Adaptive infinite impulse response system identification using an enhanced golden jackal optimization

Jinzhong Zhang[1] · Gang Zhang[1] · Min Kong[1] · Tan Zhang[1]

Abstract

Golden jackal optimization (GJO) is inspired by the cooperative attacking behavior of golden jackals and mainly simulates searching for prey, stalking and enclosing prey, and pouncing on prey to solve complicated optimization problems. However, the basic GJO has the disadvantages of premature convergence, a slow convergence rate and low computation precision. To enhance the overall search and optimization abilities, an enhanced golden jackal optimization (EGJO) method with the elite opposition-based learning technique and the simplex technique is proposed to address adaptive infinite impulse response system identification. The intention is to minimize the error fitness value and obtain the appropriate control parameters. The elite opposition-based learning technique boosts population diversity, enhances the exploration ability, extends the search range and avoids search stagnation. The simplex technique accelerates the search process, enhances the exploitation ability, improves the computational precision and increases the optimization depth. EGJO can not only achieve complementary advantages to avoid search stagnation but also balance exploration and exploitation to arrive at the best value. Three sets of experiments are used to verify the effectiveness and feasibility of EGJO. The experimental results clearly demonstrate that the optimization efficiency and recognition accuracy of EGJO are superior to those of AOA, GTO, HHO, MDWA, RSO, WOA, TSA and GJO. EGJO has a faster convergence rate, higher computation precision, better control parameters and better fitness value, and it is stable and resilient in solving the IIR system identification problem.

Keywords Golden jackal optimization · Elite opposition-based learning technique · Simplex technique · Infinite impulse response · Experimental results

✉ Gang Zhang
  zhanggang@wxc.edu.cn

1   School of Electrical and Optoelectronic Engineering, West Anhui University, Lu'an 237012, China

1 Introduction

The adaptive IIR filter has higher computational accuracy and lower design order to achieve good frequency selection characteristics, and it has been extensively used in control systems, network communication, signal processing, geological exploration, biomedicine and so on. In IIR model identification, the design is essentially a global optimization problem of multidimensional variables, and the error surface contains some local extrema. Modeling is accomplished by comparing the input value of the unknown system with the output value of the adaptive IIR filter [1–5]. Some optimization techniques have been introduced to adjust the control parameters and enhance the overall performance of model identification, such as the arithmetic optimization algorithm (AOA) [6], gorilla troop optimization (GTO) [7], Harris hawks optimization (HHO) [8], movable damped wave algorithm (MDWA) [9], rat swarm optimization (RSO) [10], whale optimization algorithm (WOA) [11], and tunicate swarm algorithm (TSA) [12].

Niu et al. [13] suggested an enhanced artificial ecosystem optimization algorithm to address IIR system identification, and the algorithm's recognition accuracy was higher than those of other approaches. Ababneh et al. [14] found a unique cuckoo search algorithm to optimize the IIR filter, and the proposed algorithm had great robustness and stability in identifying the best result. Mohammadi et al. [15] applied a revised inclined plane system optimization method to design the IIR filter's parameters, and this algorithm attained rapid convergence efficiency and excellent optimization accuracy. Chang [16] created a new Hammerstein model based on the IIR system and Volterra neural network to identify nonlinear discrete systems, and the algorithm verified the validity and reliability of the model. Durmuş [17] used a modified average differential evolution algorithm to accomplish IIR system identification, and this method had extensive optimization ability to generate better control parameters. Kowalczyk et al. [18] published a new approach based on an IIR filter matrix to design hardware architecture, and the approach found a quicker optimization frequency and superior calculation precision. Mittal [19] tried to combine moth flame optimization with the variable neighborhood search technique to optimize the IIR filter's parameters, and this hybrid method produced better results than comparison algorithms. Ababneh et al. [20] created a differential evolution optimization method to construct IIR filters, and the results indicated that this method had strong utilization and exploration to determine the best resolution. Mohammadi et al. [21] devised an enhanced particle swarm optimization method for IIR filters, and this approach identified the best solution and good optimization parameters. Liang et al. [22] adopted a modified slime mold algorithm to perform IIR filter design, and this algorithm had powerful stability in discovering the most suitable value. Agrawal et al. [23] studied how to create digital IIR filters according to various techniques, and the results confirmed the robustness and effectiveness of the methods. Bui et al. [24] utilized the design of an electrocardiogram signal for the IIR filter, and the proposed method had certain stability and superiority. Singh et al. [25] established a teacher-learner-based optimization method to perform adaptive IIR system identification, and the optimization outcomes of this method were better. Kumar et al. [26]

integrated the interior search algorithm with the Lèvy technique to design adaptive IIR filters and proposed an extraction and utilization method to discover the best parameters and solution. Luo et al. [27] devised an updated whale optimization algorithm to investigate IIR system identification, and this approach produced superior overall results. Chang et al. [28] constructed a revised particle swarm optimization method based on multiple subpopulations to handle IIR filters, and this algorithm had a relatively strong detection performance in identifying the best solution. Zhao et al. [29] established a modified selfish herd optimization method for IIR filters, and this approach had better calculation accuracy and strong durability. Ali et al. [30] devised a salp swarm algorithm for IIR filters, and this method obtained better parameters and the best solution. Dhabal et al. [31] offered an enhanced global-best-guided cuckoo search algorithm to construct IIR filters, and the improved algorithm exhibited a better convergence efficiency and calculation precision. Cuevas et al. [32] summarized some optimization methods for IIR system identification. Mohammadi et al. [33] utilized evolutionary algorithms to design IIR filters, and these algorithms had strong robustness and stability in obtaining the best value. Mohammadi et al. [34] proposed an inclined plane system optimization algorithm to design an IIR filter, and this method had a strong search ability to obtain the global optimal solution. Mohammadi et al. [35] later applied another modified inclined plane system optimization algorithm for the IIR filter, and this proposed method had strong optimization efficiency and high calculation accuracy. Mohammadi et al. [36] analyzed swarm intelligence and evolutionary computation techniques to design IIR filters, and the reliability and accuracy of this swarm intelligence method were superior to those of evolutionary computation. Mohammadi et al. [37] described an inclined plane system optimization algorithm that had strong robustness and reliability and was widely used in various fields. Agrawal et al. [38] used the fractional derivative to design a new IIR filter, and this method was effective. Agrawal et al. [39] presented a novel approach based on the fractional derivative and swarm intelligence to design IIR filters, and the proposed approach had good optimization performance. Kumar et al. [40] combined fractional derivatives and swarm optimization to perform the Hilbert transform, and this method balanced exploration and exploitation to achieve higher computation precision. Agrawal et al. [41] employed fractional derivative constraints and hybrid particle swarm optimization to design an IIR filter, and this method had strong stability and reliability. Janjanam et al. [42] proposed a global gravitational search algorithm-assisted Kalman filter to evaluate nonlinear system identification, and this method had better optimization results. Saha et al. [43] utilized the harmony search algorithm to design IIR filters, and this algorithm had strong global and local abilities to determine the best value. Ahirwal et al. [44] integrated the cuckoo optimization algorithm and other optimization methods to design the EEG/ERP adaptive noise canceller, and the proposed method had a high computational efficiency and convergence accuracy. Ahirwal et al. [45] combined the evolutionary method with the adaptive noise canceller to enhance the SNR of contaminated EEG, and this method had strong stability and robustness. Ahirwal et al. [46] proposed the bounded-range artificial bee colony algorithm to resolve adaptive filtering of EEG/ERP, and the method achieved certain superiority and reliability.

GJO is motivated by the cooperative attacking behavior of golden jackals; it mainly imitates searching for prey, stalking and enclosing prey, and pouncing on prey to identify the best solution [47]. The basic GJO has the disadvantages of premature convergence, a slow convergence rate and low computation precision. The elite opposition-based learning technique and simplex technique are introduced into the basic GJO to enhance the convergence rate and computation precision. The elite opposition-based learning technique increases population variety and broadens the search region to avoid search stagnation and boost global search [48, 49]. The simplex technique accelerates the optimization process and increases the search depth to optimize the computational precision and boost local search [50, 51]. EGJO is applied to perform IIR system identification, and the intention is to minimize the error fitness value and obtain the appropriate control parameters. The experimental results demonstrate that EGJO not only stabilizes exploration and exploitation to avoid falling into a locally optimal value but also has excellent durability and robustness to arrive at the best solution. Additionally, EGJO has a faster convergence rate, greater computation precision, and stronger stability and robustness; the effectiveness and feasibility of EGJO have been verified.

This paper is arranged as follows. Section 2 defines the adaptive IIR system identification problem. Section 3 outlines GJO. Section 4 explores EGJO. In Sect. 5, the EGJO-based adaptive IIR system identification method is presented. The experimental results and analysis are presented in Sect. 6. Finally, conclusions and future research are addressed in Sect. 7.

2 Adaptive IIR system identification problem

The adaptive IIR filter utilizes a lower order and coefficients to ensure the overall control level of the system; it has found broad potential applications in model identification, noise elimination, prediction spectrum optimization and automatic equalization. System identification is a significant and constructive approach for performing mathematical modeling of an unidentified plant by anatomizing the input and output data. Some problems of signal processing are usually regarded as system identification problems. Some swarm intelligence optimization algorithms are introduced to tune the parameters of the IIR filter and minimize the error value of the actual system and model system, which enhances the optimization ability and search ability to satisfy various performance indicators and discover an accurate solution. A schematic of adaptive IIR system identification employing EGJO is shown in Fig. 1.

In adaptive IIR system identification, the following mathematical function is computed:

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

where $x(n)$ and $y(n)$ represent the input and output, L and K are the orders of the feedforward and feedback IIR filters, L is greater than K, a_l is the pole coefficient, and b_k is the zero coefficient. The transfer function is computed as:
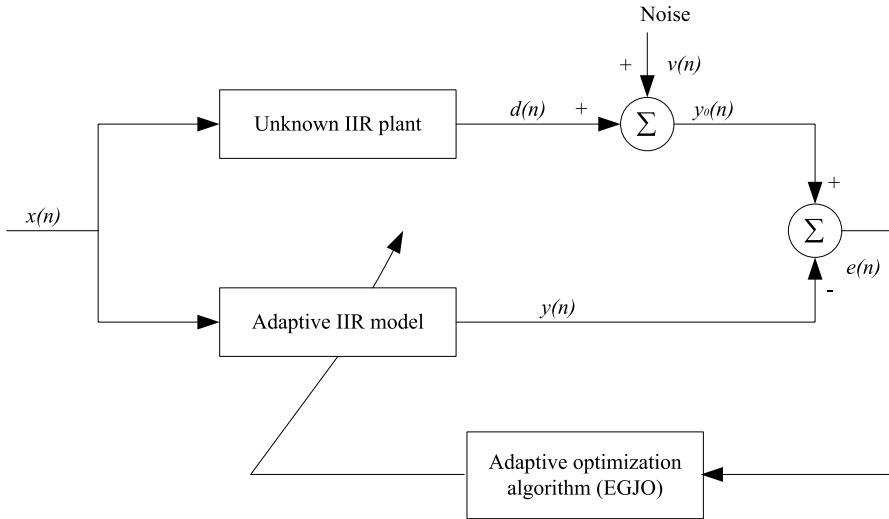
Fig. 1 Schematic of adaptive IIR system identification employing EGJO

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

In the IIR filter, $y_0(n) = d(n) + v(n)$ is the response of the unidentified IIR plant, $y_0(n)$ represents the output data of an unidentified IIR plant, $v(n)$ is the additional white Gaussian noise, and $e(n) = y_0(n) - y(n)$ is the error between the authentic plant and the IIR model. Adaptive system identification is transformed into mathematical optimization. The mean square error (MSE) is computed as:

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

where N expresses the entire quantity of specimens, w expresses the coefficient vector, and $w = (a_1, a_2, \ldots, a_L, b_0, b_1, \ldots, b_K)^T$.

3 GJO

The basic GJO method inspired by the cooperative attacking behavior of golden jackals is a novel optimization technique. In GJO, each golden jackal represents a candidate solution or search agent. The cooperative attacking behavior of a
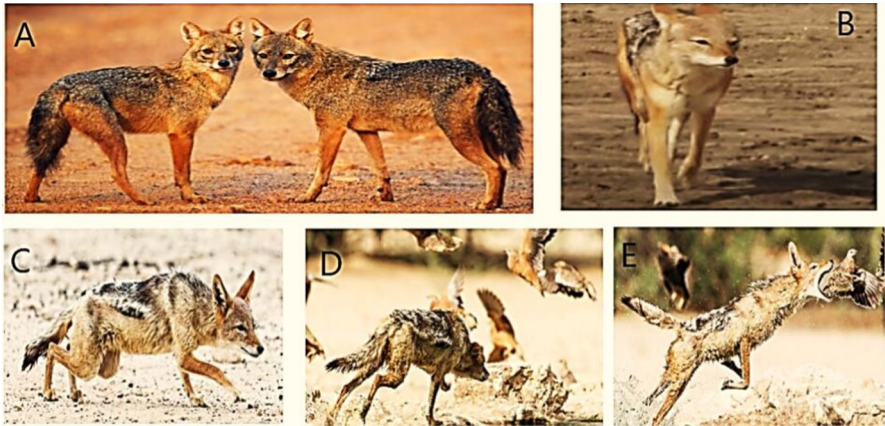
Fig. 2 **A** Pair of golden jackals, **B** golden jackal searching for prey, **C** stalking and enclosing of prey, **D** and **E** pouncing on prey

Table 1 Correspondence between the problem region and GJO region

Problem region	GJO region
Search region	Attack region
A solution to the issue	A golden jackal
Each solution's evaluation function	GJO's fitness function

golden jackal pair is illustrated in Fig. 2. The correspondence between the problem region and GJO region is described in Table 1.

3.1 Search space formulation

In GJO, we initialize the random prey population to obtain the uniformly distributed candidate solutions in the search region. The initial solution is computed as:

$$Y_0 = Y_{\min} + \text{rand}(Y_{\max} - Y_{\min}) \tag{4}$$

where Y_0 expresses the positions of the initial golden jackal population, rand expresses a uniform arbitrary vector in [0,1], and $Y_{\min}$ and $Y_{\max}$ express the lower and upper limits of the solution, respectively.

The first and second fittest individuals are a jackal pair, and the initial matrix individual is computed as:

$$\text{Prey} = \begin{bmatrix} Y_{1,1} & Y_{1,2} & \cdots & Y_{1,d} \\ Y_{2,1} & Y_{2,2} & \cdots & Y_{2,d} \\ \vdots & \vdots & \vdots & \vdots \\ Y_{n,1} & Y_{n,2} & \cdots & Y_{n,d} \end{bmatrix} \tag{5}$$

where $Y_{i,j}$ expresses the jth dimension of the ith prey, n is the entire quantity of prey, and d represents the variables of the situation. The matrix is computed as:

$$F_{OA} = \begin{bmatrix} f(Y_{1,1};Y_{1,2};Y_{1,d}) \\ f(Y_{2,1};Y_{2,2};Y_{2,d}) \\ \vdots \\ f(Y_{n,1};Y_{n,2};Y_{n,d}) \end{bmatrix} \tag{6}$$

where F_{OA} is a matrix that includes the fitness values of the prey and f expresses the fitness function. The fittest individual is interpreted as a male jackal, and the second fittest is interpreted as a female jackal. The jackal pair represents a single individual position.

3.2 Exploration phase or searching for prey

Golden jackals can predict and capture prey according to their own attacking characteristics, but the prey occasionally quickly evades and escapes the foraging jackals. Hence, the female jackals follow the male jackals to wait and search for other prey in the search region. The positions are computed as:

$$Y_1(t) = Y_M(t) - E \cdot |Y_M(t) - rl \cdot \text{Prey}(t)| \tag{7}$$

$$Y_2(t) = Y_{FM}(t) - E \cdot |Y_{FM}(t) - rl \cdot \text{Prey}(t)| \tag{8}$$

where t is the current iteration, Prey(t) is the position vector, and $Y_M(t)$ and $Y_{FM}(t)$ are the current positions of the male and female jackals, respectively. $Y_1(t)$ and $Y_2(t)$ are the updated positions of the male and female jackals.

The evading energy of prey E is computed as:

$$E = E_1 \cdot E_0 \tag{9}$$

where E_1 expresses the diminishing energy of the prey and E_0 is the original state of the energy.

$$E_0 = 2 \cdot r - 1 \tag{10}$$

where r is an arbitrary value in [0,1].

$$E_1 = c_1 \cdot (1 - (t/T)) \tag{11}$$

where T is the maximum number of iterations, c_1 is a constant with a value of 1.5, and E_1 linearly decreases from 1.5 to 0 as iteration proceeds.

rl is an arbitrary vector based on the Lévy distribution, which is computed as:

$$rl = 0.05 \cdot \mathrm{LF}(y) \tag{12}$$

LF is the fitness function of the Levy flight, which is computed as:

$$\mathrm{LF}(y) = 0.01 \times (\mu \times \sigma) \Big/ \left(\left| v^{(1/\beta)} \right| \right); \quad \sigma = \left(\frac{\Gamma(1+\beta) \times \sin(\pi\beta/2)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times \left(2^{\frac{\beta-1}{2}}\right)} \right)^{1/\beta} \tag{13}$$

where μ and v are arbitrary values in (0,1) and β is a constant with a value of 1.5.

The updated positions of the golden jackals are computed as:

$$Y(t+1) = \frac{Y_1(t) + Y_2(t)}{2} \tag{14}$$

3.3 Exploitation phase or enclosing and pouncing on prey

The evading energy of the prey will swiftly decrease when it is attacked by a jackal pair, and the golden jackals instantly enclose and capture the prey. The positions are computed as:

$$Y_1(t) = Y_{\mathrm{M}}(t) - E \cdot \left| rl \cdot Y_{\mathrm{M}}(t) - \mathrm{Prey}(t) \right| \tag{15}$$

$$Y_2(t) = Y_{\mathrm{FM}}(t) - E \cdot \left| rl \cdot Y_{\mathrm{FM}}(t) - \mathrm{Prey}(t) \right| \tag{16}$$

where t is the current iteration, $\mathrm{Prey}(t)$ is the position vector, and $Y_{\mathrm{M}}(t)$ and $Y_{\mathrm{FM}}(t)$ are the current positions of the male and female jackals, respectively. $Y_1(t)$ and $Y_2(t)$ are the refreshed positions of the male and female jackals. Some parameters have been provided in the previous section, such as E and rl. Finally, the position update of the golden jackal is computed by Formula (17).

To better represent the optimization process, the pseudocode of GJO is depicted in Algorithm 1.

Algorithm 1 GJO

Begin

Step 1. Initialize the random prey population $Y_i (i = 1, 2, ..., n)$

Step 2. Estimate the fitness value of each prey

      Display the best prey (male jackal position) Y_1, and the second best prey (female jackal position) Y_2

Step 3. while ($t < T$) do

      for each prey

      Refresh the evading energy E via Eqs. (9), (10) and (11)

      Refresh the arbitrary vector rl via Eqs. (12) and (13)

        if ($|E| \geq 1$)

          Refresh the prey position via Eqs. (7), (8) and (14) based on the exploration phase

        else ($|E| < 1$)

          Refresh the prey position via Eqs. (14)- (16) based on the exploitation phase

        end if

      end for

      Check whether any prey have strayed outside the search region and modify their positions accordingly

      Estimate the fitness value of each prey

      Refresh Y_1 if there is a superior solution

      $t = t + 1$

    end while

    restore Y_1

End

---

4 EGJO

To address the limitations of weak optimization efficiency and slow search speed, the elite opposition-based learning technique and the simplex technique are incorporated into the basic GJO to achieve supplementary superiority and avoid search stagnation, which increases the convergence rate and enhances the computation precision.

4.1 Elite opposition-based learning technique

The elite opposition-based learning technique is an efficacious and stable approach that increases the population variety, broadens the search region, avoids premature convergence and strengthens the global search. The search technique

utilizes the feasible or reverse solution to assess the fitness value of prey and then sorts out the best individual to complete the iteration. Assuming that the search agent with the optimal fitness value is regarded as an elite individual, the elite individual is computed as $x_e = (x_{e,1}, x_{e,2}, \ldots, x_{e,D})$, the feasible solution is computed as $x_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,D})$, and the reverse solution is computed as $x'_i = (x'_{i,1}, x'_{i,2}, \ldots, x'_{i,D})$. The formula is:

$$x'_{i,j} = k \cdot (da_j + db_j) - x_{e,j}, \ i = 1, 2, \ldots, n; j = 1, 2, \ldots, D \tag{17}$$

where n expresses the population size, D expresses the problem dimension, k expresses an arbitrary value such that $k \in (0, 1)$, and da_j and db_j express the dynamic limits of the jth decision variable and are computed as:

$$da_j = \min(x_{i,j}), \ db_j = \max(x_{i,j}) \tag{18}$$

The dynamic limit can save the best solution and adjust the search region of the inverse solution. The search agent $x'_{i,j}$ is computed as:

$$x'_{i,j} = \text{rand}(da_j, db_j), \ \text{if} \ x'_{i,j} < da_j \ \text{or} \ x'_{i,j} > db_j \tag{19}$$

4.2 Simplex technique

The simplex technique is a significant and practical method that accelerates the search process, improves the computational precision, increases the optimization depth and strengthens the local search. The simplex technique retains the best solution by evaluating the feasible solution with respect to the initial solution. The simplex technique schematic is illustrated in Fig. 3, and the optimization stages are computed as follows:
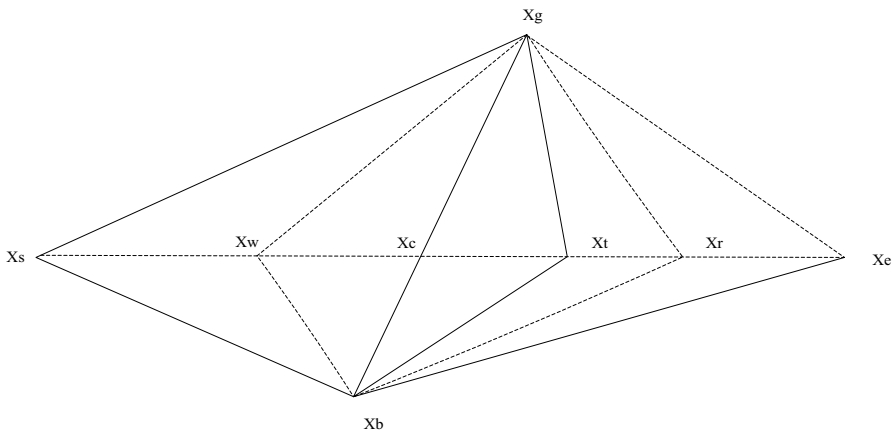

Fig. 3 Simplex technique schematic

Step 1 Estimate the fitness value of each prey, where X_g expresses the optimal value, X_b expresses the suboptimal value and X_s expresses the refreshed value; then, evaluate the fitness values $f(X_g)$, $f(X_b)$ and $f(X_s)$.

Step 2 Estimate the central point X_c between X_g and X_b.

$$X_c = \frac{X_g + X_b}{2} \tag{20}$$

Step 3 Estimate and obtain the reflection point X_r based on X_c and X_s, and evaluate the fitness value $f(X_r)$.

$$X_r = X_c + \alpha(X_c - X_s) \tag{21}$$

where α expresses reflectivity with a value of 1.

Step 4 If $f(X_g) > f(X_r)$, extend the current solution.

$$X_e = X_c + \gamma(X_r - X_c) \tag{22}$$

where X_e expresses the extension point and γ expresses the extension coefficient with a value of 1.5. Then, evaluate the fitness value $f(X_e)$. If $f(X_g) > f(X_e)$, modify X_s with X_e; otherwise, modify X_s with X_r.

Step 5 If $f(X_s) < f(X_r)$, perform a compression operation to obtain the best solution.

$$X_t = X_c + \beta(X_s - X_c) \tag{23}$$

where X_t expresses the compression point and β expresses the compression coefficient with a value of 0.5. If $f(X_s) > f(X_t)$, modify X_s with X_t; otherwise, modify X_s with X_r.

Step 6 If $f(X_s) > f(X_r) > f(X_g)$, a further compression operation is required to obtain a contraction point X_w.

$$X_w = X_c - \beta(X_s - X_c) \tag{24}$$

where β expresses the contraction coefficient with a value of 0.5. If $f(X_s) > f(X_w)$, modify X_s with X_w; otherwise, modify X_s with X_r.

EGJO not only achieves supplementary superiority to enhance the breadth and depth of the search but also arbitrarily switches between exploration and exploitation to yield a greater computation accuracy. The pseudocode of EGJO is shown in Algorithm 2.

Algorithm 2 EGJO

Begin

Step 1. Initialize the random prey population $Y_i(i = 1, 2, ..., n)$

Step 2. Estimate the fitness value of each prey

      Display the best prey (male jackal position) Y_1 and the second best prey (female jackal position) Y_2

Step 3. while ($t < T$) do

      for each prey

      Refresh the evading energy E via Eqs. (9), (10) and (11)

      Refresh the arbitrary vector rl via Eqs. (12) and (13)

      The elite opposition-based learning technique and the simplex technique are introduced into the exploration phase and exploitation phase of the basic GJO.

          if $(|E| \geq 1)$

            Refresh the prey position via Eqs. (7), (8) and (14) based on the exploration phase

          else $(|E| < 1)$

            Refresh the prey position via Eqs. (14)-(16) based on the exploitation phase

          end if

      end for

      Check whether any prey have strayed outside the search region and modify their positions accordingly

      Estimate the fitness value of each prey

      Refresh Y_1 if there is a superior solution

      $t = t + 1$

      end while

      restore Y_1

End

---

5 EGJO-based adaptive IIR system identification

EGJO effectively utilizes the three search mechanisms to handle IIR system identification, and the purpose is to minimize the error fitness value and obtain the best control parameters. The correspondence between the IIR system identification region and EGJO region is described in Table 2. EGJO based on adaptive IIR system identification is depicted in Algorithm 3. The flowchart of EGJO for adaptive IIR system identification is illustrated in Fig. 4.

Table 2 Correspondence between the IIR system identification region and EGJO region

IIR system identification region	EGJO region
A set of all schemes $(n_1, n_2, ..., n_k)$ to perform adaptive IIR system identification	A golden jackal population X with $(x_1, x_2, \ldots, x_k)$
The best scheme that obtains the optimal solution	The best golden jackal
The evaluation function of IIR system identification	The fitness function of EGJO

Algorithm 3 EGJO based on adaptive IIR system identification

Begin

Step 1. Initialize the random prey population $Y_i (i = 1, 2, ..., n)$

Step 2. Estimate the initial error fitness value of each prey via Eq. (3)

Display the best prey (male jackal position) Y_1 and the second best prey (female jackal position) Y_2

Step 3. while ($t < T$) do

    for each prey

    Refresh the evading energy E via Eqs. (9), (10) and (11)

    Refresh the arbitrary vector rl via Eqs. (12) and (13)

    The elite opposition-based learning technique and the simplex technique are introduced into the exploration phase and exploitation phase of the basic GJO.

        if $(|E| \geq 1)$

        Refresh the prey position via Eqs. (7), (8) and (14) based on the exploration phase

        else $(|E| < 1)$

        Refresh the prey position via Eqs. (14)-(16) based on the exploitation phase

        end if

    end for

    Check whether any prey have strayed outside the search region and modify their positions accordingly

    Estimate the error fitness value of each prey via Eq. (3)

    Refresh Y_1 if there is a superior solution

    $t = t + 1$

    end while

    restore Y_1

End

The computational complexity of each algorithm is considered to be the evaluation function that best links the input value with the method's run-time. Big-O notation is an effectual metric to analyze the optimization performance and evaluate computational complexity. The elite opposition-based learning technique and the simplex technique are added to the basic GJO to improve stability. EGJO is based
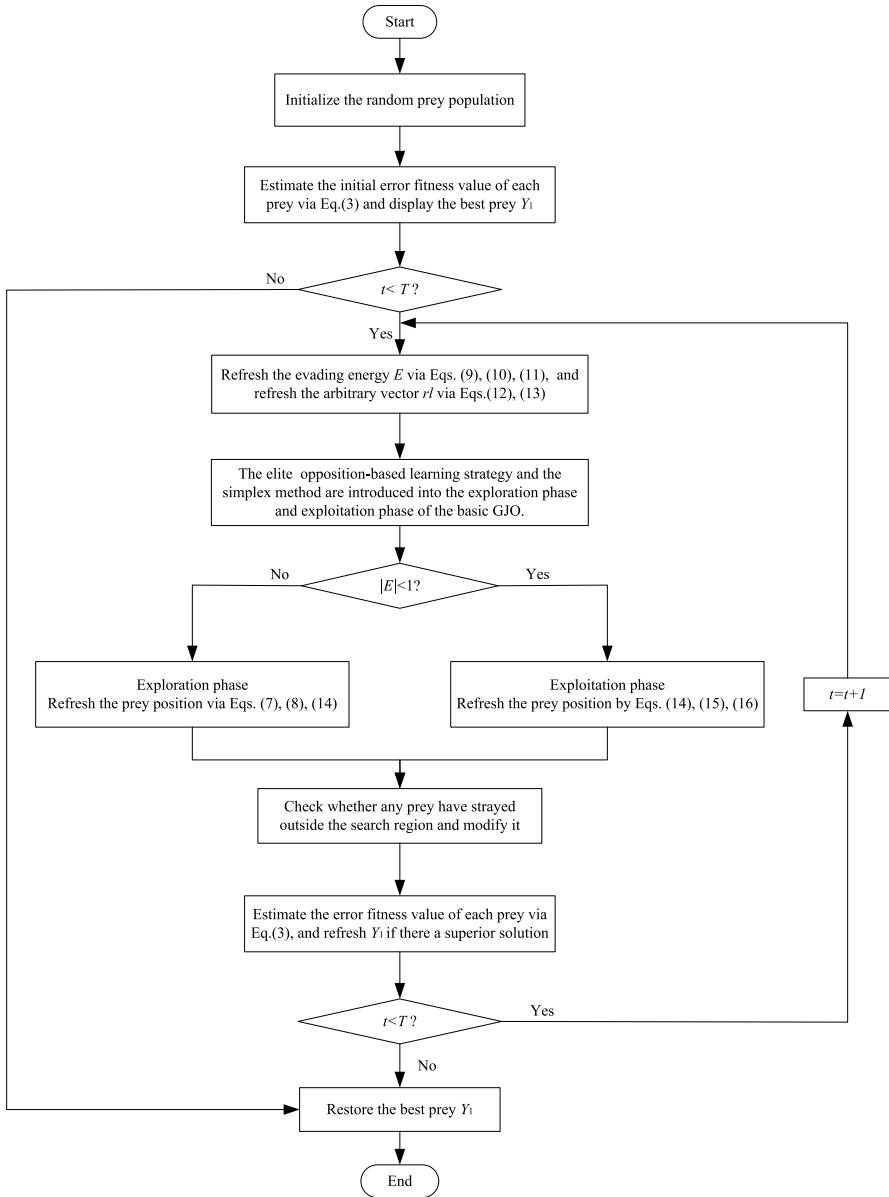
Fig. 4 Flowchart of EGJO for adaptive IIR system identification

on the cooperative attacking behavior of golden jackals, which simulates searching for prey, stalking and enclosing prey, and pouncing on prey to effectively solve a complicated optimization problem. This section provides a concise explanation of EGJO's computational complexity. EGJO has three important steps: initialization, calculating the fitness value and updating the positions of the golden jackal-based

exploration phase and exploitation phase. In EGJO, N is the population size, T is the maximum iteration number, and D is the optimization dimension. The computational complexity of initialization is $O(N)$. The computational complexity of refreshing the golden jackals' positions is $O(T \times N) + O(T \times N \times D)$, which includes searching for prey and updating the positions of all jackals. Therefore, the computational complexity of EGJO requires $O(N \times (T + T \times D + 1))$, and EGJO strikes a balance between exploration and exploitation to discover the best solution.

6 Experimental results and analysis

6.1 Experimental setup

The numerical experiment is performed on a machine with an Intel Core i7-8750H 2.2 GHz CPU, a GTX1060, and 8 GB memory running Windows 10. MATLAB R2018b is employed to design each algorithm.

6.2 Parameter settings

To demonstrate its practicality and availability, EGJO is employed to perform IIR system identification. Additionally, EGJO is compared with other algorithms, such as the AOA, GTO, HHO, MDWA, RSO, WOA, TSA and GJO. The parameters are representative experimental values determined from the source articles. The initial parameters of each algorithm are described in Table 3.

6.3 Results and analysis

Three experimental datasets utilizing various orders of IIR models are employed to examine the overall optimization efficiency of EGJO. Case 1 contains a second-order system and a first-order IIR filter model. Case 2 contains a second-order system and a second-order IIR filter model. Case 3 contains a higher-order system and a higher-order IIR filter model. The fitness function is computed according to the mean squared error (MSE), and the purpose of optimization is to determine better control parameters and the best solution.

For case 1, each algorithm utilizes a first-order IIR filter model to identify a second-order system, and the transfer functions of the second-order system $H_P(z)$ and first-order IIR filter model $H_M(z)$ are computed as:

$$H_P(z) = \frac{0.05 - 0.4z^{-1}}{1 - 1.1314z^{-1} + 0.25z^{-2}} \tag{25}$$

$$H_M(z) = \frac{b}{1 - az^{-1}} \tag{26}$$

Table 3 Initial parameters of each algorithm

Algorithms	Parameters	Values
AOA	The minimum value of the accelerated function Min	0.2
	The maximum value of the accelerated function Max	1
	A sensitive parameter α	5
	A adjust parameter μ	0.5
GTO	Arbitrary value r_1	[0,1]
	Arbitrary value r_2	[0,1]
	Arbitrary value r_3	[0,1]
	Arbitrary value r_4	[0,1]
	Arbitrary value r_5	[0,1]
	Parameter p	0.03
	Arbitrary value l	[−1,1]
	Parameter β	3
HHO	Arbitrary value r_1	[0,1]
	Arbitrary value r_2	[0,1]
	Arbitrary value r_3	[0,1]
	Arbitrary value r_4	[0,1]
	Arbitrary value r_5	[0,1]
	Arbitrary value q	[0,1]
	Escaping energy of the prey E_0	(−1,1)
	Arbitrary value u	[0,1]
	Arbitrary value v	[0,1]
	Default constant β	1.5
MDWA	Constant a_{max}	1
	Constant a_{min}	0
RSO	Arbitrary value R	[1, 5]
	Arbitrary value C	[0,2]
WOA	Arbitrary value r_1	[0,1]
	Arbitrary value r_2	[0,1]
	Convergence factor α	[0,2]
	Constant coefficient b	1
	Arbitrary value l	[−1,1]
TSA	Arbitrary value c_1	[0,1]
	Arbitrary value c_2	[0,1]
	Arbitrary value c_3	[0,1]
	Initial speed P_{min}	1
	Subordinate speed P_{max}	4
	Arbitrary value r_{and}	[0,1]

Table 3 (continued)

Algorithms	Parameters	Values
GJO	Arbitrary value rand	[0,1]
	Arbitrary value r	[0,1]
	Constant value c_1	1.5
	Arbitrary value u	(0,1)
	Arbitrary value v	(0,1)
	Constant value β	1.5
EGJO	Arbitrary value rand	[0,1]
	Arbitrary value r	[0,1]
	Constant value c_1	1.5
	Arbitrary value u	(0,1)
	Arbitrary value v	(0,1)
	Constant value β	1.5
	Arbitrary value k	(0,1)
	Reflectivity α	1
	Expansion coefficient γ	1.5
	Compression coefficient β	0.5
	Contraction coefficient β	0.2

For case 2, each algorithm utilizes a second-order IIR filter model to identify a second-order system, and the transfer functions of the second-order system $H_P(z)$ and second-order IIR filter model $H_M(z)$ are computed as:

$$H_P(z) = \frac{1}{1 - 1.4z^{-1} + 0.49z^{-2}} \tag{27}$$

$$H_M(z) = \frac{b}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{28}$$

For case 3, each algorithm utilizes a higher-order IIR filter model to identify a higher-order system, and the transfer functions of the higher-order system $H_P(z)$ and higher-order IIR filter model $H_M(z)$ are computed as:

$$H_P(z) = \frac{1 - 0.4z^{-2} - 0.65z^{-4} + 0.26z^{-6}}{1 - 0.77z^{-2} - 0.8498z^{-4} + 0.6486z^{-6}} \tag{29}$$

$$H_M(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} + b_4 z^{-4}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} + a_4 z^{-4}} \tag{30}$$

EGJO is applied to perform IIR system identification. The purpose is to obtain the best control parameters and the minimum fitness value. The experimental

Table 4 Experimental results (MSE) of each algorithm for case 1

Result	AOA	GTO	HHO	MDWA	RSO	WOA	TSA	GJO	EGJO
Best	0.009851	0.009322	0.009669	0.010691	0.012342	0.009516	0.012368	0.009483	**0.009168**
Worst	0.019396	0.018680	0.019226	0.018964	0.022423	0.018268	0.019725	0.018792	0.016810
Mean	0.016351	0.013768	0.012635	0.012902	0.016127	0.012783	0.016009	0.013328	0.012072
Std	0.002303	0.002572	0.002627	0.001732	0.002304	0.002612	0.002141	0.002054	0.001354
Rank	5	7	9	2	6	8	4	3	1

Table 5 Experimental results (MSE) of each algorithm for case 2

Result	AOA	GTO	HHO	MDWA	RSO	WOA	TSA	GJO	EGJO
Best	0.014713	0.014582	8.15E-07	1.45E-08	0.011002	0.000392	2.64E-05	8.84E-07	**5.53E-09**
Worst	0.263802	0.240881	0.203970	0.260636	0.247015	0.278381	0.245425	0.264077	8.65E-06
Mean	0.171508	0.173441	0.010845	0.030343	0.102070	0.118358	0.157619	0.100697	2.59E-06
Std	0.065306	0.069930	0.038807	0.072574	0.062958	0.104800	0.092881	0.110870	2.30E-06
Rank	4	5	2	6	3	8	7	9	1

Table 6 Experimental results (MSE) of each algorithm for case 3

Result	AOA	GTO	HHO	MDWA	RSO	WOA	TSA	GJO	EGJO
Best	0.016181	0.015563	0.016491	0.010060	0.003530	0.008712	0.014522	0.000287	**0.000120**
Worst	0.076487	0.027865	0.029821	0.048230	0.039421	0.061341	0.071397	0.018540	0.016788
Mean	0.054235	0.022889	0.023684	0.029130	0.015499	0.027015	0.050323	0.004794	0.003979
Std	0.021065	0.002477	0.003347	0.010741	0.009213	0.011093	0.020843	0.004704	0.004567
Rank	9	1	2	6	5	7	8	4	3

Table 7 The parameter estimation of each algorithm for case 1

Result	AOA	GTO	HHO	MDWA	RSO	WOA	TSA	GJO	EGJO
a	0.889462	0.916486	0.921203	0.939711	0.918949	0.930521	0.888718	0.926575	0.940133
b	-0.23312	-0.27069	-0.27306	-0.30367	-0.27806	-0.30404	-0.33452	-0.24916	-0.19151

Table 8 The parameter estimation of each algorithm for case 2

Result	AOA	GTO	HHO	MDWA	RSO	WOA	TSA	GJO	EGJO
a_1	-1.60202	-1.59953	-1.3988	-1.39991	-1.42186	-1.42554	-1.39942	-1.40085	-1.40005
a_2	0.673880	0.660885	0.488819	0.489948	0.513276	0.515343	0.489964	0.490824	0.490038
b	0.658291	0.627863	1	1	0.806369	0.998420	1	1	1

results (MSE) of each algorithm are described in Tables 4, 5 and 6. The parameter estimation of each algorithm is described in Tables 7, 8 and 9. For each algorithm, the population size is 30, the maximum iteration number is 500, and the number

Table 9 The parameter estimation of each algorithm for case 3

Result	AOA	GTO	HHO	MDWA	RSO	WOA	TSA	GJO	EGJO
a_1	-0.00080	0.875841	0.607707	0.345820	-0.01713	-0.67406	-0.10078	-0.01107	-0.02907
a_2	-0.51015	0.876974	0.586113	-0.50666	-0.32407	0.342941	-0.38709	-0.05135	-0.02004
a_3	$-3.5E-06$	0.889296	0.603236	0.004816	-0.00052	-0.15727	0.027978	-0.00080	0.025592
a_4	-0.00467	0.862314	0.207678	-0.01519	-0.51342	-0.44841	0.108418	-0.81958	-0.84339
b_0	1.021986	0.894428	0.941796	0.888708	0.853528	0.719575	0.788512	0.992582	0.988885
b_1	$-8.8E-06$	0.870726	0.651800	0.215131	-0.01710	-0.47432	-0.05636	-0.02328	-0.01877
b_2	-0.00257	0.875718	0.651227	-0.17559	-0.00012	0.482589	-0.16311	0.278890	0.310765
b_3	3.52E-06	0.880218	0.635699	0.081185	-0.01340	-0.04398	-0.09592	-0.00413	0.013029
b_4	-0.00269	0.882149	0.529053	0.107490	-0.01509	-0.09746	0.214868	-0.35868	-0.36576

of independent runs is 30. Best, worst, mean and Std represent the optimal value, worst value, mean value and standard deviation, respectively, which are taken as fundamental indicators to analyze robustness and sustainability. The optimal value is shown in bold, and the ranking is determined by the standard deviation. The elite opposition-based learning technique increases the population variety, broadens the search region, delays search stagnation and enhances the global search. The simplex technique accelerates the search process, improves the computational precision, increases the optimization depth and enhances the local search. EGJO combines the characteristics of these two search mechanisms to achieve supplementary superiority and enhance the total search ability. For cases 1 and 2, the optimal value is viewed as the minimum mean squared error (MSE) of the IIR filter model, which directly reflects the computational accuracy and optimization performance. The optimal values of EGJO are superior to those of the AOA, GTO, HHO, MDWA, RSO, WOA, TSA and GJO. The worst values and mean values of EGJO are superior to those of the other algorithms, which demonstrates that EGJO has strong robustness and superiority. The algorithm's robustness and reliability are directly reflected in the standard deviation. The algorithm's dependability increases with decreasing standard deviation. The standard deviations of EGJO are the smallest among all the algorithms, and its ranking is first, which demonstrates that EGJO has cost advantages and reliability. EGJO can switch exploration and exploitation to obtain the best control parameters. For case 3, the optimal values, worst values and mean values of EGJO are superior to those of the other algorithms. The standard deviations of EGJO are larger than those of GTO and HHO and better than those of the AOA, MDWA, RSO, WOA, TSA and GJO. The parameters of EGJO are better than those of the other algorithms. The experimental results imply that EGJO has sufficient robustness and reliability to achieve higher computation precision and better control parameters.

The elite opposition-based learning technique and the simplex technique are added to the basic GJO to obtain complementary advantages to solve various optimization problems. The elite opposition-based learning technique avoids search stagnation and enhances the global search by increasing the population variety and expanding the search region. The simplex technique improves

the computational precision and enhances the local search by accelerating the optimization process and increasing the search depth. EGJO has the characteristics of simple principles, easy implementation, few optimization parameters, avoidance of search stagnation, fast convergence rate, high computation precision, strong stability and robustness. EGJO utilizes effectual search mechanisms to switch between exploration and exploitation and then determine the optimal control parameters and the global optimal solution. The convergence curves of the algorithms are illustrated in Fig. 5a, c, e. The convergence rate and computational precision of EGJO are superior to those of the other algorithms, which indicates that EGJO has strong robustness and dependability. The ANOVA tests of the algorithms are illustrated in Fig. 5b, d, f. For cases 1 and 2, the standard deviations of EGJO are lower than those of the other algorithms. For case 3, the standard deviations of EGJO are higher than those of GTO and HHO and lower than those of the AOA, MDWA, RSO, WOA, TSA and GJO. The experimental results reveal that EGJO has excellent reliability and robustness to achieve a better convergence precision and lower standard deviation; it is an efficacious and constructive method for solving the IIR system identification problem.

The results of the p-value Wilcoxon rank-sum test are described in Table 10. The histograms of the *p value* test are illustrated in Fig. 6. Wilcoxon's rank-sum test is adapted to check whether there is a significant distinction between EGJO and the other algorithms [52]. $p < 0.05$ shows that the distinction is significant. $p \geq 0.05$ shows that the distinction is not significant. The experimental results reflect that there is a remarkable distinction between EGJO and the other algorithms, and the data have a particular level of reliability and authenticity that is not generated by chance.

Statistically, EGJO is derived from the cooperative attacking behavior of golden jackals to simulate searching for prey, stalking and enclosing prey, and pouncing on prey to determine the best solution. EGJO is employed to perform IIR system identification for the following reasons. First, the elite opposition-based learning technique and the simplex technique are integrated into the basic GJO. The elite opposition-based learning technique boosts the population variety, broadens the search region, decreases premature convergence and strengthens the global search. The simplex technique accelerates the search process, improves the computational precision, increases the optimization depth and strengthens the local search. EGJO achieves supplementary superiority to avoid falling into a local optimum. Second, EGJO has efficient and unique optimization mechanisms to update the positions of the golden jackals and reach the optimal solution. The control parameter $|E|$ is used to adjust exploration and exploitation. If $|E| \geq 1$, EGJO utilizes the search for prey to predict and find the prey, which enhances the exploration ability. If $|E| < 1$, EGJO utilizes enclosing and pouncing on the prey to adjust its position and capture the prey. Third, EGJO has simple principles, easy
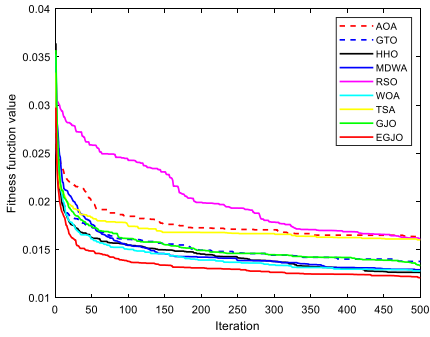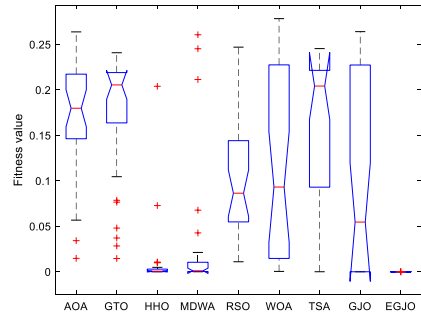
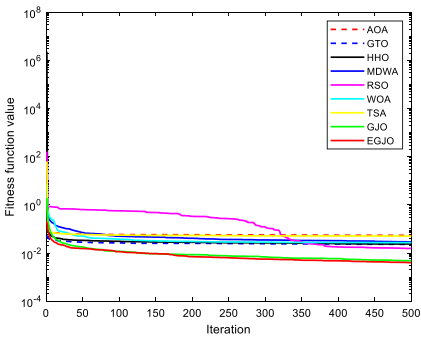**(a)** Convergence curves of algorithms for case 1
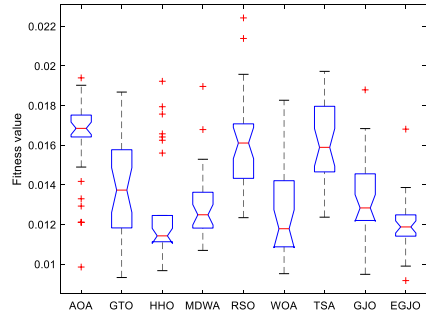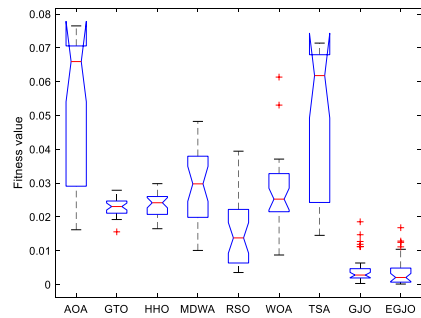
(b) ANOVA tests of algorithms for case 1
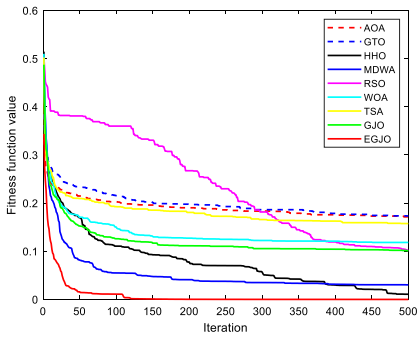
(c) Convergence curves of algorithms for case 2

(d) ANOVA tests of algorithms for case 2

(e) Convergence curves of algorithms for case 3

(f) ANOVA tests of algorithms for case 3

Fig. 5 Experimental results for cases 1–3

Table 10 Results of the p-value Wilcoxon rank-sum test

EGJO vs	AOA	GTO	HHO	MDWA	RSO	WOA	TSA	GJO
Case 1	1.85E−08	6.38E−03	3.55E−02	2.92E−02	1.41E−09	N/A	2.92E−09	3.85E−03
Case 2	3.02E−11	3.02E−11	5.07E−10	5.97E−09	3.02E−11	3.02E−11	3.02E−11	1.25E−07
Case 3	3.34E−11	3.34E−11	3.34E−11	6.07E−11	1.25E−07	1.09E−10	3.34E−11	0.145319

(a) Histogram of *p-value* test for case 1

(b) Histogram of *p-value* test for case 2

(c) Histogram of the *p-value* test for case 3
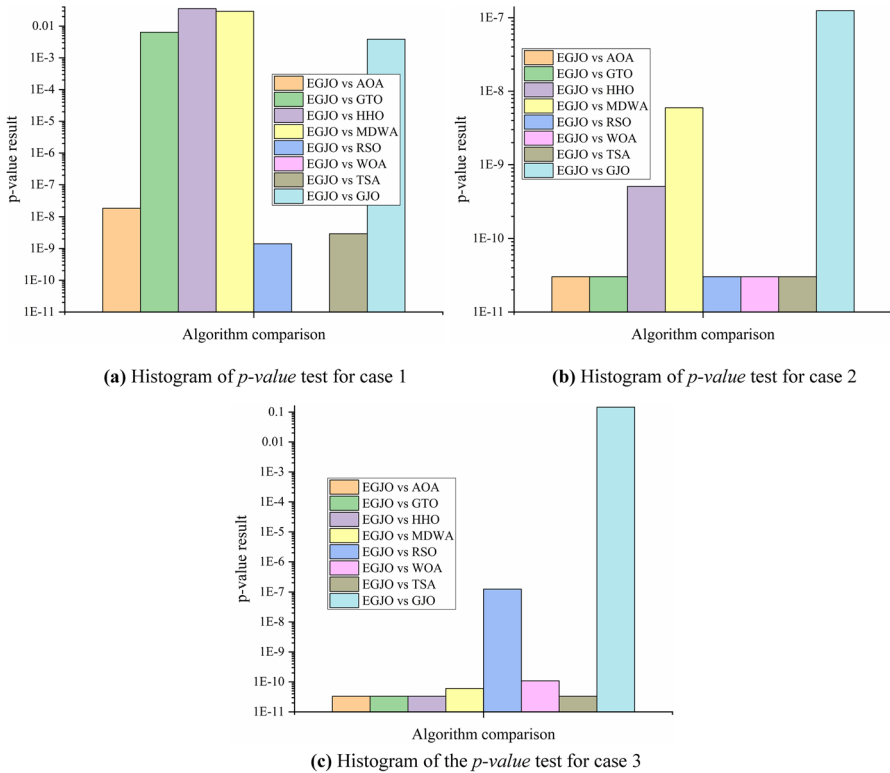
Fig. 6 Experimental results of the *p*-value test for cases 1–3

implementation, few optimization parameters, avoidance of search stagnation, a fast convergence rate, high computation precision and excellent stability and robustness. To summarize, EGJO balances exploration and exploitation to arrive at the best fitness value, which is a constructive and efficacious approach for performing IIR system identification.

7 Conclusions and future research

To address the shortcomings of premature convergence, a slow convergence rate and low computation precision faced by basic GJO, EGJO based on the elite opposition-based learning technique and the simplex technique are proposed to perform IIR system identification. The intention is to discover the optimal parameters and global optimal solution. The elite opposition-based learning technique increases the population variety and broadens the search region to avoid search stagnation and enhance the exploration ability. The simplex technique accelerates the optimization process and increases the search depth to improve the computational precision and enhance the exploitation ability. Therefore, EGJO can not

only utilize the mechanisms of searching for prey, stalking and enclosing prey, and pouncing on prey to achieve supplementary superiority and avoid premature convergence but also has strong robustness and adaptability to balance exploration and exploitation and determine the best value. To verify the practicality and availability of EGJO, EGJO is compared with AOA, GTO, HHO, MDWA, RSO, WOA, TSA and GJO through three sets of experiments. EGJO has strong global and local abilities to avoid premature convergence and obtain the best fitness value. The experimental results demonstrate that EGJO has a faster convergence rate, higher computation precision, stronger robustness and better optimization efficiency in discovering the global optimal solution. Additionally, EGJO is an efficacious and constructive approach for performing IIR system identification.

In future research, we will apply and evaluate EGJO on at least one to three real-world problems. We will further verify the effectiveness and feasibility of EGJO by comparing it to the latest works. We will evaluate EGJO on all 23 benchmark functions. We will introduce efficacious search mechanisms, unique coding methods and hybrid algorithms to achieve supplementary superiority and enhance the optimization ability, which will improve the convergence rate and the computation precision. Additionally, the modified GJO will be used to study the quality detection of the typical understory crops of *Dendrobium huoshanense* and *Camellia oleifera* with local characteristics. We will achieve the rapid collection of multidimensional data of the internal composition and external representation of understory crops and use hyperspectral technology and image processing methods to perform the intelligent detection and classification of understory crops. This will enable fast and accurate grading of the quality of understory crops, which is of key significance for improving the added value of crops.

Declarations

Conflict of interest The authors have no conflicts of interest to declare that are relevant to the content of this article.

Ethical approval This paper does not contain any studies with human or animal subjects, and all authors declare that they have no conflict of interest.

Consent to participate All authors declare that they have the consent to participate.

Consent for publication All authors declare that they have consent for publication.

References

1. Goswami OP, Rawat TK, Upadhyay DK (2020) A novel approach for the design of optimum IIR differentiators using fractional interpolation. Circuits Syst Signal Process 39:1688–1698
2. Lin Y-M, Badrealam KF, Kuo C-H et al (2021) Small molecule compound Nerolidol attenuates hypertension induced hypertrophy in spontaneously hypertensive rats through modulation of Mel-18-IGF-IIR signalling. Phytomedicine 84:153450
3. Iannelli A, Yin M, Smith RS (2021) Experiment design for impulse response identification with signal matrix models. IFAC-PapersOnLine 54:625–630
4. Chen L, Liu M, Wang Z, Dai Z (2020) A structure evolution-based design for stable IIR digital filters using AMECoDEs algorithm. Soft Comput 24:5151–5163
5. Liu Q, Lim YC, Lin Z (2019) A class of IIR filters synthesized using frequency-response masking technique. IEEE Signal Process Lett 26:1693–1697
6. Abualigah L, Diabat A, Mirjalili S et al (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 376:113609
7. Abdollahzadeh B, Soleimanian Gharehchopogh F, Mirjalili S (2021) Artificial gorilla troops optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. Int J Intell Syst 36:5887–5958
8. Heidari AA, Mirjalili S, Faris H et al (2019) Harris hawks optimization: algorithm and applications. Future Gener Comput Syst 97:849–872
9. Rizk-Allah RM, Hassanien AE (2019) A movable damped wave algorithm for solving global optimization problems. Evol Intell 12:49–72
10. Dhiman G, Garg M, Nagar A et al (2021) A novel algorithm for global optimization: rat swarm optimizer. J Ambient Intell Humaniz Comput 12:8457–8482
11. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67
12. Kaur S, Awasthi LK, Sangal A, Dhiman G (2020) Tunicate swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization. Eng Appl Artif Intell 90:103541
13. Niu Y, Yan X, Wang Y, Niu Y (2022) Dynamic opposite learning enhanced artificial ecosystem optimizer for IIR system identification. J Supercomput 78:13040–13085
14. Ababneh JI, Khodier MM (2022) Design and optimization of enhanced magnitude and phase response IIR full-band digital differentiator and integrator using the cuckoo search algorithm. IEEE Access 10:28938–28948
15. Mohammadi A, Sheikholeslam F, Mirjalili S (2022) Inclined planes system optimization: theory, literature review, and state-of-the-art versions for IIR system identification. Expert Syst Appl 200:117127
16. Chang W-D (2022) Identification of nonlinear discrete systems using a new Hammerstein model with Volterra neural network. Soft Comput 26:6765–6775
17. Durmuş B (2022) Infinite impulse response system identification using average differential evolution algorithm with local search. Neural Comput Appl 34:375–390
18. Kowalczyk M, Kryjak T (2022) Hardware architecture for high throughput event visual data filtering with matrix of IIR filters algorithm. ArXiv: 2207.00860
19. Mittal T (2022) A hybrid moth flame optimization and variable neighbourhood search technique for optimal design of IIR filters. Neural Comput Appl 34:689–704
20. Ababneh J, Khodier M (2021) Design of approximately linear phase low pass IIR digital differentiator using differential evolution optimization algorithm. Circuits Syst Signal Process 40:5054–5076
21. Mohammadi A, Zahiri SH, Razavi SM, Suganthan PN (2021) Design and modeling of adaptive IIR filtering systems using a weighted sum-variable length particle swarm optimization. Appl Soft Comput 109:107529
22. Liang X, Wu D, Liu Y et al (2021) An enhanced slime mould algorithm and its application for digital IIR filter design. Discret Dyn Nat Soc 2021:1–23. https://doi.org/10.1155/2021/5333278
23. Agrawal N, Kumar A, Bajaj V, Singh GK (2021) Design of digital IIR filter: A research survey. Appl Acoust 172:107669
24. Bui NT, Nguyen TMT, Park S et al (2021) Design of a nearly linear-phase IIR filter and JPEG compression ECG signal in real-time system. Biomed Signal Process Control 67:102431

25. Singh S, Ashok A, Kumar M, Rawat TK (2019) Adaptive infinite impulse response system identification using teacher learner based optimization algorithm. Appl Intell 49:1785–1802
26. Kumar M, Rawat TK, Aggarwal A (2017) Adaptive infinite impulse response system identification using modified-interior search algorithm with Lèvy flight. ISA Trans 67:266–279
27. Luo Q, Ling Y, Zhou Y (2020) Modified whale optimization algorithm for infinitive impulse response system identification. Arab J Sci Eng 45:2163–2176
28. Chang W-D (2018) A modified PSO algorithm for IIR digital filter modeling. J Circuits Syst Comput 27:1850073
29. Zhao R, Wang Y, Liu C et al (2020) Selfish herd optimization algorithm based on chaotic strategy for adaptive IIR system identification problem. Soft Comput 24:7637–7684
30. Ali TAA, Xiao Z, Sun J et al (2019) Optimal design of IIR wideband digital differentiators and integrators using salp swarm algorithm. Knowl-Based Syst 182:104834
31. Dhabal S, Venkateswaran P (2019) An improved global-best-guided cuckoo search algorithm for multiplierless design of two-dimensional IIR filters. Circuits Syst Signal Process 38:805–826
32. Cuevas E, Avalos O, Gálvez J (2023) IIR system identification using several optimization techniques: A Review Analysis. In: Analysis and comparison of metaheuristics, pp 89–104. https://doi.org/10.1007/978-3-031-20105-9_5
33. Mohammadi A, Zahiri SH, Razavi SM (2019) Infinite impulse response systems modeling by artificial intelligent optimization methods. Evol Syst 10:221–237
34. Mohammadi A, Zahiri SH (2018) Inclined planes system optimization algorithm for IIR system identification. Int J Mach Learn Cybern 9:541–558
35. Mohammadi A, Zahiri SH (2017) IIR model identification using a modified inclined planes system optimization algorithm. Artif Intell Rev 48:237–259
36. Mohammadi A, Zahiri SH (2016) Analysis of swarm intelligence and evolutionary computation techniques in IIR digital filters design. In: 2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC). IEEE, pp 64–69
37. Mohammadi A, Sheikholeslam F, Mirjalili S (2022) Nature-inspired metaheuristic search algorithms for optimizing benchmark problems: inclined planes system optimization to state-of-the-art methods. Arch Comput Methods Eng 30:331–389
38. Agrawal N, Kumar A, Bajaj V, Singh GK (2018) Design of bandpass and bandstop infinite impulse response filters using fractional derivative. IEEE Trans Ind Electron 66:1285–1295
39. Agrawal N, Kumar A, Bajaj V (2017) A new design method for stable IIR filters with nearly linear-phase response based on fractional derivative and swarm intelligence. IEEE Trans Emerg Top Comput Intell 1:464–477
40. Kumar A, Agrawal N, Sharma I et al (2018) Hilbert transform design based on fractional derivatives and swarm optimization. IEEE Trans Cybern 50:2311–2320
41. Agrawal N, Kumar A, Bajaj V (2020) Design of infinite impulse response filter using fractional derivative constraints and hybrid particle swarm optimization. Circuits Syst Signal Process 39:6162–6190
42. Janjanam L, Saha SK, Kar R, Mandal D (2021) Global gravitational search algorithm-aided Kalman filter design for Volterra-based nonlinear system identification. Circuits Syst Signal Process 40:2302–2334
43. Saha SK, Kar R, Mandal D, Ghoshal SP (2014) Harmony search algorithm for infinite impulse response system identification. Comput Electr Eng 40:1265–1285
44. Ahirwal MK, Kumar A, Singh GK (2013) EEG/ERP adaptive noise canceller design with controlled search space (CSS) approach in cuckoo and other optimization algorithms. IEEE/ACM Trans Comput Biol Bioinform 10:1491–1504
45. Ahirwal MK, Kumar A, Singh GK (2013) Descendent adaptive noise cancellers to improve SNR of contaminated EEG with gradient-based and evolutionary approach. Int J Biomed Eng Technol 13:49–68
46. Ahirwal MK, Kumar A, Singh GK (2014) Adaptive filtering of EEG/ERP through bounded range artificial bee colony (BR-ABC) algorithm. Digit Signal Process 25:164–172
47. Chopra N, Ansari MM (2022) Golden jackal optimization: a novel nature-inspired optimizer for engineering applications. Expert Syst Appl 198:116924
48. Yuan Y, Mu X, Shao X et al (2022) Optimization of an auto drum fashioned brake using the elite opposition-based learning and chaotic k-best gravitational search strategy based grey wolf optimizer algorithm. Appl Soft Comput 123:108947

49. Yan Z, Zhang J, Tang J (2021) Path planning for autonomous underwater vehicle based on an enhanced water wave optimization algorithm. Math Comput Simul 181:192–241

50. Nayak M, Das S, Bhanja U, Senapati MR (2022) Predictive analysis for cancer and diabetes using simplex method based social spider optimization algorithm. IETE J Res. 1–15. https://doi.org/10.1080/03772063.2022.2027276

51. Zhou Y, Zhou Y, Luo Q, Abdel-Basset M (2017) A simplex method-based social spider optimization algorithm for clustering analysis. Eng Appl Artif Intell 64:67–82

52. Wilcoxon F (1992) Individual comparisons by ranking methods. Breakthroughs in statistics. Springer, pp 196–202. https://doi.org/10.1007/978-1-4612-4380-9_16