e-Prime - Advances in Electrical Engineering, Electronics and Energy 5 (2023) 100225

Contents lists available at ScienceDirect

e-Prime - Advances in Electrical Engineering, Electronics and Energy

journal homepage: www.elsevier.com/locate/prime

Enhancing IIR system identification: Harnessing the synergy of gazelle optimization and simulated annealing algorithms

Serdar Ekinci [a], Davut Izci [a,b,*]

[a] Department of Computer Engineering, Batman University, Batman, 72100, Turkey
[b] MEU Research Unit, Middle East University, Amman, Jordan

ARTICLE INFO

Keywords:
Gazelle optimization algorithm
Simulated annealing technique
Digital filter design
IIR system identification

ABSTRACT

In the field of digital filter design and system identification, accurately modeling Infinite Impulse Response (IIR) systems is of utmost importance. This paper introduces a new adaptive algorithm that combines the gazelle optimization algorithm with simulated annealing to achieve superior optimization performance for digital IIR filters used in system identification. To evaluate the algorithm's effectiveness, extensive experiments were conducted on two fronts: the challenging CEC2020 benchmark functions and a fifth order IIR system identification problem. The proposed algorithm proves its effectiveness by achieving optimal solutions on the CEC2020 benchmark functions when compared to other well-known optimization algorithms such as arithmetic optimization, particle swarm optimization, differential evolution, sine cosine algorithm, gray wolf optimizer, biogeography-based optimization, salp swarm algorithm, and the original gazelle optimization algorithm. The algorithm's performance is measured by its ability to find the best solutions for these benchmark functions. For the IIR system identification problem, the study considers both the same order and reduced order cases. The system to be identified is represented by a fifth order plant. The algorithm's statistical analysis and convergence behavior are compared to the original gazelle optimization algorithm, artificial hummingbird algorithm, mountain gazelle optimization algorithm, and the widely used artificial bee colony algorithm. These comparisons demonstrate the proposed algorithm's superior capability and efficiency in accurately identifying the parameters of IIR systems.

1. Introduction

Digital filters play a crucial role in signal processing, and infinite impulse response (IIR) systems have emerged as a powerful class of filters that can effectively model complex system behaviors [1–3]. Unlike finite impulse response (FIR) filters, IIR filters incorporate feedback loops, allowing them to capture phenomena such as resonances and decays [4–6]. To accurately emulate real-world systems, identifying the optimal parameters for IIR filters is essential. This process involves applying input signals, measuring the corresponding outputs, and utilizing optimization algorithms to minimize the discrepancy between predicted and actual outputs [7].

Optimizing the parameter selection for IIR filter design often requires the application of metaheuristic algorithms, which excel in solving complex problems through stochastic search strategies [8–10]. In the context of system identification, metaheuristic algorithms enable exploration of the filter parameter space to discover the most suitable set of parameters that capture the system's characteristics [4]. Various metaheuristic algorithms used for IIR filter design. In this context, artificial bee colony algorithm [11], selfish herd optimization algorithm based on chaotic strategy [12], hybrid chimp with cuckoo search algorithm [13], average differential evolution algorithm with local search [14], modified inclined planes system optimization algorithm [15], differential evolution algorithm with a controllable probabilistic population size [16], modified whale optimization algorithm [17], modified-interior search algorithm with Lèvy flight [18], sigmoid attractiveness based improved firefly algorithm [19], improved chaotic harmony search algorithm [20], opposition based hybrid coral reefs optimization algorithm [21], particle swarm optimization [22], bat algorithm [23] and gravitational search algorithm [24] have been applied to this domain.

Recently, optimal modeling of IIR filtering systems has become a prominent research area. Consequently, several metaheuristic algorithms have been proposed for system identification problems involving

* Corresponding author.
E-mail address: davut.izci@batman.edu.tr (D. Izci).

https://doi.org/10.1016/j.prime.2023.100225
Received 11 May 2023; Received in revised form 26 June 2023; Accepted 14 July 2023
Available online 29 July 2023
2772-6711/© 2023 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

IIR filters. While considering the filter order is important for effective IIR filter design, it is equally crucial to utilize adaptive algorithms that reduce design complexity. This paper introduces a novel and efficient adaptive algorithm for designing digital IIR filters in system identification. By combining the gazelle optimization algorithm [25] and simulated annealing [26], the proposed approach aims to provide a compelling metaheuristic optimization procedure.

However, the original gazelle optimization algorithm may encounter issues such as premature convergence and slow convergence rates when applied to complex optimization problems. To address these challenges, hybridization with other metaheuristic algorithms has been found to be an effective solution [27]. Simulated annealing, in particular, is a point-to-point metaheuristic technique that is easy to implement, offers local search capabilities, and requires less computational time [28]. Therefore, this work adopts simulated annealing to improve the performance of the gazelle optimization algorithm, resulting in an efficient adaptive algorithm for designing digital IIR filters used in system identification.

To demonstrate the efficacy of the proposed algorithm, extensive experiments are conducted on challenging CEC2020 benchmark functions, as well as a fifth order IIR system identification problem. The performance of the simulated annealing-based gazelle optimization algorithm is initially evaluated by comparing it against several competitive optimization algorithms, such as arithmetic optimization algorithm [29], particle swarm optimization [30], differential evolution [31], sine cosine algorithm [32], grey wolf optimizer [33], biogeography-based optimization [34], salp swarm algorithm [35] and original gazelle optimization algorithm [25]. Statistical results confirm the optimization efficacy of the proposed approach.

In the context of IIR system identification, two more recent algorithms, namely the artificial hummingbird algorithm [36] and mountain gazelle optimization [37], are employed for comparison alongside the widely used artificial bee colony algorithm [38]. The statistical analysis and convergence behavior for both the same order and reduced order cases provide strong evidence for the effectiveness and reliability of the proposed algorithm in IIR system identification. It emerges as a promising and competitive approach in this field.

Furthermore, a comprehensive statistical performance comparison is conducted, comparing the proposed algorithm with various reported optimization approaches, including particle swarm optimization [39], hybrid particle swarm optimization and gravitational search algorithm [39], genetic algorithm [40], inclined planes system optimization [40], modified-interior search algorithm with Lèvy flight [41] and dynamic opposite learning enhanced artificial ecosystem optimizer [41]. The results demonstrate the significance of the proposed algorithm in IIR system identification, consistently outperforming other optimization algorithms in terms of achieving the lowest cost function (MSE) values. This highlights its superior capability and efficiency in accurately identifying IIR system parameters.

In conclusion, the proposed algorithm represents a highly effective and novel hybrid approach that combines the strengths of the gazelle optimization algorithm and simulated annealing technique. It demonstrates superior optimization performance compared to recent competitive algorithms, positioning it as a promising solution for a wide range of optimization problems. With its simple workflow, ease of use, and efficient search space exploration, the proposed algorithm holds great potential for various engineering and scientific applications.

2. Gazelle Optimization Algorithm

Gazelles possess unique attributes that contribute to their successful escape from predators. Studies have shown that predators are successful only 34% of the time in catching gazelles. The gazelle optimization algorithm (GOA) utilizes this information to develop a new metaheuristic algorithm that leverages the survival abilities of gazelles to solve real-world optimization problems [25]. Gazelles inhabit various regions, including parts of Asia, the Arabian Peninsula, the Sahara Desert, and sub-Saharan Sahel and northeast Africa. They are one of the common prey for many predators. Gazelles exhibit adaptive characteristics such as agility, keen senses, and living in groups for security and social reasons. They primarily feed on plants and have a social structure where females live in smaller groups with their fawns, while males form bachelor herds. Gazelles are hunted by predators like cheetahs, jackals, hyenas, leopards, and lions. They have evolved effective strategies for evading predators, including stotting (leaping in the air) and running at exceptional speeds.

2.1. Population Initialization

The GOA is a population-based optimization algorithm that utilizes randomly initialized gazelles (X) as search agents. The search agents are represented by an $n \times d$ matrix of candidate solutions (Eq. (1)). The algorithm incorporates the problem's upper bound (UB) and lower bound (LB) constraints to define the range of values for the population vector.

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d-1} & x_{1,d} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,d-1} & x_{2,d} \\ \vdots & \vdots & x_{i,j} & \vdots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,d-1} & x_{n,d} \end{bmatrix} \tag{1}$$

Here, X represents the matrix of candidate population's position vectors. Each position vector ($x_{i,j}$) is stochastically generated using Eq. (2), where *rand* denotes a random number, and UB_j and LB_j are the upper and lower bounds for the problem, respectively. The variables n and d represent the number of gazelles and the search space dimension of the optimization problem, respectively.

$$x_{i,j} = rand \times \left(UB_j - LB_j\right) + LB_j \tag{2}$$

In each iteration, $x_{i,j}$ generates a candidate solution, and the best solution obtained so far is identified as the minimum solution. In nature, the strongest or fittest gazelles have superior abilities in spotting threats, alerting others, and evading predators. Thus, the best-obtained solution is designated as the top gazelle and used to construct an Elite $n \times d$ matrix (Eq. (3)). This matrix serves as a reference for the gazelles to determine their next steps during the search process.

$$Elite = \begin{bmatrix} x'_{1,1} & x'_{1,2} & \cdots & x'_{1,d-1} & x'_{1,d} \\ x'_{2,1} & x'_{2,2} & \cdots & x'_{2,d-1} & x'_{2,d} \\ \vdots & \vdots & x'_{i,j} & \vdots & \vdots \\ x'_{n,1} & x'_{n,2} & \cdots & x'_{n,d-1} & x'_{n,d} \end{bmatrix} \tag{3}$$

Here, $x'_{i,j}$ represents the position vector of the top gazelle. The GOA considers both predators and gazelles as search agents because when a predator is spotted, both parties flee in the same direction towards safety. As the gazelles escape, the predator also explores the search space. At the end of each iteration, the Elite matrix is updated if a better gazelle replaces the top gazelle.

2.1.1. Brownian Motion

The displacement in a random motion is governed by the Normal (Gaussian) probability distribution function with a mean (μ) of 0 and a unit variance (σ^2) of 1. At a given point x, the standard Brownian motion [42] is described by the following definition.

$$f_B(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)} = \frac{1}{\sqrt{2\pi}} e^{\left(\frac{-x^2}{2}\right)} \tag{4}$$

2.1.2. Lévy Flight

The Lévy flight involves a random walk utilizing the Lévy distribution, which exhibits a power-law tail. Eq. (5) describes the Lévy distribution [43].

$$L(x_j) \approx |x_j|^{1-\alpha} \tag{5}$$

In this equation, x_j represents the flight distance, and α is the power-law exponent, constrained within the range (1,2]. The Lévy stable process, denoted as an integral in Eq. (6) [44], is defined as follows:

$$f_L(x;\alpha,\ \gamma) = \frac{1}{\pi}\int_0^{\infty} exp(-\gamma q^{\alpha})cos(qx)\delta q \tag{6}$$

Here, α serves as the distribution index that governs the motion, and γ represents the scale unit. GOA utilizes an algorithm proposed by [44] to generate stable Lévy motion. The algorithm adopts α values ranging from 0.3 to 1.99, and its formulation is provided in Eq. (7).

$$Levy(\alpha) = 0.05\ \times \frac{x}{|y|^{\frac{1}{\alpha}}} \tag{7}$$

The variables α, x, and y are defined as follows: x follows a normal distribution with mean 0 and variance σ_x^2, y follows a normal distribution with mean 0 and variance σ_y^2, and α is set to 1.5. The values of σ_x and σ_y are calculated as: $\sigma_x = (\Gamma(1+\alpha)\sin(\pi\alpha/2)/(\Gamma((1+\alpha)/2)\alpha 2^{(\alpha-1/2)}))^{1/\alpha}$and $\sigma_y = 1$.

2.2. Modeling the GOA

The proposed GOA algorithm emulates the survival behavior of gazelles, encompassing grazing in the absence of predators and fleeing to a safe haven when a predator is detected. Therefore, the optimization process of the GOA algorithm can be divided into two distinct phases.

2.2.1. Exploitation

In the exploitation phase, the gazelles peacefully forage without a predator or in the presence of a stalking predator. To effectively explore the neighboring areas of the domain, a controlled Brownian motion with uniform and controlled steps is employed. The gazelles are assumed to move in a Brownian motion pattern during grazing. The mathematical model representing this behavior is given by Eq. (8).

$$gazelle_{i+1} = gazelle_i + s{\cdot}R * {\cdot}R_B * {\cdot}(Elite_i - R_B * {\cdot}gazelle_i) \tag{8}$$

Here, $gazelle_{i+1}$ represents the solution at the next iteration, while $gazelle_i$ represents the solution at the current iteration. The parameter s denotes the grazing speed of the gazelles. R_B is a vector comprising random numbers that simulate the Brownian motion, and R is a vector of uniform random numbers ranging from 0 to 1.

2.2.2. Exploration

The exploration phase begins when a predator is spotted. In response to danger, gazelles exhibit various behaviors such as tail flicking, foot stomping, or stotting up to a height of 2 m, which is scaled to a value between 0 and 1. This phase of the algorithm incorporates the use of Lévy flight, which involves taking small steps interspersed with occasional long jumps. This technique has demonstrated improved searchability in optimization studies. When the predator is sighted, the gazelle initiates a run while the predator gives chase. Both runs involve sudden changes in direction, represented by the parameter μ. In this study, it is assumed that the change in direction occurs in every iteration, with the gazelle moving in one direction during odd iterations and the opposite direction during even iterations. The gazelle responds first, employing Lévy flight for its escape, while the predator initially uses Brownian motion before transitioning to Lévy flight. The mathematical model describing the gazelle's behavior upon spotting the predator is presented in Eq. (9).

$$\overrightarrow{gazelle_{i+1}} = \overrightarrow{gazelle_i} + S{\cdot}\mu{\cdot}\overrightarrow{R} * {\cdot}\overrightarrow{R_L} * {\cdot}\left(\overrightarrow{Elite_i} - \overrightarrow{R_L} * {\cdot}\overrightarrow{gazelle_i}\right) \tag{9}$$

Here, S represents the top speed that the gazelle can achieve, $\overrightarrow{R_L}$ is a vector of random numbers based on Lévy distributions. The behavior of the predator chasing the gazelle is described by the mathematical model in Eq. (10).

$$\overrightarrow{gazelle_{i+1}} = \overrightarrow{gazelle_i} + S{\cdot}\mu{\cdot}CF * {\cdot}\overrightarrow{R_B} * {\cdot}\left(\overrightarrow{Elite_i} - \overrightarrow{R_L} * {\cdot}\overrightarrow{gazelle_i}\right) \tag{10}$$

In Eq. (10), CF represents the cumulative effect of the predator, calculated as $CF = (1 - iter/iterMax)^{(2\ *\ iter/iterMax)}$. The authors of a study on Mongolian Gazelles reported an annual survival rate of 0.66, implying that predators are successful in only 0.34 situations [45]. The predator success rate ($PSRs$) has an impact on the gazelle's ability to escape and prevents the algorithm from being trapped in local minima. The effect of $PSRs$ is modeled in Eq. (11).

$$\overrightarrow{gazelle_{i+1}} = \begin{cases} \overrightarrow{gazelle_i} + CF\left[\overrightarrow{LB} + \overrightarrow{R} * {\cdot}\left(\overrightarrow{UB} - \overrightarrow{LB}\right)\right] * {\cdot}\overrightarrow{U}; if\ r \le PSRs \\ \overrightarrow{gazelle_i} + [PSRs(1-r) + r]\left(\overrightarrow{gazelle_{r1}} - \overrightarrow{gazelle_{r2}}\right); else \end{cases} \tag{11}$$

In Eq. (11), $\overrightarrow{U}$ is a binary vector constructed by generating a random number r from the interval [0, 1], such that $\overrightarrow{U} = 0$ for $r < 0.34$; otherwise $\overrightarrow{U} = 1$. $r1$ and $r2$ are random indexes of the gazelle matrix.

3. Proposed simulated annealing-based gazelle optimization algorithm

Population-based algorithms strive to strike a balance between exploitation and exploration, which is crucial for effectively exploring promising regions in the search space and converging towards the global optimum solution. Exploration involves investigating unknown regions in the search space to identify the global optimum, while exploitation leverages previously satisfactory solutions to obtain better ones. The development of the proposed algorithm, called simulated annealing-based gazelle optimization algorithm (GOASA), aims to create an efficient method for optimizing parameter selection in the design of IIR filters.

The original version of the GOA uses a blind operator for exploitation without considering the fitness value of the current solution. To improve the exploitation process, the simulated annealing (SA) algorithm, a widely used local search metaheuristic, is incorporated. SA draws inspiration from the annealing process of solids and is renowned for its effectiveness in global optimization [26]. The acceptance probability (P) of a trial solution point, X_{i+1}, from the current solution point, X_i, is governed by $P = 1$ for $F(X_{i+1}) \le F(X_i)$ and $P = exp([F(X_i) - F(X_{i+1})]/T_i)$ for $F(X_{i+1}) > F(X_i)$, where F represents the objective function and T is the temperature. The acceptance probability depends on the comparison between the fitness values of the current and trial solutions. The SA algorithm requires defining an initial temperature (T_o) and a minimum temperature (T_{min}) before the optimization process starts. This technique facilitates early convergence to the global minimum by enabling escape from local minima. Cooling of the temperature is achieved using the cooling rule described by $T_{i+1} = \alpha{\cdot}T_i$, with the parameter α representing the cooling factor. The SA offers the advantage of escaping local minima, thereby preventing stagnation at suboptimal solutions.

The proposed GOASA algorithm combines the GOA and SA algorithms to leverage their strengths. In this hybrid approach, the SA algorithm is utilized to enhance the best solution found after each iteration of the GOA algorithm. The goal is to search the most promising regions identified by GOA and improve the exploitation process. The procedure of the hybrid GOASA algorithm is illustrated in Fig. 1, which outlines the steps involved. Initially, the parameters of both the original GOA and SA algorithms are defined. Then, a random initial population is generated to calculate the objective function values. Subsequently, the basic steps of the GOA and SA algorithms are followed until the termination condition is met. Once the maximum number of iterations is reached, the proposed algorithm concludes, providing the optimized

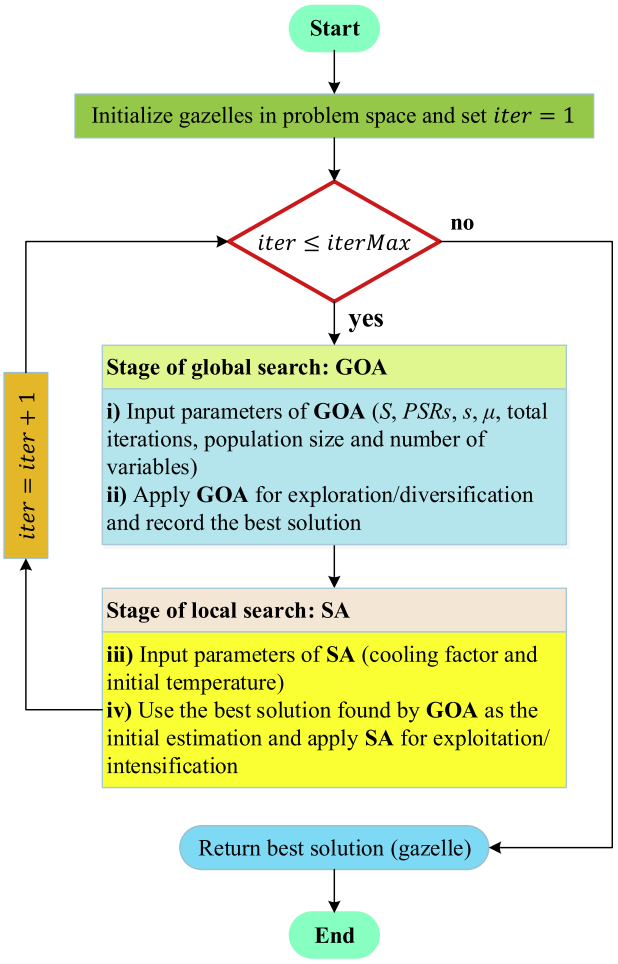


Fig. 1. Flowchart of proposed GOASA.

parameters and the corresponding objective function value.

4. CEC2020 benchmark functions and comparative results

In this study, the CEC2020 benchmark functions were employed to evaluate the performance of the proposed GOASA algorithm. Table 1 provides a comprehensive overview of the CEC2020 benchmark functions, presenting their details and characteristics. In this study, recent and good performing optimizers known as GOA [25], arithmetic optimization algorithm (AOA) [29], particle swarm optimization (PSO) [30], differential evolution (DE) [31], sine cosine algorithm (SCA) [32], grey wolf optimizer (GWO) [33], biogeography-based optimization (BBO) [34], and salp swarm algorithm (SSA) [35] are used along for the initial performance evaluation of the proposed GOASA optimizer.

To ensure consistency and comparability, a population size of 50 and a maximum number of iterations set at 1000 were used for all algorithms. To obtain reliable and robust results, each algorithm was executed for 30 independent runs. Throughout the experiments, the proposed GOASA algorithm was compared against the aforementioned algorithms, with a focus on assessing its superiority in terms of optimization performance. Table 2 presents the control parameter settings used for all algorithms in the comparative study. These settings were carefully chosen to ensure a fair evaluation and to allow for meaningful comparisons between the algorithms.

Finally, Table 3 provides the statistical results of the CEC2020 benchmark functions obtained from the experiments. The statistical results of the CEC2020 benchmark functions demonstrate that the GOASA algorithm outperforms the GOA, AOA, PSO, DE, SCA, GWO, BBO, and SSA algorithms in terms of finding better solutions. GOASA consistently achieves superior results with lower standard deviations, highlighting its effectiveness and efficiency in optimizing complex benchmark functions.

Table 1
CEC2020 benchmark functions [25].

ID	Type	Function name	Range	Dimension	F^*_{min}
$F_{CEC2020-1}$	Unimodal function	Shifted and rotated bent cigar function	[−100, 100]	10	100
$F_{CEC2020-2}$	Basic functions	Shifted and rotated Schwefel's function	[−100, 100]	10	1100
$F_{CEC2020-3}$		Shifted and rotated Lunacek bi-Rastrigin function	[−100, 100]	10	700
$F_{CEC2020-4}$		Expanded Rosenbrock's plus Griewangk's function	[−100, 100]	10	1900
$F_{CEC2020-5}$	Hybrid functions	Hybrid function 1 ($N = 3$)	[−100, 100]	10	1700
$F_{CEC2020-6}$		Hybrid function 2 ($N = 4$)	[−100, 100]	10	1600
$F_{CEC2020-7}$		Hybrid function 3 ($N = 5$)	[−100, 100]	10	2100
$F_{CEC2020-8}$	Composition functions	Composition function 1 ($N = 3$)	[−100, 100]	10	2200
$F_{CEC2020-9}$		Composition function 2 ($N = 4$)	[−100, 100]	10	2400
$F_{CEC2020-10}$		Composition function 3 ($N = 5$)	[−100, 100]	10	2500

Table 2
Control parameters settings of all algorithms.

Algorithm	Parameter	Value
GOASA	s	[0, 1]
	μ	[−1, 1]
	S	0.88
	$PSRs$	0.34
	Cooling factor	0.982
	Initial temperature	0.1
GOA	s	[0, 1]
	μ	[−1, 1]
	S	0.88
	$PSRs$	0.34
AOA	α	5
	μ	0.05
PSO	c_1 and c_2	2
	w_{min}	0.2
	w_{max}	0.9
DE	β_{min}	0.2
	β_{max}	0.8
	Crossover probability	0.8
SCA	a	2
GWO	a	[0, 2]
	r_1 and r_2	[0, 1]
BBO	n_{keep}	0.2
	Mutation probability	0.9
SSA	c_2 and c_3	[0, 1]

Table 3
Statistical results of CEC2020 benchmark functions.

ID	Metric	GOASA	GOA	AOA	PSO	DE	SCA	GWO	BBO	SSA
$F_{CEC2020-1}$	Best	**100**	100.04	1.62E+09	101.9	199.57	100.04	5734.3	103.43	147.43
	Worst	**100**	101.51	1.03E+10	8494.5	6437.4	1.91E+09	3.43E+08	9805	9096.2
	Mean	**100**	100.43	5.11E+09	2008.6	1759.2	6.96E+08	1.33E+07	1343.2	2893.4
	Standard Deviation	**2.6306E−13**	0.35345	2.22E+09	2336.6	1467.8	3.24E+08	6.28E+07	2201.4	2845.9
$F_{CEC2020-2}$	Best	**1100.1**	1106.6	1266	1107	1246.5	1106.6	1116.7	1113.6	1320.7
	Worst	**1123.6**	1495.8	2787.2	1709.9	1804.1	2760.4	1941.2	2115	2378.9
	Mean	**1103.9**	1246.4	2124	1399.6	1540.9	2308.3	1535.4	1681.2	1861.6
	Standard Deviation	**5.2782**	107.35	329.5	137.58	130.05	224.26	233.62	244.51	276.66
$F_{CEC2020-3}$	Best	**700.61**	711.23	759.98	703.6	716.7	711.23	714.36	715.05	716.96
	Worst	**705.8**	724.6	821.47	725.1	725.64	802.59	749.27	735.73	772.3
	Mean	**703.8**	719.59	798.12	716.98	721.45	776.62	727.71	722.02	734.23
	Standard Deviation	**2.0197**	3.0154	17.539	4.9699	2.3513	9.9176	9.3905	4.9793	12.331
$F_{CEC2020-4}$	Best	**1900**	1900.5	3735.6	1900.3	1901.1	1900.5	1900.7	1900.5	1900.8
	Worst	**1900.4**	1901.8	2.77E+05	1902	1902.2	1941.3	1904.1	1906.5	1904.4
	Mean	**1900.1**	1901.2	50,020	1901.1	1901.6	1919	1902.4	1901.9	1901.6
	Standard Deviation	**0.12811**	0.31242	56,624	0.46064	0.25486	8.2064	0.75162	1.2235	0.9302
$F_{CEC2020-5}$	Best	**1707.7**	1734.7	3968.2	1871.2	4359.2	1734.7	3029.7	2461	2681.8
	Worst	**1749.1**	1799.6	6.78E+05	10,218	80,032	77,282	6.11E+05	1.31E+05	15,641
	Mean	**1729.2**	1756.1	1.43E+05	4709.3	28,146	31,187	69,600	24,327	5928.7
	Standard Deviation	**11.145**	16.75	1.33E+05	2794.8	17,680	21,634	1.69E+05	33,785	3600.6
$F_{CEC2020-6}$	Best	**1600**	**1600**	1600.5	**1600**	**1600**	**1600**	**1600**	**1600**	**1600**
	Worst	**1600**	**1600**	1622.9	1658.5	1600.4	1601.7	1625.5	1617.6	1617.1
	Mean	**1600**	**1600**	1611.5	1612.6	1600.2	1601	1603	1602.8	1601.6
	Standard Deviation	**0.0001819**	0.000893	8.3504	14.912	0.10355	0.21847	6.6623	5.8203	4.2149
$F_{CEC2020-7}$	Best	**2101.3**	2102.9	2768.6	2217.7	2271.8	2102.9	2708.6	2161.5	2424.6
	Worst	**2106**	2122.5	9834.9	4542.1	8615.2	15,558	18,025	28,362	26,567
	Mean	**2102.9**	2110	6090.4	2718.8	3340.3	7865	9151.4	9738.9	6743.2
	Standard Deviation	**1.5215**	5.5831	2245.1	456.78	1260.2	3543	4892.4	7763.2	6332.9
$F_{CEC2020-8}$	Best	**2200**	2200.2	2478.1	2300.7	2261.9	2200.2	2300.9	2300.3	2223
	Worst	**2302**	2303.5	3865.3	2892.2	2302.1	2502	2322.5	2304.6	2308.8
	Mean	**2270.8**	2286.7	2786.2	2321.6	2298.8	2374.9	2304.7	2302	2299
	Standard Deviation	48.847	35.252	274.14	107.77	7.5443	39.318	4.7891	**1.043**	18.101
$F_{CEC2020-9}$	Best	**2400**	2418.4	2569.3	2729.4	2646.2	2418.4	2502	2400	2500
	Worst	**2732.5**	2734.1	2892.7	2756.2	2755.4	2798.3	2774.7	2760.4	2786.3
	Mean	**2503.3**	2523.3	2799.1	2740.5	2737	2774.8	2735.5	2725.3	2740.7
	Standard Deviation	90.636	67.32	78.55	**6.0925**	23.448	40.458	45.785	76.252	46.867
$F_{CEC2020-10}$	Best	**2600**	2601.3	2929.6	2897.8	2899.2	2601.3	2898.1	2897.8	2897.8
	Worst	**2897.7**	2897.8	3625.1	2947.1	2946.6	2988.6	2949.7	2950.6	2949.3
	Mean	**2838.2**	2871.8	3171.8	2921.6	2918.5	2962.1	2937.3	2932	2918.8
	Standard Deviation	125.54	80.727	148.27	23.392	14.58	**12.341**	17.896	22.204	23.897

5. Implementation to high-order IIR system identification

5.1. Problem formulation and proposed novel approach for IIR system identification

System identification is the process of finding the mathematical representation of an unknown system by analyzing its input and output data. In this context, optimization algorithms are employed to minimize the error between the output of a candidate model and the actual output of the system. This allows for the discovery of an optimal model that accurately represents the unknown system. In real-world applications, infinite impulse response (IIR) models are often utilized due to their ability to achieve accurate representation with fewer model parameters [3]. The relationship between the input and output of an IIR system can be expressed through the following equation, where b_j represents the numerator coefficients and a_i represents the denominator coefficients of the IIR filter. The orders of the numerator and denominator are denoted by M and N ($N \geq M$), respectively:

$$y(k) = \sum_{j=0}^{M} b_j x(k-j) - \sum_{i=1}^{N} a_i y(k-i) \tag{12}$$

In the problem of modeling an IIR system, it is assumed that the actual system is known by considering standard IIR plants. The objective is to design an IIR filter with a transfer function that can accurately replicate the characteristics of the unknown system. By assuming a coefficient of $a_0 = 1$, the following transfer function can be obtained for a digital IIR filter:

$$H(z^{-1}) = \frac{\sum_{j=0}^{M} b_j z^{-j}}{1 + \sum_{i=1}^{N} a_i z^{-i}} \tag{13}$$

As observed in the above equation, the z-transform is utilized to represent digital IIR filters, as they operate in discrete time [40]. In this study, the optimal coefficients of the IIR filter, represented as the vector $[a_i, b_j]$, are determined to approach the response of the actual system using an adaptive IIR filter. Fig. 2 illustrates the block diagram of an adaptive IIR filter designed through the proposed GOASA algorithm for the purpose of system identification. In the diagram, $x(k)$ and $y(k)$ represent the input and output of the filter, respectively.

5.2. Simulation results

In the simulations conducted for this study, the system input $x(k)$ is generated as a uniformly distributed white sequence, taking values

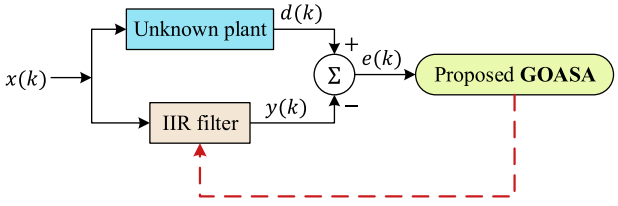


Fig. 2. Detailed block diagram of IIR system identification problem via proposed GOASA.

within the range of $(-0.5,\ 0.5)$. The length of data used to calculate the *MSE* cost function is set to $W = 200$. For the experiments, the algorithm parameters are configured as follows: 30 runs, 500 total iterations, and a population size of 50. In addition to the original GOA and the proposed GOASA algorithms, two more recent algorithms, artificial hummingbird algorithm (AHA) [36] and mountain gazelle optimization (MGO) [37], are employed for comparison. The widely used artificial bee colony (ABC) [38] algorithm is also adopted to assess the performance of the GOASA algorithm. The benchmark example chosen for the experiments involves a highly complex fifth-order plant, considering both same-order and reduced-order IIR filters [5,23,40,41,46,47]. The transfer function of the fifth-order plant is expressed as follows.

$$H_P(z^{-1}) = \frac{0.1084+0.5419z^{-1}+1.0837z^{-2}+1.0837z^{-3}+0.5419z^{-4}+0.1084z^{-5}}{1+0.9853z^{-1}+0.9738z^{-2}+0.3864z^{-3}+0.1112z^{-4}+0.0113z^{-5}} \tag{14}$$

For the same-order IIR filter approximation of the fifth-order plant, Eq. (15) represents the transfer function.

$$H_S(z^{-1}) = \frac{b_0 + b_1z^{-1} + b_2z^{-2} + b_3z^{-3} + b_4z^{-4} + b_5z^{-5}}{1 - a_1z^{-1} - a_2z^{-2} - a_3z^{-3} - a_4z^{-4} - a_5z^{-5}} \tag{15}$$

And for the reduced-order IIR filter approximation, Eq. (16) provides the transfer function.

$$H_R(z^{-1}) = \frac{b_0' + b_1'z^{-1} + b_2'z^{-2} + b_3'z^{-3} + b_4'z^{-4}}{1 - a_1'z^{-1} - a_2'z^{-2} - a_3'z^{-3} - a_4'z^{-4}} \tag{16}$$

We analyzed the statistical results obtained from the MSE values of the proposed GOASA algorithm in comparison to the GOA, AHA, MGO, and ABC algorithms for IIR system identification. The statistical metrics, including the best, worst, mean, and standard deviation, are presented for both the same order and reduced order cases in Table 4. For the same order case, the GOASA algorithm demonstrates superior performance in terms of the MSE values compared to the other algorithms. The best MSE value achieved by GOASA (8.4199E-10) is significantly smaller than those of the other algorithms, indicating its ability to provide highly accurate approximations of the unknown system. Furthermore, the worst MSE value for GOASA (1.2312E-08) is notably lower than the worst values of the other algorithms, highlighting its robustness in avoiding high errors. The mean MSE value for GOASA (3.3646E-09) is considerably smaller compared to the mean values of the other algorithms, indicating its effectiveness in consistently achieving more accurate results. Additionally, the standard deviation of MSE values for GOASA (2.7401E-09) is significantly smaller than those of the other algorithms, signifying its stability and reliability in producing consistent results.

Moving on to the reduced order case, GOASA continues to exhibit competitive performance. The best MSE value obtained by GOASA (3.9165E-07) is comparable to those of the other algorithms, demonstrating its ability to accurately approximate the system even with reduced-order filters. The worst MSE value for GOASA (4.5756E-07) is lower than the worst values of some other algorithms, indicating its capability to avoid extremely poor approximations. The mean MSE value for GOASA (4.1907E-07) is relatively lower compared to most of the other algorithms, indicating its effectiveness in achieving reasonably accurate results on average. Moreover, the standard deviation of MSE values for GOASA (1.7476E-08) is smaller than that of most other algorithms, further emphasizing its consistency and reduced variability in the obtained results.

Overall, based on the statistical analysis of MSE values for both the same order and reduced order cases, it is evident that the proposed GOASA algorithm outperforms the GOA, AHA, MGO, and ABC algorithms in terms of IIR system identification. GOASA consistently achieves lower MSE values, indicating higher accuracy and improved performance. The significant reduction in both the best and worst MSE values, as well as the smaller mean and standard deviation, highlight the statistical significance and superiority of the GOASA algorithm in approximating the unknown system. These results provide strong evidence for the effectiveness and reliability of the proposed GOASA algorithm for IIR system identification, positioning it as a promising and competitive approach in this field.

Table 5 presents the estimated coefficient values for both the same order and reduced order cases, obtained through the utilization of various optimization algorithms, namely GOASA, GOA, AHA, MGO, and ABC. This table provides valuable insights into the performance of these algorithms in terms of achieving accurate parameter estimations for system identification. Analyzing the results, it becomes evident that the GOASA optimizer exhibits superior performance compared to the other algorithms. The estimated coefficient values obtained by GOASA demonstrate higher accuracy and precision, indicating its effectiveness in capturing the underlying characteristics of the system. This can be observed through the closer alignment of the estimated coefficients with the true or desired values.

The convergence capability of the GOASA algorithm for IIR system identification is further demonstrated through the convergence curves presented in Figs. 3 and 4, which correspond to the same order and reduced order models, respectively. These convergence curves provide valuable visual insights into the optimization performance of each algorithm. Fig. 3 showcases the convergence curve of the MSE cost function for the same order model. It is evident from the figure that the GOASA optimizer exhibits remarkable convergence behavior, consistently approaching the lowest MSE values as the iterations progress. The curve demonstrates a clear downward trend, indicating the algorithm's effectiveness in minimizing the error between the estimated model output and the actual output of the system. This observation signifies the superior performance of the GOASA algorithm in terms of achieving highly accurate and precise system identification results for the same order model.

Similarly, Fig. 4 illustrates the convergence curves of the MSE cost functions for the reduced order model. Once again, the GOASA algorithm stands out as it converges towards the lowest MSE values during the iterative process. The convergence curve demonstrates the algorithm's ability to effectively optimize the reduced order model, capturing the essential dynamics of the system while minimizing the estimation error. The consistently decreasing trend of the curve highlights the GOASA algorithm's robustness and reliability in identifying the parameters of the reduced order model with high accuracy.

Comparing the convergence curves of the GOASA algorithm with

Table 4
Statistical results of MSE values for the same and reduced order cases.

Type	Metric	GOASA	GOA	AHA	MGO	ABC
$H_S(z^{-1})$	Best	**8.4199E−10**	4.2881E−07	3.6019E−07	2.8017E−06	2.5507E−05
	Worst	**1.2312E−08**	2.2957E−06	4.0503E−04	7.9614E−04	1.8296E−03
	Mean	**3.3646E−09**	1.1445E−06	4.1289E−05	2.5299E−04	3.4294E−04
	Standard Deviation	**2.7401E−09**	5.7453E−07	9.1860E−05	1.7497E−04	3.2330E−04
$H_R(z^{-1})$	Best	**3.9165E−07**	5.2329E−07	1.9048E−05	4.1060E−05	4.1528E−05
	Worst	**4.5756E−07**	1.3382E−06	1.8565E−04	1.1097E−03	3.3087E−04
	Mean	**4.1907E−07**	8.2704E−07	5.4919E−05	3.7023E−04	1.3519E−04
	Standard Deviation	**1.7476E−08**	1.8816E−07	4.0700E−05	2.8610E−04	6.6252E−05

Table 5
Parameter estimation for the same and reduced order cases.

Type	Coefficients	GOASA	GOA	AHA	MGO	ABC
$H_S(z^{-1})$	a_1	−0.9855	−0.0764	0.1259	−0.1052	−0.2341
	a_2	−0.9736	−0.4426	−0.2857	−0.3294	−0.3779
	a_3	−0.3867	0.2351	0.3824	0.2192	0.1737
	a_4	−0.1110	−0.0152	0.0249	0.0492	0.0519
	a_5	−0.0114	0.0269	0.0320	0.0079	0.0070
	b_0	0.1084	0.1077	0.1081	0.1088	0.0974
	b_1	0.5419	0.4430	0.4213	0.4458	0.4490
	b_2	1.0838	0.6306	0.5253	0.6313	0.6826
	b_3	1.0838	0.2859	0.0907	0.2534	0.3480
	b_4	0.5419	−0.1003	−0.2709	−0.1601	−0.1030
	b_5	0.1084	−0.0936	−0.1541	−0.1258	−0.1282
$H_R(z^{-1})$	a_1'	−0.5855	−0.6044	−0.1182	−0.0950	−0.2598
	a_2'	−0.6891	−0.6929	−0.5712	−0.4970	−0.6032
	a_3'	−0.1045	−0.1150	0.1225	0.1193	0.0530
	a_4'	−0.0438	−0.0427	−0.0629	−0.0364	−0.0646
	b_0'	0.1080	0.1087	0.1103	0.1106	0.1021
	b_1'	0.4986	0.5008	0.4509	0.4486	0.4483
	b_2'	0.8787	0.8871	0.6634	0.6425	0.7144
	b_3'	0.7099	0.7228	0.3848	0.3329	0.4658
	b_4'	0.2247	0.2323	0.0197	−0.0287	0.0765

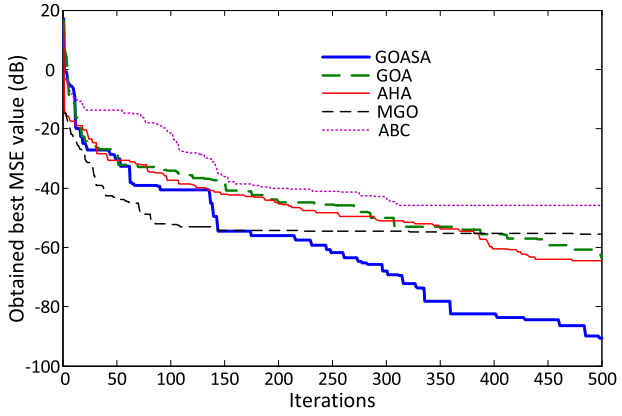


Fig. 3. Convergence curves of MSE cost functions for same order model.

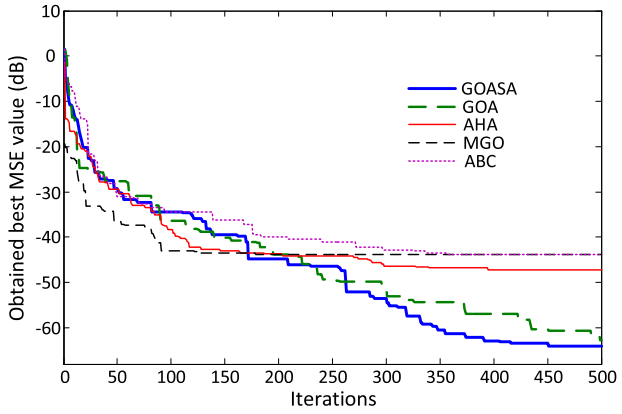


Fig. 4. Convergence curves of MSE cost functions for reduced order model.

those of the other optimization algorithms, it is evident that GOASA outperforms its counterparts in terms of achieving the lowest MSE values. The convergence curves affirm the superior optimization capability of the GOASA algorithm for IIR system identification, demonstrating its effectiveness in converging towards highly accurate and reliable model representations.

5.3. Performance comparison with reported approaches

In order to assess the significance of the proposed GOASA algorithm for IIR system identification, a comprehensive statistical performance comparison is conducted against various other reported optimization approaches. Table 6 presents the comparison results, which include metrics for both the same order and reduced order cases. The evaluated algorithms include particle swarm optimization (PSO) [39], hybrid particle swarm optimization and gravitational search algorithm (HPSO-GSA) [39], genetic algorithm (GA) [40], inclined planes system optimization (IPO) [40], modified-interior search algorithm with Lèvy flight (M-ISA) [41] and dynamic opposite learning enhanced artificial ecosystem optimizer (DAEO) [41].

In terms of the same order case, the results demonstrate the superior performance of the GOASA algorithm, as it achieves the lowest MSE value among all the compared algorithms. The best MSE value obtained by GOASA is 8.4199E-10, significantly outperforming other algorithms such as PSO, HPSO-GSA, GA, IPO, M-ISA, and DAEO. Additionally, GOASA exhibits a small standard deviation, indicating consistent and reliable results.

For the reduced order case, the GOASA algorithm again proves its effectiveness by achieving the lowest MSE value compared to the other algorithms. With a best MSE value of 3.9165E-07, GOASA surpasses the performance of PSO, HPSO-GSA, GA, IPO, M-ISA, and DAEO. The standard deviation for GOASA is also relatively small, indicating stability and consistency in the obtained results.

Overall, the statistical analysis and comparison of the reported approaches highlight the significance of the proposed GOASA algorithm for IIR system identification. It consistently outperforms other optimization algorithms in terms of reaching the lowest MSE values, demonstrating its superior capability and efficiency in accurately identifying IIR system parameters.

6. Conclusion

In conclusion, this paper presented a novel and efficient adaptive algorithm for designing digital IIR filters used in system identification. By combining the gazelle optimization algorithm and simulated annealing technique, the proposed algorithm demonstrated superior optimization performance compared to several competitive algorithms. Extensive experiments conducted on both the challenging CEC2020 benchmark functions and a fifth order IIR system identification problem showcased the effectiveness of the proposed approach. In the realm of

Table 6
Statistical performance comparison of different reported approaches.

Type	Metric	GOASA	PSO [39]	HPSO-GSA [39]	GA [40]	IPO [40]	M-ISA [41]	DAEO [41]
$H_S(z^{-1})$	Best	**8.4199E−10**	7.7470E−04	1.9663E−09	8.8033E−05	4.8542E−05	2.61E−06	8.32E−07
	Worst	**1.2312E−08**	2.3963E−02	3.3217E−08	8.8033E−05	8.1884E−05	5.58E−03	2.30E−04
	Mean	**3.3646E−09**	5.7836E−03	6.1274E−08	8.8033E−05	6.5836E−05	4.83E−04	3.58E−05
	Standard Deviation	2.7401E−09	1.0107E−02	3.4572E−09	**0**	1.1977E−05	1.08E−03	5.84E−05
$H_R(z^{-1})$	Best	**3.9165E−07**	4.2651E−04	5.5196E−06	9.0526E−05	5.8187E−05	4.25E−05	1.69E−05
	Worst	**4.5756E−07**	9.3628E−03	9.2659E−05	2.8314E−04	1.2361E−04	2.24E−03	1.46E−04
	Mean	**4.1907E−07**	2.9377E−03	5.6151E−05	1.5238E−04	8.2263E−05	3.18E−04	5.12E−05
	Standard Deviation	**1.7476E−08**	3.6667E−03	3.4011E−05	6.7789E−05	1.5695E−05	3.91E−04	3.35E−05

digital filter design and system identification, accurately modeling IIR systems has been a subject of significant interest. The proposed algorithm addressed this challenge by offering an adaptive and efficient solution for optimizing filter parameters. The statistical analysis and convergence behavior of the proposed algorithm, when compared against existing algorithms, highlighted its capability and reliability in accurately identifying IIR system parameters. The evaluation of the proposed algorithm on the CEC2020 benchmark functions revealed its effectiveness in achieving optimal solutions, surpassing several well-known optimization algorithms. Furthermore, the algorithm was applied to the same order and reduced order cases of a fifth order IIR system identification problem. The results showcased its superior performance compared to the original gazelle optimization algorithm, artificial hummingbird algorithm, mountain gazelle optimization algorithm, and the widely used artificial bee colony algorithm. With its simple and intuitive workflow, ease of use, and efficient search space exploration, the proposed algorithm holds great potential for various engineering and scientific applications. The combination of the gazelle optimization algorithm and simulated annealing technique provided a powerful hybrid approach that outperformed other algorithms in terms of optimization performance. Overall, the proposed algorithm represents a highly effective and novel solution for designing digital IIR filters used in system identification. Its superior performance, as demonstrated through extensive experiments and statistical comparisons, establishes it as a promising approach in the field of digital filter design and system identification. The algorithm's capability to accurately model IIR systems opens up new possibilities for various engineering and scientific applications, where precise system emulation is crucial.

In addition to the impressive results obtained and the potential applications highlighted, there are several future research directions that can be explored to further enhance the understanding and utilization of the proposed algorithm for designing digital IIR filters used in system identification. While the proposed algorithm has shown superior performance, there is room for investigating different strategies for parameter tuning and adaptation. Exploring adaptive techniques that dynamically adjust the algorithm's parameters during the optimization process could potentially enhance its convergence speed and robustness. The combination of the gazelle optimization algorithm and simulated annealing has proven to be highly effective. However, exploring hybridization with other metaheuristic algorithms could lead to even better optimization performance. Investigating the potential synergies between the proposed algorithm and other state-of-the-art algorithms could provide new insights and contribute to the development of advanced optimization techniques. Extending the application of the proposed algorithm to real-world problems in various domains would be an interesting avenue for future research. Testing its performance on complex and practical system identification scenarios, such as in signal processing, control systems, telecommunications, and biomedical engineering, would demonstrate its adaptability and effectiveness in real-world applications.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] S. Chen, B.L. Luk, Adaptive simulated annealing for optimization in signal processing applications, Signal Process. 79 (1999) 117–128, https://doi.org/10.1016/S0165-1684(99)00084-5.

[2] N. Karaboga, A new design method based on artificial bee colony algorithm for digital IIR filters, J. Frankl. Inst. 346 (2009) 328–348, https://doi.org/10.1016/j.jfranklin.2008.11.003.

[3] S. Zhang, Y. Zhou, Grey wolf optimizer with ranking-based mutation operator for IIR model identification, Chin. J. Electron. 27 (2018) 1071–1079, https://doi.org/10.1049/cje.2018.06.008.

[4] B. Durmuş, G. Yavuz, D. Aydın, Adaptive IIR filter design using self-adaptive search equation based artificial bee colony algorithm, Turkish J. Electr. Eng. Comput. Sci. 27 (2019) 4797–4817, https://doi.org/10.3906/elk-1809-83.

[5] P. Upadhyay, R. Kar, D. Mandal, S.P. Ghoshal, A new design method based on firefly algorithm for IIR system identification problem, J. King Saud Univ. – Eng. Sci. 28 (2016) 174–198, https://doi.org/10.1016/j.jksues.2014.03.001.

[6] D.-X. Zou, S. Deb, G.-G. Wang, Solving IIR system identification by a variant of particle swarm optimization, Neural Comput. Appl. 30 (2018) 685–698, https://doi.org/10.1007/s00521-016-2338-0.

[7] D. İzci, S. Ekinci, M. Güleydin, Application of Slime Mould Algorithm to Infinite Impulse Response System Identification Problem, Comput. Sci. IDAP-2022 (2022) 45–51, https://doi.org/10.53070/bbd.1172833.

[8] C. Dai, W. Chen, Y. Zhu, Seeker optimization algorithm for digital IIR filter design, IEEE Trans. Ind. Electron. 57 (2010) 1710–1718, https://doi.org/10.1109/TIE.2009.2031194.

[9] N. Karaboga, Digital IIR filter design using differential evolution algorithm, EURASIP J. Adv. Signal Process. 2005 (2005), 856824, https://doi.org/10.1155/ASP.2005.1269.

[10] A. Kalinli, N. Karaboga, A new method for adaptive IIR filter design based on tabu search algorithm, AEU – Int. J. Electron. Commun. 59 (2005) 111–117, https://doi.org/10.1016/j.aeue.2004.11.003.

[11] N. Karaboğa, M.B. Çetinkaya, A novel and efficient algorithm for adaptive filtering: artificial bee colony algorithm, Turkish J. Electr. Eng. Comput. Sci. (2011), https://doi.org/10.3906/elk-0912-344.

[12] R. Zhao, Y. Wang, C. Liu, P. Hu, H. Jelodar, C. Yuan, Y. Li, I. Masood, M. Rabbani, H. Li, B. Li, Selfish herd optimization algorithm based on chaotic strategy for adaptive IIR system identification problem, Soft Comput. 24 (2020) 7637–7684, https://doi.org/10.1007/s00500-019-04390-9.

[13] M. Kaur, R. Kaur, N. Singh, A novel hybrid of chimp with cuckoo search algorithm for the optimal designing of digital infinite impulse response filter using high-level synthesis, Soft Comput. 26 (2022) 13843–13867, https://doi.org/10.1007/s00500-022-07410-3.

[14] B. Durmuş, Infinite impulse response system identification using average differential evolution algorithm with local search, Neural Comput. Appl. 34 (2022) 375–390, https://doi.org/10.1007/s00521-021-06399-4.

[15] A. Mohammadi, S.H. Zahiri, IIR model identification using a modified inclined planes system optimization algorithm, Artif. Intell. Rev. 48 (2017) 237–259, https://doi.org/10.1007/s10462-016-9500-z.

[16] W. Zhu, J. Fang, Y. Tang, W. Zhang, W. Du, Digital IIR filters design using differential evolution algorithm with a controllable probabilistic population size, PLoS One 7 (2012) e40549, https://doi.org/10.1371/journal.pone.0040549.

[17] Q. Luo, Y. Ling, Y. Zhou, Modified whale optimization algorithm for infinitive impulse response system identification, Arab J. Sci. Eng. 45 (2020) 2163–2176, https://doi.org/10.1007/s13369-019-04093-1.

[18] M. Kumar, T.K. Rawat, A. Aggarwal, Adaptive infinite impulse response system identification using modified-interior search algorithm with Lèvy flight, ISA Trans. 67 (2017) 266–279, https://doi.org/10.1016/j.isatra.2016.10.018.

[19] A. Liu, P. Li, X. Deng, L. Ren, A sigmoid attractiveness based improved firefly algorithm and its applications in IIR filter design, Conn Sci 33 (2021) 1–25, https://doi.org/10.1080/09540091.2020.1742660.
[20] M. Shafaati, H. Mojallali, IIR filter optimization using improved chaotic harmony search algorithm, Automatika 59 (2018) 331–339, https://doi.org/10.1080/00051144.2018.1541643.
[21] Y. Yang, B. Yang, M. Niu, Adaptive infinite impulse response system identification using opposition based hybrid coral reefs optimization algorithm, Appl. Intell. 48 (2018) 1689–1706, https://doi.org/10.1007/s10489-017-1034-9.
[22] S. Chen, B.L. Luk, Digital IIR filter design using particle swarm optimisation, Int. J. Model. Identif. Control 9 (2010) 327, https://doi.org/10.1504/IJMIC.2010.033208.
[23] M. Kumar, A. Aggarwal, T.K. Rawat, Bat algorithm: application to adaptive infinite impulse response system identification, Arab J. Sci. Eng. 41 (2016) 3587–3604, https://doi.org/10.1007/s13369-016-2222-3.
[24] E. Rashedi, H. Nezamabadi-pour, S. Saryazdi, Filter modeling using gravitational search algorithm, Eng. Appl. Artif. Intell. 24 (2011) 117–122, https://doi.org/10.1016/j.engappai.2010.05.007.
[25] J.O. Agushaka, A.E. Ezugwu, L. Abualigah, Gazelle optimization algorithm: a novel nature-inspired metaheuristic optimizer, Neural Comput. Appl. (2022), https://doi.org/10.1007/s00521-022-07854-6.
[26] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science (1979) 220 (1983) 671–680, https://doi.org/10.1126/science.220.4598.671.
[27] D. Izci, S. Ekinci, A.G. Hussien, Effective PID controller design using a novel hybrid algorithm for high order systems, PLoS One 18 (2023), e0286060, https://doi.org/10.1371/journal.pone.0286060.
[28] D. Izci, S. Ekinci, B. Hekimoğlu, Fractional-order PID controller design for buck converter system via hybrid Lévy flight distribution and simulated annealing algorithm, Arab J. Sci. Eng. 47 (2022) 13729–13747, https://doi.org/10.1007/s13369-021-06383-z.
[29] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, A.H. Gandomi, The arithmetic optimization algorithm, Comput. Methods Appl. Mech. Eng. 376 (2021), 113609, https://doi.org/10.1016/j.cma.2020.113609.
[30] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of ICNN'95-International Conference on Neural Networks, IEEE, 1995, pp. 1942–1948.
[31] R. Storn, K. Price, Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces, J. Glob. Optim. 11 (1997) 341–359, https://doi.org/10.1023/A:1008202821328.
[32] S. Mirjalili, SCA: a sine cosine algorithm for solving optimization problems, Knowl. Based Syst. 96 (2016) 120–133, https://doi.org/10.1016/j.knosys.2015.12.022.
[33] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61, https://doi.org/10.1016/j.advengsoft.2013.12.007.
[34] D. Simon, Biogeography-Based Optimization, IEEE Trans. Evol. Comput. 12 (2008) 702–713, https://doi.org/10.1109/TEVC.2008.919004.
[35] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp swarm algorithm: a bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (2017) 163–191, https://doi.org/10.1016/j.advengsoft.2017.07.002.
[36] W. Zhao, L. Wang, S. Mirjalili, Artificial hummingbird algorithm: a new bio-inspired optimizer with its engineering applications, Comput. Methods Appl. Mech. Eng. 388 (2022), 114194, https://doi.org/10.1016/j.cma.2021.114194.
[37] B. Abdollahzadeh, F.S. Gharehchopogh, N. Khodadadi, S. Mirjalili, Mountain gazelle optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems, Adv. Eng. Softw. 174 (2022), 103282, https://doi.org/10.1016/j.advengsoft.2022.103282.
[38] D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm, Appl. Soft Comput. J. 8 (2008) 687–697, https://doi.org/10.1016/j.asoc.2007.05.007.
[39] S. Jiang, Y. Wang, Z. Ji, A new design method for adaptive IIR system identification using hybrid particle swarm optimization and gravitational search algorithm, Nonlinear Dyn. 79 (2015) 2553–2576, https://doi.org/10.1007/s11071-014-1832-0.
[40] A. Mohammadi, S.H. Zahiri, S.M. Razavi, Infinite impulse response systems modeling by artificial intelligent optimization methods, Evol. Syst. 10 (2019) 221–237, https://doi.org/10.1007/s12530-018-9218-z.
[41] Y. Niu, X. Yan, Y. Wang, Y. Niu, Dynamic opposite learning enhanced artificial ecosystem optimizer for IIR system identification, J. Supercomput. 78 (2022) 13040–13085, https://doi.org/10.1007/s11227-022-04367-w.
[42] I. Karatzas, S.E. Shreve, Brownian motion, in: 1998: pp. 47–127. https://doi.org/10.1007/978-1-4612-0949-2_2.
[43] D. Izci, S. Ekinci, S. Mirjalili, L. Abualigah, An intelligent tuning scheme with a master/slave approach for efficient control of the automatic voltage regulator, Neural Comput. Appl. (2023), https://doi.org/10.1007/s00521-023-08740-5.
[44] R.N. Mantegna, Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes, Phys Rev E 49 (1994) 4677–4683, https://doi.org/10.1103/PhysRevE.49.4677.
[45] K.A. Olson, E.A. Larsen, T. Mueller, P. Leimgruber, T.K. Fuller, G.B. Schaller, W. F. Fagan, Survival probabilities of adult Mongolian gazelles, J. Wildl Manag. 78 (2014) 35–41, https://doi.org/10.1002/jwmg.640.
[46] S.K. Saha, R. Kar, D. Mandal, S.P. Ghoshal, Harmony search algorithm for infinite impulse response system identification, Comput. Electr. Eng. 40 (2014) 1265–1285, https://doi.org/10.1016/j.compeleceng.2013.12.016.
[47] G. Panda, P.M. Pradhan, B. Majhi, IIR system identification using cat swarm optimization, Expert Syst. Appl. 38 (2011) 12671–12683, https://doi.org/10.1016/j.eswa.2011.04.054.



Serdar Ekinci received his B.Sc. degree in Control Engineering, and his M.Sc. and Ph.D. degrees in Electrical Engineering all from Istanbul Technical University (ITU), in 2007, 2010 and 2015, respectively. He is currently an Associate Professor working in Department of Computer Engineering at Batman University, Turkey. His areas of interest are electrical power systems, stability, control technology and the applications of metaheuristic optimization algorithms to various control systems.



Davut Izci received his B.Sc. degree from Dicle University, Turkey, in Electrical and Electronic Engineering and his M.Sc. and Ph.D. degrees from Newcastle University, England - UK, in Mechatronics and Microsystems, respectively. He is currently an Associate Professor working on optimization, control system design, sensing applications, energy harvesting, microsystems development and applications of metaheuristic optimization techniques to different control systems and real-world engineering problems.