Evolving Systems (2025) 16:100
https://doi.org/10.1007/s12530-025-09730-9

ORIGINAL PAPER

Parameter estimation based IIR system identification using improved arithmetic optimization algorithm

Deepak Goyal[1,2] · Puneet Khanna[1] · Sandeep Singh[2]

Received: 1 June 2024 / Accepted: 25 July 2025

Abstract
System identification plays a crucial role in various fields like control systems, communication networks, biomedical signal processing, and many more. Among the different system identification techniques, infinite impulse response (IIR) system offer flexibility in capturing complex dynamics. However, accurately estimating the parameters of an IIR model can be challenging due to its inherent nonlinearity and potential instability. This paper presents a novel method for IIR system identification utilizing an enhanced arithmetic optimization algorithm (IAOA) in order to overcome this problem. The IAOA leverage the strengths of evolutionary computation and numerical optimization techniques to improve the precision and efficacy of the parameter estimation process. By combining concepts from genetic algorithms, particle swarm optimization, and simulated annealing, the proposed algorithm aims to overcome the limitations of traditional optimization methods and provide a more robust and effective solution. The performance of the IAOA is evaluated through comprehensive comparisons and simulations with existing optimization methods on various benchmark IIR system identification problems. The results demonstrate its superiority in terms of parameter estimation and convergence.

Keywords Infinite impulse response system · System identification · Improved arithmetic optimization · Whale optimization algorithm · Mean square error

1 Introduction

One of the digital system type utilized frequently in various fields is the infinite impulse response (IIR) system. IIR system identification refers to the process of estimating the parameters of an IIR model based on observed input–output data. An IIR model is a type of linear time-invariant system characterized by recursive feedback loops, which allows it to capture complex dynamics with fewer parameters compared to finite impulse response (FIR) systems (Mitra and Kuo 2006; Sayed 2003). The identification of IIR systems is essential in various fields, including target tracking (Zarai and Cherif 2021), signal processing (Pai et al. 2013; Le et al. 2021), control systems (Zhou et al. 2014), communication system (Li et al. 2021), signal denoising, navigation and positioning (Zhang et al. 2021), data transmission (Czapiewska et al. 2020) and audio processing. By accurately evaluating the parameters of an IIR model, one can understand the underlying system behaviour, design appropriate control strategies, optimize system performance, and perform system analysis.

The IIR system identification problem can be defined as utilizing an adaptive IIR filter to characterize an unknown system (Dai et al. 2009). Its goal is to get close to the unknown system coefficients using the adaptive filter coefficients. In this case, the adaptive system and the unknown system both receive the same input signal, and their respective outputs are noted. The IIR system identification issue is an optimization problem that optimizes the error between the output of the adaptive system and the output of the unknown system (Janjanam et al. 2024).

Numerous real-world industries are plagued with optimization issues. As science and computing develop, a growing number of issues become more challenging for large-scale optimization. However, solving traditional mathematical programming algorithms is challenging. In order to solve problems involving global optimization, researchers have put forth a wide variety of meta-heuristic algorithms that

✉ Sandeep Singh
er.sandeep85@gmail.com

1 IFTM university, Moradabad, Uttar Pradesh, India

2 Maharaja Surajmal Institute of Technology, New Delhi, India

Published online: 07 August 2025

simulate the behaviour of natural objects or creatures. Metaheuristic algorithms are a fantastic tool for dealing with non-linear optimization issues because of the simplicity of the underlying principle, insensitivity to beginning values, and simplicity of implementation. Additionally, it has been demonstrated that the meta-heuristic approach is more confident in solution accuracy and application because it is not dependent on the gradient of the objective function. The various metaheuristic algorithms applied for the IIR system identification problem are particle swarm optimization (PSO) algorithm (Zou et al. 2018), flower pollination algorithm (FPA) (Singh et al. 2016), cat swarm optimization (CSO) algorithm (Panda et al. 2011), modified whale optimization algorithm (MWOA) (Luo et al. 2020), bat algorithm (BAT) (Saha et al. 2013), teacher learner based optimization (TLBO) (Singh et al. 2019) algorithm, gravitational search algorithm (GSA)(Rashedi et al. 2009; Jiang et al. 2015) and artificial intelligent optimization (Mohammadi et al. 2018).

Zuo et al. (2018) proposed the modified version of PSO for processing the IIR system identification problem . Singh et al. (2019) applied a human based optimization called TLBO for solving the problem of unknown IIR system identification. The simulated results using TLBO shows the effectiveness of the applied algorithm. Later, Luo et al. (2020) proposed an improved WOA with a rank-based mutation parameter called RWOA, to process the IIR system identification problem. Singh et al. (2016) exploits the pollination process in flowers for IIR system identification. Further, IIR system was identified using CSO which is based on the behaviour of cats (Panda et al. 2011). According to the widely recognized no-free-lunch theorem (NFL) (Wolpert and Macready 1997), no optimization method can outperform all other methods for any optimization problem. It is conceivable for an algorithm to perform well on one test example but poorly on other because the IIR model contains different cases. As a result, this has inspired academicians and researchers to look into the effectiveness of new algorithms to address the IIR model identification challenge as well as problems in other fields.

The key contributions of this paper lie in two aspects. Firstly, a novel representation scheme is introduced to encode and decode IIR model parameters, facilitating the search space's exploration and exploitation. This encoding scheme optimally balances the exploration of a wide range of potential solutions and the exploitation of promising regions, thereby enhancing the convergence speed and global search capabilities of the algorithm. Secondly, a set of improved arithmetic operators is devised, inspired by the principles of numerical stability and precision. These operators aim to mitigate the effects of numerical difference and ill-conditioning that commonly arise during the optimization process. By employing these operators, the applied algorithm enhances the reliability of the parameter estimates and promotes the stability of the identified IIR system.

A recently employed population-based approach is the arithmetic optimization algorithm (AOA) (Abualigah et al. 2021). Thus far, the AOA has been used for a variety of real-world optimization problems, such as the identification of models (Xu et al. 2021), the tracking of maximum power points (Mirza et al. 2021), the segmentation of images (Abualigah et al. 2021), and the detection of structural damage (Khatir et al. 2021). The typical AOA, however, has been discovered to have poor exploration and to be prone to get trapped in local optima (Xu et al. 2021; Abualigah et al. 2021). In order to overcome the issues of AOA, this paper applied improved arithmetic optimization algorithm (IAOA). Dynamic probability coefficient and triangular mutation techniques are employed to enhance AOA's convergence speed, and dynamic inertia weights are used to the capacity of AOA to exit the local optimal region. The obtained outcomes indicate that the stability, convergence speed and accuracy of the suggested algorithm are greatly enhanced, and IAOA performs exceptionally well in the optimization of the parameters of IIR system.

Comparing the proposed IAOA to the existing AOA, the following are its significant contributions:

(1) To address its imperfections, the basic AOA's exploration and exploitation capabilities are improved.
(2) Since there are fewer algorithm-specific parameters required, the suggested IAOA is easier to construct than the conventional AOA.
(3) The use of triangular mutation strategy in IAOA provide the global optimal solution in terms of objective function value and parameter estimation avoiding local solution.

Compared to the basic AOA and other current methods, these enhancements make IAOA a more reliable and effective algorithm for IIR system identification, yielding better performance. Also, this is the first attempt that IAOA is applied to IIR system identification problem. Two benchmark examples of unknown IIR system are used to show how well the proposed IAOA performs. The outcomes obtained by the AOA and IAOA are contrasted with those of other optimization techniques that have been documented in the literature.

The remainder of the paper is laid out as follows. The mathematical description of the IIR system identification problem is introduced in Sect. 2. Section 3 is a brief explanation of the AOA and improved AOA. Simulation results for two Examples are presented in Sect. 4. Finally, in Sect. 5, a conclusion is formed.

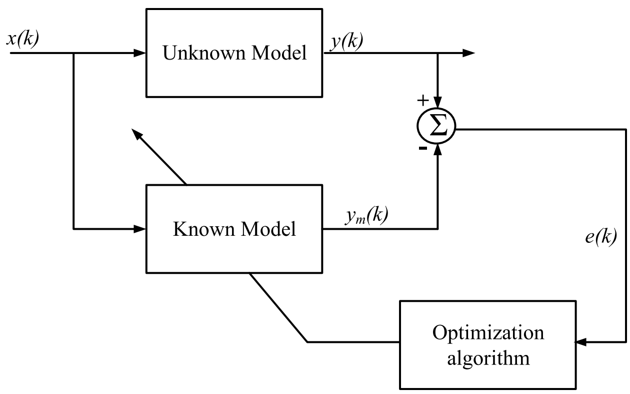


Fig. 1 Block diagram of system identification problem using improved AOA

2 Promlem statement

Since many signal processing issues may be treated as system identification problems, the adaptive IIR system has been extensively utilized in system identification. The adaptive algorithm's main goal is to identify optimal system coefficients so that the adaptive system's output closely resembles that of an unknown system. Figure 1 displays the block diagram for an adaptive IIR system identification. The relationship between the input and output of the IIR system can be explained as follows (Luo et al. 2020; Singh et al. 2019):

$$y(k) + \sum_{j=1}^{L} d_j y(k-j) = \sum_{i=0}^{M} b_i x(k-i) \quad (1)$$

where $(L > M)$ denotes the system's order and $x(k)$ and $y(k)$ denote the system's input and output, respectively. Let $d_0 = 1$. The IIR system's transfer function is therefore described as follows:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{a_0 + a_1 z^{-1} + \ldots + a_M z^{-M}}{1 + d_1 z^{-1} + \ldots + d_L z^{-L}} \quad (2)$$

The unknown system's transfer function, $H_p(z)$, and the IIR system's transfer function, $H_m(z)$, are both used in the design procedure. It is evident from Fig. 1 that the output of the IIR system is $y_m(k)$, whereas the output of the unidentified system is $y(k)$. The error signal is represented by the equation $e(k) = y(k) - y_m(k)$. Therefore, the main objective of identification is to design a minimization problem, where the cost function $D(w)$ can be written as follows:

$$D(w) = \frac{1}{K} \sum_{n=1}^{K} e^2(k) \quad (3)$$

where K is the length of input signal used in the objective function's computation. The IIR model's coefficient vectors are produced by $D(w)$ which is equal to mean square error (MSE). The algorithm's goal is to reduce the MSE by modifying the coefficient vector of the transfer function $H_p(z)$.

3 Basic arithmetic optimization algorithm (AOA)

In 2021, Abualigah et al. developed the arithmetic optimization (AOA) which is a population-based meta-heuristic algorithm that utilizes the basic mathematical operations (addition, subtraction, multiplication, and division) while optimization (Habib and Cherri 1998). The distribution features of the basic arithmetic operations of multiplication (M), division (D), addition (A), and subtraction (S) are thus simulated by AOA (Abualigah et al. 2021). Like other meta-heuristic algorithms, AOA's search procedure is broken down into two main stages: exploration and exploitation. The exploration phase uses multiplication and division operators to update the search agents' position (potential solutions), whereas the exploitation phase uses addition and subtraction operations. Figure 2 depicts the order of arithmetic operations and their superiority from outside to inside. The AOA is mathematically described in the subsequent subsections.

3.1 Initialization phase

As demonstrated in Eq. (4), AOA starts the search process with a population of randomly distributed solutions (X), just like other population-based meta-heuristic algorithms. The

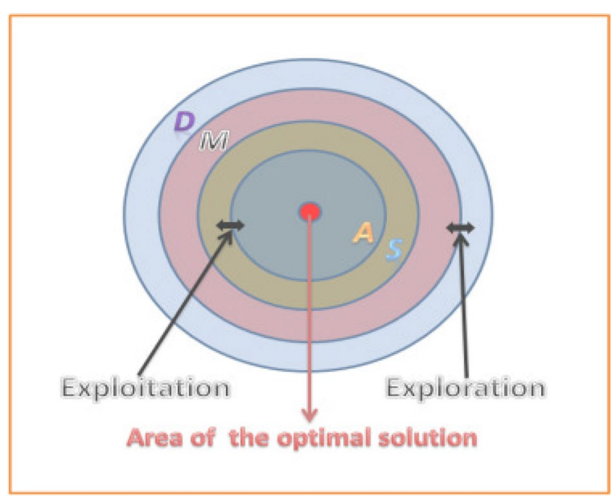


Fig. 2 The hierarchy and importance of arithmetic operations (from the outside to the inside)

solution is updated whenever the better solution is obtained for the current iteration.

$$X = \begin{bmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1n} \\ \dots & \dots & \dots & \dots & \dots \\ x_{i1} & \dots & x_{ij} & \dots & x_{in} \\ \dots & \dots & \dots & \dots & \dots \\ x_{N1} & \dots & x_{Nj} & \dots & x_{Nn} \end{bmatrix} \quad (4)$$

where n represents the number of design variables in the problem, N the number of candidate solutions in the population, and $x_{i,j}$ the j^{th} design variable of the i^{th} candidate solution in the initial population.

The AOA switches between the exploitation and exploration stages of the search process using a dynamic function known as math optimizer accelerated (MOA). The following equation gives the definition of the MOA function:

$$MOA(C_{iter}) = Min + C_{iter} \times (\frac{Max - Min}{M_{iter}}) \quad (5)$$

where $MOA(C_{iter})$ is the value of MOA at current iteration; M_{iter} denotes the maximum iterations; C_{iter} is the present iteration number and ranges between 1 and M_{iter}; and Min and Max are the minimum and maximum values of the MOA, respectively. According to Abualigah et. al (2021) recommendation , Min and Max are set in this work to 0.2 and 0.9, respectively.

The multiplication (M) and division (D) operators are used to explore the search space when $MOA(C_{iter}) < c_1$, but the addition (A) and subtraction (S) operators are used to exploit the search space's promising regions when $MOA(C_{iter}) > c_1$. The pseudorandom number c_1 has a uniform distribution and ranges from 0 to 1. Equation (5) shows that MOA increases linearly from $Min + (Max - Min)/M_{iter}$ at the first iteration (i.e., $C_{iter} = 1$) to Max at the last iteration (i.e., $C_{iter} = M_{iter}$). $MOA(C_{iter})$ is able to seamlessly transition between the exploration and exploitation phases as a result.

3.2 Exploration phase

As indicated, in the exploration phase of AOA, the arithmetic operations of multiplication (M) and division (D) are used to guide the investigation of the search area. The AOA's exploration phase is carried out in accordance with the next position update rule (Abualigah et al. 2021):

$$X_{i,j}(C_{iter} + 1) = \begin{cases} best(X_j) \div (NOP + \epsilon) \times ((uv_j - lv_j) \times \beta + lv_j), & c_2 < 0.5 \\ best(X_j) \times NOP \times ((uv_j - lv_j) \times \beta + lv_j), & \text{otherwise} \end{cases} \quad (6)$$

where $X_{i,j}(C_{iter} + 1)$ represents the j^{th} position of the i^{th} solution in next iteration. The best solution thus far is $best(X_i)$. ϵ is a constant added in the denominator to ensure numerical stability, β is a control parameter for modifying the search procedure, lv and uv are the lower and upper value of the j^{th} position. Moreover, NOP stands for a function known as numerical optimizer probability, and is defined as follows:

$$NOP(C_{iter}) = 1 - \frac{C_{iter}^{1/\alpha}}{M_{iter}^{1/\alpha}} \quad (7)$$

where α is a sensitive parameter representing the accuracy of exploitation over the iterations and is set to 5 (Abualigah et al. 2021), and $NOP(C_{iter})$ specifies the value of the coefficient NOP at the current iteration number.

3.3 Exploitation phase

The subtraction (S) and addition (A) operators provide small step sizes that result in a very dense population of candidate solutions, as opposed to the large step sizes produced by division (D) and multiplication (M), which produce a widely spread population of candidate solutions. The exploitation phase of the AOA is based on the position updating rule that follows (Abualigah et al. 2021):

$$X_{i,j}(C_{iter} + 1) = \begin{cases} best(X_j) - (NOP + \epsilon) \times ((uv_j - lv_j) \times \beta + lv_j), & c_3 < 0.5 \\ best(X_j) + NOP \times ((uv_j - lv_j) \times \beta + lv_j), & otherwise \end{cases} \quad (8)$$

Algorithm 1 Improved arithmetic optimization algorithm

```
Initialize N, M_iter, C_iter = 0, α, β, ϵ
Initialize the position of the individuals x_i (i = 1, 2, ..., N).
While (C_iter < M_iter)
Evaluate w(C_iter) using Eq. (9).
Evaluate MOA and NOP using Eq. (5) and Eq. (7) respectively.
Determine the optimal solution by finding the best fitness value.
for i = 1, 2, ...N do
    for j = 1, 2, ...n
        Create the random numbers (c1, c2, c3) between [0, 1]
            if c1 > MOA
                Evaluate X(C_iter + 1) using Eq. (10).
        else
                Evaluate X(C_iter + 1) using Eq. (11).
        end if
          Calculate the value of p using Eq. (12).
          if p > c
                Evaluate X(C_iter + 1) using Eq. (13).
         end if
            end for
         end for
C_iter = C_iter + 1
end While
Provide the best solution X
```

3.4 Proposed improved arithmetic optimization algorithm (IAOA)

Shi and Eberhart were the first to propose inertia weights; larger inertia weights are advantageous for global exploration while smaller inertia weights are advantageous for local exploitation (Shi and Eberhart 1998). Therefore, to increase the search efficiency improved version of AOA (IAOA) is proposed in this paper. IAOA incorporates dynamic inertia weights and mutation probability coefficients (Fang et al. 2022). By varying the inertia weights dynamically, the algorithm makes the balance between exploration and exploitation more effectively. Also, by dynamically adjusting the mutation probability, the IAOA maintains diversity in the population and prevent premature convergence, which significantly improve its convergence speed compared to the basic AOA (Fang et al. 2022). This means it can find optimal solutions faster, which is crucial for real-time applications like IIR system identification. These weights decrease nonlinearly and exponentially with the number of iterations. The dynamic inertia weights are given in Eq. (9):

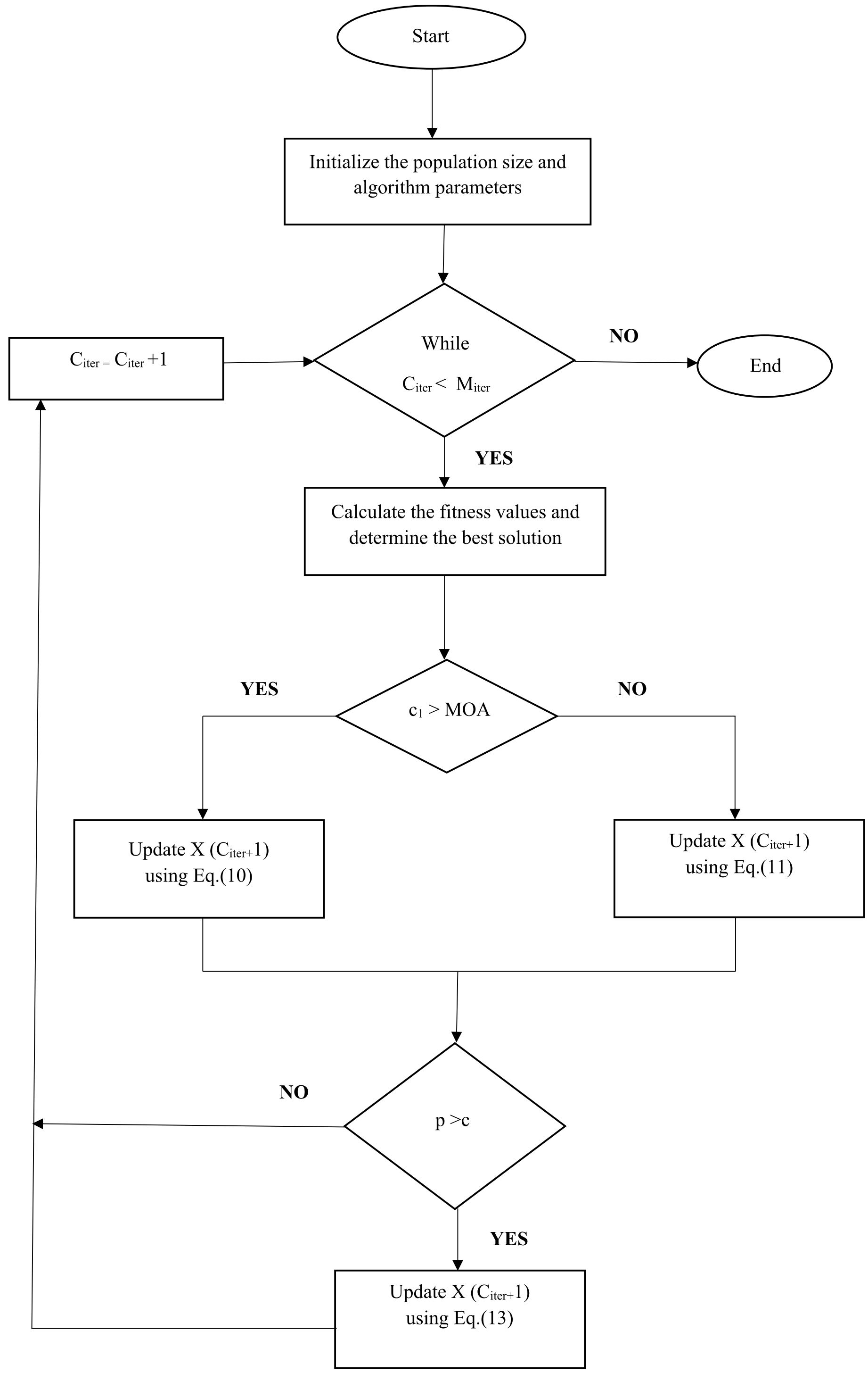
$$w(C_{iter}) = rand \times w_{init}\left(\frac{w_{init}}{w_{final}}\right)^{\frac{1}{1+\frac{C_{iter}}{M_{iter}}}} \tag{9}$$

where w_{init} and w_{final} are the maximum and minimum values of inertia weights, *rand* is random value lies around 1. When the total of dynamic inertia weights is incorporated Eqs. (6) and (8) are updated as Eqs. (10) and (11)

$$X_{i,j}(C_{iter}+1) = \begin{cases} w(C_{iter}) \times best(X_j) \div (NOP + \varepsilon) \times ((uv_j - lv_j) \times \beta + lv_j), & c_2 < 0.5 \\ w(C_{iter}) \times best(X_j) \times NOP \times ((uv_j - lv_j) \times \beta + lv_j), & \text{otherwise} \end{cases} \tag{10}$$

$$X_{i,j}(C_{iter}+1) = \begin{cases} w(C_{iter}) \star best(X_j) - (NOP + \varepsilon) \times ((uv_j - lv_j) \times \beta + lv_j), & c_3 < 0.5 \\ w(C_{iter}) \star best(X_j) + NOP \times ((uv_j - lv_j) \times \beta + lv_j), & otherwise \end{cases} \tag{11}$$

Fig. 3 Flow chart of IAOA



3.5 Dynamic mutation probability coefficient and triangular mutation technique

In order to give individuals a chance to enter in different search spaces, this paper's "mutation" operation employs a mutation probability coefficient that is dynamic and increases as the number of iteration increases.. This effectively broadens the search space and improves the capacity of the algorithm to depart from the local optimum solution. The coefficient of dynamic mutation probability, p is given by Eq. (12).

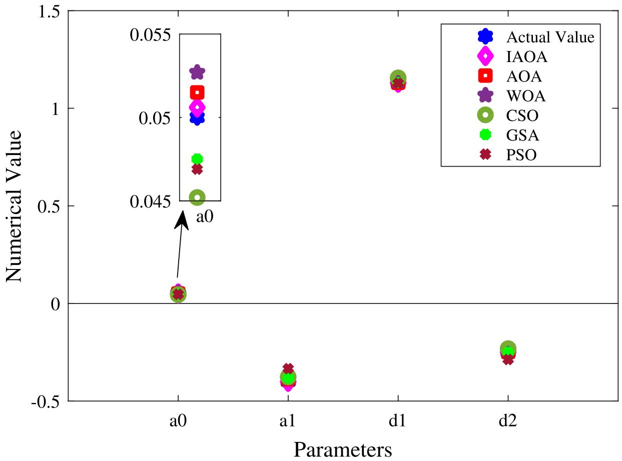


Fig. 4 Comparison of evaluated parameters for Example 1 Case 1

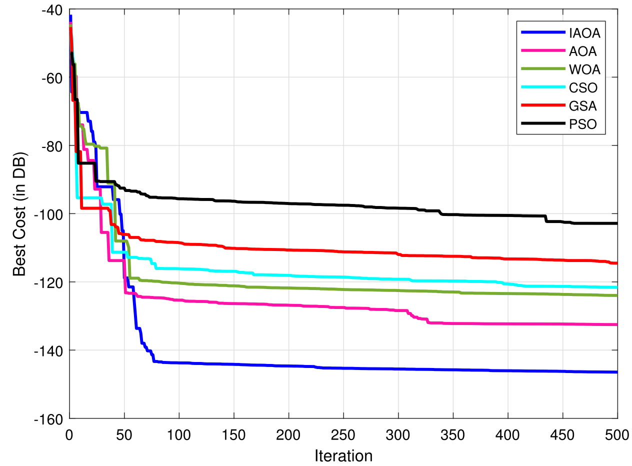


Fig. 5 Comparison of best optimization runs for Example 1 Case 1

$$p = 0.2 + 0.5 \times \frac{C_{iter}}{M_{iter}} \tag{12}$$

The triangle mutation technique (Fan and Lampinen 2003) fully utilizes each individual's information in the population, allowing each individual's knowledge to cross over with each other, increasing the population's variety and preventing the algorithm from reaching a local optimum during the search phase. Equation (13) displays one of the triangular mutation formulas:

$$X(C_{iter}) = (X_{r1} + X_{r2} + X_{r3})/3 + (C_{iter2} - C_{iter1}) \star (X_{r1} - X_{r2}) + (C_{iter3} - C_{iter2}) \star (X_{r2} - X_{r3}) + (C_{iter1} - C_{iter3}) \star (X_{r3} - X_{r1}) \tag{13}$$

The three randomly chosen individuals are designated as X_{r1}, X_{r2}, and X_{r3}, and the weights of the perturbed part are indicated by $(C_{iter2} - C_{iter1})$, $(C_{iter3} - C_{iter2})$, and $(C_{iter1} - C_{iter3})$. The triangle mutation approach is comparable to a genetic algorithm's cross mutation, which combines the information of random people. This tactic helps to increase the algorithm's ability to exit local minima by preventing individuals from updating themselves solely in the vicinity of a single local optimal location. The flow chart of IAOA is shown in Fig. 3. The pseudocode of IAOA is given in Algorithm 1.

4 Simulation analysis

An comprehensive experiment is carried out on two different types of benchmark IIR plants, with orders two, and three, which are selected from Janjanam et al. (2024), Luo et al. (2020) and Singh et al. (2019, 2016) in order to confirm the effectiveness of the IAOA for identifying unknown IIR systems. The results obtained using IAOA was compared with that of the five existing algorithms

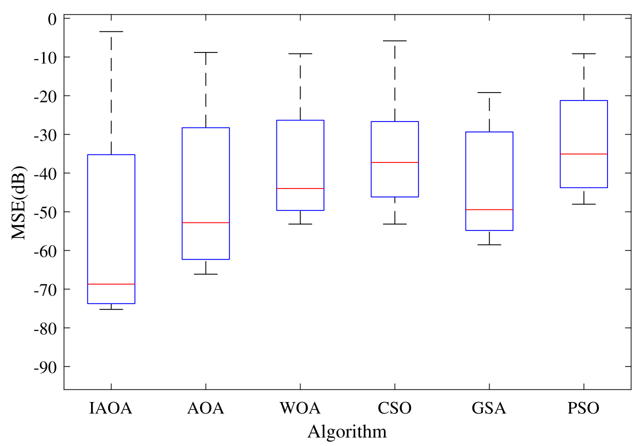


Fig. 6 Boxplot for Example 1 Case 1

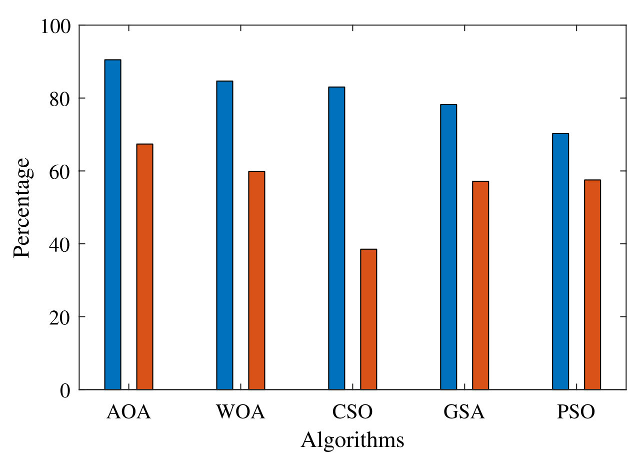


Fig. 7 Bar plot of IAOA with other existing algorithms for Example 1 Case 1

named AOA, WOA, CSO, GSA, and PSO. The reasons behind selecting these algorithms are based on certain criteria. These criteria include popularity and acceptance, diversity, performance in similar domains and availability and ease of implementation. These algorithms (AOA, WOA, CSO, GSA, PSO) fit for the above mentioned criteria and so as selected for comparison of results obtained using IAOA. The MSE, which is described in Sect. 2, is considered as the performance metric. The simulation was carried out on MATLAB 2019a.

Example 1 For the first experiment, a second-order system is considered whose system function is given by Janjanam et al. (2024), Luo et al. (2020) and Singh et al. (2019):

$$H_p(z) = \frac{0.05 - 0.4z^{-1}}{1 - 1.314z^{-1} + 0.25z^{-2}} \tag{14}$$

This second-order system is modelled by taking two cases.

Case 1 The system function of the second-order system modelled using second-order (same-order) system is given by

$$H_s(z) = \frac{a_0 + a_1 z^{-1}}{1 - d_1 z^{-1} - d_2 z^{-2}} \tag{15}$$

a_0, a_1, d_1 and d_2 are the numerator and denominator coefficients to be estimated. Table 1 shows the estimated parameters obtained by the six algorithms, and it demonstrates that AOA and IAOA outperformed the other four algorithms in terms of coefficient estimation. This shows that the IAOA has a good ability to gather precise estimated parameter values. The numerical values of the parameter are shown in Fig 4. The statistical findings in terms of MSE are shown in Table 2. The best evaluated values of MSE using IAOA, AOA, WOA, CSO, GSA, and PSO are 4.7677×10^{-08}, 2.3667×10^{-07}, 6.3297×10^{-07}, 8.1722×10^{-07}, 1.8818×10^{-06} and 7.2069×10^{-06} respectively. The convergence curve (MSE in dB) for the Example 1 Case 1 is demonstrated in Fig. 5 which reveals that IAOA has faster convergence than the other compared algorithms. Moreover, Box plots provide a concise summary of the data distribution. They offer a visual representation of the key statistical measures and characteristics without overwhelming the viewer with detailed data points. The box plots for the employed algorithms in Example 1 are depicted in Fig. 6. The MSE comparison of IAOA with other employed algorithms is also shown in terms of Bar plot (Fig. 7). It is evident from the results obtained that IAOA achieves lower (best) MSE values, indicating its effectiveness in minimizing the error and converging to the better solutions. The computational complexity of the IAOA is compared with that of the existing algorithms and it is found that IAOA is computationally same (approximately) to that of AOA, WOA and PSO but more fast in comparison to CSO and GSA. This is illustrated in terms of runtime listed in Table 9.

Case 2 The second-order system can be modeled by a reduced-order IIR system. The transfer function of the reduced-order system is given by

$$H_r(z) = \frac{a_0}{1 - d_1 z^{-1}} \tag{16}$$

The simulated results obtained using IAOA and other existing algorithms in terms of MSE and MSE in dB are listed in Table 3. It is apparent from Table 3 that the IAOA outperformed the other employed (AOA, WOA, CSO, GSA and PSO) algorithms in identifying the unknown IIR system offering the best MSE values. The best MSE value obtained using IAOA, AOA, WOA, CSO, GSA and PSO are 4.0802×10^{-04}, 2.2770×10^{-03}, 3.1200×10^{-03}, 4.4932×10^{-03}, 8.1680×10^{-03}, and 1.4087×10^{-02}

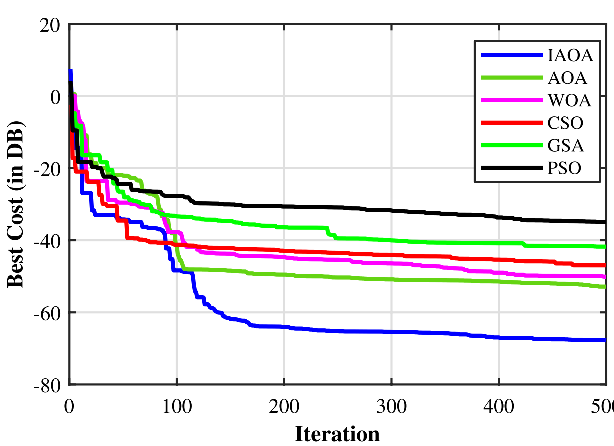


Fig. 8 Comparison of best optimization runs for Example 1 Case 2

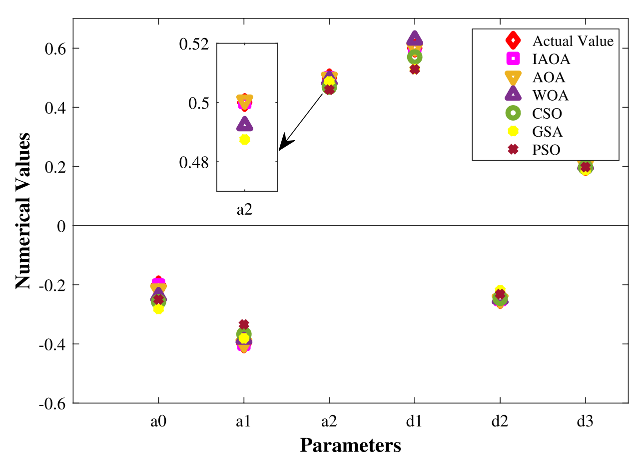


Fig. 9 Comparison of evaluated parameters for Example 2 Case 1

Table 1 Parameters of second-order IIR system evaluated using IAOA, AOA, WOA, GSA and PSO algorithms

Parameters	Actual value	IAOA	AOA	WOA	CSO	GSA	PSO
a_0	0.05	0.0506	0.0515	0.0527	0.0452	0.0475	0.0469
a_1	−0.4	−0.4011	−0.3882	−0.3824	−0.3762	−0.3811	−0.3340
d_1	1.1314	1.1314	1.1303	1.1390	1.1557	1.1316	1.1289
d_2	−0.25	−0.2501	−0.2492	−0.2468	−0.2315	−0.2506	−0.2873

Table 2 Result comparison in terms of MSE values evaluated using IAOA, AOA, WOA, CSO, GSA and PSO for Example 1 in case of same-order system

Algorithm	MSE				MSE (in dB)		
	Best	Worst	Average	SD	Best	Worst	Average
IAOA	$\mathbf{4.7677 \times 10^{-08}}$	$\mathbf{5.6125 \times 10^{-06}}$	$\mathbf{2.3540 \times 10^{-07}}$	$\mathbf{4.9625 \times 10^{-12}}$	**−146.4339**	**−105.016**	**−132.517**
AOA	2.3667×10^{-07}	2.9050×10^{-05}	2.7400×10^{-06}	4.3256×10^{-11}	−132.517	−90.7370	−111.2450
WOA	6.3297×10^{-07}	1.0441×10^{-04}	7.7718×10^{-06}	5.8016×10^{-05}	−123.9724	−79.6250	−102.4160
CSO	8.1722×10^{-07}	1.7080×10^{-05}	2.8554×10^{-06}	1.3579×10^{-05}	−121.573	−95.3500	−110.8867
GSA	1.8818×10^{-06}	8.1157×10^{-05}	1.0131×10^{-05}	1.3579×10^{-04}	−114.5087	−81.8135	−99.8867
PSO	7.2069×10^{-06}	4.7090×10^{-04}	4.9837×10^{-05}	3.6521×10^{-04}	−102.8450	−66.5414	−86.0490

Bold values signifies the minimum value of MSE obtained by the algorithm among existing algorithms

Table 3 Result comparison in terms of MSE values evaluated using IAOA, AOA, WOA, CSO, GSA and PSO for Example 1 in case of reduced-order system

Algorithm	MSE				MSE (in dB)		
	Best	Worst	Average	SD	Best	Worst	Average
IAOA	$\mathbf{4.0802 \times 10^{-04}}$	$\mathbf{2.0325 \times 10^{-02}}$	$\mathbf{2.6790 \times 10^{-03}}$	$\mathbf{3.7821 \times 10^{-04}}$	**−67.7864**	**−33.8394**	**−51.4405**
AOA	2.2770×10^{-03}	2.0900×10^{-02}	1.3418×10^{-02}	3.4060×10^{-03}	−52.8527	−33.5971	−37.4579
WOA	3.1200×10^{-03}	6.1205×10^{-01}	3.2366×10^{-02}	4.3256×10^{-03}	−50.1169	−24.2643	−29.7982
CSO	4.4926×10^{-03}	9.7650×10^{-02}	5.1235×10^{-02}	1.3509×10^{-03}	−46.9500	−20.2066	−25.8087
GSA	8.1680×10^{-03}	2.3917×10^{-02}	1.5781×10^{-02}	2.9079×10^{-03}	−41.7577	−32.4320	−36.0820
PSO	1.4087×10^{-02}	1.7964×10^{-02}	1.6220×10^{-02}	2.2091×10^{-03}	−37.0236	−34.9119	−35.7990

Bold values signifies the minimum value of MSE obtained by the algorithm among existing algorithms

respectively. The superiority of IAOA is also demonstrated in terms of convergence curves for the second-order system modelled using reduced-order (first-order) system. The IAOA converged much more quickly and had the lowest MSE values when compared to the other five algorithms, as shown in Fig. 8.

Example 2 The second experiment consider the third order plant whose transfer function is given by Eq. (17) (Janjanam et al. 2024; Luo et al. 2020; Singh et al. 2019).

$$H_p(z) = \frac{-0.2 - 0.4z^{-1} + 0.5z^{-2}}{1 - 0.6z^{-1} + 0.25z^{-2} - 0.2z^{-2}} \tag{17}$$

Case 1 It is possible to define this third-order plant modelled as a third-order (same-order) IIR system whose transfer function is given in Eq. (18).

Table 4 Parameters of third-order IIR system evaluated using IAOA, AOA, WOA, CSO, GSA and PSO algorithms

Parameters	Actual value	IAOA	AOA	WOA	CSO	GSA	PSO
a_0	−0.20	−0.2005	−0.2156	−0.2385	−0.2567	−0.2815	- 0.2497
a_1	−0.40	−0.4011	−0.3982	−0.3824	−0.3671	−0.3811	−0.3341
a_2	0.50	0.5000	0.5007	0.4923	0.4687	0.4875	0.4598
d_1	0.60	0.6014	0.6033	0.6290	0.5703	0.5296	0.5286
d_2	−0.25	−0.2501	−0.2492	−0.2468	−0.2456	−0.2186	−0.2314
d_3	0.20	0.2001	0.2080	0.2055	0.1966	0.1875	0.1982

Table 5 Result comparison in terms of MSE values evaluated using IAOA, AOA, WOA, CSO, GSA and PSO for Example 2 in case of same-order system

Algorithm	MSE				MSE (in dB)		
	Best	Worst	Average	SD	Best	Worst	Average
IAOA	**1.7310×10^{-04}**	**2.6100×10^{-02}**	**2.2000×10^{-03}**	**4.9625×10^{-02}**	**−75.2339**	**−31.6714**	**−53.1517**
AOA	4.9285×10^{-04}	5.8000×10^{-02}	5.5000×10^{-03}	4.3256×10^{-02}	−66.1457	−24.7270	−45.2450
WOA	6.7759×10^{-04}	0.0609	7.6000×10^{-03}	5.3261×10^{-08}	−63.3807	−24.3060	−42.4160
CSO	0.0022	0.0540	0.0090	1.3579×10^{-09}	−53.1973	−25.3509	−40.8867
GSA	0.0012	0.1095	0.1123	1.3579×10^{-02}	−58.5387	−19.2085	−39.0275
PSO	0.0040	0.1489	0.0177	3.6521×10^{-02}	−48.0554	−16.5414	−35.0490

Bold values signifies the minimum value of MSE obtained by the algorithm among existing algorithms

Table 6 Result comparison in terms of MSE values evaluated using IAOA, AOA, WOA, CSO, GSA and PSO for Example 2 in case of reduced-order system

Algorithm	MSE				MSE (in dB)		
	Best	Worst	Average	SD	Best	Worst	Average
IAOA	**1.4782×10^{-02}**	**0.5170**	**0.2892**	**5.9005×10^{-02}**	**−36.6053**	**−5.7302**	**−10.7760**
AOA	3.1145×10^{-02}	0.5867	0.4681	6.1196×10^{-02}	−30.1322	−4.6317	−6.5932
WOA	0.6073	0.9941	0.5806	4.3261×10^{-04}	−4.3319	−0.0514	−4.7225
CSO	0.2442	1.0054	0.6190	4.0394×10^{-07}	−12.2451	0.0468	−4.1662
GSA	0.1892	1.1095	0.5623	3.4679×10^{-02}	−14.4616	0.9025	−5.0006
PSO	0.8040	1.7609	0.9037	4.6571×10^{-02}	−1.8949	4.9147	−0.8795

Bold values signifies the minimum value of MSE obtained by the algorithm among existing algorithms

Table 7 r values evaluated for the Wilcoxon rank-sum test for two IIR models

IAOA Vs	AOA	WOA	CSO	GSA	PSO
Example 1					
Case 1	1.0457×10^{-10}	3.3341×10^{-09}	0.3811×10^{-08}	3.8241×10^{-05}	7.3982×10^{-06}
Case 2	0.0572	2.5497×10^{-07}	3.2815×10^{-04}	<u>0.5385</u>	<u>0.7156</u>
Example 2					
Case 1	3.8514×10^{-11}	<u>1.6120</u>	5.5574×10^{-09}	6.1114×10^{-07}	<u>0.1174</u>
Case 2	7.4074×10^{-11}	7.4074×10^{-09}	7.4074×10^{-08}	7.4074×10^{-05}	7.4074×10^{-09}

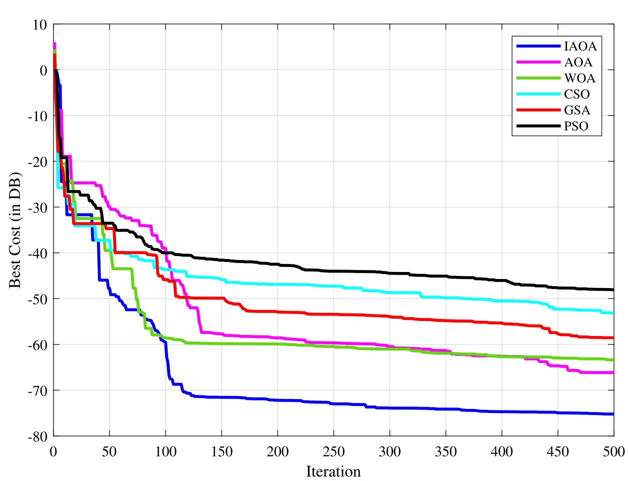


Fig. 10 Comparison of best optimization runs for Example 2 Case 1

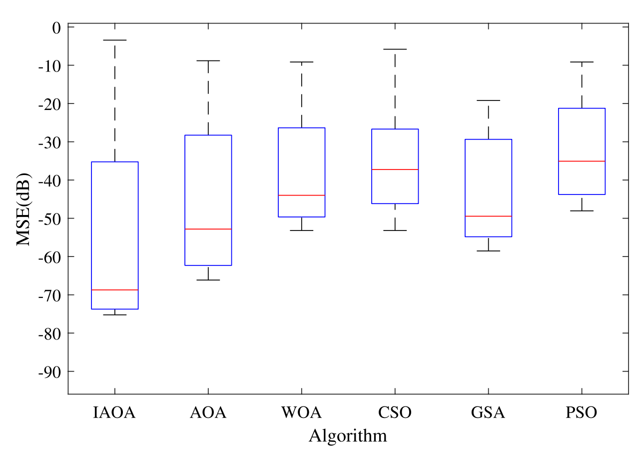


Fig. 11 Boxplot for Example 2 Case 1

Table 8 The comparison of the complexity analysis of IAOA with other employed algorithms

Algorithms	Population initialization	Fitness evaluation	Update process	Overall complexity
IAOA	$O(ND)$	$O(NKD)$	$O(2ND)$	$O(TNKD)$
AOA	$O(ND)$	$O(NKD)$	$O(ND)$	$O(TNKD)$
WOA	$O(ND)$	$O(NKD)$	$O(ND)$	$O(TNKD)$
CSO	$O(ND)$	$O(NKD)$	$O(tNKD)$	$O(TtNKD)$
GSA	$O(ND)$	$O(NKD)$	$O(N^2D)$	$O(TN^2D + TNKD)$
PSO	$O(ND)$	$O(NKD)$	$O(ND)$	$O(TNKD)$

$$H_s(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - d_1 z^{-1} - d_2 z^{-2} - d_3 z^{-3}} \tag{18}$$

In this subsection the performance IAOA is evaluated for the third-order system. Following 500 iterations for each algorithm, Table 4 lists the detailed comparison of the best estimated unknown system parameter values using IAOA and other existing algorithms. These evaluated parameters using IAOA and other existing methods are also demonstrated in Fig. 9. Analyzing the obtained model parameter values with IAOA reveals that it generally yield the parameter values near to the actual values in comparison to AOA and other existing algorithms. Furthermore, statistical results in terms of best, worst, average and standard deviation (SD) are evaluated and are listed in Table 5. The convergence curve for Example 2 Case 1 is shown in Fig. 10. It is demonstrated in Fig. 10 and Table 5 that analogous to the second-order system, IAOA convergence speed is faster and achieves lower MSE value compared to other existing algorithms, indicating its effectiveness in minimizing the error and converging to superior solutions. Box plot for Example 2 Case 1 is demonstrated in Fig. 11. Based on the observations, the IAOA and other existing algorithms can be placed as IAOA>AOA>WOA>CSO>GSA>PSO.

Case 2 The third-order plant described in Eq. (19) is modeled in this part using a second-order IIR as follows:

$$H_s(z) = \frac{a_0 + a_1 z^{-1}}{1 - d_1 z^{-1} - d_2 z^{-2}} \tag{19}$$

Figure 9 compares the convergence of all six methods and makes it evident that the minimal MSE is acquired by the IAOA. The performance of PSO was the lowest of all the algorithms and stagnated in an early stage of the optimization process, whereas the CSO and GSA gained the almost same minimum MSE value. The MSE values are listed in Table 6 and best MSE values observed for IAOA, AOA, WOA, CSO, GSA and PSO are 1.4782×10^{-02}, 3.1145×10^{-02}, 0.6073, 0.2442, 0.1892, and 0.8040 respectively. These results proved that IAOA is superior in achieving lower MSE value and hence effective for unknown IIR system identification compared to other existing algorithms.

Table 9 Runtime comparison of IAOA in comparison to other employed algorithms for Example 1 Case 1 and Example 2 Case 1

Algorithms	No. of iterations	Runtime (in s)	
		Example 1	Example 2
IAOA	500	8.0451	9.5603
AOA	500	7.5896	9.1425
WOA	500	8.5468	10.8742
CSO	500	14.5468	17.4562
GSA	500	15.8362	16.7803
PSO	500	8.3783	11.4567

To assess the difference in significance between the IAOA and the other methods on two types of benchmark IIR plants, the authors additionally use the Wilcoxon nonparametric statistical test in their simulations (Wilcoxon 1945). According to the stochastic nature of meta-heuristics, the statistical test must be performed. Derrac et al. (2011) and Mirjalili and Lewis (2013). r values < 0.05 demonstrate the results' statistically significant superiority. Table 7 provides a summary of the r values' statistical findings. Take note that the Table 7 highlights r values > 0.05. The statistical Wilcoxon sum test also shows the superiority of IAOA in comparison to other algorithms and the same can be verified from Table 7.

4.1 Computational complexity of IAOA in comparison to other employed algorithms

The computational complexity of an optimization algorithm depends on factors such as population size (N), number of iterations (T), dimension of the search space (D), length of input signal (K) and function evaluation cost (F). The comparison of the computational complexity of Improved AOA (IAOA), Standard AOA, Whale optimization algorithm (WOA), Cat Swarm Optimization (CSO), Gravitational Search Algorithm (GSA), and Particle Swarm Optimization (PSO) is given below. Each algorithm updates solution iteratively while evaluating fitness using MSE, which involves filtering and computing errors. The complexity of AOA is described by the following factors:

1. Complexity to initialize population: $O(ND)$
2. Fitness evaluation complexity: $O(NKD)$
3. Complexity of position update: $O(ND)$
4. Overall complexity: $O(T \times (ND + NKD)) = O(TNKD)$

The complexity of IAOA is comparable to AOA, but with extra dynamic inertia and mutation probability coefficient. Complexity due to dynamic inertia: $O(ND)$, complexity due to mutation probability: $O(ND)$, and the total complexity: $O(T \times (ND + NKD + 2ND)) = O(TNKD)$. So, the complexity of IAOA is similar to AOA but slightly more expensive due to additional mutation and inertia updates. CSO is more expensive $O(TtNKD)$ due to its seeking mode requiring multiple evaluations (t as no of trials). GSA is the most expensive $O(TN^2D + TNKD)$ due to the pairwise gravitational force computation. The comparison is illustrated in Table 8 for the employed algorithms.

The complexity of the employed algorithms are also calculated in terms of runtime analysis for Example 1 Case 1. The runtime is noted for each employed algorithm and is shown in Table 9. It is evident from Table 9 that IAOA has an execution time of 8.0451 s whcih is greater than AOA which is 7.5896 s for the 500 iterations. This is due to the presence of dynamic inertia and mutation probability constant parameters.

5 Conclusions

This paper discussed the unknown IIR system identification using the improved AOA. Two Examples have been considered for the unknown system identification. In the first example, a second order unknown system is identified using the same-order (second-order) and the reduced-order (first-order) system. The second example consider the third-order IIR system which is being identified with the same-order and reduced-order system. Unknown system parameters and convergence curve are considered as the performance measures for IAOA. The results obtained using IAOA have been compared with the other existing algorithms namely AOA, WOA, CSO, GSA, and PSO. IAOA exhibits faster convergence compared to existing optimization algorithms. It reduces the time required to identify the system parameters, making it an efficient choice for IIR system identification. However, this work focused on the evaluation of a second-order and third-order IIR system, and the scalability of IAOA for higher-order systems remains unexplored. Performance may degrade in more complex system models. The effectiveness of IAOA depends on parameter tuning (mutation probability, inertia weight, etc.), which requires empirical adjustment and may not generalize across all problem instances. Here, the authors primarily considered noise-free systems with linear parameter estimation. The performance of IAOA in noisy, nonlinear, or time-varying environments remains unexplored. For future work, IAOA can be applied to higher-order, multivariable and real world IIR systems. Also, IAOA can be extended to handle nonlinear system identification such as Hammerstein and Volterra model.

References

Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 376:113609

Abualigah L, Diabat A, Sumari P, Gandomi AH (2021) A novel evolutionary arithmetic optimization algorithm for multilevel thresholding segmentation of COVID-19 CT images. Processes 9(7):1155

Czapiewska A, Luksza A, Studanski R, Zak A (2020) Application of Diversity combining with RLS adaptive filtering in data transmission in a hydroacoustic channel. Sensors 20(24):7255

Dai C, Chen W, Zhu Y (2009) Seeker optimization algorithm for digital IIR filter design. IEEE Trans Ind Electron 57(5):1710–1718

Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18

Fan HY, Lampinen J (2003) A trigonometric mutation operation to differential evolution. J Global Optim 27:105–129

Fang H, Fu X, Zeng Z, Zhong K, Liu S (2022) An improved arithmetic optimization algorithm and its application to determine the parameters of support vector machine. Mathematics 10(16):2875

Habib MK, Cherri AK (1998) Parallel quaternary signed-digit arithmetic operations: addition, subtraction, multiplication and division. Optics Laser Technol 30(8):515–525

Janjanam L, Saha SK, Kar R, Mandal D (2024) Adaptive recursive system identification using optimally tuned Kalman filter by the metaheuristic algorithm. Soft Comput, pp 1–25

Jiang S, Wang Y, Ji Z (2015) A new design method for adaptive IIR system identification using hybrid particle swarm optimization and gravitational search algorithm. Nonlinear Dyn 79(4):2553–2576

Khatir S, Tiachacht S, Le Thanh C, Ghandourah E, Mirjalili S, Wahab MA (2021) An improved artificial neural network using arithmetic optimization algorithm for damage assessment in FGM composite plates. Compos Struct 273:114287

Le DC, Zhang J, Pang Y (2021) A novel pipelined neural FIR architecture for nonlinear adaptive filter. Neurocomputing 440:220–229

Li W, Xiang Z, Ren P, Li Q (2021) Adaptive filter bank multi-carrier waveform design for joint communication-radar system. Digital Signal Process 110:102950

Luo Q, Ling Y, Zhou Y (2020) Modified whale optimization algorithm for infinitive impulse response system identification. Arab J Sci Eng 45(3):2163–2176
Mirjalili S, Lewis A (2013) S-shaped versus V-shaped transfer functions for binary particle swarm optimization. Swarm Evol Comput 9:1–14
Mirza AF, Mansoor M, Zerbakht K, Javed MY, Zafar MH, Khan NM (2021) High-efficiency hybrid PV-TEG system with intelligent control to harvest maximum energy under various non-static operating conditions. J Clean Prod 320:128643
Mitra SK, Kuo Y (2006) Digital signal processing: a computer-based approach, vol 2. McGraw-Hill, New York
Mohammadi A, Zahiri SH, Razavi SM (2018) Infinite impulse response systems modeling by artificial intelligent optimization methods. Evolv Syst pp 1–17
Pai PF, Nguyen BA, Sundaresan MJ (2013) Nonlinearity identification by time-domain-only signal processing. Int J Non-Linear Mech 54:85–98
Panda G, Pradhan PM, Majhi B (2011) IIR system identification using cat swarm optimization. Expert Syst Appl 38(10):12671–12683
Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179(13):2232–2248
Saha SK, Kar R, Mandal D, Ghoshal SP, Mukherjee V (2013) A new design method using opposition-based BAT algorithm for IIR system identification problem. Int J Bio-Inspired Comput 5(2):99–132
Sayed AH (2003) Fundamentals of adaptive filtering. Wiley, Hoboken
Shi Y, Eberhart R (1998) A modified particle swarm optimizer. In: 1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360), pp 69–73. IEEE
Singh S, Ashok A, Kumar M, Rawat TK (2019) Adaptive infinite impulse response system identification using teacher learner based optimization algorithm. Appl Intell 49(5):1785–1802
Singh S, Ashok A, Rawat TK, Kumar, M (2016) Optimal IIR system identification using flower pollination algorithm. In: 2016 IEEE 1st international conference on power electronics, intelligent control and energy systems (ICPEICES), pp 1–6. IEEE
Wilcoxon F (1945) Individual comparisons by ranking methods. Biom Bull 1(6):80–83
Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1(1):67–82
Xu YP, Tan JW, Zhu DJ, Ouyang P, Taheri B (2021) Model identification of the proton exchange membrane fuel cells by extreme learning machine and a developed version of arithmetic optimization algorithm. Energy Rep 7:2332–2342
Zarai K, Cherif A (2021) Adaptive filter based on Monte Carlo method to improve the non-linear target tracking in the radar system. Aerospace Syst 4(1):67–74
Zhang Q, Luodi Z, Long Z (2021) A two-step robust adaptive filtering algorithm for GNSS kinematic precise point positioning. Chin J Aeronaut 34(10):210–219
Zhou X, Yang C, Gui W (2014) Nonlinear system identification and control using state transition algorithm. Appl Math Comput 226:169–179
Zou DX, Deb S, Wang GG (2018) Solving IIR system identification by a variant of particle swarm optimization. Neural Comput Appl 30(3):685–698