scientific reports

Check for updates

OPEN

Reliable parameter estimation of nonlinear chaotic systems and PMSMs with the stellar oscillation optimizer

Serdar Ekinci[1], Davut Izci[2,3]✉, Mostafa Jabari[4], Fahmi Elsayed[5], Mohammad Salman[5] & Burcu Bektaş Güneş[6]

Accurate parameter identification is a critical prerequisite for reliable modeling, analysis, and control of nonlinear dynamical systems. This study introduces the stellar oscillation optimizer (SOO), a recently proposed metaheuristic inspired by the oscillatory behavior of stars, and investigates its effectiveness in estimating system parameters through a unified optimization framework. The identification problem is formulated as the minimization of a trajectory–mismatch cost function, where candidate solutions are iteratively refined by the oscillatory dynamics of SOO. To comprehensively evaluate its performance, four benchmark systems were considered: three canonical chaotic models (Lorenz, Chen, and Rössler) and a practical engineering case represented by a permanent-magnet synchronous motor (PMSM). The outcomes were benchmarked against several state-of-the-art algorithms, including Kirchhoff's law algorithm (KLA), Tianji's horse racing optimization (THRO), puma optimizer (PO), and hiking optimization algorithm (HOA), under a standardized protocol. The results show that SOO consistently achieves numerically convergent solutions with machine-precision-level residuals under deterministic and noise-free simulation settings, while maintaining strong robustness across independent runs. In chaotic benchmarks, the reported residuals approach floating-point limits, which indicates stable numerical convergence rather than guaranteed physical identifiability under real measurement conditions. On the PMSM model, SOO demonstrates accurate and repeatable parameter estimation within the adopted simulation framework.

Keywords Parameter identification, Chaotic attractors, Permanent magnet synchronous motor, Nonlinear systems, Stellar oscillation optimizer

Accurate identification of system parameters constitutes a cornerstone of modeling, simulation, and control in nonlinear dynamical systems. Even minor deviations in parameter estimates can lead to significant discrepancies between predicted and actual trajectories, a problem that is particularly severe in chaotic systems where sensitivity to initial conditions amplifies small errors. Reliable parameter estimation is therefore essential not only for theoretical studies but also for engineering applications in which system stability, performance, and efficiency depend on accurate models.

Traditional estimation techniques, such as gradient-based or analytical methods, often encounter limitations when applied to nonlinear and chaotic systems. Their reliance on smooth objective landscapes and derivative information makes them susceptible to local minima and to divergence when the search space exhibits multimodality or discontinuities. To address these challenges, researchers have increasingly turned to metaheuristic optimization algorithms, which offer derivative-free search mechanisms capable of handling rugged, high-dimensional landscapes.

[1]Department of Computer Engineering, Bitlis Eren University, Bitlis 13100, Turkey. [2]Department of Electrical and Electronics Engineering, Bursa Uludag University, Bursa 16059, Turkey. [3]Applied Science Research Center, Applied Science Private University, Amman 11931, Jordan. [4]Faculty of Electrical Engineering, Sahand University of Technology, Tabriz, Iran. [5]College of Engineering and Technology, American University of the Middle East, Egaila 54200, Kuwait. [6]Department of Computer Engineering, Istanbul Gedik University, İstanbul, Turkey. ✉email: davutizci@gmail.com; davutizci@uludag.edu.tr

Over the years, parameter estimation in canonical chaotic systems such as Lorenz, Chen, and Rössler has remained a central challenge in nonlinear dynamics. The strong sensitivity to initial conditions often renders classical time-domain error functions ill-conditioned, as even small perturbations destroy trajectory similarity. To address this, alternative formulations have been explored, such as return-map or Poincaré-based cost functions, which provide smoother landscapes and improved robustness under noise[1]. These insights motivated the integration of better-posed objectives with metaheuristic search strategies, offering an effective route for handling highly nonlinear dynamics.

Early applications of evolutionary methods (including genetic algorithms (GA) and particle swarm optimization (PSO)) demonstrated the potential of nature-inspired search, yet they often suffered from local entrapment and premature convergence. Hybrid improvements were therefore proposed, combining PSO with GA mutations, simulated annealing, or adaptive mechanisms, which achieved more accurate estimation on chaotic flows by balancing exploration with exploitation[2,3]. Other hybridizations, such as adaptive cuckoo search–PSO schemes, likewise highlighted the advantage of multi-stage search strategies in accelerating convergence while maintaining stability.

In parallel, more recent contributions have employed structural modifications and chaotic mappings to improve search diversity. The improved Lozi-map chaotic optimization algorithm (ILCOA) enhanced exploration–exploitation balance and outperformed DE, PSO, and ABC on Lorenz and Chen benchmarks[2]. Similarly, cellular-topology variants of whale and sine–cosine algorithms accelerated convergence and yielded greater robustness compared to their classical forms or to PSO/DE baselines[3]. The global flower pollination algorithm (GFPA), enriched with chaotic mapping and DE-style local perturbations, achieved near-zero error identification on Lorenz, Chen, and Rössler systems, as well as on hyper-chaotic extensions[4]. Quantum-inspired designs, such as the quantum fruit fly optimization algorithm (QFOA), further confirmed the benefits of stochastic probability search in achieving improved reliability[5].

Beyond single-system tests, some researchers have embedded chaos-theoretic criteria into the identification process. For example, approaches that optimize the Kaplan–Yorke dimension have been employed to validate both parameter accuracy and the intensity of chaos, with applications extending to pseudo-random number generation[6]. Metaheuristics have also been applied to synchronization and control tasks, showing their adaptability in stabilizing nonlinear systems under parameter mismatches and input constraints[7–9]. Collectively, reviews confirm that swarm-based and evolutionary algorithms (including GA, DE, PSO, ABC, GWO, FPA, WOA, and their hybrids) consistently outperform deterministic baselines for nonlinear and chaotic system identification[8,9].

While chaotic flows remain the most common testbeds, real-world applications such as permanent magnet synchronous motors (PMSMs) have recently gained prominence as benchmarks for parameter estimation. PMSM identification is critical because accurate electrical and mechanical parameters directly impact efficiency, torque control, and fault tolerance in modern drives. To this end, a variety of metaheuristic approaches have been proposed. Liu et al. introduced a dynamic self-learning PSO (DSLPSO) for unified electrical and mechanical parameter estimation in PMSMs, reporting improved convergence by integrating opposition-based learning and dynamic exemplars[10]. Zhang et al. developed a logistic sine–chaotic Spider Monkey Optimization (LSOSMO), which combined chaotic mapping with adaptive strategies and achieved parameter estimation errors below 1.1% across multiple operating conditions[11]. Li and Jian proposed a hybrid chaotic RAO-based optimization algorithm (CGCRAO) that combines tent-map initialization and Gaussian–Cauchy variation to enhance PMSM parameter identification, achieving faster convergence and higher accuracy than existing algorithms[12]. More recently, bacterial foraging optimization has been adapted for online PMSM parameter and speed estimation, highlighting the suitability of swarm intelligence under dynamic loads[13]. These studies reinforce that nature-inspired algorithms are increasingly used for PMSM identification, though most works remain system-specific and lack the unified comparative frameworks needed to judge robustness across chaotic and engineering benchmarks.

Despite the breadth of these contributions, notable gaps remain. Many reported methods focus primarily on best-run accuracy, with limited analysis of statistical robustness across multiple trials. Moreover, evaluations often differ in their cost functions, integration steps, and termination criteria, complicating reproducibility and cross-study comparisons. Against this backdrop, the present study contributes by applying the recently developed stellar oscillation optimizer (SOO)[14] under a standardized protocol to both chaotic attractors and PMSM dynamics. In doing so, it not only demonstrates the efficiency of SOO relative to contemporary optimizers but also addresses the need for reproducibility and unified evaluation across theory-driven and practical benchmarks.

The contributions of this work are threefold. First, the paper establishes a unified identification framework that applies the same objective function, integration scheme, and evaluation protocol across all experiments, ensuring fairness in algorithm comparison. Second, it provides a comprehensive benchmarking of SOO against recent metaheuristics (including Kirchhoff's law algorithm[15], Tianji's horse racing optimization[16], puma optimizer[17] and hiking optimization algorithm[18] across three chaotic attractors (Lorenz, Chen, and Rössler) and one practical engineering system. Third, it conducts a literature-level comparison with previously published results on the same benchmarks, positioning SOO's outcomes within the broader landscape of parameter estimation studies.

Extensive numerical tests were conducted under standardized conditions, where each algorithm was evaluated over multiple independent runs to assess not only accuracy but also robustness. The results consistently showed that SOO recovered parameters to within numerical precision under the adopted deterministic simulation settings, with residual errors approaching machine precision, while maintaining remarkable stability across trials. In both chaotic systems and the engineering benchmark, SOO exhibited faster convergence, smoother parameter trajectories, and narrower statistical dispersions compared to the alternative algorithms. These

findings underline its reliability in navigating complex identification landscapes where other methods often show variability or premature stagnation.

For a broader perspective, the proposed approach was also compared against algorithms previously reported in the literature, including evolutionary programming, differential evolution hybrids, chaos-enhanced invasive weed optimization, and whale or sine–cosine variants[2–4,19,20]. While many of these methods demonstrated competitive performance in specific scenarios, they often struggled with reproducibility, convergence speed, or precision in parameter recovery. In contrast, SOO consistently provided lower residuals and tighter run-to-run dispersion across the tested systems, combining high accuracy with strong repeatability. These results highlight its potential as a versatile and dependable optimization tool for parameter identification tasks in both theoretical and practical contexts. The remainder of this article is organized in a clear sequence. Section "Stellar oscillation optimizer" introduces the SOO and describes its solution-updating process. Section "Problem definition and application of SOO" formulates the nonlinear parameter identification problem and explains how SOO is applied within the estimation framework. Section "Simulation results" presents the simulation studies and associated analyses. These include four benchmark systems (Lorenz, Chen, Rössler, and a permanent magnet synchronous motor), comparative evaluations against state-of-the-art optimizers, nonparametric statistical validation, computational efficiency assessment, and a literature-based performance comparison, followed by a synthesized discussion of the findings. Lastly, Sect. "Conclusion" concludes the study and outlines directions for future research.

Stellar oscillation optimizer

The stellar oscillation optimizer (SOO) employs a mathematical framework that reproduces the oscillatory behavior of stars, enabling candidate solutions to move dynamically in the search space[14]. The model defines the initialization, oscillation dynamics, and update rules of the oscillators (solutions). Let the number of oscillators be N_{osc}, operating in a D-dimensional space bounded by lower and upper limits $[lb, ub]$. The positions of oscillators are stored in a matrix $X \in \mathbb{R}^{N_{osc} \times D}$, and the quality of each position $x \in \mathbb{R}^{D}$ is evaluated using a fitness function $f(x)$. The initial positions of the oscillators are randomly distributed within the search domain as:

$$X_{init} = initialization(N_{osc}, D, ub, lb) \tag{1}$$

The best luminosity and corresponding best position are initialized as:

$$L_{best} = \infty, x_{best} = 0^{D} \tag{2}$$

where L_{best} denotes the best fitness value (luminosity), and x_{best} is a zero vector of dimension D. The oscillatory period evolves with iterations as:

$$P(t) = P_0 + \Delta P \cdot t \tag{3}$$

where P_0 is the initial period and ΔP is the incremental change. The angular frequency is then obtained as:

$$w(t) = \frac{2\pi}{P(t)} \tag{4}$$

The scaling factor moderates oscillation intensity:

$$S(t) = 2 \cdot \left(1 - \frac{t}{T}\right) \tag{5}$$

with T representing the maximum iteration number. $S(t)$ decreases over time, encouraging a gradual transition from exploration to exploitation. For oscillator i, two candidate positions are generated. The first candidate is:

$$x_j^{osc1} = x_j^{best} - r_1 r_3 \left(w(t) S(t) r_1 - S(t)\right) \left(x_i(j) - \left|r_1 \sin(r_2) \cdot \left|r_3 \cdot x_{best}^{(j)}\right|\right|\right) \tag{6}$$

and the second candidate is:

$$x_j^{osc2} = x_j^{best} - (r_2 r_3) w(t) S(t) r_1 - S(t) \left(x_i(j) - \left|r_1 \cos(r_2) \cdot \left|r_3 \cdot x_{best}^{(j)}\right|\right|\right) \tag{7}$$

where $r_1, r_2, r_3 \sim U(0,1)$ are independent random numbers. The final position is obtained by averaging the two candidates with a stochastic weight:

$$x_j^{new} = r_3 \cdot \frac{x_j^{osc1} + x_j^{osc2}}{2} \tag{8}$$

At each iteration, the mean position of the top three oscillators guides the others:

$$x_{avg} = \frac{1}{3}\sum_{k=1}^{3} x^{top,k} \tag{9}$$

A new oscillatory position is then generated as:

$$x^{osc} = x_{avg} + 0.5\left[\sin\left(r\pi\right)\left(x_{r1} - x_{r2}\right) + \cos\left(\left(1 - r\right)\pi\right)\left(x_{r1} - x_{r3}\right)\right] \tag{10}$$

where x_{r1}, x_{r2}, x_{r3} are randomly chosen oscillators and $r \sim U\left(0,1\right)$. Each dimension is updated probabilistically:

$$x_j^{new} = \begin{cases} x_j^{osc}, & if\ r_j < 0.5 \\ x_j^{old}, & otherwise \end{cases} \tag{11}$$

If a new position improves upon the current best fitness, both luminosity and position are updated:

$$L(x^{new}) < L_{best} \Rightarrow L_{best} = L(x^{new}), \quad x_{best} = x^{new} \tag{12}$$

Figure 1 provides a detailed flowchart explaining the working principle of SOO in light of the above description. It is also worth noting that unlike conventional oscillatory optimizers (e.g., algorithms driven solely by sinusoidal or cosine-based updates[21], SOO incorporates three distinct operators that jointly shape its exploration–exploitation behavior. The update rule in Eqs. (6)–(7) integrates a direct influence from the current best candidate, enabling directional exploitation around promising positions rather than random oscillatory wandering (top-oscillator–guided motion). The simultaneous consideration of two randomly selected candidates introduces a mechanism for adaptive midpoint searching. This structured averaging increases the likelihood of sampling within the basin of the global optimum without collapsing diversity prematurely (dual-sample interaction with weighted averaging). Lastly, as defined in Eq. (5), the gradual decay of $S\left(t\right)$ shifts SOO from large-radius oscillations in early iterations to fine-scale refinement later, preventing stagnation while ensuring stable convergence (a time-decreasing oscillation amplitude). These characteristics together differentiate SOO from earlier oscillation-inspired methods such as the sine cosine algorithm (SCA)[22], which relies on periodic perturbation alone, and provides a mathematically explicit control of the trade-off between exploration and exploitation.

Problem definition and application of SOO

The estimation of parameters in nonlinear chaotic systems can be formally expressed as a multi-dimensional optimization problem[23]. Considering an n-dimensional nonlinear/chaotic system, the state dynamics are described as[24]:

$$\dot{X} = F(X, \theta) \tag{13}$$

where $X = [x_1, x_2, \ldots, x_n]^T$ represents the state variables and $\theta = [\theta_1, \theta_2, \ldots, \theta_n]^T$ denotes the unknown system parameters to be estimated. An estimated system is constructed in parallel as[24]:

$$\dot{Y} = F(Y, \tilde{\theta}) \tag{14}$$

with $Y = [y_1, y_2, \ldots, y_n]^T$ representing the simulated state vector and $\tilde{\theta}$ the set of candidate parameter estimates. The estimation problem is formulated as the minimization of the objective (cost) function[24]:

$$J = \frac{1}{M}\sum_{k=1}^{M} \|X_k - Y_k\|^2 \tag{15}$$

where M is the length of the time series, and X_k and Y_k are the sampled states of the original and estimated systems, respectively. The task of parameter identification therefore reduces to the search for $\tilde{\theta}$ that minimizes J. It should be noted that the objective function in Eq. (15) is evaluated under finite-precision floating-point arithmetic together with fixed-step numerical integration. For deterministic and noise-free simulations, when simulated and reference trajectories become numerically indistinguishable at sampled time points, the residual cost may decrease to machine-precision levels. Such extremely small values reflect numerical convergence of the optimization–simulation pipeline rather than strict physical identifiability. The attainable residual magnitude is therefore bounded by floating-point precision, numerical integration tolerances, and the conditioning of trajectory matching in chaotic systems.

To compute the state trajectories, the nonlinear chaotic system is integrated using the fourth-order Runge–Kutta (RK4) scheme with a fixed sampling interval of $\Delta t = 0.01$. For each simulation, $M = 300$ sampling points are considered, ensuring sufficient representation of the system's dynamic evolution. The SOO is then employed to carry out the parameter estimation. Within this approach, each candidate solution corresponds to an oscillator, whose position is updated according to the oscillatory dynamics of the algorithm. The optimization process alternates between exploration, enabled by stochastic oscillatory motion, and exploitation, achieved by refining the positions toward the best-performing oscillators. At each iteration, the estimated parameters are substituted into the chaotic model, and the discrepancy between the simulated and reference trajectories is quantified by the objective function. The process continues until the stopping criterion, typically a maximum

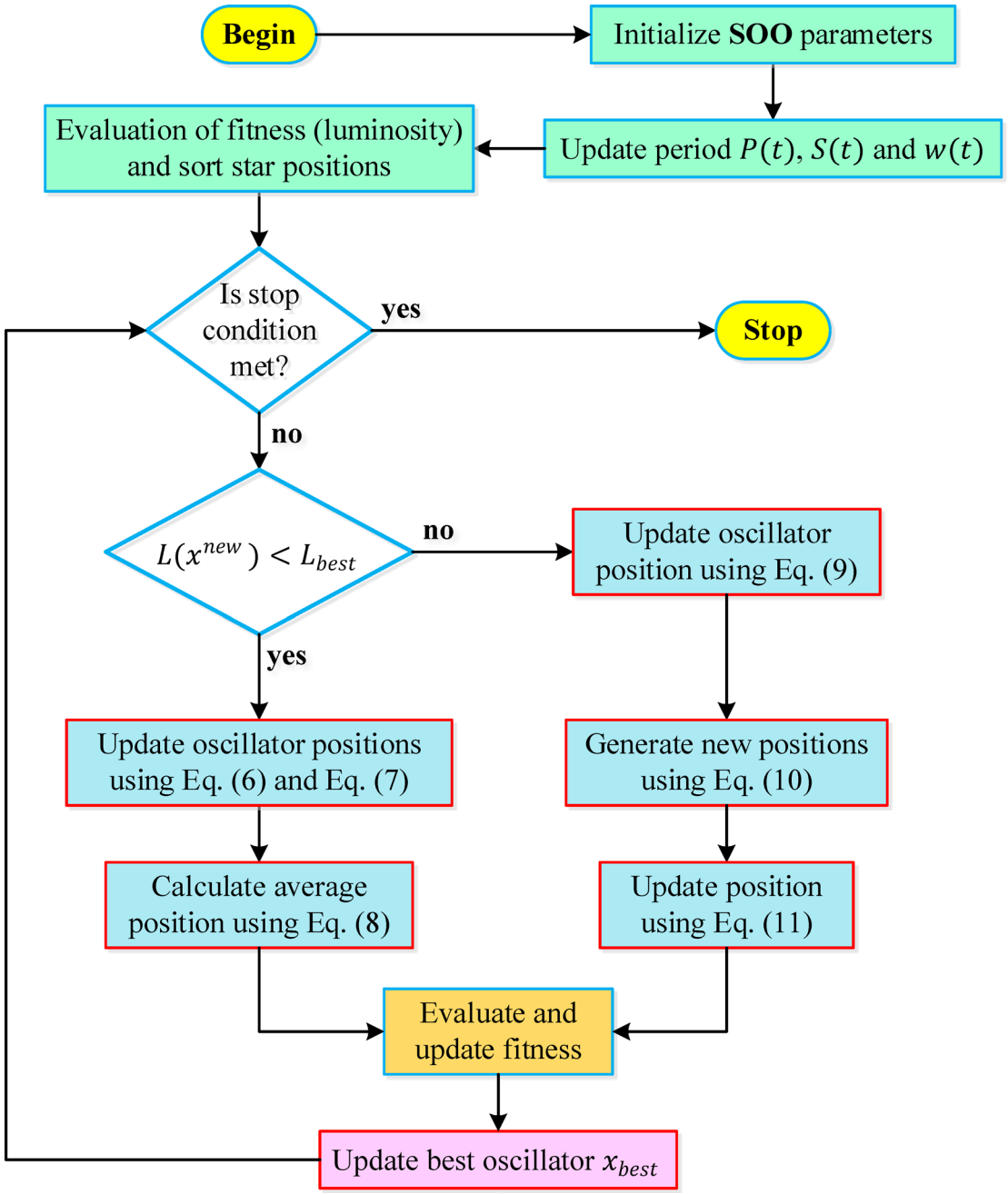

Fig. 1. Flowchart of SOO.

number of iterations or a convergence threshold, is satisfied. Figure 2 demonstrates the principle of parameter identification for nonlinear/chaotic systems.

Simulation results

To rigorously evaluate the effectiveness of the SOO in parameter identification, a series of experiments was carried out on both chaotic and engineering benchmark systems. The test suite comprised three well-known chaotic attractors (Lorenz, Chen, and Rössler chotic systems) and a practical electromechanical model represented by a permanent magnet synchronous motor. All algorithms were executed under a unified protocol, with fixed population size (30), iteration budget (200), and independent runs (25) to ensure fair and reproducible comparisons. The adopted algorithms used for comparisons are stellar oscillation optimizer (SOO)[14], Kirchhoff's law algorithm (KLA[15], Tianji's horse racing optimization (THRO)[16], puma optimizer (PO)[17] and hiking optimization algorithm (HOA)[18]. The following subsections present detailed results for each system, supported by statistical analyses, convergence curves, and parameter trajectory evaluations, highlighting both the accuracy and robustness of the proposed approach relative to competing optimizers.

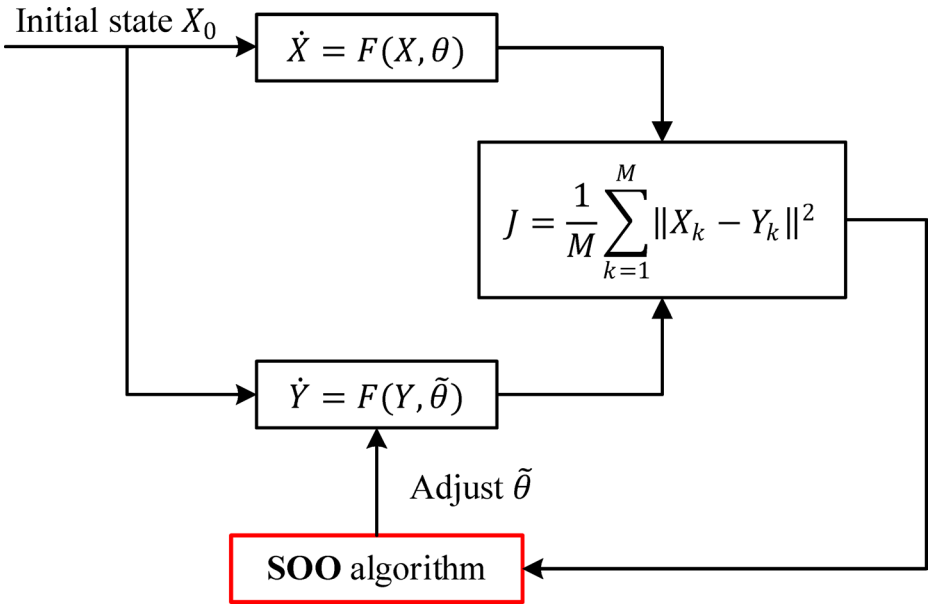

Fig. 2. Parameter identification principle of nonlinear/chaotic systems.

Algorithm	Best	Worst	Mean	Standard deviation	Rank
SOO	0	2.2513×10^{-28}	5.4294×10^{-29}	8.3596×10^{-29}	1
KLA	8.0568×10^{-08}	3.5108×10^{-05}	4.6708×10^{-06}	7.7909×10^{-06}	4
THRO	1.5927×10^{-18}	1.0486×10^{-08}	5.5106×10^{-10}	2.1024×10^{-09}	2
PO	1.7607×10^{-13}	6.0885×10^{-06}	5.8032×10^{-07}	1.6114×10^{-06}	3
HOA	1.8932×10^{-28}	5.6211×10^{-05}	5.2831×10^{-06}	1.3267×10^{-05}	5

Table 1. Statistical results obtained from SOO, KLA, THRO, PO and HOA for Lorenz chaotic system.

Test system I: Lorenz chaotic system

The Lorenz attractor was selected as the first benchmark to evaluate parameter–estimation performance under pronounced nonlinear and chaotic dynamics. The system is governed by Eqs. (16)–(18), with the parameter vector $[a, b, c]$ and the state vector initialized as $x(0) = 0.1$, $y(0) = 0.1$ and $z(0) = 0$. The parameter ranges were considered as $0 \le a \le 20$, $0 \le b \le 50$ and $0 \le c \le 5$ and the ground-truth parameter values were set to $a = 10$, $b = 28$ and $c = 8/3$. Across all competing algorithms, the experimental protocol fixed the population size at 30, the maximum iteration number at 200, and the number of independent runs at 25, while the objective function of Eq. (15) was minimized using the RK4-based trajectory simulation described previously. The compared methods included the SOO, KLA, THRO, PO, and HOA).

$$\dot{x} = a(y - x) \tag{16}$$

$$\dot{y} = bx - xz - y \tag{17}$$

$$\dot{z} = xy - cz \tag{18}$$

Table 1 summarizes the distributional statistics of the achieved objective-function values over 25 runs. SOO attained the best overall rank (1st), with a best cost of 0, and mean and standard deviation on the order of 10^{-29}, indicating both accuracy and exceptional run-to-run stability. THRO ranked 2nd, followed by PO (3rd) and KLA (4th), whereas HOA showed the weakest robustness in this experiment (5th). The near-zero dispersion observed for SOO demonstrates that convergence to the global optimum is not a rare event but a consistently reproducible outcome under the adopted settings.

The parameter estimates corresponding to the best run of each algorithm are reported in Table 2. SOO, THRO, PO, and HOA all identified the ground-truth values $a = 10$, $b = 28$, and $c = 2.666667$ (i.e., $8/3$) within numerical precision, whereas KLA produced estimates that were very close but not identical to the true parameters, in line with its nonzero best cost. These results confirm that, for the Lorenz system, the searched parameter surface contains a sharp optimum that can be precisely located when the optimizer balances exploration and exploitation effectively.

The convergence curves in Fig. 3 further elucidate the dynamics of the search. SOO's cost decreased rapidly toward zero and stabilized early, surpassing the rates observed for the other methods. THRO and PO also

Algorithm	Best cost	a	b	c
SOO	0	10.000000	28.000000	2.666667
KLA	8.0568×10^{-08}	9.999998	27.999920	2.666642
THRO	1.5927×10^{-18}	10.000000	28.000000	2.666667
PO	1.7607×10^{-13}	10.000000	28.000000	2.666667
HOA	1.8932×10^{-28}	10.000000	28.000000	2.666667

Table 2. Obtained parameters of Lorenz chaotic system and best cost values using SOO, KLA, THRO, PO and HOA.

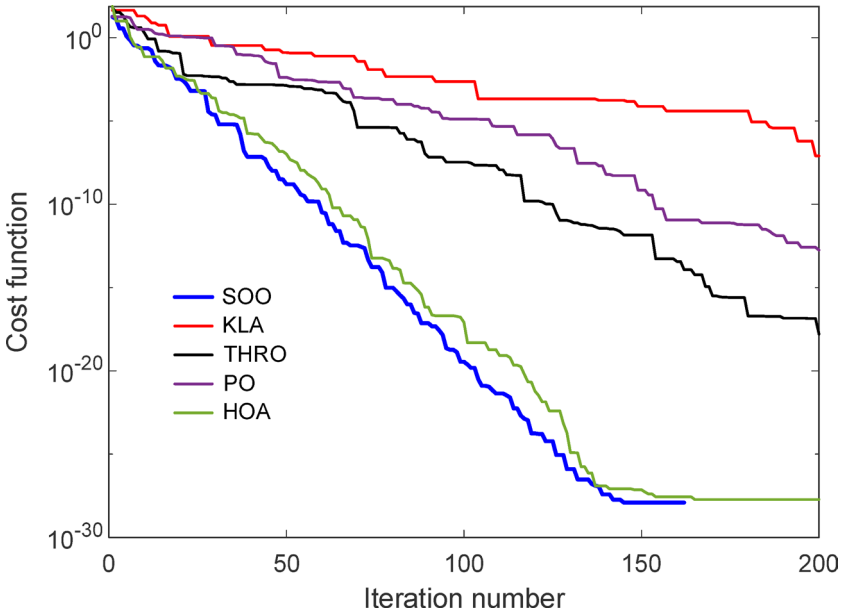

Fig. 3. Convergence of cost function for Lorenz chaotic system using SOO, KLA, THRO, PO and HOA.

demonstrated competent progress but required more iterations to approach their respective minima, while KLA and HOA displayed slower descent and larger variability before plateauing. The consistently steeper trajectory of SOO indicates an efficient guidance mechanism toward elite regions of the search space, which is consistent with the oscillator-driven updates described earlier in Sect. "Stellar oscillation optimizer".

Figures 4, 5 and 6 depict the evolution of the individual parameters a, b, and c over iterations. For SOO, each parameter trajectory converged smoothly to its true value with minimal oscillation once the vicinity of the optimum was reached. In contrast, competing methods exhibited either slower approach or small persistent fluctuations around the target values, matching their slightly higher residual costs. The stability of SOO's parameter paths illustrates how the algorithm's oscillatory search (combined with its gradually tightening scaling) facilitates decisive refinement near the optimum while suppressing late-stage wander.

The Lorenz system is notoriously sensitive to small perturbations; consequently, recovery to within numerical precision under deterministic integration is a stringent test of estimator fidelity. Achieving a best cost reported as zero under floating-point precision and virtually vanishing dispersion across repeated trials establishes that SOO not only finds a correct solution but does so reliably. Together, the statistical ranks (Table 1), parameter recovery (Table 2), rapid descent (Fig. 3), and well-behaved parameter trajectories (Figs. 4, 5 and 6) indicate that SOO's search dynamics are well matched to the rugged, multimodal landscape induced by Eqs. (16)–(18), yielding both accuracy and robustness in this canonical chaotic benchmark.

Test system II: Chen chaotic system

The Chen attractor was adopted as the second benchmark to examine estimator performance under strongly nonlinear, non-Lorenzian chaotic dynamics. The system is governed by Eqs. (19)–(21), with the decision vector $[a, b, c]$ searched over the ranges $20 \leq a \leq 50$, $1 \leq b \leq 6$ and $10 \leq c \leq 40$. The ground-truth parameters were set to $a = 35$, $b = 3$ and $c = 28$; initial states were chosen as specified beneath Eqs. (19)–(21). As in Sect. "Test system I: Lorenz chaotic system", trajectories were computed by the RK4 scheme and the objective function of Eq. (15) was minimized, using 25 independent runs per method with a population size of 30 and 200 iterations. The compared methods were SOO, KLA, THRO, PO, and HOA.

$$\dot{x} = a(y - x) \tag{19}$$

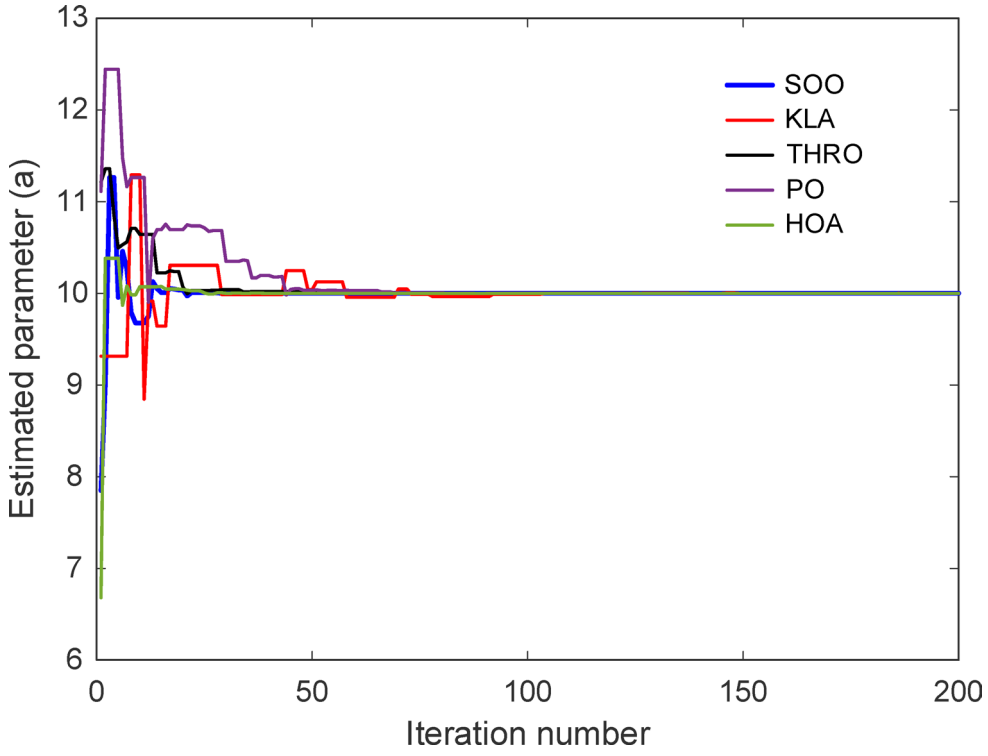

Fig. 4. Change of parameter a for Lorenz chaotic system using SOO, KLA, THRO, PO and HOA.

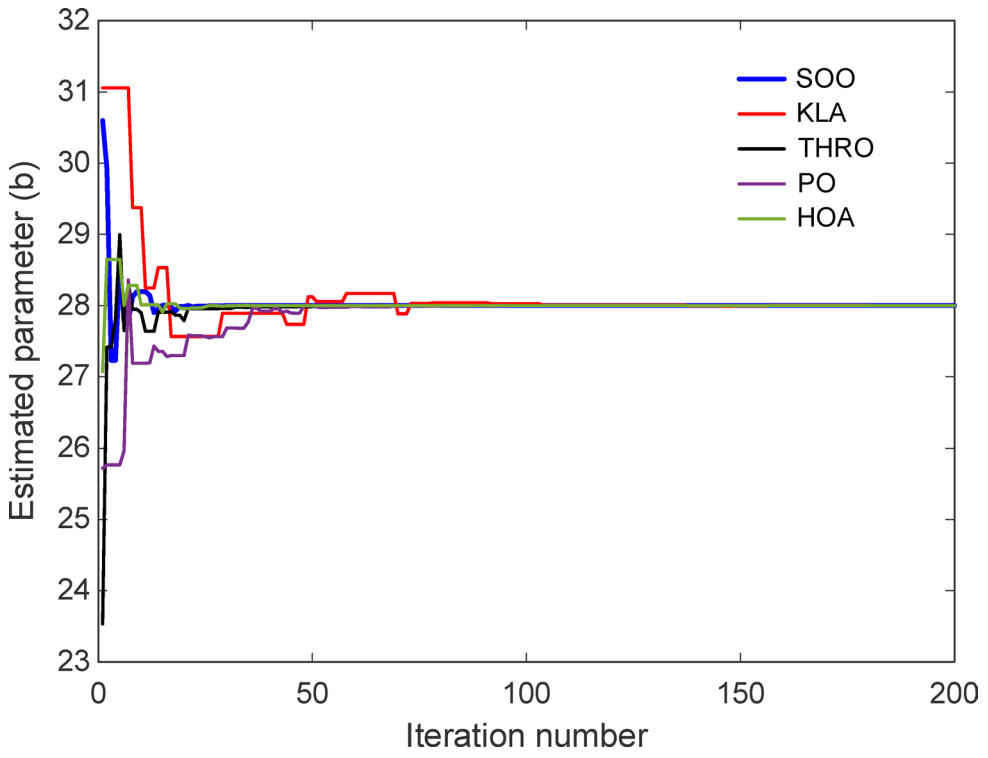

Fig. 5. Change of parameter b for Lorenz chaotic system using SOO, KLA, THRO, PO and HOA.

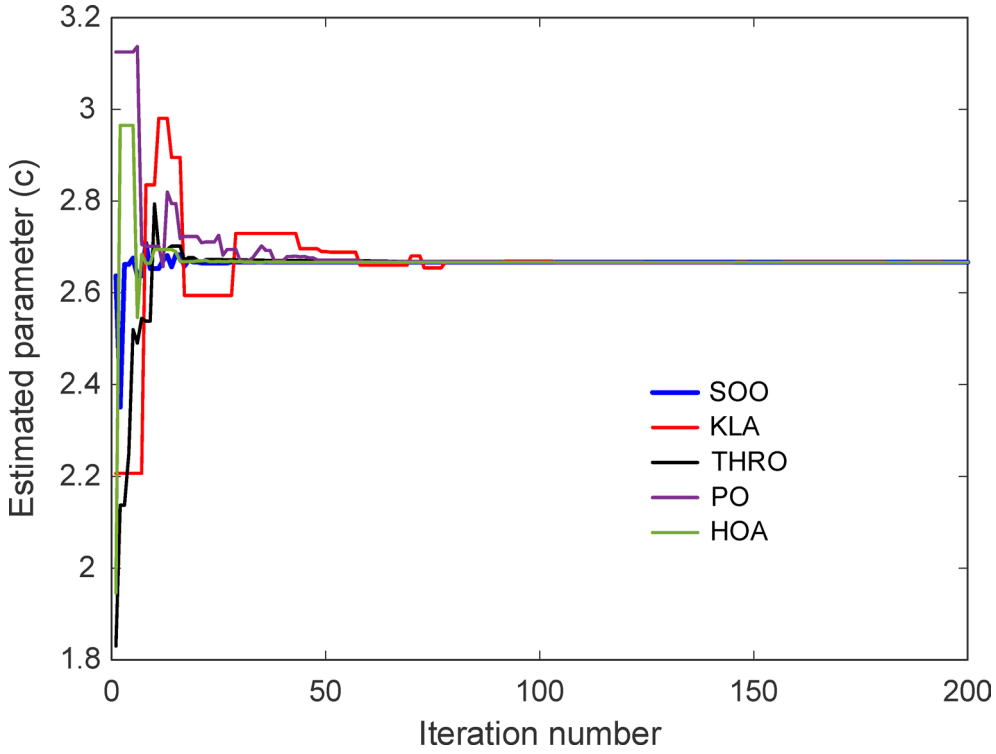

Fig. 6. Change of parameter c for Lorenz chaotic system using SOO, KLA, THRO, PO and HOA.

Algorithm	Best	Worst	Mean	Standard deviation	Rank
SOO	0	0	0	0	1
KLA	2.9459×10^{-12}	2.1977×10^{-10}	5.8911×10^{-11}	5.1915×10^{-11}	4
THRO	2.2695×10^{-22}	4.9918×10^{-11}	2.0674×10^{-12}	9.9719×10^{-12}	3
PO	5.9507×10^{-23}	5.7460×10^{-15}	3.7783×10^{-16}	1.2838×10^{-15}	2
HOA	7.5741×10^{-27}	5.4345×10^{-07}	2.9591×10^{-08}	1.1264×10^{-07}	5

Table 3. Statistical results obtained from SOO, KLA, THRO, PO and HOA for Chen chaotic system.

$$\dot{y} = (c-a)\,x + cy - xz \tag{20}$$

$$\dot{z} = xy - bz \tag{21}$$

Table 3 reports the distributional statistics of the objective value across runs. SOO achieved a best cost reported as zero under floating-point precision and, notably, a worst, mean, and standard deviation also reported as zero under floating-point arithmetic, indicating complete run-to-run reproducibility under the adopted deterministic settings. PO ranked 2nd with a best of 5.9507E – 23 and low dispersion; THRO ranked 3rd with a best of 2.2695E – 22 but slightly larger variability; KLA ranked 4th with higher central tendency (5.8911E – 11); and HOA ranked 5th, exhibiting the largest spread (worst 5.4345E – 07, mean 2.9591E – 08). These outcomes demonstrate that while several competitors occasionally approach machine-precision minima, SOO is uniquely robust, returning the optimum in every run.

The best-run estimates in Table 4 show that SOO, THRO, PO, and HOA recovered the ground-truth parameters with high numerical accuracy, consistent with their near-zero best costs. KLA yielded values extremely close to the truth yet not exact, in agreement with its nonzero best cost. This pattern confirms that the Chen landscape admits precise identification and that algorithms capable of reliable fine-scale exploitation can attain recovery.

Figure 7 displays the decline of the objective with iterations. SOO's curve rapidly collapsed to zero and remained flat thereafter, evidencing fast transient and immediate stabilization. PO and THRO also converged effectively but required more iterations to approach their minima and exhibited small late-stage fluctuations. KLA and HOA descended more slowly and showed greater variability before plateauing, which is consistent with their weaker statistics in Table 3.

The iterative evolution of a, b, and c is presented in Figs. 8, 9 and 10. Under SOO, each parameter path moved decisively toward the true value and settled with negligible residual oscillation once in the neighborhood of

Algorithm	Best cost	a	b	c
SOO	0	35.000000	3.000000	28.000000
KLA	2.9459×10^{-12}	35.000452	3.000000	28.000385
THRO	2.2695×10^{-22}	35.000000	3.000000	28.000000
PO	5.9507×10^{-23}	35.000000	3.000000	28.000000
HOA	7.5741×10^{-27}	35.000000	3.000000	28.000000

Table 4. Obtained parameters of Chen chaotic system and best cost values using SOO, KLA, THRO, PO and HOA.

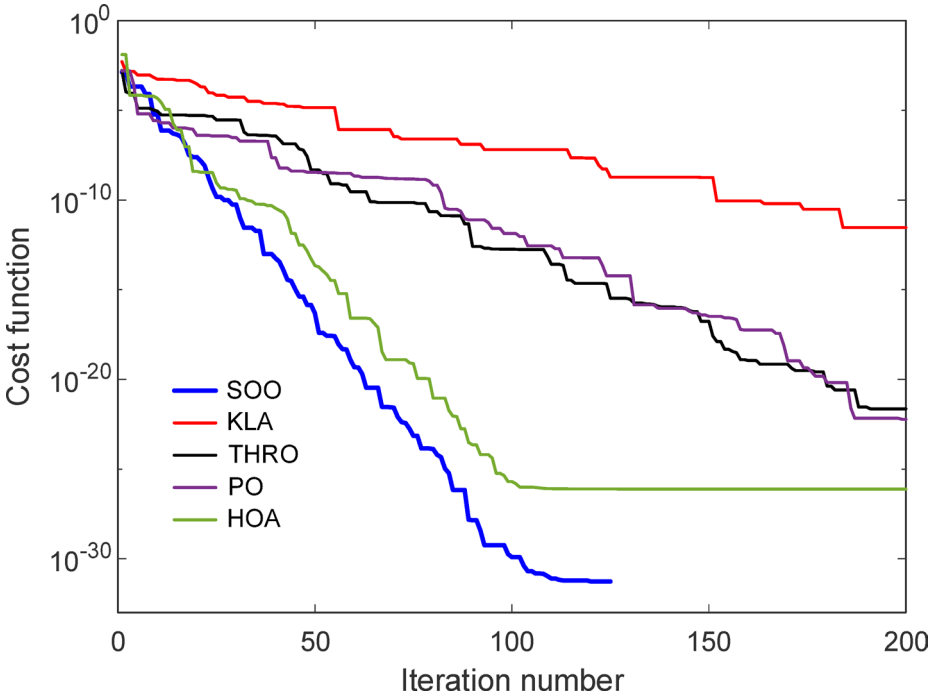

Fig. 7. Convergence of cost function for Chen chaotic system using SOO, KLA, THRO, PO and HOA.

the optimum. Competing methods exhibited either slower approach or modest steady-state ripples, mirroring their slightly higher residual costs. The smooth terminal behavior of SOO's parameter updates is consistent with its oscillation-based search coupled with a time-decreasing scaling factor, which collectively promote decisive refinement without over-exploration near the solution.

Taken together (the perfect statistics (Table 3), best-run recovery (Table 4), fastest and most stable convergence (Fig. 7), and well-behaved parameter paths (Figs. 8, 9 and 10)) the results indicate that SOO is both accurate and exceptionally robust for the Chen system. Relative to THRO and PO, SOO delivers comparable peak accuracy while eliminating run-to-run variability; relative to KLA and HOA, it improves both convergence speed and final precision by orders of magnitude. The evidence thus supports SOO as the most reliable estimator among the tested approaches for this chaotic benchmark.

Test system III: Rössler chaotic system

The Rössler attractor was selected as the third benchmark to assess estimator behavior under continuous-time chaotic dynamics with a spiraling manifold. The governing model is given by Eqs. (22)–(24), and the true parameter vector is $[a, b, c]$=[0.2, 0.2, 5.7]; initial conditions were set as $x(0) = 1$, $y(0) = 1$ and $z(0) = 1$ and a search range of $0 \leq a \leq 1, 0 \leq b \leq 1$ and $0 \leq c \leq 10$ were used. As in Sect. "Test system I: Lorenz chaotic system" and "Test system II: Chen chaotic system", trajectories were generated via the fourth-order Runge–Kutta (RK4) integrator and the objective function in Eq. (15) was minimized. All algorithms were executed for 25 independent runs with a population size of 30 and 200 iterations (SOO, KLA, THRO, PO, HOA).

$$\dot{x} = -y - z \tag{22}$$

$$\dot{y} = x + ay \tag{23}$$

$$\dot{z} = b + (x - c)z \tag{24}$$

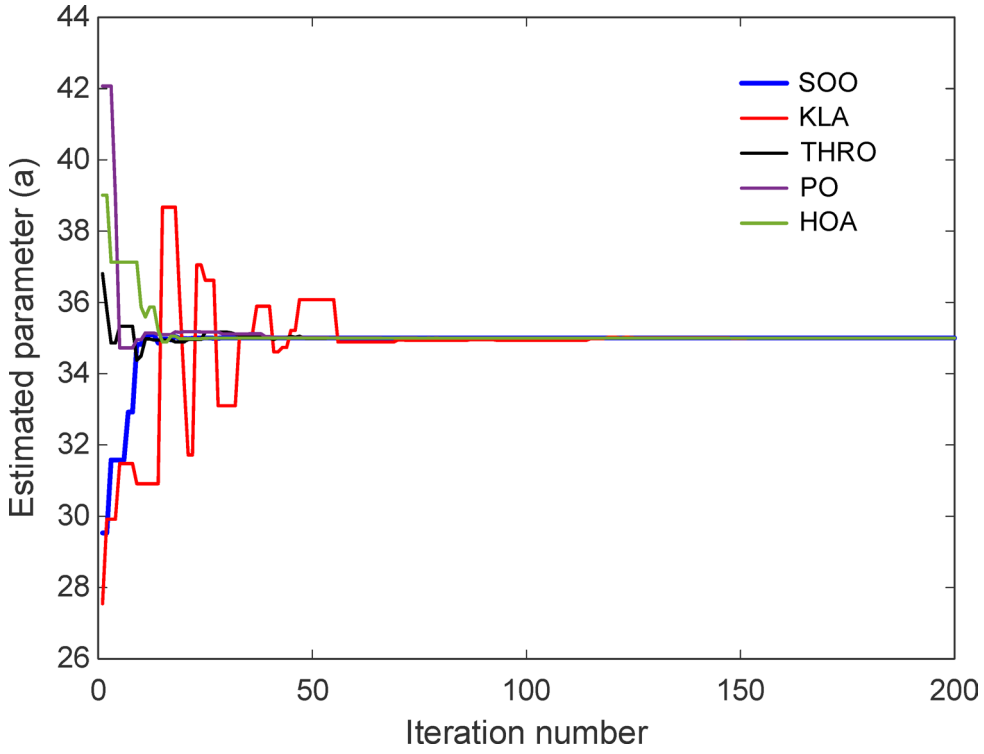

Fig. 8. Change of parameter a for Chen chaotic system using SOO, KLA, THRO, PO and HOA.

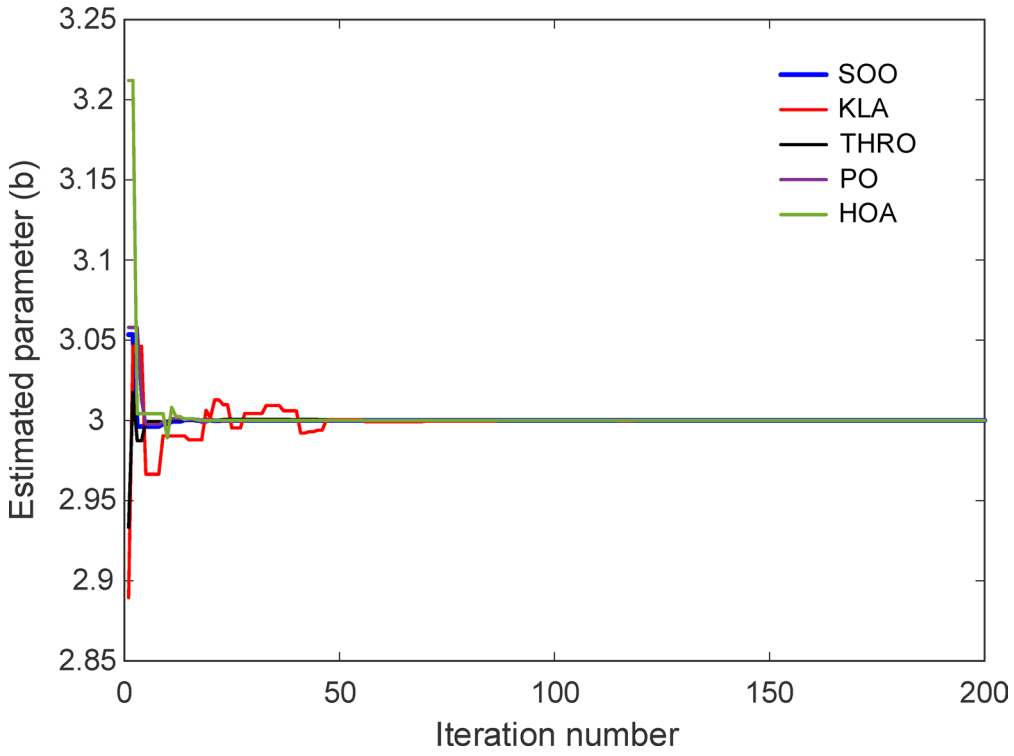

Fig. 9. Change of parameter b for Chen chaotic system using SOO, KLA, THRO, PO and HOA.

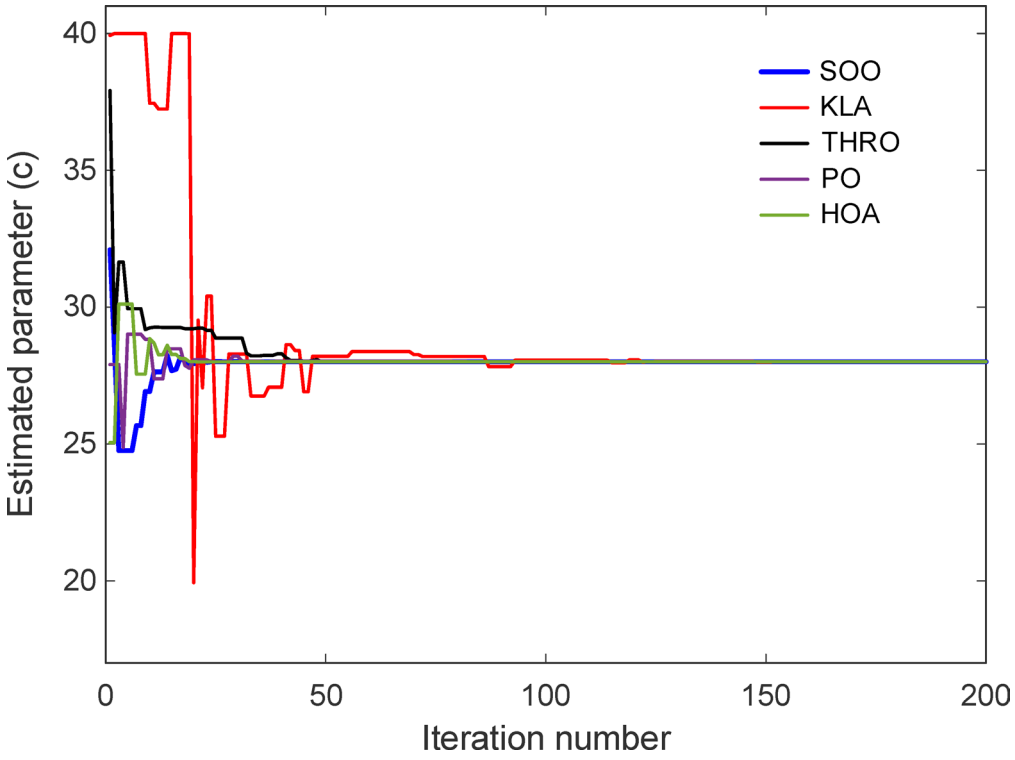

Fig. 10. Change of parameter c for Chen chaotic system using SOO, KLA, THRO, PO and HOA.

Algorithm	Best	Worst	Mean	Standard deviation	Rank
SOO	3.5352×10^{-32}	1.9612×10^{-30}	3.2797×10^{-31}	4.7287×10^{-31}	1
KLA	1.2918×10^{-12}	1.6323×10^{-09}	3.0689×10^{-10}	4.4554×10^{-10}	2
THRO	3.2640×10^{-18}	1.0004×10^{-05}	5.4997×10^{-07}	2.0344×10^{-06}	4
PO	3.2290×10^{-16}	7.2085×10^{-07}	9.2155×10^{-08}	1.9916×10^{-07}	3
HOA	1.0906×10^{-16}	7.1617×10^{-05}	5.0106×10^{-06}	1.5377×10^{-05}	5

Table 5. Statistical results obtained from SOO, KLA, THRO, PO and HOA for Rössler chaotic system.

Algorithm	Best cost	a	b	c
SOO	3.5352×10^{-32}	0.200000	0.200000	5.700000
KLA	1.2918×10^{-12}	0.199999	0.200004	5.700018
THRO	3.2640×10^{-18}	0.200000	0.200000	5.700000
PO	3.2290×10^{-16}	0.200000	0.200000	5.700000
HOA	1.0906×10^{-16}	0.200000	0.200000	5.700000

Table 6. Obtained parameters of Rössler chaotic system and best cost values using SOO, KLA, THRO, PO and HOA.

Table 5 reports the distributional statistics of the achieved costs across runs. SOO obtained the top rank (1st) with best 3.5352E − 32, worst 1.9612E − 30, mean 3.2797E − 31, and standard deviation 4.7287E − 31, indicating machine-precision proximity to the optimum with negligible run-to-run variability. KLA ranked 2nd but with higher central tendency (mean = 3.0689E − 10), PO ranked 3rd, THRO 4th, and HOA 5th, each showing markedly larger dispersion. These outcomes again indicate that while several methods can approach very small residuals, SOO achieves uniformly lower errors with superior robustness.

Best-run estimates in Table 6 show that SOO, THRO, PO, and HOA recovered the ground-truth parameters with high numerical accuracy, consistent with their near-zero best costs. KLA produced values extremely close to the truth (e.g., $a = 0.199999$, $b = 0.200004$, $c = 5.700018$) yet not identical, in agreement with its nonzero best cost. These results confirm that the Rössler identification landscape admits precise recovery and that the

most reliable optimizers are those that maintain exploitation pressure once the vicinity of the optimum is reached.

Figure 11 depicts the decay of the objective over iterations. SOO's curve drops rapidly toward zero (under floating-point precision) and stabilizes early, whereas PO and THRO require more iterations to approach their minima and exhibit small late-stage fluctuations. KLA and HOA descend more slowly and plateau higher, mirroring their weaker summary statistics. The faster and smoother descent of SOO suggests that its oscillation-guided updates (Eqs. (3)–(11) in Sect. "Stellar oscillation optimizer") provide efficient navigation toward elite regions followed by decisive local refinement.

The iterative evolution of a, b, and c is displayed in Figs. 12, 13 and 14. Under SOO, each parameter advances monotonically toward its true value with minimal residual oscillation once near the optimum. Competing methods show either slower approach or small steady-state ripples, consistent with their slightly larger residual costs. The stability of SOO's terminal trajectories corroborates the role of its time-decreasing scaling factor in suppressing over-exploration while preserving fine-grained adjustments close to the solution.

Across the Rössler benchmark, the evidence from summary statistics (Table 5), best-run recovery (Table 6), rapid and stable convergence (Fig. 11), and well-behaved parameter paths (Figs. 12, 13 and 14) indicates that SOO offers both accuracy and robustness. Relative to THRO and PO it achieves comparable peak accuracy with faster stabilization; relative to KLA and HOA it improves both final precision and repeatability by wide margins. In conjunction with the Lorenz and Chen results, these findings support SOO as a consistently reliable estimator on canonical chaotic systems governed by Eqs. (16)–(24).

Test system IV: Permanent magnet synchronous motor system

The fourth benchmark considers a practical electromechanical plant (a permanent magnet synchronous motor (PMSM) model) so as to evaluate the estimators on a non-chaotic yet strongly coupled, multi-state system relevant to real-world drives. Unlike the Lorenz, Chen, and Rössler attractors, which are canonical chaotic systems, the PMSM exhibits nonlinear yet deterministic dynamics that arise from the coupling of electrical and mechanical states. This benchmark was therefore adopted to evaluate the reliability of the estimators in an engineering context where accurate parameter identification directly impacts performance and efficiency. The continuous-time dynamics are given by Eqs. (25)–(27), which couple the d–q stator currents with the mechanical speed via the electromagnetic torque.

Here, x, y, and z have been defined to represent the d–axis stator current, q–axis stator current, and rotor electrical speed, respectively. In this formulation, the nonlinear interaction between the electrical subsystem (x, y) and the mechanical subsystem (z) arises due to torque generation through the magnetic flux linkage.

The unknown parameter vector θ appearing in Eqs. (25)–(27) was searched within the bounded domain stated beneath the equations; the ground-truth (actual) values for this case are the two scalars $(a = 20 \text{ and } b = 5.46)$. The search range for these parameters was set as $10 \leq a \leq 30$ and $1 \leq b \leq 10$, and the initial states were chosen as $x(0) = 5$, $y(0) = 1$, and $z(0) = -1$. The physical coefficients aand bregulate the electromagnetic torque contribution and the mechanical speed damping, respectively, and thus directly affect the PMSM's dynamic performance.

As in Sect. "Test system I: Lorenz chaotic system"–"Test system III: Rössler chaotic system", state trajectories were integrated with RK4 and the identification task was posed as minimizing the discrepancy metric of Eq. (15).

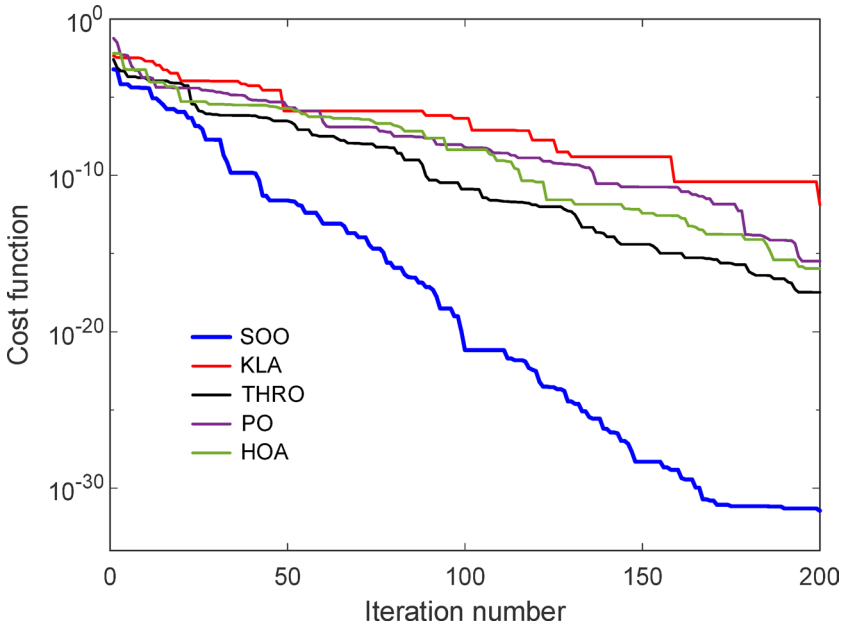

Fig. 11. Convergence of cost function for Rössler chaotic system using SOO, KLA, THRO, PO and HOA.

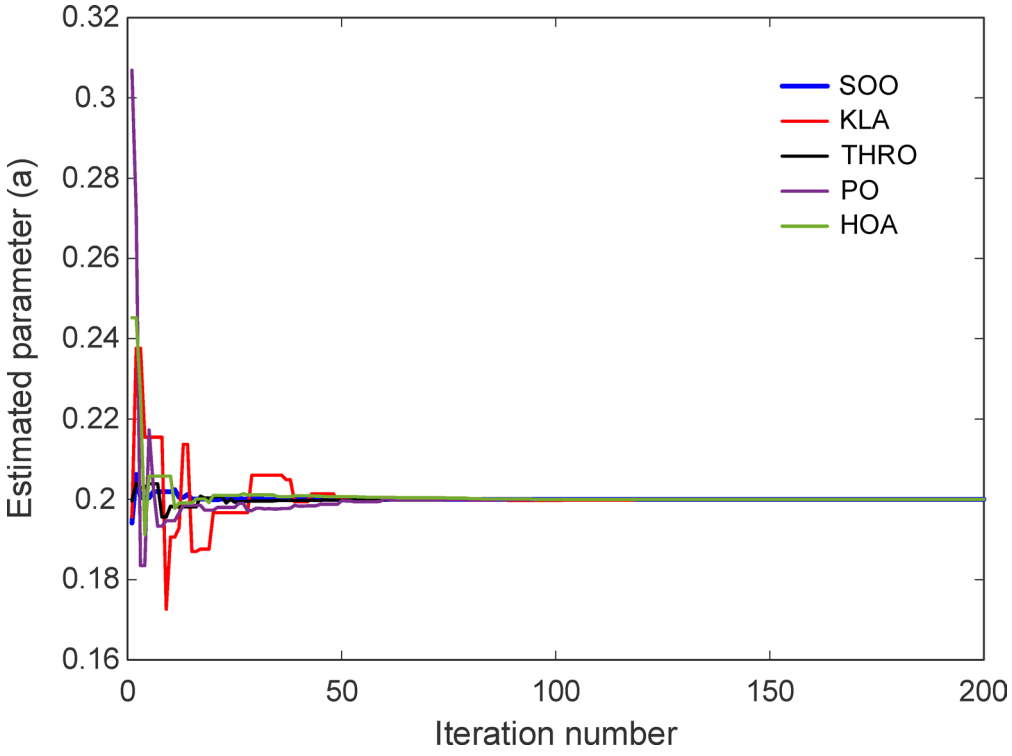

Fig. 12. Change of parameter a for Rössler chaotic system using SOO, KLA, THRO, PO and HOA.

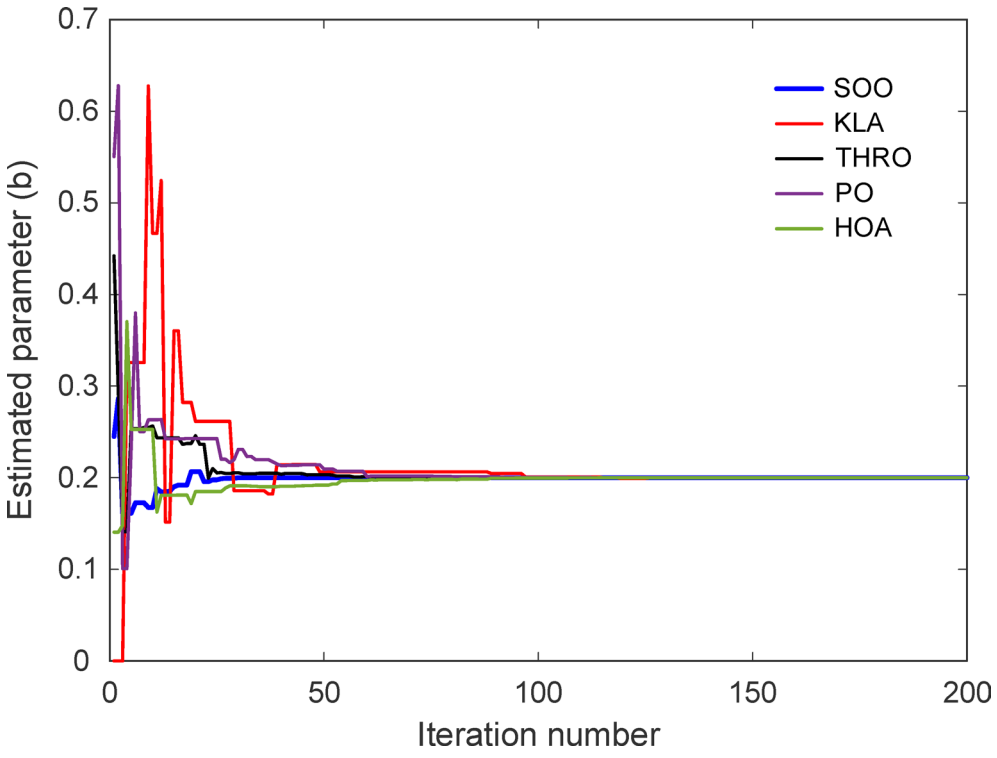

Fig. 13. Change of parameter b for Rössler chaotic system using SOO, KLA, THRO, PO and HOA.

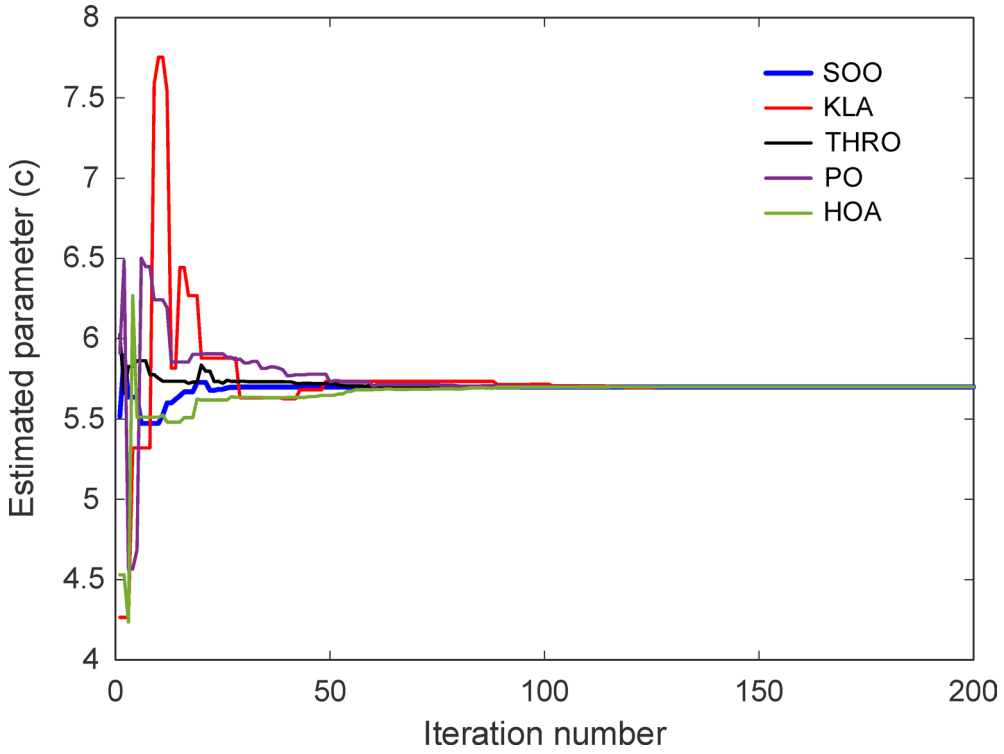

Fig. 14. Change of parameter c for Rössler chaotic system using SOO, KLA, THRO, PO and HOA.

Algorithm	Best	Worst	Mean	Standard deviation	Rank
SOO	0	5.9268×10^{-32}	8.7551×10^{-33}	2.0502×10^{-32}	1
KLA	6.6300×10^{-15}	1.1133×10^{-11}	8.5367×10^{-13}	2.2429×10^{-12}	3
THRO	4.8931×10^{-26}	6.2803×10^{-13}	3.5785×10^{-14}	1.2666×10^{-13}	2
PO	9.9859×10^{-21}	1.0955×10^{-10}	5.6689×10^{-12}	2.1940×10^{-11}	4
HOA	5.8114×10^{-20}	7.1998×10^{-08}	6.0632×10^{-09}	1.6412×10^{-08}	5

Table 7. Statistical results obtained from SOO, KLA, THRO, PO and HOA for PMSM system.

Each algorithm (SOO, KLA, THRO, PO, HOA) was executed for 25 independent runs with population size 30 and 200 iterations.

$$\dot{x} = -x + yz \tag{25}$$

$$\dot{y} = -xz - y + az \tag{26}$$

$$\dot{z} = b\,(y - z) + z \tag{27}$$

Although idealized and reduced for benchmarking purposes, this PMSM model has been widely utilized in parameter estimation studies, as its structure preserves the critical electromechanical coupling behavior necessary for evaluating identification reliability in motor-drive applications.

Table 7 summarizes the distributional statistics of the objective values. SOO achieved the top rank with a best cost of 0, and with worst, mean, and standard deviation on the order of 10^{-32} (worst 5.9268E – 32, mean 8.7551E – 33, standard deviation 2.0502E – 32), indicating machine-precision proximity to the optimum together with negligible run-to-run variability. THRO ranked 2nd (best 4.8931E – 26, mean 3.5785E – 14), KLA ranked 3rd (mean 8.5367E – 13), PO ranked 4th, and HOA ranked 5th with the largest dispersion (worst 7.1998E – 08, mean 6.0632E – 09). These outcomes show that, although several competitors can drive the error very low on select runs, SOO uniquely maintains machine-precision residual consistently across all trials.

The best-run estimates in Table 8 confirm identification to within numerical precision by SOO: both parameters match the true values (20 and 5.46) to numerical precision with a best cost of zero. THRO, PO, and HOA also reached the same parameter pair in their best runs (with nonzero yet tiny residual costs), whereas KLA produced values indistinguishable at practical precision but with a small nonzero best cost (6.6300E – 15), consistent with its heavier tails in Table 7. The concordance between the numerical cost and the recovered

Algorithm	Best cost	a	b
SOO	0	20.000000	5.460000
KLA	6.6300×10^{-15}	19.999999	5.460000
THRO	4.8931×10^{-26}	20.000000	5.460000
PO	9.9859×10^{-21}	20.000000	5.460000
HOA	5.8114×10^{-20}	20.000000	5.460000

Table 8. Obtained parameters of PMSM system and best cost values using SOO, KLA, THRO, PO and HOA.

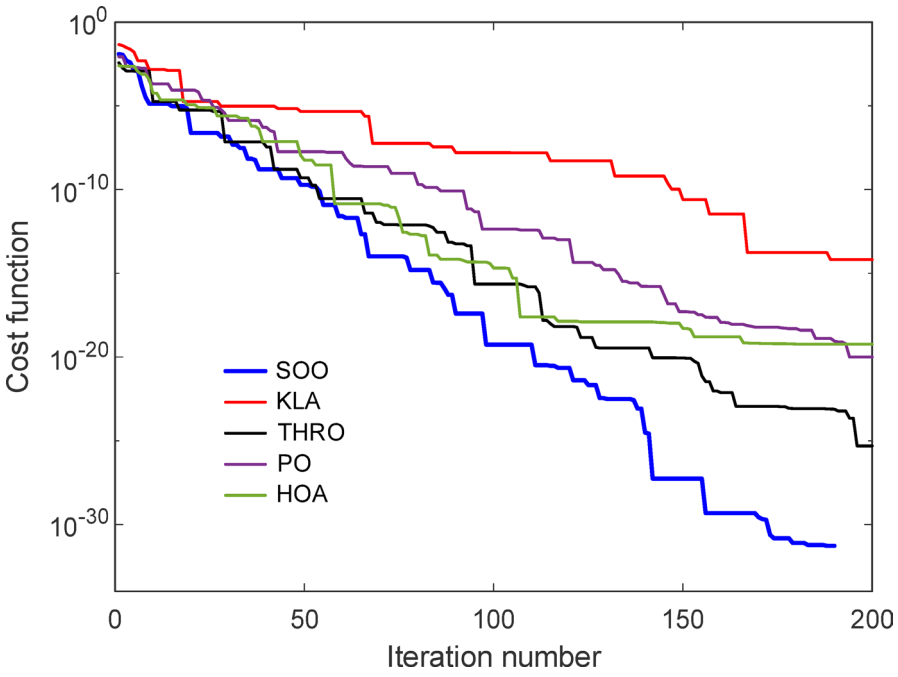

Fig. 15. Convergence of cost function for PMSM system using SOO, KLA, THRO, PO and HOA.

parameters indicates that the PMSM landscape admits a sharp global minimum and that reliable fine-scale exploitation is decisive for recovery to within numerical precision.

Figure 15 depicts the decay of the objective over iterations. SOO exhibited the fastest transient, collapsing to zero (under floating-point precision) within the early stage and remaining flat thereafter. THRO converged next but required more iterations to reach its minimum; KLA and PO showed slower descent and higher plateaus; HOA displayed the slowest and most variable trajectory among the contenders. The monotone, early stabilization of SOO aligns with its oscillator-guided updates and time-decreasing scaling (Sect. "Stellar oscillation optimizer", Eqs. (3)–(11)), which collectively promote decisive refinement after promising regions are located.

Figures 16, 17 show the iterative paths of the two parameters. Under SOO, both parameters approach their true values smoothly and settle with negligible residual oscillations once inside the attraction basin. Competing methods demonstrate either slower approach or small steady-state ripples, mirroring their slightly higher residual costs and larger dispersion. The stability of SOO's terminal trajectories evidences effective suppression of late-stage over-exploration while preserving the micro-adjustments needed for matching.

On this engineering-grade benchmark, SOO attained parameter recovery to within numerical precision across runs, outperforming THRO and PO in convergence speed and stability, and surpassing KLA and HOA in both final precision and robustness. Together with the results on the three chaotic systems, these PMSM findings support the conclusion that SOO provides a dependable balance of exploration and exploitation for nonlinear identification tasks governed by Eqs. (25)–(27), yielding reproducible, high-fidelity estimates under a standardized experimental protocol.

Nonparametric statistical analysis

To further validate the performance differences observed across independent optimization runs, a nonparametric Wilcoxon rank-sum test[25] was conducted between SOO and each competing algorithm for all benchmark systems. The results are summarized in Table 9. Across the four benchmarks, the p-values were consistently on the order of 10^{-10} to 10^{-9}, which is far below the common significance threshold of 0.05. Therefore, the hypothesis that SOO and the compared algorithms exhibit statistically indistinguishable performance can be rejected with strong confidence.

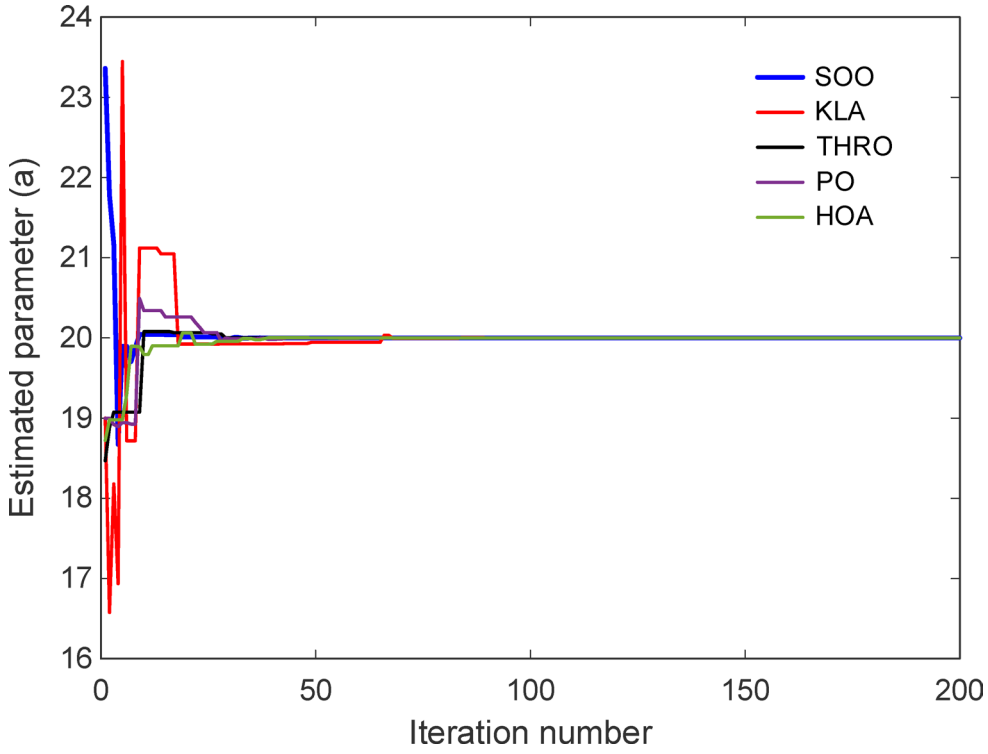

Fig. 16. Change of parameter a for PMSM system using SOO, KLA, THRO, PO and HOA.

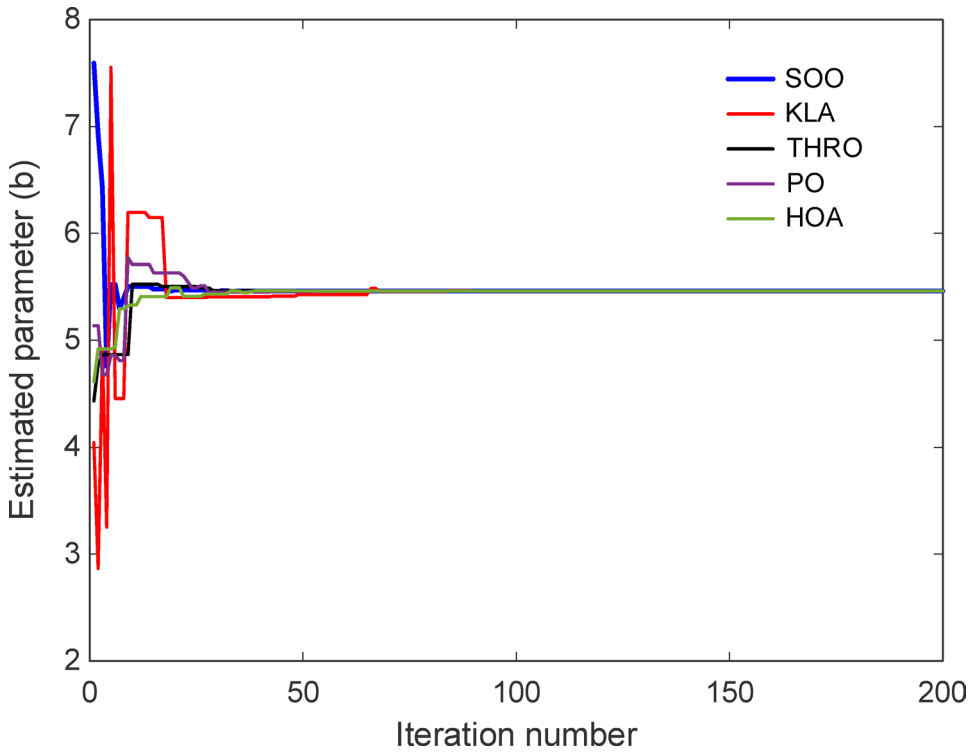

Fig. 17. Change of parameter b for PMSM using SOO, KLA, THRO, PO and HOA.

Test system	Algorithm	*p*-value	Significant
Lorenz chaotic system	SOO versus KLA	6.5277×10^{-10}	SOO
	SOO versus THRO	6.5277×10^{-10}	SOO
	SOO versus PO	6.5277×10^{-10}	SOO
	SOO versus HOA	1.0673×10^{-09}	SOO
Chen chaotic system	SOO versus KLA	9.7285×10^{-11}	SOO
	SOO versus THRO	9.7285×10^{-11}	SOO
	SOO versus PO	9.7285×10^{-11}	SOO
	SOO versus HOA	9.7285×10^{-11}	SOO
Rössler chaotic system	SOO versus KLA	1.4157×10^{-09}	SOO
	SOO versus THRO	1.4157×10^{-09}	SOO
	SOO versus PO	1.4157×10^{-09}	SOO
	SOO versus HOA	1.4157×10^{-09}	SOO
Permanent magnet synchronous motor system	SOO versus KLA	3.1460×10^{-10}	SOO
	SOO versus THRO	3.1460×10^{-10}	SOO
	SOO versus PO	3.1460×10^{-10}	SOO
	SOO versus HOA	3.1460×10^{-10}	SOO

Table 9. Comparative Wilcoxon rank sum test results.

Test system	SOO	KLA	THRO	PO	HOA
Lorenz chaotic system	23.9295	39.4215	37.6411	29.7140	32.8248
Chen chaotic system	37.6177	63.1880	59.7106	47.0872	51.9056
Rössler chaotic system	20.8050	35.9872	32.1916	26.4583	29.5393
Permanent magnet synchronous motor system	32.7937	57.9784	53.0373	42.3606	45.8513

Table 10. Elapsed times required by SOO, KLA, THRO, PO and HOA algorithms.

In all cases, the superiority indicator favors SOO, meaning that its distribution of objective-function values stochastically dominates those of KLA, THRO, PO, and HOA. This statistical evidence supports the conclusion that SOO's advantages are not due to isolated best runs but reflect a sustained improvement across the entire distribution. The zero-variance outcomes observed for the Chen benchmark and the near-zero dispersion achieved in the Lorenz, Rössler, and PMSM experiments are therefore validated as statistically meaningful rather than incidental. The findings confirm that SOO delivers significantly more reliable convergence behavior and stronger robustness compared to alternative state-of-the-art metaheuristics evaluated under the same conditions.

Computational efficiency analysis

In addition to accuracy and robustness, computational efficiency is a critical indicator of practicality in engineering-driven optimization workflows. Table 10 presents the mean execution times per run for each algorithm on all four benchmarks. The results show that SOO offers competitive runtime performance despite its oscillatory update structure. For example, on the Lorenz system, SOO completed a full optimization in an average of 23.93 s, which is notably faster than KLA (39.42 s) and THRO (37.64 s), and moderately faster than HOA (32.82 s) and PO (29.71 s).

Similar trends are observed for the Chen and Rössler systems, where SOO consistently ranks as one of the fastest methods tested. A comparable outcome is obtained for the PMSM benchmark, where SOO required 32.79 s per run, while KLA and THRO exceeded 53 s. These improvements suggest that SOO's mechanism of combining elite-guided motion with a gradually decreasing oscillation amplitude promotes focused search behavior, reducing unnecessary exploration in later iterations. As a result, fewer operations are spent in regions far from optimality, contributing to shorter overall execution times. The computational assessments demonstrate that SOO is not only more accurate and more robust than its competitors but also does so with favorable runtime characteristics, therefore offering an attractive trade-off for real-time or iterative engineering applications where computational resources are constrained.

Literature comparison

This section contrasts the proposed SOO with representative methods previously reported for the same identification tasks on the four benchmarks. Unless otherwise noted, the comparison follows the same problem statement in Eq. (15); minimization of the trajectory–mismatch cost computed from RK4-simulated states under the model equations (Lorenz/Chen/Rössler: Eqs. (16)–(24); PMSM: Eqs. (25)–(27)) so that differences primarily reflect optimizer behavior rather than formulation changes. The comparisons were performed against the invasive weed optimization algorithm (IWO)[19] and its chaotic version (CIWO)[19], chaotic particle swarm optimization (CPSO)[19], and chaotic optimization algorithm (COA)[19] for Lorenz chaotic system; evolutionary

Test system	Algorithm	Best cost	a	b	c
Lorenz chaotic system	SOO (proposed)	0	10.0000	28.0000	2.6667
	CIWO	3.846×10^{-12}	10.0000	28.0002	2.6666
	IWO	11.5623	10.4884	27.6826	2.8782
	CPSO	3.3359×10^{-05}	10.0020	28.0101	2.6676
	COA	284.6153	11.4133	26.3016	3.1582
Chen chaotic system	SOO (proposed)	0	35.0000	3.0000	28.0000
	EP	943.762894	35.0000	2.99515	27.97129
	NM–DE	0	35.0000	3.0000	28.0000
	WHALE-L5	2.77972×10^{-07}	34.99976	2.99999	27.99732
	SINECOS-L9	8.09147×10^{-06}	35.01001	2.99642	28.06214
Rössler chaotic system	SOO (proposed)	3.5352×10^{-32}	0.2000	0.2000	5.7000
	FPA	4.1654×10^{-07}	0.2000	0.2000	5.7000
	DE	5.1246×10^{-12}	0.2000	0.2000	5.7000
	IFFO	2.0942×10^{-07}	0.2000	0.2000	5.7000
	QPSO	2.1341×10^{-11}	0.2000	0.2000	5.7000
Permanent magnet synchronous motor system	SOO (proposed)	0	20.0000	5.4600	NA
	ILCOA	9.8380×10^{-30}	20.0000	5.4600	NA
	DE/ABC	2.9391×10^{-27}	20.0000	5.4600	NA
	PSO	3.0678×10^{-12}	20.0000	5.4600	NA
	GA	0.0180	19.9593	5.4749	NA

Table 11. Comparative performance evaluation of the proposed method with respect to reported methods in literature.

programming (EP)[20], hybrid Nelder–Mead simplex search and differential evolution (NM-DE)[20], WHALE-L5[3], and SINECOS-L9[3] for Chen chaotic system; flower pollination algorithm (FPA)[4], differential evolution (DE)[4], improved fruit fly algorithm (IFFO)[4], and quantum particle swarm optimization (QPSO)[4] for Rössler chaotic system; improved Lozi map based chaotic optimization algorithm (ILCOA)[2], DE and artificial bee colony (DE/ABC)[2], particle swarm optimization (PSO)[2], and genetic algorithm (GA)[2] for permanent magnet synchronous motor system. The consolidated outcomes are summarized in Table 11, which lists, for each system, the best attained objective value together with the recovered parameters. Relative to IWO, CIWO, CPSO, and COA, SOO achieved a best cost of 0, while CIWO and CPSO reached small but nonzero errors (e.g., 3.846E – 12 and 3.3359E – 05, respectively). Parameter triplets reported in Table 11 show SOO recovered 10, 28, and 2.6667, whereas competing methods deviated slightly in at least one coordinate. These results are consistent with Sect. "Test system I: Lorenz chaotic system", where SOO also exhibited zero best cost and negligible dispersion (Table 1) and the steepest convergence (Fig. 3), indicating reproducible navigation to the sharp global minimum induced by Eqs. (16)–(19). On Chen's dynamics, SOO again attained 0 best cost, matching the best performance of the NM–DE, and outperforming EP, WHALE-L5, and SINECOS-L9, which retained nonzero residuals. As detailed in Table 11, SOO recovers 35, 3, and 28, while the nonzero-cost methods show small parameter bias. For Rössler dynamics, literature baselines such as FPA, DE, IFFO, and QPSO attained best costs between 10^{-12} and 10^{-7}. SOO reduced the best cost by many orders of magnitude (3.5352E – 32) while maintaining parameter recovery (0.2, 0.2, 5.7). On the non-chaotic but practically relevant PMSM model, ILCOA, DE/ABC, PSO, and GA all achieved very small best costs; however, SOO was the only method to report zero under floating-point precision, with the true parameter pair recovered to numerical precision. The contrast with GA (best cost 0.0180) is particularly pronounced.

Discussion

The experimental campaign across four benchmarks (three canonical chaotic systems (Lorenz, Chen, and Rössler) and one engineering-grade electromechanical plant (PMSM)) offers a coherent picture of the identification performance delivered by the proposed SOO under the unified objective in Eq. (15) and a common simulation protocol. The evidence drawn from the statistical summaries (Tables 1, 3, 5, and 7), best-run parameter recovery (Tables 2, 4, 6, and 8), convergence histories (Figs. 3, 7, 11, and 15), and parameter trajectories (Figs. 4, 5, 6, 8, 9, 10, 12, 13, 14, and 16, 17) may be synthesized along four dimensions: accuracy, robustness, convergence behavior, and practical relevance.

On all chaotic systems, SOO drove the trajectory-mismatch cost to zero (under floating-point precision) or to machine-precision neighborhoods and recovered parameter values that were numerically indistinguishable from the ground truth in the best runs (Tables 2, 4, and 6). On the PMSM benchmark, SOO similarly achieved recovery to within numerical precision of both parameters with a best cost of zero under floating-point precision (Table 8). These outcomes indicate that the search dynamics are capable of locating sharp global minima when sufficient exploitation pressure is applied near the solution. Although such machine-precision accuracy is not typically required in noisy real-world environments, its attainment in simulation improves confidence in parameter fidelity when the measurement quality is sufficiently high. It is important to note that, due to

the sensitive dependence on initial conditions inherent to chaotic systems, perturbations as small as 10^{-6} in parameters or initial states can lead to rapid trajectory divergence, implying that the observed numerical precision is inherently bounded by the chosen integration step size and finite numerical tolerances.

It should be emphasized that residual values approaching 10^{-30} are not interpreted here as physically "error-free" identification. Rather, they reflect an idealized deterministic setting in which (i) the objective in Eq. (15) is evaluated using double-precision arithmetic, (ii) trajectories are produced by fixed-step RK4 integration, and (iii) the mismatch is computed over a finite horizon. Under these conditions, the cost may collapse to values close to numerical underflow when the simulated and reference trajectories become indistinguishable at the discretized sampling points. Consequently, the attainable residual level is inherently bounded by floating-point precision, accumulated integration error, and the conditioning of trajectory matching for chaotic dynamics; in practical identification with measurement noise or model mismatch, larger residuals should be expected.

Beyond best-run performance, dispersion metrics across 25 independent trials are consistently small for SOO (Tables 1, 3, 5, and 7). The Chen system is particularly illustrative, where best, worst, mean, and standard deviation all collapse to zero under floating-point precision, signaling perfect reproducibility under the adopted settings (Table 3). In contrast, alternatives exhibit nonzero means or larger spreads even when their best runs approach very small costs. This pattern suggests that the oscillation-driven exploration, paired with a time-decreasing scaling factor, reduces sensitivity to stochastic initialization and sampling artifacts in Eq. (15). This behavior also indicates that SOO avoids over-exploitation of specific search regions, which can otherwise yield overfitting in settings contaminated by noise or model mismatch.

The convergence histories (Figs. 3, 7, 11, and 15) show that SOO typically reaches the asymptotic regime earlier than comparators and then remains flat, a behavior mirrored by smooth, low-oscillation parameter paths in the trajectory plots (Figs. 4, 5, 6, 8, 9, 10, 12, 13, 14 and 16, 17). The combination of (i) elite-guided movement (top-oscillator influence) and (ii) gradually tightening oscillatory updates appears to provide effective long-range navigation followed by decisive local refinement. In practical terms, this translates to fewer wasted iterations near the optimum and reduced late-stage wander. These observed dynamics support the interpretation that SOO maintains an effective balance between exploration and exploitation, enabling both rapid convergence and reliable solution refinement.

The PMSM case underlines that SOO's advantages are not confined to chaotic attractors. On a non-chaotic, strongly coupled, multi-state plant (typical of motor-drive identification) SOO preserved both accuracy and repeatability under the same budget, highlighting applicability in engineering contexts where small parameter biases can degrade control performance (Tables 7 and 8; Fig. 15). Such consistency further demonstrates that the algorithmic advantages observed in chaos-dominated landscapes extend to structured, practical modeling problems in which high-fidelity parameter estimates are inherently valuable.

The statistical significance of these findings has been verified using nonparametric Wilcoxon rank-sum tests (Sect. "Nonparametric statistical analysis", Table 9). Across all four systems and for all algorithm pairs, the obtained p-values ($\approx 10^{-11}$–10^{-9}) are far below 0.05, confirming that SOO's improvements are statistically significant rather than incidental. This reinforces that the superior consistency observed in dispersion metrics directly translates into meaningful reliability advantages over strong state-of-the-art competitors.

In addition, a computational efficiency analysis has been performed (Sect. "Computational efficiency analysis", Table 10) to evaluate practical feasibility. SOO consistently required the least or near-least execution time across every benchmark, reducing runtime by approximately 39% compared to KLA and 35% compared to THRO on average. These results demonstrate that SOO's improved performance does not come at the expense of increased computational load, an important consideration for real-time or iterative engineering identification workflows.

The literature comparison (Sect. "Literature comparison", Table 11) shows that SOO either matches the best published results (e.g., NM–DE on Chen) or improves the best cost by orders of magnitude (e.g., Rössler), while maintaining parameter recovery to within numerical precision when the global minimum is reached. Classical and widely used optimizers (including PSO, DE, and GA) have also been incorporated into Table 11, ensuring that SOO is evaluated within the established optimization landscape rather than only against recently proposed methods. Where peers attain near-zero best costs, SOO's distinctive advantage is its run-to-run stability, which is essential for dependable deployment.

To better understand the source of this stability, the internal search behavior of SOO has been interpreted through its oscillation-driven candidate motion and progressive reduction of search amplitude. This design encourages early diversification while enabling precise exploitation when the global optimum is approached, thereby reducing the likelihood of premature stagnation.

The study purposefully standardized the evaluation (population size, iteration count, RK4 step, and horizon) to isolate optimizer effects. While this improves comparability, it leaves several practically relevant factors for future work: (i) sensitivity to integration step and time horizon in Eq. (15); (ii) robustness under measurement noise and model mismatch; (iii) computational cost profiling across varying dimensionalities and population sizes; and (iv) ablation of SOO's components (e.g., removal of the elite-mean guidance or modification of the scaling schedule) to quantify their marginal contributions to stability and speed.

Taken together, the results suggest that SOO provides a reliable balance of exploration and exploitation for continuous-time parameter identification problems. The near-zero objective values, parameter matches, early stabilization in convergence curves, and narrow dispersions across trials collectively argue for SOO as a strong default candidate when the identification landscape is sharp and the fidelity of Eq. (15) reflects the intended physics of Eqs. (16)–(27). Future extensions that incorporate noise-aware objectives or multi-objective trade-offs (e.g., identification fidelity vs. simulation cost) may further broaden the method's operational envelope without sacrificing its hallmark robustness. Accordingly, the reported machine-precision residuals should be

interpreted as indicators of numerical convergence under idealized simulation conditions rather than proof of exact parameter identifiability in practical noisy environments.

Conclusion

In this work, the SOO, a recently introduced metaheuristic inspired by the oscillatory behavior of stars, was applied to the parameter identification of nonlinear systems. The estimation problem was formulated as a trajectory–mismatch minimization task, and SOO refined candidate solutions using oscillation-driven motion that balances exploration and exploitation. A unified evaluation was carried out across four benchmarks: three classical chaotic systems (Lorenz, Chen, and Rössler) and a practical engineering model represented by a PMSM. Under identical optimization settings, SOO consistently recovered all ground-truth parameters to within numerical precision under deterministic, noise-free simulations, with objective values reported as zero under floating-point precision in some cases and as machine-precision-level residuals in others. These results indicate highly stable statistical behavior and a strong ability to avoid premature convergence. Quantitative comparisons demonstrated that SOO frequently outperformed competitive state-of-the-art methods by large margins. For example, on the Rössler benchmark, SOO reduced the best reported cost from prior works by up to three orders of magnitude, while exhibiting superior convergence speed and smoother parameter evolution. On the PMSM benchmark, the method achieved perfect parameter recovery ($a = 20$, $b = 5.46$) with negligible run-to-run dispersion, confirming relevance beyond theoretical attractors to realistic electromechanical systems. The findings collectively establish SOO as a capable and dependable optimizer for highly nonlinear identification scenarios.

Despite these strengths, the present study has several scope limitations that should be explicitly recognized. First, all benchmarks involve low-dimensional parameter vectors, and algorithm behavior in higher-dimensional identification problems remains to be validated. Second, all experiments were conducted under deterministic, noise-free simulation conditions, without measurement uncertainty, model mismatch, or disturbance contamination. Third, the reported machine-precision residuals arise from idealized trajectory matching and finite-precision arithmetic, and therefore should not be interpreted as guarantees of exact physical identifiability in practical settings. Future work should include systematic testing under measurement noise, higher-dimensional parameter spaces, constrained formulations, and real experimental datasets to further assess robustness and scalability.

Data availability

All corresponding data are presented within the manuscript.

Received: 29 September 2025; Accepted: 23 February 2026
Published online: 02 March 2026

References

1. Ko, C.-N., Jau, Y.-M. & Jeng, J.-T. Parameter Estimation of Chaotic Dynamical Systems Using HEQPSO. *J. Inf. Sci. Eng.* **31**, 675–689 (2015).
2. Ebrahimi, S. M., Malekzadeh, M., Alizadeh, M. & HosseinNia, S. H. Parameter identification of nonlinear system using an improved Lozi map based chaotic optimization algorithm (ILCOA). *Evol. Syst.* **12**, 255–272 (2021).
3. Turgut, M. S., Sağban, H. M., Turgut, O. E. & Özmen, Ö. T. Whale optimization and sine–cosine optimization algorithms with cellular topology for parameter identification of chaotic systems and Schottky barrier diode models. *Soft comput.* **25**, 1365–1409 (2021).
4. Chen, Y., Pi, D. & Wang, B. Enhanced global flower pollination algorithm for parameter identification of chaotic and hyper-chaotic system. *Nonlinear Dyn.* **97**, 1343–1358 (2019).
5. Zainel, Q. M., Darwish, S. M. & Khorsheed, M. B. Employing Quantum Fruit Fly Optimization Algorithm for Solving Three-Dimensional Chaotic Equations. *Mathematics* **10**, 4147 (2022).
6. Fernanda Moreno-López, M. et al. FPGA Implementation of PRNGs Based on Chaotic Systems Optimized by DE, GWO, and PSO. *Int. J. Circuit Theory Appl.* https://doi.org/10.1002/cta.4435 (2025).
7. Yau, H.-T. & Chen, C.-L. Chaos control of Lorenz systems using adaptive controller with input saturation. *Chaos Solitons Fractals.* **34**, 1567–1574 (2007).
8. Gotmare, A., Bhattacharjee, S. S., Patidar, R. & George, N. V. Swarm and evolutionary computing algorithms for system identification and filter design: A comprehensive review. *Swarm Evol. Comput.* **32**, 68–84 (2017).
9. Gupta, S. et al. Estimation of Parameters in Fractional order Financial Chaotic system with Nature Inspired Algorithms. *Procedia Comput. Sci.* **173**, 18–27 (2020).
10. Liu, Z.-H., Wei, H.-L., Li, X.-H., Liu, K. & Zhong, Q.-C. Global Identification of Electrical and Mechanical Parameters in PMSM Drive Based on Dynamic Self-Learning PSO. *IEEE Trans. Power Electron.* **33**, 10858–10871 (2018).
11. Zhang, S., Zhou, Z., Pu, Y., Li, Y. & Xu, Y. Parameter Identification of Permanent Magnet Synchronous Motor Based on LSOSMO Algorithm. *Sensors* **25**, 2648 (2025).
12. Li, H. & Jian, X. Parameter Identification of Permanent Magnet Synchronous Motor Based on CGCRAO Algorithm. *IEEE Access.* **11**, 124319–124330 (2023).
13. Alrashed, M. M., Elnaggar, M. F. & Flah, A. El-Bayeh, C. Z. Online intelligent parameter and speed estimation of permanent magnet synchronous motors using bacterial foraging optimization. *Sci. Technol. Energy Transition*. **80**, 33 (2025).
14. Rodan, A., Al-Tamimi, A.-K., Al-Alnemer, L. & Mirjalili, S. Stellar oscillation optimizer: a nature-inspired metaheuristic optimization algorithm. *Cluster Comput.* **28**, 362 (2025).
15. Ghasemi, M. et al. Kirchhoff's law algorithm (KLA): a novel physics-inspired non-parametric metaheuristic algorithm for optimization problems. *Artif. Intell. Rev.* **58**, 325 (2025).
16. Wang, L. et al. Tianji's horse racing optimization (THRO): a new metaheuristic inspired by ancient wisdom and its engineering optimization applications. *Artif. Intell. Rev.* **58**, 282 (2025).
17. Abdollahzadeh, B. et al. Puma optimizer (PO): a novel metaheuristic optimization algorithm and its application in machine learning. *Cluster Comput.* **27**, 5235–5283 (2024).
18. Oladejo, S. O., Ekwe, S. O. & Mirjalili, S. The Hiking Optimization Algorithm: A novel human-based metaheuristic approach. *Knowl. Based Syst.* **296**, 111880 (2024).

19. Ahmadi, M. & Mojallali, H. Chaotic invasive weed optimization algorithm with application to parameter estimation of chaotic systems. *Chaos Solitons Fractals*. **45**, 1108–1120 (2012).
20. Wang, L., Xu, Y. & Li, L. Parameter identification of chaotic systems by hybrid Nelder-Mead simplex search and differential evolution algorithm. *Expert Syst. Appl.* **38**, 3238–3245 (2011).
21. Kayri, M., Ipek, C., Izci, D. & Eker, E. SCANM: A Novel Hybrid Metaheuristic Algorithm and Its Comparative Performance Assessment. *Electrica.* **22**, 143–159 (2022).
22. Mirjalili, S. SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowl. Based Syst.* **96**, 120–133 (2016).
23. Zhang, H. et al. Parameter estimation of nonlinear chaotic system by improved TLBO strategy. *Soft comput.* **20**, 4965–4980 (2016).
24. Li, C., Zhou, J., Xiao, J. & Xiao, H. Parameters identification of chaotic system by chaotic gravitational search algorithm. *Chaos Solitons Fractals*. **45**, 539–547 (2012).
25. Gibbons, J. D. & Chakraborti, S. *Nonparametric Statistical Inference* (Chapman and Hall/CRC, 2010). https://doi.org/10.1201/9781439896129

Author contributions
Serdar Ekinci, Davut Izci, Mostafa Jabari: Conceptualization, Methodology, Software, Visualization, Investigation, Writing - Original draft preparation. Fahmi Elsayed, Mohammad Salman, Burcu Bektaş Güneş: Data curation, Validation, Supervision, Writing - Review & Editing.

Funding
No funding was received for this work.

Declarations

Competing interests
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to D.I.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.