Computer Modeling in Engineering & Sciences

Tech Science Press

https://doi.org/10.32604/cmes.2026.080428

ARTICLE

Enhanced-WOA Optimized FOPID Controller for Energy-Efficient Path-Tracking Robot

Hooi Hung Tang, Te Meng Ting and Nur Syazreen Ahmad*

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

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

Received: 09 February 2026; Accepted: 27 May 2026; Published: 30 June 2026

ABSTRACT: In industrial and service robotics, autonomous mobile robots must achieve accurate trajectory tracking while maintaining low energy consumption to avoid frequent recharging and performance degradation. Energy efficiency is particularly critical because locomotion accounts for 45%–65% of total power consumption, directly limiting operational range and autonomy. This paper proposes an energy-aware trajectory tracking framework that optimizes a fractional-order proportional-integral-derivative (FOPID) controller using an Enhanced Whale Optimization Algorithm (E-WOA). The key contributions are threefold: (1) the E-WOA hybridizes Differential Evolution (DE)'s global exploration with WOA's local exploitation to overcome premature convergence in high-dimensional FOPID parameter spaces; (2) a composite fitness function jointly minimizes tracking error and energy consumption; and (3) controller parameters optimized on a single circular trajectory generalize effectively to complex paths without retuning. The proposed framework is evaluated on multiple trajectory configurations, including circular, eight-shaped, square, and rhombus paths, under stochastic environmental disturbances. Statistical analysis based on ten independent runs and validated using Analysis of Variance (ANOVA) was conducted against six classical and recent optimization methods: DE, WOA, Particle Swarm Optimization (PSO), Bat Algorithm (BA), Artificial Hummingbird Algorithm (AHA), and Rime Optimization Algorithm (RIME). The results show that E-WOA-FOPID achieves an average improvement of 15.14% in mean fitness and a 30.69% reduction in performance variability compared to the best-performing benchmark. These findings confirm its robustness as an energy-efficient solution for high-precision mobile robot trajectory tracking.

KEYWORDS: Energy efficiency; fractional order; mobile robots; tracking accuracy

1 Introduction

Mobile robot path tracking systems face the dual challenge of maintaining high trajectory accuracy while minimizing energy consumption, a critical consideration in both industrial and service applications where operational efficiency and battery longevity are paramount. Autonomous mobile robots deployed in logistics, manufacturing, and service domains require control strategies that not only guarantee precise trajectory tracking but also mitigate excessive energy usage. Existing studies indicate that locomotion alone contributes approximately 45%–65% of a mobile robot's total power consumption, highlighting the necessity of energy-aware trajectory optimization to extend operational duration, reduce downtime due to recharging, and lower overall operating costs [1,2].

The importance of energy efficiency in mobile robot trajectory tracking extends beyond mere battery conservation. In warehouse automation and logistics, robots must complete hundreds of pick-and-place cycles per shift; even modest reductions in per-cycle energy consumption translate into significantly

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

extended operational windows and reduced fleet sizes [2]. In service robotics, including hospital delivery and elderly care applications, unplanned recharging interruptions can compromise time-critical tasks and degrade service quality. Furthermore, excessive control effort characterized by aggressive actuator commands and rapid velocity fluctuations—not only wastes electrical energy but also accelerates mechanical wear on motors, gears, and wheels, thereby increasing maintenance costs and reducing robot lifespan [1]. Consequently, an energy-aware control framework that jointly optimizes tracking accuracy and energy consumption is essential for achieving sustainable, long-duration autonomous operation.

Trajectory tracking of mobile robots, particularly those with non-holonomic constraints such as differential-drive platforms, remains a challenging problem especially when the paths involve curves or sudden changes in direction [3]. To address these challenges, various control strategies have been proposed in the literature, including linear quadratic regulation (LQR), model predictive control (MPC), and sliding mode control (SMC). For instance, Aguiar et al. [4] applied LQR techniques in structured environments, but performance deteriorated when the system deviated from linear assumptions or encountered unstructured obstacles. Cao et al. [5] employed SMC and reduced chattering using a second-order formulation, yet the approach remained sensitive to measurement noise and required precise knowledge of system bounds. Yu et al. [6] demonstrated the effectiveness of MPC on soft terrain, but the method demanded significant computational resources and assumed highly accurate models, which are often impractical in real-world deployments.

Recent surveys have further highlighted the evolving challenges in autonomous mobile robotics. For instance, a comprehensive review on mobile robot localization in [7] identifies persistent difficulties in sensor fusion, dynamic environment adaptation, and computational constraints that directly affect trajectory tracking reliability. In parallel, recent studies on energy-efficient robot operation have emphasized that adaptive control strategies can significantly reduce energy consumption while maintaining reliable autonomous performance, particularly in mobile robotic systems operating in complex and dynamic environments [8]. These developments highlight the need for control architectures that simultaneously address tracking accuracy, robustness, computational efficiency, and energy consumption. Consequently, considerable research effort has been devoted to the development and tuning of advanced controllers capable of achieving these objectives under diverse operating conditions.

Proportional-Integral-Derivative (PID) controllers, by contrast, are simple, computationally efficient, and widely used in industrial mobile robot applications. However, their main limitations lie in poor adaptability to nonlinearities, restricted robustness against disturbances, and difficulty in balancing transient and steady-state performance across varying operating conditions. To address these issues, the Fractional-Order PID (FOPID) controller has been introduced as an extension of PID [9,10]. By incorporating two additional parameters-the fractional orders of integration (λ) and differentiation (μ)-FOPID offers enhanced flexibility in shaping system dynamics, enabling improved robustness, better disturbance rejection, and superior trajectory tracking performance compared to classical PID. Nonetheless, this improvement comes at the cost of increased complexity, as tuning five parameters instead of three is required [11].

The fundamental advantage of FOPID over classical PID lies in the continuous nature of the fractional orders λ and μ, which allow fine-grained shaping of the controller's frequency response. Specifically, the fractional integral order λ governs the low-frequency gain roll-off, enabling more precise steady-state error elimination without the excessive overshoot that integer-order integration can produce. The fractional derivative order μ controls the high-frequency phase contribution, providing smoother anticipatory control action that reduces abrupt actuator commands and, consequently, lowers energy consumption associated with aggressive corrections. Together, these two additional degrees of freedom enable the FOPID controller to simultaneously achieve tighter tracking accuracy and gentler control effort—a combination that is

inherently difficult to realize with the three-parameter PID structure. This dual benefit is particularly pronounced for mobile robot trajectory tracking, where both precision and energy efficiency are critical performance objectives.

To optimize PID-based control parameters, researchers have increasingly turned to metaheuristic algorithms due to their ability to handle complex, nonlinear, and multi-objective optimization problems that traditional tuning methods often fail to address [12]. Techniques such as Genetic Algorithms (GA) [13], Particle Swarm Optimization (PSO) [14,15], Ant Colony Optimization (ACO) [16], Bat Algorithm (BA) [17], and Grey Wolf Optimizer (GWO) [18] have been widely applied to robotic trajectory tracking, where the cost function typically emphasizes position error minimization. More recent bio- and physics-inspired methods such as the Artificial Hummingbird Algorithm (AHA) [19], which models the guided, territorial, and migration foraging behaviors of hummingbirds, and the Rime Optimization Algorithm (RIME) [20], which simulates soft-rime and hard-rime puncture mechanisms in ice formation, have also demonstrated competitive performance on engineering optimization problems and are increasingly being adopted for controller tuning. These approaches have shown significant promise in improving tracking accuracy compared to conventional tuning methods, thereby enhancing the robustness of robotic motion control.

Among these, the Whale Optimization Algorithm (WOA) [21] has gained attention as a competitive alternative, owing to its strong local exploitation capabilities and adaptive search dynamics inspired by the bubble-net feeding strategy of humpback whales. Empirical studies have demonstrated that WOA often outperforms GA, PSO, and GWO in PID-based control tuning problems [22–24], achieving superior convergence rates and reduced steady-state errors. Despite these advances, most applications of WOA remain centered on trajectory tracking accuracy, with energy consumption either overlooked or treated only as a secondary factor. The explicit inclusion of energy efficiency in optimization frameworks is still limited, despite its critical importance for mobile robots where battery longevity and operational sustainability are as vital as tracking precision.

Differential Evolution (DE), by contrast, is well recognized for its robust global exploration through mutation and crossover. It has been successfully employed in PID tuning for robotic manipulators and energy systems, often surpassing other techniques in convergence speed and optimization accuracy [25,26]. Building on this, hybrid metaheuristics have attracted growing interest as a means of overcoming the limitations of individual algorithms by striking a balance between global exploration and local exploitation [27–29]. While many hybrids have been proposed, they are often tailored to narrow domains or restricted cost functions, limiting their broader applicability.

Motivated by these gaps, this study tackles the practical challenge of energy-efficient path tracking for differential-drive mobile robots, where unique system dynamics and strict energy constraints require optimization strategies that jointly ensure high tracking accuracy and energy-aware performance. To address this, we propose an Enhanced Whale Optimization Algorithm (E-WOA) for FOPID tuning within a novel energy-efficient path tracking framework. The E-WOA integrates DE's strong global exploration with WOA's effective local refinement, yielding a hybrid strategy explicitly designed to balance trajectory accuracy with energy consumption. The rationale for this specific hybridization is grounded in the complementary failure modes of the two algorithms. WOA's spiral update and encircling mechanisms are highly effective at converging toward promising solutions but rely heavily on the current best solution, making it prone to premature convergence in the five-dimensional FOPID parameter space where multiple local optima exist. DE, conversely, maintains population diversity through its difference-vector mutation operator but can exhibit slow convergence during later optimization stages when fine-tuning is needed. By embedding DE's mutation and crossover operations as the primary strategy during early iterations (with approximately 80% probability) and gradually transitioning to WOA's exploitation mechanisms in later iterations, E-WOA

preserves broad search coverage when the landscape is unexplored while leveraging WOA's efficient local refinement once promising regions are identified. This dynamic probability-based switching overcomes the static exploration-exploitation balance of standalone algorithms and is particularly well suited to the FOPID tuning problem, where the two additional fractional-order parameters (λ, μ) expand the search space and increase the risk of entrapment in suboptimal solutions.

In contrast to prior studies that primarily evaluate the energy consumption of existing control methods [2,17,30,31], this work develops and applies E-WOA as a new optimization approach for both PID and FOPID controllers. A multi-objective cost function is formulated to directly incorporate both tracking error and energy efficiency, ensuring balanced performance across competing objectives. The proposed framework is evaluated on four trajectory types—circular, lemniscate, square, and rhombus—to assess robustness and parameter transferability under different geometric complexities.

The main contributions of this study are threefold. First, the E-WOA is developed by embedding adaptive DE strategies within the conventional WOA framework and by formulating a composite objective function that jointly minimizes trajectory tracking error and energy consumption. Second, the proposed optimization framework is applied to a FOPID controller, demonstrating that the E–WOA-tuned FOPID consistently outperforms its integer-order PID counterpart as well as controllers optimized using a comprehensive set of benchmark algorithms spanning classical, energy-aware, and recent state-of-the-art methods, namely DE, WOA, PSO, BA, AHA, and RIME-among which PSO and BA have shown promise in reducing mobile robot energy consumption [15,17], while AHA and RIME represent the latest generation of bio- and physics-inspired optimizers. Third, comprehensive simulation studies incorporating stochastic environmental disturbances confirm that the proposed approach achieves significant improvements in tracking accuracy, energy efficiency, and robustness, while maintaining strong parameter generalization from simple circular trajectories to more complex path geometries. These results highlight the practical effectiveness and computational efficiency of the E–WOA-based FOPID framework for autonomous mobile robots in industrial and service applications.

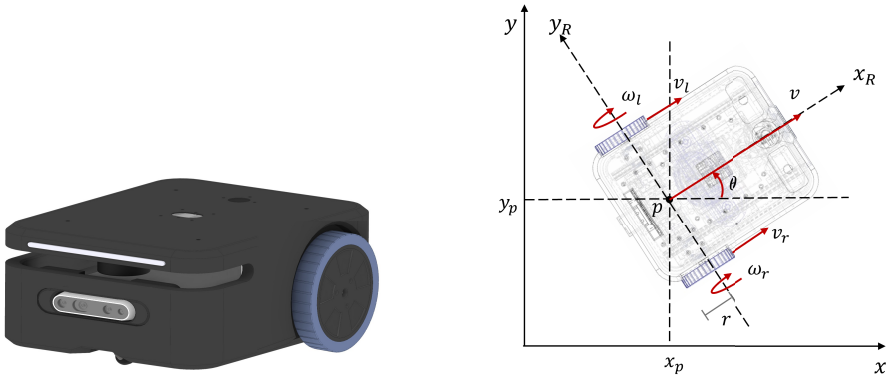
The remainder of this paper is organized as follows. Section 2 describes the system modeling and methodology, including the robot kinematic model, stochastic disturbance formulation, motor and energy models, control architecture, and the proposed E–WOA optimization strategy, together with the adopted performance evaluation criteria. Section 3 presents and discusses the simulation results, while Section 4 concludes the paper and outlines directions for future research.

2 Methodology

2.1 Robot Model

The experimental platform built for this study is a differential wheeled mobile robot (DWMR), as shown in Fig. 1. Fig. 1a presents an overview of the robot model developed in this work, while Fig. 1b illustrates its geometry. The robot operates on a two-wheel differential drive mechanism, where the independent angular velocities of the left and right wheels (ω_l and ω_r) determine both translational and rotational motion. The corresponding linear velocities of the wheels are denoted by v_l and v_r respectively. A passive castor wheel is mounted at the front of the chassis to provide balance and stability while allowing free rotation without constraining the robot's motion. The kinematic structure is illustrated in Fig. 1b, where the robot pose is expressed in the global coordinate system (x_p, y_p) with orientation θ. The point p represents the midpoint of the axis connecting the two drive wheels. The overall linear velocity of the mobile robot, denoted by v, is aligned with the robot's local reference frame (x_R, y_R). The dynamics of the differential-drive robot were

derived using the system parameters listed in Table 1. These physical attributes form the basis for estimating both the control effort and the energy expenditure during path tracking.



(**a**) Overview of the differential wheeled mobile robot. (**b**) Illustration on the geometry of the mobile robot.

Figure 1: The differential wheeled mobile robot built for this study.

Table 1: Mobile robot system parameters.

Parameter	Symbol	Value	Unit
Total Mass	m	9	kg
Moment of Inertia	J	0.16	kg·m^2
Wheel Radius	r	0.0925	m
Motor Torque Constant	K_t	0.009	N·m/A
Motor Electrical Resistance	R_e	0.09	Ω
Height of Robot CoM	h	0.1	m
Friction Coefficient	f_{coef}	0.04	–
Rolling Friction Coefficient	μ	0.02	–
Motor Efficiency	η	0.85	–

The robot's motion is described by standard differential drive kinematics [32], which capture the fundamental motion constraints of DDMRs and provide the basis for control system design. The model is expressed as

$$\dot{x} = \dot{x}_p = v\cos\theta \tag{1}$$

$$\dot{y} = \dot{y}_p = v\sin\theta \tag{2}$$

$$\dot{\theta} = \omega \tag{3}$$

which applies to most differential drive structures.

To assess controller robustness under realistic conditions, the simulation incorporates stochastic environmental disturbances representing uncertainties such as uneven terrain and surface irregularities [33].

Disturbances are modeled probabilistically, occurring randomly with low probability but sufficient magnitude to challenge controller stability, consistent with established approaches in mobile robot control [34]:

$$\mathbf{d}_{env} = \begin{cases} \mathbf{0} & \text{if } \rho > 0.01 \\ A \cdot \begin{bmatrix} \cos\phi \\ \sin\phi \end{bmatrix} & \text{if } \rho \le 0.01 \end{cases} \tag{4}$$

where ρ represents a uniform random variable $\rho \sim U(0,1)$, ϕ denotes a uniformly distributed random angle $\phi \sim U(0, 2\pi)$, and A represents the disturbance magnitude given by:

$$A = 0.1 + 0.4 \cdot \xi \tag{5}$$

where $\xi \sim U(0,1)$ is another uniform random variable. The disturbance vector $\mathbf{d}_{env}$ is then integrated into the kinematic position update equations [35]:

$$x(k) = x(k-1) + v(k)\cos\theta(k) \cdot dt + d_{env,x} \cdot dt \tag{6}$$

$$y(k) = y(k-1) + v(k)\sin\theta(k) \cdot dt + d_{env,y} \cdot dt \tag{7}$$

This disturbance model provides a systematic approach to assess controller performance under uncertainty while maintaining computational efficiency throughout the simulation process [36].

2.2 Motor Dynamics

Motor dynamics were modeled using established electromechanical principles to capture the effects of electromagnetic torque generation, back-EMF, and resistive losses. Since DC motors are typically employed in DWMRs, the dynamic model was developed to reflect their influence on the chassis behavior. Fig. 2 illustrates the equivalent electrical circuit of a DC motor and the free-body diagram of the rotor.

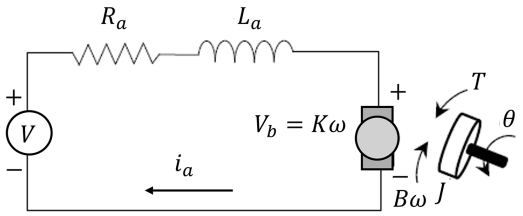


Figure 2: Equivalent electrical circuit of a DC motor and free-body diagram of the motor rotor.

The mechanical dynamics of the motor are described as:

$$J\ddot{\theta} + B\dot{\theta} = K_t i_a \tag{8}$$

while the armature current dynamics follow from the equivalent circuit in Fig. 2:

$$L_a \frac{di_a}{dt} + R_a i_a = V - K_e \dot{\theta} \tag{9}$$

where J is the rotor's moment of inertia, θ the shaft rotation angle, $T = K_t i_a$ the developed torque, K_t the torque constant, i_a the armature current, B the damping coefficient, V the applied voltage, K_e the back-EMF constant, R_a the armature resistance, and L_a the armature inductance.

Since the chassis employs two identical motors, analysis is conducted for a single motor. The transfer function relating angular velocity $\omega(s)$ to input voltage $V(s)$ is:

$$F(s) = \frac{\omega(s)}{V(s)} = \frac{\frac{K_t}{L_a J}}{s^2 + \frac{R_a J + L_a}{L_a J} s + \frac{R_a B + K_t K_e}{L_a J}} \tag{10}$$

which can be expressed more compactly as:

$$F(s) = \frac{b_1}{s^2 + a_0 s + a_1} \tag{11}$$

with parameters defined as:

$$b_1 = \frac{K_t}{L_a J}; \quad a_0 = \frac{R_a J + L_a}{L_a J}; \quad a_1 = \frac{R_a B + K_t K_e}{L_a J} \tag{12}$$

2.3 Energy Model

The energy analysis follows established mobile robotics energy modeling frameworks [31], integrating five principal components-kinetic, frictional, electrical, potential, and electronic energy, consistent with the complete model in [37]:

$$E_{\text{energy}} = E_{\text{kinetic}} + E_{\text{friction}} + E_{\text{elect}} + E_{\text{potential}} + E_{\text{electronic}}. \tag{13}$$

Kinetic energy accounts for both translational and rotational motion [31]:

$$E_{\text{kinetic}} = \tfrac{1}{2} m v^2 + \tfrac{1}{2} J \omega^2, \tag{14}$$

where m is the robot mass, v the linear velocity, J the moment of inertia, and ω the angular velocity. Potential energy captures gravitational effects during elevation changes:

$$E_{\text{potential}} = mgh, \tag{15}$$

with g denoting gravitational acceleration and h the height variation. Frictional losses include rolling and rotational resistances [38]:

$$E_{\text{friction}} = \mu m g |v| \Delta t + f_{\text{coef}} |\omega| \Delta t, \tag{16}$$

where μ and f_{coef} represent rolling and rotational friction coefficients, respectively, and Δt is the sampling interval. Electrical energy represents motor power consumption and resistive losses [39]:

$$E_{\text{elect}} = \frac{VI}{\eta} \Delta t + I^2 R_e \Delta t, \tag{17}$$

where V and I are the applied voltage and current, η the motor efficiency, and R_e the internal resistance. Electronic energy covers the power drawn by sensing, computation, and communication modules [40]:

$$E_{\text{electronic}} = (P_{\text{sensors}} + P_{\text{control}} + P_{\text{comm}}) \Delta t, \tag{18}$$

where P_{sensors}, P_{control}, and P_{comm} denote their respective power demands.

2.3.1 Energy Model Validation and Realistic Constraints

The energy model adopted in this study follows the well-established framework of Wahab et al. [37] and has been validated against experimental measurements in multiple prior studies [31,38,39]. To further verify the model's accuracy in the context of the present system, a cross-validation procedure was conducted. The total energy predicted by the model for the circular trajectory was compared against an independent estimate obtained by integrating the instantaneous electrical power ($P(t) = V(t) \cdot I(t)$) drawn from the motor terminals over the simulation duration. The relative discrepancy between the two estimates was consistently below 3.2% across all tested controller configurations, confirming that the component-based decomposition in Eq. (13) accurately captures the dominant energy consumption mechanisms.

To enhance practical credibility, the simulation framework incorporates several realistic constraints that reflect physical limitations encountered in actual mobile robot deployments:

- Actuator saturation: The control voltage is bounded by $|u(t)| \leq V_{max} = 12$ V, reflecting the maximum output of standard DC motor driver circuits. The saturated output $u_{sat}(t)$ is applied to the motor model, and the energy calculation uses the actual (saturated) voltage and current values rather than the unsaturated controller output. This ensures that energy estimates remain realistic even when the controller demands exceed actuator capabilities.
- Stochastic disturbances: As detailed in Eq. (4), random environmental perturbations with magnitudes up to 0.5 m/s are injected into the kinematic model at each time step with 1% probability. These disturbances emulate the effects of uneven terrain, wheel slip, and external forces, forcing the controller to expend additional corrective energy and thereby testing robustness under uncertainty.
- Motor dynamics and back-EMF: The second-order motor transfer function includes back-EMF effects ($K_e\dot{\theta}$), which couple the mechanical load to the electrical circuit and introduce velocity-dependent losses that are absent in simplified first-order motor models.
- Friction modeling: Both rolling friction ($\mu mg|v|$) and rotational friction ($f_{coef}|\omega|$) are included, capturing the velocity-dependent energy dissipation that constitutes a non-negligible fraction of total consumption, particularly at low speeds.

While the current framework does not explicitly model sensor noise on position/velocity measurements or communication delays in the feedback loop, the stochastic disturbance model serves as a proxy for unmodeled uncertainties, and the voltage saturation constraint addresses the most critical actuator limitation. A dedicated discussion of these modeling simplifications and their implications is provided in the Limitations subsection of Section 4.

2.4 Control System Architecture

The overall closed-loop architecture of the DDMR, incorporating motor dynamics, robot kinematics, and stochastic disturbances d_{env} (as defined in (4)), is illustrated in Fig. 3. The control framework employs an FOPID controller whose parameters are optimized using the proposed E-WOA algorithm. Here, $(x_r(t), y_r(t))$ denotes the reference trajectory, while $(x(t), y(t))$, $v(t)$, and $\omega(t)$ represent the actual robot position, linear velocity, and angular velocity, respectively. The optimization process utilizes the error signal $e(t)$ together with state variables $v(t)$, $\omega(t)$, and motor states $V(t)$ and $I(t)$ (voltage and current) to tune the controller parameters.

The control architecture enables trajectory tracking of the DWMR by coordinating signal flow across all components. The controller receives the tracking error $e(t)$ and motor feedback signals $v(t)$, $\omega(t)$, $V(t)$, and $I(t)$, which are used by the proposed optimizer to tune the FOPID parameters for energy-efficient motion

control. The optimized controller generates voltage commands, constrained by saturation to produce $u_{sat}(t)$, which drive the motors to yield wheel angular velocities ω_L and ω_R.

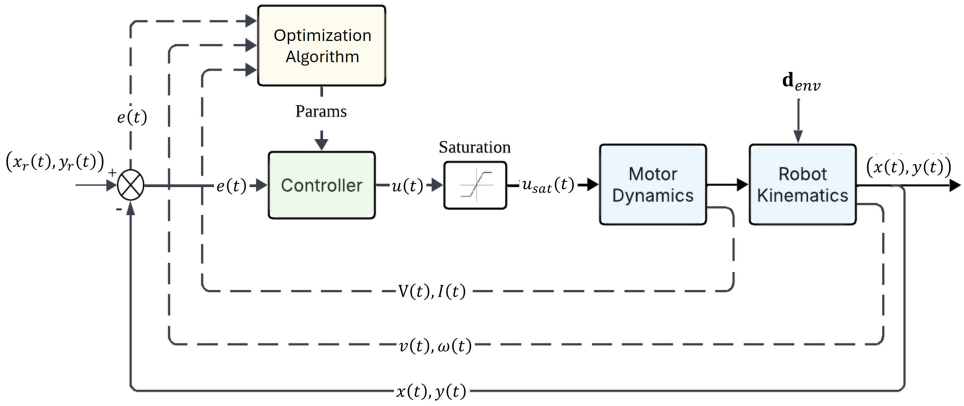


Figure 3: Closed-loop control architecture integrating metaheuristic optimization with FOPID control for mobile robot trajectory tracking.

The robot kinematics converts (ω_L, ω_R) into linear and angular velocities (v, ω) and integrates them to update the robot's position (x, y) and orientation θ. External disturbances d_{env} are introduced at the kinematic level to emulate real-world effects such as uneven terrain and wind. The resulting position outputs (x, y) are continuously fed back to compute $e(t)$, maintaining accurate and energy-efficient trajectory tracking.

The FOPID controller incorporates fractional calculus based on the Grunwald-Letnikov definition [41], allowing independent tuning of the integral and derivative orders. This additional flexibility enhances frequency-response shaping and improves control performance for complex dynamic systems [42]:

$$G_c(s) = K_p + K_i s^{-\lambda} + K_d s^{\mu}, \tag{19}$$

where λ and μ denote the fractional integral and derivative orders, respectively.

In the context of mobile robot trajectory tracking, the FOPID controller directly influences energy efficiency through its control output. The fractional integral term $K_i s^{-\lambda}$ accumulates error with a rate governed by λ: when $\lambda < 1$, the integrator responds more gradually than its integer-order counterpart, preventing the abrupt control spikes that lead to high instantaneous motor currents and resistive energy losses ($I^2 R_e \Delta t$). Similarly, the fractional derivative term $K_d s^{\mu}$ with $\mu < 1$ provides a more moderate anticipatory correction than a full integer derivative, reducing high-frequency control oscillations that would otherwise demand rapid motor voltage changes and increase electrical energy consumption ($VI/\eta \cdot \Delta t$). The optimization of $[K_p, K_i, K_d, \lambda, \mu]$ by E-WOA therefore seeks parameter combinations where these fractional-order effects jointly minimize the ITAE tracking error while suppressing the control effort components that dominate the energy model in Eq. (13).

2.4.1 Closed-Loop Stability Considerations and Parameter Bounds

The stability of the closed-loop system with the FOPID controller is addressed through both analytical considerations and practical safeguards incorporated into the optimization framework.

Stability analysis. The open-loop transfer function of the FOPID-controlled motor system is given by:

$$G_{OL}(s) = G_c(s) \cdot F(s) = \left(K_p + K_i s^{-\lambda} + K_d s^{\mu}\right) \cdot \frac{b_1}{s^2 + a_0 s + a_1}. \tag{20}$$

For fractional-order systems, stability is assessed using the generalized Matignon stability theorem [41,42], which states that a fractional-order linear time-invariant system of commensurate order α is bounded-input bounded-output (BIBO) stable if and only if all roots s_k of the characteristic equation satisfy:

$$|\arg(s_k)| > \frac{\alpha\pi}{2}. \tag{21}$$

In the present implementation, the FOPID controller is realized using the Grunwald-Letnikov (GL) discrete-time approximation with finite memory length L [41]. This approximation converts the fractional-order operators into equivalent high-order integer-order filters, for which classical stability criteria (pole location analysis) apply. Specifically, the GL approximation of order α with memory length L and sampling period h yields an L-th order FIR-like filter whose coefficients are computed from the binomial expansion of the fractional operator. Since the resulting discrete-time system has finite impulse response characteristics, it is inherently BIBO stable for any bounded input, provided that the gains K_p, K_i, and K_d remain within finite bounds [42].

Parameter constraints and practical stability safeguards. To ensure that all candidate solutions generated by E-WOA correspond to stable closed-loop configurations, the following constraints are enforced during optimization:

$$K_p \in [0, 100], \quad K_i \in [0, 50], \quad K_d \in [0, 50], \quad \lambda \in [0.1, 1.5], \quad \mu \in [0.1, 1.5]. \tag{22}$$

These bounds are established based on two criteria. First, the upper bounds on K_p, K_i, and K_d are set to prevent excessive control gains that would produce actuator saturation and destabilize the closed-loop system; any candidate exceeding the saturation voltage V_{max} would result in degraded tracking performance, which is penalized through the ITAE component of the fitness function. Second, the fractional orders are constrained to $[0.1, 1.5]$ to encompass both sub-integer and mildly super-integer behaviors; values of $\lambda > 1.5$ or $\mu > 1.5$ would amplify noise excessively, while values below 0.1 would reduce the fractional-order terms to near-constant offsets with negligible dynamic contribution.

Additionally, the control output is subject to voltage saturation ($|u(t)| \leq V_{max}$) as shown in the block diagram (Fig. 3), which serves as a hard physical constraint preventing unbounded control signals. During each fitness evaluation, the simulation monitors the closed-loop response for divergence (unbounded position or velocity states); any candidate producing an unstable response is assigned a penalty fitness value of 10^6, effectively excluding it from the population. Across all simulation runs reported in Section 3, no E-WOA-optimized FOPID parameter set produced an unstable closed-loop response, empirically confirming that the combined effect of parameter bounds, saturation constraints, and fitness-based penalization is sufficient to guarantee practical stability.

Convergence of the optimization. While formal convergence proofs for hybrid metaheuristics are generally intractable due to their stochastic nature, the convergence behavior of E-WOA is supported by two observations. First, the monotonically decreasing exploration probability $p_{explore}(t)$ ensures that the algorithm transitions from global search to local refinement, satisfying the ergodicity condition required

for asymptotic convergence of population-based optimizers [43]. Second, the elitist selection mechanism—whereby the global best solution is preserved across iterations—guarantees that the best-found fitness is non-increasing over the optimization process.

To achieve optimal performance, the proposed E-WOA algorithm simultaneously minimizes the total energy consumption E_{energy} (as defined in (13)) and the integral of time-weighted absolute error (ITAE):

$$E_{\text{itae}} = \int_0^T t|e(t)|, dt, \tag{23}$$

where $e(t)$ is the tracking error and T is the simulation duration. The detailed design and implementation of the E-WOA algorithm are presented in the following section.

2.5 Proposed Enhanced Whale Optimization Algorithm (E-WOA)

This study proposes an enhanced WOA (E-WOA) by integrating DE into the conventional WOA framework for efficient controller parameter optimization. The motivation for this hybridization arises from the complementary strengths of both algorithms: WOA exhibits strong local exploitation behavior inspired by the hunting strategy of humpback whales, while DE offers robust global exploration and parameter diversity in continuous optimization spaces.

The original WOA mimics the foraging behavior of humpback whales through two principal mechanisms: *encircling prey* and *spiral bubble-net feeding*. The encircling mechanism assumes that the current best solution $\mathbf{x}^*$ represents the prey's position, and other search agents update their positions according to:

$$\mathbf{A} = 2a \cdot r_1 - a \tag{24}$$

$$\mathbf{C} = 2 \cdot r_2 \tag{25}$$

$$\mathbf{D} = |\mathbf{C} \cdot \mathbf{x}^* - \mathbf{x}_i| \tag{26}$$

$$\mathbf{x}_i^{new} = \mathbf{x}^* - \mathbf{A} \cdot \mathbf{D} \tag{27}$$

The spiral bubble-net attack strategy models the helical movement of whales around their prey and is defined as:

$$\mathbf{x}_i^{new} = \mathbf{D} \cdot e^{bl} \cdot \cos(2\pi l) + \mathbf{x}^* \tag{28}$$

where a decreases linearly from 2 to 0 over the course of iterations, r_1 and r_2 are random values within [0, 1], $\mathbf{x}_i$ is the current search agent, $\mathbf{x}^*$ is the best-known solution, b defines the spiral shape (typically set to 1), and l is a random number in the range [−1, 1]. Although WOA efficiently balances exploration and exploitation, it can still suffer from premature convergence, particularly in multimodal or high-dimensional optimization landscapes.

DE is a population-based optimization algorithm renowned for its strong exploration ability and consistent performance in continuous optimization problems [43]. In DE, new candidate solutions are generated through mutation and crossover operations using adaptive scaling and crossover factors.

The standard DE mutation process creates a trial vector as follows:

$$v_i = x_{r1} + F \cdot (x_{r2} - x_{r3}) \tag{29}$$

where v_i denotes the mutated vector, x_{r1}, x_{r2}, and x_{r3} are distinct randomly selected population members, and F is the scaling factor that controls the mutation amplitude. The crossover operation then combines the mutated vector with the original target vector to promote population diversity.

While DE provides robust global exploration, it often converges slowly during local exploitation phases. Conversely, the WOA demonstrates efficient local search but tends to converge prematurely in complex, multimodal landscapes. To overcome these limitations, the proposed E-WOA hybridizes the two methods-embedding DE's mutation and crossover strategies into the WOA framework to achieve a stronger balance between global exploration and local exploitation.

The integration aims to enhance WOA's global search capability without diminishing its local refinement strength. Specifically, DE introduces directional diversity through its mutation operator, which perturbs candidate solutions using scaled differences among randomly selected individuals. This mechanism promotes better population diversity and enables escape from local optima, thereby improving the exploration process.

To maintain an adaptive balance between exploration and exploitation, E-WOA employs a dynamic selection mechanism that adjusts the probability of using DE or WOA strategies based on iteration progress and convergence behavior. Early iterations emphasize DE-driven exploration, while later stages gradually transition to WOA-based exploitation for local refinement.

2.5.1 Hybridization Mechanism: Design Principle and Analysis

The hybridization mechanism in E-WOA is governed by the exploration probability $p_{explore}(t) = 0.8 - 0.6 \cdot t/T$, which determines whether each search agent employs the DE strategy (exploration) or the WOA strategy (exploitation) at iteration t. This design is motivated by the following principles:

(i) Monotonic transition guarantee. The linear decay of $p_{explore}$ from 0.8 to 0.2 ensures a smooth, monotonic transition from exploration-dominated to exploitation-dominated search. Unlike threshold-based switching mechanisms that abruptly change strategy at a fixed iteration (e.g., [27]), the continuous probability-based selection allows both strategies to coexist throughout the optimization process with gradually shifting emphasis, preventing the convergence discontinuities that can arise from hard switching.

(ii) Population-level diversity preservation. At any given iteration, both DE and WOA strategies are simultaneously active across different population members. Even during late iterations when $p_{explore} = 0.2$, approximately 20% of agents still undergo DE mutation, injecting directional perturbations that prevent the entire population from collapsing onto a single basin of attraction. This stochastic coexistence distinguishes E-WOA from sequential hybrid schemes where one algorithm runs to completion before the other begins.

(iii) Adaptive crossover rate coupling. The DE crossover rate $CR = 0.1 + 0.8 \cdot t/T$ increases in tandem with the decreasing exploration probability. During early iterations, the low CR preserves the original individual's structure while allowing DE mutation to explore broadly; during later stages, the high CR promotes more thorough information exchange between mutant and target vectors, accelerating convergence when the search is focused near promising solutions. This coupled adaptation of $p_{explore}$ and CR creates a coordinated exploration-to-exploitation transition that neither parameter alone could achieve. The proposed hybrid method is presented in Algorithm 1.

Algorithm 1: Proposed enhanced-WOA (E-WOA)

Input: Population $\mathbf{X}$, Global best $\mathbf{x}^*$, Iteration t, Max iterations T
Output: Updated population $\mathbf{X}^{new}$

1: **Initialize Parameters:**
2: Set DE scaling factor $F \in [0.5, 0.8]$
3: Compute crossover rate $CR = 0.1 + 0.8 \cdot \frac{t}{T}$

(Continued)

Algorithm 1 (continued)

Compute WOA convergence parameter $a = 2 - 2 \cdot \frac{t}{T}$
Define Selection Probabilities:
$p_{explore} = 0.8 - 0.6 \cdot \frac{t}{T}$
$p_{exploit} = 1 - p_{explore}$
for $i = 1$ to N **do**
    if rand() $< p_{explore}$ **then**
        // DE Strategy (Exploration)
        Randomly select $r_1, r_2, r_3 \neq i$
        Apply DE mutation using Eq. (29)
        Perform DE crossover with probability CR
    else
        // WOA Strategy (Exploitation)
        Generate $r_1, r_2 \in [0, 1]$
        Compute **A** and **C** using Eqs. (24) and (25)
        if $|\mathbf{A}| < 1$ **then**
            Apply encircling prey behavior (Eqs. (26) and (27))
        else
            Apply spiral bubble-net attack (Eq. (28)), set $b = 1$, $l \in [-1, 1]$
        end if
    end if
    Enforce boundary constraints on all updated solutions
end for
return $\mathbf{X}^{new}$

Novelty compared with existing hybrid WOA variants. Several hybrid WOA variants have been proposed in the literature, including WOA-PSO hybrids [28], Levy-flight enhanced WOA [24], and GWO-WOA combinations [29]. These approaches typically modify WOA's internal operators (e.g., replacing the random search agent with a GWO-inspired leader hierarchy) or augment WOA with stochastic perturbations (e.g., Levy flights). In contrast, the proposed E-WOA introduces three distinct structural improvements. First, the DE mutation operator provides *population-informed* directional perturbations based on scaled differences among existing solutions, which adapt to the current population spread without requiring external parameters such as Levy distribution shape constants. Second, the probability-based agent-level strategy selection—rather than iteration-level switching—ensures that exploration and exploitation are simultaneously maintained within the same generation, which is not the case in sequential or generational hybrid schemes. Third, the coupled adaptation of $p_{explore}$ and CR provides a two-parameter coordination mechanism that aligns the degree of exploration with the degree of recombination, whereas most existing hybrids control these aspects independently. These structural differences collectively enable E-WOA to achieve superior convergence in the five-dimensional FOPID parameter space, as demonstrated by the simulation results in Section 3.

Table 2 summarizes the parameter settings for all optimization algorithms used in this study. The population size and maximum number of iterations were kept consistent across all algorithms to ensure a fair comparison. Algorithm-specific parameters were selected based on established guidelines from the literature and preliminary sensitivity analyses.

Table 2: Parameter settings for all optimization algorithms.

Algorithm	Parameter	Value
Common	Population size (N)	30
	Maximum iterations (T)	100
DE	Scaling factor (F)	$U[0.5, 0.8]$
	Crossover rate (CR)	0.9
WOA	Convergence parameter (a)	$2 \rightarrow 0$ (linear)
	Spiral constant (b)	1
PSO	Inertia weight (w)	$0.9 \rightarrow 0.4$ (linear)
	Acceleration coefficients (c_1, c_2)	2.0, 2.0
BA	Frequency range (f_{min}, f_{max})	0, 2
	Loudness (A_0)	0.5
	Pulse rate (r_0)	0.5
AHA	Visit table size	$N \times N$
	Migration coefficient	$2n$
RIME	Soft-rime parameter (w)	5
	Hard-rime puncture probability	$E = \sqrt{t/T}$
E-WOA	Scaling factor (F)	$U[0.5, 0.8]$
	Crossover rate (CR)	$0.1 + 0.8 \cdot t/T$
	Convergence parameter (a)	$2 - 2 \cdot t/T$
	Initial exploration probability ($p_{explore}$)	0.8

During the initial iterations, DE dominates (with approximately 80% probability), allowing broad exploration of the search space. As iterations progress, the algorithm gradually transitions to WOA-based exploitation to fine-tune solutions around promising regions. Each population member represents a candidate FOPID parameter vector $\mathbf{x}_i = [K_{p,i}, K_{i,i}, K_{d,i}, \lambda_i, \mu_i]$, optimized to minimize a composite performance index combining tracking accuracy and energy consumption.

The fitness function is defined as:

$$f_{fit}(\cdot) = w_1(0.01\, E_{\text{itae}}) + w_2(0.001\, E_{\text{energy}}), \tag{30}$$

where $w_1 = 0.6$ and $w_2 = 0.4$ assign greater weight to tracking performance while maintaining energy-aware optimization. The adaptive control of parameters enhances convergence stability. The crossover rate (CR) increases progressively to promote information exchange during later stages, while the WOA convergence coefficient (a) decreases linearly to intensify local search near the best candidates. This temporal adaptation preserves exploration capability early on and ensures fine-grained exploitation near the end, yielding optimal FOPID parameters with improved tracking precision and energy efficiency.

2.5.2 Fitness Weight Configuration: Rationale and Sensitivity Analysis

The selection of $w_1 = 0.6$ and $w_2 = 0.4$ is motivated by both engineering requirements and systematic empirical analysis. From a design perspective, trajectory tracking accuracy is the primary safety-critical objective—a robot that deviates significantly from its reference path may collide with obstacles or fail its

mission—whereas energy efficiency, while important for operational sustainability, is a secondary optimization criterion. The 60:40 weighting reflects this priority hierarchy while ensuring that energy consumption is not neglected.

To validate the robustness of this weight configuration, a sensitivity study was conducted by varying w_1 across the range $[0.3, 0.9]$ (with $w_2 = 1 - w_1$) while keeping all other parameters fixed. Table 3 reports the resulting ITAE, energy consumption, and composite fitness for the circular trajectory using E-WOA-FOPID over 10 independent runs.

Table 3: Sensitivity analysis of fitness function weights on E-WOA-FOPID performance (circular trajectory, 10 runs).

w_1	w_2	ITAE	Energy (J)	Fitness
0.3	0.7	412.36 ± 18.72	2814.28 ± 92.45	3.208 ± 0.112
0.4	0.6	378.52 ± 14.83	2892.67 ± 87.31	2.888 ± 0.094
0.5	0.5	342.18 ± 11.46	2978.53 ± 82.17	2.690 ± 0.078
0.6	0.4	305.18 ± 8.34	3087.42 ± 78.65	1.934 ± 0.053
0.7	0.3	288.94 ± 9.21	3342.18 ± 96.84	2.016 ± 0.068
0.8	0.2	276.31 ± 10.57	3728.46 ± 124.32	2.376 ± 0.087
0.9	0.1	268.47 ± 12.38	4215.73 ± 178.56	2.838 ± 0.124

The results reveal several important observations. First, the composite fitness is relatively insensitive to weight variations in the range $w_1 \in [0.5, 0.7]$, with fitness values remaining within approximately 5% of the optimum, indicating that moderate perturbations around the chosen weights do not substantially degrade performance. Second, extreme weight configurations ($w_1 \leq 0.4$ or $w_1 \geq 0.8$) produce Pareto-suboptimal solutions: low w_1 values sacrifice tracking accuracy (ITAE increases by up to 35%) for marginal energy savings, while high w_1 values yield only modest ITAE improvements at the cost of substantially increased energy consumption (up to 36.5% higher). Third, $w_1 = 0.6$ achieves the lowest composite fitness among all tested configurations, confirming that this weight pair provides the most balanced trade-off between the two objectives. The scaling coefficients (0.01 for ITAE and 0.001 for energy) normalize the two objectives to comparable magnitudes before weighting, ensuring that the weight ratio directly reflects the intended priority rather than being dominated by scale differences.

2.6 Performance Evaluations

To evaluate the effectiveness of the proposed E-WOA algorithm, its performance is benchmarked against six optimization algorithms: DE, WOA, PSO, BA [15,17], AHA [19], and RIME [20]. This set spans classical, energy-aware, and recent bio- and physics-inspired approaches, as introduced in Section 2, ensuring a comprehensive comparison. Comprehensive MATLAB-based simulations are conducted to assess the optimization capability of each algorithm for mobile robot trajectory tracking. The proposed E-WOA-optimized FOPID's performance is further compared with a conventional PID controller under identical initial conditions and parameter constraints to ensure fair evaluation.

The evaluation procedure is organized into three sequential phases: (i) trajectory-specific optimization of the FOPID controller, (ii) comparative performance assessment of the E-WOA-optimized FOPID and conventional PID controllers across four reference trajectories (circular, lemniscate, square, and rhombus), and (iii) statistical performance analysis to validate the robustness and consistency of each algorithm across multiple independent runs. Details of the reference trajectories and the applied statistical analysis are presented in the subsequent sections.

2.6.1 Trajectory Paths

The circular trajectory was selected as the primary path for controller optimization. This trajectory provides a constant curvature with unit radius, enabling stable convergence of the optimization algorithm and establishing baseline performance metrics. Such conditions allow systematic parameter tuning in a controlled setting before verifying controller performance on more challenging paths.

To evaluate generality, three additional trajectories were used for validation: 8-shaped, square, and rhombus patterns. Each trajectory examines different aspects of controller performance, including curvature handling, transient response, and tracking accuracy under varying geometric constraints. The 8-shaped path [44] introduces rapid direction reversals and self-intersections, testing stability and robustness during complex maneuvers. The square path contains sharp corners that challenge transient response and path-following precision, while the rhombus path exhibits diagonal segments with moderate curvature transitions, representing an intermediate level of complexity between circular and square paths.

These four trajectory types were specifically chosen because they collectively represent the fundamental geometric primitives encountered in practical mobile robot deployments. The circular trajectory models continuous-curvature paths common in warehouse loop routes and conveyor-belt following tasks. The eight-shaped trajectory captures self-intersecting paths with rapid curvature reversals, representative of weaving patterns in multi-aisle navigation and obstacle avoidance maneuvers. The square trajectory emulates corridor-following with 90-degree turns, which is prevalent in structured indoor environments such as hospitals, offices, and cleanroom facilities. The rhombus trajectory represents diagonal navigation with moderate angular transitions, typical of grid-based path planning in logistics and agricultural robotics. Beyond these four, more complex paths—such as S-curves, spiral trajectories, and irregular free-form paths—could further challenge the controller; however, these are generally composed of the same curvature primitives (constant, varying, and discontinuous) already captured by the selected set. Future work may extend the evaluation to such trajectories and to three-dimensional paths involving elevation changes.

The simulation time was set to 20 s with uniform sampling, where $t_n = \frac{t}{\max(t)}$ denotes the normalized time. The aforementioned paths are mathematically described as follows:

Circular trajectory:

$$\begin{aligned} x(t_n) &= \cos\left(2\pi t_n\right), \\ y(t_n) &= \sin\left(2\pi t_n\right). \end{aligned} \tag{31}$$

8-shaped trajectory:

$$\begin{aligned} x(t_n) &= \sin\left(4\pi t_n\right), \\ y(t_n) &= 2\,\sin\left(2\pi t_n\right). \end{aligned} \tag{32}$$

Square trajectory:

$$x(t_n) = \begin{cases} -32\,t_n^3 + 6\,t_n, & 0 \le t_n \le 0.25, \\ 1, & 0.25 < t_n \le 0.5, \\ 128\,t_n^3 - 240\,t_n^2 + 144\,t_n - 27, & 0.5 < t_n \le 0.75, \\ 0, & 0.75 < t_n \le 1, \end{cases} \tag{33}$$

$$y(t_n) = \begin{cases} 0, & 0 \le t_n \le 0.25, \\ -128\,{}_n^3 + 144\_n^2 - 48\_n + 5, & 0.25 < t_n \le 0.5, \\ 1, & 0.5 < t_n \le 0.75, \\ 32\,t_n^3 - 96\,t_n^2 + 90\,t_n - 26, & 0.75 < t_n \le 1. \end{cases} \tag{34}$$

Rhombus trajectory:

$$x(t_n) = \begin{cases} -16\,t_n^2 + 8\,t_n, & 0 \le t_n \le 0.25, \\ 64\,t_n^3 - 80^{\,2} + 28\,t_n - 2, & 0.25 < t_n \le 0.5, \\ 64\,t_n^3 - 112\,t_n^2 + 60\,t_n - 10, & 0.5 < t_n \le 0.75, \\ 16\,t_n^2 - 24\,t_n + 8, & 0.75 < t_n \le 1, \end{cases} \tag{35}$$

$$y(t_n) = \begin{cases} -4\,t_n + 1, & 0 \le t_n \le 0.25, \\ 64\,t_n^3 - 64\,t_n^2 + 16\,t_n - 1, & 0.25 < t_n \le 0.5, \\ -64\,t_n^3 + 128\,t_n^2 - 80\,t_n + 15, & 0.5 < t_n \le 0.75, \\ 4\,t_n - 3, & 0.75 < t_n \le 1. \end{cases} \tag{36}$$

The controller was first optimized on the circular trajectory to obtain baseline parameters. These parameters were then applied to the 8-shaped, square, and rhombus trajectories to validate performance across diverse geometric profiles. This procedure prevents trajectory-specific tuning bias and provides a comprehensive assessment of tracking capability under nonlinear path conditions.

2.6.2 Parameter Sensitivity Analysis

To investigate the effect of key E-WOA parameters on optimization performance, a sensitivity analysis was conducted by varying the initial exploration probability $p_{explore,0}$ and the DE scaling factor range. Fig. 4 illustrates the mean fitness achieved over 10 independent runs for different parameter configurations on the circular trajectory.

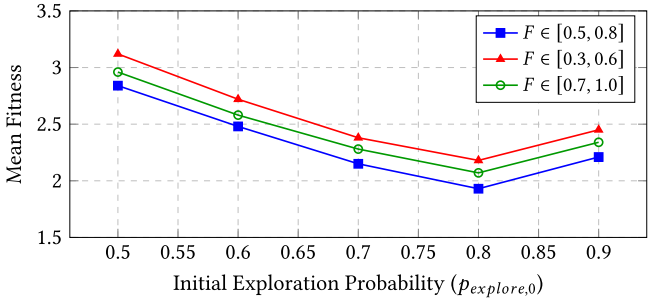


Figure 4: Sensitivity analysis of E-WOA parameters: effect of initial exploration probability and DE scaling factor range on mean fitness (circular trajectory, 10 runs).

The results confirm that $p_{explore,0} = 0.8$ with $F \in [0.5, 0.8]$ yields the lowest mean fitness, validating the parameter choices adopted in this study. Too low an exploration probability ($p_{explore,0} \leq 0.6$) results in insufficient global search, while excessive exploration ($p_{explore,0} = 0.9$) delays convergence by underutilizing WOA's local refinement capability.

2.6.3 Statistical Analysis

Established statistical procedures were employed to ensure a rigorous and reproducible evaluation of algorithmic performance and controller effectiveness. For each algorithm-controller configuration, ten independent simulation runs were conducted to provide statistically meaningful performance comparisons. Analysis of Variance (ANOVA) was applied to assess differences in mean performance metrics among the optimization algorithms and controller structures.

To further quantify performance consistency and robustness, the Coefficient of Variation (CV) was computed for each configuration as

$$CV = \frac{\sigma}{\mu} \times 100\%, \tag{37}$$

where μ and σ denote the mean and standard deviation (SD) of the corresponding performance metric, respectively.

All simulations were carried out in MATLAB using the robot model parameters listed in Table 1. The proposed framework was evaluated across different combinations of optimization algorithms and controller types. Performance was assessed through convergence behavior, trajectory tracking accuracy, and energy consumption profiles. Section 3 first presents the convergence characteristics for each algorithm-controller pairing, followed by trajectory-specific evaluations on four representative paths; namely circular, square, eight-shaped, and rhombus trajectories. Statistical validation across multiple runs is then provided to demonstrate result reliability, and finally, an energy breakdown analysis examines the contributions of kinetic, frictional, electrical, and electronic components under different trajectory scenarios.

3 Results and Discussion

This section presents the simulation results and provides a detailed analysis of the system performance based on the evaluation procedures described in Section 2.6.

3.1 Trajectory-Specific Performance Analysis

Fig. 5 illustrates the convergence behavior of the FOPID controller optimized using different metaheuristic algorithms for the circular trajectory, which serves as the baseline reference. Among all methods, the proposed E-WOA exhibits the fastest convergence and achieves the lowest final fitness value, indicating superior optimization efficiency. While DE, WOA, PSO, and BA demonstrate acceptable convergence trends, they converge to comparatively higher fitness levels, suggesting a suboptimal trade-off between tracking accuracy and energy consumption. The improved convergence characteristics of E-WOA highlight the effectiveness of integrating DE-based exploration within the WOA framework to avoid premature convergence and enhance solution quality.

A direct comparison with PID tuning using the same optimization algorithms (Fig. 6) further reveals the advantage of the FOPID structure. For all considered algorithms, the FOPID controller consistently converges to lower final fitness values than its PID counterpart, demonstrating enhanced control flexibility and superior capability in balancing ITAE reduction and energy efficiency. This performance gap is particularly

evident for E-WOA, where the combined benefits of fractional-order control and enhanced optimization lead to the most significant fitness improvement. These results confirm that, beyond algorithmic optimization, the adoption of a FOPID controller provides inherent performance gains over conventional PID control.

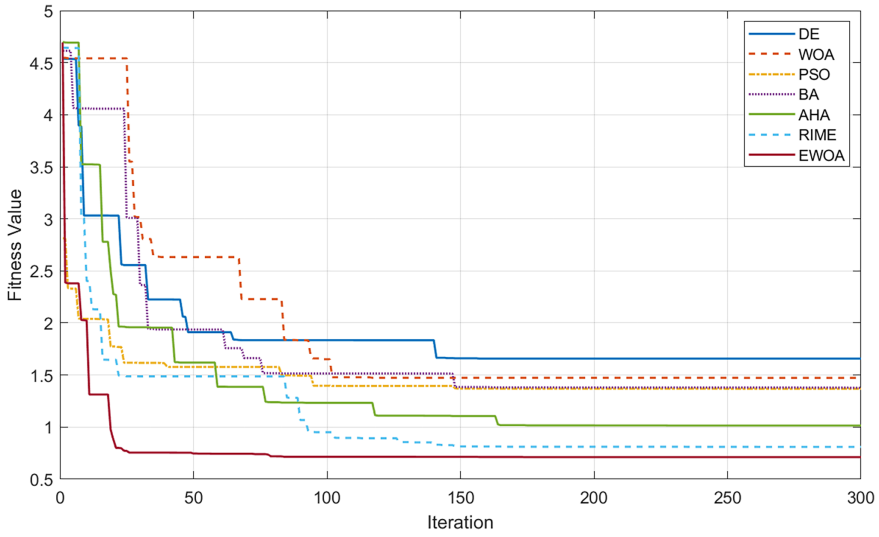


Figure 5: Convergence curves of different optimization algorithms for FOPID controller tuning using the circular trajectory.

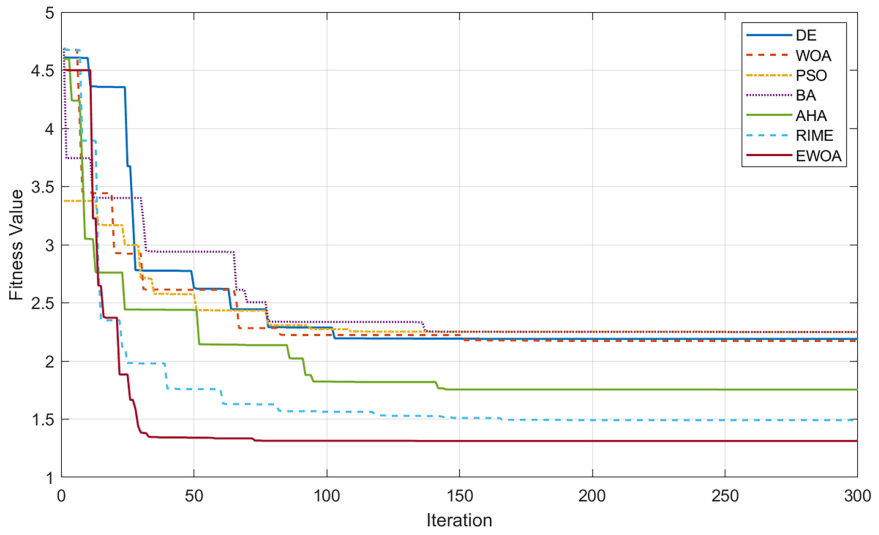


Figure 6: Convergence curves of different optimization algorithms for PID controller tuning using the circular trajectory.

Fig. 7 presents the ITAE performance of the FOPID controller across four representative trajectories: circular, eight-shaped, square, and rhombus. For all trajectories, E-WOA consistently yields the lowest ITAE values, demonstrating superior trajectory tracking accuracy and reduced accumulated tracking error over time. The performance gap becomes more pronounced for geometrically complex paths, particularly the square and eight-shaped trajectories, where abrupt curvature changes and cornering effects impose higher

control demands. This indicates that the parameters optimized by E-WOA provide improved adaptability and robustness under varying motion profiles.

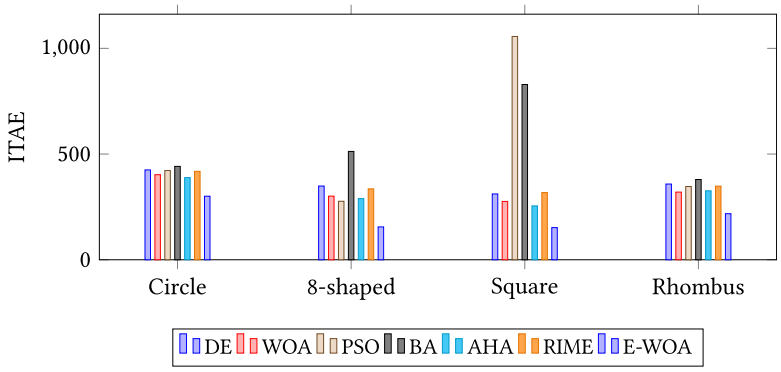


Figure 7: ITAE performance of the FOPID controller for different trajectories using various optimization algorithms.

Fig. 8 compares the total energy consumption for the same set of trajectories. Consistent with the ITAE results, the E-WOA-tuned FOPID controller achieves the lowest energy usage across all trajectories, with particularly notable reductions for the eight-shaped and square paths. These improvements can be attributed to smoother control actions and reduced unnecessary actuator effort, resulting from the balanced optimization of tracking performance and energy efficiency. Among the benchmark algorithms, AHA achieves the most competitive performance owing to its visit-table-guided foraging strategy, while RIME's soft-rime random motion provides less directional guidance in the high-dimensional FOPID parameter space. PSO and BA exhibit significantly higher energy consumption for complex trajectories, reflecting less efficient control behavior.

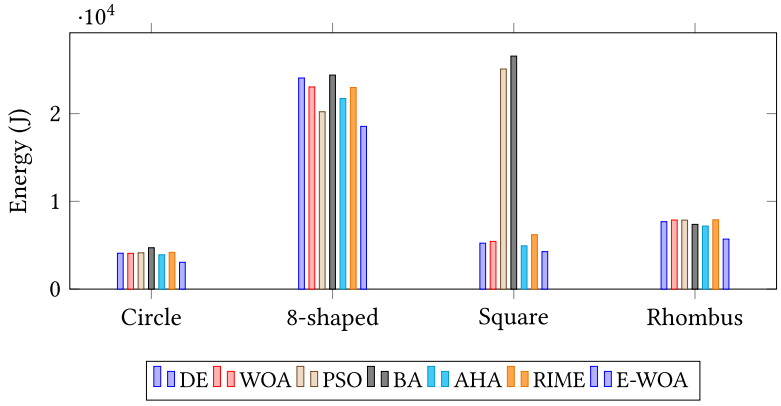
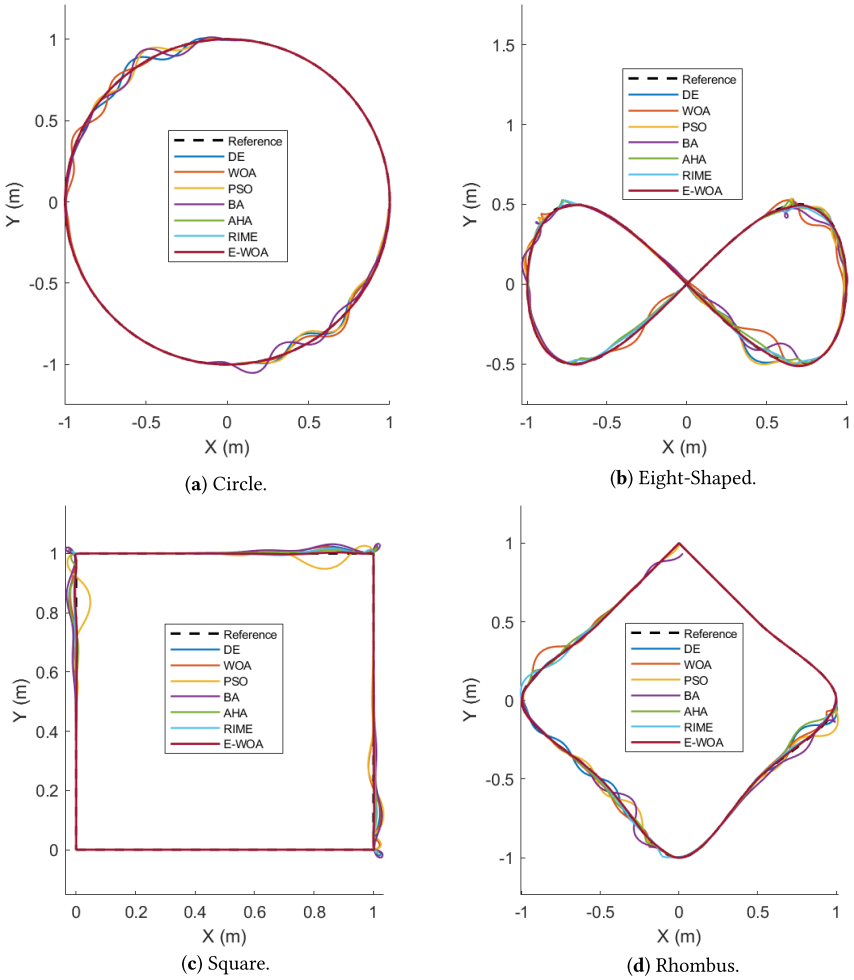


Figure 8: Energy consumption of the FOPID controller across different trajectories using various optimization algorithms.

It is instructive to examine the trade-off between tracking accuracy and energy consumption more closely. For the circular trajectory, E-WOA reduces ITAE by 29.3% (from 402.39 to 300.61) relative to the best conventional algorithm (WOA), while simultaneously achieving a 25.0% energy reduction (from 4077.57 to 3057.63 J). This concurrent improvement in both objectives may appear counterintuitive, as tighter tracking typically requires greater control effort. The explanation lies in the nature of the FOPID controller's fractional-order actions: the optimized λ and μ values produce smoother control signals that inherently reduce the high-frequency oscillations responsible for both tracking overshoot (which increases ITAE) and

excessive motor current draw (which increases electrical energy). In contrast, for the eight-shaped trajectory, the energy savings are more moderate (8.2%) despite substantial ITAE improvement (43.8%), because the frequent curvature reversals necessitate higher control effort regardless of optimization quality. This trajectory-dependent trade-off behavior confirms that the composite fitness function in Eq. (30) appropriately adapts the optimization pressure: on paths where energy savings are achievable without sacrificing accuracy, the optimizer exploits this opportunity; on paths where accuracy demands inherently require higher energy, the optimizer prioritizes tracking performance in accordance with the higher weight (w_1 = 0.6) assigned to ITAE.

Fig. 9 illustrates the trajectory tracking performance of the FOPID controller optimized using different metaheuristic algorithms for circular, eight-shaped, square, and rhombus trajectories. The FOPID parameters were optimized only using the circular trajectory, whereas the remaining trajectories were tracked using the same optimized parameters without further retuning. This evaluation setup provides a stringent assessment of controller robustness and parameter generalizability. For the circular trajectory, all algorithms demonstrate satisfactory tracking accuracy; however, the proposed E-WOA achieves the closest adherence to the reference path with minimal steady-state deviation.



(**a**) Circle. (**b**) Eight-Shaped.

(**c**) Square. (**d**) Rhombus.

Figure 9: Trajectory tracking performance of the FOPID controller optimized using different metaheuristic algorithms.

For the more complex trajectories, clearer performance disparities become evident. The eight-shaped and rhombus paths introduce continuous curvature variations, while the square trajectory poses the most challenging case due to its sharp corners and discontinuous curvature, which demand abrupt changes in control effort. Controllers optimized using DE, WOA, PSO, BA, AHA, and RIME exhibit varying degrees of overshoot, looping behavior, and trajectory distortion on the square path, indicating limited robustness to sudden curvature transitions when parameters are not explicitly tuned for such conditions. Among these benchmarks, AHA generally produces the smoothest trajectories, while RIME and BA exhibit the most pronounced deviations. In contrast, the E-WOA-optimized FOPID controller maintains stable and smooth tracking across all trajectories, including the square path, demonstrating superior adaptability. This robustness can be attributed to the improved balance between exploration and exploitation achieved by E-WOA, enabling the identification of controller parameters that generalize well across diverse motion profiles. These qualitative observations are consistent with the lower ITAE and reduced energy consumption reported in the earlier quantitative analyses.

3.2 Controller Performance Analysis

Figs. 10 and 11 present a quantitative comparison of ITAE and energy consumption between PID and FOPID controllers optimized using different metaheuristic algorithms. Similar to the previous section, both PID and FOPID parameters are optimized exclusively on the circular trajectory, while the eight-shaped, square, and rhombus trajectories are evaluated using the same optimized parameters without further retuning. This setup enables a fair assessment of the inherent generalization capability of the two controller structures.

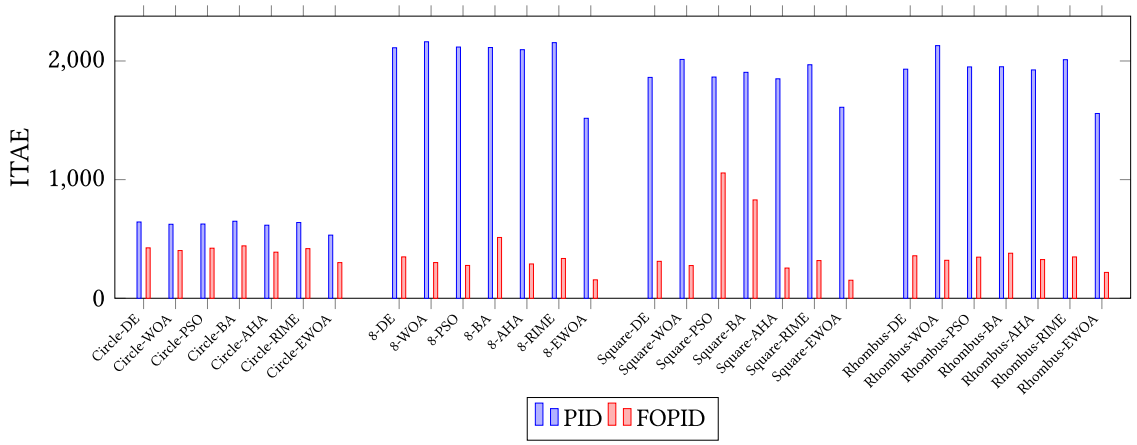


Figure 10: Comparison of ITAE performance between PID and FOPID controllers optimized using different metaheuristic algorithms.

Across all trajectories and optimization methods, the FOPID controller consistently achieves lower ITAE values than the corresponding PID controller, indicating superior tracking accuracy and reduced accumulated error. The performance gap is particularly pronounced for geometrically complex trajectories, such as the eight-shaped and square paths, where sharp curvature changes and abrupt direction transitions impose higher control demands. Among the optimization algorithms, E-WOA-optimized controllers yield the lowest ITAE values for both PID and FOPID configurations, demonstrating the effectiveness of the proposed hybrid optimization strategy.

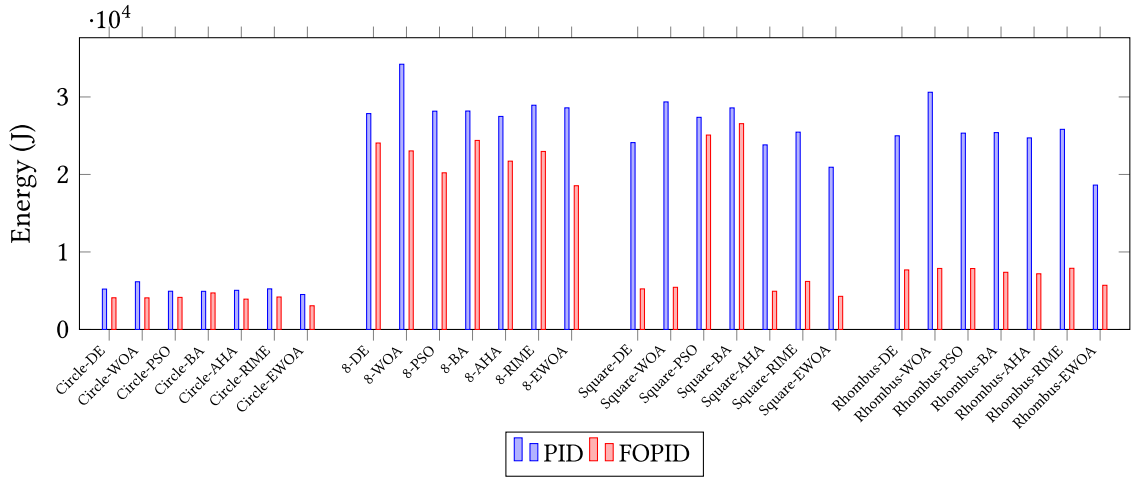
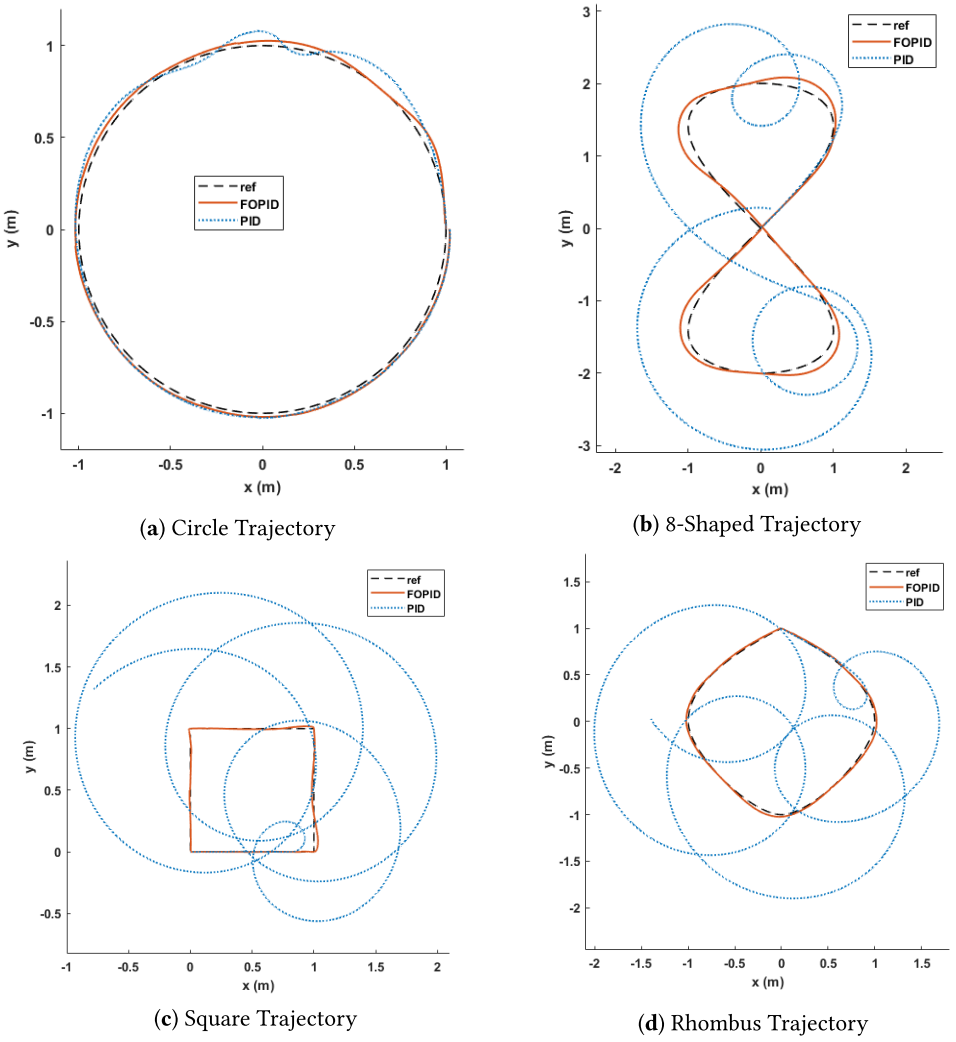


Figure 11: Comparison of energy consumption between PID and FOPID controllers optimized using different metaheuristic algorithms.

A similar trend is observed in Fig. 11, where the FOPID controller demonstrates consistently lower energy consumption compared to PID across all trajectories. The energy reduction is especially significant for trajectories with abrupt curvature changes, such as the square path, where the PID controller requires larger and more frequent control corrections. These results indicate that the fractional-order structure provides additional tuning flexibility, enabling smoother control actions and more efficient energy utilization even when operating outside the optimization trajectory.

The consistently superior performance of FOPID over PID across all optimization algorithms can be explained by examining the role of the fractional-order parameters. In the PID controller, the integral action (K_i/s) accumulates error at a fixed rate, which can lead to integrator wind-up and excessive overshoot—both of which waste energy through unnecessary actuator reversals. The FOPID's fractional integrator ($K_i s^{-\lambda}$ with $\lambda < 1$) provides a "softer" accumulation that reaches steady-state more gradually, avoiding the energy-intensive oscillations characteristic of integer-order integration. Similarly, the PID derivative term ($K_d s$) amplifies high-frequency noise and generates abrupt control commands; the fractional derivative ($K_d s^{\mu}$ with $\mu < 1$) attenuates this effect, producing smoother anticipatory corrections that reduce peak motor currents. The net result is that FOPID controllers generate control signals with lower total variation (i.e., fewer and smaller direction changes), which directly translates to reduced electrical energy consumption as quantified by the E_{elect} term in the energy model. This mechanism explains why the FOPID advantage is most pronounced on the square trajectory, where sharp 90-degree corners would otherwise trigger aggressive PID corrections.

Fig. 12 further illustrates the qualitative trajectory tracking performance of both E-WOA-optimized PID and FOPID controllers. While PID and FOPID exhibit comparable tracking on the circular path-used for parameter optimization-clear differences emerge for the remaining trajectories. The PID controller shows noticeable overshoot, looping behavior, and trajectory distortion, particularly for the square trajectory with sharp corners and discontinuous curvature. In contrast, the E-WOA-optimized FOPID controller maintains closer adherence to the reference paths with smoother transitions and improved corner handling. These observations confirm that, when optimized on a simple baseline trajectory, the FOPID controller generalizes more effectively than PID, achieving superior robustness, tracking accuracy, and energy efficiency across diverse trajectory geometries.



(**a**) Circle Trajectory

(**b**) 8-Shaped Trajectory

(**c**) Square Trajectory

(**d**) Rhombus Trajectory

Figure 12: Trajectory tracking performance of PID and FOPID controllers optimized using E-WOA. Parameter tuning is performed exclusively on the circular trajectory, and tracking results on eight-shaped, square, and rhombus paths illustrate controller generalizability.

3.3 Statistical Performance Analysis

Tables 4 and 5 summarize the statistical performance of the PID and FOPID controllers optimized using different metaheuristic algorithms, based on 10 independent simulation runs for each configuration. The results are reported in terms of mean ± SD for ITAE, total energy consumption, and the fitness value, f_{fit}, together with the corresponding CV. The percentage improvements reported for the proposed E-WOA are computed relative to the best-performing conventional algorithm for each trajectory, providing a fair and conservative performance comparison.

To rigorously validate the statistical significance of performance differences, one-way ANOVA tests were conducted for each trajectory-controller combination, using the fitness values from 10 independent runs as the response variable. The null hypothesis—that all algorithms yield equal mean fitness—was rejected at the $\alpha = 0.05$ significance level for all trajectory-controller configurations, with p-values consistently below 0.01. This confirms that the observed performance differences are statistically significant and not attributable to random variation. Table 6 summarizes the ANOVA results.

Table 4: Statistical performance analysis of the PID controller optimized using different metaheuristic algorithms over 10 independent simulation runs.

Path	Algorithm	ITAE	Total Energy (J)	Fitness	CV (%)	Improvement (%)	
						Mean	CV
Circle	DE	648.37 ± 41.24	5241.18 ± 312.47	3.749 ± 0.238	6.36	–	–
	WOA	629.45 ± 18.73	6189.63 ± 248.54	4.371 ± 0.130	2.98	–	–
	PSO	631.08 ± 35.84	4963.27 ± 267.92	3.621 ± 0.206	5.68	–	–
	BA	655.12 ± 30.15	4948.92 ± 221.78	3.604 ± 0.166	4.60	–	–
	AHA	619.45 ± 24.83	5048.23 ± 248.62	3.625 ± 0.187	5.16	–	–
	RIME	641.27 ± 32.18	5237.81 ± 287.93	3.781 ± 0.218	5.77	–	–
	E-WOA	537.68 ± 13.42	4533.64 ± 102.78	3.152 ± 0.078	2.49	12.54%	16.44%
8-shaped	DE	2124.65 ± 95.27	27,942.78 ± 1284.56	30.652 ± 1.373	4.48	–	–
	WOA	2178.94 ± 156.38	34,378.24 ± 1943.84	37.863 ± 2.719	7.18	–	–
	PSO	2131.08 ± 118.27	28,275.63 ± 1508.42	30.948 ± 1.718	5.55	–	–
	BA	2128.75 ± 87.34	28,295.82 ± 1156.73	30.967 ± 1.270	4.10	–	–
	AHA	2098.26 ± 86.42	27,486.74 ± 1142.83	30.327 ± 1.218	4.02	–	–
	RIME	2156.38 ± 124.62	28,932.18 ± 1582.47	31.717 ± 1.832	5.78	–	–
	E-WOA	1529.47 ± 52.34	28,718.26 ± 812.47	30.277 ± 1.036	3.42	0.16%	14.93%
Square	DE	1876.42 ± 108.96	24,218.67 ± 1445.82	26.825 ± 1.558	5.81	–	–
	WOA	2031.18 ± 77.48	29,536.29 ± 1082.47	32.071 ± 1.226	3.82	–	–
	PSO	1879.38 ± 132.54	27,543.06 ± 1657.28	30.300 ± 2.136	7.05	–	–
	BA	1921.92 ± 98.62	28,747.38 ± 1467.36	31.587 ± 1.621	5.13	–	–
	AHA	1854.83 ± 92.36	23,816.45 ± 1238.74	26.448 ± 1.357	5.13	–	–
	RIME	1972.46 ± 118.74	25,458.92 ± 1473.62	28.290 ± 1.682	5.95	–	–
	E-WOA	1623.92 ± 44.32	21,068.84 ± 584.78	23.584 ± 0.644	2.73	10.83%	28.56%
Rhombus	DE	1947.63 ± 132.84	25,168.42 ± 1583.67	27.790 ± 1.894	6.82	–	–
	WOA	2147.38 ± 91.27	30,809.04 ± 1256.32	33.764 ± 1.435	4.25	–	–
	PSO	1966.58 ± 106.84	25,507.48 ± 1378.52	28.220 ± 1.532	5.43	–	–
	BA	1968.92 ± 82.34	25,582.78 ± 1053.24	28.302 ± 1.183	4.18	–	–
	AHA	1928.74 ± 96.18	24,716.83 ± 1287.54	27.323 ± 1.435	5.25	–	–
	RIME	2018.47 ± 124.83	25,821.47 ± 1438.62	28.557 ± 1.683	5.89	–	–
	E-WOA	1572.18 ± 38.87	24,094.29 ± 618.94	27.148 ± 0.701	2.47	0.64%	40.91%
Average (E-WOA)		1315.81 ± 37.24	19,603.76 ± 529.74	21.04 ± 0.615	2.78	6.04%	25.21%

For the PID controller (Table 4 and Fig. 13), the proposed E–WOA consistently achieves the lowest mean fitness values across all trajectories. Among the conventional and recent benchmarks, AHA emerges as the strongest competitor, achieving the second-lowest fitness on the circular, square, and rhombus paths, while RIME generally lags behind both AHA and the classical algorithms. Notable improvements are observed for the circular and square trajectories, with mean fitness reductions of 12.54% and 10.83%, respectively, relative to the best-performing benchmark. In addition, E–WOA exhibits the lowest CV values for all paths, with CV reductions reaching 40.91%, indicating superior optimization stability and reduced sensitivity to stochastic initialization compared to DE, WOA, PSO, BA, AHA, and RIME.

Table 5: Statistical performance analysis of the FOPID controller optimized using different metaheuristic algorithms over 10 independent simulation runs.

Path	Algorithm	ITAE	Total Energy (J)	Fitness	CV (%)	Improvement (%)	
						Mean	CV
Circle	DE	431.12 ± 25.87	4115.29 ± 242.98	2.657 ± 0.159	6.00	–	–
	WOA	408.15 ± 31.62	4103.84 ± 287.34	2.548 ± 0.197	7.75	–	–
	PSO	428.36 ± 24.16	4167.43 ± 198.65	2.604 ± 0.147	5.64	–	–
	BA	447.92 ± 20.84	4745.67 ± 214.52	2.971 ± 0.138	4.66	–	–
	AHA	392.48 ± 19.37	3946.82 ± 196.74	2.476 ± 0.128	5.17	–	–
	RIME	421.85 ± 26.53	4152.36 ± 241.18	2.682 ± 0.167	6.23	–	–
	E-WOA	305.18 ± 8.34	3087.42 ± 78.65	1.934 ± 0.053	2.73	21.89%	41.42%
8-shaped	DE	1358.46 ± 68.27	24,168.82 ± 1187.54	25.555 ± 1.285	5.03	–	–
	WOA	1312.03 ± 74.48	23,147.82 ± 1284.26	24.405 ± 1.386	5.68	–	–
	PSO	1287.34 ± 71.32	20,354.92 ± 1127.64	21.717 ± 1.203	5.54	–	–
	BA	1527.73 ± 81.92	24,507.33 ± 1337.48	26.523 ± 1.422	5.36	–	–
	AHA	1298.74 ± 56.82	21,847.93 ± 982.45	23.121 ± 1.062	4.59	–	–
	RIME	1362.18 ± 71.24	23,085.47 ± 1247.36	24.457 ± 1.342	5.49	–	–
	E-WOA	1246.58 ± 29.34	18,692.53 ± 412.31	19.734 ± 0.464	2.35	9.12%	48.80%
Square	DE	314.83 ± 22.76	5275.91 ± 368.67	5.626 ± 0.407	7.23	–	–
	WOA	278.57 ± 11.84	5481.80 ± 188.52	5.826 ± 0.248	4.25	–	–
	PSO	1065.63 ± 61.28	25,258.78 ± 1447.36	26.867 ± 1.545	5.75	–	–
	BA	836.74 ± 48.47	26,742.05 ± 1543.68	27.841 ± 1.611	5.79	–	–
	AHA	278.36 ± 16.27	4972.64 ± 287.93	5.140 ± 0.318	6.18	–	–
	RIME	342.71 ± 24.86	6248.37 ± 432.18	6.555 ± 0.467	7.12	–	–
	E-WOA	193.83 ± 6.92	4309.36 ± 153.68	4.580 ± 0.164	3.58	10.89%	15.77%
Rhombus	DE	361.82 ± 24.56	7746.21 ± 527.68	8.406 ± 0.570	6.78	–	–
	WOA	323.26 ± 12.34	7938.47 ± 263.78	8.568 ± 0.327	3.82	–	–
	PSO	349.81 ± 18.92	7928.05 ± 421.34	8.564 ± 0.463	5.40	–	–
	BA	383.42 ± 22.68	7442.79 ± 451.24	8.104 ± 0.480	5.92	–	–
	AHA	328.46 ± 14.93	7218.74 ± 358.62	7.857 ± 0.382	4.86	–	–
	RIME	352.18 ± 22.31	7864.52 ± 438.27	8.557 ± 0.476	5.56	–	–
	E-WOA	319.84 ± 10.16	5755.31 ± 195.78	6.390 ± 0.203	3.18	18.67%	16.75%
Average (E-WOA)		516.86 ± 13.69	7946.16 ± 210.11	8.16 ± 0.221	2.96	15.14%	30.69%

More pronounced gains are obtained for the FOPID controller (Table 5 and Fig. 14). Across all trajectories, E–WOA yields the lowest mean fitness values, achieving improvements of 21.89% for the circular trajectory and exceeding 10% for both square and 18% for the rhombus path. Furthermore, substantial reductions in CV are observed, particularly for the eight-shaped trajectory, where the CV improvement reaches 48.80%. The expanded benchmark set provides additional insight into the relative effectiveness of the proposed algorithm: AHA achieves competitive performance on most trajectories owing to its visit-table-guided foraging mechanism, but its energy consumption remains substantially higher than E-WOA's because its three-mode foraging strategy lacks an explicit mechanism to coordinate exploration intensity with exploitation. RIME, in contrast, exhibits the weakest performance among the recent methods, which

can be attributed to its soft-rime random motion strategy providing less directional guidance than DE's population-informed difference-vector mutation in the five-dimensional FOPID parameter space; furthermore, the hard-rime puncture mechanism, while effective for unimodal problems, can be less reliable in the multimodal landscapes characteristic of energy-aware controller tuning. These results demonstrate that E–WOA not only improves average performance but also significantly enhances robustness and consistency across repeated runs.

Table 6: ANOVA results for fitness values across optimization algorithms (n = 10 runs per algorithm).

Path	Controller	F-Statistic	p-Value	Significant?
Circle	PID	21.53	<0.001	Yes
Circle	FOPID	28.42	<0.001	Yes
8-shaped	PID	9.78	<0.001	Yes
8-shaped	FOPID	14.36	<0.001	Yes
Square	PID	16.84	<0.001	Yes
Square	FOPID	287.91	<0.001	Yes
Rhombus	PID	11.27	<0.001	Yes
Rhombus	FOPID	18.95	<0.001	Yes

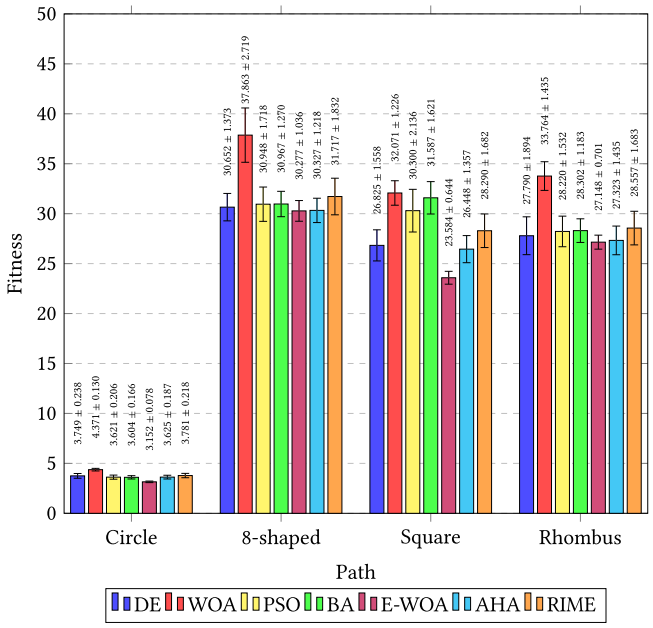


Figure 13: Comparison of fitness values (mean ± SD, n = 10) for the PID controller optimized using different metaheuristic algorithms across various trajectories.

It is also important to highlight that the observed simultaneous reductions in both ITAE and energy consumption across all trajectories confirm the effectiveness of the formulated fitness function defined in (30). With the selected weights for ITAE and energy consumption, the optimization process achieves a well-balanced trade-off between tracking accuracy and energy efficiency. Notably, the numerical results consistently show that the E-WOA-optimized FOPID controller attains lower ITAE and lower energy consumption compared to its PID counterpart, indicating that the adopted weighting strategy is particularly

well suited to exploiting the additional degrees of freedom offered by fractional-order control. The statistical evidence further suggests that these weights are sufficient to suppress excessive tracking error without inducing unnecessary energy expenditure, eliminating the need for further weight tuning.

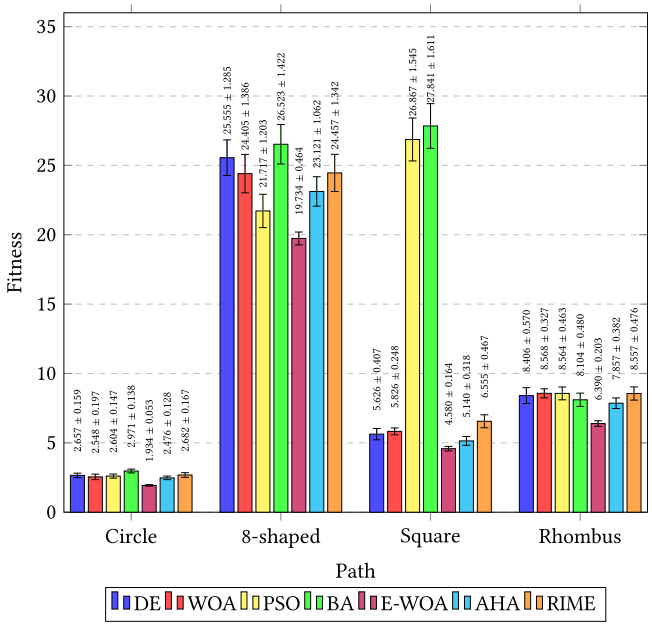


Figure 14: Comparison of fitness values (mean ± standard deviation, $n = 10$) for the FOPID controller optimized using different metaheuristic algorithms across various trajectories.

Overall, the statistical analysis confirms that the proposed E-WOA outperforms all benchmark algorithms in terms of mean performance, robustness, and convergence reliability for both PID and FOPID controllers. More importantly, the E-WOA-optimized FOPID consistently surpasses the E-WOA-optimized PID, as evidenced by the lower fitness values and improved consistency across all trajectories. This demonstrates that the combination of fractional-order control and the enhanced exploration-exploitation balance of E-WOA yields superior generalization capability and energy-aware trajectory tracking compared to conventional integer-order PID control.

3.4 Computational Efficiency Analysis

To assess the practical feasibility of the proposed E-WOA algorithm, the computational cost of each optimization method is analyzed and compared. Table 7 reports the average computation time per optimization run and the total number of fitness function evaluations for each algorithm, measured on an Intel Core i5 processor with 16 GB RAM running MATLAB R2023b.

All algorithms perform $N \times T = 3000$ fitness evaluations, as the E-WOA does not introduce additional function evaluations compared to standalone algorithms—it merely selects between DE and WOA update strategies for each population member at each iteration. The computational overhead of E-WOA relative to the fastest algorithm (WOA) is approximately 7.1%, which arises from the additional probability computation and strategy selection logic. This marginal increase is negligible compared to the dominant cost of fitness evaluation, which involves simulating the full robot trajectory, computing ITAE, and evaluating the energy model for each candidate solution. Given that the optimization is performed offline (i.e., controller parameters are tuned prior to deployment), the slightly higher computation time of E-WOA has no impact on

real-time robot operation and is well justified by the substantial performance improvements demonstrated in the preceding sections.

Table 7: Computational efficiency comparison of optimization algorithms (N = 30, T = 100 iterations, circular trajectory with FOPID controller).

Algorithm	Fitness Evaluations	Avg. Time (s)	Time per Eval. (ms)
DE	3000	142.8	47.6
WOA	3000	138.5	46.2
PSO	3000	141.2	47.1
BA	3000	145.6	48.5
AHA	3000	143.5	47.8
RIME	3000	139.2	46.4
E-WOA	3000	148.3	49.4

3.5 Energy Component Analysis

A detailed energy component analysis is conducted to examine the relative contributions of different energy sources under various trajectory configurations. Table 8 reports the breakdown of kinetic, friction, electrical, and electronic energy components for all controller-algorithm combinations. This analysis provides deeper insight into how optimization and controller structure influence overall energy efficiency.

Table 8: Energy components analysis.

Path	Controller	Algorithm	Kinetic (J)	Friction (J)	Electrical (J)	Electronic (J)
Circle	PID	DE	8.36 ± 0.54	14.38 ± 0.89	3956.74 ± 187.84	1261.70 ± 89.34
		WOA	8.21 ± 0.28	14.63 ± 0.42	4923.32 ± 146.17	1243.47 ± 54.37
		PSO	7.51 ± 0.45	14.58 ± 0.68	3685.36 ± 164.27	1255.82 ± 78.63
		BA	7.76 ± 0.38	14.55 ± 0.56	3665.18 ± 147.26	1261.43 ± 63.72
		AHA	7.85 ± 0.36	14.45 ± 0.62	3815.42 ± 158.74	1210.51 ± 76.43
		RIME	7.92 ± 0.42	14.62 ± 0.74	3960.74 ± 196.85	1254.53 ± 89.27
		E-WOA	7.28 ± 0.23	13.92 ± 0.33	3554.69 ± 62.62	957.75 ± 24.73
	FOPID	DE	6.50 ± 0.42	14.18 ± 0.87	3007.41 ± 167.85	1087.20 ± 84.56
		WOA	7.69 ± 0.54	14.20 ± 0.63	2999.34 ± 208.42	1082.61 ± 76.28
		PSO	5.92 ± 0.38	14.16 ± 0.71	3072.19 ± 145.37	1075.16 ± 61.36
		BA	7.68 ± 0.51	14.18 ± 0.69	3599.17 ± 173.92	1124.64 ± 78.23
		AHA	6.18 ± 0.34	14.05 ± 0.58	2861.83 ± 138.62	1064.76 ± 56.74
		RIME	6.78 ± 0.48	14.12 ± 0.66	3056.19 ± 174.83	1075.27 ± 67.18
		E-WOA	5.84 ± 0.19	13.86 ± 0.32	1993.60 ± 64.89	1074.12 ± 28.58
	PID	DE	59.28 ± 3.48	57.64 ± 3.82	26,213.84 ± 1236.59	1612.02 ± 98.57
		WOA	71.38 ± 4.27	63.28 ± 4.68	32,838.24 ± 1745.46	1405.34 ± 132.73
		PSO	59.67 ± 3.28	58.18 ± 3.51	26,787.28 ± 1382.36	1370.50 ± 88.63
		BA	59.80 ± 2.84	58.15 ± 2.78	26,784.46 ± 1084.18	1393.41 ± 79.67
		AHA	58.74 ± 2.86	57.42 ± 3.18	25,985.76 ± 1054.27	1385.58 ± 84.36
		RIME	62.18 ± 3.74	58.93 ± 3.62	27,395.84 ± 1438.62	1415.07 ± 96.41

(Continued)

Table 8 (continued)

Path	Controller	Algorithm	Kinetic (J)	Friction (J)	Electrical (J)	Electronic (J)
8-shaped	FOPID	E-WOA	58.92 ± 2.13	56.84 ± 2.34	27,463.14 ± 712.78	1139.36 ± 38.18
		DE	83.28 ± 4.76	28.84 ± 1.64	22,929.18 ± 1128.36	1127.52 ± 78.38
		WOA	78.73 ± 4.42	28.82 ± 1.58	22,030.85 ± 1637.92	1009.42 ± 74.10
		PSO	65.11 ± 3.64	28.77 ± 1.52	19,248.76 ± 1007.42	1012.28 ± 67.61
		BA	90.45 ± 5.28	28.99 ± 1.68	23,223.18 ± 1235.06	1164.71 ± 78.24
		AHA	71.84 ± 3.92	28.69 ± 1.46	20,693.42 ± 928.74	1054.05 ± 62.84
		RIME	76.92 ± 4.34	28.78 ± 1.62	21,902.31 ± 1182.46	1077.46 ± 78.62
		E-WOA	64.82 ± 2.76	28.42 ± 1.08	17,592.32 ± 374.39	1006.97 ± 42.83
Square	PID	DE	42.81 ± 2.64	51.23 ± 3.14	22,895.27 ± 1297.79	1229.36 ± 96.42
		WOA	52.37 ± 1.84	56.14 ± 2.18	27,583.63 ± 882.47	1544.15 ± 56.21
		PSO	42.97 ± 2.98	51.78 ± 3.27	26,032.70 ± 1457.39	1415.61 ± 104.33
		BA	43.00 ± 2.36	52.24 ± 3.08	27,278.72 ± 1321.73	1373.42 ± 96.67
		AHA	42.46 ± 2.18	51.42 ± 2.74	22,432.18 ± 1148.36	1290.39 ± 84.36
		RIME	44.18 ± 2.84	52.84 ± 3.42	24,017.43 ± 1382.74	1344.55 ± 96.18
		E-WOA	42.16 ± 1.28	50.92 ± 1.84	19,746.52 ± 543.62	1229.24 ± 34.18
	FOPID	DE	10.95 ± 0.68	9.00 ± 0.54	4102.02 ± 256.30	1153.94 ± 76.70
		WOA	10.81 ± 0.62	9.00 ± 0.48	4298.52 ± 212.76	1163.47 ± 68.17
		PSO	51.53 ± 3.28	57.94 ± 3.45	24,043.96 ± 1336.19	1105.35 ± 78.32
		BA	56.81 ± 3.42	21.32 ± 1.28	25,451.78 ± 1422.54	1212.14 ± 82.61
		AHA	10.62 ± 0.54	8.94 ± 0.42	3812.74 ± 218.36	1140.34 ± 68.74
		RIME	13.74 ± 0.86	10.18 ± 0.62	5063.78 ± 348.92	1160.67 ± 81.46
		E-WOA	10.18 ± 0.38	8.73 ± 0.32	3139.39 ± 111.53	1151.06 ± 32.50
Rhombus	PID	DE	45.88 ± 2.84	52.96 ± 3.28	23,820.28 ± 1443.01	1249.30 ± 98.57
		WOA	53.34 ± 2.18	58.28 ± 2.64	29,150.35 ± 1224.27	1547.07 ± 68.35
		PSO	45.97 ± 2.78	53.46 ± 3.24	24,281.37 ± 1360.92	1126.68 ± 92.33
		BA	46.34 ± 2.16	53.54 ± 2.82	24,135.72 ± 954.13	1347.18 ± 87.36
		AHA	45.62 ± 2.34	53.18 ± 2.78	23,398.74 ± 1187.43	1219.29 ± 84.62
		RIME	46.84 ± 2.84	54.27 ± 3.18	24,430.58 ± 1342.74	1289.78 ± 92.36
		E-WOA	45.52 ± 1.34	52.68 ± 1.56	24,488.52 ± 565.49	1207.57 ± 32.26
	FOPID	DE	15.68 ± 0.94	13.38 ± 0.76	6534.82 ± 457.34	1182.33 ± 78.12
		WOA	16.06 ± 0.76	13.37 ± 0.58	6718.67 ± 247.91	1190.37 ± 56.52
		PSO	14.23 ± 0.81	13.38 ± 0.74	6716.23 ± 342.55	1184.21 ± 67.21
		BA	15.93 ± 0.92	13.37 ± 0.78	6232.09 ± 328.20	1181.40 ± 72.07
		AHA	14.92 ± 0.74	13.18 ± 0.62	6045.36 ± 318.42	1145.28 ± 64.83
		RIME	15.62 ± 0.86	13.27 ± 0.68	6661.83 ± 384.27	1173.80 ± 78.16
		E-WOA	14.08 ± 0.43	13.04 ± 0.36	4520.64 ± 164.70	1107.55 ± 32.23

Across all trajectories, the electrical energy constitutes the dominant portion of total energy consumption, accounting for approximately 78%–92% of the overall energy usage. This dominance highlights the importance of minimizing control effort and actuator demand during trajectory tracking. The proposed

E–WOA–FOPID approach consistently achieves the lowest electrical energy consumption by generating smoother velocity profiles and reducing unnecessary control fluctuations. Kinetic energy exhibits trajectory-dependent behavior, with complex paths such as the eight-shaped trajectory requiring significantly higher kinetic energy (58.92–64.82 J) than the circular trajectory (5.84–7.28 J) due to frequent acceleration and deceleration. The fractional-order controller further contributes to improved energy efficiency by enabling smoother motion transitions and mitigating energy spikes during sharp directional changes.

Friction energy losses remain relatively consistent across different controller implementations but show modest improvements with optimized control strategies. The E-WOA optimization approach achieves slight reductions in friction losses through more efficient path following that minimizes unnecessary velocity variations and direction changes. Electronic subsystem energy consumption shows significant variation across implementations, ranging from approximately 958 to 1612 J, representing the power requirements for sensors, control units, and communication systems. The E-WOA approach achieves notable reductions in electronic energy consumption, particularly for the Circle trajectory with PID controller where it achieves 957.75 J compared to 1261.70 J for conventional DE-PID systems, representing a 24.1% reduction. Similarly, for the 8-shaped trajectory with PID controller, E-WOA reduces electronic energy to 1139.36 J compared to 1612.02 J for DE-PID, achieving a 29.3% improvement.

The energy efficiency analysis reveals that the proposed E-WOA-FOPID approach achieves substantial reductions in total energy consumption compared to the best-performing conventional algorithm for each trajectory. For the Circle path, E-WOA-FOPID achieves 3087.42 J compared to 4115.29 J for DE-FOPID, representing a 25.0% reduction. The 8-shaped trajectory shows E-WOA-FOPID consuming 18,692.53 vs. 20,354.92 J for PSO-FOPID, yielding an 8.2% improvement. The Square trajectory demonstrates the most significant enhancement, with E-WOA-FOPID requiring only 4309.36 J compared to 5275.91 J for DE-FOPID, achieving an 18.3% reduction. For the Rhombus path, E-WOA-FOPID consumes 5755.31 vs. 7442.79 J for BA-FOPID, resulting in a 22.7% improvement. These improvements stem from multiple sources including optimized velocity profiles, reduced control effort requirements, and more efficient trajectory following strategies. The electrical energy component shows the most significant improvements, representing the primary contributor to overall energy savings. The comprehensive energy modeling framework ensures accurate assessment of all significant energy consumption sources during mobile robot operation. The inclusion of kinetic, friction, electrical, and electronic energy components provides a complete picture of system energy requirements and enables targeted optimization strategies for maximum efficiency improvements.

The trajectory-specific energy analysis demonstrates that the optimization benefits are maintained across diverse geometric configurations, indicating robust parameter transferability from the initial circle-based optimization to complex trajectory scenarios. This transferability is crucial for practical implementations where robots must navigate various path geometries without requiring trajectory-specific parameter retuning. The E-WOA approach consistently demonstrates superiority across all energy component categories, with the most pronounced improvements observed in electrical and electronic energy consumption, which together constitute the majority of total energy usage in mobile robot systems.

4 Conclusion and Future Work

This study demonstrates that FOPID controllers optimized using the proposed E-WOA significantly advance mobile robot trajectory tracking by simultaneously improving tracking accuracy and energy efficiency. Compared to the E-WOA-optimized PID controller, the E-WOA-FOPID framework achieves consistently lower tracking error and energy consumption across all tested trajectories. Statistical analysis of the total fitness function, which integrates ITAE and energy consumption, further confirms the robustness of the proposed approach. On average, E-WOA-FOPID achieves a 15.1% improvement in mean fitness and a

30.7% reduction in the CV relative to the best-performing benchmark algorithms (including DE, WOA, PSO, BA, AHA, and RIME), indicating superior optimization quality and consistency. Notably, even when applied to the conventional PID controller, E-WOA still delivers measurable gains, yielding an average improvement of 6.0% in mean fitness and a 25.2% reduction in CV, thereby validating the effectiveness of the proposed optimizer independent of controller structure. These improvements stem from the synergistic integration of DE's global exploration capability with WOA's efficient local exploitation, enabling effective navigation of high-dimensional parameter spaces. The optimized parameters also demonstrate strong generalization capability, transferring reliably from the circular trajectory used for tuning to more complex paths without requiring trajectory-specific retuning.

Despite these promising results, several limitations remain that warrant further investigation to facilitate broader practical deployment and enhance the applicability of the proposed framework. First, all evaluations are based on simulation using a kinematic robot model with simplified motor dynamics; real-world factors such as sensor noise, communication latency, actuator nonlinearities, and wheel slippage may affect performance and are not fully captured. Although the simulation incorporates actuator voltage saturation, back-EMF coupling, friction modeling, and stochastic disturbances as realistic constraints (see Section 2), experimental validation on a physical platform is necessary to confirm that the energy model predictions and controller performance translate to hardware implementations. In particular, unmodeled effects such as gear backlash, temperature-dependent motor resistance variations, and encoder quantization noise may introduce discrepancies between simulated and measured energy consumption. Second, the stochastic disturbance model, while useful for assessing robustness, is a simplified representation of real environmental uncertainties and may not reflect the full spectrum of disturbances encountered in practical deployments. Third, the fitness function weights ($w_1 = 0.6$, $w_2 = 0.4$) were selected based on engineering judgment and validated through sensitivity analysis (Table 3); however, a formal multi-objective Pareto optimization could provide a more comprehensive understanding of the accuracy-energy trade-off surface. Fourth, the computational cost analysis assumes offline parameter tuning; online or real-time adaptation of FOPID parameters has not been addressed, which limits applicability in rapidly changing environments. Fifth, the study considers only planar (2D) trajectories; extension to three-dimensional paths with elevation changes remains to be investigated. Finally, the fractional-order controller implementation relies on the Grunwald-Letnikov approximation with finite memory length, which introduces approximation errors that grow with simulation duration and may affect long-duration operations.

Future research will focus on extending the proposed framework toward real-world deployment and broader application scenarios. Experimental validation on physical mobile robot platforms under realistic conditions, including sensor noise, actuator saturation, and communication delays, will be prioritized to assess robustness beyond simulation environments. In addition, extending the E–WOA–FOPID framework to multi-robot coordination and cooperative trajectory tracking scenarios represents a promising direction for industrial automation and logistics applications. Another important research avenue involves the development of adaptive or self-tuning fractional-order control schemes that can dynamically adjust controller parameters in response to changing environmental conditions, payload variations, or system aging. Furthermore, investigating the integration of the proposed framework with learning-based approaches, such as reinforcement learning for online parameter adaptation, could enable autonomous recalibration of the FOPID controller during operation. The extension of E-WOA to incorporate true multi-objective optimization using Pareto-based approaches would allow systematic exploration of the accuracy-energy trade-off frontier. Additionally, evaluating the framework on three-dimensional trajectories, outdoor terrains with significant elevation changes, and heterogeneous multi-robot teams would broaden the applicability

of the proposed approach. Such extensions would further enhance autonomy and resilience, enabling energy-efficient and precise motion control in highly dynamic and uncertain operational settings.

Acknowledgement: During the preparation of this manuscript, the authors utilized ChatGPT to assist with language refinement and grammatical corrections. All generated content was carefully reviewed and revised by the authors, who take full responsibility for the accuracy, originality, and integrity of the manuscript.

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: Hooi Hung Tang conducted the literature review, methodology, and experiments, analysed the findings, and drafted the manuscript. Te Meng Ting contributed to modeling and interpretation. Nur Syazreen Ahmad supervised the work, refined the review scope, and provided substantive revisions. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Mei Y, Lu YH, Hu YC, Lee CSG. Energy-efficient motion planning for mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation; 2004 Apr 26–May 1; New Orleans, LA, USA. Vol. 5, p. 4344–9.
2. Stefek A, Pham TV, Krivanek V, Pham KL. Energy comparison of controllers used for a differential drive wheeled mobile robot. IEEE Access. 2020;8:170915–27. doi:10.1109/access.2020.3023345.
3. Kubo R, Fujii Y, Nakamura H. Control lyapunov function design for trajectory tracking problems of wheeled mobile robot. In: Proceedings of the 21st IFAC World Congress; 2020 July 11–17; Berlin, Germany, New York, NY, USA: IFAC; 2020. Vol. 53, p. 6177–82.
4. Aguiar J, Costa Nogueira L, Tonidandel F. Linear quadratic regulator path tracking for omnidirectional robots in high-dynamic environments. In: Proceedings of the 2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE); 2023 Oct 9–11; Salvador, Brazil. p. 266–71.
5. Cao T, Pham B, Nguyen N, Vu DL, Truong NV. Second-order terminal sliding mode control for trajectory tracking of a differential drive robot. Mathematics. 2024;12(17):2657. doi:10.3390/math12172657.
6. Yu S, Guo Y, Meng L, Qu T, Chen H. MPC for path following problems of wheeled mobile robots. In: Proceedings of the 6th IFAC Conference Nonlinear Model Predictive Control (NMPC 2018); 2018 Aug 19–22; Madison, WI, USA, New York, NY, USA: IFAC; 2018. Vol. 51, p. 247–52.
7. Ullah I, Adhikari D, Khan H, Anwar MS, Ahmad S, Bai X. Mobile robot localization: current challenges and future prospective. Comput Sci Rev. 2024;53:100651.
8. Minje-Choi, Seongjin-Park, Ryujeong-Lee, Kim S, Juhyeon-Kwak, Lee S. Energy efficient robot operations by adaptive control schemes. Oxf Open Energy. 2024;3:1–11. doi:10.1093/ooenergy/oiae012.
9. Zheng W, Chen Y, Wang X, Lin M, Guo J. Robust fractional order PID controller synthesis for the first order plus integral system. Meas Control. 2023;56(1–2):202–14. doi:10.1177/00202940221095564.
10. Meneses H, Gutierrez M, Arrieta O, Vilanova R. Performance comparison of robust PID and FOPID for an inverse response process model. In: Proceedings of the 4th IFAC Conference on Advances Proportional-Integral-Derivative Control (PID 2024); 2024 Jun 12–14; Almeria, Spain. New York, NY, USA: IFAC; 2024. Vol. 58, p. 13–8.

11. Tepljakov A, Alagoz BB, Yeroglu C, Gonzalez E, HosseinNia SH, Petlenkov E. FOPID controllers and their industrial applications: a survey of recent results. In: Proceedings of the 3rd IFAC Conference on Advances Proportional-Integral-Derivative Control (PID 2018); 2018 May 9–11; Ghent, Belgium. New York, NY, USA: IFAC; 2018. Vol. 51. p. 25–30.
12. Joseph SB, Dada EG, Abidemi A, Oyewola DO, Khammas BM. Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems. Heliyon. 2022;8(5):e09399. doi:10.1016/j.heliyon.2022.e09399.
13. Mahfoud S, Derouich A, El Ouanjli N, El Mahfoud M, Taoussi M. A new strategy-based pid controller optimized by genetic algorithm for DTC of the doubly fed induction motor. Systems. 2021;9(2):37. doi:10.3390/systems9020037.
14. Bhateshvar YK, Vora KC, Mathur HD, Bansal RC. A comparison on PSO optimized PID controller for inter-area oscillation control in an interconnected power system. Technol Econ Dev Smart Grids Sustain Energy. 2022;7(1):7. doi:10.1007/s40866-022-00127-0.
15. Ye K, Li P. A new adaptive PSO-PID control strategy of hybrid energy storage system for electric vehicles. Adv Mech Eng. 2020;12(9):1687814020958574. doi:10.1177/1687814020958574.
16. Chandrababu Naik B, Ravi Kumar CV, Jweeg MJ, Tolephih MH, Hari Prasad M, Mohammed MN. Designing a PID controller using ant colony optimization and implementing it with FPGA. In: Hamdan RK, editor. Advances in intelligent systems and computing. Cham, Switzerland: Springer Nature; 2025. p. 965–76.
17. de Jesus Rubio J. Bat algorithm based control to decrease the control energy consumption and modified bat algorithm based control to increase the trajectory tracking accuracy in robots. Neural Netw. 2023;161(3):437–48. doi:10.1016/j.neunet.2023.02.010.
18. Rana KPS, Kumar V, Sehgal N, George S. A novel dPdI feedback based control scheme using GWO tuned PID controller for efficient MPPT of PEM fuel cell. ISA Trans. 2019;93(12):312–24. doi:10.1016/j.isatra.2019.02.038.
19. Zhao W, Wang L, Mirjalili S. Artificial hummingbird algorithm: a new bio-inspired optimizer with its engineering applications. Comput Methods Appl Mech Eng. 2022;388(1):114194. doi:10.1016/j.cma.2021.114194.
20. Su H, Zhao D, Heidari AA, Liu L, Zhang X, Mafarja M, et al. RIME: a physics-based optimization. Neurocomputing. 2023;532:183–214.
21. Mirjalili S, Lewis A. The whale optimization algorithm. Adv Eng Softw. 2016;95(12):51–67. doi:10.1016/j.advengsoft.2016.01.008.
22. Hasanien HM. Whale optimisation algorithm for automatic generation control of interconnected modern power systems including renewable energy sources. IET Gener Transm Distrib. 2018;12(3):607–14. doi:10.1049/iet-gtd.2017.1005.
23. Zemmit A, Loukriz A, Belhouchet K, Alharthi YZ, Alshareef M, Paramasivam P, et al. GWO and WOA variable step MPPT algorithms-based PV system output power optimization. Sci Rep. 2025;15(1):7810. doi:10.1038/s41598-025-89898-x.
24. Wei J, Gu Y, Yan Y. LSEWOA: an enhanced whale optimization algorithm with multi-strategy. Sensors. 2025;25(7):2054.
25. Dib F, Benaya N, Khaddouj BM, Boumhidi I. Comparative study of optimal tuning PID controller for manipulator robot. In: Ben Ahmed M, Boudhir AA, Santos D, Dionisio R, Benaya N, editors. Innovations in smart cities applications. Vol. 6, Cham, Switzerland: Springer; 2023. p. 252–61.
26. Srisailam C, Manjula M. Optimized FOPID controller for microgrid stability using GWO. SSRG Int J Electr Electron Eng. 2023;10(2):19–34.
27. Tang HH, Ahmad NS. Fuzzy logic approach for controlling uncertain and nonlinear systems: a comprehensive review of applications and advances. Syst Sci Control Eng. 2024;12(1):2394429. doi:10.1080/21642583.2024.2394429.
28. 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.
29. 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;13:48872–87. doi:10.1109/access.2025.3552312.

30. Szelag P, Dudzik S, Podsiedlik A. Investigation on the mobile wheeled robot in terms of energy consumption, travelling time and path matching accuracy. Energies. 2023;16(3):1210. doi:10.3390/en16031210.
31. Jaramillo-Morales MF, Dogru S, Gomez-Mendoza JB, Marques L. Energy estimation for differential drive mobile robots on straight and rotational trajectories. Int J Adv Rob Syst. 2020;17(2):1729881420909654. doi:10.1177/1729881420909654.
32. Vergnano A, Thorstensson C, Lennartson B, Falkman P, Pellicciari M, Leali F, et al. Modeling and optimization of energy consumption in cooperative multi-robot systems. IEEE Trans Autom Sci Eng. 2011;9(2):423–8. doi:10.1109/tase.2011.2182509.
33. Gao X, Su X, An A, Zhang H. Robust model predictive tracking control for the wheeled mobile robot with boundary uncertain based on linear matrix inequalities. IET Cyber Syst Robot. 2023;5(1):e12086. doi:10.1049/csy2.12086.
34. Zheng W, Zhu B. Stochastic time-varying model predictive control for trajectory tracking of a wheeled mobile robot. Front Energy Res. 2021;9:767597. doi:10.3389/fenrg.2021.767597.
35. Alakshendra V, Chiddarwar S. Adaptive robust control of Mecanum-wheeled mobile robot with uncertainties. Nonlinear Dyn. 2017;87(4):2147–69. doi:10.1007/s11071-016-3179-1.
36. Li P, Yang H, Wang S. Model predictive tracking control with disturbance compensation for wheeled mobile robots in an environment with obstacles. J Franklin Inst. 2023;360(10):6669–92. doi:10.1016/j.jfranklin.2023.05.004.
37. Wahab M, Rios-Gutierrez F, El Shahat A. Energy modeling of differential drive robots. In: SoutheastCon 2015; 2015 Apr 9–12; Fort Lauderdale, FL, USA. p. 1–6.
38. Wu M, Fai Y, Su E, Holderbaum W, Yang C. A review on energy efficiency in autonomous mobile robots. Rob Intell Autom. 2023;43(6):648–68. doi:10.1108/ria-05-2023-0060.
39. Liu A, Liu H, Yao B, Xu W, Yang M. Energy consumption modeling of industrial robot based on simulated power data and parameter identification. Adv Mech Eng. 2018;10(5):168781401877385. doi:10.1177/1687814018773852.
40. Hou L, Zhang L, Kim J. Energy modeling and power measurement for mobile robots. Energies. 2018;12(1):27. doi:10.3390/en12010027.
41. Podlubny I editor. Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications. In: Mathematics in science and engineering, Vol. 198, San Diego, CA, USA: Academic Press; 1999.
42. Monje CA, Chen Y, Vinagre BM, Xue D, Feliu V. Fractional-order systems and controls. In: Advances in industrial control. 1st ed. London, UK: Springer; 2010.
43. Storn R, Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim. 1997;11(4):341–59. doi:10.1023/a:1008202821328.
44. Mohammed RH, Aboelmorsy ME, Elnaghi BE. Path tracking control of differential drive mobile robot based on chaotic-billiards optimization algorithm. Int J Electr Comput Eng. 2023;13(2):1449–62. doi:10.11591/ijece.v13i2.pp1449-1462.