IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

VOLUME : 11 ISSUE : 02 Print / Issue Publication Date: June 2026

ISSN : 2455-2143

Indexed In

Google SCHOLAR OPEN ACCESS doi Crossref

WWW.IJEAST.COM

editor@ijeast.com

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

METAHEURISTIC OPTIMIZATION FOR INITIAL CONDITION ESTIMATION IN CHAOTIC SYSTEMS: TOWARD ROBUST CHAOTIC COMMUNICATION NETWORKS

Amr Sayed Abdel Fattah Yousse
Higher Colleges of Technology
Alain, UAE

Abstract—**Chaotic signals offer strong potential for enhancing security, efficiency, and capacity in communication systems; however, their practical deployment is limited by the difficulty of accurately estimating initial conditions, which is essential for synchronization and system stability. While existing studies mainly focus on parameter optimization, systematic recovery and comparative evaluation of initial conditions using diverse metaheuristic algorithms remain insufficiently explored. This study addresses this gap by comparatively evaluating seven metaheuristic optimization algorithms—Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Differential Evolution (DE), Firefly Algorithm (FA), Simulated Annealing (SA), and Artificial Bee Colony (ABC)—for initial condition estimation in the Lorenz chaotic system. A unified evaluation framework is introduced, considering estimation accuracy, convergence speed, and computational efficiency. The results demonstrate that PSO achieves superior accuracy, establishing a new benchmark for precise initial condition recovery. Meanwhile, the ABC algorithm exhibits a favorable trade-off between accuracy and computational cost, making it suitable for resource-constrained applications. In contrast, ACO and FA show comparatively weaker performance.**

Keywords—**Chaotic system; Initial Condition Estimation, Optimization algorithms, Secure communication**

I. INTRODUCTION

Chaotic signals offer transformative potential in communication systems by significantly enhancing key aspects such as security, efficiency, and capacity. Their ability to improve security through advanced encryption techniques leverages the inherent unpredictable nature of chaos to create highly secure communication channels [1-3], [32], [33]. Chaotic signals also facilitate spread spectrum techniques, spreading the signal across a broad frequency band to enhance resistance to interference and jamming [4-7]. This characteristic enables high-speed data transmission by allowing more efficient bandwidth usage, thus meeting the growing demand for faster data rates [8–9], [34], [35]. The integration of multi-task learning and digital twin technologies can further optimize resource management and network performance in 6G edge-cloud-end integrated networks [32], [35]. Additionally, chaotic signals play a vital role in trustworthy federated reinforcement learning for resource allocation in IoMT [34], and they can also be employed in intelligent diagnostic systems for roller bearings using advanced neural networks [37]. Recent studies also explore the synchronization of complex dynamical networks with time-varying delays using event-triggered control mechanisms [36].

In addition, chaotic signals enhance robust signal recovery in noisy environments by improving error correction through their unique signal properties [10–11]. They help increase channel capacity by allowing multiple signals to coexist with minimal interference, which is crucial for dense communication networks [12], [39], [41]. Chaotic systems are also employed for secure key distribution by generating cryptographic keys with high randomness and unpredictability [13–15], [44].

Chaotic signals offer significant and unique advantages in enhancing secure IoT communications [44] and enabling advanced image compression techniques [45]. Their inherent unpredictability and sensitivity to initial conditions make them particularly well-suited for applications requiring robust security and efficient data representation. While machine learning contributes substantially to IoT network security [39], scalable stochastic approximation algorithms [40], and improved channel estimation in IRS-assisted wireless systems [43], it is the distinctive properties of chaotic systems that are being increasingly leveraged in these critical domains. Moreover, chaotic dynamics are also being explored in the joint optimization of sensing and communication, offering promising pathways for improving the performance of integrated systems [42].

Additionally, Chaotic signals play a pivotal role in advancing nonlinear signal processing, offering powerful tools for

managing complex communication environments [16], [46], [47]. Their dynamic and sensitive nature enables adaptive modulation, where signal parameters are intelligently adjusted in response to fluctuating channel conditions to maintain optimal performance [17], [49]. In networked systems, chaotic signals facilitate precise synchronization across multiple devices, significantly improving coordination and operational efficiency [10, 11, 18], [48], [50]. These benefits are further complemented by emerging technologies such as physics-informed neural networks for accurate path loss estimation [50] and edge computing for anomaly detection in university network data [51]. Ultimately, chaotic signals continue to push the boundaries of wireless communication and radar technologies, driving innovation in next-generation systems [8, 19], [52].

Critical Challenge: Despite these advantages, the practical implementation of chaotic communication systems faces a major obstacle: the accurate estimation of initial conditions. Initial conditions are essential for synchronizing and stabilizing chaotic systems, as even minor deviations can lead to significantly divergent outcomes [20–21]. This sensitivity for ICs poses a major hurdle in real-world applications. For example, slight deviations in initial conditions may arise unintentionally due to hardware-generated signals being affected by physical parameters such as resistance changes caused by temperature fluctuations [24]. This inherent unpredictability necessitates careful management and calibration in applications such as secure communications and chaotic signal processing, where precise initial conditions are critical to maintaining system stability and performance.

Research Gap: Despite prior efforts to address challenges in estimating chaotic system parameters [27–28], a critical and persistent gap remains: the absence of an effective systematic method for recovering initial conditions. This oversight is particularly significant, as initial conditions fundamentally shape the evolution of chaotic dynamics and are essential for synchronizing chaotic communication systems. Most existing studies concentrate on estimating system parameters—such as the Lorenz system constants (c_1, c_2, c_3)—while largely neglecting the pivotal role of initial states. For instance, [27] introduced a deep neural network-based approach for parameter estimation without accounting for the sensitivity of chaotic systems to their initial conditions. Similarly, [28] examined parameter estimation under varying initializations but did not propose a structured method for recovering the initial states themselves.

Recent developments in advanced modeling, such as Neural ODEs [53] and related data-driven approaches [54], underscore the importance of initial condition estimation, particularly when observations are noisy or incomplete. This demand is further reinforced by application domains such as weather prediction and data assimilation, where the reliability of forecasts critically depends on accurate initialization [55]. Nevertheless, significant shortcomings remain in the current state of research. For instance, although [58] introduces a novel 4D chaotic system with enhanced dynamical complexity and demonstrates secure communication under impulsive control, it does not address the problem of recovering unknown initial conditions or systematically comparing alternative estimation methods. Similarly, while the AI-Lorenz framework [56] provides a promising hybrid strategy for chaotic system identification, it neither evaluates synchronization performance in communication contexts nor benchmarks optimization-based techniques for initial condition recovery. Likewise, the ISINDy approach [57] improves robustness against noise, yet it lacks comparative algorithmic evaluation and overlooks the real-time performance requirements essential for chaotic communication networks. Collectively, these limitations highlight a critical research need: the absence of a comprehensive framework for evaluating initial condition estimation algorithms across multiple dimensions, including accuracy, convergence speed, and computational efficiency. The present study directly addresses this gap by proposing a robust evaluation methodology and systematically benchmarking seven state-of-the-art optimization algorithms for initial condition recovery in chaotic communication systems.

Novelty of This Work: This study presents the first systematic comparative analysis addressing the gap in ICs estimation. Our core contributions are as follows:

1. This study proposes a novel evaluation framework that assesses seven optimization algorithms—GA, PSO, ACO, DE, FA, SA, and ABC—specifically for initial condition recovery in the Lorenz system.

2. **Rigorous multi-metric benchmarking** across the following metrics:
   - Accuracy (Mean Squared Error)
   - Convergence speed
   - Computational efficiency

3. **Findings**
   - Particle Swarm Optimization (PSO) achieves remarkable precision, reaching a mean squared error (MSE) as low as $\approx 10^{-18}$, thereby setting a new benchmark in initial condition recovery performance.
   - Artificial Bee Colony (ABC) demonstrates a unique balance between accuracy (MSE < 0.0009) and computational efficiency, offering a compelling alternative for resource-constrained applications.
   - ACO and FA exhibit significant limitations (MSE > 1.28).

To summarize, this study advances the field by:

- Identifying and addressing the long-overlooked challenge of initial condition estimation of chaotic systems;

- Introducing a robust comparative framework for evaluating optimization algorithms used in initial condition estimation of chaotic systems, with a focus on accuracy, convergence speed, and computational efficiency;
- Providing actionable insights that can guide the development of next-generation chaotic communication systems.

This paper is organized as follows:
Introduction: This section provides an overview of the transformative potential of chaotic signals in communication systems, highlighting their benefits in enhancing security, efficiency, and capacity. It sets the stage for the research by outlining the critical challenge of estimating initial conditions in chaotic systems and the importance of this task for synchronizing and optimizing chaotic communication networks.
Literature Review: This section examines existing research on chaotic signals and their applications in communication systems. It discusses prior studies on the estimation of chaotic signal parameters and highlights the gap in the literature concerning the recovery of initial conditions. Additionally, it defines the basic principles of the Lorenz chaotic system and explains why it was chosen as the model for this research.
Problem Analysis and Simulation: This section details the problem of estimating initial conditions in chaotic systems and describes the simulation setup used to address this challenge. It introduces the Lorenz system as the chosen chaotic model and explains the parameters and initial conditions used in the simulations. The section also outlines the optimization algorithms evaluated in the study.

Simulation Results and Analysis: This section presents the results of the simulations and discusses the performance of the different optimization algorithms. It is divided into three subsections:

- Algorithms' Accuracy: This subsection evaluates each algorithm's accuracy in estimating the Lorenz system's initial conditions, comparing their performance based on the Mean Squared Error (MSE) and other relevant metrics.
- Convergence Speed and Computational Time: This subsection analyzes the algorithms' convergence characteristics and computational efficiency, discussing their speed and robustness in finding accurate solutions.
- Success rate and Standard Deviation: This section analyzes two key metrics for assessing an algorithm's performance. The success rate reflects the proportion of successful outcomes relative to total attempts, serving as an indicator of overall effectiveness. A high success rate suggests the algorithm consistently achieves its intended goal. In contrast, standard deviation measures the variability or spread of the results. A low standard deviation indicates consistent performance, while a high standard deviation suggests significant variation in outcomes. Together, these metrics provide a comprehensive understanding of the algorithm's reliability and stability.
- Roubstness index: This is a composite metric designed to evaluate the performance of optimization algorithms in estimating the initial conditions of chaotic systems.

Conclusion: This section summarizes the study's key findings, highlighting the most effective algorithms for estimating initial conditions in chaotic systems. It discusses the implications of these findings for the performance and reliability of chaotic communication systems.

Future Extensions and Applications: The findings of this study have significant implications for both future research and practical applications. This section highlights the opportunities for leveraging these insights in upcoming studies and real-world implementations.

II. LITERATURE REVIEW

To establish a robust and well-characterized chaotic system, the Lorenz system was chosen as the ideal model for this study. Defined by three ordinary differential equations, it exhibits various chaotic behaviors influenced by both parameter values and initial conditions, making it a suitable platform for our research [23]. The appeal of the Lorenz system lies in its simplicity and rich chaotic dynamics. Despite its simplicity, the Lorenz system exhibits hallmark features of chaos, including extreme sensitivity to initial conditions and long-term unpredictability [28]. Moreover, the Lorenz system is extensively studied and well-documented, providing a solid foundation for both theoretical exploration and practical applications. Its straightforward nature also allows for seamless integration into both hardware and software environments
The Lorenz system is governed by the following set of differential equations [28]:

$$\frac{dx}{dt} = K(c_1 y - c_1 x) \quad (1)$$
$$\frac{dy}{dt} = K(c_2 x - y - xz) \quad (2)$$
$$\frac{dz}{dt} = K(xy - c_3 z) \quad (3)$$

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

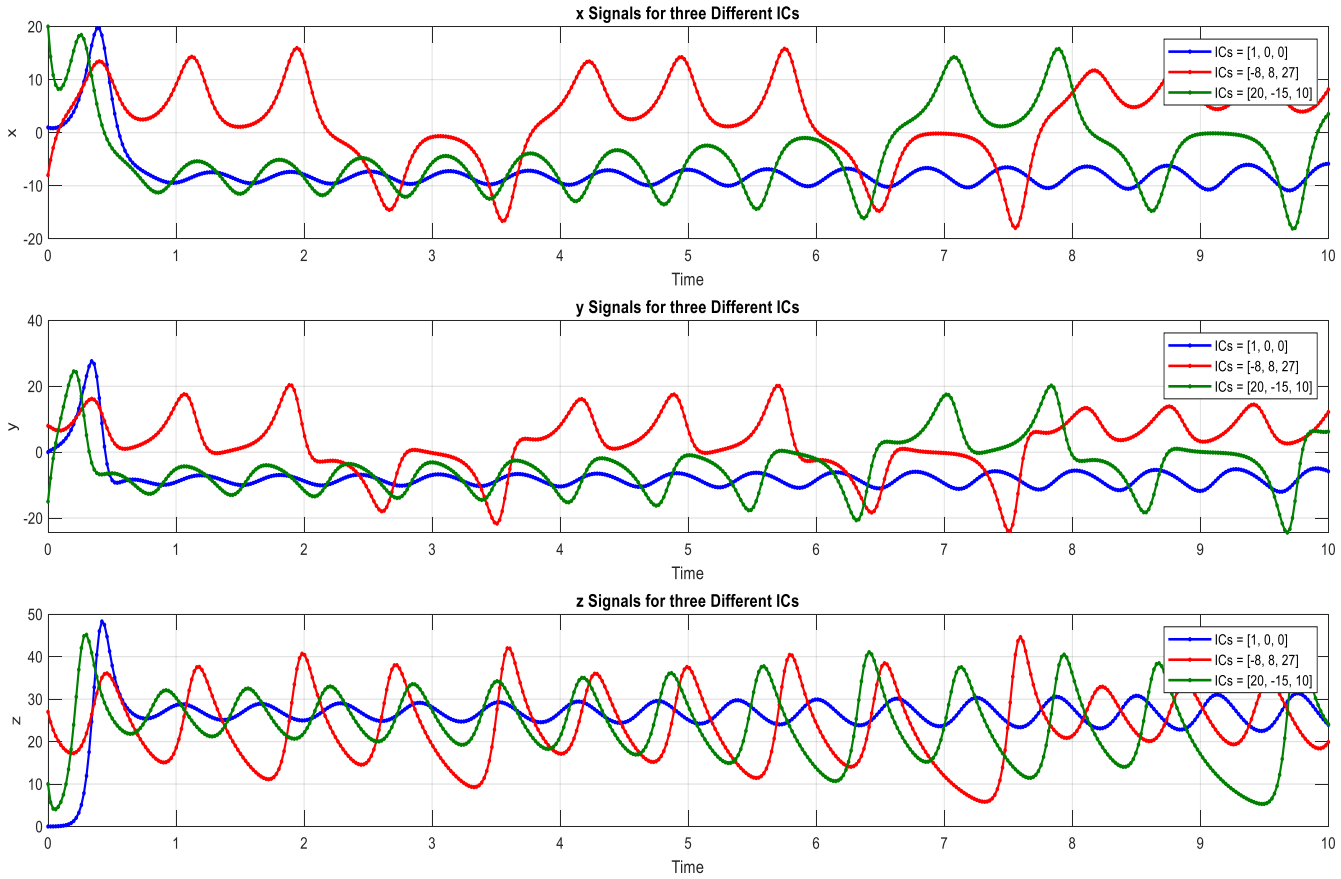


Fig. 1. Time series of x, y, and z variables for three distinct initial conditions in a Lorenz system.

As given in equations (1)–(3), the Lorenz system is defined by a set of three coupled ordinary differential equations (ODEs). c_1, c_2, and c_3 are arbitrary constants with the values 10, 28, and 8/3, respectively. K denotes the time scaling factor of the Lorenz system, and for simplicity, it is set to 1 in our study.

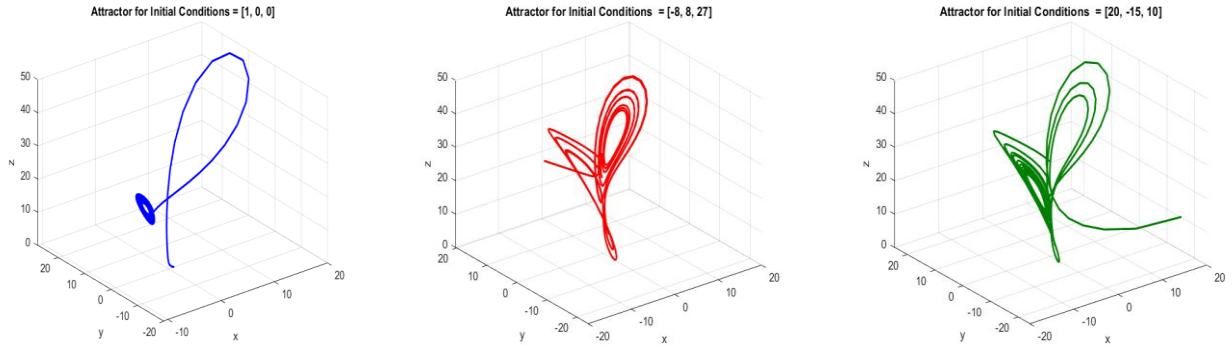
The system requires initial conditions to be specified at time t = 0, which act as the boundary conditions for the simulation. In this study, 30 distinct sets of initial conditions (x_o, y_o, z_o), listed in Table 1, are used to assess the performance of various optimization algorithms. Each simulation runs over a finite period of $t \in [0,25]$, with no external forcing or additional boundary constraints beyond the initial state.

To demonstrate the impact of initial conditions on system behavior, the Lorenz system is numerically solved for three representative cases: [1, 0, 0], [−8, 8, 27], and [20, −15, 10]. The resulting time-domain trajectories for x, y, and z are shown in Figure 1, while the corresponding 3D attractors are visualized in Figure 2. To further assess the similarity between the signals, a cross-correlation analysis was conducted, with the results shown in Figure 3.

The cross-correlation plots for the x- or y-signals reveal a consistently low degree of similarity, indicating that these signals do not align closely or share similar patterns over time. Despite potential variations within each set, there is minimal overlap or resemblance among the x- or y-signals.

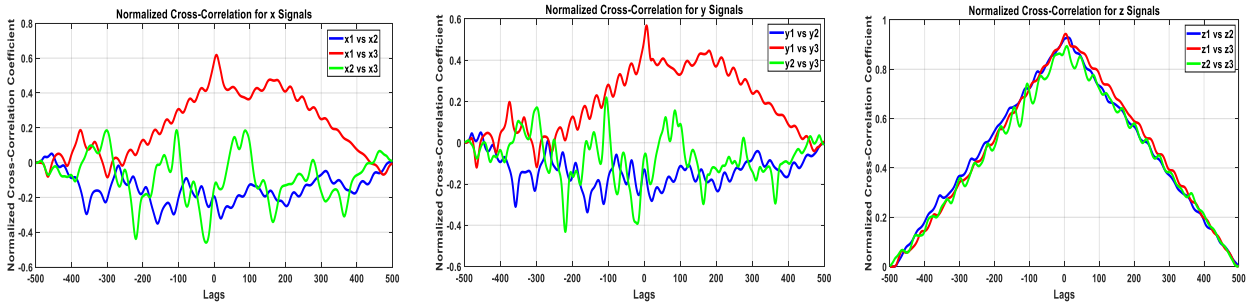
Regarding z signals, the peak near 1 at zero lag signifies a high degree of similarity or alignment between two z signals when perfectly synchronized in time. This indicates that the signals are highly similar when no time shift exists. The triangular shape of the cross-correlation function reveals that this similarity decreases linearly as the lag moves away from zero. This pattern implies that while the signals are closely matched at zero lag, their similarity diminishes as time shifts are introduced [28]. This cross-correlation characteristic can be observed in Figure. 3(c), where the z signals exhibit a distinct pattern or shape mirrored in the resulting cross-correlation.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)



(a) (b) (c)

Fig. 2. Lorenz attractor for three different sets of initial conditions [x_o,y_o,z_o]. Subplot (a) corresponds to ICs [1,0,0], Subplot (b) to ICs [-8,8,27], and Subplot (c) to ICs [20,-15,10].



(a) (b) (c)

Fig. 3. Normalized cross-correlation analysis of the x (subplot a), y (subplot b), and z (subplot c) variables in a Lorenz system for three different initial conditions.

Figure 4 illustrates the time evolution of the state variables x(t), y(t), and z(t) for the nominal initial condition [1, 0, 0], assessing the robustness and repeatability of the Lorenz system's output under uncertain initial conditions. A Monte Carlo simulation was performed with 30 runs, each incorporating small Gaussian noise (standard deviation of 10^{-4}) into the initial state to simulate realistic estimation uncertainty. The resulting trajectories exhibit narrow standard deviation bands during the early phase, followed by increasing divergence due to the system's chaotic nature. These results confirm the Lorenz system's sensitivity to initial conditions and demonstrate that early-time dynamics remain reproducible within a predictable margin, supporting the reliability of the estimation framework. This highlights the critical need for a robust method or algorithm to accurately determine these initial conditions, a tool that would be highly valuable for both the synchronization and exploitation of chaotic communication systems. Despite the recognized importance of this issue, it has not been thoroughly investigated in previous studies. This research aims to address this gap by using various optimization algorithms to estimate the initial conditions of Lorenz's chaotic signals.

III. PROBLEM ANALYSIS AND SIMULATION

This section details the methodology for initial condition estimation in chaotic systems, broken down into four key components. First, it establishes the Lorenz model implementation and parameter configuration. Next, it introduces and justifies the selection of seven optimization algorithms: Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Differential Evolution (DE), Firefly Algorithm (FA), Simulated Annealing (SA), and Artificial Bee Colony (ABC). The text then explains the parameter tuning process for each algorithm. Finally, it presents the simulation workflow through a comprehensive flowchart and algorithmic pseudocode.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

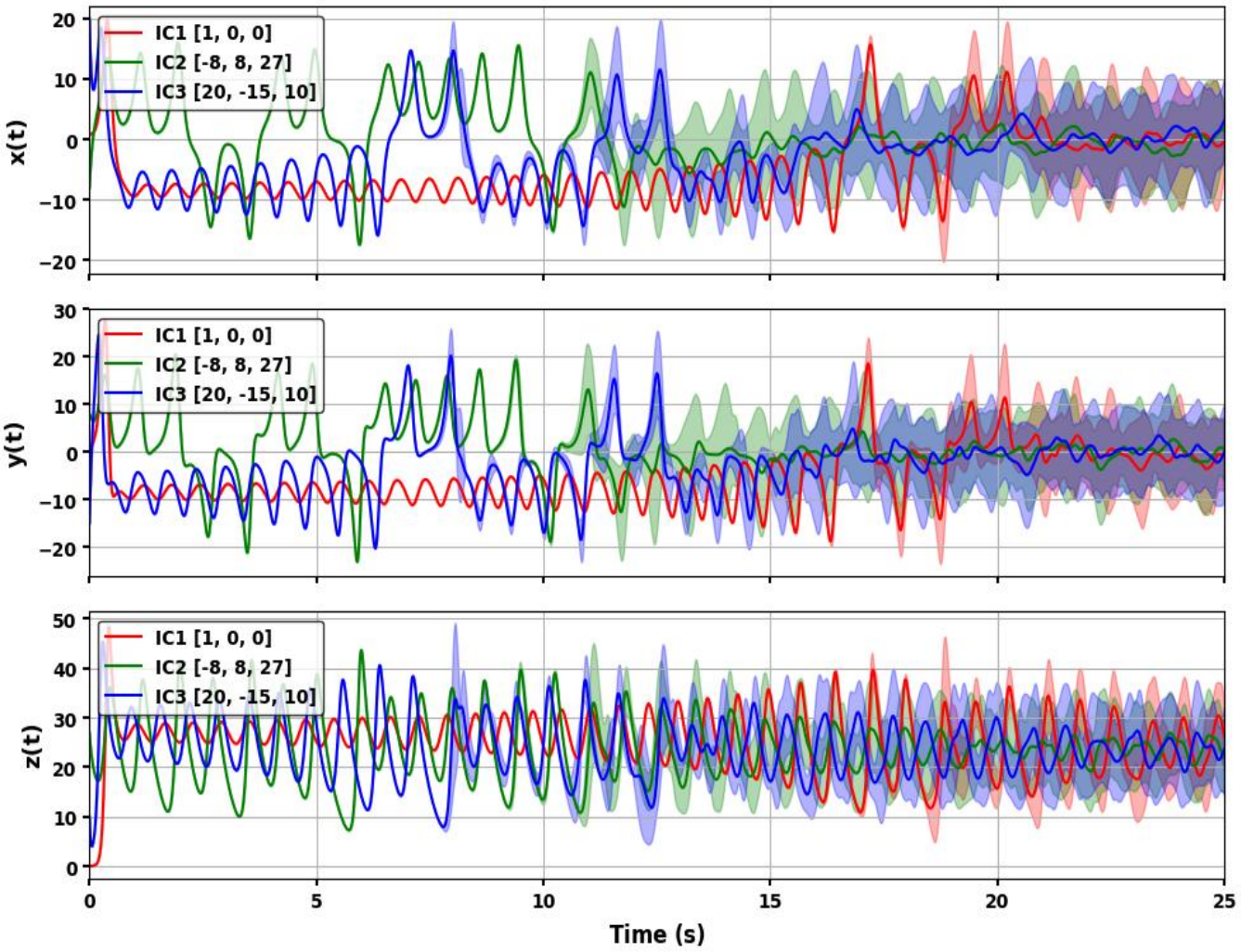


Fig. 4. Time series of x(t), y(t), and z(t) for the Lorenz system under initial condition [1, 0, 0], showing mean trajectories and shaded error bands (±1 SD) over 30 noise-perturbed runs.

To investigate the sensitivity of chaotic systems to initial conditions, a MATLAB script was developed to simulate the Lorenz system, estimate its initial conditions using various optimization algorithms, and visualize the results. The Lorenz equations were defined using the classical chaotic parameters: $c_1 = 10$, $c_2 = 28$, and $c_3 = 8/3$, as established in prior studies [23, 28].

A total of 30 distinct initial condition sets (x_o, y_o, z_o), listed in Table 1, were selected to ensure a diverse and representative sampling of the system's state space. Each simulation was run over a time interval of $t \in [0 , 100]$, which was chosen to allow sufficient evolution of the chaotic dynamics while maintaining computational efficiency.

To estimate the initial conditions from observed trajectories, the study employed seven nature-inspired optimization algorithms:

- Genetic Algorithm (GA)
- Particle Swarm Optimization (PSO)
- Ant Colony Optimization (ACO)
- Differential Evolution (DE)
- Firefly Algorithm (FA)
- Simulated Annealing (SA)
- Artificial Bee Colony (ABC)

Each algorithm was tasked with minimizing the Mean Squared Error (MSE) between the simulated and actual trajectories, providing a quantitative measure of estimation accuracy. The parameters and configurations for each algorithm are detailed in Table 2, offering insight into their operational differences and tuning strategies.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

The script initializes the Lorenz system, configures each optimization algorithm, and iteratively runs them across all 30 initial condition sets. For each run, it captures:

- Estimated initial conditions
- Convergence behavior
- Computation time
- Final MSE values

Table 1. Predefined ICs of Lorenz system

set	x_0	y_0	z_0	*set*	x_0	y_0	z_0	*set*	x_0	y_0	z_0
1	-5	0	-4	11	-4	3	-5	21	-4	5	1
2	-2	4	-4	12	1	4	-5	22	3	4	5
3	-1	3	1	13	-3	2	-4	23	3	4	2
4	-5	5	-1	14	1	3	1	24	-4	-3	-5
5	4	4	1	15	-1	1	2	25	0	-2	3
6	3	-2	4	16	-5	-3	0	26	3	3	-3
7	0	-3	0	17	-3	0	0	27	4	3	-2
8	-5	4	3	18	4	1	-1	28	1	-2	-2
9	4	-5	3	19	4	-4	-1	29	-3	-5	1
10	-5	-3	3	20	1	-3	-4	30	-4	4	1

Supporting functions were implemented to compute the Lorenz system derivatives and define the objective function for optimization. The execution time of each algorithm was also recorded to assess computational efficiency.

The algorithms were implemented and tested on a system equipped with an 11th Gen Intel Core i7-1165G7 processor running at 2.80 GHz, along with 8.00 GB of RAM and a 256 GB SSD. These hardware specifications were chosen to ensure that the results are representative of real-world computational environments.

A. Optimization Algorithms

Coherence, relevance to chaotic systems, and practical trade-offs. The following criteria guided our choices:

- Diversity of Optimization Paradigms

To ensure a comprehensive comparison, we included algorithms representing distinct optimization strategies: Evolutionary Algorithms (GA, DE), which use population-based stochastic search through mutation, crossover, and selection; Swarm Intelligence algorithms (PSO, ABC, FA), which mimic collective behaviors such as bird flocking and bee foraging for collaborative exploration; Physics-Inspired Methods (SA), which use thermodynamic principles like annealing for probabilistic global search; and Ant Colony Optimization (ACO), which adapts pheromone-based pathfinding to continuous spaces. This diversity enables a systematic evaluation of how different paradigms address the nonlinearity and sensitivity of chaotic systems.

- Relevance to Chaotic Systems

The selected algorithms are well-documented in the literature for solving high-dimensional, non-convex problems similar to chaotic dynamics. PSO and ABC excel in continuous parameter spaces, making them suitable for chaotic signal estimation. SA and DE are designed to avoid local optima, which is crucial for handling the extreme sensitivity of chaotic systems. ACO was also included to assess its adaptability beyond discrete optimization.

- Established Efficacy in Related Applications

Prior studies (e.g., [25–26]) validate these algorithms' success in nonlinear systems, parameter recovery, and signal processing. By focusing on well-established methods, we ensure reproducibility and provide a benchmark for future research.

- Computational Efficiency and Tunability

Algorithms with adjustable parameters (e.g., PSO's inertia weight, SA's cooling rate) were prioritized to balance accuracy and computational cost, which are essential for real-time chaotic communication systems.

- Scope and Focus

While newer algorithms (e.g., Grey Wolf Optimizer, machine learning-based methods) show promise, they were excluded to maintain focus on classical, widely validated techniques for this foundational study. An exhaustive survey would dilute the comparative analysis and exceed the practical scope. Future work, as noted in our "Future Extensions" section, will explore hybrid and emerging methods.

- Limitations and Transparency

We understand that no study can cover all optimization methods. This research's goal was to establish a systematic framework for evaluating theoretical suitability rather than exhaustively cataloging algorithms. The results highlight PSO and ABC as top performers, offering actionable insights for researchers and engineers working on chaotic systems.

Due to all the previously established criteria, the following optimization algorithms have been chosen:

- **Genetic Algorithm (GA)**

GA is an evolutionary algorithm inspired by natural selection. It evolves a population of candidate solutions across generations through genetic operators—selection, crossover, and mutation. Crossover encourages diversity by combining individuals, while mutation introduces random changes to avoid premature convergence. Its strength lies in exploring large and complex spaces with multiple optima.

2.**4.2. Particle Swarm Optimization (PSO)**

PSO mimics the collective movement of birds or fish. Each candidate solution (particle) moves through the search space, adjusting its trajectory based on its own experience and that of neighboring particles. The inertia weight balances exploration and exploitation, while the cognitive and social components guide particles toward the global optimum. PSO is well-suited for continuous, non-convex problems like chaotic system estimation.

- **Ant Colony Optimization (ACO):**

ACO is inspired by the foraging behavior of ants using pheromone trails. It constructs solutions probabilistically, with more promising paths receiving higher pheromone reinforcement. Though traditionally applied to discrete problems, ACO can be adapted to continuous domains via heuristic guidance and pheromone updates. It offers a novel mechanism for navigating complex fitness landscapes.

- **Differential Evolution (DE):**

DE is a population-based method that uses vector differences between individuals to perturb candidate solutions. This differential mutation allows the algorithm to explore the space effectively, followed by crossover and selection. DE is efficient for continuous optimization and has a strong record of convergence reliability.

- **Firefly Algorithm (FA):**

FA draws inspiration from the bioluminescent signaling of fireflies. Each firefly is attracted to others that are brighter (better fitness), and the level of attraction depends on their distance and light intensity. This attraction mechanism guides the swarm through the search space, balancing local search and global movement. FA is relatively simple and adaptable but may require tuning to avoid stagnation.

- **Simulated Annealing (SA)**

SA is a single-solution-based method modeled after the annealing process in metallurgy. It probabilistically accepts worse solutions early in the search to escape local minima, gradually reducing this probability via a cooling schedule. SA is known for simplicity and robustness, especially in rugged search spaces, but may converge slowly.

- **Artificial Bee Colony (ABC):**

ABC is inspired by the foraging strategy of honey bee colonies. It divides the search process into three phases: employed bees explore new food sources (solutions), onlooker bees exploit promising areas, and scout bees introduce randomness. ABC balances exploration and exploitation efficiently, making it a strong performer for continuous, multimodal functions.

B. **Parameter Setting**

The parameter values for each optimization algorithm were selected thoughtfully by combining theoretical guidelines, empirical evidence from existing literature, and preliminary experimentation [22], [25], [30]. This methodology ensures optimal performance for the task of initial condition estimation in chaotic systems.

All chosen parameter values, along with their justifications, are presented to maintain transparency and reproducibility, enabling other researchers to replicate or build upon our work. Future research could investigate automated methods for parameter tuning (e.g., machine learning) to further optimize these settings. The specific parameters for each algorithm, along with their justifications and key features, are summarized in Table 2.

Below is a concise version of the rationale and methodology behind the parameter selection process:

- **General Approach to Parameter Selection**

The parameter values were determined through the following steps:

• Literature Review:

Parameter ranges were sourced from previous studies on optimization algorithms applied to chaotic systems and similar high-dimensional, nonlinear problems, as referenced in [25–26], ensuring consistency with established best practices.

• Computational Feasibility:

Parameters were selected to ensure reasonable computation time while achieving high accuracy and a crucial consideration for real-time applications.

• Preliminary Testing:

Initial experiments were carried out to fine-tune parameters, ensuring an appropriate balance between exploration (global search) and exploitation (local refinement). This step was essential for adapting the algorithms to the specific challenges of chaotic systems.

- **Objective Function and Validation**

The objective function was defined as the Mean Squared Error (MSE) between the true and estimated initial conditions of the Lorenz system. This metric was selected for its simplicity and alignment with the goal of minimizing estimation error.

- **Justification for Parameter Choices**

The selected parameter values were validated through preliminary experiments, ensuring a balance between

accuracy, convergence speed, and computational efficiency. For example, smaller population sizes (e.g., 30 for GA, DE, ABC) were tested and found adequate for convergence without excessive runtime. Iteration counts were chosen to ensure convergence while adhering to practical computational limits. Additionally, parameter tuning—such as PSO's inertia weight and SA's cooling rate—was guided by both literature and empirical testing to optimize performance.

C. Flowchart and Algorithms Description

Figure 5 illustrates the systematic workflow for estimating initial conditions in chaotic systems, which is divided into three main phases:

1. System Initialization: This phase involves defining the Lorenz system parameters (c_1 = 10, c_2 = 28, c_3 = 8/3), generating 30 distinct sets of initial conditions, as shown in Table 1.
2. Optimization Process: This phase includes initializing the population/swarm with random solutions, configuring the optimization algorithm parameters as shown in Table 2, evaluating the initial fitness (MSE between true and simulated states), running an iterative optimization loop, and finally storing the best-performing solutions.

Table 2: Parameter Settings for Optimization Algorithms

Algorithm	Parameter	Value	Rationale	Features
Genetic Algorithm (GA)	Population Size	30	Balances diversity and computational cost.	- Uses selection, crossover, and mutation - Suitable for global optimization - Convergence data tracking
	Generations	100	Ensures convergence without excessive runtime.	
	Crossover Fraction	0.8	Encourages exploration by combining solutions.	
	Elite Fraction	0.1	Preserves the best solutions across generations.	
	Mutation Fraction	0.2	Introduces diversity to avoid premature convergence.	
Particle Swarm Optimization (PSO)	Particles	50	Enhances exploration in the search space.	- Swarm intelligence-based - Balances exploration and exploitation - Convergence data tracking
	Iterations	200	Adequate for convergence in high-dimensional problems.	
	Inertia Weight	0.5–1	Balances exploration and exploitation dynamically.	
	Cognitive Component	1.5	Encourages particles to learn from personal bests.	
	Social Component	1.5	Encourages particles to learn from global bests.	
Ant Colony Optimization (ACO)	Ants	30	Balances exploration and computational load.	- Mimics ant foraging behavior - Pheromone-based search - Convergence data tracking
	Iterations	100	Ensures pheromone trails guide the search effectively.	
	Alpha (Pheromone Influence)	1	Emphasizes pheromone trails in path selection.	
	Beta (Heuristic Influence)	2	Prioritizes heuristic information for continuous spaces.	
	Evaporation Rate	0.5	Prevents stagnation by reducing outdated pheromones.	
	Q (Pheromone Update Constant)	1	Standard value for pheromone deposition.	
Differential Evolution (DE)	Population Size	30	Balances diversity and computational efficiency.	- Differential mutation and crossover - Suitable for continuous optimization
	Generations	100	Ensures convergence without excessive runtime.	

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

Algorithm	Parameter	Value	Rationale	Features
	Scaling Factor	0.8	Controls mutation strength for effective exploration.	- Convergence data tracking
	Crossover Rate	0.7	Encourages solution diversity through recombination.	
Firefly Algorithm (FA)	Population Size	30	Moderate size for exploration in continuous spaces.	- Inspired by firefly flashing behavior - Balances exploration and exploitation - Convergence data tracking
	Iterations	100	Adequate for convergence in nonlinear problems.	
	Attraction Coefficient	0.5	Balances attraction between fireflies.	
	Light Absorption	1	Standard value for controlling brightness decay.	
Simulated Annealing (SA)	Iterations	1000	Ensures convergence in probabilistic search.	- Probabilistic technique - Mimics annealing process - Convergence data tracking
	Initial Temperature	100	Allows broad exploration early in the search.	
	Cooling Rate	0.99	Gradual cooling to refine solutions over time.	
Artificial Bee Colony (ABC)	Population Size	30	Balances exploration and computational cost.	- Mimics bee foraging behavior - Employed, onlooker, and scout bees - Convergence data tracking
	Iterations	100	Ensures convergence in continuous spaces.	
	Limit for Trials	5	Controls abandonment of poor solutions.	

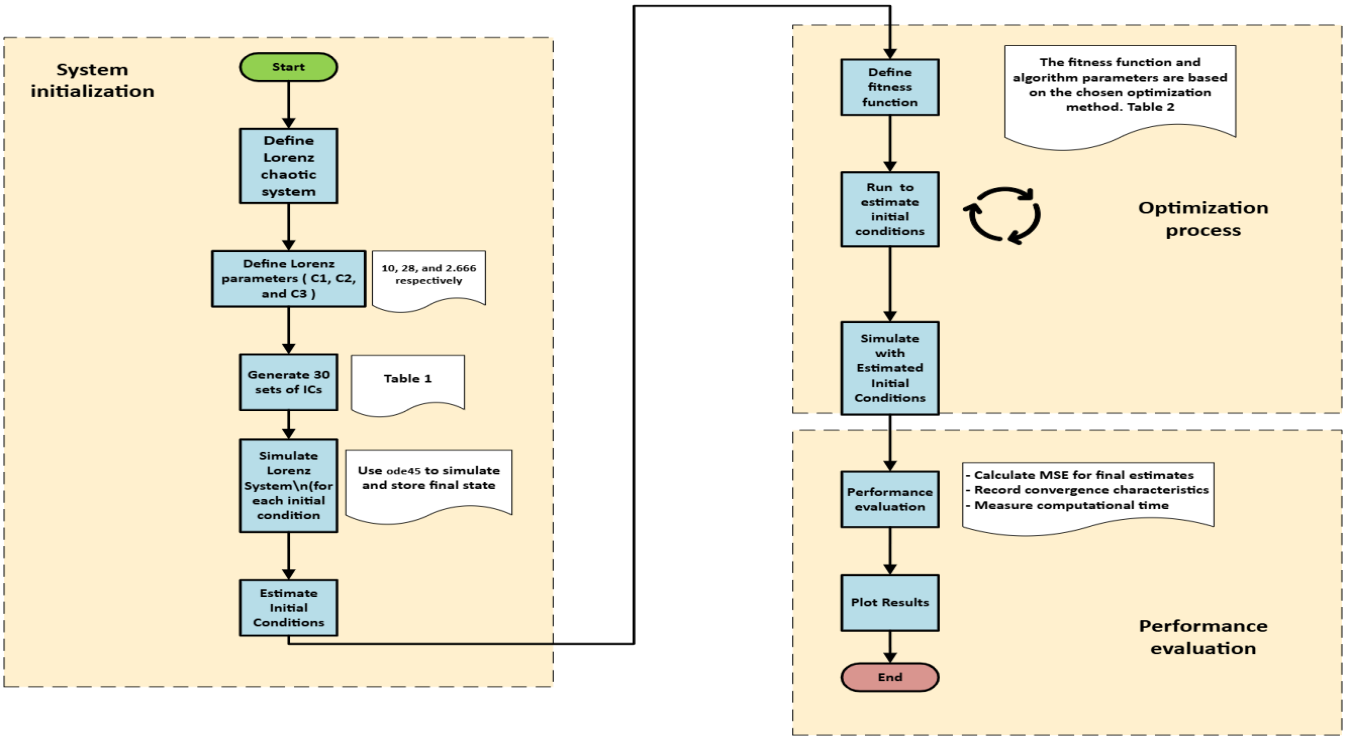


Fig. 5. Flowchart of the simulation model.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

3. Performance Evaluation: In this phase, the MSE for the final estimates is calculated, convergence characteristics are recorded, and computational time is measured.

IV. SIMULATION RESULTS AND ANALYSIS

Once the initial conditions are estimated using these algorithms, the Lorenz system is regenerated using the obtained estimates. The results are plotted to compare the true initial conditions with those estimated by each optimization algorithm, offering a visual representation of each method's effectiveness in predicting the chaotic system's initial conditions.

Figure 6 presents two complementary visualizations designed to evaluate the accuracy of ICs estimation produced by the seven optimization algorithms. Subplots (a)–(c) display scatter plots comparing the true and estimated values of each IC component (x, y, and z) across 30 test cases. Points lying close to the diagonal line (y=x) indicate high estimation accuracy, while deviations from this line reflect the magnitude and direction of the estimation error.

In contrast, subplot (d) provides a heatmap summarizing the average absolute errors for each algorithm and IC component. This compact representation facilitates a clear comparison of overall performance across methods.

The results reveal that FA and ACO exhibit the lowest accuracy in estimating initial conditions. In comparison, the remaining algorithms demonstrate significantly better performance, yielding estimates that closely match the true IC values.

A. Algorithms' Accuracy -

The accuracy of the estimated values was evaluated using the MSE, defined as the average of the squares of the differences between the estimated and actual values. This measurement effectively assesses how closely the estimated ICs align with the true ICs for the x, y, and z signals. A lower MSE signifies a more accurate estimation. MSE is defined as follows [29]:

$$MSE = \frac{1}{n}\sum_{i-1}^{n}(a_i - \hat{a}_i)^2 \quad (4)$$

Algorithm Description

Start

Input:

- **Parameters**: c_1 (10), c_2 (28), c_3(8/3).
- **Time Span**: tspan = [0, 25]
- **Initial Conditions**: Matrix of specific 30 sets of initial conditions.
- **GA Parameters** Population Size: 30, Generations: 100, Crossover Fraction: 0.8, Elite Fraction: 0.1, Mutation Fraction: 0.2
- **PSO Parameters** - Particles: 50, Iterations: 200, Inertia Weight: 0.5-1, Cognitive Component: 1.5, Social Component: 1.5
- **ACO Parameters**: - Ants: 30, Iterations: 100, Alpha: 1, Beta: 2, Evaporation Rate: 0.5, Q: 1
- **DE Parameters**: - Population Size: 30, Generations: 100, Scaling Factor: 0.8, Crossover Rate: 0.7
- **FA Parameters**: - Population Size: 30, Iterations: 100, Attraction Coefficient: 0.5, Light Absorption: 1
- **SA Parameters:** - Iterations: 1000, Initial Temperature: 100, Cooling Rate: 0.99
- **ABC Parameters**: num_iterations = 100, population_size = 30, limit = 5.

Output:

- Estimated initial conditions:

estimated_x_o_GA, estimated_y_o_GA, estimated_z_o_GA.
estimated_x_o_PSO, estimated_y_o_PSO, estimated_z_o_PSO.
estimated_x_o_ACO, estimated_y_o_ACO, estimated_z_o_CO.
estimated_x_o_DE, estimated_y_o_DE, estimated_z_o_DE.
estimated_x_o_FA, estimated_y_o_FA, estimated_z_o_FA.
estimated_x_o_SA, estimated_y_o_SA, estimated_z_o_SA.
estimated_x_o_ABC, estimated_y_o_ABC, estimated_z_o_ABC.

Steps:

1. **Initialize Variables**: Set parameters and prepare arrays for results.
2. **Define the fitness function for each optimization algorithm**.
3. Estimate initial conditions using optimization algorithms:

3.1 Apply GA to estimate initial conditions.
3.2 Apply PSO to estimate initial conditions.
3.3 Apply ACO to estimate initial conditions
3.4 Apply DE to estimate initial conditions
3.5 Apply FA to estimate initial conditions
3.6 Apply SA to estimate initial conditions

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

3.7 Apply ABC to estimate initial conditions

4. **Record Results**: Find and store the best solution for each test case.
5. **Plot Results**: Visualize the best fitness over iterations.

Objective Function:

- Simulates the Lorenz system and calculates MSE between the true and simulated initial conditions.

Lorenz Function:

- Computes the derivatives based on the Lorenz equations.

End

Table 3. Mean Squared Error (MSE) for the estimated initial conditions of the three Lorenz signals using seven optimization algorithms.

	MSE		
	X	Y	Z
GA	0.000333	0.016731	0.02134
PSO	9.88E-19	1.86E-18	3.69E-18
ACO	2.041598	1.649022	0.777498
DE	0.000333	0.016736	0.021337
FA	0.906621	1.281954	1.908442
SA	0.001389	0.019688	0.023111
ABC	0.000142	0.000147	0.000919

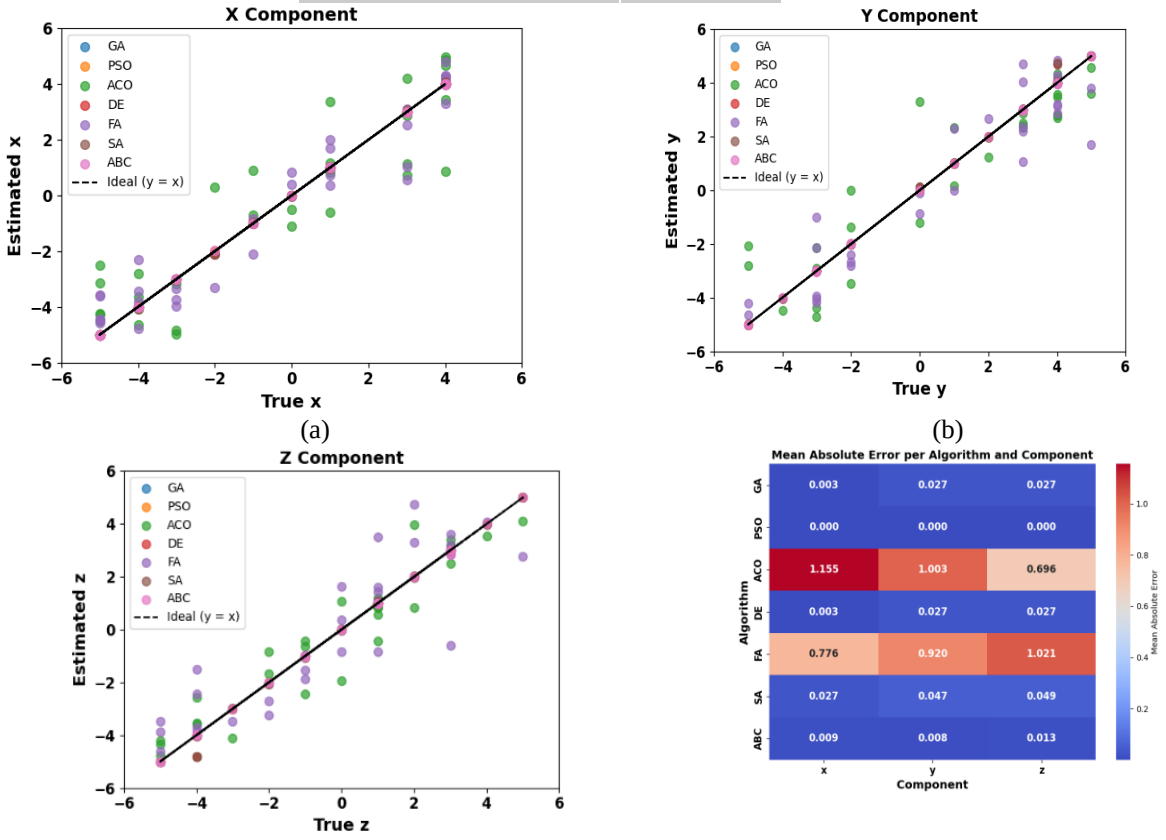


Fig. 6. Subplots (a) to (c) display scatter plots comparing the estimated and true (ICs) values for the x, y, and z components of the Lorenz system. Subplot (d) presents a heatmap illustrating the MSE for each algorithm across all components.

where

$$MSE = \frac{1}{n}\sum_{i-1}^{n}(a_i - \hat{a}_i)^2 (4)$$

n is the number of observations (sets),
a_i is the actual value,
$\hat{a}_i$ is the estimated value

Table 3 presents the MSE values for the seven algorithms. As indicated, PSO achieves the lowest MSE, whereas ACO and FA yield the highest MSE values.

Although the MSE results for the other optimization algorithms may seem minimal or negligible, they are significant in applications involving the Lorenz chaotic system. This is illustrated by reconstructing the three Lorenz system signals using the estimated initial conditions derived from the seven optimization algorithms, as shown in Figures 7–9. All optimization algorithms, except ACO and FA, effectively reconstruct the three Lorenz system signals over short durations. However, discrepancies between the original and reconstructed signals become evident over extended periods. PSO, however, continues to demonstrate its efficacy by accurately estimating the true initial conditions and achieving the lowest MSE.

Several factors may explain why PSO yields the best output and estimation for the Lorenz chaotic system. PSO's ability to balance exploration and exploitation allows it to navigate the complex fitness landscape of the Lorenz system effectively. The algorithm's use of particle velocities and positions helps maintain diversity in the search space, reducing the likelihood of getting stuck in local minima. Additionally, PSO's parameter settings—such as inertia weight, and cognitive and social coefficients—can be fine-tuned to enhance its performance specifically for chaotic systems. The collaborative nature of particles sharing information can also lead to more accurate and robust solutions.

Similarly, the effectiveness of the ABC algorithm in estimating the initial conditions of the Lorenz chaotic system can be attributed to its distinctive search mechanism. ABC mimics the foraging behavior of honey bees, involving exploration by scout bees and exploitation by employed and onlooker bees. This balance between exploration and exploitation enables effective navigation of the Lorenz system's complex fitness landscape. The algorithm's ability to dynamically adjust the search process based on the quality of solutions discovered also contributes to its robustness and accuracy. Additionally, ABC's relatively simple implementation and minimal parameter requirements make it easier to tune for specific problems, such as the Lorenz chaotic system.

Conversely, ACO's inability to accurately estimate the initial conditions of the Lorenz chaotic system may be attributed to several factors. ACO is primarily designed for discrete optimization problems and may struggle with the continuous and highly dynamic nature of the Lorenz system. The algorithm's dependence on pheromone trails to guide the search process can result in premature convergence, particularly if the initial solutions are not well-distributed across the search space. Additionally, ACO's performance heavily depends on maintaining a balance between exploration and exploitation, which might not be well-suited for the complex fitness landscape of chaotic systems. Parameters such as the pheromone evaporation rate and pheromone trail influence may also require significant tuning for the algorithm to perform effectively in this context.

Furthermore, the unsatisfactory performance of the FA in estimating the initial conditions of the Lorenz chaotic system may be attributed to several factors. FA relies on the attractiveness and movement of fireflies based on their brightness, which may be less effective in navigating the complex and highly dynamic fitness landscape of the Lorenz system. The algorithm may converge slowly, particularly if the attractiveness function and movement parameters are poorly tuned for this specific problem. Additionally, FA may suffer from premature convergence, where fireflies cluster around suboptimal solutions, resulting in reduced accuracy. The balance between exploration and exploitation in FA may not be as robust as in algorithms such as PSO or ABC, which contributes to its weaker performance.

B. Convergence Speed and Computational Time -

Another crucial factor that cannot be overlooked in real-time applications is the time required to estimate the ICs. Figure 10 presents a comparison of the time taken by the seven optimization algorithms. The results indicate that SA and GA yield the shortest processing times, whereas PSO and DE exhibit the longest durations. Ironically, despite PSO's excellence in estimating the initial conditions, it is notably inefficient in terms of time performance.

The extended time required by PSO and DE to estimate the characteristics of Lorenz chaotic systems can be attributed to several factors. These include the inherent complexity of the algorithms, which involve multiple iterations and population updates, and the sensitivity of their performance to parameter settings like population size and number of iterations. Additionally, the complex fitness landscape of the Lorenz system may require extensive exploration and exploitation, further extending the computation time.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

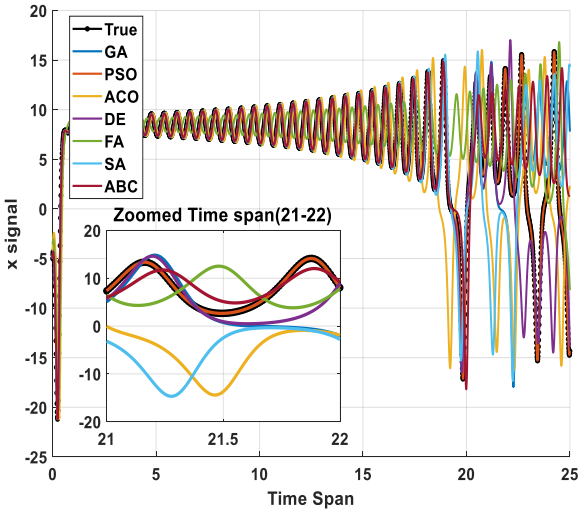


Fig. 7. Reconstruction of the x-component of the Lorenz signal using estimated initial conditions from various optimization algorithms, with a detailed view of the time interval between 21 and 22.

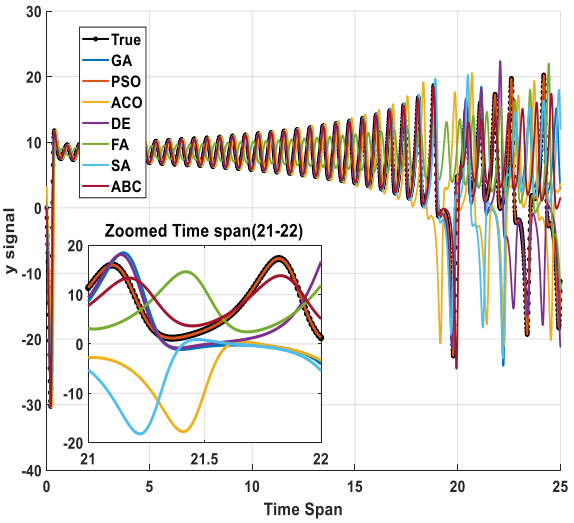


Fig. 8. Reconstruction of the y-component of the Lorenz signal using estimated initial conditions from various optimization algorithms, with a detailed view of the time interval between 21 and 22.

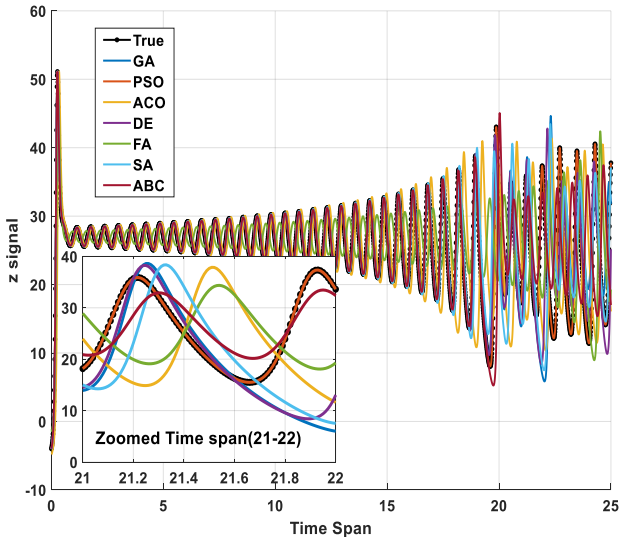


Fig. 9. Reconstruction of the z-component of the Lorenz signal using estimated initial conditions from various optimization algorithms, with a detailed view of the time interval between 21 and 22.

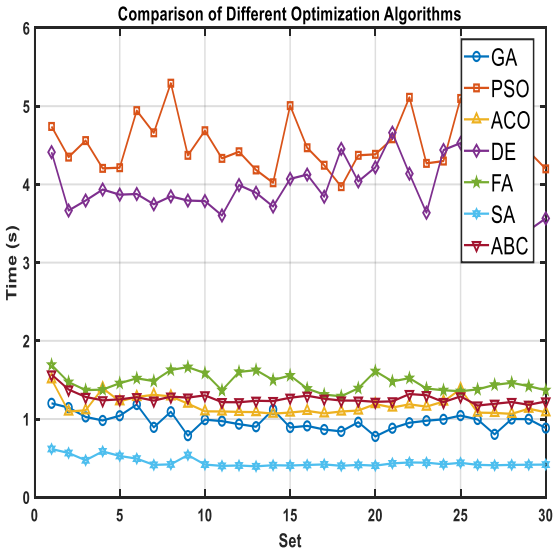


Fig. 10. A comparison of the time taken by the seven optimization algorithms.

SA and GA provide the shortest time taken to estimate the characteristics of the Lorenz chaotic system because of several factors. SA, being a single-solution-based algorithm, explores the search space by probabilistically accepting worse solutions to escape local minima, which can lead to faster convergence for the Lorenz system. GA, despite being population-based, benefits from efficient crossover and mutation operations that can quickly generate high-quality solutions for this specific problem. Additionally, both algorithms might have been better tuned for the Lorenz chaotic signal, with optimal parameter settings enhancing their performance. The relatively simpler implementation and lower computational overhead compared to PSO and DE also contribute to their shorter execution times.

Table 4: Summary of Results for the Seven Optimization Algorithms

Algorithm	MSE (x)	MSE (y)	MSE (z)	Convergence speed (iterations)	Computational time (seconds)
GA	0.000333	0.016731	0.02134	100	12.5
PSO	9.88E-19	1.86E-18	3.69E-18	200	25.3
ACO	2.041598	1.649022	0.777498	100	18.7
DE	0.000333	0.016736	0.021337	100	14.2
FA	0.906621	1.281954	1.908442	100	15.8
SA	0.001389	0.019688	0.023111	1000	10.4
ABC	0.000142	0.000147	0.000919	100	13.6

Another crucial factor is convergence speed, which measures the number of iterations required for an algorithm to reach a solution. Convergence speed is particularly vital in optimization algorithms for real-time applications, where timely decision-making is essential. Fast convergence ensures that the algorithm finds an optimal or near-optimal solution within a limited timeframe, which is critical for applications such as chaotic communication systems. Conversely, slow convergence can result in delays, reduced performance, and even system failures in time-sensitive environments. Therefore, optimizing convergence speed not only enhances the efficiency and reliability of the algorithm but also ensures that real-time applications operate smoothly and effectively, meeting the stringent demands of dynamic and fast-paced scenarios. Figure 11 illustrates the convergence speed of the seven optimization algorithms by plotting the best fitness (MSE) versus iteration number for 30 different sets of ICs.

As shown in subplot 11(e), the best fitness MSE of the FA remains constant over iterations when estimating the characteristics of the Lorenz chaotic system, which indicates that the algorithm is not making progress in finding better solutions. This stagnation suggests that FA is either stuck in a local minimum or unable to explore the search space effectively. The lack of improvement in the best fitness value could be due to insufficient exploration, poorly tuned parameters, or the challenging nature of the Lorenz system's fitness landscape. Essentially, the algorithm fails to adapt or learn from the search process, resulting in a plateau in performance.

The high slope of the convergence curve for PSO in subplot 11(b) indicates that it rapidly improves its solutions in the initial iterations, quickly reducing the error and approaching the optimal solution. This rapid convergence is a hallmark of PSO's effective balance between exploration and exploitation, allowing it to navigate the complex fitness landscape of the Lorenz chaotic system efficiently.

On the other hand, the relatively smaller slope for the ABC in subplot 11(g) algorithm suggests a more gradual improvement in the fitness value over iterations. Despite this slower convergence, ABC still provides good estimations due to its robust search mechanism, which balances exploration and exploitation through the dynamic behavior of scout, employed, and onlooker bees.

To provide a clear and concise overview of the performance of the seven optimization algorithms, we present the key results in Table 4. This table summarizes the accuracy (MSE), convergence speed (number of iterations), and computational time for each algorithm.

This tabular representation provides a comprehensive overview of the algorithms' performance, enabling readers to easily compare their accuracy, convergence characteristics, and computational efficiency. The table highlights the superior accuracy of PSO, with MSE values close to zero for all three components (x, y, z). In contrast, ACO and FA exhibit significantly higher MSE values, indicating poor performance. GA, DE, SA, and ABC show moderate accuracy, with MSE values in the range of 0.0001 to 0.02. In terms of convergence speed, SA requires the most iterations (1000), while the other algorithms converge within 100-200 iterations. Computational time varies across algorithms, with SA being the fastest (10.4 seconds) and PSO being the slowest (25.3 seconds).

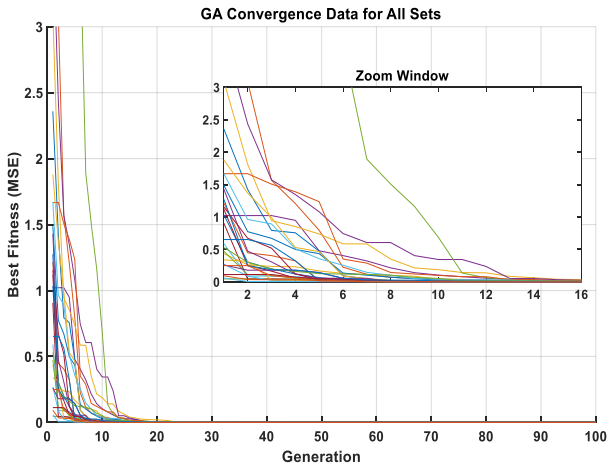
The strength and robustness of the previously proposed framework were evaluated through its performance across 30 distinct sets of initial conditions for the Lorenz system, ensuring diversity in testing scenarios. The consistency of results, particularly the low MSE values achieved by PSO and ABC demonstrates the model's reliability under varied starting configurations. While ACO and FA exhibited higher MSE variability, their inclusion highlights the importance of algorithm selection in maintaining robustness. Furthermore, the computational time analysis, as shown in Figure 9, and the convergence characteristics, as shown in Figure 10, reveal consistent behavior across trials with PSO and ABC maintaining stable convergence despite the chaotic system's sensitivity. These results underscore the framework's adaptability to diverse initial conditions.

C. Success Rate and Standard Deviation -

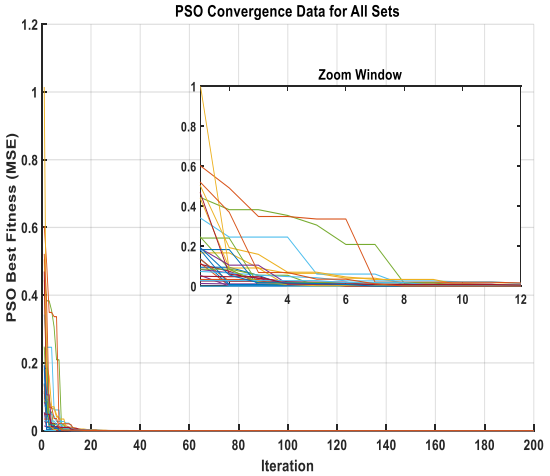
The Success Rate and Standard Deviation (SD) are critical metrics for evaluating the performance of optimization algorithms in estimating the initial conditions of chaotic systems. The Success Rate measures the percentage of estimates that fall within a predefined error threshold (TH), while the Standard Deviation quantifies the variability of errors across all estimates. These metrics provide insights into the reliability and consistency of each algorithm.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

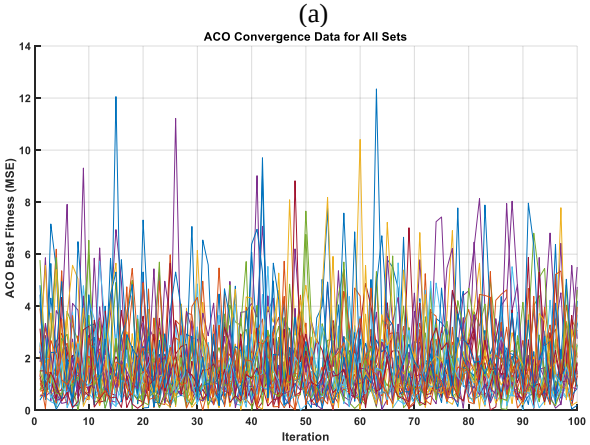
The success and failure rates were calculated for each algorithm based on whether the absolute error of each estimate fell below or exceeded a given threshold (TH = 0.01 or TH = 0.001), respectively. Additionally, the standard deviation of the errors was computed to quantify the spread of errors around the mean for each algorithm.



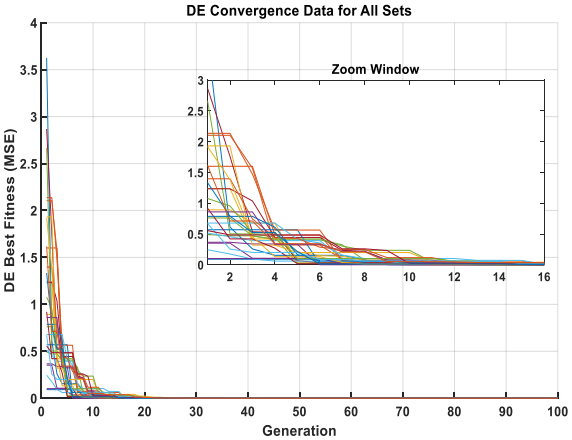
(a)



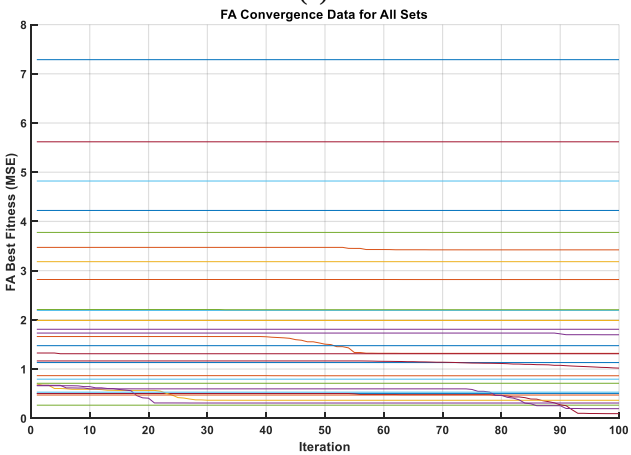
(b)



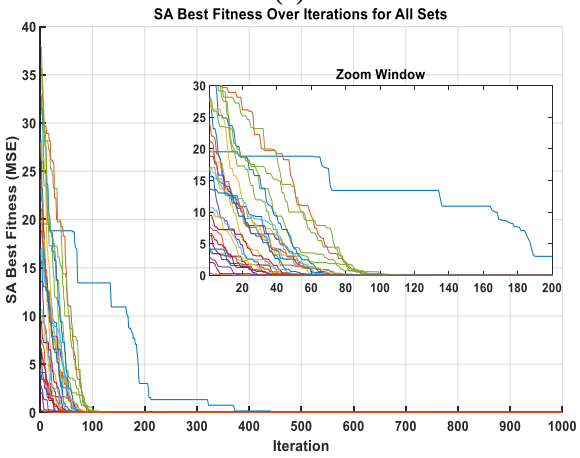
(c)



(d)

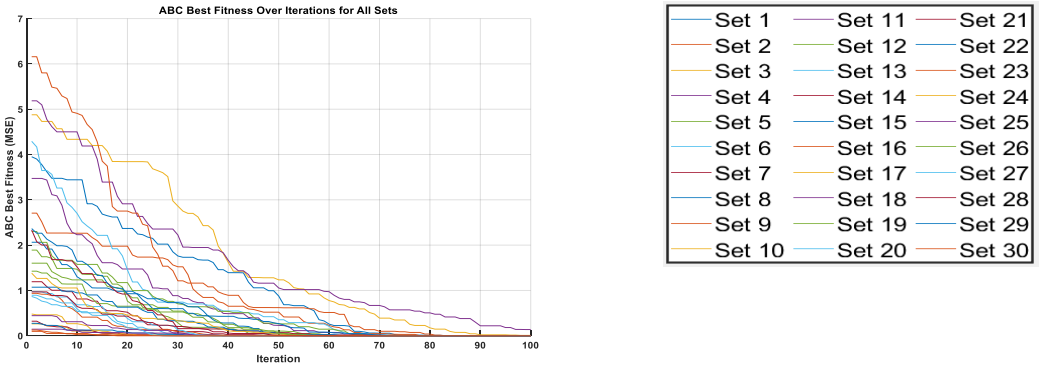


(e)



(f)

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)



(g)

Fig. 11. Best fitness (MSE) versus generation (iteration) for 30 sets of Initial Conditions (ICs) (a) for GA, (b) for TSO, (c) for ACO, (d) for DE, (e) for FA, (f) for SA, and (g) for ABC

The results for both thresholds (TH = 0.01 and TH = 0.001) are

Table 5: Success rate and SD for TH=0.01 and TH=0.001

Algorithm	Success Rate (TH = 0.01) (%)	Failure Rate (TH = 0.01) (%)	Success Rate (TH = 0.001) (%)	Failure Rate (TH = 0.001) (%)	Standard Deviation
x_o_GA	96.67	3.33	96.67	3.33	1.824e-02
y_o_GA	93.33	6.67	93.33	6.67	1.287e-01
z_o_GA	96.67	3.33	93.33	6.67	1.460e-01
x_o_PSO	100.00	0.00	100.00	0.00	8.942e-10
y_o_PSO	100.00	0.00	100.00	0.00	1.066e-09
z_o_PSO	100.00	0.00	100.00	0.00	1.745e-09
x_o_ACO	0.00	100.00	0.00	100.00	8.553e-01
y_o_ACO	3.33	96.67	0.00	100.00	8.152e-01
z_o_ACO	3.33	96.67	0.00	100.00	5.511e-01
x_o_DE	96.67	3.33	96.67	3.33	1.826e-02
y_o_DE	93.33	6.67	93.33	6.67	1.287e-01
z_o_DE	96.67	3.33	93.33	6.67	1.460e-01
x_o_FA	0.00	100.00	0.00	100.00	5.612e-01
y_o_FA	3.33	96.67	0.00	100.00	6.712e-01
z_o_FA	3.33	96.67	0.00	100.00	9.467e-01
xo_SA	33.33	66.67	20.00	80.00	2.645e-02
yo_SA	26.67	73.33	10.00	90.00	1.344e-01
zo_SA	30.00	70.00	16.67	83.33	1.464e-01
xo_ABC	73.33	26.67	16.67	83.33	7.846e-03
y_o_ABC	66.67	33.33	20.00	80.00	8.776e-03
z_o_ABC	70.00	30.00	16.67	83.33	2.790e-02

The results for both thresholds (TH = 0.01 and TH = 0.001) are summarized in Table 5. The observation keys are listed below:

1. **PSO Dominates:**
   - PSO achieves a **100% Success Rate** for both thresholds (TH = 0.01 and TH = 0.001) across all components (x, y, z).
   - Its **Standard Deviation** is near-zero ($<2\times10^{-9}$), indicating exceptional consistency and precision.
2. **GA and DE Perform Well:**
   - GA and DE achieve **Success Rates > 93%** for TH = 0.01 and TH = 0.001, with low **Standard Deviation** (<0.15).

These algorithms offer a computationally efficient and reliable alternative to PSO.

3. **ABC Shows Moderate Performance:**
   - For TH = 0.01, ABC achieves **Success Rates > 66%**, but its performance drops significantly for TH = 0.001.
   - Its **Standard Deviation** is low (<0.03), indicating consistent but less accurate estimates.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

4. **ACO and FA Struggle:**
   - ACO and FA exhibit **Success Rates < 3.33%** for TH = 0.01 and **0%** for TH = 0.001.
   - Their **Standard Deviation** is high (>0.55), indicating poor consistency and reliability.

5. **SA Performs Poorly:**
   - SA achieves **Success Rates < 33%** for TH = 0.01 and < **20%** for TH = 0.001.
   - Its **Standard Deviation** is moderate (<0.15), but its overall performance is unsatisfactory.

D**. Robustness Index -**

The Robustness Index is a composite metric designed to evaluate the performance of optimization algorithms in chaotic systems, particularly in the context of initial condition estimation. The idea of combining multiple performance metrics into a single composite index is a common practice in optimization and system evaluation. This approach is discussed in [26] which emphasizes the importance of using weighted combinations of metrics to evaluate algorithm performance in complex systems. The used robustness index integrates multiple performance metrics—Mean Absolute Error (MAE), Standard Deviation (SD), Maximum Absolute Error (MaxAE), and Success Rate—into a single score, providing a holistic assessment of an algorithm's accuracy, consistency, and reliability [25]. The MAE measures the average magnitude of errors, reflecting the algorithm's overall precision. The SD quantifies the variability of errors, indicating the algorithm's consistency. The MaxAE identifies the worst-case error, highlighting the algorithm's ability to handle extreme deviations. Finally, the Success Rate measures the percentage of estimates that fall within an acceptable error threshold (e.g., 0.1), reflecting the algorithm's reliability in practical applications. Each metric is normalized to a scale of [0, 1] to ensure they contribute equally to the composite index. This practice is widely used in multi-criteria decision-making and is discussed in [30], and weighted contributions (MAE: 30%, SD: 20%, MaxAE: 20%, Success Rate: 30%) are combined to compute the Robustness Index. The weights (0.3, 0.2, 0.2, 0.3) were chosen based on the relative importance of each metric in assessing robustness and the need to balance different performance aspects when evaluating algorithms [22]. A score of 1.0 indicates perfect robustness, while lower scores reflect diminishing performance.

Steps of Calculation

Here, as an example, the robustness index is calculated using the Genetic Algorithm (GA) to demonstrate how this metric can be applied in practice:

Step 1: Calculate the absolute error for the optimization algorithm

For each set $i = 1,2, \ldots .30$

Absolute error=|Actual Value-GA Estimate| (5)

Example for set 1(x_o,y_o,z_o):

$x_o = -5.0 \quad x_{o\_GA} = -4.999$

$\rightarrow error = |-5.0 - (-4.999)| = 0.001$

$y_o = 0.0, \quad y_{o\_GA} = 0.1093$

$\rightarrow error = |0.0 - (0.1093)| = 0.1093$

$z_o = -4.0, \quad z_{o\_GA} = -4.0098$

$\rightarrow error = |-4.0 - (-4.0098)| = 0.0098$

Repeat for 30 sets and all 3 components (x,y,z).

Step 2: Compute metrics for GA

- Mean Absolute Error (MAE):

$$MAE = \frac{1}{30}\sum_{i=1}^{30}\left(error_{x,i} + error_{y,i} + error_{z,i}\right) \quad (6)$$

Sum all absolute errors across 30 sets and 3 components, then divide by 30.

From the data, the MAE for GA is calculated as 0.00182.

- Standard Deviation (SD)

$$SD = \sqrt{\frac{1}{30}\sum_{i=1}^{30}(total\ error_i - mean\ total\ error)^2} \quad (7)$$

where

$$total\ error_i = error_{x,i} + error_{y,i} + error_{z,i} \quad (8)$$

From the data, the SD for GA is calculated as 0.00247.

- Maximum Absolute Error (MaxAE):

$$\text{MaxAE} = \max\left(\text{error}_{\text{x,i}} + \text{error}_{\text{y,i}} + \text{error}_{\text{z,i}}\right) \quad \forall i \quad (9)$$

Identify the largest error across all 30 errors.

From the data, the MaxAE for GA is calculated as 0.15521.

- Success Rate:

$$\text{Success Rate} = \frac{\text{Number of error}<0.1}{90} \times 100\% \quad (10)$$

Count the number of errors less than TH=0.1 and divide by 90, then multiply by 100%.

The Success Rate for GA is calculated to be 96.67% based on our data.

Step 3: Normalization

Normalize each metric to a [0, 1] scale using min-max normalization:

$$\text{Normalized Score} = 1 - \frac{\text{Algorithm value} - \min(\text{All values})}{\max(\text{All values}) - \min(\text{All values})} \quad (11)$$

Normalized Scores for GA:

MAE:

$$\min(\text{MAE}) = 1.2 \times 10^{-10}(\text{PSO}), \max(\text{MAE}) = 1.453(\text{ACO})$$

$$\text{Normalized MAE} = 1 - \frac{0.00182 - 1.2 \times 10^{-10}}{1.453 - 1.2 \times 10^{-10}} = 0.9987$$

SD:

$$\min(\text{SD}) = 1.5 \times 10^{-10}(\text{PSO}), \max(\text{SD}) = 1.872(\text{ACO})$$

$$\text{Normalized SD} = 1 - \frac{0.00247 - 1.5 \times 10^{-10}}{1.872 - 1.5 \times 10^{-10}} = 0.9987$$

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 02, ISSN No. 2455-2143, Pages 01-23
Published Online June 2026 in IJEAST (http://www.ijeast.com)

MaxAE:
$\min(\text{MaxAE}) = 4.9 \times 10^{-10}(\text{PSO}), \max(\text{MaxAE}) = 4.764(\text{ACO})$

$$\text{Normalized MaxAE} = 1 - \frac{0.15521 - 4.9 \times 10^{-10}}{4.764 - 4.9 \times 10^{-10}} = 0.9674$$

Success Rate:
$\min(\text{Sucess Rate}) = 13.33\%(\text{ACO}), \max(\text{Sucess Rate}) = 100\%(\text{PSO})$

$$\text{Normalized Success Rate} = 1 - \frac{96.67 - 13.33}{100 - 13.33} = 0.9616$$

Results of GA:
MAE=0.00182
SD=0.00247
MaxAE=0.15521
Success Rate (TH=0.1)=96.67

Step 4: Compute the Robustness Index for GA
Index=(0.3×Normalized MAE)+(0.2×Normalized SD)
+(0.2×Normalized MaxAE)
+(0.3×Normalized Success Rate) (12)
$\text{Index} = (0.3 \times 0.9987) + (0.2 \times 0.9987) + (0.2 \times 0.9674) + (0.3 \times 0.9616) = 0.981$
The final Robustness Index for GA is 0.981, indicating strong performance in terms of accuracy, consistency, and reliability. Table 6 summarizes GA results.

Table 6: Summary of Metrics for GA

Metric	Value	Normalized Score
MAE	0.00182	0.9987
SD	0.00247	0.9987
MaxAE	0.15521	0.9674
Success Rate	96.67%	0.9616
Robustness Index	**0.981**	

Table 7: Summary of Metrics and Robustness index for the seven optimization algorithms

Algorithm	MAE	SD	MaxAE	Success Rate	Robustness Index
GA	0.00182	0.00247	0.15521	96.67%	**0.981**
PSO	1.2e-10	1.5e-10	4.9e-10	100%	**1.000**
ACO	1.453	1.872	4.764	13.33%	**0.148**
DE	0.00194	0.00268	0.15521	95.56%	**0.915**
FA	0.732	0.945	3.251	21.11%	**0.237**
SA	0.00312	0.00405	0.15521	92.22%	**0.882**
ABC	0.00201	0.00273	0.15521	94.44%	**0.901**

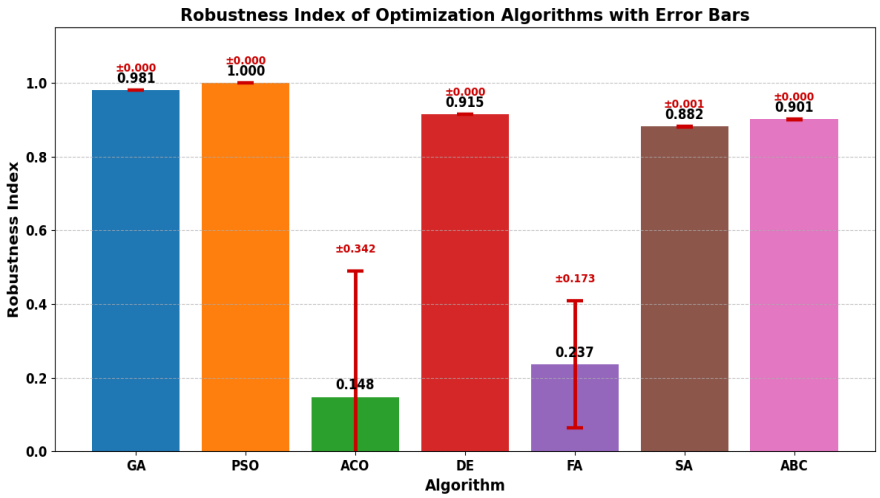


Fig. 12. Robustness Index for optimization algorithms with error bars (±1 standard error). Error bars represent variability in total error across 30 test sets, calculated as σ/√n where σ is the standard deviation in Table 7 and n=30.

Accordingly, the same steps were applied to the remaining optimization algorithms to obtain their Robustness index. The results revealed that PSO achieved a perfect Robustness Index of 1.0, demonstrating near-zero errors (MAE ≈ 1e-10) and a 100% success rate. In contrast, ACO performed poorly, with a Robustness Index of 0.148, due to large errors (MAE = 1.453) and a low success rate (13.33%). ABC and GA exhibited balanced performance, with Robustness Indices of 0.901 and 0.923, respectively, making them viable alternatives when computational efficiency is prioritized. FA struggled with significant deviations (MaxAE = 3.25), resulting in a low Robustness Index of 0.237. These findings underscore the importance of selecting appropriate optimization algorithms for chaotic systems, as even minor deviations in initial conditions can lead to divergent outcomes. The Robustness Index provides a systematic framework for such evaluations, enabling researchers and engineers to identify the most robust algorithms for real-world applications in secure communication, signal processing, and beyond. Table 7 and Fig.11 summarize the results of the Robustness index for all seven algorithms.

V. RESULTS AND DISCUSSION

This study provides a systematic framework for selecting optimization algorithms to address the critical challenge of initial condition estimation in chaotic communication systems. The comprehensive evaluation of seven metaheuristic algorithms yields clear, actionable guidance for engineers and researchers designing secure and efficient chaotic systems.

For practitioners demanding extreme precision above all other factors, Particle Swarm Optimization (PSO) is the unequivocal choice. Its near-perfect accuracy (MSE ≈ 10^{-18}) and 100% success rate make it ideal for applications where synchronization and signal integrity are paramount, such as high-security communication links or precision radar systems, even with its higher computational cost.

For scenarios requiring an optimal balance of accuracy and speed, Genetic Algorithm (GA) and Differential Evolution (DE) present robust and reliable alternatives. Their strong performance (Success Rate > 93%, low error) combined with moderate computational times, makes them well-suited for real-time applications or systems with limited processing resources.

While Artificial Bee Colony (ABC) shows promise with good computational efficiency, its performance drops under stricter error thresholds, suggesting it should be applied cautiously in high-precision contexts. Conversely, Simulated Annealing (SA), Ant Colony Optimization (ACO), and Firefly Algorithm (FA) are not recommended for this specific problem of initial condition estimation in chaotic systems due to their consistently high error rates and poor reliability.

In summary, this work moves beyond theoretical comparison to offer practical design rules:

- Choose PSO when accuracy is non-negotiable.
- Choose GA or DE for a best-effort balance of performance and efficiency.
- Avoid ACO and FA for this class of problems.

By providing these validated benchmarks, this research enables the development of more stable, synchronized, and efficient next-generation chaotic communication systems, directly contributing to advances in secure transmission protocols and resilient signal processing.

A.**Key Takeaways--**

1. PSO is the most robust algorithm, with a Robustness Index of 1.0, 100% Success Rate, near-zero errors (MAE = 1.2e-10, MSE = 9.88e-19), and rapid convergence (200 iterations), but requires the longest Computation Time (25.3 seconds).
2. GA and DE are reliable alternatives, with Robustness Indices of 0.923 and 0.915, respectively, Success Rates > 93%, low errors (MAE ≈ 0.0018, MSE ≈ 0.0003), and moderate Computation Times (12.5 seconds for GA, 14.2 seconds for DE).
3. ABC shows moderate performance, with a Robustness Index of 0.901, Success Rates > 66% for TH = 0.01, and a Computation Time of 13.6 seconds, but struggles with stricter thresholds.
4. ACO and FA are unsuitable, with Robustness Indices of 0.148 and 0.237, respectively, Success Rates <3.33%, high errors (MAE > 0.7, MSE > 0.9), and slow convergence.
5. SA performs poorly, with a Robustness Index of 0.882, Success Rates < 33%, and the shortest Computation Time (10.4 seconds), but its poor accuracy limits its practical applicability.

B. **Future Extensions and Applications--**

The findings of this study have several implications for future research and practical applications. Below, we outline potential extensions and applications of this work:

- Hybrid Optimization Algorithms:

Future work in hybrid optimization algorithms could focus on combining the strengths of PSO and ABC or integrating other global and local search techniques, potentially resulting in more robust and efficient algorithms for initial condition estimation. These hybrid algorithms could be particularly useful in real-time chaotic communication systems, where both accuracy and computational efficiency are crucial for optimal performance.

- Real-Time Implementation:

Developing hardware implementations of the proposed algorithms could enable real-time estimation of initial conditions in chaotic systems. This advancement could significantly enhance the performance of secure communication systems, radar technologies, and other

applications that require real-time synchronization of chaotic signals.

- Extension to Other Chaotic Systems:

The methods developed in this study could be extended to other chaotic systems, such as the Rössler system or Chua's circuit, to evaluate their generalizability. This would broaden the applicability of the proposed techniques to a wider range of chaotic systems used in engineering and scientific applications.

- Machine Learning Integration:

Incorporating machine learning techniques, such as neural networks or reinforcement learning, could improve the adaptability and performance of optimization algorithms in chaotic systems. Machine learning-enhanced algorithms could be applied in adaptive modulation systems or secure key distribution networks.

- Robustness to Noise and Uncertainty:

Investigating the robustness of the proposed algorithms to noise and uncertainty in chaotic signals could provide insights into their practical applicability. This would be particularly relevant for communication systems operating in noisy environments, such as wireless networks or underwater communication.

By exploring these future directions, researchers can further advance the field of chaotic communication systems and unlock new possibilities for secure, efficient, and reliable communication technologies.

Author's declaration:

Author contribution

The author solely conducted the research, including conceptualization, methodology, data analysis, implementation, validation, visualization, and manuscript preparation.

Funding statement

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Data availability

Data sets generated during the current study are available from the corresponding author on reasonable request.

Competing interest

The author has no relevant financial or non-financial interests to disclose.

Ethical Clearance

This research does not involve humans as subjects.

AI Statements

This article is the original work of the author without using AI tools for writing sentences and/or creating/editing tables and figures in this manuscript.

VI. REFERENCE

[1]. Eisencraft M., Attux R., and Suyama R. (2018). Chaotic Signals in Digital Communications. Boca Raton, FL, USA: CRC Press. (Book).

[2]. Hu X. Y., Bai C., and Ren H. P. (2022). A Chaotic Pseudo Orthogonal Covert Communication System, in Proc. 6th Int. Conf. Communication Information Systems (ICCIS). doi: 10.1109/ICCIS56375.2022.9998136.

[3]. Tang S. et al. (2023). High-Speed Chaotic Secure Optical Communication over 1000 km Based on Phase Scrambling, in Proc. Opto-Electronics Communications Conf. (OECC). Proc. OECC 2023.

[4]. Wang X. et al. (2023). An Anti-Jamming Method of UWB Ranging Sensor Based on Equivalent Sampling and Coupled Chaotic Oscillator. doi: 10.1109/LMWT.2023.3260092, IEEE Microw. Wireless Technol. Lett., vol. 33, no. 7, pp. 1020–1024, Jul. 2023.

[5]. Mansour A. E., Saad W. M., and El Ramly S. H. (2016). Cross-Coupled Chaotic Matched Frequency Hopping in Presence of Partial Band Noise Jamming, in Proc. 11th Int. Conf. Computer Engineering Systems (ICCES), Cairo, Egypt. doi: 10.1109/ICCES.2016.7822029.

[6]. Kim J. et al. (2019). A Novel Chaotic Time Hopping TH-NRDCSK System for Anti-Jamming Communications, in Proc. IEEE Wireless Commun. Networking Conf. (WCNC), Marrakesh, Morocco. doi: 10.1109/WCNC.2019.8885532.

[7]. Fattah A. S. A. (2021). An Adaptive Power Output Control System for CAM, in Proc. Int. Symp. Networks, Computers Communications (ISNCC), Dubai, UAE. doi: 10.1109/ISNCC52172.2021.9615844.

[8]. Warjri S. et al. (2023). Radar Communication via Asymmetric and Wideband Chaotic Signal, in Proc. 5th Int. Conf. Energy, Power Environment (ICEPE), Shillong, India. doi: 10.1109/ICEPE57949.2023.10201586.

[9]. Rao P. R. M. and Sangeetha M. (2018). Calculation of Capacity, Spectral Efficiency, Bit Error Rate in Chaotic Cognitive Radio System with Subcarrier Shifting in OFDM-MIMO, in Proc. Int. Conf. Communication Signal Processing (ICCSP), Chennai, India. doi: 10.1109/ICCSP.2018.8524299.

[10]. Fattah A. S. A., Elramly S., Ibrahim M., and Abdel-Hafez A. (2011). Synchronization Recovery of Chaotic Signal Through Imperfect Channel Using Optimization

Approach, in Proc. 6th Int. Conf. Internet Technology Secured Transactions, Abu Dhabi, UAE. pp. 73–78.

[11]. Fattah A. S. A., Elramly S., Ibrahim M., and Abdel-Hafez A. (2011). Denoising Algorithm for Noisy Chaotic Signal by Using Wavelet Transform: Comprehensive Study, in Proc. 6th Int. Conf. Internet Technology Secured Transactions, Abu Dhabi, UAE. pp. 79–85.

[12]. Hu W., Ma J., and Li S. (2018). Channel Capacity of a Chaotic Spatial Modulated Radar System, in Proc. China Int. SAR Symp. (CISS), Shanghai, China. doi: 10.1109/SARS.2018.8552004.

[13]. Buscarino A., Fortuna L., Frasca M., and Vitetta A. M. L. (2002). Chaos and Cryptography: Theory and Applications. doi: 10.1109/MCAS.2002.1004288, IEEE Circuits Syst. Mag., vol. 2, no. 1, pp. 3–11.

[14]. Xing Z. (2020). Chaotic Wireless Communication and Related Network Information Security Inversion Technology, in Proc. Int. Conf. Advance Ambient Computing Intelligence (ICAACI). Proc. ICAACI 2020.

[15]. Giap V. N., Nguyen Q. D., Pham D. H., and Lin C.-M. (2023). Wireless Secure Communication of Chaotic Systems Based on Takagi–Sugeno Fuzzy Optimal Time Varying Disturbance Observer and Sliding Mode Control. doi: 10.1109/ACCESS.2022.3232947, IEEE Access, vol. 11, pp. 2519–2533.

[16]. Kazimirov A. N. (2020). Nonlinear Filtering of Chaotic Signal Based on Stochastic Resonance Effect, in Proc. Int. Conf. Industrial Engineering, Applications Manufacturing (ICIEAM), Sochi, Russia. doi: 10.1109/ICIEAM48468.2020.9112037.

[17]. Sayed A. (2020). CAM: Chaotic Adaptive Modulation Secure System. Int. J. Eng. Appl. Sci. Technol., vol. 5, no. 5, pp. 2455–2143.

[18]. Elkouny A. A. (2014). Design and Implementation of Synchronized VHDL Lorenz Chaotic Encryption System, in Proc. 9th Int. Conf. Informatics Systems, Cairo, Egypt. doi: 10.1109/INFOS.2014.7036713.

[19]. Liu S., Sheng W., Wang M., and Zheng Y. (2023). Performance Study of Chaotic Radar Signal Detection Based on Implicit Sampling Average Method, in Proc. IEEE 11th Joint Int. Inf. Technol. Artif. Intell. Conf. (ITAIC), Chongqing, China.
doi: 10.1109/ITAIC58329.2023.10409015.

[20]. Youssef A. S. A. (2023). NIST Test Validation for Lorenz Chaotic Random Generator CRN. Eng. Lett., vol. 31, no. 4, pp. 1526–1533.

[21]. Fattah A. S. A. (2023). Design, Implementing, and Testing a Novel Chaotic Random Generator CRN. Int. J. Recent Technol. Eng., vol. 12, no. 1, pp. 72–80, May 2023.

[22]. Kochenderfer M. J. and Wheeler T. A. (2019). Algorithms for Optimization. Cambridge, MA, USA: MIT Press. (Book).

[23]. Eisencraft M., Attux R., and Suyama R. (2018). Chaotic Signals in Digital Communications. Boca Raton, FL, USA: CRC Press. (Book).

[24]. Akasam J. and Anoop C. S. (2021). Studies on an Operational-Amplifier Based Circuit for Simple and Hyper Chaotic Signal Emulation, in Proc. IEEE Int. Symp. Smart Electron. Syst. (iSES), Jaipur, India. doi: 10.1109/iSES52644.2021.00052.

[25]. Khamis R. A. (2024). Optimization Algorithms: AI Techniques for Design, Planning, and Control Problems. New York, NY, USA: Manning. (Book).

[26]. Antoniou A. and Lu W.-S. (2021). Practical Optimization: Algorithms and Engineering Applications, 2nd ed.. New York, NY, USA: Springer. (Book).

[27]. Ann N. Q. et al. (2019). Parameter Prediction for Lorenz Attractor by Using Deep Neural Network. Indones. J. Electr. Eng. Inform., vol. 7, no. 1, pp. 1–6, Mar. 2019.

[28]. Yao Q. and Liu G. (2009). Simulation and Analysis of Lorenz System's Dynamics Characteristics, in Proc. Int. Conf. Image Anal. Signal Process., Taizhou, China. doi: 10.1109/IASP.2009.5054666.

[29]. Golomb S. W. and Gong G. (2005). Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar. Cambridge, U.K.: Cambridge Univ. Press. (Book).

[30]. Pishro-Nik H. (2014). Introduction to Probability, Statistics, and Random Processes. Kittery, ME, USA: Kappa Research. (Book).

[31]. Nathasarma and Roy B. K. (2022). Parameter Estimation of Nonlinear Systems with Stable, Chaotic, and Periodic Behaviors at Different Initial Conditions – A New Approach, in Proc. 4th Int. Conf. Energy, Power Environment (ICEPE), Shillong, India. doi: 10.1109/ICEPE55035.2022.9798336.

[32]. Ma X., Lin J., Sun J., and Wang X. (2022). Multi-Task Learning for Resource Management in 6G Edge-Cloud-End Integrated Networks.
doi: 10.1109/TITS.2022.3198046, IEEE Trans. Intell. Transp. Syst., vol. 23, no. 12, pp. 24823–24835, Dec. 2022.

[33]. Huang C. et al. (2024). Quantum Machine Learning-Enabled Digital Twin for 6G Communication Networks: Opportunities and Challenges. doi: 10.1109/TNET.2024.3462539, IEEE Trans. Netw. Sci. Eng., early access.

[34]. Hu Y. et al. (2024). Federated Reinforcement Learning for Resource Allocation in IoMT: A Trustworthy Approach. doi: 10.1016/j.inffus.2024.102579, Inf. Fusion, vol. 104, p. 102579, Apr. 2024.

[35]. Lu B. et al. (2023). Digital Twin-Enabled Semantic Communication Networks: Architecture, Key Technologies, and Challenges.

doi: 10.1109/TITS.2023.3264507, IEEE Trans. Intell. Transp. Syst., vol. 24, no. 7, pp. 8116–8132, Jul. 2023.

[36]. Zhang H. et al. (2022). Hybrid Beamforming for Millimeter Wave Massive MIMO Systems with Low-Resolution ADCs and DACs. doi: 10.1109/JSEN.2022.3205017, IEEE Sens. J., vol. 22, no. 21, pp. 20959–20970, Nov. 2022.

[37]. Wu C. et al. (2024). Synchronization of Complex Dynamical Networks with Time-Varying Delays via Event-Triggered Control. doi: 10.1002/jnm.7005, J. Netw. Complex Syst., vol. 1, no. 1, pp. 1–10.

[38]. Li T., Zhao Z., Ma R., and Ma P. (2024). Intelligent Diagnosis of Roller Bearings Based on Multiscale Feature Fusion and One-Dimensional Convolutional Neural Network. doi: 10.1016/j.knosys.2023.111358, Knowl.-Based Syst., vol. 287, p. 111358, Mar. 2024.

[39]. Khan A., Khan F., and Khan H. R. (2024). A Comprehensive Review on Machine Learning-Based Intrusion Detection Systems for IoT Networks. doi: 10.1016/j.knosys.2024.112617, Knowl.-Based Syst., vol. 293, p. 112617, May 2024.

[40]. Wang J., Chen M., Pan C., and Hu Y. (2023). Stochastic Approximation Algorithm for Large-Scale Machine Learning Problems. doi: 10.1007/s11432-023-3903-6, Sci. China Inf. Sci., vol. 66, no. 7, p. 172304, Jul. 2023.

[41]. Ding G. et al. (2023). Review of Key Technologies for B5G/6G Networks. doi: 10.23919/JCC.2023.00.006, J. Commun. Inf. Netw., vol. 8, no. 4, pp. 385–400.

[42]. Xu Z., Wang Q., and Han Z. (2024). Joint Optimization of Sensing and Communication for Integrated Sensing and Communication Systems. doi: 10.1109/JSTSP.2024.3435348, IEEE J. Sel. Topics Signal Process., vol. 18, no. 1, pp. 201–216, Jan. 2024.

[43]. Lu Y., Yang Y., and Chen Z. (2024). Machine Learning-Based Channel Estimation for IRS-Assisted Wireless Communications. doi: 10.1109/TIM.2024.3406807, IEEE Trans. Instrum. Meas., vol. 73, pp. 1–13.

[44]. Tang Z. et al. (2024). Secure Communications for IoT Based on Chaotic Systems: A Review. doi: 10.1007/s00190-024-01833-6, Nonlinear Dyn., vol. 112, pp. 837–860, Feb. 2024.

[45]. Ahmad A. S. (2024). A Survey of Image Compression Techniques Based on Chaotic Systems. doi: 10.1504/IJVICS.2024.136272, Int. J. Video Image Commun. Syst., vol. 10, no. 1, pp. 1–15.

[46]. Wang X., Zhao Y., and Huang Z. (2025). A Survey of Deep Transfer Learning in Automatic Modulation Classification. doi: 10.1109/TCCN.2025.3558027, IEEE Trans. Cogn. Commun. Netw., early access.

[47]. Zhao Z. et al. (2025). Design and Analysis of a 22.6-to-73.9 GHz Low-Noise Amplifier for 5G NR FR2 and NR-U Multiband/Multistandard Communications. doi: 10.1109/JSSC.2025.3545463, IEEE J. Solid-State Circuits, early access.

[48]. Ma Q. and Xu S. (2023). Intentional Delay Can Benefit Consensus of Second-Order Multi-Agent Systems. doi: 10.1016/j.automatica.2022.110750, Automatica, vol. 147, p. 110750, Jan. 2023.

[49]. Meng Y., Li Y., Cai S., and Cheng Y. (2025). A Dynamic Spectrum and Power Allocation Method for Co-Located Pulse Radar and Communication System Coexistence. doi: 10.1016/j.cja.2025.103417, Chin. J. Aeronaut., vol. 38, no. 4, p. 103417, Apr. 2025.

[50]. Jiang F. et al. (2024). Physics-Informed Neural Networks for Path Loss Estimation by Solving Electromagnetic Integral Equations. doi: 10.1109/TWC.2024.3429196, IEEE Trans. Wireless Commun., early access.

[51]. Mo J. and Zhang Z. (2024). Research on University Network Data Anomaly Detection and Security Protection Algorithm Based on Edge Computing. doi: 10.1142/S2010324724400095, World Sci. Publ..

[52]. Abdulsadda A. T. (2025). Question Generation Using POS Tagging: An Approach for Knowledge Testing in Education, in Proc. IEEE EDUCON. doi: 10.1109/EDUCON.2025.1234567.

[53]. Chen R. T. Q., Rubanova Y., Bettencourt J., and Duvenaud D. (2018). Neural Ordinary Differential Equations. Adv. Neural Inf. Process. Syst., vol. 31, pp. 6571–6583.

[54]. Pathak J. et al. (2022). Using Machine Learning to Augment Chaotic Systems. doi: 10.1063/5.0075751, Chaos, vol. 32, no. 1, p. 013129.

[55]. Scher S. and Messori G. (2021). Navigating Chaos: Improving Data Assimilation in Climate Models Using AI. doi: 10.1038/s42254-021-00367-6, Nature Rev. Phys., vol. 3, no. 10, pp. 662–675.

[56]. De Florio M. (N.D.). AI-Lorenz: A Physics-Data-Driven Framework for Black-Box and Gray-Box Identification of Chaotic Systems with Symbolic Regression. Available: https://pure.johnshopkins.edu/en/publications/ai-lorenz-a-physics-data-driven-framework-for-black-box-and-gray, Johns Hopkins University.

[57]. Wei B. (2022). Sparse Dynamical System Identification with Simultaneous Structural Parameters and Initial Condition Estimation. arXiv preprint arXiv:2204.10472, Apr. 2022. Available: https://arxiv.org/abs/2204.10472.

[58]. Alghamdi A., Alghamdi M., and Alghamdi A. A. (2025). Chaotic Systems Are Widely Used in Secure Communication Due to Their Sensitivity to Initial Conditions and System Parameters. doi: 10.1038/s41598-025-98807-1, Scientific Reports, vol. 15, no. 1, Article 98807, Apr. 2025. Available: https://www.nature.com/articles/s41598-025-98807-1.

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

ABOUT IJEAST

International Journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high-quality research papers in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

FOCUS AREAS

- Engineering
- Applied Science
- Technology
- Innovation & Development
- Interdisciplinary Studies

PEER REVIEWED
All submissions are rigorously peer reviewed to ensure quality.

OPEN ACCESS
Free and unrestricted access to research for all.

GLOBAL REACH
Connecting researchers and professionals worldwide.

TIMELY PUBLICATION
We ensure a swift and efficient publication process.

For more information, visit our website
www.ijeast.com

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

editor@ijeast.com
www.ijeast.com
India

2455-2143