



REVIEW

Review and Comparative Analysis of System Identification Methods for Perturbed Motorized Systems

Helen Shin Huey Wee and Nur Syazreen Ahmad*

School of Electrical and Electronic Engineering, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia

*Corresponding Author: Nur Syazreen Ahmad. Email: syazreen@usm.my

Received: 19 January 2025; Accepted: 08 May 2025; Published: 30 May 2025

ABSTRACT: This paper reviews recent advancements in system identification methods for perturbed motorized systems, focusing on brushed DC motors, brushless DC motors, and permanent magnet synchronous motors. It examines data acquisition setups and evaluates conventional and metaheuristic optimization algorithms, highlighting their advantages, limitations, and applications. The paper explores emerging trends in model structures and parameter optimization techniques that address specific perturbations such as varying loads, noise, and friction. A comparative performance analysis is also included to assess several widely used optimization methods, including least squares (LS), particle swarm optimization (PSO), grey wolf optimizer (GWO), bat algorithm (BA), genetic algorithm (GA) and neural network for system identification of a specific case of a perturbed DC motor in both open-loop (OL) and closed-loop (CL) settings. Results show that GWO achieves the lowest error overall, excelling in OL scenarios, while PSO performs best in CL due to its synergy with feedback control. LS proves efficient in CL settings, whereas GA and BA rely heavily on feedback for improved performance. The paper also outlines potential research directions aimed at advancing motor modeling techniques, including integration of advanced machine learning methods, hybrid learning-based methods, and adaptive modeling techniques. These insights offer a foundation for advancing motor modeling techniques in real-world applications.

KEYWORDS: Motor modeling; data-driven modeling; particle swarm optimization; genetic algorithm; grey wolf optimization

1 Introduction

Motorized systems are fundamental to a wide range of advanced applications, driving a diverse array of devices including industrial machinery, semiconductor wafer production, robotics, and electric vehicles. The crux of leveraging these systems lies in the development of computational or mathematical models that faithfully emulate their dynamic behavior. Such motor models are tailored to meet the varying degrees of sophistication and accuracy required by their application domains, empowering researchers and engineers to simulate, predict, and fine-tune system performance with high precision [1]. These models are informed by data sourced from an array of sensors and other inputs, translating into precise mathematical representations of the motors' dynamics [2]. Through this modeling, intricate insights into the functioning of motorized systems under diverse conditions are unearthed [3].

The challenge becomes even more critical when dealing with perturbed motorized systems—those subjected to disturbances, noise, and environmental variations—which directly impact their efficiency, reliability, and overall performance. In such systems, precise system identification is essential, as it enables the



detection, quantification, and modeling of perturbation effects. Accurate identification lays the foundation for developing robust and adaptive control strategies capable of anticipating and mitigating unpredictable operational disruptions [4]. Without a well-structured approach to system identification, control mechanisms may struggle to maintain stability and efficiency, particularly in dynamic environments.

This becomes especially crucial in high-precision industries such as aerospace, automotive engineering, and industrial automation, where even minor deviations can lead to significant performance degradation or safety risks. By integrating perturbation models into system identification, motorized systems can achieve enhanced resilience, stability, and adaptability, even under fluctuating operating conditions. Thus, advancing system identification techniques for perturbed motorized systems is not just an academic pursuit but a practical necessity for ensuring reliable and high-performance operation in increasingly complex and demanding applications.

Modeling approaches for dynamic systems can be broadly categorized into two principal types: data-driven modeling and first-principle modeling. First-principle modeling, also referred to as white-box modeling, relies entirely on fundamental physical laws, such as Ohm's law and Newton's laws of motion, to describe system dynamics without the need for experimental data [5]. In contrast, data-driven modeling constructs system representations using input-output data and includes black-box and grey-box approaches [6,7]. Black-box modeling develops models solely from input-output data without any knowledge of internal system structures, while grey-box modeling integrates both empirical data and partial system knowledge to improve model accuracy [8]. A thorough understanding of these modeling paradigms is crucial for building accurate system representations and designing effective control strategies.

In recent years, advanced nonlinear modeling techniques have been proposed to capture complex system behaviors with greater precision. Block-oriented models, including Hammerstein and Wiener models, are particularly prominent due to their capability to represent nonlinear systems through a combination of linear dynamic components and static nonlinearities [9–11]. Furthermore, neural networks and neuro-fuzzy systems have gained considerable attention owing to their powerful approximation capabilities and adaptive learning features, rendering them suitable for modeling highly nonlinear and time-varying systems [12]. These methodologies have been extensively investigated in the literature to enhance modeling accuracy and robustness.

The optimization of motor parameters through grey- and black-box modeling has been extensively studied in prior research, employing various algorithms. Least square (LS) serves as a foundational method for this purpose. An extension of LS, the Levenberg-Marquardt (LM) algorithm [13], enhances the traditional approach by incorporating mechanisms to improve convergence speed and robustness, especially in nonlinear optimization problems. Additionally, metaheuristic algorithms are pivotal in optimizing parameters based on selected cost functions. These algorithms are categorized into four main types: swarm intelligence, evolutionary, physics-based, and human-inspired algorithms. Swarm intelligence algorithms emulate cooperative behaviors among agents to tackle complex problems, drawing inspiration from social organisms such as ants and birds [14,15]. Notable examples include particle swarm optimization (PSO), where particles adjust their positions in the solution space based on personal and collective experiences [16]. Other prominent algorithms in this category include whale optimization algorithm (WOA), grey wolf optimization (GWO), ant colony optimization (ACO), artificial bee colony (ABC), bat algorithm (BA), and cuckoo search (CS) [17,18]. There are also novel swarm algorithms proposed including marine predator algorithm, artificial circulation system algorithm (ACSA), wolverine optimization algorithm (WoOA), and synergistic swarm optimization algorithm (SSOA) [19–21]. These techniques harness natural dynamics and social structures to enhance search efficiency effectively.

Evolutionary algorithms on the other hand imitate the evolutionary process to search for the optimal solutions in challenging situations. These algorithms, including differential evolution (DE) [22] and genetic algorithms (GA) maintain a population of potential solutions that evolve across generations through mutation, crossover, and selection methods. Human-based algorithms use human judgment and knowledge to address optimization challenges [23]. These algorithms often require the development of rule-based systems, manual optimization, and expert judgment, such as the jaya algorithm (JA), teaching–learning–based optimization (TLBO). Physics-based algorithms, in contrast, model the physical behaviors of systems through the application of physics and mathematics principles. These methods are crucial for precise simulation and analysis, commonly employing tools like computational fluid dynamics and finite element analysis [24]. Utilized extensively in engineering and scientific research, physics-based algorithms enable detailed investigation and optimization of complex physical phenomena, facilitating advanced developments in motor parameter optimization and other areas of applied mechanics.

Recent surveys in the field have predominantly focused on electromagnetic field modeling perspectives [25], laboratory testing methodologies [26], and parameter identification approaches tailored to specific types of motors such as permanent magnet synchronous motors (PMSM) [27], inverter-fed induction motors [28], three-phase induction motor drives [29], and permanent magnet DC motors [30]. While some surveys, such as [31], have addressed model structures, they often lack comprehensive discussions on model parameter optimization. None have systematically explored the relationship between different model structure types and parameter optimization methods concerning perturbations. A thorough exploration of parameter optimization methods and state-of-the-art techniques for managing perturbations remains underrepresented in the existing literature. This paper aims to fill these gaps with the following primary contributions:

- Providing a comprehensive review of recent advancements in system identification methods for perturbed motorized systems, including data acquisition processes and setups, with a focus on brushed DC motors, brushless DC motors (BLDC), and PMSMs.
- Systematically evaluating both conventional and metaheuristic algorithms as parameter optimization strategies, analyzing their strengths, limitations, and practical applications in system modeling.
- Investigating emerging trends in model structures and optimization techniques, specifically in the context of perturbations such as varying loads, noise, and friction, to understand their impact on system identification accuracy.
- Introducing a comparative analysis of grey-box and black-box system identification methods, including grey-box modeling using metaheuristic algorithms (PSO, GWO, and GA) and black-box modeling with neural networks, for a perturbed DC motor under both open-loop and closed-loop conditions. This study uniquely evaluates their generalization ability and robustness in handling system perturbations.
- Identifying key research gaps and proposing future directions to enhance system identification methodologies, including the integration of advanced machine learning techniques and adaptive optimization strategies for improved performance in real-world applications.

The remainder of the paper is structured as follows: [Section 2](#) describes an organized research framework that help to ensure the transparency of systematic review reporting. [Section 3](#) provides a comprehensive examination of motor mathematical models. [Section 4](#) delves into the synthesized findings from the preceding systematic review of the literature concerning system identification techniques for motorized systems under various perturbations. [Section 5](#) reviews various parameter optimization methods in optimizing motor model parameters and performance. [Section 6](#) provides a systematic analysis and discussion of the literature reviewed in [Sections 4](#) and [5](#). [Section 7](#) presents a comparative performance analysis of several widely used optimization methods for system identification, applied to a perturbed DC motor

under both open-loop and closed-loop settings. Section 8 outlines potential directions for future research. Finally, Section 9 concludes the study by summarizing the key findings and highlighting their significance.

2 Methodology

In this section, an organized research framework was developed. Fig. 1 shows the flow diagram of preferred reporting items for systematic reviews and meta-analyses PRISMA 2020 as a graphical representation of the study screening and selection process. The flow diagram can be expressed in three major steps, namely, identification, screening, and included.

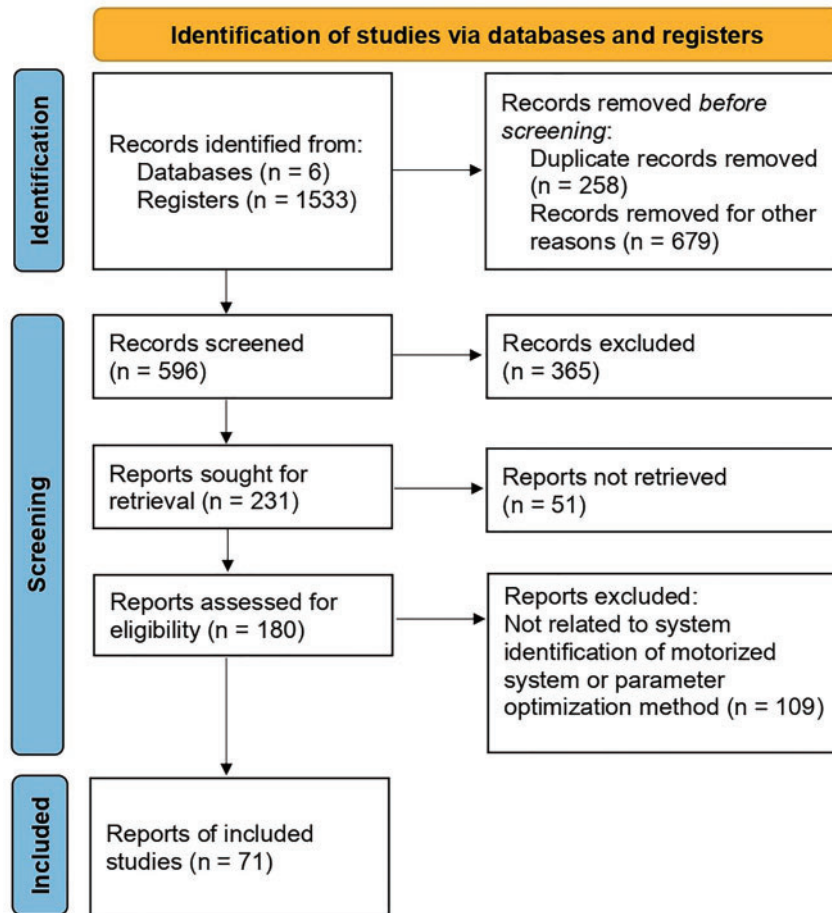


Figure 1: PRISMA flow diagram

In the identification phase, research area and goal were first defined. The research areas are the system identification of perturbed motorized systems and parameter optimization methods. The research goal is to provide a systematic research scheme for existing modeling and optimization methods. The articles were gathered as references by using key terms, such as system identification, motor modeling, motor parameter optimization, LS parameter estimation, motor identification using PSO, etc. Online journal database used for searching research articles based on key terms as listed as Web of Science, Scopus, IEEE Xplore, ScienceDirect, Taylor & Francis, and Springer.

In screening phase, the articles are screened for eligibility and relevancy with the research area and goal. Inclusion and exclusion criteria are specified to ease the screening phase. Inclusion criteria such as relevance to system identification, motor modeling, and parameter optimization. Exclusion criteria such as low citation, irrelevant to motorized systems, reports before 2010, etc. In third phase which is the included phase, the remaining articles are selected for review and analysis.

3 Motor Modeling

This section describes modeling methods for brushed DC motors, BLDC, and PMSM based on first principles, which is also known as white-box modeling. The models of these motors, which detail their structural and functional aspects, are in some cases crucial for applying both grey-box and black-box modeling techniques. Grey-box modeling strikes a balance between theoretical physical models and empirical data, allowing for predictions that account for known behaviors and measured data simultaneously. This approach is particularly useful when some system parameters are unknown or difficult to measure directly but where some theoretical knowledge about the system's behavior is available. The black-box modeling, on the other hand, which relies solely on input-output data without any presumption about the system's internal workings, benefits from the detailed model structures to better interpret the data patterns and relationships that emerge during the modeling process.

3.1 Mathematical Model of Brushed Motor

A DC motor is a common actuator utilized in control systems, primarily for generating rotational motion. Its key components comprise the housing and bearings, the stator (which may consist of a magnet, permanent magnet, or electromagnet), the rotor, the commutator, and the brushes. Upon the application of voltage to the motor terminals, current flows through the armature windings, establishing an electromagnetic field. This field interacts with the magnetic field generated by either stationary magnets or field coils, generating a force that initiates armature rotation. As the armature turns, the commutator segments make contact with the brushes, thereby reversing the current direction in the windings to sustain continuous rotation. The mathematical modeling of a DC motor circuit integrates both electrical and mechanical aspects, providing a fundamental tool for control system design. Accurate determination of model parameters is crucial as they significantly influence the motor's behavior. Fig. 2 illustrates a schematic diagram of a DC motor on no load condition [32].

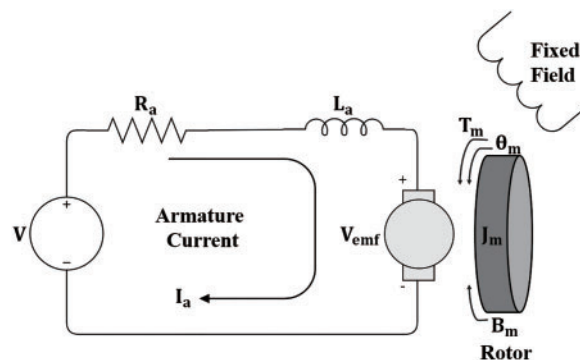


Figure 2: Brushed DC motor schematic diagram for no load condition

In general, a DC motor's torque is directly correlated with its armature current and magnetic field intensity. Assuming a constant magnetic field, the motor's torque, τ_m is proportional to the armature current,

I_a represented by motor torque constant, K_t as follows:

$$\tau_m = K_t I_a. \quad (1)$$

The back electromotive force (EMF), V_{emf} is proportional to the shaft's angular velocity, $\dot{\theta}_m$ by a EMF constant, K_e expressed as:

$$V_{emf} = K_e \dot{\theta}_m. \quad (2)$$

From Fig. 2, equations can be formulated using Newton's Second Law for the mechanical aspect and Kirchhoff's Voltage Law for the electrical aspect as follows:

$$\tau_m - B_m \dot{\theta}_m = J_m \ddot{\theta}_m, \quad (3)$$

$$V - I_a R_a - L_a \frac{di}{dt} - V_{emf} = 0, \quad (4)$$

where B_m is the motor viscous friction constant, J_m is the moment inertia of the rotor, $\ddot{\theta}_m$ is shaft's angular acceleration, R_a is the armature resistance, L_a is the inductance, V is the voltage source, and V_{emf} is the back EMF voltage. By substitution of (1) into (3) and (2) into (4) and applying Laplace transform, the following equations can be derived:

$$s (J_m s + B_m) \theta_m (s) = K_t I_a (s), \quad (5)$$

$$(L_a s + R_a) I_a (s) = V (s) - K_e s \theta_m (s). \quad (6)$$

The armature voltage is regarded as the input and the motor shaft's position as the output, the transfer function (TF) of the DC motor position, θ_m with respect to the voltage source at no load condition can then be written as:

$$\frac{\theta_m (s)}{V (s)} = \frac{K_t}{s [(J_m s + B_m) (L_a s + R_a) + K_t K_e]}. \quad (7)$$

At no load condition, the armature current is zero, and the angular speed will try to reach infinite speed. For load condition as depicted in Fig. 3, an additional load mechanical equation is added for calculation by assuming $\theta_L > \theta_m$ as shown in (8) and (9):

$$J_L s^2 \theta_L (s) + K_L \theta_L (s) - K_L \theta_m (s) = 0, \quad (8)$$

$$J_m s^2 \theta_m (s) + B_m s \theta_m (s) + K_L \theta_m (s) - K_L \theta_L (s) - T_m (s) = 0, \quad (9)$$

where J_L is the moment inertia of the load, K_L is the load constant, and θ_L is the corresponding motor load shaft's position. From (6):

$$I_a (s) = \frac{V (s) - K_e s \theta_m (s)}{L_a s + R_a}. \quad (10)$$

By substituting (10) into (1), (11) is obtained, then followed by substituting (11) into (9), (12) is obtained:

$$\tau_m = K_t \left(\frac{V (s) - K_e s \theta_m (s)}{L_a s + R_a} \right) \quad (11)$$

$$J_m s^2 \theta_m(s) + B_m s \theta_m(s) + K_L \theta_m(s) - K_L \theta_L(s) - K_t \left[\frac{V(s) - K_e s \theta_m}{L_a s + R_a} \right] = 0. \quad (12)$$

DC motor position with respect to the voltage source at load condition can then be expressed as:

$$\frac{\theta_L(s)}{V(s)} = \frac{K_t}{\left(\frac{J_m s^2 + K_L}{K_L} \right) \left(J_m s^2 + B_m s + K_L + \frac{K_t K_e}{s L_a + R_a} \right) - K_L}. \quad (13)$$

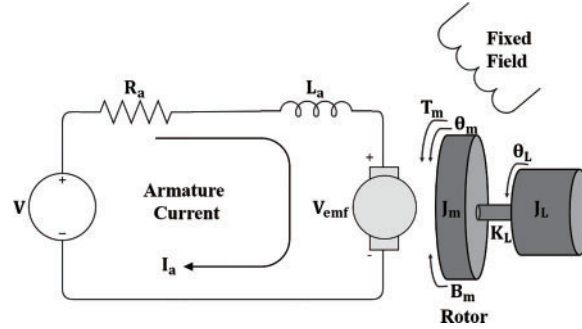


Figure 3: Brushed DC motor schematic diagram for load condition

3.2 Mathematical Model of Brushless DC Motors (BLDC)

Fig. 4 depicts a schematic diagram of a BLDC motor [33]. While both brushed DC and BLDC motors adhere to similar principles, BLDC motors operate with more precision. Leveraging rotor position feedback from Hall sensors, the motor controller of a BLDC motor determines the optimal timing and sequence of current pulses to the stator coils. By energizing the coils in the stator in the correct sequence, the motor controller generates a rotating magnetic field. This field interacts with the fixed magnetic field of the rotor, resulting in torque generation and motor rotation. The circuit equation of the three windings in phase variables is given by:

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L_{aa} & L_{ab} & L_{ac} \\ L_{ba} & L_{bb} & L_{bc} \\ L_{ca} & L_{cb} & L_{cc} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix}. \quad (14)$$

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L & M & M \\ M & L & M \\ M & M & L \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix}. \quad (15)$$

where R is the stator resistance, V_a , V_b and V_c are stator phase voltages, i_a , i_b and i_c are stator phase currents, e_a , e_b and e_c are stator induced EMF, $L_{aa} = L_{bb} = L_{cc} = L$ is the stator inductance, $L_{ab} = L_{ac} = L_{ba} = L_{bc} = L_{ca} = L_{cb} = M$ is mutual inductance between phase a, b and c. The stator phase currents are constraints to be balanced:

$$i_a + i_b + i_c = 0. \quad (16)$$

All windings are assumed to have equal resistance, and the rotor reluctance is assumed to remain constant with angle. The simplification of stator phase voltages, V_a , V_b and V_c in matrix form can be

expressed as:

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & R \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} L-M & 0 & 0 \\ 0 & L-M & 0 \\ 0 & 0 & L-M \end{bmatrix} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix}. \quad (17)$$

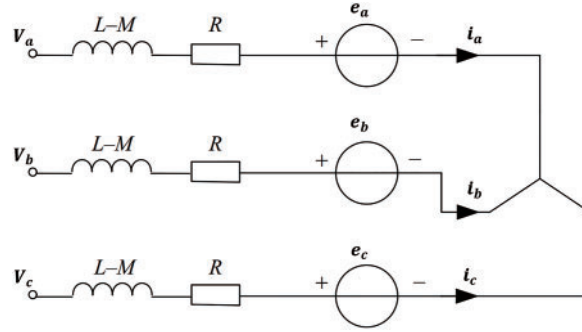


Figure 4: BLDC motor schematic diagram

The induced EMF equation of the BLDC motor is all assumed to be trapezoidal and can be written as:

$$\begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} = \omega_m \lambda_m \begin{bmatrix} f_{as}(\theta_r) \\ f_{bs}(\theta_r) \\ f_{cs}(\theta_r) \end{bmatrix}. \quad (18)$$

The electromagnetic torque, T_e defined as:

$$T_e = \frac{e_a i_a + e_b i_b + e_c i_c}{\omega_m}, \quad (19)$$

where ω_m is motor angular velocity, λ_m is flux linkage, $f_{as}(\theta_r)$, $f_{bs}(\theta_r)$, $f_{cs}(\theta_r)$ are unit function generator depending on rotor angular position with maximum magnitude of ± 1 . The mechanical part of BLDC motor accounts for motor's mechanical movement can be expressed as:

$$T_e - T_L = J \frac{d\omega_m}{dt} + B\omega_m \quad (20)$$

where J is moment of inertia, B is frictional coefficient, T_L is load torque. The moment of inertia, J is described as:

$$J = J_m + J_L, \quad (21)$$

where J_m is motor inertia and J_L is load inertia. The derivative of the electrical rotor position, θ_r are related by:

$$\frac{d\theta_r}{dt} = \frac{p}{2} \omega_m, \quad (22)$$

where p is number of pole pairs. The system in the state space form is:

$$\dot{x} = Ax + Bu, \quad (23)$$

where $x = [i_a \ i_b \ i_c \ \omega_m \ \theta_r]'$, $u = [V_a \ V_b \ V_c \ T_L]'$. Let $P = -\frac{R}{L-M}$, $T = \frac{1}{L-M}$, $Q = \frac{\lambda_m}{J} f_{as}(\theta_r)$, $U = \frac{\lambda_m}{J} f_{bs}(\theta_r)$, $S = \frac{\lambda_m}{J} f_{cs}(\theta_r)$. A and B can be written as:

$$A = \begin{bmatrix} P & 0 & 0 & -Q & 0 \\ 0 & P & 0 & -U & 0 \\ 0 & 0 & P & -S & 0 \\ Q & U & S & -\frac{B}{J} & 0 \\ 0 & 0 & 0 & \frac{P}{2} & 0 \end{bmatrix}; \quad B = \begin{bmatrix} T & 0 & 0 & 0 & 0 \\ 0 & T & 0 & 0 & 0 \\ 0 & 0 & T & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{J} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

3.3 Mathematical Model of Permanent Magnet Synchronous Motor (PMSM)

The PMSM is a brushless machine with sinusoidal flux distribution and has very high reliability and efficiency. It is also known as a brushless AC machine. The dynamic model of PMSM was first derived using equivalent two-phase motor system formulated in direct (d) and quadrature (q) axes. Fig. 5 shows the equivalent circuit at d- and q-axes PMSM model in rotor reference frame is obtained as [34]:

$$\begin{bmatrix} V_{qs}^r \\ V_{ds}^r \end{bmatrix} = \begin{bmatrix} R_s + L_q \rho & \omega_r L_d \\ -\omega_r L_q & R_s + L_d \rho \end{bmatrix} \begin{bmatrix} i_{qs}^r \\ i_{ds}^r \end{bmatrix} + \begin{bmatrix} \omega_r \lambda_{af} \\ 0 \end{bmatrix}, \quad (24)$$

where the saliency ratio is $\rho = \frac{L_q}{L_d}$, V_{qs}^r and V_{ds}^r are stator voltage in q- and d-axes windings, R_s is stator resistance, i_{qs}^r and i_{ds}^r are q- and d-axes stator currents, ω_r is the rotor speed in electrical radians per second, λ_{af} is rotor magnet flux. In PMSM, $L_q > L_d$, L_d is minimum inductance, L_q is maximum inductance, L_d and L_q are the d- and q-axes inductances, respectively.

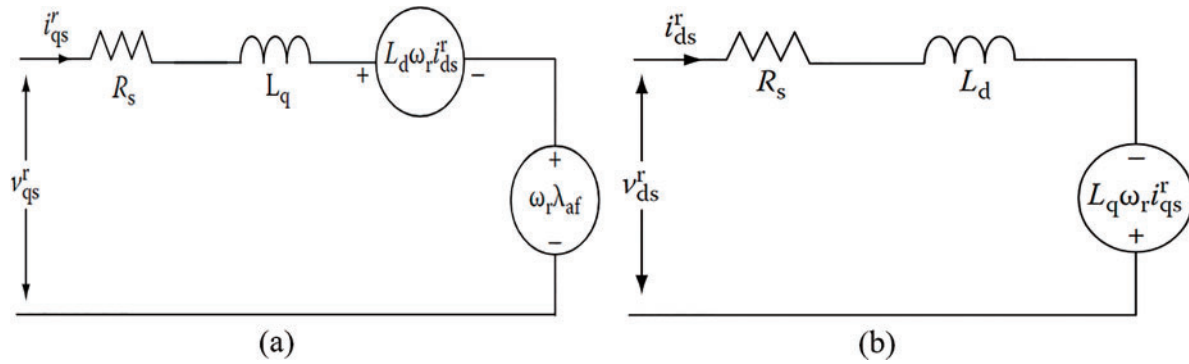


Figure 5: Equivalent circuit of PMSM: (a) q-axis (b) d-axis

PMSMs with three-phase are prevalent but two-phase machines are rarely used in industrial applications. A dynamic model for three-phase PMSM can be derived from two-phase machine. Assume that each of the stator three-phase windings has equal current magnitude and T_1 turns per phase, the equivalent two-phase winding will have $\frac{3}{2} T_1$ turns per phase. The relationship between qdo and abc current is:

$$\begin{bmatrix} i_{qs}^r \\ i_{ds}^r \\ i_o \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos \theta_r & \cos(x) & \cos(y) \\ \sin \theta_r & \sin(x) & \sin(y) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix}, \quad (25)$$

where θ_r is instantaneous rotor position, $x = \theta_r - \frac{2\pi}{3}$, $y = \theta_r + \frac{2\pi}{3}$, i_{as} , i_{bs} , i_{cs} are phase currents, i_o represents the imbalance in the a, b, c phase currents. The stator voltage equations in terms of stator currents and flux linkage in rotor reference frame can be expressed as:

$$V_{ds}^r = R_s i_{ds}^r - \omega_r \lambda_{qs}^r + L_d \frac{di_{ds}^r}{dt} = R_s i_{ds}^r + L_d \frac{di_{ds}^r}{dt} - \omega_r L_q i_{qs}^r + \frac{d}{dt} \lambda_{af}. \quad (26)$$

$$V_{qs}^r = R_s i_{qs}^r + \omega_r \lambda_{ds}^r + L_q \frac{di_{qs}^r}{dt} = R_s i_{qs}^r + L_q \frac{di_{qs}^r}{dt} + \omega_r L_d i_{ds}^r + \omega_r \lambda_{af}. \quad (27)$$

where $\lambda_{ds}^r = L_d i_{ds}^r + \lambda_{af}$ and $\lambda_{qs}^r = L_q i_{qs}^r$ are the d, q axis stator flux linkages in rotor reference frame. Rotor electrical speed, ω_r can be written as:

$$\omega_r = \left(\frac{p}{2}\right) \omega_m. \quad (28)$$

where p is the number of pole pairs, ω_m is rotor mechanical speed. Electromagnetic torque, T_e can be expressed as:

$$T_e = \left(\frac{3}{2}\right) \left(\frac{P}{2}\right) (\lambda_{ds}^r i_{qs}^r - \lambda_{qs}^r i_{ds}^r) = \left(\frac{3}{2}\right) \left(\frac{P}{2}\right) (\lambda_{af} i_{qs}^r + (L_d - L_q) i_{ds}^r i_{qs}^r). \quad (29)$$

4 System Identification

In the context of a perturbed motorized system, where factors like friction, varying loads, and other disturbances introduce complexities that challenge traditional first principles modeling, system identification emerges as a powerful solution. By leveraging observed input-output data, system identification enables the creation of accurate models that capture the dynamic behavior of the system amidst perturbations. These models not only adapt to changing conditions but also provide insights into the underlying dynamics, facilitating the design of robust control strategies tailored to mitigate the effects of disturbances.

The diagram in Fig. 6 outlines a standard procedure in system identification, starting from “Experiment Design” which is informed by “Prior Knowledge” [35]. Here, data is gathered and used to “Choose Model Set” based on a selected “Criterion of Fit.” This stage ensures the model aligns well with the collected data and system dynamics. The motor model structures based on first principles as described in the previous section can usually serve as hypotheses or initial guesses on the model structure. Next, the “Calculate Model” step involves applying the chosen model to predict or simulate system behavior. If the model predictions do not align with the expectations or further data (“Not OK”), the process includes a “Revise” loop, indicating iterative refinement of the model. Upon successful validation (“Validate Model”), where the model sufficiently represents the system under study, it is deemed acceptable for use (“OK: Use it”). This structured approach emphasizes iterative refinement and validation, leveraging both empirical data and theoretical knowledge to develop an accurate model of the system.

In [36], the study employs system identification to model a perturbed BLDC motor to achieve precise speed control. It introduces an ideal back EMF for enhanced precision in speed control in the face of varying load torques ranging from 0 to 20 Nm. The work in [37] utilizes system identification to design a proportional-integral-derivative (PID) control with auto-tuning. In [38], the system identification method is used to estimate both static and dynamic characteristics of a DC motor using starting rheostat as the starting current limiter.

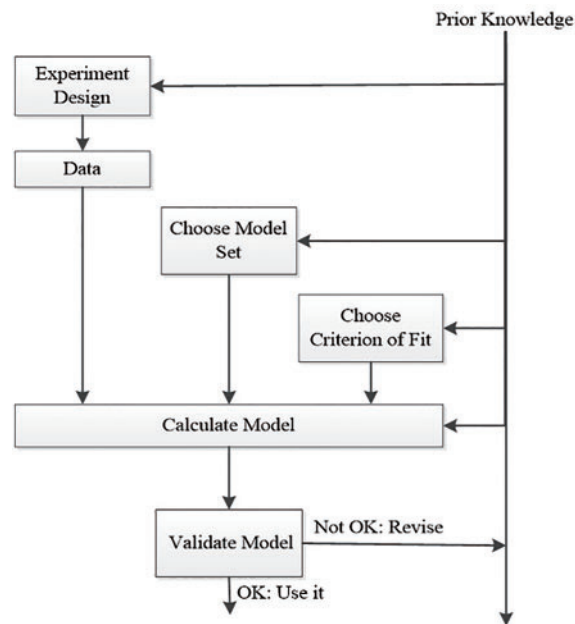


Figure 6: System identification's standard procedure [35]

A recent comparative study in [39] demonstrates that AutoRegressive with eXogenous input (ARX) and second-order continuous-time TF models offer a more accurate representation of the mathematical model of a DC motor under varying payloads. Additionally, in a separate study [40], it was demonstrated that the Autoregressive Moving Average Exogenous (ARMAX) structure provides a superior representation of the mathematical model of a DC servo motor compared to the ARX structure. In [41], a subspace-aided closed-loop system identification method is introduced for a DC motor system which demonstrates superior performance in pole estimation compared to a class of subspace identification methods utilizing principal component analysis algorithms. The study in [42] employs system identification to establish a low-order TF model of a physiological system (PSys) using a Multi-Input Multi-Output (MIMO) ARMAX model. Fig. 7 illustrates the flow chart of the proposed method where the model obtained from the system identification is then used to optimize the control parameters.

Batool et al. [43] discussed three estimation techniques for multi-domain DC servo motor model parameters: the Compare Coefficient Method, MATLAB Parameter Estimation Toolbox, and System Identification Toolbox. Experimental data is used to compare the accuracy of these methods, with the parameter estimation technique proving to be the most accurate and computationally efficient, guiding researchers in preference for DC servo motor simulation model parameter estimation. The proposed method in [44] optimizes accuracy and reliability of permanent magnet DC (PMD) motor models while considering dynamic characteristics and operational conditions. Eight parameters were identified, such as brush current constant, back EMF constant, motor resistance, brush voltage current, motor inertia, viscous friction, motor inductance, and stiffness constant.

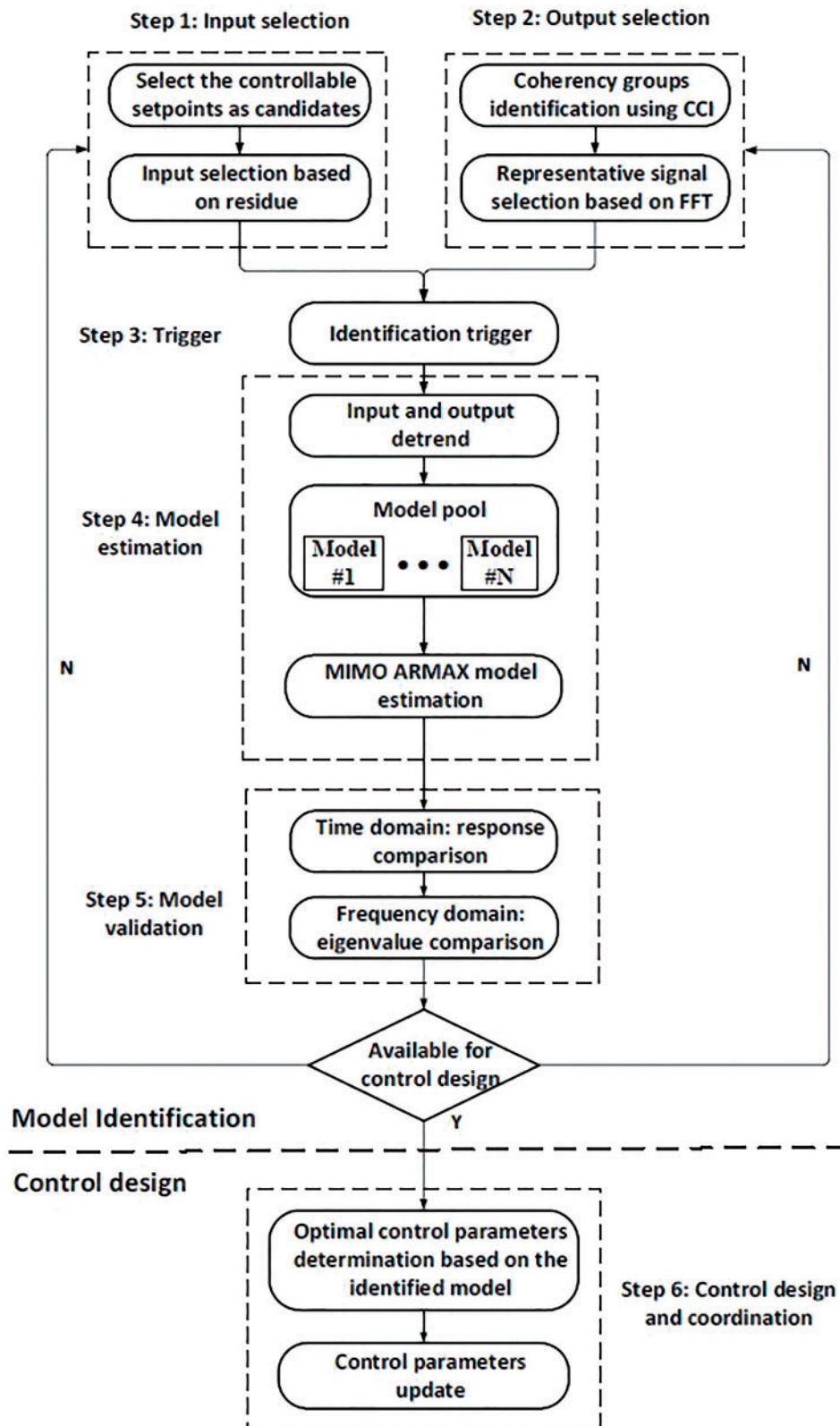


Figure 7: Methodology of model identification to build MIMO ARMAX model using measurement data as proposed in [42]

The study in [45] utilized system identification to estimate load torque and moment of inertia in PMSM drives based on sliding mode observer (SMO), which offered precise estimation and quick convergence. In [46], various estimators for PMSM electrical parameter identification were compared where the finding revealed that the moving horizon estimation outperformed the rest in accuracy. The Kalman filter (KF) of extended KF (EKF) and unscented KF (UKF) performed similarly, while the model reference adaptive system (MRAS) was sensitive to disturbances and required extensive tuning recursive LS yielded inaccurate results.

The method proposed in [47] utilizes motor impedance characteristics for parameter estimation, suitable for both offline and online environments. It achieves high accuracy even at low or zero current and speed, minimizing the influence of identification errors on different parameters. A KF-based approach discussed in [48] enables continuous updates of motor parameters, effectively handling noisy measurements for improved accuracy. EKF and dual EKF (DEKF) are compared, with DEKF performing better in estimation quality and resilience to initial states. In [49], a deadbeat control is introduced for tracking reference currents with zero steady-state error, adapting PMSM parameters during control. It effectively eliminates control offsets caused by parameter errors. The study in [50] enhances control robustness, minimizes torque ripples, and improves disturbance rejection, mainly integrating deadbeat control while addressing torque oscillations and noise reduction.

The system identification method is employed in [51] to model the inertia of a PMSM drive. It uses a sinusoidal perturbation torque with a DC offset to drive the PMSM, and then measures the amplitudes of the torque and speed response to calculate the system's inertia. This method is robust and provides accurate inertia values quickly compared to other approaches. The studies in [52–56] utilized a black-box modeling approach for unmanned aerial vehicles (UAV), PSys stabilizer, flexible maneuvering systems (FMS), ground robots, and fuel control unit, respectively. This approach allows for adaptability to system changes without requiring extensive recalibration.

5 Parameter Optimization

In system identification, after selecting or obtaining the model structure, the subsequent step involves parameter optimization. This section explores diverse parameter optimization methods utilized in the literature, which have shown promising results. In optimization algorithms, the error-or more specifically, the fitness value, a standard term in metaheuristic algorithms-serves as a quantitative measure of how well the identified model represents the actual system dynamics. Commonly used metrics include Sum of Squared Errors (SSE), Mean Squared Error (MSE), Integral of Absolute Errors (IAE), and Integral of Time-weighted Absolute Error (ITAE). These metrics are calculated by comparing the observed outputs (y_{obs}) to the predicted outputs (y_{pred}) of the model. For instance, SSE is expressed as $\sum_{i=1}^n (y_{\text{obs},i} - y_{\text{pred},i})^2$, MSE as $\frac{1}{n} \sum_{i=1}^n (y_{\text{obs},i} - y_{\text{pred},i})^2$, IAE as $\sum_{i=1}^n |y_{\text{obs},i} - y_{\text{pred},i}|$, and ITAE as $n \sum_{i=1}^n |y_{\text{obs},i} - y_{\text{pred},i}|$. These error metrics guide the optimization process, ensuring accurate parameter estimation and reliable system representation by minimizing the discrepancies between observed and predicted outputs.

5.1 Problem Formulation

Fig. 8 illustrates the general framework for parameter optimization. The process of tuning model coefficients using metaheuristic optimization algorithms begins with preprocess data, followed by the identification of the model structure and the selection of parameters to be optimized. In the context of electric motor systems, key parameters such as armature resistance (R_a), armature inductance (L_a), moment of inertia (J), viscous friction coefficient (B), torque constant (K_t), and back electromotive force constant (K_e) are commonly adjusted to achieve optimal system performance. Some techniques directly utilize the transfer

function coefficients for modeling and estimate parameters using optimization algorithms within predefined boundaries. Therefore, the order of the transfer function must be determined prior to the optimization phase.

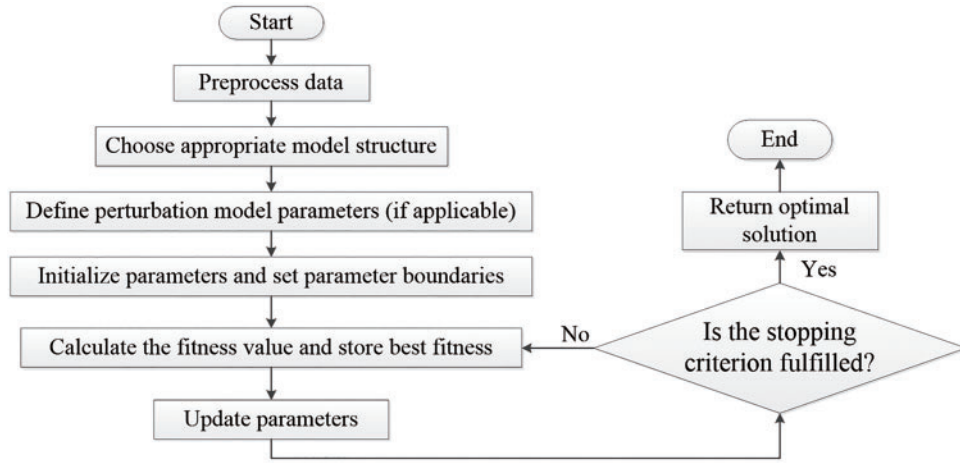


Figure 8: Generic parameter optimization procedure

The optimization problem is formulated by defining decision variables, selecting an appropriate objective function, and imposing relevant constraints based on system dynamics and performance requirements. The choice of the objective function significantly influences the optimization outcome, with commonly used objective functions including the IAE, ITAE, MSE and SSE. The optimization algorithm is then initialized by generating a population of candidate solutions in population-based methods. Each candidate solution is evaluated through model simulation, where the objective function quantifies system performance based on predefined criterion. The algorithm iteratively refines the solutions by leveraging exploration mechanisms to enhance global search capabilities and exploitation mechanisms to fine-tune local optima. Convergence is assessed based on predefined stopping criterion, such as a maximum number of iterations or a threshold for minimal improvement in the objective function. Once an optimal solution is identified, its validity can be examined through several methods such as step response analysis, frequency response evaluation, and robustness testing against parametric variations and external disturbances. Finally, the optimized parameters are implemented, and key performance metrics are analyzed to ensure a comprehensive and unbiased comparison across different optimization algorithms.

5.2 Optimization Methods

5.2.1 Least Squares (LS)

The LS method is commonly used for parameter optimization by minimizing the SSE between the observed output data and the model predictions [57]. This involves formulating an objective function that represents the difference between the actual output and the model's output. LS is suitable for linear problems, whereas LM is more effective at handling nonlinearity. In [58], the MATLAB System Identification Toolbox, which uses LS as the default search algorithm, is employed to identify a DC motor model. The resulting error between the real and estimated data for transfer function prediction is less than 2%. Let \mathbf{y} be the vector of observed motor response, \mathbf{X} be the matrix of input data, and θ be the vector of unknown parameters to be estimated. The LS method aims to find the parameter vector $\hat{\theta}$ that minimizes the objective function:

$$J(\theta) = \|\mathbf{y} - \mathbf{X}\theta\|^2. \quad (30)$$

This objective function is typically minimized by solving the following equation:

$$\hat{\theta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}. \quad (31)$$

Once $\hat{\theta}$ is obtained, it represents the estimated parameters of the motor model.

Fig. 9 demonstrates the working principle of LS. It begins by initializing the parameters to be optimized, then transforms the equation into a linear form to simplify the solution process. The next step is to check whether the adjustment value is below a specified threshold and if the stopping criteria are met to obtain the parameter solution. The LS approach is particularly useful when dealing with linear models or when the objective function can be linearized, as it provides a computationally efficient and statistically robust way to estimate model parameters. In [59], the LS method is utilized to derive both static and dynamic models for DC motors. Fig. 10 depicts the experimental setup proposed in [59] for estimating motor parameters, implemented using Arduino and Python. The results indicate that motor models adopting the ARX structure achieve greater accuracy when optimized using the LS method compared to pattern search (PS) and nonlinear LS (NLS) methods, particularly in terms of angular velocity responses.

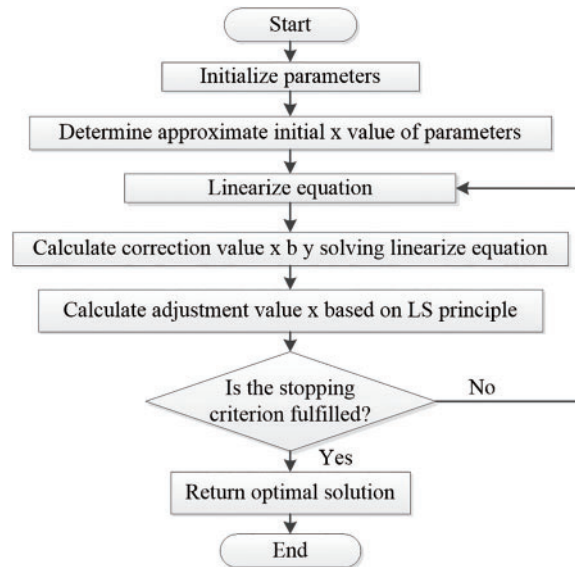


Figure 9: LS flowchart

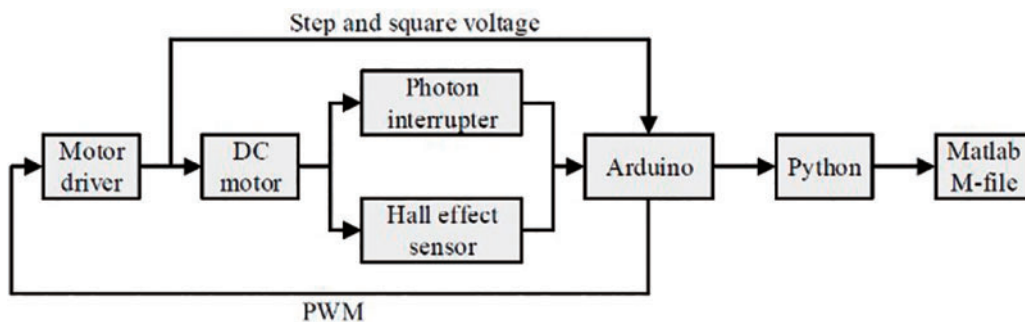


Figure 10: Experiment setup for system identification in [59]

For perturbed BLDC motors, the LS method can be employed to estimate cogging torque as it effectively approximates the true nonlinear dynamics of the systems. Fig. 11 illustrates the approach introduced in [60] for cogging torque estimation which leverages an ideal model derived from the linear mechanical dynamics of the BLDC motor without any external load torque. In the work, a combination of frequency domain identification of linear terms and time domain estimation method is used to estimate unknown parameters. The study in [61] employs a DC motor with a minimum-phase mathematical model controlled by a self-tuning regulator without model pole cancellation. The standard system identification process is then detailed by parameterizing the system for LS estimation, incorporating exponential forgetting to address time-varying plants.

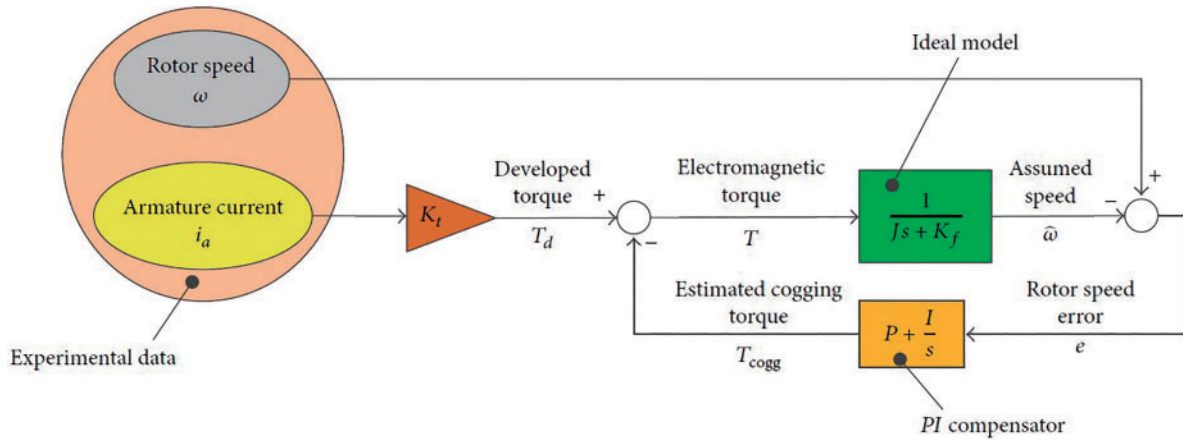


Figure 11: Cogging torque estimation technique using LS for nonlinear BLDC modeling [60]

In perturbed motor modeling, various LS variants have been proposed in the literature to either lessen computational burden or tailor it to specific applications. One such method is the recursive LS (RLS) approach, which updates parameter estimates recursively as new data is acquired in order to enable efficient adaptation to changing system dynamics and time-varying parameters [62]. Thus, instead of constructing \mathbf{X} to solve for $\hat{\theta}$ in (30)–(31), one can simply compute $\hat{\theta}$ as follows:

$$\hat{\theta}(n) = \hat{\theta}(n-1) + \mathbf{P}(n)\mathbf{x}(n) \left(y(n) - \mathbf{x}(n)^T \hat{\theta}(n-1) \right), \quad (32)$$

where $\mathbf{P}(n)$ is the covariance matrix, $\mathbf{x}(n)$ is the input vector at time n , y is the observed output at time n , and $\hat{\theta}(n-1)$ is the estimated parameter vector from the previous step. $\mathbf{P}(n)$ is then updated as follows:

$$\mathbf{P}(n) = \mathbf{P}(n-1) - \frac{\mathbf{P}(n-1)\mathbf{x}(n)\mathbf{x}(n)^T\mathbf{P}(n-1)}{1 + \mathbf{x}(n)^T\mathbf{P}(n-1)\mathbf{x}(n)}. \quad (33)$$

This method generally demands fewer computational resources than batch LS estimation, rendering it well-suited for online and real-time applications [63]. The work in [64] for instance, utilizes the RLS method for online identification of the dynamic model of a differential drive mobile robot, while employing the LM method for offline identification. A similar approach is adopted in [65] for PMSM with sensorless control technique. LM is an optimization technique which is particularly effective when dealing with NLS problems. It can handle situations in which the underlying relationship between variables is nonlinear.

In [66], the RLS is employed to estimate the disturbance voltage in voltage source inverter operation in a PMSM system. It proposes two methods to reduce this estimation error; one uses an iterative learning

controller to suppress d-axis current harmonics, and the other bases the disturbance voltage estimation on the average value of the q-axis equation variables, making it immune to current harmonics and enhancing motor parameter identification accuracy. Such a method is illustrated in Fig. 12. The work in [67] presents a methodology combining finite-control-set model predictive control with online LS system identification, which can maintain optimal performance despite changes in motor dynamics or external disturbances. This approach enhances the control performance of PMSM drives by integrating predictive control with real-time system identification. It is capable of handling constraints on control inputs and system states, making it suitable for applications with strict operational requirements or safety constraints.

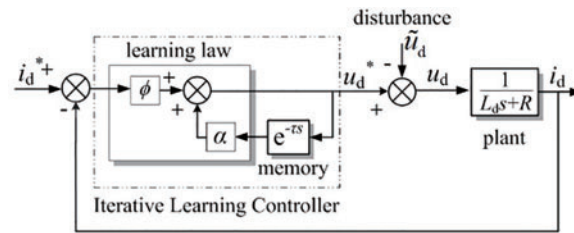


Figure 12: Iterative learning controller in the d-axis current control loop for a perturbed PMSM system [66]

In [68], the RLS method is utilized for a PMDC motor with an ARX model structure. Subsequently, an adaptive discrete pole placement controller is proposed and designed to regulate the motor's rotational speed. Another LS variant, termed sequential LS, is proposed in [69] which is specifically tailored for ARMAX dynamic network identification. This approach offers advantages in terms of accuracy, convergence speed, robustness, and improved approximation by introducing new criteria to refine the parameter estimates. The study in [70] introduced ordinary LS (OLS) for BLDC motor modeling, but it has limited ability to handle complex or noisy data and not suitable for systems with highly dynamic behavior or significant uncertainties. Palpacelli et al. [71] employed OLS for the identification of industrial robots. This method utilizes motor current measurements obtained during numerous slow-motion tests. This study takes into consideration the Coulomb friction and gravity contributions to motor torques. The model incorporates and analyzes the effect of external pressures applied at the end-effector. A combination of RLS and OLS has been introduced in [72] to improve the system identification of a DC motor while minimizing computational burden. This approach is suitable for relatively well-behaved systems with low noise levels.

The study in [73] proposed a combination of EKF and LS to optimize parameters of both kinematic and dynamic models for a tracked mobile robot. The findings illustrate the effectiveness and reliability of the proposed approach in identifying terrain parameters, which could prove valuable for future path planning efforts. Li and Ma [74] proposed a compound LS (CLS) for DC motor parameter identification, aiming to address the volatility of fault-prone results in forgetting factor LS (FFLS) identification. By integrating selection control, this method offers faster, more accurate, and stable real-time identification of DC motor parameters, even as these parameters change. Simulation results demonstrate that the hybrid least square parameter identifier outperforms the FFLS method by offering faster identification speed, higher accuracy, better stability, and effective online identification of motor parameters across different speeds.

5.2.2 Particle Swarm Optimization (PSO)

PSO is modeled after the social behavior of schools of fish or flocks of birds, in which single particles cooperate to find the utmost solution. The PSO approach was first introduced by social psychologist James Kennedy and electrical engineer Eberhart and Kennedy [75]. In the domain of identification, PSO is used

to iteratively change the locations of a population of particles in a multidimensional search space to identify the ideal model parameters or structures [76]. Each particle in the PSO represents a possible solution, which could be a set of model parameters or even a DC motor model. Each particle keeps track of its position and speed inside the search space while assessing its fitness using a predetermined objective function. On the basis of their individual knowledge, which is also known as local best, and the aggregate wisdom of the entire swarm, which is global best, particles modify their positions and velocities [77,78]. The results possess improved accuracy in parameter estimation and enhanced model performance for dynamic systems.

Fig. 13 indicates the working principle of PSO, which optimizes solutions by simulating the movement and interactions of particles in a solution space. Each particle starts by calculating its fitness value based on an objective function. The algorithm then checks if each particle's current fitness is better than its best previous fitness, known as pBest. If it is, pBest is updated; if not, it remains the same. The best pBest among all particles is then assigned to the gBest, representing the best solution found so far. The particles' velocities are updated based on both their distance from pBest and gBest, guiding them towards potentially better solutions. Each particle then uses its updated velocity to move to a new position in the solution space. The algorithm repeats this process until the maximum number of iterations is reached or the solution stabilizes, ultimately yielding the best solution found by the swarm.

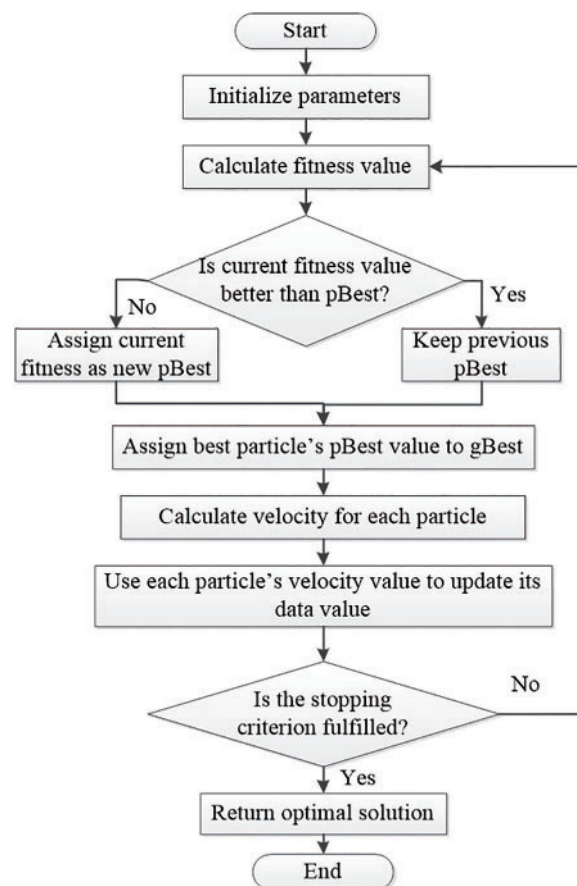


Figure 13: PSO flowchart

PSO can be mathematically described as:

$$v_i(k+1) = wv_i(k) + c_1r_1 \cdot (p_{\text{best},i} - x_i(k)) + c_2r_2 \cdot (g_{\text{best}} - x_i(k)) \quad (34)$$

$$x_i(k+1) = x_i(k) + v_i(k+1) \quad (35)$$

where $i = 1, 2, \dots, N$ represents the particle index, k denotes the iteration number, w is the inertia weight, c_1 and c_2 are positive constants, r_1 and r_2 are random parameters, $p_{\text{best},i}$ denotes the best position found by particle i , g_{best} represents the best position among all particles in the swarm, $x_i(k)$ and $v_i(k)$ denote the position and velocity of particle i at iteration k , respectively. The PSO algorithm begins with the initialization of particle positions and velocities. Then, at each iteration, particles update their velocities based on their previous velocities and the differences between their current positions and both their individual best positions ($p_{\text{best},i}$) and the global best position (g_{best}). Subsequently, particles update their positions based on their updated velocities. The process continues iteratively until a termination criterion is met.

In [79], a systematic evaluation of the influence of the swarm size parameter on the effectiveness of the PSO-based Nonlinear ARX (NARX) structure selection is proposed. By doing so, it offers insights into optimizing the swarm size for better model performance. The findings show that a swarm size of 20 to 30 was best for optimizing the DC motor structure selection because it required a minimum of iterations to converge and had high fitness values. A comparison study discussed in [80] indicates good fitting results using the orthogonal LS approach and PSO for the NARX parameter estimation of a DC motor. The plots of autocorrelation and cross-correlation showed minor correlations between the output and the residual as well as between the residual and itself.

The cognitive coefficient (c_1) and social coefficient (c_2) are critical parameters in PSO as they regulate the balance between exploration and exploitation in the search space. The parameter c_1 influences each particle's tendency to move toward its own best-known position (p_{Best}). A higher c_1 encourages particles to focus on exploring their personal best areas, thereby promoting individual exploration. Conversely, c_2 governs each particle's inclination to move toward the swarm's global best-known position (g_{Best}). A higher c_2 encourages particles to follow the group's best solution, facilitating collective learning. The cognitive and social coefficients used in PSO for system identification vary across studies.

Liu et al. [81] employed parallel computing on a graphics processing unit (GPU) to enhance the efficiency, speed of parameter identification, and temperature monitoring. Parameter estimation was carried out under a variation of temperature conditions. A heater is used to heat the prototype for twenty minutes before the identification experiment is carried out. The data indicates that the parallel co-evolutionary immune PSO accelerated by GPU (G-PCIPSO), performs better with respect to mean, standard deviations, and t-values. The confidence level of G-PCIPSO reached 98% compared to other hybrid PSOs. In terms of speed and computing efficiency, this method can perform noticeably better than conventional central processing unit based techniques, enabling real-time and increased accuracy.

Another study in [82] critically evaluates the use of PSO for optimizing control parameters to improve motor control performance. Utilization of back EMF detection for feedback can provide information about the motor's actual speed and position, contributing to more accurate and responsive control. Fig. 14 illustrates the resulting system responses of proportional-integral (PI) and PID controllers optimized via PSO against other optimization methods. From the findings, PSO consumed the smallest search time, providing the fastest response with the shortest rise and settling time. The optimized controller can be applied to various BLDC motor control scenarios, making it versatile and adaptable to different applications. Thus, it enhances the capability to deal with non-linear models and dynamic behavior with more accurate parameter identification.

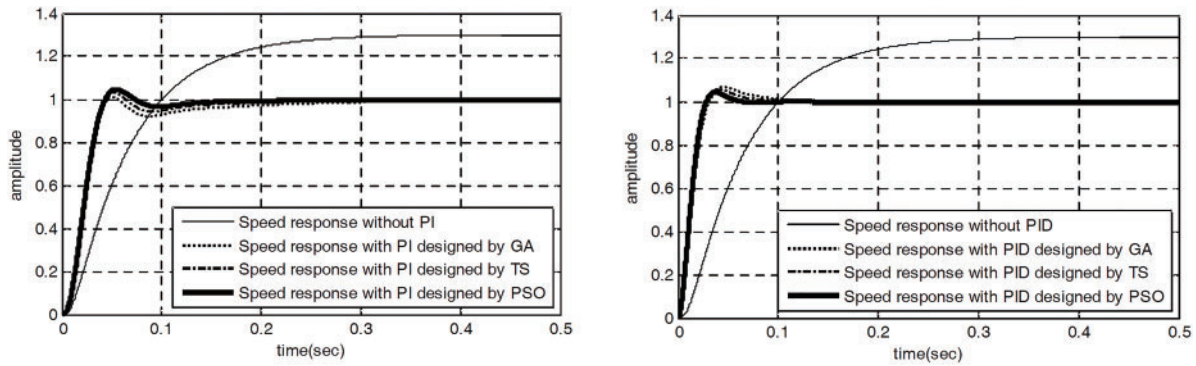


Figure 14: System response of PI (left) and PID (right) controllers with GA, TS and PSO [82]

In [83], the results suggest that PSO offers a unified and potentially more efficient method for accurately predicting the behavior of a DC motor, while the modified KF successfully reduces noise in measurement. This approach is particularly well-suited for nonlinear and dynamic systems. Meanwhile, Ref. [84] focuses on enhancing the precision and efficiency of parameter estimation, crucial for improving system control and performance. The study explores various PSO variants, including Constricted PSO (CI PSO), Standard PSO (STD PSO), Chaos Initialized PSO (CIW PSO), and Adaptive Inertia Weight Factor PSO (AIWF PSO), to identify the best-performing algorithm. CIW PSO emerges as the top performer, surpassing other algorithms by avoiding local optima, achieving close-to-actual values, and minimizing the cost function. With its collective intelligence and exploration-exploitation balance, PSO proves to be an effective method for identifying motor parameters.

Kanojiya and Meshram [85] proposed an automated tuning process that reduces manual parameter adjustment, introducing a hybrid topology PSO concept advantageous for solving optimal control problems due to its convergence, flexibility, and adaptability. Another study conducted by Abedinifar et al. [86] focused on nonlinear model identification and statistical verification of UR5 manipulator joint parameters using experimental data, resulting in improved accuracy and reliability in capturing dynamic behavior. The PSO algorithm facilitated parameter determination for nonlinear terms, achieving a high R^2 value of 97.1% and precise computation of joint torques, with a maximum identification error as low as 5%. Additionally, Sandre-Hernandez et al. [87] employed PSO to identify stator inductances of a PMSM using experimental measurements, demonstrating fast and stable convergence characteristics without complex control strategies, as depicted in Fig. 15. Experimental validation confirmed the effectiveness of the proposed method, showing feasibility with a commercial PMSM and real-time microcontroller.

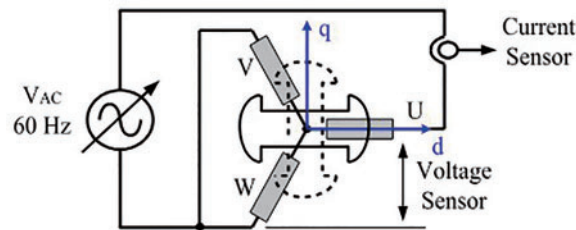


Figure 15: Connection scheme for identification of the stator inductances using PSO as proposed in [87]

The study in [88] aims to estimate the parameters of a PMDC motor for a wheelchair, utilizing standard PSO and dynamic PSO (DPSO), ACO, and ABC, alongside experimental methods. Parameters such as torque constant, back-EMF constant, moment of inertia, friction coefficient, armature inductance, and resistance are estimated and compared using both experimental and optimization approaches, with simulations showing that DPSO with varying inertia weight and ABC algorithm yield motor parameters with significantly lower speed and current errors compared to other methods. The work in [89] introduces the G-DPSO-RE method, which combines fast DPSO with a receptor editing strategy, leveraging GPU acceleration to establish an accurate parameter estimation model for surface PMSMs while considering voltage-source-inverter (VSI) nonlinearity. The method demonstrates effectiveness in estimating PMSM parameters and disturbance voltage from experimental data and analyzing the impact of VSI nonlinearity on estimation accuracy. In [90,91], the dynamic self-learning feature enhances the algorithm's adaptability to changes in operating conditions. Fig. 16 illustrates the schematic of the parameter estimation system as proposed in [91], which utilizes a prototype of a PMSM and a DSP vector control hardware platform as the experimental setup.

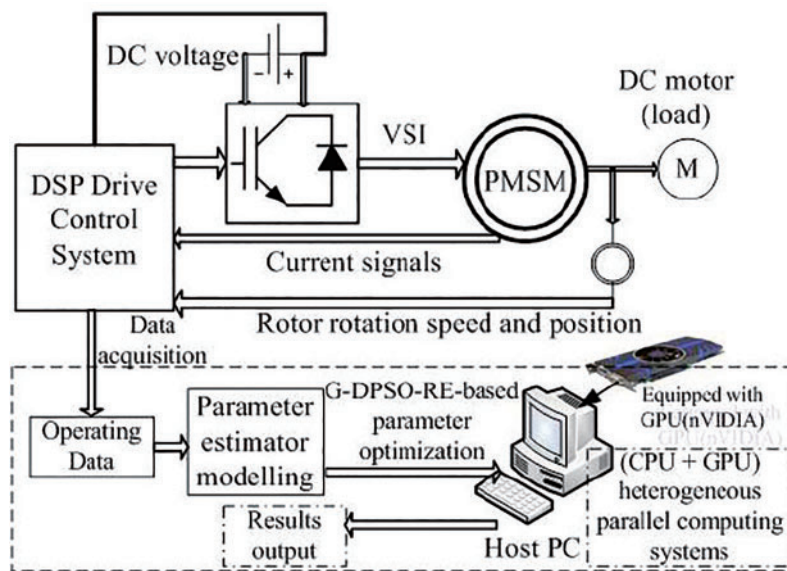


Figure 16: The schematic of the parameter estimation system as proposed in [91], which utilizes a prototype of a PMSM and a DSP vector control hardware platform as the experimental setup

5.2.3 Genetic Algorithm (GA)

GA, drawing inspiration from natural selection, is an influential optimization method widely applied in parameter identification tasks for motorized systems. It aims to determine crucial parameters like friction coefficients, moment of inertia, and back EMF constant to accurately model the behavior and dynamics of DC motors for control, simulation, and system analysis. It begins with a population of potential solutions, each evaluated by a fitness function that indicates how well it meets the objective. High-fitness solutions are selected to reproduce, creating new solutions through crossover by combining traits of two solutions and mutation by introducing random changes. This process iterates, with each generation evolving toward better solutions.

For instance, in [92], GA is employed for parameter estimation of a geared DC motor, while in [93], it is used to estimate the PMDC motor used in a wheelchair. The study in [94] explores the application of GA for parameter identification in PMSMs under information interference. It assesses the algorithm's effectiveness

by comparing the motor's response with its dynamic model under disturbing influences, demonstrating successful parameter identification despite resource-intensive computational requirements. Fig. 17 depicts the GA working principle. Each chromosome's fitness is evaluated and best chromosomes are chosen for reproduction. Following selection, crossover and mutation processes are applied to introduce variation and explore new solutions. Crossover combines parts of selected chromosomes, while mutation slightly alters them to maintain genetic diversity. After these operations, the best parameters are added to a new population, replacing the old one.

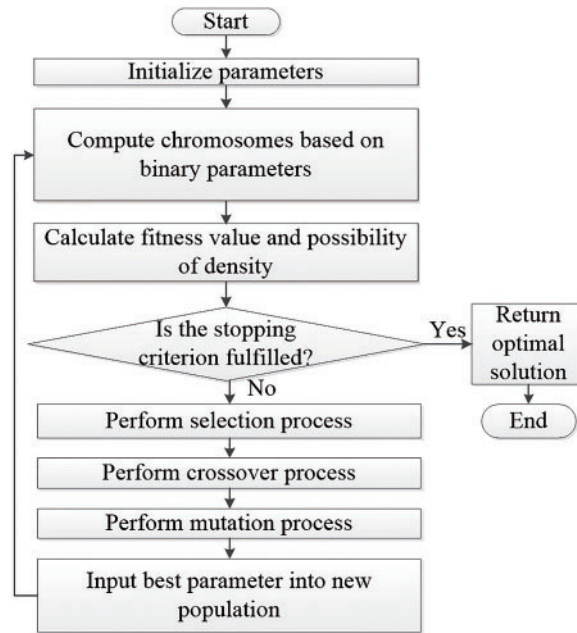


Figure 17: GA flowchart

In [95], a multi-objective GA (MOGA) is utilized for dynamic modeling and parameter estimation of a hydraulic robot manipulator, facilitating simultaneous evaluation of performance objectives such as accuracy, speed, and energy efficiency, leading to improved convergence of estimated parameters compared to single-objective approaches. Similarly, in [96], GA, MOGA, and Niche Pareto GA (NPGA) are employed for analyzing magnetic flux distribution, with NPGA demonstrating superior performance in providing optimal solutions due to its enhanced growth and convergence characteristics compared to proportionate selection.

The study in [97] aims to enhance the tracking accuracy of a BLDC-based linear motor stage used in printing applications by addressing nonlinear friction dynamics. The Hsieh-Pan model [98] is used to account for friction-induced behaviors, such as stick-slip motion, which can disrupt motor precision. GA iteratively adjust parameters by evolving a population of possible solutions and minimizes these friction effects. Through selection and mutation, GA finds an optimal parameter set to counteract static friction, ensuring smoother, more accurate motor performance. This model has been employed to describe the nonlinear static friction as depicted in Fig. 18, and GA is used to identify its parameters. Additionally, PSO-based optimization is applied to fine-tune parameters in a disturbance-observer-based variable structure controller, resulting in improved tracking response. The PSO with constriction coefficient is used to aid particles in converging quickly over time by reducing their oscillation amplitude as the particles focus on nearby points that have yielded optimal results previously. To incorporate the constriction factor, the PSO's

velocity update equation is modified as follows [99]:

$$v_i(k+1) = \kappa [wv_i(k) + c_1 r_1 \cdot (p_{\text{best},i} - x_i(k))] \quad (36)$$

where κ is the constriction factor, and it is typically defined as:

$$\kappa = \frac{2}{|2 - \phi - \sqrt{\phi^2 - 4\phi}|} \quad (37)$$

with $\phi = c_1 + c_2$ being the sum of the acceleration coefficients. The constriction factor helps to prevent the velocity from growing too large and stabilizes the particle movement, especially as the algorithm converges toward an optimal solution. It adjusts the exploration-exploitation balance by limiting the maximum velocity of the particles. Real-time implementation of the proposed controller achieves positional accuracy better than 0.1 μm , demonstrating its effectiveness in practice.

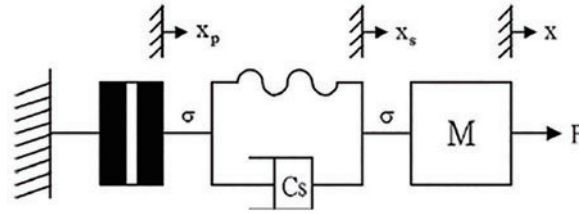


Figure 18: Static friction based on Hsieh-Pan model employed in [97] where σ is the static friction force that arises when an external force F is applied on M which is the mass of the positioning stage

The work in [100] addresses the challenge of system identification in process industries when open-loop data is unavailable, emphasizing the importance of selecting an appropriate model structure. It compares a first-order integer model with four different fractional models for closed-loop identification of a DC motor, optimizing fractional model parameters using GA to minimize the sum of squared errors (SSE). The results indicate that fractional order (FO) models generally provide a better fit than the first-order integer model, with the fractional model featuring the fewest parameters demonstrating the best performance among the identified models.

In [101], a technique for identifying seven parameters of a DC motor and drive is outlined, taking into account speed and current step responses. Fig. 19 depicts the system identification workflow. Various evolutionary methods, such as GA, DE, TLBO, and ABC, are utilized and compared, with DE/rand/1/exp demonstrating superior performance for this task. Additionally, strategies such as dividing the motor model simulation time and employing short-term memory assistance are suggested to improve results and decrease computation time, respectively.

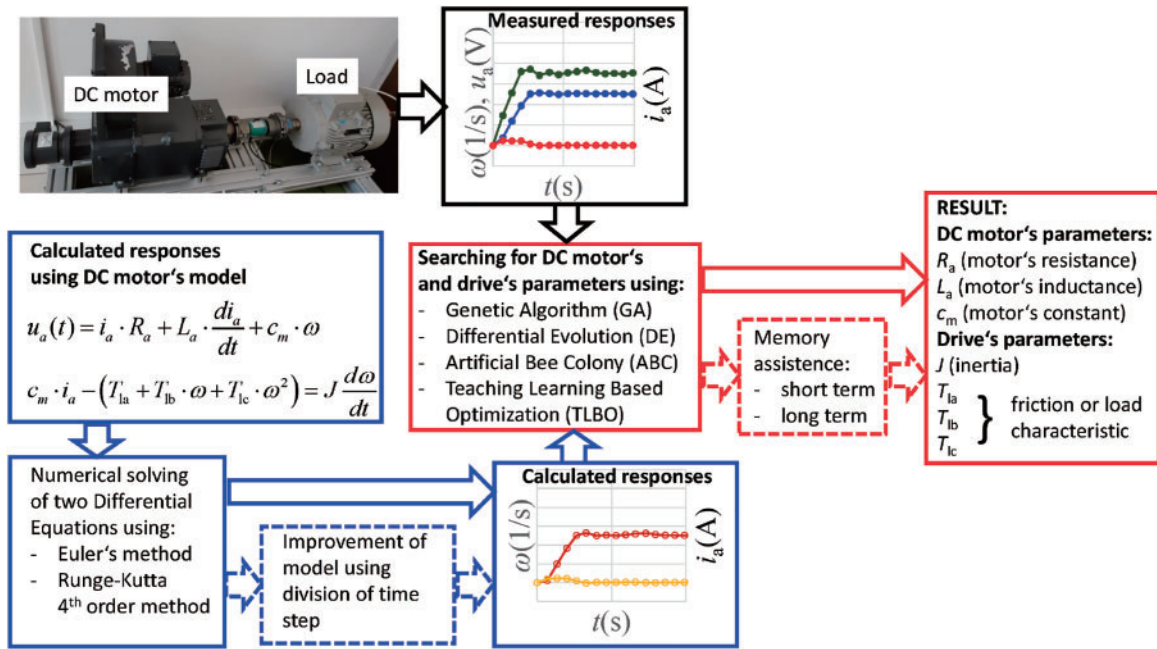


Figure 19: The system identification process in [101]

5.2.4 Grey Wolf Optimization (GWO)

GWO, a metaheuristic optimization algorithm developed in 2014, seeks optimal solutions by mimicking the hierarchical hunting behavior of grey wolves in nature [18]. The algorithm begins by generating a set of random candidate solutions. The grey wolf population is divided into four categories: alpha (α), beta (β), delta (δ), and omega (ω). The α wolf represents the fittest solution, followed by β and δ wolves. The remaining candidate solutions are denoted by the ω category. During the hunting process, the top three wolves, namely α , β , and δ , guide the other wolves towards promising areas in the search space. The ω wolves update their positions as described below:

$$\vec{D} = |\vec{C} \cdot \vec{X}_p - \vec{X}(t)| \quad (38)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (39)$$

where \vec{X} is the position vector of grey wolf, \vec{X}_p is the position vector of the prey, t denotes the current iteration, \vec{D} is distance vector, \vec{A} and \vec{C} are coefficient vectors as shown in (40):

$$\vec{A} = 2\vec{a} \cdot \vec{r}_1 - \vec{a}; \quad \vec{C} = 2 \cdot \vec{r}_2 \quad (40)$$

where \vec{a} is linearly decreased from 2 to 0 for both the exploration and exploitation phases, \vec{r}_1 , \vec{r}_2 are the random vectors in $[0, 1]$. During the algorithm's execution, half of the iterations focus on exploitation, indicated by $A < 1$, while the other half prioritize exploration with $A \geq 1$. The hunting process concludes when the prey ceases to move, and the grey wolves proceed to attack it.

Throughout the optimization process, the ω wolves simultaneously update their positions by encircling the α , β , and δ wolves. The current position vectors of the α , β , and δ wolves are represented by \vec{X}_1 , \vec{X}_2 , and \vec{X}_3 , respectively. The position update of the ω wolves is performed in the following manner:

$$\begin{cases} \vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}| \\ \vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}| \\ \vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}| \end{cases} \quad (41)$$

$$\begin{cases} \vec{X}_1 = \vec{X}_\alpha - \vec{A}_1 \cdot \vec{D}_\alpha \\ \vec{X}_2 = \vec{X}_\beta - \vec{A}_2 \cdot \vec{D}_\beta \\ \vec{X}_3 = \vec{X}_\delta - \vec{A}_3 \cdot \vec{D}_\delta \end{cases} \quad (42)$$

$$\vec{X}_i(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (43)$$

where \vec{A}_1, \vec{A}_2 and \vec{A}_3 represent the adaptive vectors, and $\vec{X}(t+1)$ represents the position update for the ω wolf. Fig. 20 provides an overview of the GWO algorithm.

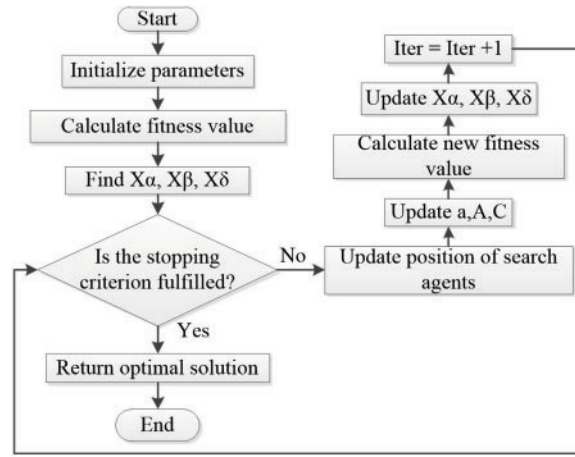


Figure 20: GWO flowchart

The work in [102] utilized GWO, JA, and CS to identify the parameters of DC motors, focusing on their dynamic responses to step inputs and employing steady-state and transient-state relationships as search constraints. The data acquisition setup for system identification is depicted in Fig. 21 where a PIC18F4550 with a sampling period of 0.001 s is used as the main controller. The three algorithms were tested using real-world signals to assess their performance under practical conditions. Unlike synthetic data, real-world signals typically contain noise and higher uncertainty due to factors like measurement system characteristics and environmental conditions. In this scenario, each algorithm was evaluated with a population size of 30 individuals and a maximum of 50 iterations. The results showed that all three algorithms performed competitively, with GWO slightly outperforming the others.

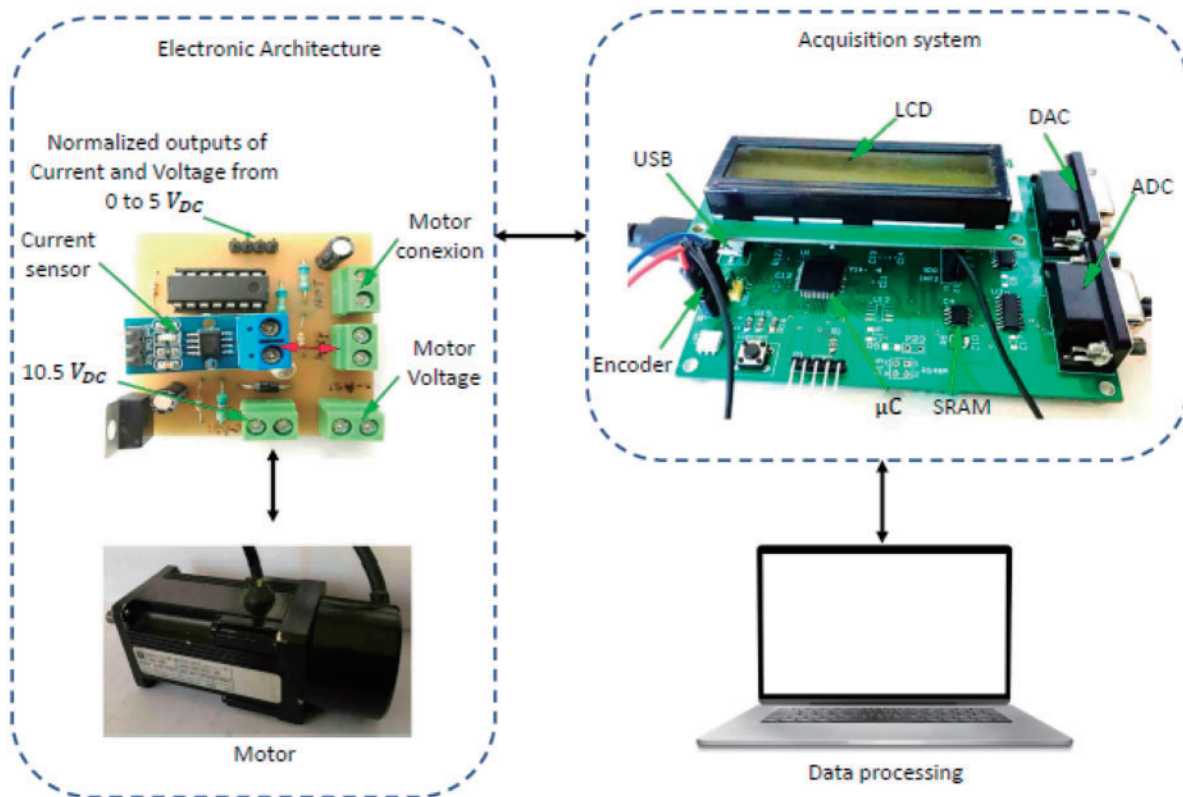


Figure 21: Illustration of the data acquisition setup for DC motor system identification in [102]

The work in [103] addresses the challenge of choosing the correct parameters for motor controllers in a wheeled mobile robot's design. It emphasizes the importance of accurately tuning these controllers based on known parameters of the robot's PMDC motors to improve its dynamics. The approach combines the GWO algorithm with initial parameter estimates from the characteristic point method (CPM), resulting in CPM-GWO. Findings indicate that CPM-GWO outperforms its individual components in terms of RMSE. Another approach in [104] using GWO attained the lowest MSE of 0.43%, demonstrating superior accuracy in parametric estimation compared to Steiglitz-McBride, JA, and GA. The algorithms were evaluated using real current and velocity signals containing inherent noise. To mitigate its impact on parameter estimation, the velocity signal was processed using a Chebyshev software filter, while the current signal underwent hardware-based filtering.

5.2.5 Bat Algorithm (BA)

BA is a metaheuristic optimization algorithm which mimics the echolocation behavior of bats to solve complex optimization problems [105]. Bats use echolocation to navigate and hunt prey by emitting sound waves and listening to the echoes that bounce back from obstacles and prey. To identify the most optimal solution, the algorithm generates a bat population and allows the bats to roam randomly within a restricted space. Each bat discovers the best answer, followed by rating the bats to choose the best option among them. BA often exhibits strong convergence properties which leads to faster and more reliable parameter estimation.

Fig. 22 presents the working principle of BA. The algorithm begins by initializing parameters such as frequency, velocity, position, pulse emission rate, and loudness for each bat in the population. The fitness of each bat is then evaluated, and the best solution is recorded. If a random number exceeds a threshold, the best solution is selected to explore nearby areas for refinement. After generating a new population, the fitness of the bats is recalculated. The bats are then ranked to identify the best solution in the current iteration. This method balances exploration and exploitation to find the best solution efficiently.

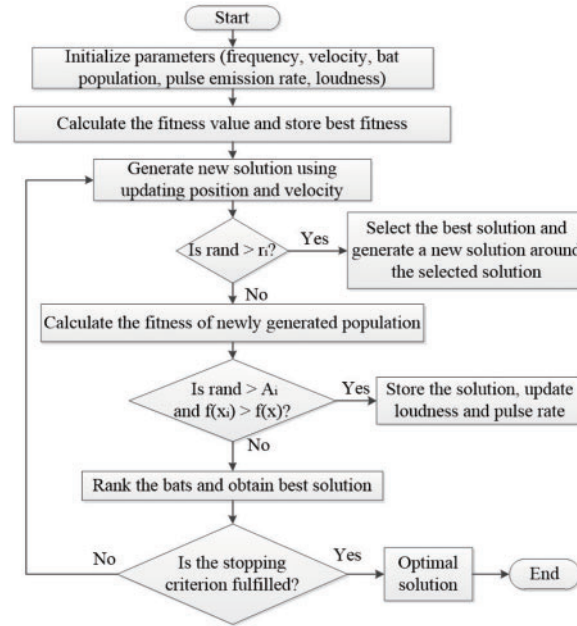


Figure 22: BA flowchart

Each bat is represented as a solution in the search space and the algorithm updates its position and velocity to move towards better solutions. Velocity, position and frequency update can be expressed as:

$$v_i^t = v_i^{t-1} + (x_i^t - x_*) f_i \quad (44)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (45)$$

$$f_i = f_{\min} + (f_{\max} - f_{\min}) \beta \quad (46)$$

where t is iteration, i represented bat, v_i^t is updated velocity, x_i^t is updated position, x_* is global best solution, f_i is frequency for bat i , f_{\min} and f_{\max} are the minimum and maximum frequencies, respectively, $\beta \in [0, 1]$ is a random vector drawn from a uniform distribution. For local search,

$$x_{\text{new}} = x_{\text{old}} + \epsilon A^t \quad (47)$$

where x_{new} is new position after local search, x_{old} is current best solution, $\epsilon \in [-1, 1]$ is a random number while $A^t = \langle A_i^t \rangle$ is the average loudness of all bats at this time step. The loudness and pulse emission rate are dynamically adjusted as the algorithm progresses. These two factors balance exploration (global search) and exploitation (local search),

$$A_i^{t+1} = \alpha A_i^t ; r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)] \quad (48)$$

where A_i^t and r_i^t are the loudness and pulse emission rate at iteration t , r_i^0 is initial pulse emission rate, α is a constant that controls the decrease in loudness over time. γ is a constant that control how fast the emission rate increases over time. For $0 < \alpha < 1$ and $\gamma > 0$, $A_i^t \rightarrow 0$ and $r_i^t \rightarrow r_i^0$ as $t \rightarrow \infty$.

In [106], the Bat Algorithm (BA) is utilized to tackle BLDC wheel motor challenges, encompassing 5 design parameters and 6 constraints for the mono-objective problem, and 2 objectives, 5 design parameters, and 5 constraints for the multiobjective version. Additionally, the results are compared with the Non-Dominated Sorting Genetic Algorithm—Version II (NSGA-II), which stands as one of the most efficient multiobjective algorithms in the literature. The findings reveal promising outcomes for the mono-objective BA, with efficiency consistently nearing 95.23% when addressing the BLDC motor problem. The study outlined in [107] applies the BA to estimate parameters of a closed-loop operating DC servo motor in a robot. This approach proves particularly valuable for robots or machines lacking complete information.

5.2.6 Neural Network (NN)

A neural network (NN) can model motor behavior using input parameters like voltage, current, temperature, and load conditions, learning nonlinear relationships with motor performance metrics such as speed, torque, and efficiency through historical data or simulations. NN can be effectively used to model the behavior of DC motors by learning the relationships between the motor's input and output variables. The inputs required for training include a dataset of historical input-output data, network architecture specifications (such as the number of neurons and layers), weights, biases, learning rate, step size, loss function, batch size, and the number of epochs. The core idea is to treat the DC motor as a black-box system and leverage the neural network to approximate its dynamics. By doing so, the neural network can capture the motor's nonlinear characteristics without the need for a detailed mathematical model, making it a powerful tool for system identification and control.

Fig. 23 shows the NN working principle. It begins by defining input and target data, which is then divided into three sets: training, testing, and validation data. Next, different model architectures are applied, consisting of input, hidden, and output layers with neurons. The weights and biases are initialized for these layers, and the network begins training by adjusting weights based on the data. During training, the model calculates the error, which represents the difference between the network's output and the target data. If the error is within acceptable limits, training ends, otherwise, adjustments are made and training continues.

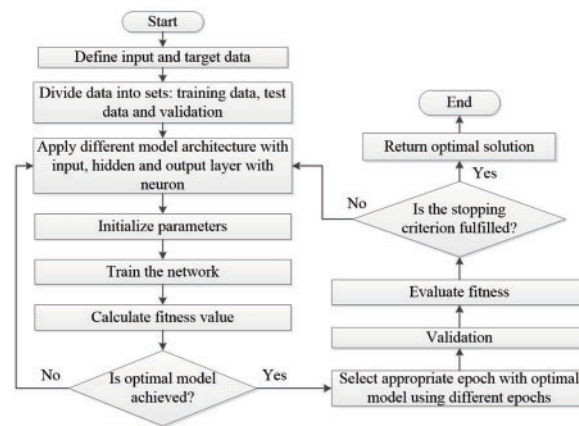


Figure 23: NN flowchart

In [108], a novel method is introduced for identifying robot arm manipulators driven by DC motors with a 131.25 to 1 reduction gearbox, employing convolutional NN (CNN) for efficient feature extraction and pattern recognition. This approach optimizes network architecture, hyperparameters, and training procedures to enhance accuracy and generalization, ensuring reliable parameter identification across varied operating conditions. However, the CNN's effectiveness heavily relies on high-quality, representative training data, as insufficient or biased data can result in inaccurate parameter identification and compromised performance.

The study in [109] highlights the improvement of the accuracy of modeling PMSMs by integrating computational simulations with experimental data, which combines finite element calculations with data-based system identification techniques. Integrating computational simulations with experimental data offers a holistic understanding of PMSM behavior. This approach can capture both electromagnetic phenomena and real-world operating conditions. Another work in [110] investigated data-driven approaches for diagnosing incipient faults in DC motors. The methods used in this research analyze operational data and find patterns suggestive of impending DC motor problems using machine learning algorithms. For instance, decision trees, artificial NN (ANN), and support vector machines (SVMs). This involves preprocessing the operational data, selecting appropriate features, tuning hyperparameters, and evaluating model performance using suitable metrics.

In [111], a recurrent neural network (RNN) is utilized to identify PMDC motors. RNNs are specialized for processing sequential data by retaining past input memory, allowing them to learn directly from input-output data without the need for complex models or extensive experimentation. This approach saves time by reducing the requirement for manual experimentation and iterative model tuning. The experimental setup, illustrated in Fig. 24, involves supplying the PMDC machine with an H-bridge converter controlled using a bipolar control strategy. Two PWM signals generated by the control system OP5600 regulate IR2111 drivers, which control MOSFET transistors (T1–T4). Current sensing is facilitated by a ACS712T sensor in the armature circuit, while speed sensing is conducted using an IRC sensor LARM, providing 1000 impulses per revolution.

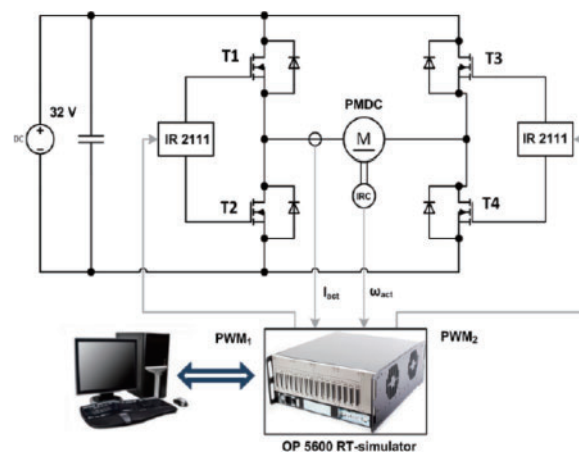


Figure 24: Experimental setup for PMDC motor identification via RNN [111]

In [112], a novel feedforward neural network (FFNN) and Elman neural network (ENN) are proposed for parameter estimation of induction motors. The training process of these neural network structures was conducted with unmemorized training, demonstrating better performance for the single-cage model,

which has 5 parameters compared to the double-cage model with 7 parameters. Fig. 25 illustrates successful convergence and improved estimation for the single-cage model using both ENN and FFNN networks.

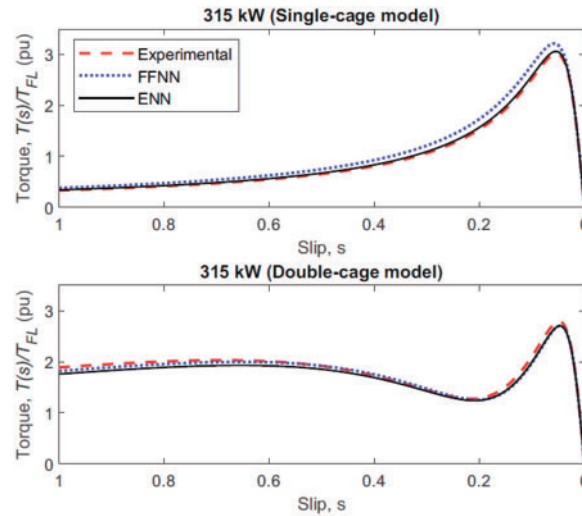


Figure 25: Torque-speed curves for 315 kW motor comparing single and double cage model [112]

Additionally, in [113], an approach employing artificial neural networks (ANN), support vector machines (SVM), and extreme learning machines (ELM) for power consumption estimation of UAVs is presented. The mean absolute percentage error values for ANN, ELM, and SVM are reported as 0.389%, 1.258%, and 0.269%, respectively, with SVM outperforming ANN and ELM. This approach enables the determination of safe flight routes by adjusting the speed and pulse degree of the UAV based on predicted battery life during strategic flights.

5.2.7 Other Metaheuristic Methods

In [114], the WOA is employed to identify unknown parameters of a nonlinear PMSM, with validation against known parameter values. Comparative analysis with other algorithms such as BA, Self Adaptive Learning BA (SALBA), and PSO demonstrates superior convergence rates of the proposed WOA-based approach. While its mean square error performance is better than BA and PSO but not SALBA, the scheme shows promise for real-time implementation in hardware.

This study in [115] proposes a multi-objective online tuning optimization method for adaptively adjusting the velocity control parameters of a PMDC motor as illustrated in Fig. 26. By considering both modeled error and sensitivity, it selects the best compromise solution through a Pareto dominance-based selection process, while introducing modifications to the DE algorithm. Simulation results show improved velocity regulation control under parametric uncertainties and discontinuous dynamic loads compared to other optimization algorithms such as multi-objective DE, PSO, and NSGA-II.

Another approach presented in [116] focuses on applying the Flower Pollination Algorithm (FPA) to parameter identification for a DC motor model. FPA reduces the likelihood of getting trapped in local minima and provides accurate identification of eight ideal DC motor model parameters that capture system dynamics effectively.

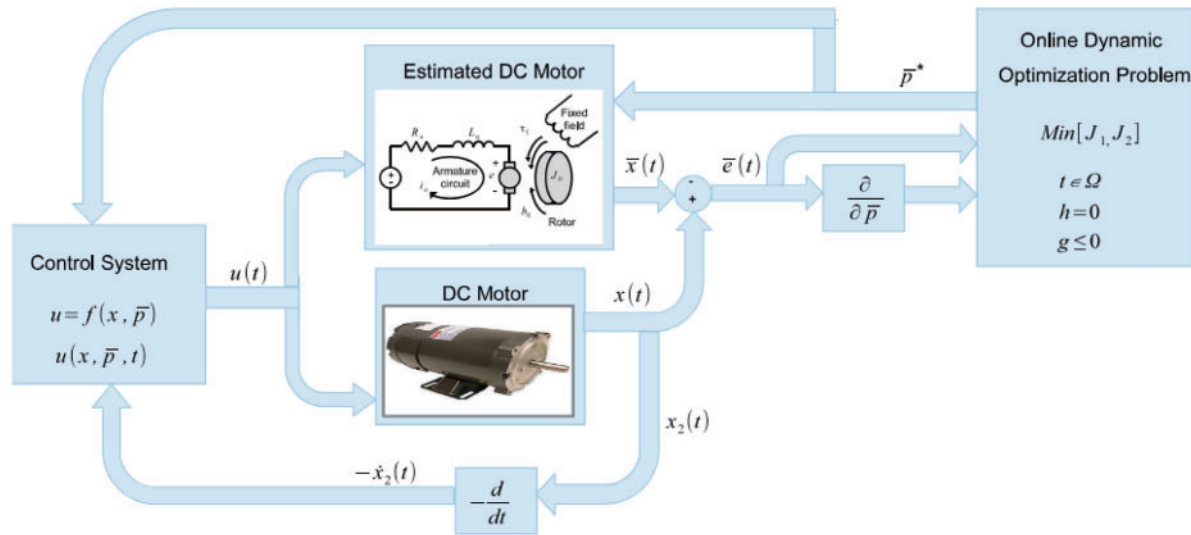


Figure 26: Schematic diagram of the dynamic optimization process for the on-line PMDC control parameter estimation in [115]

Rodriguez-Abreo et al. [117] focused on motor parameter identification using CS, leveraging steady-state relations for precise parameter identification under stable operating conditions. The modifications made to the original CS algorithm allowed for the determination of motor parameter settings with minimal root mean square error, ensuring accurate results for both real signals and simulation data.

In [118], PMSM parameter identification using chaotic Gaussian-Cauchy RAO (CGCRAO) algorithm employs Tent chaotic mapping for population initialization, improving population diversity and promoting a more uniform distribution of initial solutions across the search space. CGCRAO algorithm retains the simplicity of the original RAO-1 algorithm, with fewer parameters and a concise core formula [119]. The CGCRAO algorithm achieved identification errors of less than 1% for resistance and flux while maintaining an error of only 1.0547% for inductance, demonstrating superior performance compared to other algorithms. CGCRAO attained the lowest fitness value (0.02411), outperforming PSO (0.03682), DE (0.03845), GWO (0.02661), and RAO-1 (0.02935).

The study in [120] focuses on identifying the continuous-time Hammerstein model under conditions of sparse measurement data using random average marine predators algorithm (RAMPA) with a tunable step-size adaptive coefficient (CF) (RAMPA-TCF). MPA utilizes Lévy flight and Brownian motion to improve global search efficiency while incorporating adaptive mechanisms for local refinement. This dynamic search strategy facilitates faster convergence and mitigates the risk of premature stagnation [121]. The main challenge was incomplete data affecting parameter estimation accuracy. The piecewise affine function improves nonlinear subsystem modeling, enhancing robustness against noise. Performance was assessed under sparse measurement data conditions of 5%, 15%, 35%, and 50% missing data. Even with 50% missing data, RAMPA-TCF demonstrated the highest accuracy in identifying the Hammerstein model compared to other algorithms.

5.2.8 Hybrid Methods

In [122], a hybrid algorithm combining the Arithmetic Optimization Algorithm (AOA) [123] and the Harris Hawk Optimization (HHO) algorithm [124] is proposed for more efficient exploitation mechanisms.

The results indicate that the AOA-HHO hybrid outperforms both AOA and other algorithms in terms of minimizing the IAE while achieving reasonable overshoot, settling time, and rise time. Hybridization of PSO and gravitational search algorithm (PSOGSA) demonstrates the ability to achieve its global optimal solution within a reduced number of iterations [125]. Hybridizing metaheuristic methods typically offers the advantage of combining the strengths of multiple algorithms to enhance optimization performance and address the limitations inherent in individual approaches. However, achieving a more effective hybrid requires careful design and technique to ensure it performs better than its individual components [126–128].

The hybrid PSO and Sine Cosine Algorithm (SCA) optimization algorithm (HPSOA) proposed in [129] attained a minimal error of 0.0013, which is approximately ten times lower than the minimal errors produced by PSO and seven times lower than those obtained by SCA. The parameters estimated using HPSOA were validated against real motor data, demonstrating a high degree of accuracy. The steady-state speed error was measured at 0.58%, while the steady-state current error was 3.75%. These minimal errors substantiate the robustness and effectiveness of HPSOA in real-world applications.

Another approach in [130] employs simulated annealing adaptive particle swarm optimization (SAAPSO) to optimize the position control parameters. A Hammerstein-Wiener model based on a backpropagation neural network is developed. It integrates both linear and nonlinear modules to effectively capture the system's linear and nonlinear characteristics. The proposed model achieves a substantial improvement in accuracy compared to traditional linear models help to reduce the maximum absolute position errors by 44.801% along the *X*-axis and 45.876% along the *Y*-axis. The symmetrical mean absolute percentage errors (SMAPEs) decreased by 86.804% along the *X*-axis and 85.996% along the *Y*-axis. Hybrid algorithms tend to produce more accurate results because they combine the best features of multiple optimization techniques [131].

6 Analysis and Discussion

This section delves into the synthesized findings from the preceding systematic review of literature concerning system identification techniques for motorized systems under various perturbations. Tables 1 and 2 summarize the system identification techniques employed for perturbed DC and AC motors, respectively. Each table organizes the studies according to the modeling method, motor structure, parameter optimization method, source of perturbations, and their respective applications or use cases. The diverse range of studies documented in the tables has been graphically represented below to offer a clearer overview of prevalent trends and distinct methodologies in both DC and AC motor applications.

Table 1: Comparative analysis of system identification method for perturbed DC motors

Ref.	Modeling method	Structure	Parameter optimization method	Type/source of perturbations	Application
[36]	Grey	1 st Order TF	–	Load Torque	SC
[37]	Grey	2 nd Order TF	–	Speed	SC
[38]	Grey	2 nd Order TF	–	Speed, excessive heating of anchor's winding	UMLI
[40]	Grey	ARX,ARMAX	–	–	DCS
[41]	Grey	SS	–	Noise	GM

(Continued)

Table 1 (continued)

Ref.	Modeling method	Structure	Parameter optimization method	Type/source of perturbations	Application
[42]	Black	ARMAX	–	Load	PSys
[43]	Grey	–	–	Load	GM
[44]	Grey	2 nd Order TF	–	–	CD
[52]	Grey	–	–	Load, speed	UAV
[53]	Black	ARMAX	–	Load	PSys
[54]	Black	ARX, ARMAX	–	Noise	FMS
[55]	Black	ARMAX	–	Friction	WMR
[59]	Grey	ARX	LS- ARX, NLS	–	CD
[60]	Grey	3 rd Order TF	LS	Speed	UAV
[61]	Grey	2 nd Order TF	LS	Discontinuities	CD
[63]	Grey	–	RLS	Nonlinear Load Torque	CD
[68]	Grey	ARX	RLS	–	CD
[69]	Grey	ARMAX	LS	Noise, gaussian disturbance	GM
[70]	Grey	4 th Order TF	LS	Disturbance, noise	CD
[71]	Grey	–	LS	Load	HPKM TR
[73]	Grey	–	LS	Speed	WMR
[74]	Grey	–	CLS	Speed	SC
[56]	Black	NARX	LM	Hysteresis and friction in valves	CD
[58]	Black	2 nd Order TF	LM	Sensor noise	SC
[64]	Grey	–	LM, RLS	Position, orientation and speed error	WMR
[82]	Grey	3 rd Order TF	PSO	Load, speed	SC
[83]	Grey	2 nd Order TF	PSO	Disturbance	CD
[84]	Grey	3 rd Order TF	PSO	–	GM
[86]	White	–	PSO	Noise, nonlinear friction	UR5 RM
[88]	Grey	–	PSO, ACO, ABC	Load, angular speed error, armature current error	CD
[95]	Grey	–	GA	Noise, friction	Hydraulic RM

(Continued)

Table 1 (continued)

Ref.	Modeling method	Structure	Parameter optimization method	Type/source of perturbations	Application
[92]	Grey	3 rd Order TF	GA	Voltage	GM
[93]	Grey	SS	GA	Angular speed error, armature current error	WC
[96]	Grey	–	GA	–	GM
[97]	Grey	–	GA, PSO	Load, position, friction	CD
[100]	Black	–	GA	Disturbance, sensor noise	GM
[101]	Grey	–	GA, DE, TLBO, ABC	Load	CD
[102]	Grey	1 st Order TF	GWO, JA, CS	Noise	GM
[103]	Grey	–	GWO	Friction	WMR
[104]	Grey	2 nd Order TF	GWO	–	GM
[106]	Grey	–	BA	–	GM
[107]	Grey	2 nd Order TF	BA	Speed	RM
[108]	Grey	–	CNN	Noise	RM
[110]	Grey	–	CNN, RNN	Noise, load	GM
[111]	Black	NARX	RNN	–	GM
[113]	Black	–	NN	Speed	UAV
[115]	Grey	SS	DE	Load	CD
[116]	Grey	2 nd Order TF	FPA	Speed	GM
[117]	Grey	–	CS	Noise	GM
[122]	Grey	2 nd Order TF	AOA-HHO	Speed	CD

Table 2: Comparative analysis of system identification method for perturbed PMSM/AC motors

Ref.	Modeling method	Parameter optimization method	Type/source of perturbations	Application
[45]	Grey	–	Load	GM
[47]	Grey	–	Load, speed	GM
[48]	Grey	–	Load, speed, noise	GM

(Continued)

Table 2 (continued)

Ref.	Modeling method	Parameter optimization method	Type/source of perturbations	Application
[49]	Grey	–	Speed	CD
[50]	Grey	–	Speed	CD
[51]	Grey	–	Load, speed	CD
[65]	Grey	RLS	Speed	CD
[66]	Grey	RLS	Voltage	GM
[67]	Grey	LS	Voltage, speed	CD
[81]	Grey	PSO	Noise	AIS
[87]	Grey	PSO	Speed	GM
[89]	Grey	PSO	Load, speed	AIS
[90]	Grey	PSO	Load, speed	HEV
[91]	Grey	PSO	Voltage, speed	GM
[94]	Grey	GA	Noise	GM
[109]	Grey	ANN	Load	GM
[112]	Grey	NN	Load, speed	GM
[114]	Grey	WOA	Noise	GM
[118]	Grey	CGROA	Speed, load	GM
[125]	Grey	PSOGSA	Load	GM
[129]	Grey	HPSOA	Load, friction	GM

Such perturbations can significantly impact real-time control applications by affecting system stability, accuracy, and overall performance. Load variations cause sudden torque demands, leading to speed deviations that can destabilize the system if not properly managed. Speed fluctuations, often resulting from external disturbances or supply inconsistencies, introduce tracking errors that reduce the precision of motion control. Friction, particularly at low speeds, leads to nonlinear effects such as stick-slip motion and increased energy losses, making precise control difficult. Hysteresis, commonly found in magnetic materials and mechanical systems, results in lagging system responses and inconsistent behavior, which complicates accurate control. Additionally, noise, both mechanical and electrical, interferes with sensor measurements and control signals, leading to erroneous feedback and degraded system performance.

These perturbations collectively challenge the reliability and responsiveness of motorized systems, particularly in high-precision and high-speed applications. Variations in load and speed can cause oscillations, overshooting, or delays in response, reducing efficiency and increasing wear on components. Friction and hysteresis introduce unwanted nonlinearities that make modeling and prediction difficult, leading to unpredictable deviations in system behavior. Noise further compounds these issues by distorting control signals, potentially causing instability or reduced sensitivity to actual system dynamics. In critical applications such as robotics, industrial automation, and electric vehicles, these perturbations can lead to performance degradation, increased energy consumption, and even system failures if not properly accounted for in the control design.

Fig. 27 demonstrates the distribution of parameter optimization methods utilized in the reviewed articles, where LS and PSO emerge as the most frequently used techniques. As detailed in Table 3, LS is widely adopted due to its versatility in handling both linear and nonlinear functions and its adaptability to real-time applications, although it remains sensitive to data outliers which can affect parameter estimation. In addition, the preference for LS can be attributed to its widespread availability in system identification tools and its designation as a default technique in popular software applications like MATLAB, which enhances its accessibility and ease of use. PSO is favored for its robustness and efficient control parameter tuning but can struggle with exploring certain regions of the search space. Additionally, the table highlights other methods like GA and NN, which excel in exploring large search spaces and extracting relevant features from complex data, respectively. However, these methods also face challenges such as high computational demand and overfitting. The diversity of other population-based optimization strategies such as TLBO, GWO, DE, and CS, reflects ongoing innovation in the field to address the unique challenges posed by different motor configurations and perturbation types. Hybrid algorithms integrate the advantages of multiple optimization techniques, enhancing both exploration and exploitation capabilities, thereby achieving superior performance. This analysis underscores the importance of selecting an optimization method that not only aligns with the specific dynamics and challenges of motor system perturbations but also balances between computational efficiency and the ability to handle complex data structures.

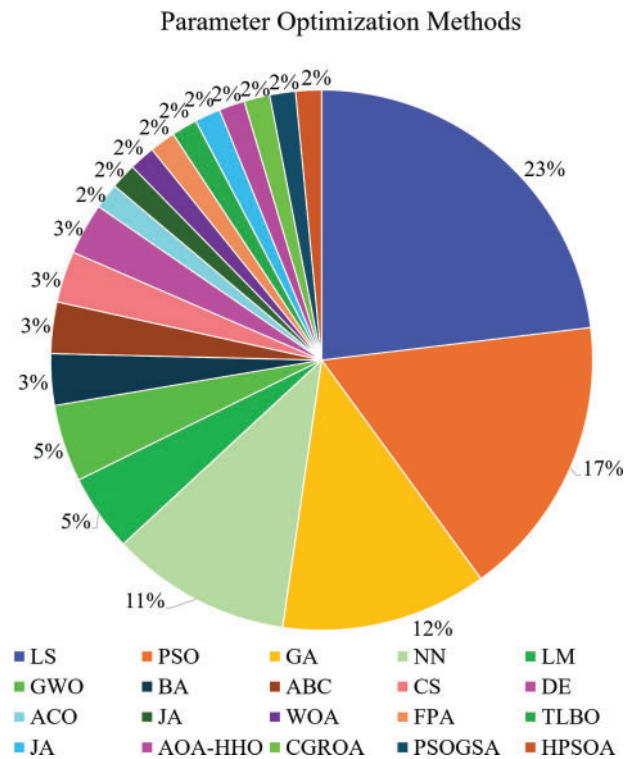


Figure 27: Distribution of parameter optimization methods used in system identification for motorized systems

Table 3: Advantages and disadvantages of parameter optimization methods in system identification for perturbed motorized systems

Method	Advantages	Disadvantages
LS	Handles linear/nonlinear functions; low computational demand; default in many software tools	Sensitive to outliers; difficulties in handling constraints on the parameters
LM	Fast, accurate parameter estimation; default in many software tools	Sensitive to initial parameters; memory intensive for large-scale optimization
PSO	Robust; simple implementation	Parameter sensitive; limited exploration area
GA	Explores large spaces; robust against local optima	High computational cost; complex tuning
NN	Handles complex behavior; extracts features	Needs large data; may overfit
GWO	Efficient in high-dimensional spaces; simple	Limited large-scale scalability
BA	Good for dynamic problems; converges well	Struggles with complex constraints
ACO	Adapts based on pheromone trails; explores diverse regions	Slow convergence
ABC	Effectively explores diverse spaces	Can prematurely converge
CS	Strong global search capabilities	Sensitive to control parameters
WOA	Robust against global optima; adaptable	Slow convergence; low precision
FPA	Versatile and easy to implement	Lacks maturity; limited scalability
DE	Efficient in finding global optima	Struggles with noise interference
TLBO	Simple; quick convergence	Difficulty in handling constraints
JA	Simple; parameter-free	Limited scalability
HHO	Fast and strong local search capabilities	Computationally complex
ROA	High flexibility and versatile	Limited large-scale scalability
MPA	Efficient in finding local optima	Slow convergence
GSA	Able to handle multimodal problem	High computational cost
SCA	Simple, fast convergence	Can prematurely converge
SA	Explores large search spaces	Slow convergence

Figs. 28 and 29 offer insights into the types of perturbations and their correlation with modeling methods for both DC and AC motors. From Fig. 28, it is clear that fluctuations in loads and speed are extensively studied in the literature concerning system identification. These variations present challenges in accurately modeling motor behavior due to their dynamic and unpredictable nature. System identification methods enable researchers to derive meaningful insights from data collected during perturbed operation, facilitating the development of models that capture the intricate interactions between motor dynamics and external influences. Understanding how motors respond to these perturbations is essential for optimizing performance across various operational scenarios, ranging from industrial applications to consumer electronics. Furthermore, the precise characterization and simulation of these behaviors contribute to

advancements in control algorithms designed to mitigate the impacts of load and speed fluctuations, thereby enhancing overall motor efficiency and reliability in diverse environments.

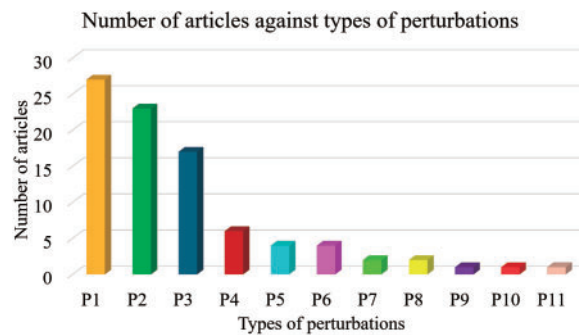


Figure 28: Number of articles against type of perturbations (Note: P1: Speed, P2: Load, P3: Noise, P4: Friction, P5: Voltage, P6: Disturbance, P7: Current, P8: Position & Orientation, P9: Heat, P10: Discontinuities, P11: Hysteresis)

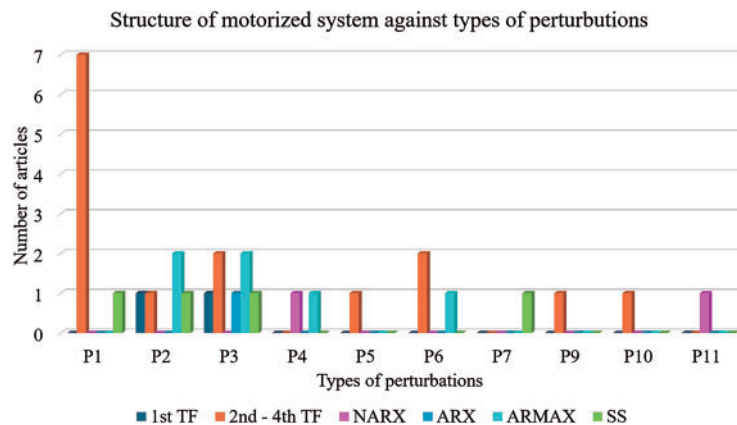


Figure 29: Number of articles for structure of motorized systems against types of perturbations (Note: P1: Speed, P2: Noise, P3: Load, P4: Friction, P5: Disturbance, P6: Voltage, P7: Current, P9: Heat, P10: Discontinuities, P11: Hysteresis)

The model structure selection procedure in system identification is a critical step that determines the accuracy and effectiveness of the identified model in representing system dynamics. The first step involves data analysis, where the system's input-output behavior is examined to determine whether it follows a linear or nonlinear trend. This initial assessment guides the choice between simpler linear models, such as ARX, or more complex nonlinear models, such as NARX. A TF model is an appropriate choice for system identification when the system is linear time-invariant and its behavior can be effectively described by input-output relationships without explicitly modeling internal states. TF models are also beneficial when data availability is limited, as TF allow system characterization based solely on input-output measurements. To ensure model reliability, validation techniques such as Mean Squared Error (MSE), R-squared values, and residual analysis are employed to compare different models and identify the one that achieves the best balance between accuracy and generalization. Additionally, cross-validation is essential to assess the model's performance on unseen data by dividing the dataset into training and validation sets. This step helps mitigate overfitting and enhances the model's robustness for real-world applications.

Based on the plot in Fig. 29, it is clear that the TF structures of order two to four were the most commonly employed across different types of perturbations, particularly excelling in addressing speed and voltage

fluctuations. This is followed by ARMAX models, which also show significant usage, particularly effective in handling noise and load perturbations. SS models, although not as prevalent, are notably utilized for handling friction and voltage disturbances. The ARX model appears to be favored for disturbances like load and voltage, whereas NARX models are selectively applied, mainly for noise and current perturbations. First-order TF models, while less common compared to higher-order TFs, are still relevant in scenarios involving heat and hysteresis perturbations. This distribution of modeling techniques illustrates a targeted approach where specific models are chosen based on their efficacy in handling particular disturbances in motorized systems, ensuring optimal performance and robustness under varying operational conditions.

Fig. 30 illustrates the distribution of research articles focused on the target applications for modeling perturbed motorized systems. Control development (CD) emerges as a primary area, underscoring the importance of capturing motor uncertainties and unmodeled dynamics for designing robust and adaptive controllers. Following closely is generic modeling (GM), which provides foundational insights into various motor types, offering substantial benefits for those seeking to deepen their understanding of motor behaviors. Additional target applications span a diverse range of fields, including UAV, wheeled mobile robots (WMR), robot manipulators (RM), speed control (SC), hybrid electric vehicles (HEV), artificial immune systems (AIS), underground mine lifting installations (UMLI), wheelchairs (WC), DC servomotors (DCS), PSys, FMS, and hybrid parallel kinematic machines tricept robots (HPKM TR). Each application highlights different challenges and requirements, demonstrating the extensive reach and relevance of system identification methods across varied motorized systems.

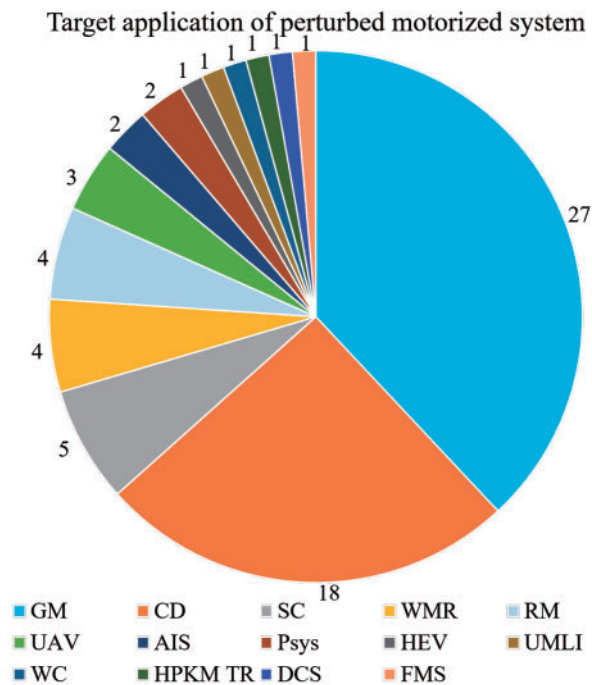


Figure 30: Distribution of articles by target application of perturbed motorized systems

Fig. 31 complements this by presenting a comparative analysis of the model structures employed across various applications, indicating that different models are tailored to the unique challenges of each application. The standard forms of perturbed motor models, specifically TF and SS structures, are prevalently used in GM and CD. The figure highlights the dominance of 2nd–4th order TF models in CD, reflecting their capability to handle the complexities and dynamic behaviors inherent in control systems. SS models are also prominently used in CD and other applications such as DCS, demonstrating their versatility and effectiveness in capturing system dynamics.

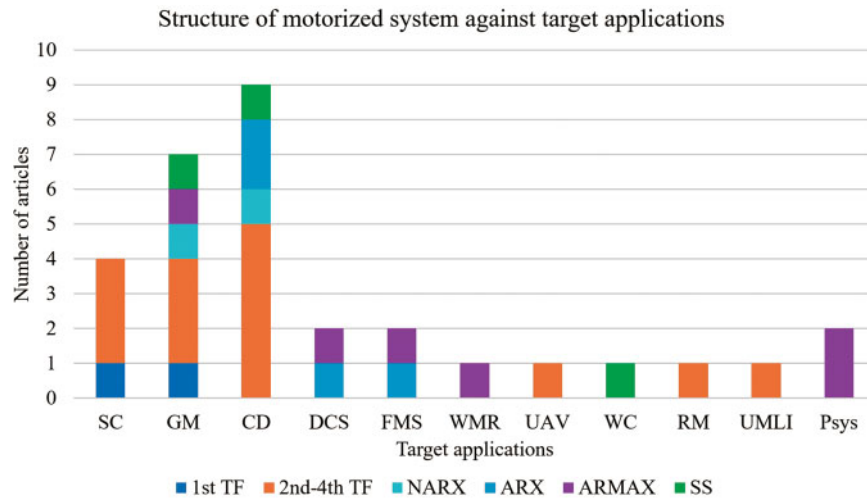


Figure 31: Comparative analysis of model structure of motorized systems across different applications. The bar chart displays the number of articles categorized by target applications such as SC, GM, CD and DCS

Additionally, the figure reveals that other modeling techniques like ARMAX and NARX are applied across various applications, although less frequently. These methods are valuable for their ability to model nonlinear behaviors and handle a variety of perturbations. The diversity in modeling approaches underscores the necessity of selecting appropriate models based on the specific requirements of each application, such as the precision needed in SC, the robustness required in HEV, and the adaptability crucial for WMR. This tailored approach ensures that the unique challenges of each application are effectively addressed, optimizing the performance and reliability of motorized systems in different operational contexts.

Fig. 32 displays the distribution of parameter optimization methods used across various types of perturbations in motorized systems. From the figure, it is evident that parameter optimization methods are selectively utilized across various types of perturbations in motorized systems. LS and LM are commonly used for various perturbations including speed, noise, load, disturbance, and voltage, leveraging their respective capabilities in minimizing residuals and efficiently exploring parameter spaces.

Swarm intelligence algorithms, such as PSO, GWO, ACO, ABC, BA, WOA, CS, AOA, HHO, ROA, MPA, GSA, SCA, and SA are prominently employed for managing perturbations. PSO is favored for its adaptability in systems affected by speed, noise, load, disturbance, friction, current, and voltage, despite its slower convergence speed. GWO is effective in handling noise and friction-related challenges through iterative optimization processes. BA, AOA and HHO are specialized for speed-perturbed systems, while CS is applied to noise-related systems. These algorithms cater to specific perturbation types, adapting their strategies to optimize performance in motorized systems.

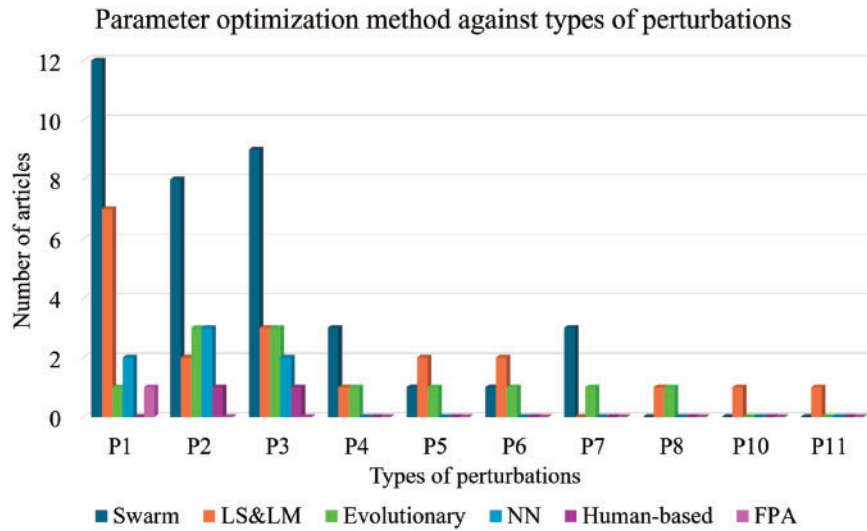


Figure 32: Number of articles for parameter optimization methods against type of perturbations (Note: P1: Speed, P2: Load, P3: Noise, P4: Friction, P5: Voltage, P6: Disturbance, P7: Current, P8: Position & Orientation, P10: Discontinuities, P11: Hysteresis)

Evolutionary algorithms which include GA and DE find application across diverse perturbations such as speed, noise, load, friction, voltage, and current, demonstrating their adaptability in optimizing motorized systems. Human-based algorithms such as TLBO and JA are deployed for load and noise systems, respectively, offering streamlined parameter requirements and effective optimization strategies. NN excel in handling noise, speed, and load systems by learning from extensive data, providing robust solutions to complex perturbations. Finally, methods like FPA specialize in accurately estimating motor model parameters under speed-related perturbations. These trends illustrate a strategic use of optimization methods tailored to specific challenges encountered in perturbed motorized systems.

7 Comparative Performance Analysis

This section compares the performance of several widely used methods identified via the analysis in the previous section, which include grey-box modeling using LS, PSO, GWO, BA, GA, and black-box modeling using NN, to parameterize the DC motor shown in Fig. 3, which is perturbed by a time delay, τ_d . The parameters of the perturbed DC motor are set to $R_a = 2\Omega$, $L_a = 2$ H, $J_m = 0.5$ kgm², $B_m = 1$ Nm/rads⁻¹, $K_t = 5$ Nm/A, $K_e = 0.6$ V/rads⁻¹, and $\tau_d = 1$ s. The resulting plant can be expressed as $P(s) = be^{-\tau_d s}/(a_2 s^2 + a_1 s + a_0)$ where $b = K_t$, $a_2 = J_m L_a$, $a_1 = J_m R_a + B_m L_a$, and $a_0 = B_m R_a + K_t R_a$. Two approaches are employed: the first using the standard open-loop setup, and the second using a closed-loop setup that incorporates a PID controller where the parameters are set as $K_p = 8$, $K_i = 5$, and $K_d = 2$. The maximum number of iterations is set to 50, the population size is fixed at 30. Each algorithm is executed ten times to evaluate its performance metrics. In PSO, w , c_1 and c_2 are set to 0.7, 1.5 and 1.5, respectively. In BA, the frequency is set within the range [0, 1], with both loudness and pulse rate set to 0.5. For GA, the mutation rate and crossover rate are set to 0.1 and 0.8, respectively. LS represents the default method used in the MATLAB System Identification Toolbox. Similarly, the NN structure is optimized using MATLAB's Regression Learner. ITAE is used as the fitness function for each algorithm.

During optimization, a step input is used to estimate the parameters. Table 4 presents a comparison of the performance of different methods in open-loop (OL) and closed-loop (CL) setups during optimization.

For each method, the table shows the statistical analysis of ITAE values, the number of iterations required to achieve them and average CPU time. Based on the ITAE values obtained in this phase, it is evident that the closed-loop approach significantly reduces ITAE values across all algorithms compared to the open-loop case. The fixed PID controller in the CL configuration helps stabilize the system and improve identification accuracy. Among the metaheuristic algorithms, GWO achieves the lowest ITAE in the open-loop case (0.108766), while PSO shows the best performance in the closed-loop setting (0.003729). This suggests that GWO is more effective in open-loop identification, whereas PSO benefits more from the control feedback in the closed-loop system.

Table 4: Numerical performance analysis of PSO, GWO, BA, GA, LS, and NN using the open-loop and closed-loop system identification approach. The table presents a comparison of the ITAE values, the number of iterations needed to achieve those values during optimization and average CPU time

Algorithm	Open-loop					
	PSO	GWO	BA	GA	LS	NN
ITAE						
Mean	0.116400	0.108766	0.429319	0.292678	0.229115	0.221304
Min	0.041662	0.047495	0.102241	0.067564	0.114662	0.091546
Max	0.240532	0.226417	0.761809	0.587750	0.597561	0.989074
Std. Dev.	0.092653	0.081053	0.218425	0.200852	0.207816	0.194561
Iterations						
Mean	46.8	46.9	46.2	47.2	38.5	43.9
Min	45	43	43	45	35	42
Max	49	50	50	49	41	46
Average CPU time (s)	1729.741	1239.095	1794.492	1836.451	1219.264	2036.262
Closed-loop						
ITAE						
Mean	0.003729	0.003880	0.016652	0.013148	0.073398	0.015920
Min	0.003471	0.003653	0.006594	0.005102	0.009460	0.010370
Max	0.004483	0.004176	0.039456	0.023378	0.222000	0.027102
Std. Dev.	0.000406	0.000186	0.009956	0.008048	0.088374	0.005516
Iterations						
Mean	46.4	46.8	47.1	45.9	34.1	44.6
Min	43	45	45	42	28	42
Max	49	48	49	48	37	49
Average CPU time (s)	1698.117	1275.936	1702.824	1792.693	1427.257	2085.348

The standard deviations of ITAE indicate that the performance consistency varies among the algorithms. GWO exhibits the lowest standard deviation in both open-loop (0.081053) and closed-loop (0.000186), indicating stable optimization results. In contrast, NN and GA show higher variability, suggesting inconsistency across multiple runs.

With regard to iteration count, all algorithms complete identification within a similar range (approximately 43–50 iterations), with LS requiring the fewest iterations (38.5 in OL, 34 in CL). This highlights LS's computational efficiency. The CPU time results further confirm this, as LS has the lowest computational cost

(1219.264 s in OL, 1427.257 s in CL). On the other hand, NN is the most computationally expensive, requiring over 2000s in both settings.

To provide an unbiased evaluation of the algorithms' performance, the response of the estimated DC motor model is tested using a variable step input and compared to the actual output in the validation phase. Table 5 tabulates the estimated perturbed DC motor parameters based on the best results (i.e., lowest ITAE) obtained in Table 4.

Table 5: Estimated DC motor parameters via the grey-box modeling using OL and CL setups based on the best results obtained in Table 4

Parameter	Actual values	Estimated values									
		PSO		GWO		BA		GA		LS	
		CL	OL	CL	OL	CL	OL	CL	OL	CL	OL
a_0	5	5.929	5.722	5.077	5.162	7.695	6.361	5.832	5.407	6.310	5.550
a_1	3	3.807	4.357	3.666	2.790	4.990	4.228	4.241	3.637	4.203	4.169
a_2	1	1.173	1.332	1.333	0.755	1.252	1.319	1.722	1.227	1.507	1.453
b	5	4.653	5.360	4.852	5.169	5.104	5.415	4.888	4.691	5.636	5.321
$\tau_d(s)$	1	1.206	1.426	1.084	1.452	1.688	1.597	1.380	1.461	1.265	1.391

The results from the validation phase, recorded in Table 6, provide a deeper insight into the generalization capability of each system identification method. Among the tested approaches, PSO demonstrates the best overall performance in the CL case, achieving an ITAE of 1.055644. This suggests that PSO benefits from the presence of a PID controller, allowing it to accurately model the system dynamics under control. GWO, on the other hand, performs best in the OL validation, achieving an ITAE of 0.525352, which is the lowest among all methods. This aligns with its strong ITAE results in Table 4 and indicates that GWO is the most effective when no feedback control is present. Interestingly, while PSO outperforms GWO in the CL case, GWO excels overall, as it maintains the lowest ITAE across both OL and CL scenarios.

Table 6: Comparison of ITAE values for PSO, GWO, BA, GA, LS, and NN using both open-loop and closed-loop system identification approaches during the validation phase based on the best results (i.e., lowest ITAE) obtained in Table 4

Algorithm	ITAE	
	Open-loop	Closed-loop
PSO	1.171853	1.055644
GWO	0.525352	1.058793
BA	5.560765	1.951218
GA	4.151888	1.891103
LS	4.217755	1.057431
NN	5.596660	1.722316

The LS method also shows strong CL performance (1.057431 ITAE), nearly matching PSO, reinforcing its effectiveness as a computationally efficient and reliable approach when a controller is available. However, its OL performance is notably weaker than GWO's, suggesting that LS may not generalize as well in uncontrolled

conditions. The metaheuristic algorithms BA and GA struggle in the OL validation phase, with significantly higher ITAE values (5.560765 and 4.151888, respectively). However, their performance improves drastically in the CL case, indicating that these methods rely heavily on the presence of a controller to enhance system identification accuracy. This highlights a key limitation of BA and GA-while they can achieve reasonable results in a controlled environment, their robustness is questionable when no feedback is available.

In contrast, NN performs the worst in both OL and CL validation, suggesting that it may require more extensive tuning, additional training data, or alternative architectures to perform optimally in this context. Its high ITAE values indicate a lack of generalization and possible overfitting to the training data.

Overall, the results suggest that while PSO is the best choice for CL system identification, GWO excels in OL scenarios and performs best overall due to its consistently low ITAE across both validation conditions. LS remains a competitive alternative due to its efficiency, but it may be less effective in OL settings. The findings emphasize that selecting the best system identification method depends on whether a controller is present, with PSO and GWO emerging as the most effective depending on the scenario.

8 Potential Research Directions

Motorized systems play crucial roles across diverse applications but are challenged by noise, load fluctuations, and environmental changes. Advancing system identification methods is essential to tackle these issues effectively. This section presents potential research directions aimed at improving the robustness, efficiency, and adaptability of system identification techniques for perturbed motorized systems. Each direction explores innovative approaches to advance the field and meet evolving application needs. The potential research directions include:

i. *Integration of Machine Learning Techniques* Investigate the integration of advanced machine learning techniques, such as deep learning and reinforcement learning, into system identification for perturbed motorized systems. Explore their potential in capturing complex dynamics and improving model accuracy under diverse perturbation scenarios.

ii. *Robust Model Predictive Control (MPC)*: Explore robust MPC strategies that incorporate detailed system models derived from advanced identification methods. Investigate how these models can enhance control performance by mitigating the effects of perturbations such as noise, load fluctuations, and parameter variations.

iii. *Multi-Objective Optimization*: Investigate multi-objective optimization approaches for system identification that consider conflicting objectives, such as model accuracy vs computational efficiency, robustness against perturbations, and adaptability to varying operational conditions.

iv. *Real-Time Implementation and Hardware-in-the-Loop Testing*: Focus on real-time implementation challenges of identified models and their validation through HIL testing. Explore methodologies to ensure model accuracy and robustness in practical motorized systems subjected to dynamic perturbations.

v. *Model Transferability and Generalization*: Investigate techniques to improve the transferability and generalization of identified models across different motor types, operating conditions, and environmental settings. Address challenges related to model adaptation and calibration in diverse applications.

vi. *Online Learning and Adaptive Control*: Explore online learning algorithms and adaptive control strategies that continuously update system models based on real-time sensor data. Investigate their effectiveness in maintaining optimal performance amidst varying perturbation intensities.

vii. *Fault Diagnosis and Condition Monitoring*: Extend system identification methodologies to incorporate fault diagnosis and condition monitoring capabilities. Develop integrated frameworks that utilize identified models for early fault detection.

viii. *Hybrid Learning-Based Methods*: A promising research direction is the development of hybrid learning-based methods that integrate physics-based models with machine learning techniques to enhance system identification of perturbed motorized systems. By combining first-principle equations with data-driven approaches such as NNs or neuro-fuzzy systems, future studies can improve the adaptability and accuracy of motor models, particularly in handling nonlinear perturbations like load variations, friction, and hysteresis. Additionally, exploring the use of metaheuristic-assisted deep learning for optimizing neural network parameters could further enhance robustness and generalization, making hybrid modeling a powerful tool for real-time motor control applications.

ix. *Adaptive Modeling Techniques*: Since PSO and GWO demonstrate complementary strengths in system identification-PSO performing better in CL settings and GWO excelling in OL scenarios-a promising research direction is the development of an adaptive modeling technique that dynamically integrates both methods. This approach could involve real-time switching between PSO and GWO based on system conditions, such as error trends, stability requirements, or perturbation levels. Alternatively, a hybrid optimization strategy could be designed to combine PSO's rapid convergence with GWO's strong exploration ability, enhancing overall robustness across varying operating conditions. Future research can explore intelligent adaptation mechanisms, such as reinforcement learning or fuzzy logic-based decision frameworks, to autonomously determine the optimal optimization method under different system states.

By exploring these avenues, researchers can contribute to enhancing the reliability, efficiency, and adaptability of motorized systems in various operational environments.

9 Conclusion

In conclusion, this review has explored various system identification methods for perturbed motorized systems, highlighting their applications, challenges, and recent advancements. The findings emphasize the importance of accurate modeling in understanding and mitigating perturbations such as noise, load variations, and environmental disturbances. This study conducted a comprehensive identification and statistical analysis of perturbed motorized systems using several widely used optimization algorithms, providing valuable insights into their comparative performance.

Furthermore, future research should explore innovative approaches to enhance system identification for perturbed motorized systems. The integration of machine learning techniques, such as deep learning and reinforcement learning, offers promising potential for capturing complex system dynamics and improving model accuracy across varying perturbation scenarios. Dynamic parameter adaptation strategies should also be investigated to improve the metaheuristic algorithms. This could involve hybridization, such as combining GWO's exploration strengths with PSO's exploitation efficiency or integrating machine learning-based adaptations for improved responsiveness. Additionally, robust MPC frameworks leveraging advanced system identification techniques could be explored to enhance control performance under noise, load fluctuations, and parameter variations.

Other key research directions include multi-objective optimization, where system identification balances competing factors like accuracy, computational cost, and robustness against perturbations. Real-time implementation challenges and HIL testing should also be addressed to validate identified models in dynamic environments. Model transferability and generalization across different motor types and conditions remain crucial for broader applicability. Further, online learning and adaptive control strategies could be developed to continuously update models based on real-time data, improving resilience against varying perturbation intensities. Finally, expanding system identification methodologies to include fault diagnosis and condition monitoring could provide early fault detection capabilities. By pursuing these research

directions, future studies can drive significant advancements in the accuracy, efficiency, and adaptability of system identification techniques for motorized systems.

Acknowledgement: The authors like to convey their sincere thanks to journal editor and reviewers for their constructive criticism, which enhanced the content and quality of our paper.

Funding Statement: This work was supported by the Malaysia Ministry of Higher Education under Fundamental Research Grant Scheme with Project Code: FRGS/1/2024/TK07/USM/02/3.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization: Helen Shin Huey Wee; Methodology: Helen Shin Huey Wee; Formal analysis: Helen Shin Huey Wee; Investigation: Helen Shin Huey Wee, Nur Syazreen Ahmad; Project administration: Nur Syazreen Ahmad; Supervision: Nur Syazreen Ahmad; Funding acquisition: Nur Syazreen Ahmad; Writing—original draft: Helen Shin Huey Wee; Writing—review & editing: Nur Syazreen Ahmad. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: All data generated and analyzed throughout the research process are given in the published article.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Abbreviations

Notation	Details
ABC	Artificial Bee Colony
AC	Alternating Current
ACO	Ant Colony Optimization
AIS	Artificial Immune System
AOA	Arithmetic Optimization Algorithm
ARX, NARX	AutoRegressive with eXogenous Input, Nonlinear ARX
ARMAX	AutoRegressive Moving Average Model with eXogenous Inputs
BA	Bat Algorithm
CD	Control Development
CGCRAO	Chaotic Gaussian-Cauchy RAO
CO	Cheetah Optimization
CS	Cuckoo Search
CPM	Characteristic Point Method
DC, DCS, BLDC, PMDC	Direct current, DC Servomotor, Brushless DC, Permanent Magnet DC
d (axis)	Direct
DE	Differential Evolution
DPSO-LS	Dynamic PSO with Learning Strategy
ELM	Extreme Learning Machine
EMF	Electromotive force
FELS	Forgetting Factor Least Squares
FMS	Flexible Manoeuvring System
FO	Fractional Order
FPA	Flower Pollination Algorithm
GA, MOGA	Genetic Algorithm, Multi-objective GA
GM	Generic Modeling

GPU	Graphics Processing Unit
GSA	Gravitational Search Algorithm
GWO	Grey Wolf Optimization
HEV	Hybrid Electric Vehicles
HHO	Harris Hawk Optimization
HPKM TR	Hybrid Parallel Kinematic Machines Tricept Robot
HPSOA	Hybrid Particle Swarm Optimization and Sine Cosine Algorithm
IAE	Integral of Absolute Error
JA	Jaya Algorithm
KF, EKF, DEKF, UKF	Kalman Filter, Extended KF, Dual EKF, Unscented KF
LM	Levenberg-Marquardt
LS, CLS, NLS, OLS, RLS	Least Squares, Compound LS, Nonlinear LS, Ordinary LS, Recursive LS
L-SHADE	Success-History based Adaptive Differential Evolution with Linear Population Size Reduction
MIMO	Multiple-Input Multiple-Output
MPC	Model Predictive Control
MRAS	Model Reference Adaptive System
NN, ANN, CNN, ENN, FFNN, RNN	Neural network, Artificial NN, Convolutional NN, Elman NN, Feed forward NN, Recursive NN
PI, PID	Proportional-integral, Proportional-integral-derivative
PMSM	Permanent Magnet Synchronous Motor
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-analyses
PS	Pattern Search
PSO, AIWF PSO, CI PSO, CIW PSO, STD PSO	Particle Swarm Optimization, Adaptive Inertia Weight Factor PSO, Constricted PSO, Chaos Initialized PSO, Standard PSO
PSys	Physiological System
q (axis)	Quadrature
RAMPA-TCF	Random Average Marine predators Algorithm with a Tunable Step-size Adaptive Coefficient
RM	Robot Manipulators
SAAPSO	Simulated Annealing Adaptive PSO
SC	Speed Control
SMO	Sliding Mode Observer
SVM	Support Vector Machine
TF	Transfer Function
TLBO	Teaching-learning-based Optimization
UAV	Unmanned Aerial Vehicles
UMLI	Underground Mine Lifting Installation
VSI	Voltage-Source Inverter
WC	Wheelchair
WMR	Wheeled Mobile Robots
WOA	Whale Optimization Algorithm

References

1. Singh A, Grant B, DeFour R, Sharma C, Bahadoorsingh S. A review of induction motor fault modeling. *Electr Power Syst Res.* 2016;133(4):191–7. doi:10.1016/j.epsr.2015.12.017.
2. Lau LB, Ahmad NS, Goh P. Self-balancing robot: modeling and comparative analysis between PID and linear quadratic regulator. *Int J Reconfig Embed Syst.* 2023;12(3):351–9. doi:10.11591/ijres.v12.i3.pp351-359.
3. Torres FJ, Guerrero GV, García CD, Zavala-Yoe R, García MA, López AR. Synchronization of robot manipulators actuated by induction motors with velocity estimator. *Comput Model Eng Sci.* 2019;121(2):609–30. doi:10.32604/cmes.2019.07153.
4. Qiang L, Tang HH, Ahmad NS. Improving trajectory tracking of differential wheeled mobile robots with enhanced GWO-optimized back-stepping and FOPID controllers. *IEEE Access.* 2025;13(10):48872–87. doi:10.1109/access.2025.3552312.
5. Rhinehart RR. Nonlinear model-based control: using first-principles models in process control. Research Triangle Park, NC, USA: International Society of Automation; 2024.
6. Kutz JN, Brunton SL, Brunton BW, Proctor JL. Dynamic mode decomposition: data-driven modeling of complex systems. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics; 2016.
7. Habib MK, Ayankoso SA, Nagata F. Data-driven modeling: concept, techniques, challenges and a case study. In: 2021 IEEE International Conference on Mechatronics and Automation; 2021 Aug 8–11; Takamatsu, Japan. p. 1000–7.
8. Pintelas E, Livieris IE, Pintelas P. A Grey-Box ensemble model exploiting black-box accuracy and white-box intrinsic interpretability. *Algorithms.* 2020;13(1):17. doi:10.3390/a13010017.
9. Song L, Shi JZ. Nonlinear Hammerstein model of ultrasonic motor for position control using differential evolution algorithm. *Ultrasonics.* 2019;94(3):20–7. doi:10.1016/j.ultras.2018.12.012.
10. Shi JZ, Zhao JP, Huang JT, Xu MY, Zhang L. Identification of ultrasonic motor's nonlinear hammerstein model. *J Control Autom Electr Syst.* 2014;25(5):537–46. doi:10.1007/s40313-014-0135-5.
11. Syed Mubarak Ali SAA, Ahmad NS, Goh P. Flex sensor compensator via hammerstein-wiener modeling approach for improved dynamic goniometry and constrained control of a bionic hand. *Sensors.* 2019;19(18):3896. doi:10.3390/s19183896.
12. Tang HH, Ahmad NS. Enhanced fuzzy logic control for active suspension systems via hybrid water wave and particle swarm optimization. *Int J Control Autom Syst.* 2025;23(2):560–71. doi:10.1007/s12555-024-0513-0.
13. Wu T, Wu X, Huang S, Lu K, Cui H. An optimized PLL with time delay and harmonic suppression for improved position estimation accuracy of PMSM based on Levenberg-Marquardt. *IEEE Trans Ind Electron.* 2023;70(10):9847–58. doi:10.1109/tie.2022.3219106.
14. Gad AG. Particle swarm optimization algorithm and its applications: a Systematic Review. *Arch Comput Methods Eng.* 2022;29(5):2531–61. doi:10.1007/s11831-021-09694-4.
15. Altay EV, Altay O, Özçevik Y. A comparative study of metaheuristic optimization algorithms for solving real-world engineering design problems. *Comput Model Eng Sci.* 2024;139(1):1039–94. doi:10.32604/cmes.2023.029404.
16. Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks; 1995 Nov 27–Dec 1; Perth, WA, Australia. p. 1942–8.
17. Mirjalili S, Lewis A. The whale optimization algorithm. *Adv Eng Softw.* 2016;95:51–67. doi:10.1016/j.advengsoft.2016.01.008.
18. Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. *Adv Eng Softw.* 2014;69:46–61. doi:10.1016/j.advengsoft.2013.12.007.
19. Özcan N, Utku S, Berber T. Artificial circulation system algorithm: a novel bio-inspired algorithm. *Comput Model Eng Sci.* 2025;142(1):635–63. doi:10.32604/cmes.2024.055860.
20. Hamadneh T, Batiha B, Alsayyed O, Werner F, Monrazeri Z, Dehghani M, et al. Using the novel wolverine optimization algorithm for solving engineering applications. *Comput Model Eng Sci.* 2024;141(3):2253–323. doi:10.32604/cmes.2024.055171.
21. Alzoubi S, Abualigah L, Sharaf M, Daoud MS, Khodadadi N, Jia H. Synergistic swarm optimization algorithm. *Comput Model Eng Sci.* 2024;139(3):2557–604. doi:10.32604/cmes.2023.045170.

22. Villarreal-Cervantes MG, Mezura-Montes E, Guzmán-Gaspar JY. Differential evolution based adaptation for the direct current motor velocity control parameters. *Math Comput Simul.* 2018;150(8):122–41. doi:10.1016/j.matcom.2018.03.007.
23. Chatterjee S, Naithani A, Mukherjee V. Small-signal stability analysis of DFIG based wind power system using teaching learning based optimization. *Int J Electr Power Energy Syst.* 2016;78(6):672–89. doi:10.1016/j.ijepes.2015.11.113.
24. Ainslie MD, Jiang Y, Xian W, Hong Z, Yuan W, Pei R, et al. Numerical analysis and finite element modelling of an HTS synchronous motor. *Physica C Supercond Appl.* 2010;470(20):1752–5. doi:10.1016/j.physc.2010.05.200.
25. Wang X, Wang Y, Wu T. The review of electromagnetic field modeling methods for permanent-magnet linear motors. *Energies.* 2022;15(10):3595. doi:10.3390/en15103595.
26. Kuczmam M. Review of DC motor modeling and linear control: theory with laboratory tests. *Electronics.* 2024;13(11):2225. doi:10.3390/electronics13112225.
27. Ahn H, Park H, Kim C, Lee H. A review of state-of-the-art techniques for PMSM parameter identification. *J Electr Eng Technol.* 2020;15(3):1177–87. doi:10.1007/s42835-020-00398-6.
28. Tang J, Yang Y, Blaabjerg F, Chen J, Diao L, Liu Z. Parameter identification of inverter-fed induction motors: a review. *Energies.* 2018;11(9):2194. doi:10.3390/en11092194.
29. Sengamalai U, Anbazhagan G, Thamizh Thentral TM, Vishnuram P, Khurshaid T, Kamel S. Three phase induction motor drive: a systematic review on dynamic modeling, parameter estimation, and control schemes. *Energies.* 2022;15(21):8260. doi:10.3390/en15218260.
30. Fazdi MF, Hsueh PW. Parameters identification of a permanent magnet DC motor: a review. *Electronics.* 2023;12(12):2559. doi:10.3390/electronics12122559.
31. Arifin B, Nugroho AA, Suprpto B, Prasetyowati SAD, Nawawi Z. Review of method for system identification on motors. In: 2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI); 2021 Oct 20–21; Semarang, Indonesia. p. 257–62.
32. Hughes A, Drury B. D.C. motors. In: Hughes A, Drury B, editors. *Electric motors and drives*. Newnes, NSW, Australia: Elsevier; 2019. p. 89–129. doi:10.1016/b978-0-08-102615-1.00003-9.
33. Cl Xia. *Permanent magnet brushless DC motor drives and controls*. Singapore: Wiley-Science Press; 2012.
34. Krishnan R. *Permanent magnet synchronous and brushless DC motor drives*. Boca Raton, FL, USA: CRC Press; 2017.
35. Ljung L. *System identification: theory for the user*. Prentice hall information and system sciences series. Upper Saddle River, NJ, USA: Prentice Hall PTR; 1999.
36. Mondal S, Mitra A, Chattopadhyay M. Mathematical modeling and simulation of brushless DC motor with ideal back EMF for a precision speed control. In: 2015 IEEE International Conference on Electrical, Computer and Communication Technologies; 2015 Mar 5–7; Coimbatore, India. p. 1–5.
37. Tang WJ, Liu ZT, Wang Q. DC motor speed control based on system identification and PID auto tuning. In: 2017 36th Chinese Control Conference (CCC); 2017 Jul 26–28; Dalian, China. p. 6420–3.
38. Semenov AS, Khubieva VM, Kharitonov YS. Mathematical modeling of static and dynamic modes DC motors in software package MATLAB. In: International Russian Automation Conference; 2018 Sep 9–16; Sochi, Russia. p. 1–5.
39. Wee HSH, Ahmad NS. Comparative analysis of data-driven models for DC motors with varying payloads. In: Proceedings of the 12th International Conference on Robotics, Vision, Signal Processing and Power Applications; 2024 Aug 28–29; Penang, Malaysia. p. 231–7.
40. Mokhlis Se, Sadki S, Bensassi B. System identification of a DC servo motor using ARX and ARMAX models. In: International Conference on Systems of Collaboration Big Data, Internet of Things and Security; 2019 Dec 12–13; Casablanca, Morocco. p. 1–4.
41. Li K, Luo H, Yang C, Yin S. Subspace-aided closed-loop system identification with application to DC motor system. *IEEE Trans Ind Electron.* 2020;67(3):2304–13. doi:10.1109/tie.2019.2907447.

42. Liu H, Zhu L, Pan Z, Bai F, Liu Y, Liu Y, et al. ARMAX-based transfer function model identification using wide-area measurement for adaptive and coordinated damping control. *IEEE Trans Smart Grid*. 2017;8(3):1105–15. doi:10.1109/tsg.2015.2470648.
43. Batool A, Nu Ain, Amin AA, Adnan M, Shahbaz MH. A comparative study of DC servo motor parameter estimation using various techniques. *Automatika*. 2022;63(2):303–12. doi:10.1080/00051144.2022.2036935.
44. Pandya V, Mehta S, Pramod P. Modeling, characterization, and identification of permanent magnet DC motors. In: *IEEE International Electric Machines and Drives Conference*; 2023 May 15–18; San Francisco, CA, USA. p. 1–7.
45. Lian C, Xiao F, Gao S, Liu J. Load torque and moment of inertia identification for permanent magnet synchronous motor drives based on sliding mode observer. *IEEE Trans Power Electron*. 2019;34(6):5675–83. doi:10.1109/tpe.2018.2870078.
46. Li X, Kennel R. Comparison of state-of-the-art estimators for electrical parameter identification of PMSM. In: *IEEE International Symposium on Predictive Control of Electrical Drives and Power Electronics*; 2019 May 31–Jun 2; Quanzhou, China. p. 1–6.
47. Wang Q, Wang G, Zhao N, Zhang G, Cui Q, Xu D. An impedance model-based multiparameter identification method of PMSM for both offline and online conditions. *IEEE Trans Power Electron*. 2021;36(1):727–38. doi:10.1109/tpe.2020.3000896.
48. Li X, Kennel R. General formulation of kalman-filter-based online parameter identification methods for VSI-Fed PMSM. *IEEE Trans Ind Electron*. 2021;68(4):2856–64. doi:10.1109/tie.2020.2977568.
49. Wang Z, Chai J, Xiang X, Sun X, Lu H. A novel online parameter identification algorithm designed for deadbeat current control of the permanent-magnet synchronous motor. *IEEE Trans Ind Appl*. 2022;58(2):2029–41. doi:10.1109/tia.2021.3136807.
50. Agoro S, Husain I. Robust deadbeat finite-set predictive current control with torque oscillation and noise reduction for PMSM drives. *IEEE Trans Ind Appl*. 2022;58(1):365–74. doi:10.1109/tia.2021.3130022.
51. Liu K, Hou C, Hua W. A novel inertia identification method and its application in PI controllers of PMSM drives. *IEEE Access*. 2019;7:13445–54. doi:10.1109/access.2019.2894342.
52. Szafranski G, Czyba R, Błachuta M. Modeling and identification of electric propulsion system for multirotor unmanned aerial vehicle design. In: *International Conference on Unmanned Aircraft Systems*; 2014 May 27–30; Orlando, FL, USA. p. 470–6.
53. Bombois X, Vanfretti L. Performance monitoring and redesign of power system stabilizers based on system identification techniques. *Sustain Energy Grids Netw*. 2024;38(8):101278. doi:10.1016/j.segan.2024.101278.
54. Moness M, Dīaa-Eldeen T. Experimental black-box dynamic modelling of a flexible manoeuvring system. In: *12th International Conference on Computer Engineering and Systems*; 2017 Dec 19–20; Cairo, Egypt. p. 259–65.
55. Trombetta EI, Du Mutel de Pierrepont Frauzetti ID, Carminati D, Scanavino M, Capello E. Data-driven identification method and simulation modeling of a ground robot. In: *IEEE 8th International Workshop on Metrology for AeroSpace*; 2021 Jun 23–25; Naples, Italy. p. 602–7. doi:10.1109/metroaerospace51421.2021.9511747.
56. Salehi A, Montazeri-Gh M. Black box modeling of a turboshaft gas turbine engine fuel control unit based on neural NARX. *Proc Inst Mech Eng M J Eng Marit Environ*. 2018;233(3):949–56. doi:10.1177/1475090218797496.
57. Nievergelt Y. Total least squares: state-of-the-art regression in numerical analysis. *SIAM Rev*. 1994;36(2):258–64. doi:10.1137/1036055.
58. El-Saadawi M, Gouda E, Elhosseini M, Essa M. Identification and speed control of DC motor using fractional order PID: microcontroller. *Eur J Electr Eng Comput Sci*. 2020;4(1):1–8. doi:10.24018/ejece.2020.4.1.170.
59. Tolun OC, Tutsoy O. Estimation of DC motor parameters using least square-based optimization algorithm. In: *2023 Innovations in Intelligent Systems and Applications Conference*; 2023 Oct 11–13; Sivas, Türkiye. p. 1–7.
60. Xiang C, Wang X, Ma Y, Xu B. Practical modeling and comprehensive system identification of a BLDC motor. *Math Probl Eng*. 2015;2015:79581. doi:10.1155/2015/879581.
61. Sands T. Nonlinear-adaptive mathematical system identification. *Computation*. 2017;5(4):47. doi:10.3390/computation5040047.
62. Hayes MH. Statistical digital signal processing and modeling. 1st ed. Montreal, QC, Canada: Wiley; 1996.

63. Brablc M, Sova V, Grepl R. Adaptive feedforward controller for a DC motor drive based on inverse dynamic model with recursive least squares parameter estimation. In: 17th International Conference on Mechatronics-Mechatronika; 2016 Dec 7–9; Prague, Czech Republic. p. 1–5.
64. Siwek M, Panasiuk J, Baranowski L, Kaczmarek W, Prusaczyk P, Borys S. Identification of differential drive robot dynamic model parameters. *Materials*. 2023;16(2):683. doi:10.3390/ma16020683.
65. Sun P, Ge Q, Zhang B, Wang X. Sensorless control technique of PMSM based on RLS on-line parameter identification. In: 21st International Conference on Electrical Machines and Systems; 2018 Oct 7–10; Jeju, South Korea. p. 1670–3.
66. Deng W, Xia C, Yan Y, Geng Q, Shi T. Online multiparameter identification of surface-mounted PMSM considering inverter disturbance voltage. *IEEE Trans Energy Convers*. 2017;32(1):202–12. doi:10.1109/tec.2016.2621130.
67. Hanke S, Peitz S, Wallscheid O, Böcker J, Dellnitz M. Finite-control-set model predictive control for a permanent magnet synchronous motor application with online least squares system identification. In: IEEE International Symposium on Predictive Control of Electrical Drives and Power Electronics; 2019 May 31–Jun 2; Quanzhou, China. p. 1–6.
68. Alkamachi A. Permanent magnet DC motor (PMDC) model identification and controller design. *J Electr Eng*. 2019;70(4):303–9. doi:10.2478/je-2019-0060.
69. Weerts HHM, Galrinho M, Bottegal G, Hjalmarsson H, den Hof PMJV. A sequential least squares algorithm for ARMAX dynamic network identification. *IFAC-PapersOnLine*. 2018;51(15):844–9. doi:10.1016/j.ifacol.2018.09.119.
70. Nguyen DT. Parametric identification of electric drives using the ordinary least squares method. In: IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering; 2021 Jan 26–29; St. Petersburg, Moscow, Russia. p. 2640–4.
71. Palpacelli M, Palmieri G, Carbonari L, Callegari M. Experimental identification of the static model of the HPKM tricept industrial robot. *Adv Robot*. 2014;28(19):1291–304. doi:10.1080/01691864.2014.933127.
72. Arshad S, Qamar S, Jabbar T, Malik A. Parameter estimation of a DC motor using ordinary least squares and recursive least squares algorithms. In: Proceedings of the 8th International Conference on Frontiers of Information Technology; 2010 Dec 21–23; Islamabad, Pakistan. p. 1–5.
73. Zeng R, Kang Y, Yang J, Zhang W, Wu Q. Terrain parameters identification of kinematic and dynamic models for a tracked mobile robot. In: 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference; 2018 May 25–27; Xi'an, China. p. 575–82.
74. Li M, Ma Y. Parameter identification of DC motor based on compound least square method. In: IEEE 5th Information Technology and Mechatronics Engineering Conference; 2020 Jun 12–14; Chongqing, China. p. 1107–11.
75. Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: Proceedings of the 6th International Symposium on Micro Machine and Human Science; 1995 Oct 4–6; Nagoya, Japan. p. 39–43.
76. Jain M, Saihpal V, Singh N, Singh SB. An overview of variants and advancements of PSO Algorithm. *Appl Sci*. 2022;12(17):8392. doi:10.3390/app12178392.
77. Freitas D, Lopes LG, Morgado-Dias F. Particle swarm optimisation: a historical review up to the current developments. *Entropy*. 2020;22(3):362. doi:10.3390/e22030362.
78. Marini F, Walczak B. Particle swarm optimization (PSO). A tutorial. *Chemom Intell Lab Syst*. 2015;149(Part B):153–65. doi:10.1016/j.chemolab.2015.08.020.
79. Yassin IM, Taib MN, Adnan R, Salleh MKM, Hamzah MK. Effect of swarm size parameter on binary particle swarm optimization-based NARX structure selection. In: IEEE Symposium on Industrial Electronics and Applications; 2012 Sept 23–26; Bandung, Indonesia. p. 219–23.
80. Mohamad MSA, Yassin IM, Zabidi A, Taib MN, Adnan R. Comparison between PSO and OLS for NARX parameter estimation of a DC motor. In: IEEE Symposium on Industrial Electronics & Applications; 2013 Sept 22–25; Kuching, Malaysia. p. 27–32.
81. Liu ZH, Li XH, Wu LH, Zhou SW, Liu K. GPU-accelerated parallel coevolutionary algorithm for parameters identification and temperature monitoring in permanent magnet synchronous machines. *IEEE Trans Ind Inform*. 2015;11(5):1220–30. doi:10.1109/tii.2015.2424073.

82. Kiree C, Kumpanya D, Tunyasrirut S, Puangdownreong D. PSO-based optimal PI(D) controller design for brushless DC motor speed control with back EMF detection. *J Electr Eng Technol.* 2016;11(3):715–23. doi:10.5370/jeet.2016.11.3.715.
83. Polsena A, Kongjeen Y, Watcharakhup S. Identifying parameter and PI tuning of DC motor for predict behavior based on PSO. In: 9th International Electrical Engineering Congress; 2021 Mar 10–12; Pattaya, Thailand. p. 97–100. doi:10.1109/ieecon51072.2021.9440341.
84. Hafez I, Dhaouadi R. Parameter identification of DC motor drive systems using particle swarm optimization. In: International Conference on Engineering and Emerging Technologies; 2021 Oct 27–28; Istanbul, Turkey. p. 1–6.
85. Kanojiya RG, Meshram PM. Optimal tuning of PI controller for speed control of DC motor drive using particle swarm optimization. In: International Conference on Advances in Power Conversion and Energy Technologies; 2012 Aug 2–14; Mylavaram, India. p. 1–6.
86. Abedinifar M, Ertugrul S, Arguz SH. Nonlinear model identification and statistical verification using experimental data with a case study of the UR5 manipulator joint parameters. *Robotica.* 2023;41(4):1348–70. doi:10.1017/s0263574722001783.
87. Sandre-Hernandez O, Morales-Caporal R, Rangel-Magdaleno J, Peregrina-Barreto H, Hernandez-Perez JN. Parameter identification of PMSMs using experimental measurements and a PSO algorithm. *IEEE Trans Instrum Meas.* 2015;64(8):2146–54. doi:10.1109/tim.2015.2390958.
88. Sankardoss V, Geethanjali P. PMDC motor parameter estimation using bio-inspired optimization algorithms. *IEEE Access.* 2017;5:11244–54. doi:10.1109/access.2017.2679743.
89. Liu ZH, Wei HL, Zhong QC, Liu K, Li XH. GPU implementation of DPSO-RE algorithm for parameters identification of surface PMSM considering VSI nonlinearity. *IEEE J Emerg Sel Top Power Electron.* 2017;5(3):1334–45. doi:10.1109/jestpe.2017.2690688.
90. Liu ZH, Wei HL, Zhong QC, Liu K, Xiao XS, Wu LH. Parameter estimation for VSI-fed PMSM based on a dynamic PSO with learning strategies. *IEEE Trans Power Electron.* 2017;32(4):3154–65. doi:10.1109/tpe.2016.2572186.
91. Liu ZH, Wei HL, Li XH, Liu K, Zhong QC. Global identification of electrical and mechanical parameters in PMSM drive based on dynamic self-learning PSO. *IEEE Trans Power Electron.* 2018;33(12):10858–71. doi:10.1109/tpe.2018.2801331.
92. Amiri MS, Ibrahim MF, Ramli RB. Optimal parameter estimation for a DC motor using genetic algorithm. *Int J Power Electron Drive Syst.* 2020;11(2):1047–54. doi:10.11591/ijpeds.v11.i2.pp1047-1054.
93. Sankardoss V, Geethanjali P. Parameter estimation and speed control of a PMDC motor used in wheelchair. *Energy Proc.* 2017;117:345–52. doi:10.1016/j.egypro.2017.05.142.
94. Avdeev A, Osipov O. PMSM identification using genetic algorithm. In: 26th International Workshop on Electric Drives: Improvement in Efficiency of Electric Drives; 2019 Jan 30–Feb 2; Moscow, Russia. p. 1–4. doi:10.1109/iwed.2019.8664250.
95. Montazeri A, West C, Monk SD, Taylor CJ. Dynamic modelling and parameter estimation of a hydraulic robot manipulator using a multi-objective genetic algorithm. *Int J Control.* 2017;90(4):661–83. doi:10.1080/00207179.2016.1230231.
96. Chakkarapani K, Thangavelu T, Dharmalingam K, Thandavarayan P. Multiobjective design optimization and analysis of magnetic flux distribution for slotless permanent magnet brushless DC motor using evolutionary algorithms. *J Magn Magn Mater.* 2019;476(2019):524–37. doi:10.1016/j.jmmm.2019.01.029.
97. Lin CJ, Yau HT, Tian YC. Identification and compensation of nonlinear friction characteristics and precision control for a linear motor stage. *IEEE/ASME Trans Mechatronics.* 2013;18(4):1385–96. doi:10.1109/tmech.2012.2202679.
98. Karnopp DC. Computer simulation of stick-slip friction in mechanical dynamic systems. *J Dyn Syst Meas Control.* 1985;107(1):100–3. doi:10.1115/1.3140698.
99. Clerc M, Kennedy J. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans Evol Comput.* 2002;6(1):58–73. doi:10.1109/4235.985692.
100. Shah P, Sekhar R. Closed loop system identification of a DC motor using fractional order model. In: International Conference on Mechatronics, Robotics and Systems Engineering; 2019 Dec 4–6; Bali, Indonesia. p. 69–74.

101. Jesenik M, Hamler A, Trbušić M, Trlep M. The use of evolutionary methods for the determination of a DC motor and drive parameters based on the current and angular speed response. *Mathematics*. 2020;8(8):1269. doi:10.3390/math8081269.
102. Rodríguez-Abreo O, Rodríguez-Reséndiz J, Álvarez Alvarado JM, García-Cerezo A. Metaheuristic parameter identification of motors using dynamic response relations. *Sensors*. 2022;22(11):4050. doi:10.3390/s22114050.
103. Pawlowski A, Ciekowski M, Romaniuk S, Kulesza Z. GWO-based multi-stage algorithm for PMDC motor parameter estimation. *Sensors*. 2023;23(11):5047. doi:10.3390/s23115047.
104. Munciño DM, Damian-Ramírez EA, Cruz-Fernández M, Montoya-Santianes LA, Rodríguez-Reséndiz J. Metaheuristic and heuristic algorithms-based identification parameters of a direct current motor. *Algorithms*. 2024;17(5):209. doi:10.3390/a17050209.
105. Yang XS. In: González JR, Pelta DA, Cruz C, Terrazas G, Krasnogor N, editors. A new metaheuristic bat-inspired algorithm. Berlin/Heidelberg, Germany: Springer; 2010. p. 65–74.
106. Bora TC, Lds Coelho, Lebensztajn L. Bat-inspired optimization approach for the brushless DC wheel motor problem. *IEEE Trans Magn*. 2012;48(2):947–50. doi:10.1109/tmag.2011.2176108.
107. Tabish M, Kalam A, Zayegh A. Robot DC servo motor parameters estimation in a closed loop using BAT optimisation algorithm. In: International Conference on Electrical, Communication, and Computer Engineering; 2019 Jul 24–15; Swat, Pakistan. p. 1–5.
108. De León CLCD, Vergara-Limón S, Vargas-Treviño MAD, López-Gómez J, Gonzalez-Calleros JM, González-Arriaga DM, et al. Parameter identification of a robot arm manipulator based on a convolutional neural network. *IEEE Access*. 2022;10:55002–19. doi:10.1109/access.2022.3177209.
109. Bramerndorfer G, Winkler SM, Kommenda M, Weidenholzer G, Silber S, Kronberger G, et al. Using FE calculations and data-based system identification techniques to model the nonlinear behavior of PMSMs. *IEEE Trans Ind Electron*. 2014;61(11):6454–62. doi:10.1109/tie.2014.2303785.
110. Munikoti S, Das L, Natarajan B, Srinivasan B. Data-driven approaches for diagnosis of incipient faults in DC motors. *IEEE Trans Ind Inform*. 2019;15(9):5299–308. doi:10.1109/tii.2019.2895132.
111. Ismeal GA, Kyslan K, Fedák V. DC motor identification based on recurrent neural networks. In: Proceedings of the 16th International Conference on Mechatronics-Mechatronika; 2014 Dec 3–5; Brno, Czech Republic. p. 701–5.
112. Çetin O, Dalcı A, Temurtaş F. A comparative study on parameters estimation of squirrel cage induction motors using neural networks with unmemorized training. *Eng Sci Technol Int J*. 2020;23(5):1126–33. doi:10.1016/j.jestch.2020.03.011.
113. Sabanci K. Artificial intelligence based power consumption estimation of two-phase brushless DC motor according to FEA parametric simulation. *Measurement*. 2020;155:107553. doi:10.1016/j.measurement.2020.107553.
114. Srivastava A, Das DK, Rai A, Raj R. Parameter estimation of a permanent magnet synchronous motor using whale optimization algorithm. In: Recent Advances on Engineering, Technology and Computational Sciences (RAETCS); 2018 Feb 6–8; Allahabad, India. p. 1–6. doi:10.1109/raetcs.2018.8443839.
115. Villarreal-Cervantes MG, Rodríguez-Molina A, García-Mendoza CV, Peñaloza-Mejía O, Sepúlveda-Cervantes G. Multi-objective on-line optimization approach for the DC motor controller tuning using differential evolution. *IEEE Access*. 2017;5:20393–407. doi:10.1109/access.2017.2757959.
116. Puangdownreong D, Hlungnamtip S, Thammarat C, Nawikavatan A. Application of flower pollination algorithm to parameter identification of DC motor model. In: International Electrical Engineering Congress; 2017 Mar 8–10; Pattaya, Thailand. p. 1–4.
117. Rodríguez-Abreo O, Hernandez-Paredes JM, Rangel AF, Fuentes-Silva C, Velásquez FAC. Parameter identification of motors by cuckoo search using steady-state relations. *IEEE Access*. 2021;9:72017–24. doi:10.1109/access.2021.3078578.
118. Li H, Jian X. Parameter identification of permanent magnet synchronous motor based on CGCRAO algorithm. *IEEE Access*. 2023;11(5):124319–30. doi:10.1109/access.2023.3330495.
119. Rao RV, Pawar RB. Self-adaptive multi-population rao algorithms for engineering design optimization. *Appl Artif Intell*. 2020;34(3):187–250. doi:10.1080/08839514.2020.1712789.

120. Tumari MZM, Ahmad MA, Mohamed Z. Identification of the continuous-time hammerstein models with sparse measurement data using improved marine predators algorithm. *Arab J Sci Eng.* 2024;358:4546. doi:10.1007/s13369-024-09692-1.
121. Chuandong Qin BH. Multi-stage improvement of marine predators algorithm and its application. *Comput Model Eng Sci.* 2023;136(3):3097–119. doi:10.32604/cmesci.2023.026643.
122. Issa M. Enhanced arithmetic optimization algorithm for parameter estimation of PID controller. *Arab J Sci Eng.* 2023;48(2):2191–205. doi:10.1007/s13369-022-07136-2.
123. Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH. The arithmetic optimization algorithm. *Comput Methods Appl Mech Eng.* 2021;376(2021):113609. doi:10.1016/j.cma.2020.113609.
124. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H. Harris hawks optimization: algorithm and applications. *Future Gener Comput Syst.* 2019;97(2019):849–72. doi:10.1016/j.future.2019.02.028.
125. Vukašinović J, Milovanović M, Arsić N, Radosavljević J, Štatkić S. Parameters estimation of double-cage induction motors using a hybrid metaheuristic algorithm. In: 21st International Symposium INFOTEH-JAHORINA; 2022 Mar 16–18; East Sarajevo, Bosnia and Herzegovina. p. 1–6.
126. Loganathan A, Ahmad NS. A hybrid HHO-AVOA for path planning of a differential wheeled mobile robot in static and dynamic environments. *IEEE Access.* 2024;12(4):25967–79. doi:10.1109/access.2024.3367115.
127. Ahmad NS. Modeling and hybrid PSO-WOA-based intelligent PID and state-feedback control for ball and beam systems. *IEEE Access.* 2023;11(9):137866–80. doi:10.1109/access.2023.3339879.
128. Najm HT, Ahmad NS, Al-Araji AS. Enhanced path planning algorithm via hybrid WOA-PSO for differential wheeled mobile robots. *Syst Sci Control Eng.* 2024;12(1):2334301. doi:10.1080/21642583.2024.2334301.
129. Santana NHB, Yahyaoui I, Oliveira FDC, Amorim AEA, Simonetti DSL, Rocha HRO. Dynamic to static model comparison and hybrid metaheuristic optimization in induction motor parameter estimation. *Electronics.* 2025;14(3):524. doi:10.3390/electronics14030524.
130. Huang SD, Lin Z, Cao GZ, Liu N, Mou H, Xu J. Nonlinear dynamic model-based position control parameter optimization method of planar switched reluctance motors. *Mathematics.* 2023;11(19):4067. doi:10.3390/math11194067.
131. Seyyedabbasi A, Tareq WZ, Bacanin N. An effective hybrid metaheuristic algorithm for solving global optimization algorithms. *Multimed Tools Appl.* 2024;83(37):85103–38. doi:10.1007/s11042-024-19437-9.