatrix}^T + \begin{bmatrix} 0 \\ -\omega I_f \end{bmatrix} \tag{6}$$

$$\dot{\hat{X}} = \hat{A} \begin{bmatrix} \hat{c} & 0 \\ 0 & \hat{c} \end{bmatrix} + \begin{bmatrix} \hat{c} & 0 \\ 0 & \hat{c} \end{bmatrix} U + \begin{bmatrix} 0 \\ \omega \hat{I}_f \end{bmatrix} + \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \left(\hat{X} - X\right). \tag{7}$$

Table 1. FOC face DTC and scalar control topologies [21].

Feature	FOC	Scalar control (V/f)	DTC
Torque control	Precise	Indirect	Good but with ripple
Speed control	Accurate	Poor at low speed	Moderate
Dynamic response	Fast	Slow	Very fast
Efficiency	High	Moderate	High
Torque ripple	Low	N/A	High
Implementation complexity	High	Low	Moderate
Sensorless control feasibility	Yes	No	Yes

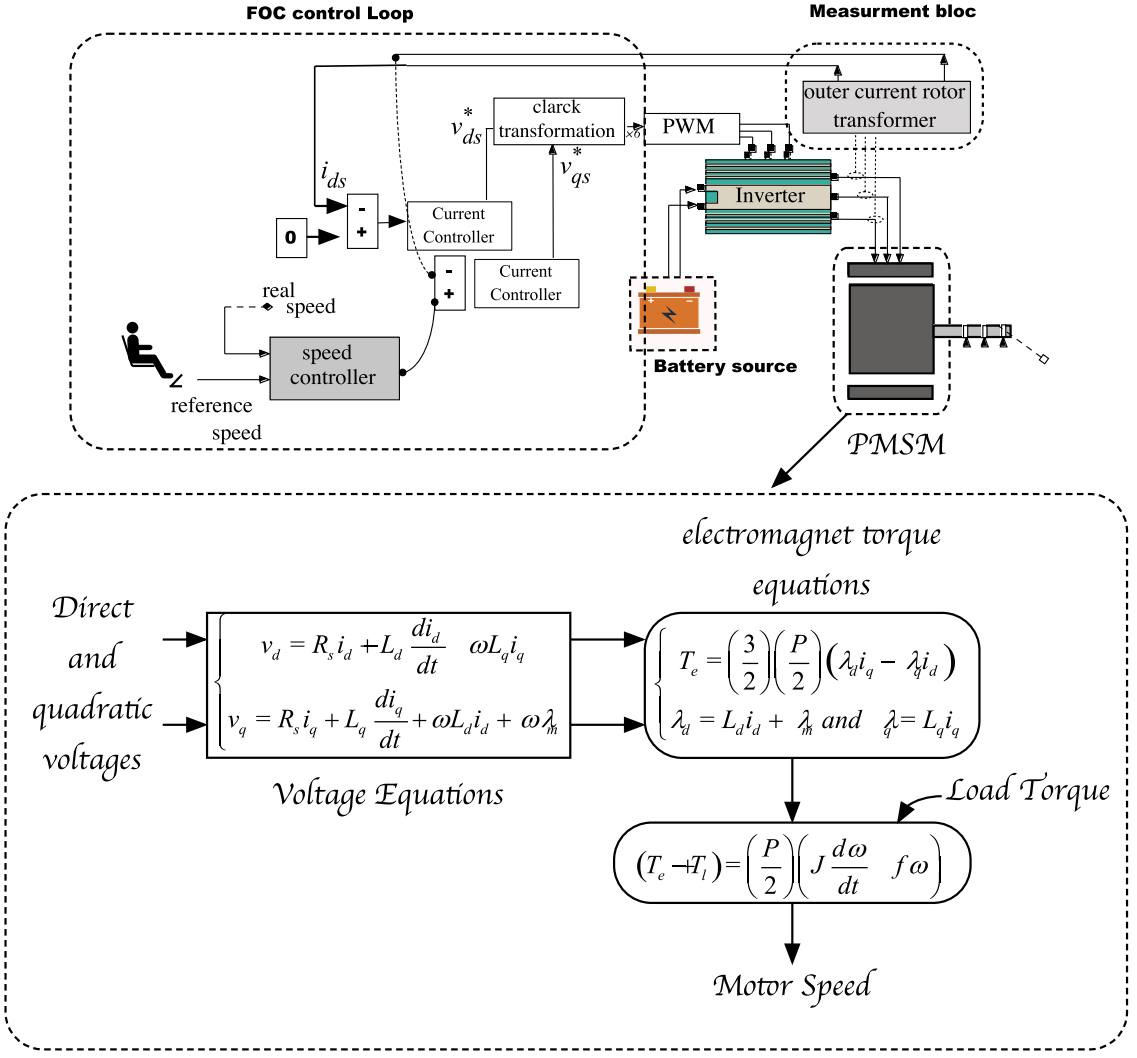


Fig. 1. Field-oriented vector control scheme (FOC).

In this context, G represents the improvement gain matrix selected to ensure specific error characteristics are met. "k_1" and "k_2" are two bounded positive real values. By taking the difference between the adjustable parameter system equation (7) and the state system equation (6) the resulting equation is expressed as equation (8). Thus, the nonlinear time-varying feedback system, consisting of a feedforward linear model combined with a nonlinear feedback expression, can be represented as it is in equation (9).

$$\dot{e} = \Delta A e + \Delta B U + \Delta C + G e \tag{8}$$

$$\text{we note as variable : } \dot{e} = \dot{\hat{X}} - \dot{\hat{X}}, e = X - \hat{X},$$

$$\Delta A = A - \hat{A}, \Delta B = \dot{B} - \hat{B}, \Delta C = C - \hat{C}$$

$$\dot{e} = (A + G)e + w \tag{9}$$

$$w = -\Delta_A \hat{X} - \Delta_B U - \Delta_C. \tag{10}$$

The stability analysis of the system can be addressed using the Popov stability theory, which requires the satisfaction of two essential conditions, as outlined in Flah *et al.* [24] and Gabbi *et al.* [25]. The transfer function matrix of the linear feedforward block must be real and strictly positive. Gao *et al.* [8] confirms that the gain matrix (G) as defined in equations (8) and (9) satisfies this requirement.

$$\int_0^{t1} w^T e \mathrm{d}t \geq -\gamma^2.$$

It is noted that γ^2 is a constant and has a positive sign. By expanding the second Popov condition outlined in equation (11) three additional conditions are derived, as expressed in equations (12)–(14). These conditions are met when the parameter adaptive laws are designed in a PI style, as shown in the resulting equations (15)–(17):

$$\int_0^{t1} w^T e \mathrm{d}t = \int_0^{t1} e^T w \mathrm{d}t =$$

$$-\int_0^{t1} e^T \left[\left(\Delta_A \hat{X} \right) + (\Delta_B U) + (\Delta_C) \right] \geq -\gamma^2 \tag{11}$$

$$\int_0^{t1} e^T \left(\hat{A} - A \right) \hat{X} \, \mathrm{d}t \geq -\gamma_A^2 \tag{12}$$

$$\int_0^{t1} e^T \left(\hat{B} - B \right) U \mathrm{d}t \geq -\gamma_B^2 \tag{13}$$

$$\int_0^{t1} e^T \left(\hat{C} - C \right) \mathrm{d}t \geq -\gamma_C^2 \tag{14}$$

$$\Delta A = \int_0^t \left(e^T \hat{X} \cdot f_{ir} \cdot + e^T \hat{X} \cdot f_{pr} \right) \mathrm{d}t \tag{15}$$

$$\Delta B = \int_0^t \left(e^T \cdot U \cdot f_{il} \cdot + e^T \cdot U \cdot f_{pl} \right) \mathrm{d}t \tag{16}$$

$$\Delta C = \int_0^t \left(e^T \cdot f_{ic} \cdot + e^T \cdot f_{pc} \right) \mathrm{d}t. \tag{17}$$

So it is possible to rewrite, the adaptive parameters expressions as in equations (18)–(20) respectively to stator resistance, stator inductance, and magnet flux.

$$\frac{\hat{R}_s}{L_s} = -\left(k_{pr} + \frac{k_{ir}}{s} \right) \left(\hat{i}_d e_d + \hat{i}_q e_q \right) + \frac{\hat{R}_s}{\hat{L}_s}(0) \tag{18}$$

$$\frac{1}{\hat{L}_s} = \left(k_{pl} + \frac{k_{il}}{s} \right) (u_d e_d + u_q e_q) + \frac{1}{\hat{L}_s}(0) \tag{19}$$

$$\frac{\hat{\lambda}_m}{\hat{L}_s} = -\left(k_{pf} + \frac{k_{if}}{s} \right) (\omega e_q) + \frac{\hat{\lambda}_m}{\hat{L}_s}(0). \tag{20}$$

Therefore, the global structure of the adaptive algorithm parameters identification is shown in Figure 2a. An anti-windup PI was used to eliminate the saturation problems, and the PI style used is illustrated in Figure 3.

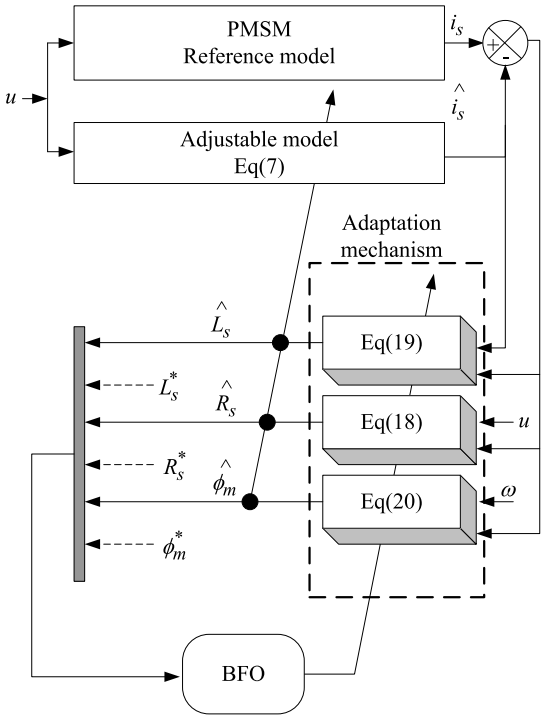
4.2 MRAS for PMSM speed estimation

One approach to efficiency estimation is the MRAS technique, which is often regarded as more effective than back-Electromagnet Force (EMF) and state observer techniques [3]. Some researchers associate this approach with intelligent estimation methods in the literature, including fuzzy logic and neural network-based solutions. However, these methods come with the challenge of requiring a comprehensive understanding of database information. Consequently, most research studies focus on methodologies utilizing mathematical models like MRAS or Luenberger observers. The mathematical models of the "reference and adjustable models," illustrated in the accompanying figure, serve as the basis for the MRAS principle (Fig. 2b). The outputs from these models are compared and processed through a specific algorithm to estimate or approximate a given parameter.

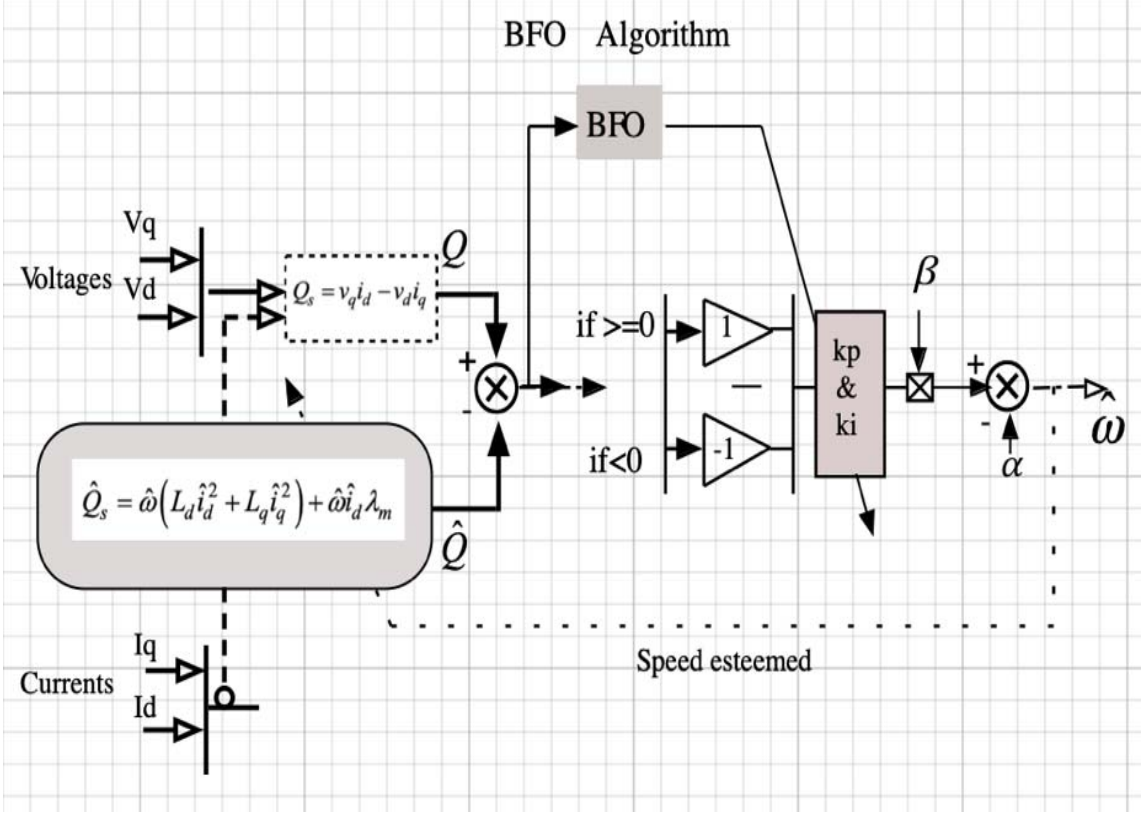
Additionally, this output signal is often used within the adjustable model itself. A key challenge with the MRAS estimator lies in stability concerns, as improper selection or configuration of the adaptation and adjustment mechanisms can lead to system instability. This issue can be effectively addressed by applying Popov's Hyper Stability Criterion.

To meet the specified application requirements, the speed of the PMSM will be estimated in real time, with the primary goal of replacing the mechanical speed encoder through a software-based approach. A flexible mathematical model must be developed using the relevant equations derived earlier as an initial step. The literature indicates that various types of MRAS observers can be employed, such as back-EMF-based MRAS (E-MRAS) reactive power-based MRAS (Q-MRAS), and active power-based MRAS (A-MRAS). However, the optimal choice of observer depends on a thorough understanding of the machine's internal design specifications. Each MRAS model offers unique benefits and faces specific limitations that should be carefully considered.

Examining the related works cited in Flah *et al.* [26], it is evident that certain standard MRAS estimators exhibit sensitivity to parameter variations, such as changes in stator resistance. The motor's operating temperature directly influences this parameter. When the temperature rises, the stator resistance may increase, leading to inaccuracies in the estimated parameters, including the stator resistance itself, deviating from actual values. The proposed model



(a)



(b)

Fig. 2. (a) MRAS-BFO tuning parameters identification scheme. (b) MRAS-BFO tuning speed identification scheme.

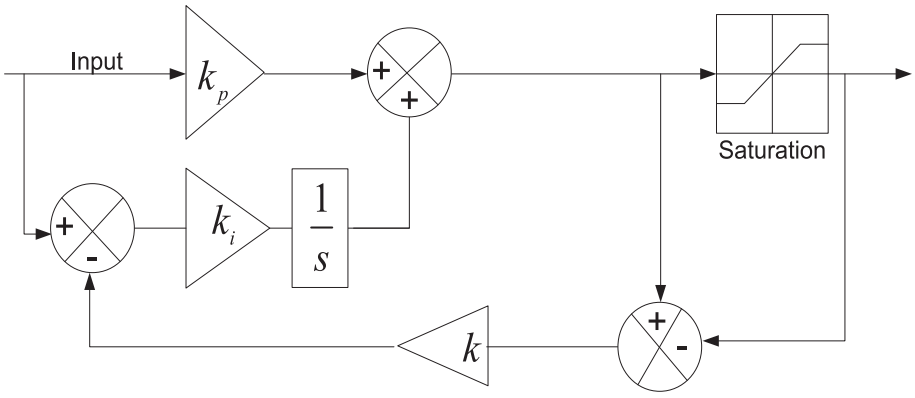


Fig. 3. Anti-windup PI.

offers a significant advantage over the referenced solutions, operating independently of this parameter. In high-speed modes, where elevated speeds often lead to increased motor temperatures, the resulting rise in stator resistance could impact performance. However, the Q-MRAS approach appears to address and mitigate this issue effectively.

Equation (21) shows the basic mathematical equation:

$$Qs = v_q i_d - v_d i_q. \tag{21}$$

From the other side, it is possible to rewrite the reactive power expression as it is in equation (22)

$$Q_s = \omega\left(L_d i_d^2 + L_d i_d^2\right) + \left(L_d i_d^2 \frac{\mathrm{d}i_d}{\mathrm{d}t} - L_a i_d \frac{\mathrm{d}i_q}{\mathrm{d}t}\right) + \omega i_d \lambda_m. \tag{22}$$

Equation (22) can be migrated to equation (23) where all currents can be assumed constant in the permanent regions where torque and motor speeds are constant.

Equation (24) which illustrates the adjustable Q-MRAS model, bases the stator currents as the direct and transversal components.

$$Q_s = \omega\left(L_d i_d^2 + L_q i_q^2\right) + \omega i_d \lambda_m \tag{23}$$

$$\hat{Q}_s = \hat{\omega}\left(L_d \hat{i}_d^2 + L_q \hat{i}_q^2\right) + \hat{\omega} \hat{i}_d \lambda_m. \tag{24}$$

The adaption protocol is constructed once the esteemed and real reactive power has been verified.

Equation (25) might clarify the power error value and the total reactive power produced by the MRAS estimator if each of equations (21) and (24) are used, followed by a subtraction action.

$$\begin{aligned} e_q = Q_s - \hat{Q}_s &= v_q i_d - v_d i_q - \left(L_d \hat{i}_d^2 + L_q \hat{i}_q^2\right) - \hat{\omega} \hat{i}_d \lambda_m \\ &= \alpha + \hat{\omega}\beta \end{aligned} \tag{25}$$

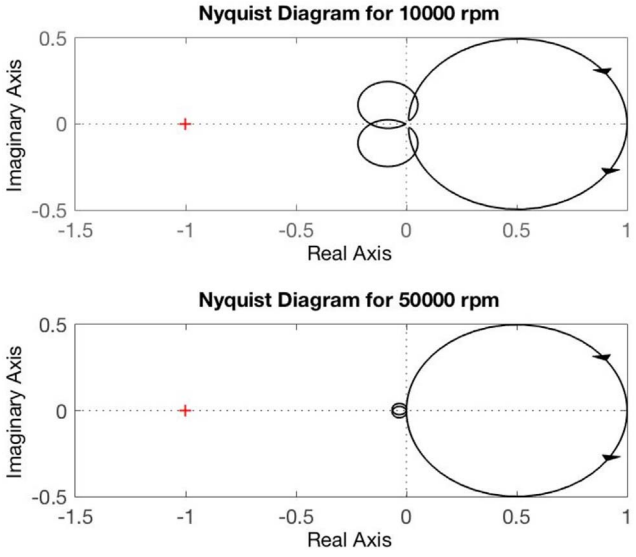


Fig. 4. Stability analysis of the MRAS loop, under various speed.

4.3 Stability analysis of the MRAS loop

A PI controller will regulate the overall transfer function, resulting in a closed-loop system with a third-order transfer function, as expressed in equation (26).

See the Equation (26) bottom of the page

with

$$\mathrm{RO} = \frac{R_s}{L_d}(v_q i_{q0} + v_d i_{d0}) + \omega(v_d d_{d0} - v_q i_{q0}).$$

As previously mentioned, flux variations occur when the motor operates above its nominal speed, necessitating sta-

$$G(s) = \frac{(v_q i_{q0} + v_d i_{d0})k_p s^2 + [\mathrm{RO}k_p s + (v_q i_{q0} + v_d i_{d0})k_i s] + \mathrm{RO}k_i}{s^3 + s^2\left[2\frac{R_s}{L_d} + k_p(v_q i_{q0} + v_d i_{d0})\right] + s[\omega^2 + \mathrm{RO}k_p + (v_q i_{q0} + v_d i_{d0})ki] + \mathrm{RO}k_i} \tag{26}$$

Table 2. A comparison between BFO, PSO, GA, DE, and SA.

Criteria	BFO	PSO	GA	DE	SA
Convergence speed	Moderate	Fast	Slow to moderate	Fast	Slow
Accuracy	High	Moderate	Moderate	High	Moderate
Robustness	High (Resistant to local minima)	Moderate (Prone to premature convergence)	Moderate (Depends on population size)	High (Good exploration-exploitation balance)	Low (Stochastic behavior affects stability)
Computational complexity	Moderate to High	Low	High	Moderate	Moderate
Exploration *vs.* Exploitation	Balanced	Good exploitation but weak exploration	More exploration than exploitation	Well-balanced	Highly stochastic, risk of poor exploitation
Handling of nonlinearities	Excellent	Moderate	Moderate	Good	Weak
Ability to escape local minima	Strong bacterial behavior prevents stagnation	Weak	Depends on mutation strategy	Strong (Adaptive mutation)	Weak (Prone to local minima)
Best suited for electrical machine parameter identification?	(Accurate and robust for nonlinear systems)	(Prone to local minima, less accurate)	(Slow convergence, needs large population)	(High computational cost)	(Unstable for complex systems)
References			[27–35]		

bility verification within this speed range. The stability assessment was conducted using a Nyquist plot of the closed-loop system, as illustrated in Figure 4 considering a speed range of "10,000 rpm" (below nominal speed) and "50,000 rpm" (within the high-speed region). In the following discussion, den($G(s)$) represents the denominator of $G(s)$ where P denotes the number of poles and Z the number of zeros. By plotting the Nyquist contour for the function (den(G(s)) − 1) it is observed that the resulting curves do not encircle the point $(-1 + j_0)$. Consequently, the number of encirclements "N" is "0". Given that the number of poles in the right-half plane is "$P = 0$", the number of zeros in the right-half plane is also "$Z = 0$". Based on the Nyquist criterion, it is concluded that the system remains stable.

5 Selection of the bacteria foraging optimization algorithm: advantage

The selection of the BFO algorithm was based on several comparison approaches based on recent research. The comparison was between BFO, Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Differential Evolution (DE), and Simulated Annealing (SA). The objective of this comparison has the objective to find the most suitable solution for electrical machine parameter identification. Table 2 highlights key aspects such as convergence speed, accuracy, robustness, and computational complexity, demonstrating why BFO is the most suitable choice.

Basing on these details, it can be seen that BFO will be highly effective for electrical machine parameter identification due to its superior handling of nonlinearities, robustness, and well-balanced exploration-exploitation capabilities. Unlike PSO and GA, which are prone to premature convergence, BFO effectively navigates challenging search spaces by mimicking the foraging behavior of bacteria. Through chemotaxis, BFO systematically explores the solution space while maintaining diversity, reducing the risk of getting trapped in local minima. It's swarming and reproduction mechanisms further refine solutions, ensuring higher accuracy in parameter estimation. Moreover, BFO's elimination and dispersal steps prevent stagnation, making it more stable than SA, which can suffer from excessive randomness. While DE offers strong search capabilities, it may require careful tuning of mutation factors, whereas BFO self-adapts through biological principles.

Bacterial forage optimization, a novel approach based on foraging behavior for resolving optimization problems, was introduced concerning *Escherichia coli* (*E. coli*) bacteria in the human gut. Therefore, under the law of bacteria, every element supports the best species to find food and eradicate the others. The chemotaxis step, swarming, reproduction, and elimination steps are the four processes that

make up the bacteria's foraging strategy [36]. The optimization approach can be determined from this equation by selecting the suitable equality in (27).

$$\mathrm{UPI} = \alpha \int_0^T e(t)\mathrm{d}t + \beta \int_0^T |e(t)|\mathrm{d}t + \gamma \int_0^T e^2(t)\mathrm{d}t. \quad (27)$$

Figure 4 shows the algorithm for bacterial foraging. If the bacterium goes in a specific way, it is considered swimming; if it moves in a completely different direction, it is tumbling. The chemotactic step represents the tactics between swimming and tumbling that the bacteria chooses in its motion. Equation (28) provides a mathematical update of the bacterial position:

$$\theta_i(j+1,k,l) = \theta_i(j,k,l) + C(i)\frac{\Delta s(i)}{\sqrt{\Delta s^T(i)\Delta s(i)}}. \quad (28)$$

The best-set bacteria get divided into two groups, where the best half replaces the second, which is eliminated. This part is called the reproduction step. The elimination and dispersal step, based on the probability value P_{ed}, consists of eliminating and dispersing the existing bacteria position in a new random direction if the probability value is less than a rand value [37]. The overall BFO flowchart can be seen in Figure 5c.

6. Results discussion

6.1 Real prototype and simulation configuration

The proposed control scheme for the PMSM is presented in Figure 5, alongside the experimental setup designed to validate the high-speed control loop. The overall hardware configuration is also detailed in Figure 5. The PMSM is integrated with a compressor system capable of achieving extremely high speeds by injecting high-pressure oil, effectively simulating high-speed operating conditions. The key specifications of the electrical machine used in this study are outlined in Table 3.

The DC bus voltage UDC max = 660 V for the inverter parameter, and the switching frequency is 10 KHz. After according to the different components in the Matlab/Simulink environment, a PMSM parameters variation and a PMSM speed variation were applied to verify the effectiveness of the proposed multiple estimators. As in the electric vehicle application, the environment supported by the PMSM comports many effects like temperature rise and dust or vibration variations, *etc.* In this section, our goal challenge is the robustness of the global system parameters identification.

6.2 MRAS performances under MRAS without optimal tuning

It is widely recognized that these parameters are typically adjusted manually. However, this approach can lead to inaccuracies due to mismatches in the parameter values, often resulting in an unstable identification response. Initially, three manual tests were conducted to adjust the PI parameters. The results highlighted the significant impact of PI parameter variations on the estimator's response for the three PMSM parameters. It can be seen in Figure 6 where three kinds of PI cases were applied to the MRAS model. The related performances of the MRAS parameters estimation, can be seen in those figures. In the case of Figure 6a where the stator resistance behaviors have three different situations. The same for the Figures 6b and 6c respectively for the Stator inductance and magnet flux estimation.

6.2 MRAS performances under MRAS BFO tuning

The proposed approach dynamically adjusts the PI adaptation mechanism parameters in real-time to address this issue. The MRAS-based PMSM parameter estimator is integrated into the overall control scheme to accurately identify parameter variations during operation. Section 4 presents the MRAS estimator, with its structure incorporating three PI adaptation mechanisms.

These parameters are coded as k_{pf}, k_{if}, k_{pl}, k_{il}, k_{pr}, and k_{ir}, respectively to permanent magnet flux, stator inductance and resistance identifiers. To ensure estimation convergence and stability, these parameters require dynamic online adjustment. This work proposes an intelligent tuning method based on the BFO algorithm. To implement the BFO algorithm, certain initial data must be predefined, as outlined in Table 4.

The adjusting PI parameter will start executing the BFO algorithm presented in Figure 4. The three fitness cost functions to be minimized are in equations (29)–(31).

$$\min f_{R_s} = \alpha_1 \int e_{R_s}^2 + \beta_1 \times \mathrm{overshoot}_{R_s} \quad (29)$$

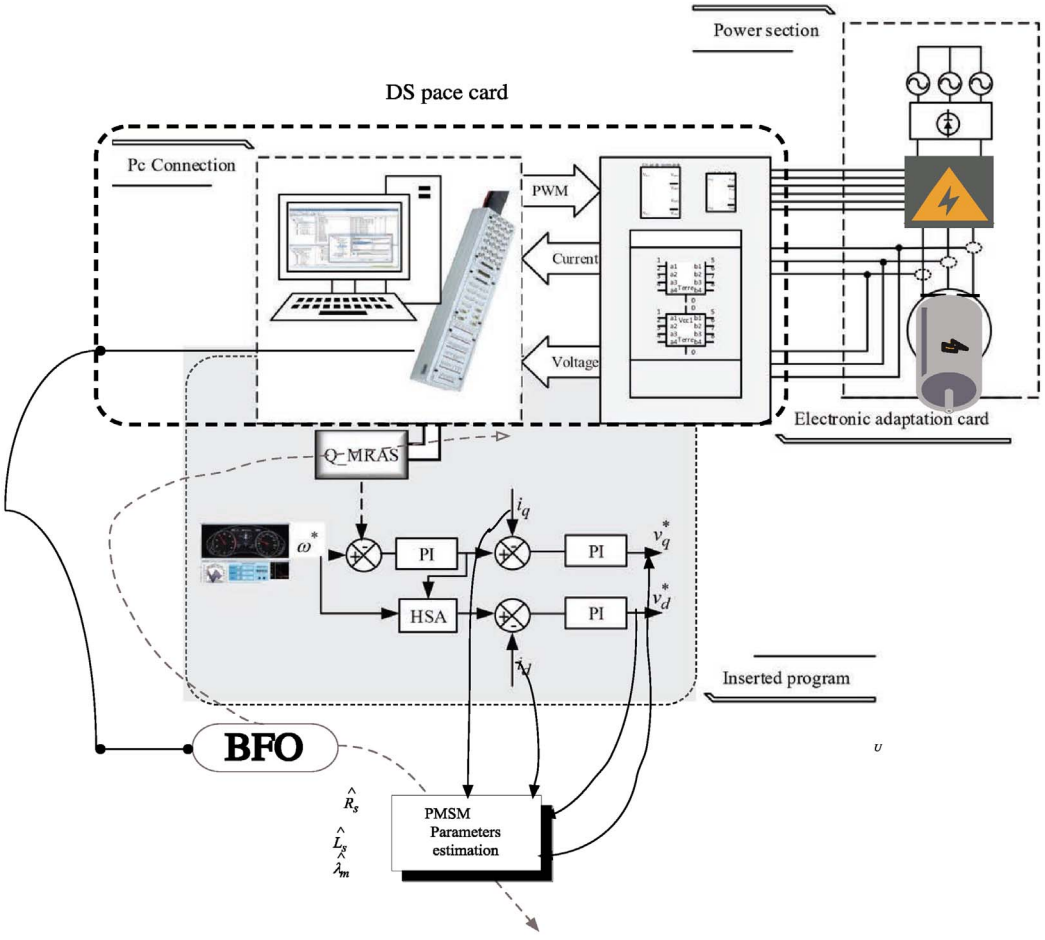
$$\min f_{L_s} = \alpha_2 \int e_{L_s}^2 + \beta_2 \times \mathrm{overshoot}_{L_s} \quad (30)$$

$$\min f_{\lambda_m} = \alpha_3 \int e_{\lambda_m}^2 + \beta_3 \times \mathrm{overshoot}_{\lambda_m} \quad (31)$$

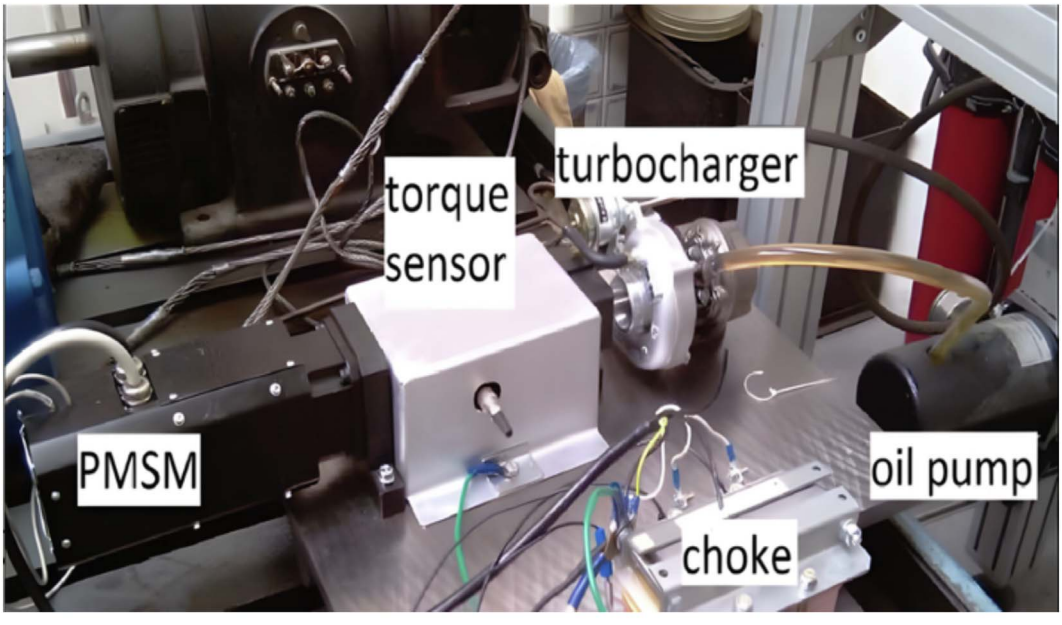
where, e_{R_s}, e_{L_s} *and* e_{λ_m} are respectively the error between the reference and the estimated permanent magnet flux, stator resistance, and inductance.

The overshoots denoted by $overshoot_{\lambda_m}$, $overshoot_{R_s}$ *and* $overshoot_{L_s}$ are respectively those related to permanent magnet flux, stator resistance, and inductance estimation results. Throughout the execution of the algorithm, each bacterium in the population explores its surroundings during every chemotactic step within the reproduction loop, which is also nested in the elimination loop. The bacteria aim to identify the optimal position, corresponding to the lowest fitness cost. This optimal position is shared among the bacteria to update their respective positions. By the end of the iterations, the algorithm ensures that the best position, representing the minimum fitness cost, is selected during the final stages of reproduction, elimination, and chemotactic loops.

The evolution of the fitness cost and the PI parameters in equations from (29) to (31) are displayed in Figures 7–9. The results were proved for two different configuration for

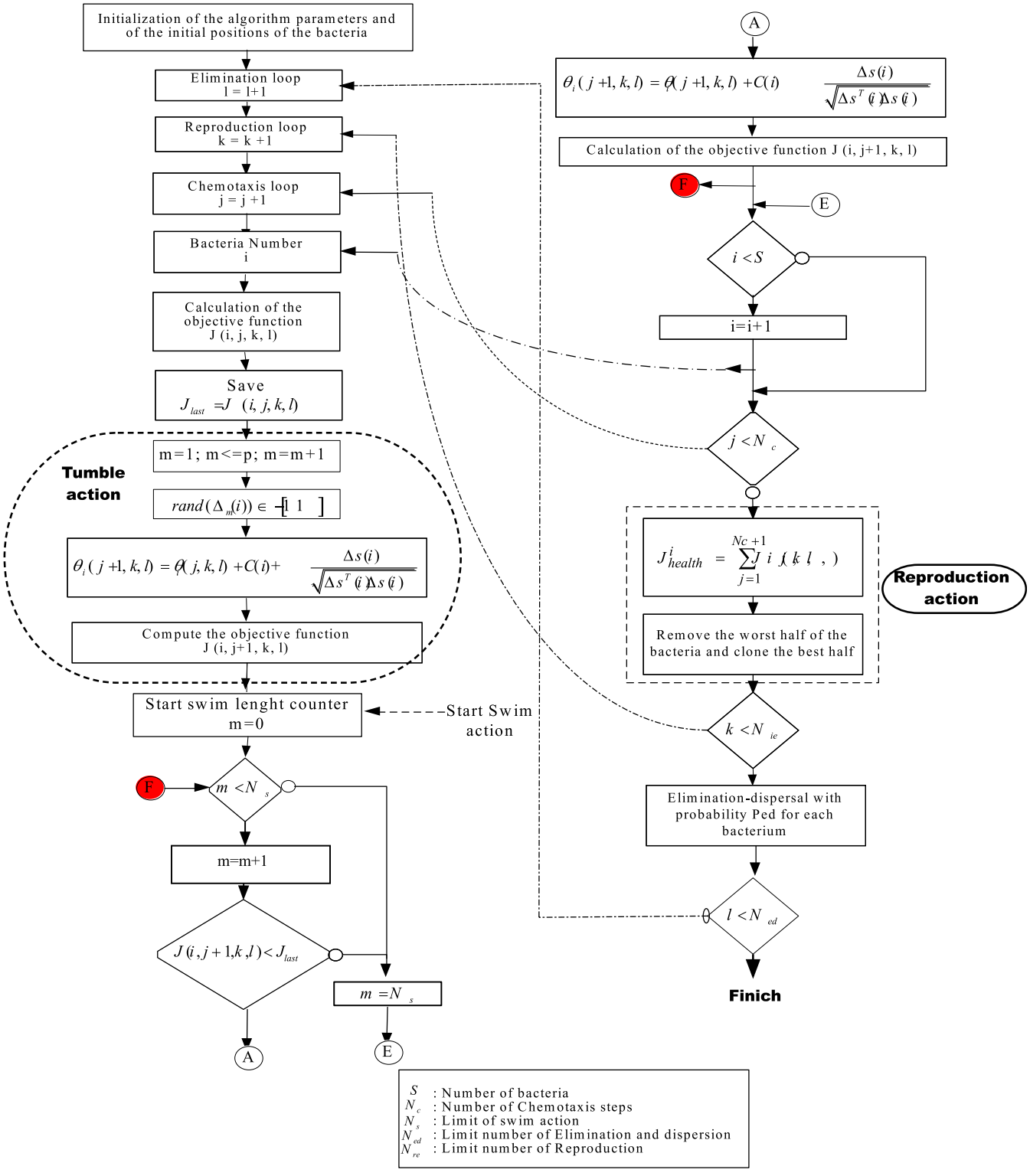


(a)



(b)

Fig. 5. (a) Block diagram of the PMSM control system. (b) exprimental steup. (c) BFO flowchart.



(c)

Fig. 5. Continued

Table 3. Electrical machine parameters.

Motor model	2AML406B-090-10-170
Speed	25 K rpm
rated current	11 A
Control topology	FOC
Ld, Lq, Ls (stator inductance)	21.5 mH
Stator resistance	5.2 **Ω**
Permanent flux ($\mathbf{\Phi}_m$)	0.24 Wb
Inertia factor	Not specified

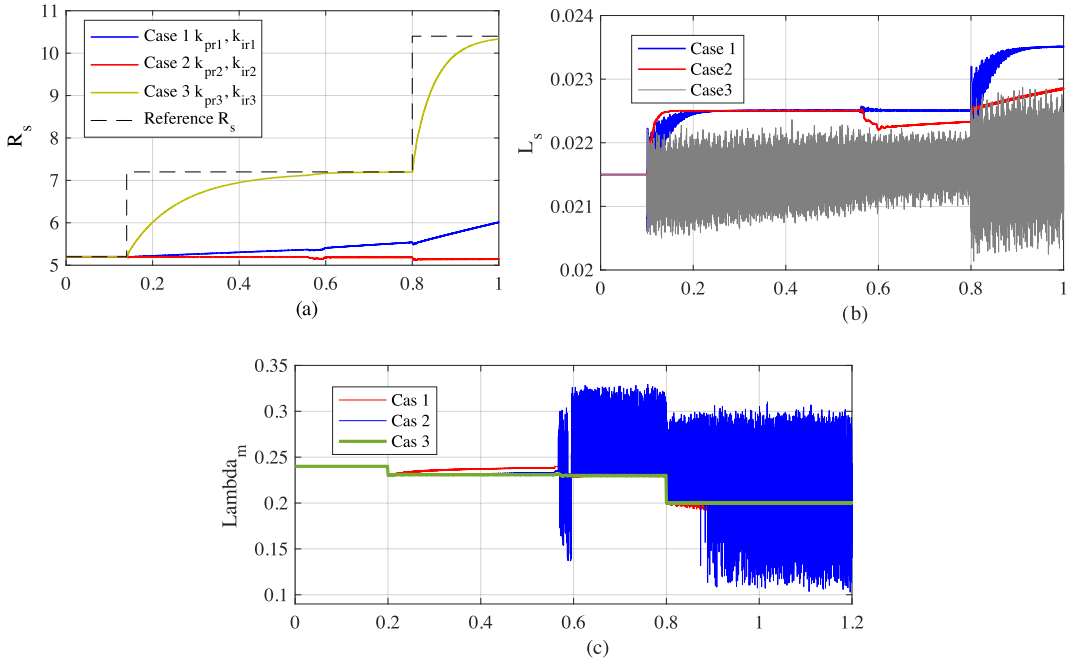


Fig. 6. Influence of manual PI parameters adjusting on estimating the PMSM parameters. (a) Stator resistance, (b) stator inductance, (c) Magnet flux.

the BFO algorithm. The first configuration, is based on Reproduction steps size ($N_{re} = 1$) and Elimination-dispersal events size ($N_{ed} = 1$) and the second configuration is based on Reproduction steps size ($N_{re} = 2$) and Elimination-dispersal events size ($N_{ed} = 2$). This is was for height bacteria in the first and last reproduction and elimination loop for the four chemotactic steps.

Effectively Figures 7a, 8a, and 9a show the results when the configuration in the BFO was based on ($N_{re} = 1$ and $N_{ed} = 1$). The fitness cost value is not fixed yet for the "s" bacteria, where each one searches for the best position, especially in the stator resistance and inductance case due to the initial bacteria position. However, in the permanent magnet flux case, the fitness cost is approximately the same in the beginning and the final iterations. These results have not guide to a precise response and the static error of the estimation phase can perturb all the process. The next step involved enhancing the BFO architecture by increasing the reproduction step size ($N_{re} = 2$) and the elimination-dispersal event size ($N_{ed} = 2$). The resulting behavior is illustrated in Figures 7b, 8b and 9b.

It can be observed that the last bacterium, identified as Ne4, consistently converges to a stable form in relation to the fitness cost in most cases. In Figure 7b, all bacteria have reached the same optimal fitness cost, indicating a single feasible solution. Similarly, Figure 8b shows that Ne4 maintains a nearly constant fitness cost of 0.06, highlighting its stability. Regarding magnet flux estimation, most bacteria, including Ne4, converge within a similar fitness cost range, demonstrating robustness in parameter tuning. Based on these results, the PI parameters will be fixed and applied to the MRAS model.

While these outcomes are promising, further performance improvements could be achieved by adjusting step size, the number of reproduction steps, and elimination events. However, it is crucial to note that such modifications may impact the computational efficiency and processing speed of the entire optimization process.

Table 4. BFO parameters.

Parameter	Value
Bacteria elements (s)	8
Chemotactic steps (Ne)	4
Swim size (Ns)	4
Reproduction steps size (Nre)	2
Elimination-dispersal events size (Ned)	2
size of bacteria reproductions per generation (Sr)	s/2
Probability of elimination/dispersal (Ped)	0.225
Run length (c (:,1))	0.025 ones (s, 1)
Current positions (x1, x2, x3)	x1 = [k_{pr}; k_{ir}], x2 = [k_{pl}; k_{ilr}] x3 = [k_{pf}; k_{if}]

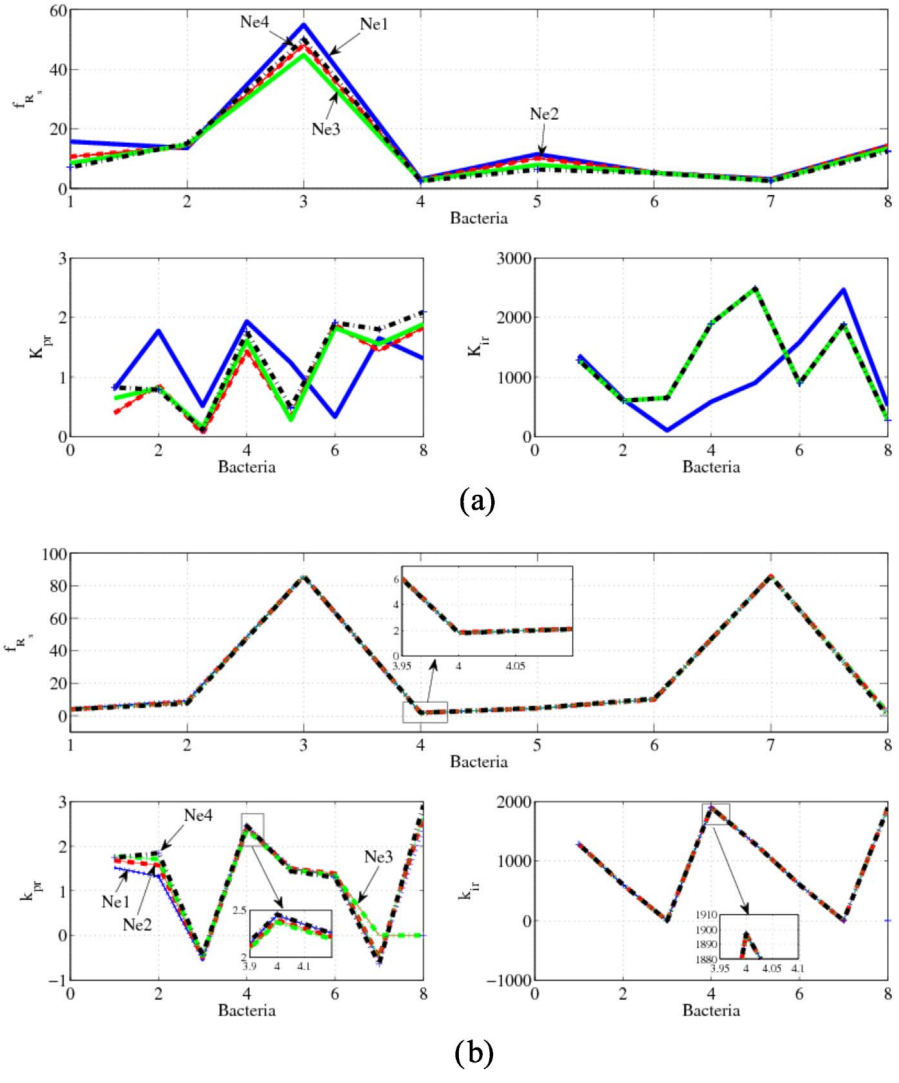


Fig. 7. Evolution of BFO parameters for the estimate of stator resistance. Two different cases in BFO: (a) Fitness cost for the Stator resistance parameter and the related PI parameters evolution for a Reproduction steps size ($N_{re} = 1$) and Elimination-dispersal events size ($N_{ed} = 1$); (b) Fitness cost for the Stator resistance parameter and the related PI parameters evolution for a Reproduction steps size ($N_{re} = 2$) and Elimination-dispersal events size ($N_{ed} = 2$).

The BFO algorithm's efficacy is seen in Figures 10a–10c, where a variable provided speed is used to determine the stator resistance, stator inductance, and permanent magnet flux. Starting from 0 rpm and increasing to 25,000 rpm and 40,000 rpm at $t = 0.58$ s. This clever approach yields an accurate assessment of PMSM parameters. The no-complete

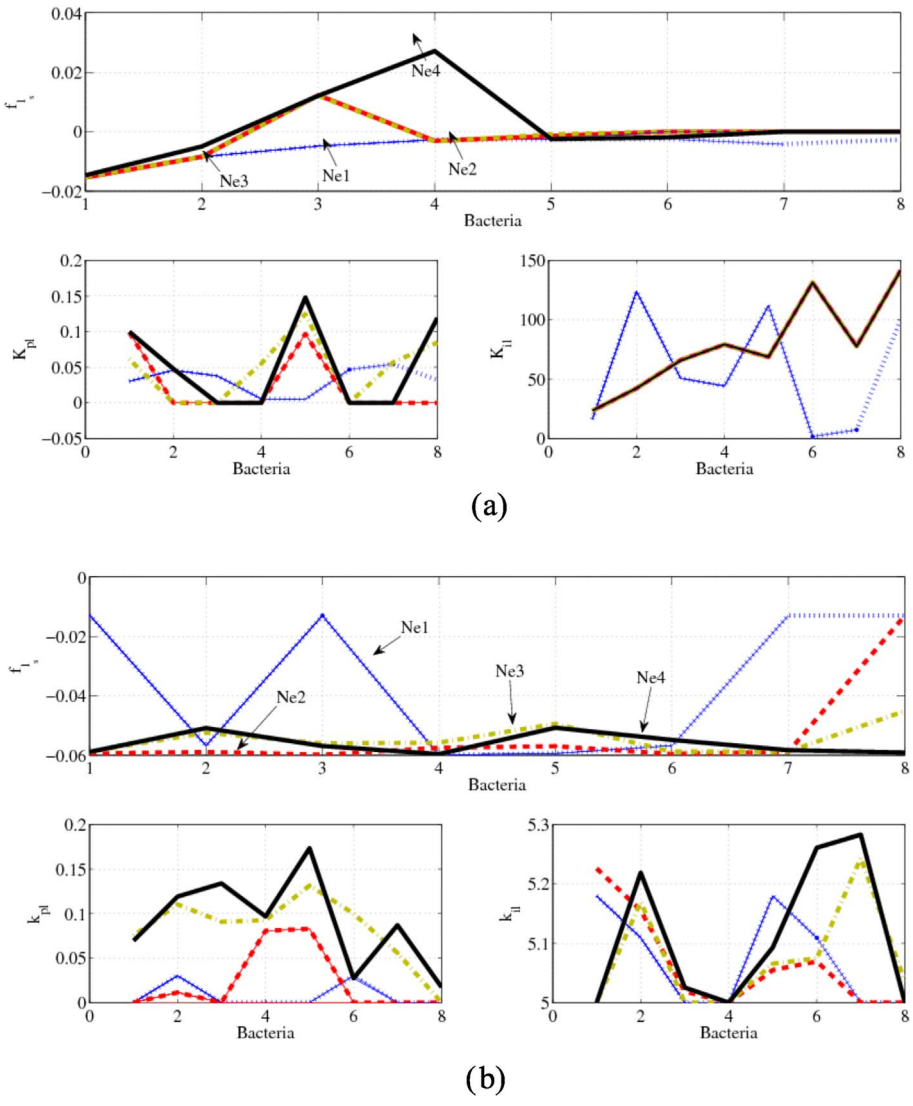


Fig. 8. Evolution of BFO parameters for the estimate of stator inductance. (a) Fitness cost for the Stator inductance parameter estimation and the related PI parameters evolution for a Reproduction steps size ($N_{re} = 1$) and Elimination-dispersal events size ($N_{ed} = 1$). (b) Fitness cost for the Stator inductance parameter estimation and the related PI parameters evolution for a Reproduction steps size ($N_{re} = 2$) and Elimination-dispersal events size ($N_{ed} = 2$).

bacteria convergence has an impact on the estimation results in Figure 10b where there is less chattering.

After demonstrating the effectiveness of the proposed MRAS-BFO estimation tool for PMSM parameter identification, the next step was to evaluate its adaptability for PMSM speed estimation. A real-world application was implemented using the prototype specifications outlined in Table 3 to achieve this.

A specific test scenario was designed based on a high-speed operating mode, with the target speed set at 32,000 rpm (650 Hz). This configuration was chosen to simulate a realistic application, such as an electric vehicle drivetrain, where high-speed operation is critical. The objective was to assess whether the proposed estimation framework remains accurate and reliable under demanding operating conditions. Figure 11 presents the obtained speed results. The real speed follows the desired one, with a chattering about the reference speed starting especially at $t = 0.04$ s. In this case, the system operates at 650 Hz, significantly impacting the commutation blocks integrated into the software speed encoder (specifically, the Sign Block shown in Fig. 2b). As a result, noticeable chattering occurs at this frequency. To address this issue, it is advised that the operating frequency be limited to a maximum of 400 Hz for this version of the software speed encoder. Figure 11 illustrates the speed error observed between the measured and estimated values.

The efficiency of the proposed MRAS estimator was validated through the previously presented results. However, additional algorithms were implemented and tested to further enhance the discussion, including the Kalman Filter and the classical MRAS as benchmark methods.

The evaluation focused on key performance metrics such as estimation accuracy, chattering levels, response time,

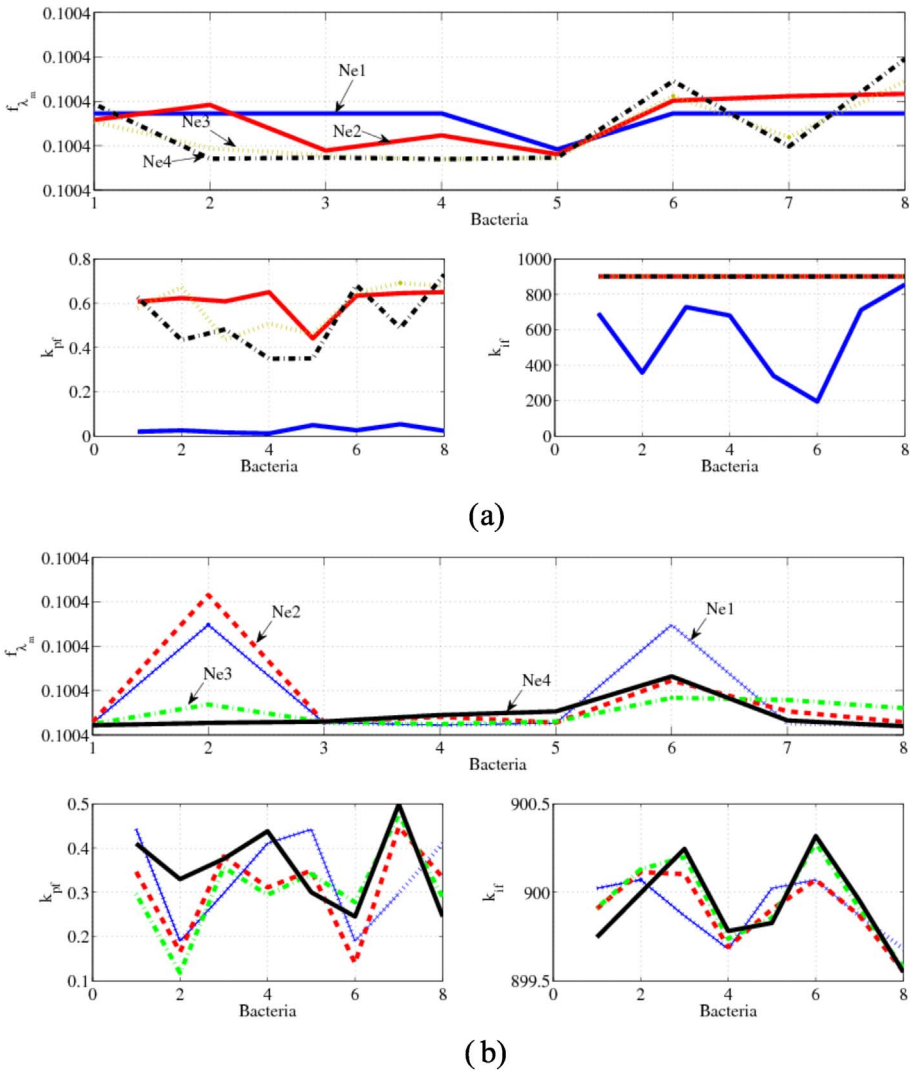


Fig. 9. BFO parameters evolution for the permanent magnet flux estimation. (a) Fitness cost for the magnet Flux parameter estimation and the related PI parameters evolution for a Reproduction steps size ($N_{re} = 1$) and Elimination-dispersal events size ($N_{ed} = 1$). (b) Fitness cost for the magnet Flux parameter estimation and the related PI parameters evolution for a Reproduction steps size ($N_{re} = 2$) and Elimination-dispersal events size ($N_{ed} = 2$).

and system speed. All tests were conducted under the same computational conditions as previously described.

Table 5 presents a comparative analysis of these methods, highlighting their performance in parameter and speed estimation. The results provide deeper insight into the advantages and limitations of the proposed MRAS-BFO approach relative to traditional estimation techniques. Table 5 shows that the MRAS-BFO method achieves the highest precision level, with a 93% accuracy rate, outperforming the conventional MRAS by 12% and the Kalman Filter approach by 5%. In terms of rapidity, the conventional MRAS demonstrates superior performance, as it does not require parallel optimization computations, making it inherently faster. This is reflected in its higher rapidity factor than the optimized MRAS-BFO method. On the other hand, the proposed approach significantly reduces chattering, showing a 5% improvement over the conventional MRAS method. This reduction enhances system stability and smoothness, making the MRAS-BFO method a more reliable and robust solution for practical applications.

6.4 Limitations and feasibility weaknesses

Despite the good performance and robustness of the proposed MRAS-BFO solution, it is important to highlight the computational resources used in the implementation. The tests were conducted on a system equipped with an Intel Core i7 processor (3.4 GHz) and 8 MB of RAM. However, these tests were performed under simple speed and parameter variations, without evaluating the algorithm's adaptability to rapid successive changes. Therefore, its suitability for direct practical application still requires further validation. Given these considerations, discussing the computational complexity, processing time limitations, and feasibility challenges for real-time applications is

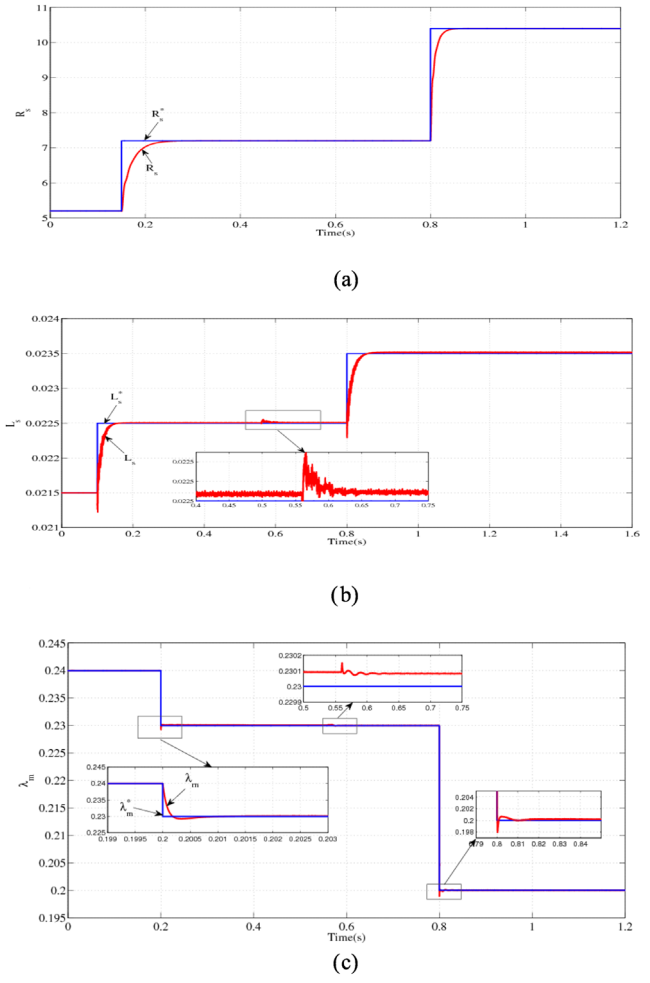


Fig. 10. A comparison between the real PMSM parameters and the Parameter estimation with the BFO algorithm. (a) Stator resistance performance: Real variation and the estimation result. (b) Stator inductance performances: Real variation and estimation result. (c) Magnet Flux performances Real variation and estimation result.

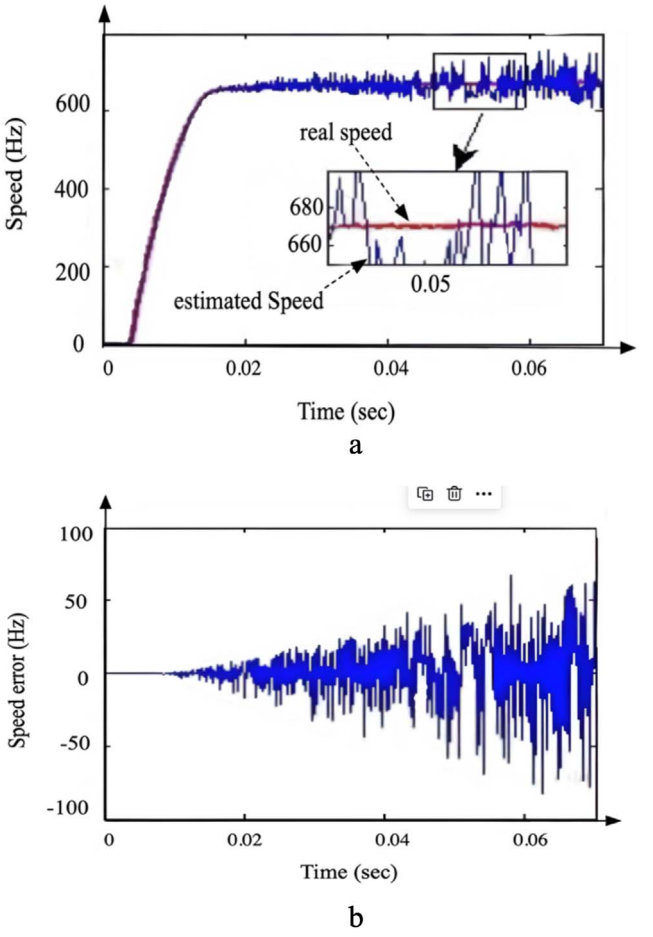


Fig. 11. Speed performances with the MRAS estimator. (a) Speed performance (target 670 Hz): Red color: The real speed, Blue color: the estimated speed. (b) Speed error between the real and esteemed MRAS speed for 670 Hz as speed target.

Table 5. Comparison between MRAS-BFO and 2 more conventional method.

Criteria	Proposed MRAS-BFO	Kalman filter	Classic MRAS
Estimation accuracy	High (93%)	Moderate (87%)	Moderate (81%)
Chattering level	Low (2%)	Low (5%)	High (7%)
Response time	Fast (0.012 s)	Moderate (0.014 s)	Fast (0.0009 s)
System rapidity	High	Moderate	High
Computational efficiency	Moderate (requires optimization steps)	High (well-structured recursive approach)	High

crucial. A thorough analysis of these factors will help assess whether the algorithm can meet real-world performance requirements and determine potential improvements for optimizing its efficiency. The estimation method based on the BFO algorithm presents several challenges related to computational complexity, processing time, and feasibility for real-time applications. BFO involves iterative processes such as chemotaxis, swarming, reproduction, and elimination-dispersal, significantly increasing computational overhead. The nonlinear nature of BFO optimization can lead to prolonged convergence times, making it unsuitable for real-time systems with stringent latency requirements.

Additionally, the extensive search space exploration of BFO demands high processing power, limiting its

applicability in resource-constrained environments. To mitigate these issues, high-performance computing hardware, including multi-core processors, GPUs, or FPGAs, is required to accelerate parallel processing. Moreover, implementing optimized numerical techniques can improve efficiency, such as adaptive step-size tuning and hybrid algorithms combining BFO with faster heuristics like PSO. Utilizing cloud or edge computing architectures can enhance feasibility by offloading heavy computations while maintaining real-time responsiveness. These solutions make BFO-based estimation more practical for time-sensitive applications.

7 Conclusion

In this study, the MRAS technique, based on the Popov hyper-stability criterion, was utilized as an online estimation method for identifying PMSM parameters and monitoring speed variation. To enhance the robustness and precision of the estimation process, the BFO algorithm was employed as an intelligent tuning mechanism for the MRAS approach. This optimization technique significantly improved the precision of the estimation process, achieving a 12% enhancement compared to traditional manual tuning methods. The BFO algorithm demonstrated superior performance as an online MRAS tuning mechanism, outperforming commonly used manual methods. The simulation and practice results validate the effectiveness of the proposed approach in achieving precise and robust online PMSM parameter identification and tuning.

Funding

The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number (PSAU/2024/01/31874).

Conflicts of interest

The authors declare no competing interests.

Data availability statement

The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Authors contribution statement

Mohammed M. Alrashed: Writing, formal analysis, and project management.

Mohamed F. Elnaggar: Writing original, concept, and mathematical analysis.

Aymen Flah: Funding, data analysis and concept, funding, software, and simulation practice

Claude Ziad El-Bayeh: Writing, photo treatment, and results analysis.

References

1 Guo Z.Q., Panda S.K. (2015) Design of a sliding mode observer for sensorless control of SPMSM operating at medium and high speeds, in: *2015 IEEE Symposium on Sensorless Control for Electrical Drives (SLED)*, Sydney, NSW, Australia, 7–8 June, IEEE, pp. 1–6.

2 Habibullah M., Lu D.D.C., Xiao D., Fletcher J.E., Rahman M.F. (2017) Predictive torque control of induction motor sensorless drive fed by a 3L-NPC inverter, *IEEE Trans. Industr. Inform.* **13**, 60–70.

3 Hammoumi D., El Bekkali C., Karim M., Taoussi M., El Ouanjli N., Bossoufi B. (2019) Direct controls for wind turbine with PMSG used on the real wind profile of Essaouira-Morocco City, *Indones. J. Electr. Eng. Comput. Sci.* **16**, 3, 1229–1239.

4 Flah A., Novák M., Sbita L., Novák J. (2014) Estimation of motor parameters for an electrical vehicle application, *Int. J. Model. Identif. Control* **22**, 2, 150–158.

5 Ademi S., Jovanović M. (2014) Maximum torque per inverter ampere control of brushless doubly-fed reluctance generators for wind turbines, in: *2014 International Symposium on Power Electronics, Electrical Drives, Automation and Motion*, Ischia, Italy, pp. 883–888.

6 Novak Z. (2024) Confidence weighted learning entropy for fault-tolerant control of a PMSM with a high-resolution hall encoder, *IEEE Trans. Ind. Electr.* **71**, 5, 5176–5186.

7 Ishikawa T., Seki Y., Kurita N. (2013) Analysis for fault detection of vector-controlled permanent magnet synchronous motor with permanent magnet defect, *IEEE Trans. Magn.* **49**, 2331–2334.

8 Gao P., Gu Y., Wang X. (2018) The design of a permanent magnet in-wheel motor with dual-stator and dual-field-excitation used in electric vehicles, *Energies* **11**, 2, 424.

9 Morandin M., Faggion A., Bolognani S. (2015) Integrated starter – alternator with sensorless ringed-pole PM synchronous motor drive, *IEEE Trans. Ind. Appl.* **51**, 1485–1493.

10 Eltuhamy R.A., Rady M., Almatrafi E., Mahmoud H.A., Ibrahim K.H. (2023) Fault detection and classification of CIGS thin-film PV modules using an adaptive neuro-fuzzy inference scheme, *Sensors* **23**, 3, 1280.

11 Stefanovski J. (2014) Kalman-Yakubovič-Popov lemma for descriptor systems, *Syst. Control Lett.* **74**, 8–13.

12 Zhu X., Li J., Zhong Y., Choi K.S., Shirinzadeh B., Smith J., Gu Cl. (2023) Iterative Kalman filter for biological tissue identification, *Int. J. Robust Nonlinear Control.*, 1–13. https://doi.org/10.1002/rnc.6742

13 Qi H., Zhang Y., Gao N. (2015) Research and implement of PMSM regenerative braking strategy based on controllable rectification, in: *2015 IEEE 11th International Conference on Power Electronics and Drive Systems*, Sydney, NSW, Australia, 9–12 June, IEEE, pp. 289–294.

14 Richardson C.R., Turner M.C., Gunn S.R. (2024) Strengthened Circle and Popov criteria and the analysis of ReLU neural networks, in: *2024 UKACC 14th International Conference on Control (CONTROL)*, Winchester, UK, 10–12 April, IEEE, pp. 127–128.

15 Kumar D., Sunori S.K., Jain S. (2024) Bacterial foraging optimization of electric vehicle charging systems, in: *2024 International Conference on Sustainable Communication Networks and Application (ICSCNA)*, Theni, India, 11–13 December, IEEE, pp. 267–272.

16 He C., Hu J., Ran X., Wei F., Li Y., Zhu Y. (2024) A simplified constrained predictive position control for PMSM drives with laguerre functions, *IEEE Trans. Ind. Electron.* **71**, 12, 15478–15487.

17 Salahuddin H., Imdad K., Chaudhry M.U., Iqbal M.M., Bolshev V., Hussain A., Flah A., Panchenko V., Jasiński M.

(2022) Electric vehicle transient speed control based on vector control FM-PI speed controller for induction motor, *Appl. Sci.* **12**, 17, 8694.
18 Farhat M., Barambones O., Flah A., Sbita L. (2016) Variable structure MPP controller for photovoltaic pumping system, *Trans. Inst. Meas. Control.* **39**, 9, 1283–1292.
19 Feifei H., Zhonghua W., Yueyang L., Tongyi H. (2015) Sensorless speed control of permanent magnet synchronous motor based on RBF neural network, in: *2015 34th Chinese Control Conference* (*CCC)*, Hangzhou, China, 28–30 July, IEEE, pp. 4325–4330.
20 Itani K., De Bernardinis A., Zoubir K., Jammal A. (2016) Extreme conditions regenerative braking modeling, control and simulation of a hybrid energy storage system for an electric vehicle, *IEEE Trans. Transp. Electrif.* **7782**, 99, 1–16.
21 Aymen F., Mohamed N., Chayma S., Reddy R., Alharthi M., Ghoneim S.S.M. (2021) An improved direct torque control topology of a double stator machine using the fuzzy logic controller, *IEEE Access* **9**, 1.
22 Gritli Y., Di Tommaso A.O., Miceli R., Filippetti F., Rossi C. (2013) Vibration signature analysis for rotor broken bar diagnosis in double cage induction motor drives, in: *4th International Conference on Power Engineering, Energy and Electrical Drives,* Istanbul, Turkey, 13–17 May, IEEE, pp. 1814–1820.
23 Oubelaid A., Taib N., Nikolovski S., Alharbi T.E.A., Rekioua T., Flah A., Ghoneim S.S.M. (2022) Intelligent speed control and performance investigation of a vector controlled electric vehicle considering driving cycles, *Electronics* **11**, 1925.
24 Flah A., Majed A., Bajaj M., Naveen K.S., Mishra S., Sharma S.K. (2021) Electric vehicle model based on multiple recharge system and a particular traction motor conception, *IEEE Access* **9**, 49308–49324.
25 Gabbi T.S., Gründling H.A., Vieira R.P. (2016) Sliding mode MRAS speed observer applied to Permanent Magnet Synchronous Motor with decoupled current control, in: *IECON 2016 – 42nd Annual Conference of the IEEE Industrial Electronics Society,* Florence, Italy, 23–26 October, IEEE, pp. 2929–2934.
26 Flah A., Novak M., Lassaad S. (2018) An improved reactive power MRAS speed estimator with optimization for a hybrid electric vehicles application, *ASME J. Dyn. Syst. Meas. Control.* **140**(6), 061016. https://doi.org/10.1115/1.4039212.
27 Aminu M. (2019) A parameter estimation algorithm for induction machines using Artificial Bee Colony (ABC) optimization, *Niger. J. Technol.* **38**, 1, 193.
28 Antonelli S.L., Donolo P.D., Pezzani C.M., Quispe E.C., De Angelo C.H. (2023) Identification of induction motor parameters using genetic algorithms, in: *2023 IEEE Workshop on Power Electronics and Power Quality Applications* (*PEPQA)*, Cali, Colombia, 5–6 October, IEEE, pp. 1–7.
29 Çanakoğlu A.I., Yetgin A.G., Temurtaş H., Turan M. (2014) Induction motor parameter estimation using metaheuristic methods, *Turk. J. Elec. Eng. Comp. Sci.* **22**(5), Article 6. https://doi.org/10.3906/elk-1211-171
30 Duan F., Živanović R., Al-Sarawi S., Mba D. (2016) Induction motor parameter estimation using sparse grid optimization algorithm, *IEEE Trans. Ind. Inform.* **12**, 4, 1453–1461.
31 Hafez I., Dhaouadi R. (2023) Identification of mechanical parameters in flexible drive systems using hybrid particle swarm optimization based on the quasi-newton method, *Algorithms* **16**, 8, 371.
32 Koubaa Y. (2004) Recursive identification of induction motor parameters, *Simul. Model. Pract. Theory* **12**, 5, 363–381.
33 Montoya O.D., De Angelo C.H., Bossio G. (2024) Parametric estimation in three-phase induction motors using torque data via the generalized normal distribution optimizer, *Results Eng.* **23**, 102446.
34 Nikranajbar A., Ebrahimi M.K., Wood A.S. (2010) Parameter identification of a cage induction motor using particle swarm optimization, *Proc. Inst. Mech. Eng. Part I J. Syst. Contr. Eng.* **224**, 5, 479–491.
35 Yang J., Shen Y., Tan Y. (2024) Parameter compensation for the predictive control system of a permanent magnet synchronous motor based on bacterial foraging optimization algorithm, *World Electr. Veh. J.* **15**, 1, 23.
36 Hung J.C. (2016) Memetic particle swarm optimization algorithm for DOA estimation under multipath environment, in: *2016 2nd International Conference on Control Science and Systems Engineering* (*ICCSSE),* Singapore, 27–29 July, IEEE, pp. 36–41.
37 Flah A., Sbita L. (2013) A novel IMC controller based on bacterial foraging optimization algorithm applied to a high speed range PMSM drive, *Appl. Intell.* **38**, 114–129.