RESEARCH ARTICLE - SYSTEMS ENGINEERING

Modified Whale Optimization Algorithm for Infinitive Impulse Response System Identification

Qifang Luo[1,2] · Ying Ling[1,2] · Yongquan Zhou[1,2]

Abstract

System identification based on infinite impulse response (IIR) models has received much attention because it is used in a variety of real-world applications. However, an IIR model might have a multimodal error surface. To attain an ideal identification, an efficient and robust method is necessary. In this study, a modified whale optimization algorithm (WOA) with a ranking-based mutation operator, called the RWOA, is presented to solve the IIR system identification problem. The RWOA integrates a ranking-based mutation operator into the basic WOA to enhance performance by speeding up the convergence rate and then enhances the exploitation capability. The experimental results of actual and reduced-order identification for a standard system using our proposed RWOA were superior to those of five state-of-the-art algorithms (including the basic WOA), in terms of improving the quality and stability of the results, in most cases, and significantly speeding up convergence.

Keywords Ranking-based mutation operator · Whale optimization algorithm · RWOA · IIR system · System identification

1 Introduction

In the last few years, the adaptive infinite impulse response (IIR) filter has become very popular and has increasingly attracted the attention of many researchers and scholars [1, 2]. It is very popular in diverse fields of application, such as control systems [3], communication [4], signal processing [5], and image processing [6]. In the case of several system identification problems, adaptive IIR filters tend to model the unknown system in accordance with some error function between the output of the candidate model and the output of the plant [7]. To acquire an ideal identification, it is necessary to search the appropriate filter parameters to enable the error surface between the output of the adaptive filter and the output of the plant to achieve the minimum value.

There is a major concern that the error surfaces produced by the IIR filter are typically multimodal, which makes them difficult to minimize, and they are quite easily trapped in

local minima. To overcome this problem, researchers have attempted to use efficient and robust evolutionary computing algorithms inspired by nature for IIR system identification. In this case, IIR system identification is modeled as a minimization problem and then solved using an evolutionary computing algorithm [8]. For example, Krusienski and Jenkins [2] used the particle swarm optimization algorithm (PSO), which is a popular and classical evolutionary computing algorithm inspired by the social behavior of flocking birds, to create the adaptive IIR filter structure. Later, Zou et al. [7] proposed an improved version of PSO called IPSO to process IIR system identification. The simulation results indicated the efficiency of IPSO compared with different versions of the PSO algorithm. Patwardhan et al. [9] used the cuckoo search algorithm (CS), which is inspired by the lifestyle of birds, to handle IIR feedback system identification. Panda et al. [10] exploited the cat swarm optimization algorithm, which is derived from the behavior of cats, to handle different cases of IIR modeling. Furthermore, Saha et al. [11] used the gravitational search algorithm (GSA), which is based on the law of gravity, to solve the IIR model, whereas Karaboga [12] used IIR models to test the efficiency of the artificial bee colony algorithm, which is inspired by the behavior of bees searching for food. The results reported in these studies show that these evolutionary computing algorithms can efficiently solve the IIR system identification

✉ Yongquan Zhou
  yongquanzhou@126.com

1 College of Information Science and Engineering, Guangxi University for Nationalities, Nanning 530006, China

2 Key Laboratories of Guangxi High Schools Complex System and Computational Intelligence, Nanning 530006, Guangxi, China

problem. Despite the merits of the aforementioned studies, according to the well-known no-free-lunch theorem (NFL), no single optimization algorithm exists that can outperform all other algorithms for all optimization problems [13]. Because the IIR model has different cases, it is possible that an algorithm performs well on a test case, but poorly on another case. Therefore, this has stimulated researchers and scholars to investigate the efficiency of new algorithms to solve the IIR model identification problem and problems in other fields.

The whale optimization algorithm (WOA) is a newly proposed metaheuristic introduced by Mirjalili et al. [14] based on the bubble-net feeding technique of baleen whales. Compared with other algorithms, the WOA can provide competitive results and has superior performance. The merits of this algorithm are that it is easy to implement, the structure is very simple, and it requires few control parameters. Thus, the original version of the WOA has attracted the attention of many scholars and has been applied in a wide variety of application studies [15–20]. Although the basic WOA has been shown to perform well in the aforementioned studies, when compared with other traditional algorithms, it still has some drawbacks, such as slow convergence and becoming stuck easily in local optima. Consequently, different WOA variants have been proposed to improve its performance. An improved WOA was developed by Oliva et al. [21], who combined the features of the standard WOA with chaotic maps to improve performance and applied it to the parameter estimation of photovoltaic cells. Hu et al. [22] also proposed a modified version of WOA. They introduced the inertia weight into the WOA to improve performance, tested the modified algorithm on benchmark functions, and predicted the daily air quality index. Moreover, Trivedi et al. [23] developed a new variant of the WOA using a hybridization of the PSO algorithm called PSO-WOA for global numerical function optimization. Another novel adaptive WOA algorithm for global optimization was proposed by Bhesdadiya et al. [24] and used as an adaptive mechanism to update the whale's position. A Kaveh [25] proposed an enhanced WOA by modifying the updating mechanism of the basic WOA and applied it to the size optimization of a skeletal structure. Mafarja and Mirjalili [26] proposed a hybrid WOA with simulated annealing and used it for feature selection. Considering all these improvements to the basic WOA, we found that they are appropriate for different types of practical applications; however, these WOA variants are not suitable for solving all classes of practical problems according to the NFL [13].

In this paper, we propose a modified WOA called RWOA that integrates a ranking-based mutation operator with the basic WOA to make the basic WOA converge faster, and more robust and suitable for practical applications. The RWOA was applied to solve the IIR model identification
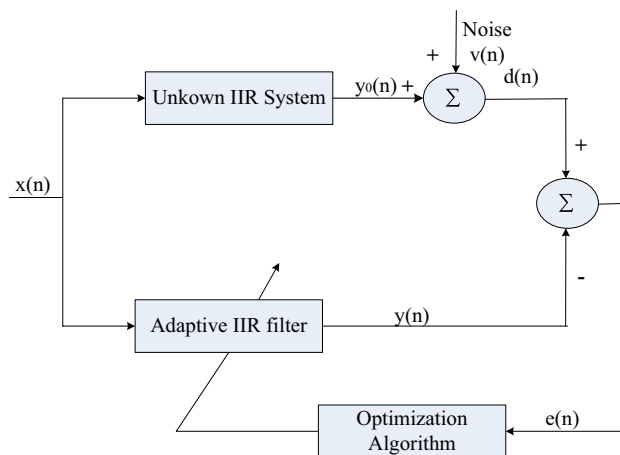

Fig. 1 Adaptive IIR filter for system identification

problem and was compared with existing evolutionary computing algorithms. The simulation results demonstrated clearly that our proposed RWOA exhibited superior identification performance.

The remainder of this paper is organized as follows: In Sect. 2, we present the problem of adaptive IIR filter model identification. In Sect. 3, we present the steps involved in whale optimization. In Sect. 4, we present our proposed RWOA. We describe the simulation results and analyze them in Sect. 5. Finally, we discuss our conclusions and future work in Sect. 6.

2 Adaptive IIR Filter Model

The adaptive IIR filter has been widely used in system identification because a number of problems encountered in signal processing can be modeled as a system identification problem. For the system identification configuration, to make the filter's output close to the output of an unknown system, the main task of the adaptive algorithm is to search for appropriate filter coefficients. The block diagram for an adaptive IIR system identification is shown in Fig. 1.

The input–output relationship of the IIR system can be described as follows [27, 28]:

$$\sum_{l=0}^{L} a_1 y(n-l) = \sum_{k=0}^{K} b_k x(n-k) \qquad (1)$$

where $x(n)$ and $y(n)$ denote the input and output of the filter, respectively, and $L(\geq K)$ represents the filter's order. Let $a_0 = 1$. Then the transfer function of this IIR filter is defined as

$$H(z) = \frac{\sum_{k=0}^{K} b_k z^{-k}}{1 + \sum_{l=1}^{L} a_l z^{-l}}. \qquad (2)$$

In the design method, $H_p(z)$ is the transfer function of the unknown plant and $H_m(z)$ is the transfer function of the IIR model. From Fig. 1, it is clear that the output of the IIR filter is $y(n)$; $d(n) = y_0(n) + v(n)$ is the overall response of the unknown IIR plant; $y_0(n)$ is the output of the unknown plant; and $v(n)$ is additive white Gaussian noise. The equation $e(n) = d(n) - y(n)$ represents the error signal. Hence, the main goal of identification is to formulate a minimization problem that can be defined as the cost function $J(w)$ as follows:

$$\text{MSE} = J(w) = \frac{1}{N} \sum_{n=1}^{N} e^2(n), \tag{3}$$

where N denotes the number of input samples used for the calculation of the objective function. The mean square error (MSE) is equal to $J(w)$, which produces the coefficient vectors of the IIR model, where $w = (a_1, \ldots a_L, b_0, b_1, \ldots b_K)^T$. The objective of the algorithm is to minimize the MSE through adjusting coefficient vector w of transfer function $H_p(z)$.

3 Whale Optimization Algorithm (WOA)

The WOA is a novel metaheuristic algorithm, which was proposed by Mirjalili and Lewis [14]. The WOA simulates the special foraging behavior of humpback whales, whereby they search for and attack prey (herds and small fish), which is called the bubble-net hunting strategy. Figure 2 shows the bubble-net hunting model.

Figure 2 clearly shows that the humpback whale hunts prey by creating distinctive bubbles and swimming in a "9"-shaped path. The motivation of the WOA is this special
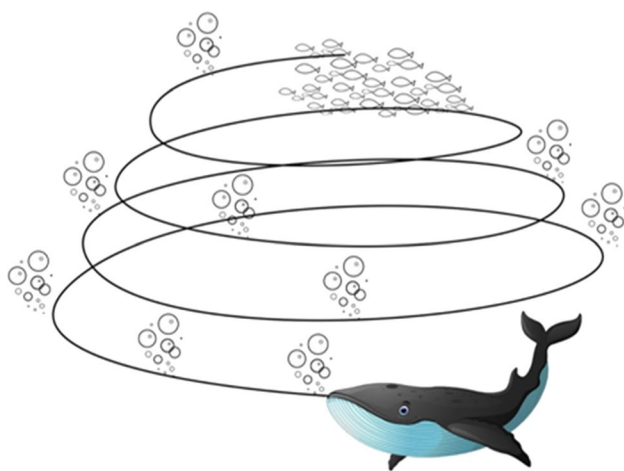

Fig. 2 Bubble-net hunting behavior of humpback whales

hunting method. In addition to simulating the bubble-net strategy in the WOA, the whale also has another strategy, which is simulating encircling the prey. This can be summarized as follows: first, the humpback whale observes the location of the prey (fish herds) and then encircles them to hunt them using the bubble-net strategy. The formulas of the two hunting strategies that is, the encircling prey strategy and bubble-net feeding strategy, proposed in the WOA are defined as

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X*}(t) - \vec{A} \cdot \vec{D} & \text{if } p < 0.5 \\ \overrightarrow{D'} \cdot e^{bl} \cdot \cos(2\pi l) + \overrightarrow{X*}(t) & \text{if } p \geq 0.5 \end{cases} \tag{4}$$

where p is an arbitrary number in $[0, 1]$, $\overrightarrow{D'} = \left| \overrightarrow{X*}(t) - \vec{X}(t) \right|$ represents the length from the ith whale to the best position of the leader whale (the best location acquired thus far), l denotes a random value in $[-1, 1]$, b denotes a constant that defines the shape of a logarithmic spiral, $\cdot$ denotes constituent-by-constituent multiplication, t denotes the current iteration, $\vec{D} = \left| \vec{C} \cdot \overrightarrow{X*}(t) - \vec{X}(t) \right|$, X^* denotes the position vector of the optimum measure acquired thus far, and $X(t)$ denotes the position vector. Note that X^* must be updated at every step of the optimization if there exists a better solution. Here, $\vec{A} = 2\vec{a} \cdot \vec{r} - \vec{a}$, $\vec{C} = 2 \cdot \vec{r}$, and the values of $\vec{A}$ are in the range $[-a, a]$, where the value of $\vec{a}$ linearly decreases from 2 to 0 because it is calculated as $a = 2 - 2 * t/t_{\max}$. Note that in the exploration and exploitation stages, the value of $\vec{a}$ remains the same and the value of r is in the range $[0, 1]$.

Equation (4) contain two formulas, where the first is the mathematical model of the encircling prey strategy of the humpback whale and the second is the mathematical model of the bubble-net hunting strategy. Note that humpback whales swim around prey inside a contracting circle or move with a conical logarithmic spiral motion to prey on fish herds simultaneously. We assume that the humpback whale adopts these two hunting mechanisms with a probability of 50%, and then variable p switches between these two components with a 50% probability. We know that both the exploitation stage and exploration stage are the main stages of the optimizing problem in population-based algorithms. Therefore, in the WOA, vector $\vec{A}$ is used for exploration to scan for prey, and its value is greater than 1 or less than -1. When $\left| \vec{A} \right| \geq 1$, the whales are forced into exploration to determine the global optimum and eliminate many local minima. The mathematical model is formulated as

$$\vec{D} = \left| \vec{C} \cdot \overrightarrow{X_{\text{rand}}} - \vec{X} \right| \tag{5}$$

$$\vec{X}(t+1) = \overrightarrow{X_{\text{rand}}} - \vec{A} \cdot \vec{D} \tag{6}$$

where $\overrightarrow{X_{\text{rand}}}$ is chosen from the current generation and indicates a random position vector (a random whale). When

$|\vec{A}| \geq 1$, the WOA focuses on exploration using Eq. (6), so the whale updates its position according to $\overrightarrow{X_{\text{rand}}}$, which is a stochastically chosen whale, whereas when $|\vec{A}| < 1$, the WOA focuses on exploitation, and the whale updates its position based on the best searching agent acquired thus far. The flowchart of the WOA is illustrated in Fig. 3.

4 Modified Whale Optimization Algorithm with Ranking-Based (RWOA)

As mentioned, in this paper, we present a modified version of the WOA called the RWOA for solving IIR system identification problems. The RWOA combines a ranking-based mutation operator with the pure WOA to speed up the rate of convergence and results in smoothly balanced intensification and diversification. Therefore, this approach improves the quality of the basic WOA. In the following subsection,

we first briefly review the ranking-based mutation operator and then present our RWOA in detail.
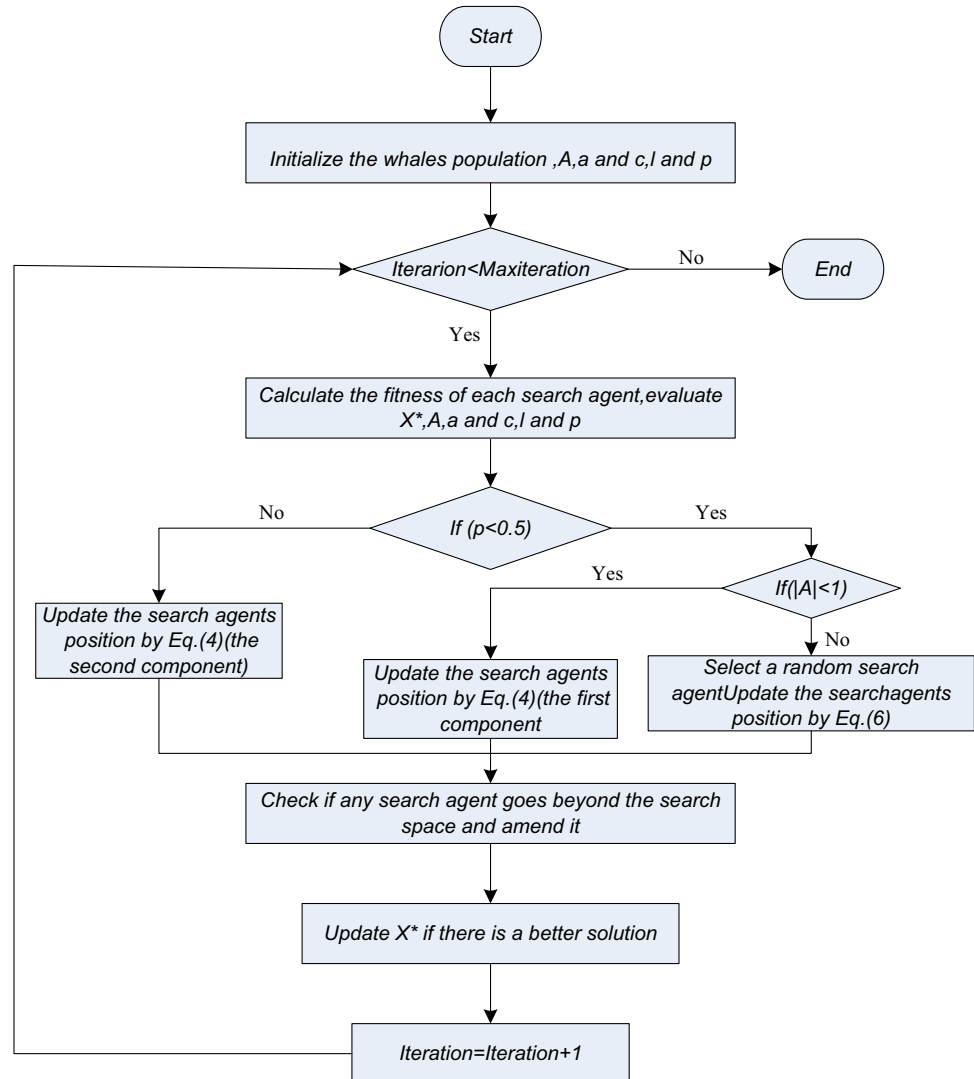
4.1 Ranking-Based Mutation Operator

1. *Ranking Assignment* For the purpose of choosing information about a good search agent from the WOA population, it is necessary to assign a rank for each whale according to the related fitness. First, the population (each whale) is sorted in ascending order (i.e., from best to worst) according to the fitness of each individual (each whale). The rank of an individual is assigned as

$$R_i = N_{\text{p}} - i, \quad i = 1, 2, \ldots, N_{\text{p}}, \tag{7}$$

where N_{p} indicates the number of the population. From Eq. (7), the highest rank is assigned to the best individual (best whale) in the current population and then other individuals acquire their corresponding ranking order.

Fig. 3 Flowchart of the WOA

2. *Selection Probability* When each individual is ranked, selection probability P_i of the ith individual (whale) is modeled as

$$P_i = \frac{R_i}{N_p}, \quad i = 1, 2, \ldots, N_p. \tag{8}$$

Algorithm 1 presents the ranking-based mutation operator of "DE/rand/1." In nature, useful information is always contained in good species; hence, good species always have a greater probability of being selected to propagate a new generation. Similar to nature, in accordance with the approach described in Algorithm 1, it is clear that the higher the ranking of the individual, the greater the probability that the individual is chosen as the base vector or terminal vector in the mutation operator to propagate useful information about the population to the offspring. Note that the starting vector is not selected based on its selection probability because, assuming that both the vectors in the difference vector are selected from better vectors, the searching step-size of the difference vector may have a probability that decreases quickly, which results in premature convergence [29, 30].

4.2.1 Complexity Analysis

In this section, both the time and space complexity of the RWOA are analyzed.

Time Complexity The time complexity of the RWOA is briefly analyzed in this subsection. The RWOA contains five major steps: initialization, ranking-based mutation, encircling prey (exploration phase), attacking prey (exploitation phase), and halting judgment. In the RWOA, N is the number of search agents, T is the maximum number of iterations, and D is the dimension of the problem. The initialization step contains a double loop (N and D times), so its time complexity is $O(N*D)$. The ranking-based mutation stage, exploration phase, and exploitation phase all contain a triple loop (N, D, and T times), so the time complexity for each step is $O(N*D*T)$. The time complexity of the last step, that is, the halting judgment step, is $O(1)$. Thus, the total time complexity of the proposed RWOA is $O(N*D*T)$. Table 1 shows a comparison of the time complexity for the RWOA and WOA.

Space Complexity The proposed RWOA uses the number of search agents (whales), which is N, to calculate the space complexity, and the dimension of the problem to be solved

---

Algorithm 1. Ranking-based mutation operator of "DE/rand/1"

1. Sort the population, and assign the ranking and selection probability P_i for each individual

2. Randomly select $r_1 \in \{1, Np\}$ {base vector index}

3. **while** rand $> p_{r_1}$ *or* $r_1 == i$

4.   Randomly select $r_1 \in \{1, Np\}$

5. **end**

6. Randomly select $r_2 \in \{1, Np\}$ {terminal vector index}

7. **while** rand $> p_{r_2}$ *or* $r_2 == r_1$ *or* $r_2 == i$

8.   Randomly select $r_2 \in \{1, Np\}$

9. **end**

10. Randomly select $r_3 \in \{1, Np\}$ {starting vector index}

11. **while** $r_3 == r_2$ *or* $r_3 == r_1$ *or* $r_3 == i$

12.   Randomly select $r_3 \in \{1, Np\}$

13. **end**

4.2 Main Procedure of the RWOA

The RWOA combines the aforementioned ranking-based mutation operator with the basic WOA. The main procedure of the proposed enhanced RWOA is summarized in Fig. 4.

is D. Therefore, the total space complexity of the RWOA is $O(N*D)$. The WOA also uses N search agents to calculate the space complexity, so its related total space complexity is $O(N*D)$, which is the same as that of the RWOA.
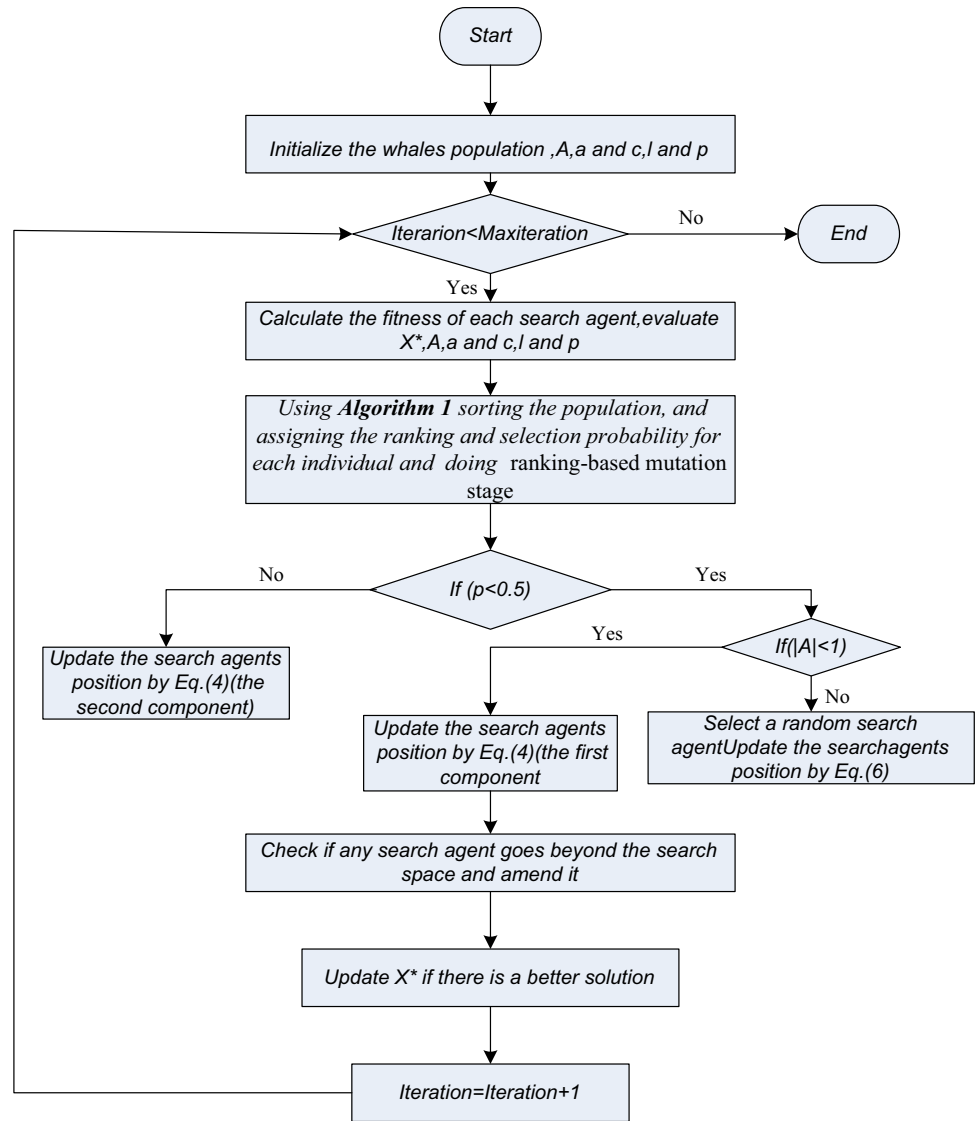
Fig. 4 Flowchart of the
enhanced RWOA

Table 1 Comparison of the time complexity

Algorithms	Time complexity
RWOA	$O(N*D*T)$
WOA	$O(N*D*T)$

4.2.2 Advantages of the RWOA

Our proposed enhanced RWOA has some major advantages, which are summarized as follows:

(A) Because good parents are more likely to be chosen, the ranking-based mutation operator is quite useful for improving the exploitation ability of the enhanced RWOA; hence, it enhances the entire performance compared with the basic WOA algorithm.

(B) Our RWOA has an uncomplicated structure, which means that it does not destroy the simple structure of the basic WOA. The enhanced RWOA only requires a small amount of extra execution time when ranking the population and choosing individuals, but the extra computational cost is negligible. According to the complexity analysis of the RWOA and WOA, the enhanced RWOA does not significantly change the entire complexity of the pure WOA.

5 Modified RWOA for IIR System Identification

In this section, our proposed enhanced RWOA algorithm is used to solve the IIR system identification problem, which is mentioned in Sect. 2. In this case, N whales, which search in

Fig. 5 Flowchart of the enhanced RWOA algorithm for IIR system identification
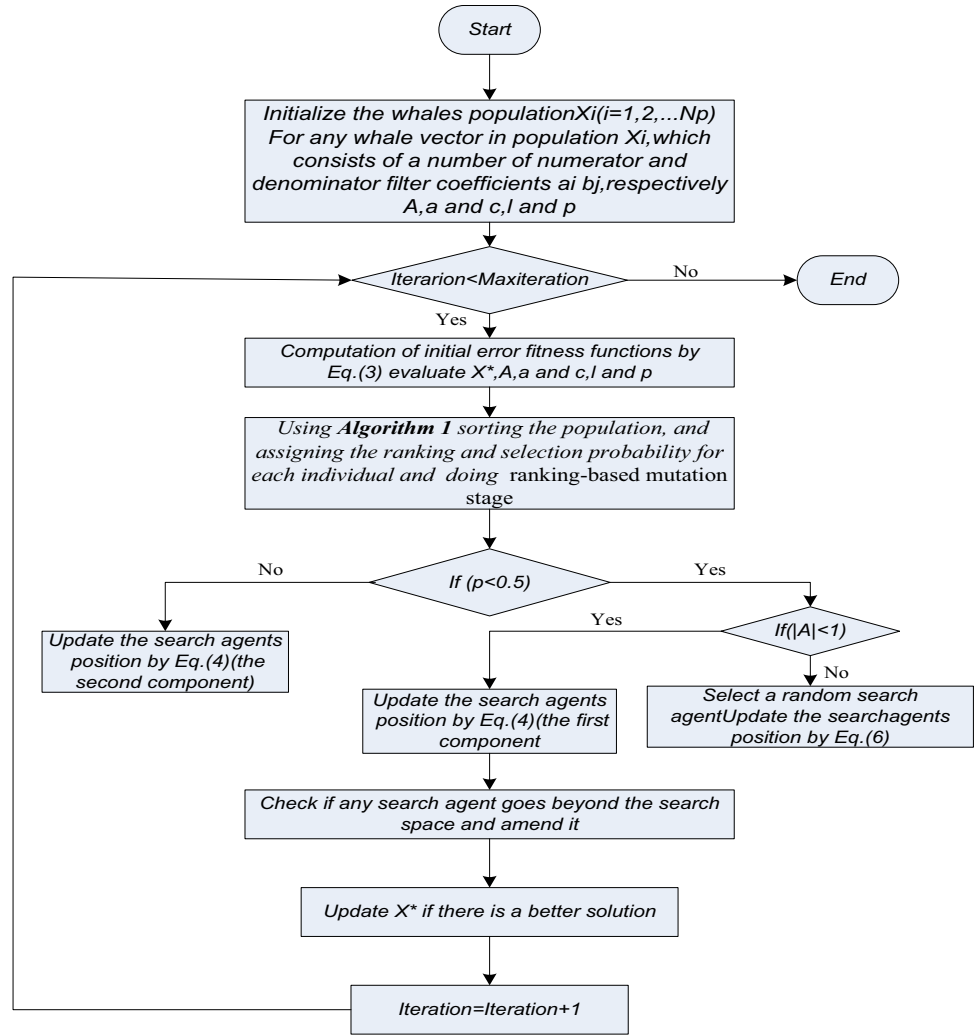
the search zone, represent the estimated parameter vectors of the filters. The locations of all the whales are updated using the enhanced RWOA to obtain the minimum error between the output of the filter and an unknown plant. The fitness for the ith whale at position $\vec{X}_i$ is represented by the objective function given by Eq. (3). The flowchart of the enhanced RWOA for IIR model identification is presented in Fig. 5.

5.1 Experimental Results

To verify the performance of our enhanced RWOA for identifying different IIR systems, an extensive experiment was conducted on three types of benchmark IIR plants, with orders two, three, and four, which were chosen from Refs. [7, 10, 12, 27]. The RWOA was compared with five state-of-the-art algorithms: PSO [31], BA [32], GSA [33], GOA [34], and pure WOA [14]. The objective function, which is the MSE described in Sect. 2, was used as the metric. In the simulations, the problem parameters were set as follows: The input used a white sequence, which contained 100 samples;
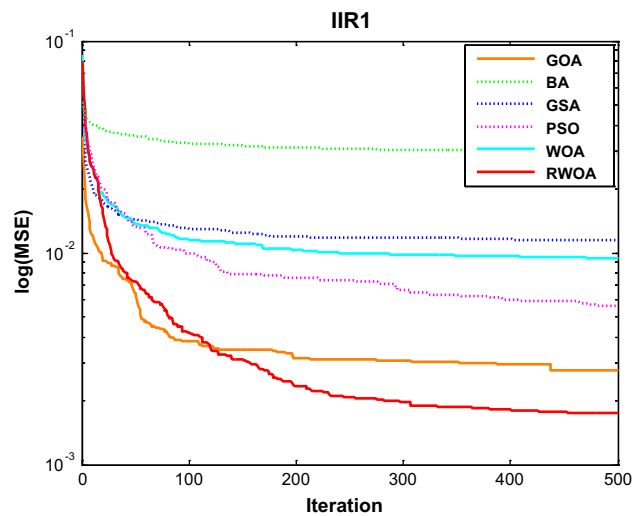

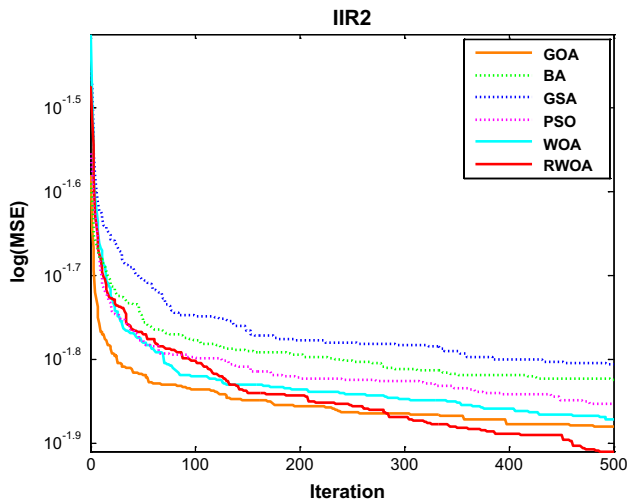**Fig. 6** Average MSE curve for model-1, case 1
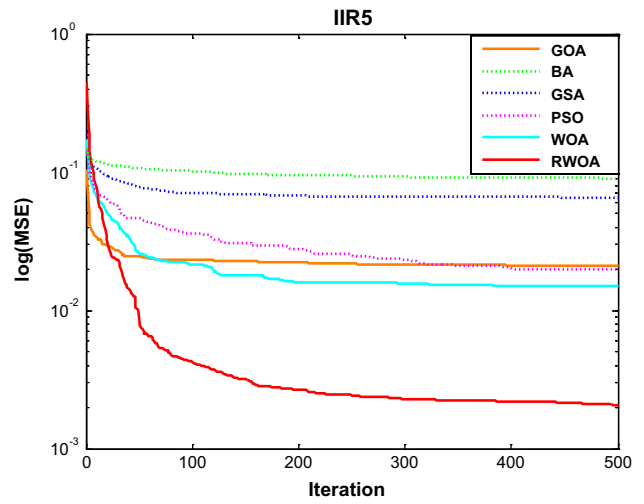
Fig. 7 Average MSE curve for model-1, case 2

Fig. 8 Average MSE curve for model-2, case 1
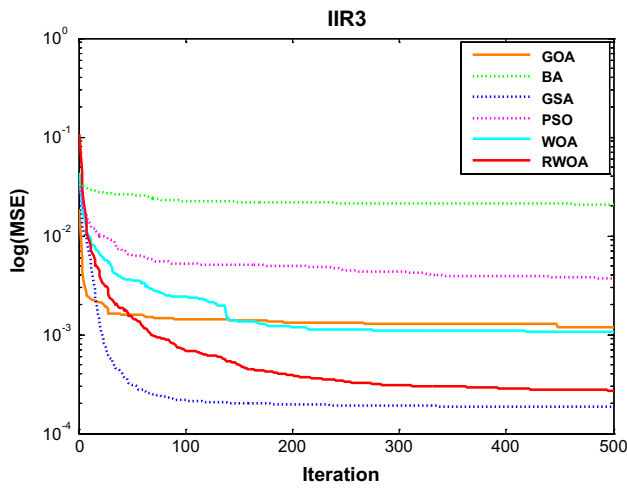

Fig. 9 Average MSE curve for model-2, case 2

Fig. 10 Average MSE curve for model-3, case 1
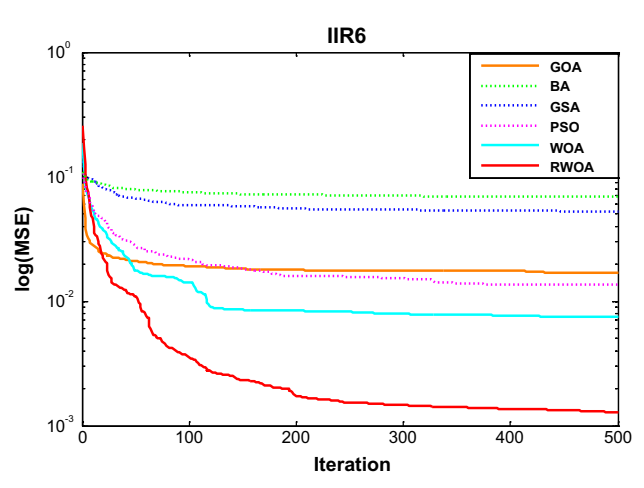
Fig. 11 Average MSE curve for model-3, case 2
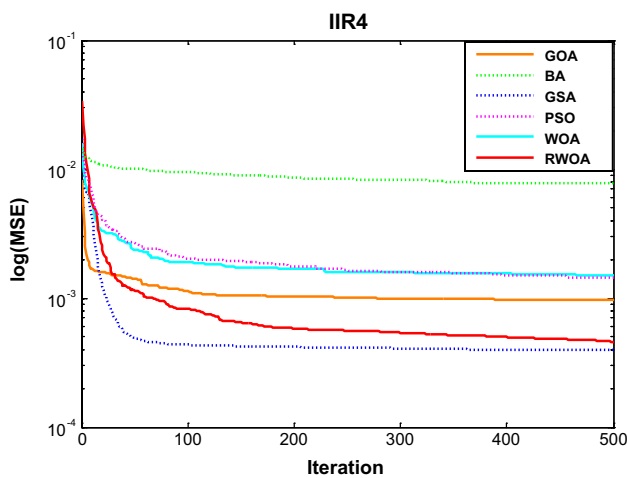
additive noise was not considered in our study. The proposed RWOA was implemented in the MATLAB programming language for the four algorithms. All the experiments were run on a computer with an AMD Athlon (tm) II*4640 processor and 4 GB of RAM using MATLAB R2012a.

The experimental results obtained for all the IIR plants test cases that contained full and reduced orders are presented in Tables 3, 4, 6, 7, 9, and 10, and illustrated in Figs. 6, 7, 8, 9, 10, and 11 for the MSE and convergence characteristics. The best average (Mean) and standard deviation (Std) are highlighted in bold. Additionally, the estimated values obtained by all the algorithms are presented in Tables 2, 5, and 8.

The Wilcoxon nonparametric statistical test [35] was also implemented for our simulations to test the difference in significance between the RWOA and the other algorithms

Table 2 Comparison of parameter estimation for model-1, case 1

Parameters	Actual values	Estimated values					
		BA	GSA	PSO	GOA	WOA	RWOA
a_0	0.05	0.0482	0.1097	0.0285	0.0499	0.0385	0.0565
a_1	-0.4	-0.3951	-0.334	-0.3811	-0.3999	-0.3824	-0.4071
b_1	1.1314	1.1457	1.2896	0.996	1.1311	1.1290	1.1303
b_2	-0.25	-0.2667	-0.3873	-0.1386	-0.2497	-0.2468	-0.2492

Table 3 Results of the MSE obtained by the BA, GSA, PSO, GOA, WOA, and RWOA for model-1, case 1

MSE	BA	GSA	PSO	GOA	WOA	RWOA
Best	2.11E−06	1.40E−03	1.60E−03	1.35E−09	2.26E−05	3.29E−06
Worst	8.95E−02	1.96E−02	1.07E−02	2.09E−02	2.02E−02	1.72E−02
Mean	2.96E−02	1.15E−02	5.60E−03	2.80E−03	9.50E−03	**1.60E−03**
Std	1.98E−02	4.80E−03	3.60E−03	6.10E−03	7.10E−03	**3.50E−03**

Table 4 Results of the MSE obtained by the BA, GSA, PSO, GOA, WOA, and RWOA for model-1, case 2

MSE	BA	GSA	PSO	GOA	WOA	RWOA
Best	8.70E−03	1.13E−02	1.10E−02	1.06E−02	9.20E−03	1.05E−02
Worst	3.39E−02	2.39E−02	1.62E−02	2.88E−02	2.09E−02	1.98E−02
Mean	1.50E−02	1.57E−02	1.40E−02	1.32E−02	1.34E−02	**1.23E−02**
Std	5.10E−03	2.90E−03	2.20E−03	4.31E−03	3.40E−03	**2.00E−03**

Table 5 Comparison of parameter estimation for model-2, case 1

Parameters	Actual values	Estimated values					
		BA	GSA	PSO	GOA	WOA	RWOA
a_0	-0.20	-0.2021	-0.2007	-0.1688	-0.1999	-0.2267	-0.2006
a_1	-0.40	-0.4071	-0.4475	-0.4388	-0.4004	-0.3766	-0.3997
a_2	0.50	0.4939	0.3809	0.4283	0.4984	0.4993	0.4997
b_1	0.60	0.5804	0.3772	0.5000	0.5971	0.6125	0.5991
b_2	-0.25	-0.2494	-0.2796	-0.2508	-0.2503	-0.2390	-0.2530
b_3	0.20	0.1908	-0.2796	0.1812	0.1983	0.2188	0.2021

Table 6 Results of MSE obtained by the BA, GSA, PSO, GOA, WOA, and RWOA for model-2, case 1

MSE	BA	GSA	PSO	GOA	WOA	RWOA
Best	7.70E−07	6.35E−05	2.65E−04	2.38E−08	5.05E−05	3.27E−07
Worst	8.08E−02	9.38E−04	7.10E−03	1.24E−02	9.70E−03	1.20E−03
Mean	2.09E−02	**1.85E−04**	3.70E−03	1.20E−03	1.10E−03	3.09E−04
Std	1.75E−02	**1.52E−04**	1.60E−03	2.83E−03	1.80E−03	2.82E−04

Table 7 Results of the MSE obtained by the BA, GSA, PSO, GOA, WOA, and RWOA for model-2, case 2

MSE	BA	GSA	PSO	GOA	WOA	RWOA
Best	3.69E−04	3.36E−04	8.09E−04	3.74E−04	4.20E−04	3.71E−04
Worst	2.37E−02	4.77E−04	2.90E−03	1.27E−02	4.70E−03	7.73E−04
Mean	7.70E−03	**3.94E−04**	1.40E−03	9.31E−04	1.50E−03	4.59E−04
Std	6.70E−03	**2.70E−05**	4.48E−04	2.26E−03	1.20E−03	7.33E−05

Table 8 Comparison of parameter estimation for model-3, case 1

Parameters	Actual values	Estimated values					
		BA	GSA	PSO	GOA	WOA	RWOA
a_0	1.00	0.9734	0.4294	0.7114	0.9989	1.00	1.00
a_1	-0.90	-0.7024	-0.7442	-0.8393	-0.9223	-0.9188	-0.9145
a_2	0.81	0.7232	0.7204	0.8438	0.8050	0.8003	0.8022
a_3	-0.729	-0.5615	-0.8599	-0.6521	-0.4167	-0.9659	-0.8123
b_1	-0.04	-0.2836	-0.2073	-0.0618	-0.0191	-0.0228	-0.0253
b_2	-0.2775	-0.5084	-0.5886	-0.5574	-0.2472	-0.2528	-0.2546
b_3	0.2101	0.0394	0.0144	0.0021	0.2354	0.2274	0.2283
b_4	-0.14	-0.1734	-0.4084	-0.3692	-0.1299	-0.1368	-0.1335

Table 9 Results of the MSE obtained by the BA, GSA, PSO, GOA, WOA, and RWOA for model-3, case 1

MSE	BA	GSA	PSO	GOA	WOA	RWOA
Best	4.55E−04	2.26E−02	7.40E−03	1.14E−05	9.02E−06	6.70E−06
Worst	2.56E−01	1.80E−01	3.42E−02	1.25E−01	9.58E−02	6.60E−03
Mean	8.19E−02	9.02E−02	1.96E−02	2.11E−02	1.50E−02	**1.50E−03**
Std	4.59E−02	4.77E−02	7.10E−03	3.31E−02	2.67E−02	**1.60E−03**

Table 10 Results of MSE obtained by the BA, GSA, PSO, GOA, WOA and RWOA for model-3, case 2

MSE	BA	GSA	PSO	GOA	WOA	RWOA
Best	4.05E−04	1.65E−02	4.60E−03	4.54E−04	7.74E−04	3.56E−04
Worst	1.49E−01	9.57E−02	2.20E−02	6.81E−02	3.92E−02	2.00E−03
Mean	6.90E−02	5.27E−02	1.36E−02	1.69E−02	7.40E−03	**8.21E−04**
Std	3.75E−02	1.87E−02	4.70E−03	1.98E−02	9.20E−03	**4.70E−04**

on three types of benchmark IIR plants. It was necessary to conduct the statistical test because of the stochastic nature of metaheuristics [36, 37]. p values < 0.05 prove the statistically significant superiority of the results. The statistical results of the p values are summarized in Table 10. Note that p values > 0.05 are underlined in the table.

5.1.1 Parameter Setup

The simulation parameters used for all the algorithms were as follows: For a fair comparison, the population size for each algorithm was set to $N = 30$ and the maximum number of termination iterations for each algorithm was set to $T = 500$. In addition to the basic control parameters, each of the algorithms also had special control parameters. For our proposed RWOA, the scaling factor was set to $F = 0.7$ and the component $\vec{a}$ was linearly diminished from 2 to 0 through the course of the iteration. For the pure WOA, the value of $\vec{a}$ was the same as that for the RWOA and decreased linearly from 2 to 0 through the course of the iteration. For the PSO, cognitive constant c_1 was 1.5 and social constant c_2 was 2. Inertia constant ω was 1. For the BA, loudness A_0 was set to 0.9 and pulse rate r_0 was 0.5. For the GSA, gravitational constant G_0 was 1 and coefficient of decrease α was 20. For

the last algorithm, GOA, the constant of intensity of attraction f was set to 0.5 and the constant of attractive of large scale l was set to 1.5. All the algorithms in our experiment were run 30 times for all the IIR plant test cases to obtain quantitative results.

Model 1 For the first experiment, the transfer function of a second-order plant is modeled as

$$H_{\mathrm{p}}(z) = \frac{0.05 - 0.4z^{-1}}{1 - 1.1214z^{-1} + 0.25z^{-2}}. \tag{9}$$

Case 1 The transfer function of this second-order plant can be defined as a second-order model:

$$H_{\mathrm{m}}(z) = \frac{a_0 + a_1 z^{-1}}{1 - b_1 z^1 - b_2 z^{-2}}. \tag{10}$$

Table 2 lists the estimated parameters acquired by the six algorithms, and it shows that GOA estimated the coefficients slightly better than the RWOA, and these two algorithms performed better than the other four algorithms. This indicates that the RWOA had a strong ability to acquire accurate estimated parameters. Table 3 presents the statistical results in terms of MSE. Clearly, the standard deviation of the MSE

Table 11 p values calculated for the Wilcoxon rank-sum test for three IIR models

RWOA vs	BA	GSA	PSO	GOA	WOA
Model 1					
Case 1	1.1737e−09	9.7555e−10	2.5721e−07	3.99E−04	6.2828e−06
Case 2	0.0406	2.1540e−06	1.1674e−05	0.2643	0.7172
Model 2					
Case 1	5.5727e−10	0.4733	1.4643e−10	0.6308	3.1821e−04
Case 2	8.1014e−10	1.7290e−06	3.0199e−11	0.3790	1.6980e−08
Model 3					
Case 1	3.0199e−11	3.0199e−11	4.9752e−11	5.08E−03	0.0156
Case 2	5.0723e−10	3.0199e−11	3.6897e−11	2.68E−04	8.8411e−07

using the RWOA was slightly better than that of the PSO, but outperformed those of the other algorithms. The average MSE obtained using the RWOA was superior to that of the basic WOA and the other four algorithms. Figure 6 shows the convergence for the average MSE values acquired by the six algorithms in solving the second-order plant problem. Figure 6 shows that the RWOA outperformed the other algorithms because it had the fastest convergence to the global optimum among the other algorithms, followed by the GOA and PSO. However, Fig. 6 also shows that the WOA, BA, and GSA were easily trapped in local optima, which resulted in stagnation. The p values reported in Table 11 also indicate that the results of the RWOA for the model-1, case 1 IIR model were significantly better than those of the other algorithms. This is evidence that the RWOA had high performance in terms of solving the second-order IIR model.

Case 2 A reduced-order IIR filter can model the second-order plant. Hence, a first-order IIR model is used to model the second-order plant defined in Eq. (9). The transfer function is

$$H_{\mathrm{m}}(z) = \frac{a_0}{1 - b_1 z^{-1}}. \tag{11}$$

Table 4 shows that the RWOA provided the best average results and standard deviation in terms of the MSE, which were slightly superior to those of the basic PSO and outperformed those of the other algorithms in solving this IIR model. As shown in Fig. 7, the RWOA algorithm converged much faster and had the lowest MSE values compared with the other five algorithms. Furthermore, the p values in Table 11 also prove that the superiority of the RWOA was significant in three out of five cases.

Model 2 For the second experiment (Table 5), the transfer function of a third-order-plant is

$$H_{\mathrm{p}}(z) = \left[\frac{-0.2 - 0.4z^{-1} + 0.5z^{-2}}{1 - 0.6z^{-1} + 0.25z^{-2} - 0.2z^{-3}} \right]. \tag{12}$$

Case 1 This third-order plant can be defined as a third IIR filter given by the transfer function

$$H_{\mathrm{m}}(z) = \left[\frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2} - b_3 z^{-3}} \right]. \tag{13}$$

The results of the MSE obtained by the algorithms presented in Table 6 show that the GSA and RWOA had almost the same quality of solution, and these two algorithms performed much better than the other algorithms. Figure 8 shows that the RWOA converged much faster than the BA, PSO, GOA, and WOA, but was slightly worse than the GSA. Table 5 shows a list of values of the best estimated model parameters after 500 iterations for each algorithm. It shows that both the RWOA and GOA obtained estimated values that matched very well with the actual values, and proves that the RWOA and GOA performed better than the BA, GSA, PSO, and WOA in terms of estimating the values of the system. According to the p values presented in Table 11, the results of the RWOA were not significantly better than those of the GSA and GOA, but were better than those of the remaining three algorithms.

Case 2 In this section, a second-order IIR is used to model the third-order plant shown in Eq. (12) as follows:

$$H_{\mathrm{m}}(z) = \left[\frac{a_0 + a_1 z^{-1}}{1 - b_1 z^{-1} - b_2 z^{-2}} \right]. \tag{14}$$

The convergence for all six algorithms is compared in Fig. 9, which shows clearly that the minimum MSE acquired for the RWOA was approximately the same as that achieved by the GSA, and these two algorithms outperformed the other algorithms, followed by GOA. During the optimization process, the PSO and WOA acquired the same minimum MSE value, and the performance of BA was the worst of all the algorithms and stagnated in an early period. The comparison in terms of MSE is listed in Table 7, which shows clearly that the average MSE obtained by all the algorithms after 500 iterations was 70E−03, 3.94E−04, 1.40E−03,

9.31E − 04, 1.50E − 03, and 4.59E − 04 for the BA, GSA, PSO, GOA, WOA, and RWOA, respectively. These results show that the proposed RWOA provided very competitive results for this IIR case. The p values in Table 11 also provide evidence that the superiority of the RWOA was significant compared with all the other algorithms, except the GOA.

Model 3 For the third experiment, a fourth-order plant with a transfer function as shown in Eq. (16) is modeled to test the efficiency of the RWOA:

$$H_{\mathrm{p}}(z) = \left[\frac{1 - 0.9z^{-1} + 0.81z^{-2} - 0.729z^{-3}}{1 + 0.04z^{-1} + 0.2775z^{-2} - 0.2101z^{-3} + 0.14z^{-4}}\right].$$
(15)

Case 1 In this section, this fourth-order plant can be defined as a fourth-order IIR filter given by the transfer function

$$H_{\mathrm{m}}(z) = \left[\frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}}{1 - b_1 z^{-1} - b_2 z^{-2} - b_3 z^{-3} - b_4 z^{-4}}\right].$$
(16)

The results in Table 8 show that the RWOA estimated the coefficients more accurately than the WOA algorithm and the other four algorithms. Table 9 shows that the RWOA provided the best average results according to the four criteria "Best," "Worst," "Mean," and "Std," which were much smaller than those of the other algorithms. Figure 10 shows a qualitative measurement of the performance of all six algorithms in terms of the MSE values. It shows clearly that the BA, GSA, GOA, and PSO became trapped in local optima quite soon, and then their convergence rates remained flat over approximately 50 iterations. Although the basic WOA took approximately 200 iterations to achieve the corresponding lowest MSE, which was faster than the RWOA, which converged to the corresponding lowest MSE after 450 iterations, the minimum MSE for the RWOA was much smaller than that of the basic WOA. The p values shown in Table 10 for the IIR model-3, case 1 suggest that the RWOA was statistically significantly superior to the other five algorithms because all the p values in this case were less than 0.05. All of the results confirmed the superior ability of the RWOA in solving the fourth-order IIR filter problem.

Case 2 In this case, the fourth-order plant shown in Eq. (15) is modeled using a reduced order, which is a third-order IIR. The transfer function of this model is defined as

$$H_{\mathrm{m}}(z) = \left[\frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2} - b_3 z^{-3}}\right].$$
(17)

The comparison of the statistical results for all the algorithms is presented in Table 10. Similar to the previous experiment, the RWOA had the best average results and the smallest standard deviation in terms of MSE. The

convergence characteristics for reduced-order modeling are shown in Fig. 11. The figure shows that the WOA, BA, GSA, GOA, and PSO were trapped in a local minimum very early and result in premature. However, the RWOA did not stagnate and continued to search for its corresponding minimum MSE during the optimization process. In this case, all the results show that the RWOA outperformed the other algorithms with the fastest convergence speed and smallest MSE value. Moreover, the p values in Table 11 provide strong evidence that the results of the RWOA for the IIR model-3, case 2 were significantly better than those of all the other algorithms.

5.2 Discussion

In this section, we discuss our incorporation of a ranking-based mutation operator into the basic WOA method before whales update their position, which we called RWOA. The RWOA was applied to three sets of IIR system identification. From the experimental results, we found that our proposed RWOA approach was effective and efficient. It provided the best results compared with other algorithms for solving the IIR model identification problem in four cases, whereas in the other two cases, it was slightly worse than the GSA algorithm, but still much better than the remaining algorithms. The well-known NFL proves that no single heuristic algorithm exists that is best for dealing with all optimization problems and outperforms all other algorithms [13]; therefore, it is reasonable that the RWOA performed well in the majority of cases, but not all.

The reasons that the performance of our proposed RWOA was superior to all the other compared algorithms, including the basic WOA, for solving different IIR models is attributed to the ranking-based mutation operator because this mechanism helped to improve the probability of choosing good solutions, and then they had more of a chance to propagate to the offspring. Thus, the exploitation ability of the basic WOA was enhanced and the entire performance also improved. Additionally, the RWOA always saved the best obtained solution during the optimization process; hence, the search agent could update its position according to the best obtained solution (the best hunting agent). Therefore, guiding search agents always existed for the exploitation of the most promising regions in the search space. Moreover, because of the value of parameter A, which helped the RWOA to conduct exploration and exploitation simultaneously, when $|A| > 1$, half the iterations were forced to perform exploration to search for a more promising space. By contrast, when $|A| < 1$, the remaining iterations were devoted to exploitation. This mechanism helped the RWOA to be more flexible, and it had a superior ability to avoid local optima, which was helpful for obtaining a tradeoff between the exploration and exploitation ability of the RWOA.

Finally, the RWOA maintained the simple framework of the basic WOA, and it had fewer parameters to control than other algorithms, such as the BA, GSA, PSO, and GOA, during the optimization process. A good combination of the above four components thus lead to the RWOA being suitable to real applications and efficiently solving the different IIR models.

It is also worth discussing the poor performance of the BA, GSA, PSO, and GOA in this subsection. These four algorithms belong to the class of swarm-based algorithms. In contrast to evolutionary algorithms, they have no mechanism for significant abrupt movements in the search space, and this is likely to be the reason for the poor performance of the BA, GSA, PSO, and GOA. Although the RWOA is also a swarm-based algorithm, its mechanisms, summarized in the previous paragraph, are the main reasons why it is advantageous in terms of IIR system identification.

The comprehensive study conducted in this paper clearly demonstrates that the RWOA has the ability to avoid local optima and obtain the global optimal solution, which makes it suitable for IIR system identification. All the statistical results proved the superiority of the RWOA.

6 Conclusions and Future Work

In this paper, we proposed a modified version of the WOA called the RWOA, in which a ranking-based mutation operator was embedded into the standard WOA to speed up the rate of convergence and then enhance the algorithm's exploitation capability. This new approach was applied to the identification of three sets of benchmark IIR plants. The performance assessment of the RWOA for different unknown identification systems with different corresponding reduced models was also presented in this paper. Additionally, the Wilcoxon rank-sum nonparametric statistical test was conducted at a 5% significance level to assess whether the results of the RWOA differed from the best results of the other algorithms in a statistically significant manner. The comparison efficiency for parameter identification achieved using the RWOA and the other five state-of-the-art evolutionary algorithms, including the basic WOA, and the simulation results clearly demonstrated that the RWOA exhibited superior identification performance in terms of convergence speed, accuracy (average MSE), and robustness (Std values of MSE) within a statistically significant framework (the Wilcoxon test). Incorporating a ranking-based mutation operator into the pure WOA helped to improve identification accuracy because it was useful to allow good solutions to have high selection probabilities so that they had more opportunities to propagate to the next generation.

Our future work will consider the following two issues: First, we will apply our RWOA to solve more practical engineering optimization problems. Additionally, we will develop other swarm-based algorithms to deal with IIR system identification.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Krusienski, D.J.; Jenkins, W.K.: Adaptive filtering via particle swarm optimization. In: Conference Record of the 37th Asilomar Conference on Signals, Systems and Computers, 2004, vol. 1, pp. 571–575. IEEE (2003)
2. Krusienski, D.J.; Jenkins, W.K.: Particle swarm optimization for adaptive IIR filter structures. In: Congress on Evolutionary Computation, 2004. CEC2004, vol. 1, pp. 965–970. IEEE (2004)
3. Zhou, X.; Yang, C.; Gui, W.: Nonlinear system identification and control using state transition algorithm. Appl. Math. Comput. **226**(1), 169–179 (2012)
4. Albaghdadi, M.; Briley, B.; Evens, M.: Event storm detection and identification in communication systems. Reliab. Eng. Syst. Saf. **91**(5), 602–613 (2006)
5. Pai, P.F.; Nguyen, B.A.; Sundaresan, M.J.: Nonlinearity identification by time-domain-only signal processing. Int. J. Non-Linear Mech. **54**(3), 85–98 (2013)
6. Chung, H.C.; et al.: Digital image processing for non-linear system identification. Int. J. Non-Linear Mech. **39**(5), 691–707 (2004)
7. Zou, D.X.; Deb, S.; Wang, G.G.: Solving IIR system identification by a variant of particle swarm optimization. Neural Comput. Appl. **30**, 1–14 (2016)
8. Ma, Q.; Cowan, C.F.N.: Genetic algorithms applied to the adaptation of IIR filters. Signal Process. **48**(2), 155–163 (1996)
9. Patwardhan, A.P.; Patidar, R.; George, N.V.: On a cuckoo search optimization approach towards feedback system identification. Dig. Signal Process. **32**(2), 156–163 (2014)
10. Panda, G.; Pradhan, P.M.; Majhi, B.: IIR system identification using cat swarm optimization. Expert Syst. Appl. **38**(10), 12671–12683 (2011)
11. Saha, S.K.; et al.: Gravitation search algorithm: application to the optimal IIR filter design. J. King Saud Univ. Eng. Sci. **26**(1), 69–81 (2014)
12. Karaboga, N.: A new design method based on artificial bee colony algorithm for digital IIR filters. J. Franklin Inst. **346**(4), 328–348 (2009)
13. Wolpert, D.H.; Macready, W.G.: No free lunch theorems for optimization. IEEE Trans. Evolut. Comput. **1**(1), 67–82 (1997)
14. Mirjalili, S.; Lewis, A.: The whale optimization algorithm. Adv. Eng. Soft. **95**, 51–67 (2016)
15. Jangir, P.; et al.: Training multi-layer perceptron in neural network using whale optimization algorithm. Indian J. Sci. Technol. **9**, 1–15 (2016)

16. Prakash, D.B.; Lakshminarayana, C.: Optimal siting of capacitors in radial distribution network using whale optimization algorithm. Alex. Eng. J. **56**, 499–509 (2016)

17. Aljarah, I.; Faris, H.; Mirjalili, S.: Optimizing connection weights in neural networks using the whale optimization algorithm. Soft Comput. **22**, 1–15 (2016)

18. Touma, H.J.: Study of the economic dispatch problem on IEEE 30-bus system using whale optimization algorithm. Int. J. Eng. Technol. Sci. **5**(1), 11–18 (2016)

19. Horng, M.F., Dao, T.K., Shieh, C.S., Nguyen, T.T.: A multi-objective optimal vehicle fuel consumption based on whale optimization algorithm. In: Pan, J.S., Tsai, P.W., Huang, H.C. (eds.) Advances in Intelligent Information Hiding and Multimedia Signal Processing. Smart Innovation, Systems and Technologies, vol. 64. Springer, Cham (2017)

20. Reddy, P.D.P.; Reddy, V.C.V.; Manohar, T.G.: Whale optimization algorithm for optimal sizing of renewable resources for loss reduction in distribution systems. Renew. Wind Water Solar **4**, 3 (2017)

21. Oliva, D.; Aziz, M.A.E.; Hassanien, A.E.: Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm. Appl. Energy **200**, 141–154 (2017)

22. Hongping, H.; Yanping, B.; Ting, X.: Improved whale optimization algorithms based on inertia weights and theirs applications. Int. J. Circuits Syst. Signal Process. **11**, 12–26 (2017)

23. Trivedi, I.N.; et al.: A novel hybrid PSO-WOA algorithm for global numerical functions optimization. In: ICCCCS (2016)

24. Jangir, P.; et al.: A novel adaptive whale optimization algorithm for global optimization. Indian J. Sci. Technol. **9**(38), 1–6 (2016)

25. Kaveh, A.; Ghazaan, M.I.: Enhanced whale optimization algorithm for sizing optimization of skeletal structures. Mech. Based Design Struct. Mach. **45**, 1–18 (2016)

26. Mafarja, M.M.; Mirjalili, S.: Hybrid whale optimization algorithm with simulated annealing for feature selection. Neurocomputing **260**, 302–312 (2017)

27. Upadhyay, P.; et al.: Craziness based particle swarm optimization algorithm for IIR system identification problem. AEU Int. J. Electron. Commun. **68**(5), 369–378 (2014)

28. Proakis, J.G.; Manolakis, D.G.: Digital Signal Processing: Principles, Algorithms, and Applications, pp. 392–394. Prentice-Hall Inc., Upper Saddle River (1992)

29. Gong, W.; Cai, Z.: Differential evolution with ranking-based mutation operators. IEEE Trans. Cybern. **43**(6), 2066 (2013)

30. Gong, W.; Cai, Z.; Liang, D.: Engineering optimization by means of an improved constrained differential evolution. Comput. Methods Appl. Mech. Eng. **268**(4), 884–904 (2014)

31. Kennedy, J.; Eberhart, R.: Particle swarm optimization. In: IEEE International Conference on Neural Networks, 1995. Proceedings, vol. 4, pp. 1942–1948. IEEE Xplore (1995)

32. Yang, Xin She: A new metaheuristic bat-inspired algorithm. Comput. Knowl. Technol. **284**, 65–74 (2010)

33. Rashedi, E.; Nezamabadi-Pour, H.; Saryazdi, S.: GSA: a gravitational search algorithm. Inf. Sci. **179**(13), 2232–2248 (2009)

34. Saremi, S.; Mirjalili, S.; Lewis, A.: Grasshopper optimisation algorithm: theory and application. Adv. Eng. Softw. **105**, 30–47 (2017)

35. Wilcoxon, Frank: Individual comparisons by ranking methods. Biom Bull. **1**(6), 80–83 (1944)

36. Derrac, J.; et al.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evolut. Comput. **1**(1), 3–18 (2011)

37. Mirjalili, S.; Lewis, A.: S-shaped versus V-shaped transfer functions for binary particle swarm optimization. Swarm Evolut. Comput. **9**, 1–14 (2013)