Journal of Innovative Technology, Vol. 6, No. 1, pp. 13-24, March 2024
http://doi.org/10.29424/JIT.202403_6(1).0002

Intelligent heuristic computing paradigm: A novel strategy to singular differential difference equations analysis

Nabeela Anwar [1], Muhammad Junaid Ali Asif Raja [2], Iftikhar Ahmad [3], Adiqa Kausar Kiani [4]

ABSTRACT

n the presented study, a novel intelligent heuristic computing paradigm based on artificial intelligence is introduced for solving linear singular differential difference equations (SDDEs). Unsuper-vised artificial neural networks (ANNs) with universal function approximation capabilities are employed to establish a mathematical model for the problem, incorporating a mean squared error function. The design parameter training for ANN models involves utilizing the global search capabilities of a genetic algorithm (GA), an effective local search through an interior point approach (IPA), and hybridization of GA-IPA. The precision, reliability, and robustness of the proposed methodology are endorsed through comprehensive comparative analysis against exact solutions and previously reported findings in the lit-erature. Statistical assessments of the outcomes are employed to confirm the accuracy and convergence of the design approach. Multiple independent runs of these algorithms are also conducted and compared against approximate numerical solutions to ensure accuracy and convergence.

Keywords: Heuristic computing paradigm; Differential difference equations; Artificial neural networks; Interior point approach; Genetic algorithm.

1. INTRODUCTION

Numerous real-world phenomena have been successfully modelled using differential difference equations (DDEs) with delay or advanced terms in engineering, economics, control theory, and other fields [1]. Ad- ditionally, DDEs have found applications in areas such as epidemiological and population dynamics modeling [2–4]. SDDEs are difficult to handle analytically. Therefore in recent years research community took serious consideration to investigate the numerical techniques for solving the linear SDDEs. Although only a few techniques are existing to tackle these problems analytically and numerically due to singularities at different points. A number of numerical methods were used, including modified Hermite operation matrix technique [5], Taylor series method [6], variational iteration method [7], Maclaurin series method [8], fitted finite difference method [9], Lagrange operational matrix technique [10], quadratic B-spline approxima- tion [11], extended cubic B-spline collocation method [12], homotopy analysis method [13], optimal homo-topy asymptotic method [14], reproducing kernel discretization technique [15], modified Legendre spectral method [16], finite difference method [17–19], Legendre wavelets spectral scheme [20], modified Adomian decomposition method [21, 22], Fibonacci operational matrix approach [23], nonconforming virtual el- ement technique [24], successive differentiation method [25], homotopy perturbation method [26–28]. Moreover, Ramos et al. presented a pair of optimized hybrid block techniques to incorporate singular initial value problems of second order [29]. Seiler et al. explored the presence, smoothness, and multiplic- ity of solution for initial value problems (IVPs) associated with scalar quasi-linear ordinary differential equations (ODEs). These solutions are characterized by an impasse point in the context of the given initial condition. [30]. Daba et al. presented a collocation method for determine the numerical solution of time dependent differential difference equations (DDEs) having small shifts [31]. The robust numerical approach for solving the semilinear partial singularly perturb DDE having spatial delay was proposed by Kabeto et al. The semilinear term is linearized using the quadratically convergent quasilinearization approach. It's described as a system of algebraic equations after discretizing the solution domain and substituting the differential equation with a finite difference approximation [32]. Xiao et al. investigated the positivity as well as asymptotic stability for a DDE having the delay in time. The authors obtained the necessary as well as sufficient conditions in the case of bounded delay on the basis of comparison principle and timescale theory while sufficient criteria are achieved in the case of unbounded delay [33]. Yu¨zba¸sı introduced a numerical approach for solving high order linear SDDEs that gives an approximate polynomial solution [34].

The linear SDDE in generic form can be expressed as [34]:

Manuscript received October 25, 2023; revised December 25, 2023; accepted January 13, 2024.

[1] Lecturer, Department of Mathematics, University of Narowal, Narowal 51600, Pakistan.
[2] Student, School of Electrical Engineering and Computer Science, National University of Sciences and Technology, H-12, Islamabad, Pakistan 44000.
Department of Computer Science and Information Engineering, National Yunlin University of Science and Technology, Douliu City, Yunlin County, Taiwan 64002.
[3] Professor, Department of Mathematics, University of Gujrat, Gujrat 50700, Pakistan.
[4] Professor, Future Technology Research Center, National Yunlin University of Science and Technology, 123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C.(email: adiqa@yuntech.edu.tw).

$$\sum_{p=0}^{m}\sum_{q=0}^{n} P_{\beta\gamma}(r)y^{(\gamma)}(\alpha_{\beta\gamma}r+\delta_{\beta\gamma})-r(r-c)f(r)=0, \qquad 0\le r\le c \quad (1)$$

along with boundary conditions:

$$\sum_{q=0}^{n-1}\left[a_{\zeta\gamma}y^{(\gamma)}(0)+c_{\zeta\gamma}y^{(\gamma)}(c)\right]=\lambda_\zeta \qquad ;\zeta=0,1,\dots,n-1 \qquad (2)$$

whereas $\alpha_{\beta\gamma}$, $\delta_{\beta\gamma}$, $a_{\zeta\gamma}$, $c_{\zeta\gamma}$ and λ_ζ are finite constants, $y^{(0)}(r) = y(r)$ is an unknown function, and $P_{\beta\gamma}(r), f(r) \in C(0, c)$, and $P_{\beta\gamma}(r)$, $f(r)$ are not defined functions at the points $r = 0$ and $r = c$ [34].

Recently, Numerous researchers have directed their efforts towards harnessing the capabilities of ANNs to tackle linear and nonlinear models encountered in a diverse range of fields such as applied mathematics, physics, and technology, as cited in references [35–39]. The well-established potential of neural networks for comprehensive function approximation is further enhanced by the development of both local and global search strategies. These approaches are particularly valuable for solving linear and nonlinear differential equations, as well as differential difference-based systems. According to the referenced literature, some examples of applications include difficulties in nanotechnology, electromagnetic theory, fluid dynamics, nonlinear influenza models, SVEIR models with vaccination, tumour modelling, wireless sensor networks, avian influenza systems, plant virus systems, and nonlinear corneal shape modelling [40–50] mentioned therein. However, such a stochastic solver has not yet been implemented for an effective and reliable solution of stiff and non-stiff SDDEs. Therefore, the above mentioned characteristics are the insipirations for authors to intoduce a new ANNs based differential difference model optimized with hybrid computing mechanism for finding an accurate, alternate, robust and stable solution for the SDDEs. However, various research workers handled linear SDDEs using different techniques, and some authors employed both analytical as well as numerical methods. In this research, we used a soft computing approach to solve linear SDDEs, and the proposed scheme provides reliable convergence as compared to the other techniques in existing literature.

The proposed study's key attributes are briefly emphasized through the following notable characteristics:

- A novel intelligent heuristic computing paradigm has been introduced for solving the SDDEs, com- bining the global search capabilities of genetic algorithms with the local search expertise of interior point algorithms.
- The computational process integrates modeling and optimization techniques, utilizing ANNs, GAs, IPA, and their hybrid GA–IPA, to derive precise solutions for the governing relations of SDDEs.
- Convergence analysis is performed across multiple independent trials to validate the efficacy of stochastic methods in achieving robust, reliable, and consistent solutions for SDDEs.
- Statistical evaluations of the results are utilized to validate the precision and the convergence of the designed intelligent heuristic computing paradigm.

The remaining portion of this paper unfolds as follows: In Section 2, we illustrate the mathematical modeling of SDDEs, employing artificial neural networks (ANNs). Section 3 provides an overview of the methodologies employed to solve SDDEs using ANNs. In Section 4, we present a comprehensive discussion of simulation-based results and their implications. Section 5 is dedicated to a thorough statistical analysis, aimed at validating the accuracy and reliability of the computed results. Lastly, in Section 6, we draw our conclusions.

2. ANN'S MATHEMATICAL MODELING

The mathematical modelling for linear SDDEs are provided in two parts: the 1st part develops ANNs model for the solution of SDDEs and the derivative terms, and the 2nd part describes fitness function composition using ANN models.

ANNs based mathematical models for SDDEs are developed as continuous mappings for the solution $y(r)$, its 1st, 2nd, and so on nth order derivatives, as follows:

$$\begin{cases}\hat{y}(r)=\sum_{k=1}^{n}\mu_k g(\omega_k r+\nu_k)\\ \frac{d^\gamma\hat{y}}{dr^\gamma}=\sum_{k=1}^{n}\mu_k\frac{d^\gamma}{dt^\gamma}g(\omega_k r+\nu_k),\end{cases} \qquad (3)$$

where n denotes the number of nerons, g is known as the activation function, μ, ω, ν are real valued adaptive parameters. Mathematical modeling can be constructed through log sigmoid function g_{ls} used as transfer function given as:

$$g_{ls}(r)=\frac{1}{1+e^{-r}}. \qquad (4)$$

Log sigmoid function g_{ls} is used to construct ANNs architecture for SDDEs to approximate solution $\hat{y}(r)$.

$$\hat{y}(r)=\sum_{k=1}^{n}\frac{\mu_k}{1+e^{(-\omega_k r-\nu_k)}} \qquad (5)$$

$$\frac{d\hat{y}}{dr}=\sum_{k=1}^{n}\mu_k\omega_k\left[\frac{e^{-(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_i)})^2}\right] \qquad (6)$$

$$\frac{d^2\hat{y}}{dr^2}=\sum_{k=1}^{n}\mu_k\omega_k^2\left[\frac{2e^{-2(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^3}-\frac{e^{-(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^2}\right] \qquad (7)$$

$$\frac{d^3\hat{y}}{dr^3}=\sum_{k=1}^{n}\mu_k\omega_k^3\left[\frac{6e^{-3(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^4}-\frac{6e^{-2(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^3}-\frac{e^{-(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^2}\right] \qquad (8)$$

$$\vdots$$

$$\frac{d^\gamma\hat{y}}{dr^\gamma}=\sum_{k=1}^{n}\mu_k\omega_k^\gamma\left[\frac{2\gamma e^{-\gamma(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^{\gamma+1}}-\frac{2\gamma e^{-(\gamma-1)(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^{\gamma}}-\dots-\frac{e^{-(\omega_k r+\nu_k)}}{(1+e^{-(\omega_k r+\nu_k)})^2}\right]. \qquad (9)$$

The fitness function for SDDE is formulated by defining an unsupervised error can be expressed as:

$$e_{SDDE}=e_1+e_2, \qquad (10)$$

where e1 corresponds to differential equation and e_2 corresponds to the initial conditions can be expressed as:

$$e_1=\frac{1}{N+1}\sum_{i=0}^{N}\left[\sum_{p=0}^{m}\sum_{q=0}^{n}P_{\beta\gamma}(r_i)\hat{y}^{(\gamma)}\left(\alpha_{\beta\gamma}(r_i)+\delta_{\beta\gamma}\right)-r_i(r_i-c)f(r_i)\right]^2 \qquad (11)$$

$$e_2=\frac{1}{n}\left[\sum_{q=0}^{n-1}\left(a_{\zeta\gamma}\hat{y}^{(\gamma)}(0)+c_{\zeta\gamma}\hat{y}^{(\gamma)}(c)\right)-\lambda_\zeta\right]^2, \qquad (12)$$

here $r \in (0, c)$, $\zeta = 0, 1, \ldots, n - 1$, $N = \frac{1}{h}$, $\hat{u}_i = \hat{u}(ri)$, $ri = ih$ and $'h'$ represents the step size.

So, the networks mentioned in equations (5) to (9) approximate the $\hat{y}(r)$ solutions when it approaches to proposed solution $y(r)$. If the weights of the networks are appropriate and eSDDE approaches zero, the approximate solution $\hat{y}(r)$ coincides with the exact solution $y(r)$ of SDDE.

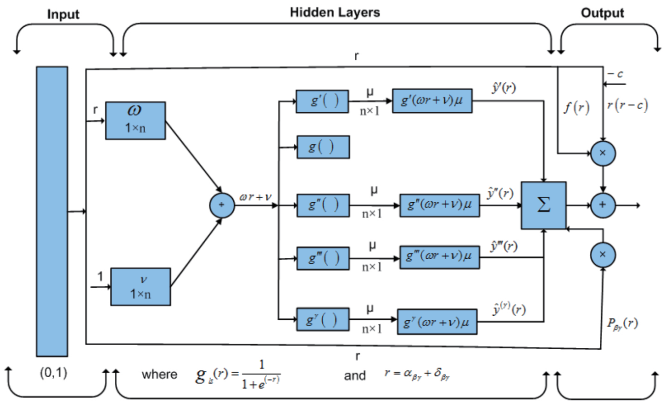

Fig. 1 Neural network architecture for the SDDEs

3. LEARNING METHODOLOGIES

The interior point algorithm (IPA), genetic algorithm (GA), and hybridization of GA-IPA based optimization mechanisms are used to learn the unknown variables of the networks, and the weights of these approaches are utilized in ANNs models to generate the approximate numerical results of linear SDDEs. This section briefly describes an overview of optimization techniques that are utilized as adaptive processes for learning weights of ANNs models is presented. Figure 1 illustrates the overall procedure of the design approach.

3.1 INTERIOR POINT ALGORITHM

IPA was firstly presented in the form of barrier methods in early 1960s. IPA build upon Karmarkar's algorithm. In 1984 Narendra Karmarkar was introduced IPA to deal with the linear programming approaches [55]. IPA is group of algorithms, which solve constrained optimization problems [57, 58] in engineering as well as in applied science. Also research on economic dispatch problem [59] along with multi area optimal reactive power flow [60] are in focus of same period of time. To cover up the convex set, the capital coefficients of interior point algorithms rely on self responsive barrier functions. Similar to the conventional simplex approach, the most acceptable solution is pursued by infusing the interior of the practicable region [61]. With either the Newton step in conjunction with a linear programming technique or the conjugate-gradient step demonstrating a reliance zone, IPA is used to solve approximation problems. Interior point techniques have been employed effectively in several examples that may be clarified by a column generating process. Wets and Van Slyke's method of L-shaped decomposition [63] for stochastic programming approaches have been employed by Bahn et al., [62]. Goffin et al., found the solution of non differentiable optimization problems [64] as well as multi commodity network flow problems iterating an interior algorithm column generation technique [65].

Table 1 Settings for the IPA parameters in the fmincon function

Parameters	Settings
Function tolerance	zero
Maximum iteration	999
Hessian	BFGS
Max. function evaluations	1000000
X tolerance	10^{-12}
Nonlinear constraint tolerance	zero
Initial point generation	(0,1)
Initial point size	30
Fin. differences type	Forward difference

3.2 GENETIC ALGORITHMS

GA consists of the postulation of natural collection and genetics.. The preliminary research suggests that the general representation is ineffective for solving dense problems in genetic algorithms in a dynamic and adaptive manner. In large-scale optimization problems, the time it takes to develop first-generation GAs increases dramatically, while the quality of the solutions decreases. Selection methods, concealing techniques, knowledge-based, and self-adaptive operators, among other aspects, play a significant role in the acquisition of extremely convex and fuzzy nature problems. Since its inception, genetic algorithms have been used to optimise adaptable domains of interest [66, 67]. GAs also used as a efficient approach for execute simulation, modeling, forecasting, control etc., As a viable alternative for tackling complex nonlinear systems described by differential equations, and with a focus on showcasing successful applica tions that illustrate the methodology and highlight the advantages of the proposed approach [68–76].

Neural networks model are trained for optimization using matlab builtin functions GA and fmincon in graphical user interface . Parameter settings for IPA and GA are listed in Table 1 and Table 2 respec tively. The step-by-step procedure of the GA-IPA can be outlined as follows:

Algorithm Initialization: Population is generated randomly with real values work as initial point for the algorithm. In the neural network model for SDDEs, each neuron possesses the same number of elements as the count of unknown weights.

Prior to initiating the algorithm, it's crucial to take into account the parameter values provided in Table 1.

Fitness Calculation: Calculation of fitness value is made for each individual's population by using equa tion (10) for neural networks model.

Classification: Classify each individual of the populations based on the lowest value of corresponding fitness function of the neural networks model.

Decisive Criteria: If any of following criteria matches, stop processing the algorithm :

- $\in 10^{-15}$
- The value of generations as setting in Table 1 is achieved
- Non-constraint tolerance and function tolerance is accomplished

If Decisive criterion is achieved liable in Table 1, then go to step 7, otherwise go to step 6 Reproduction: New population is produced at each cycle based on crossover, mutation and selection functions as given in Table 1

Hybridization: For rectification in results interior point algorithm is used by choosing of the perfect individual. Initial weights comprise on GAs results.

Table 1 shows the IPA parameter settings.

Accumulation: For each execution of the solver, accumulate the final fitness value and weight vector. The generic flowchart of overall process of GA-IPA is elaborated in Fig. 2. The values of parameters for IPA and GA are elaborated in Tables 1 and 2, as well as ANN structure based on the input/hidden/output layer is presented in Fig. 1, these settings not only used to reproduce the results illustrated numerically and graphically in the article, but also represent one of the best structure of ANN hybrid with GA-IPA to be exploited by the research community in their future studies relevant to this work.

4. RESULTS AND DISCUSSION

The numerical results along with graphical representation of the homogeneous and inhomogeneous initial value problems of SDDEs are presented in this section. The neural networks model are optimized with IPA, GA and their hybridization of GA-IPA and prove the worth of present scheme by comparing with

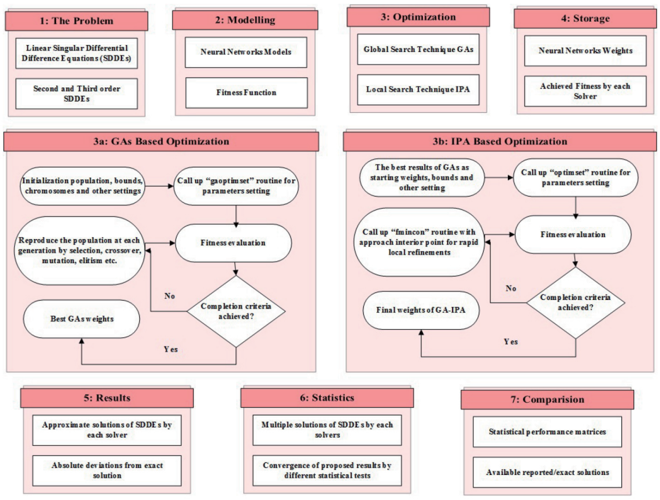

Fig. 2 Procedural steps of hybrid approach GA-IPA

Table 2 Parameter settings for GA algorithm

Parameters	Settings
Population size	[30 30 30 30 30 30 30 30 30 30]
Population type	Double vector
Scaling function	Rank
Function of mutation	Adaptive feasible
Minimum perturbation	10^{-09}
Hybrid function	FMINCON
Elite count	2, 3
Crossover fractions	0.80
Migration fractions	0.3
Size of chromosomes	40
Stall generation	60, 90
Bounds values	(−9, 9)
Initial penalty	10
Migration direction	Forward
Function Tolerance	10^{-12}, 10^{-20}
Generations	100000
Crossover function	Heuristic, scattered
Penalty factor	100
Fitness limit	10^{-15}
Stall generations	50, 100
Non-constraints tolerance	10^{-10}
Sub population	20
Selection function	Stochastic uniform, uniform
Others	Default

aforementioned solutions in literature and exact solution as well. Statistical analysis of the design scheme depend on adequately large number of independent runs is presented for each problem.

4.1 INHOMOGENEOUS LINEAR SINGULAR DIFFERENTIAL DIFFERENCE EQUATION

Consider the IVP of inhomogeneous linear SDDE as [34]:

$$\begin{cases} y^{(3)}(2r-1) - \frac{5}{r(r+1)}y^{(2)}(r+2) + \frac{4}{r(r+3)}y^{(1)}(r-1) + ry(r+1) \\ = sin(2r-1) + \frac{5}{r(r+1)}cos(r+2) - \frac{4}{r(r+3)}sin(r-1) + rcosr(r+1) \\ y(0)=1, \quad y'(0)=0 \quad and \quad y''(0)=-1, \end{cases} \tag{13}$$

where $r \in [0, 1]$, and $y(r) = \cos r$ represents the exact solution. So the fitness function is constructed by using equation (10) is as follows:

$$e_1 = \frac{1}{11}\sum_{i=0}^{10}\Big[r_i\hat{r}_i\bar{r}_i\frac{d^3\breve{y}_i}{dr^3} - 5\breve{r}_i\frac{d^2\bar{r}_m}{dr^2} + 4\hat{r}_i\frac{d\tilde{y}_i}{dr} + r_i^2\hat{r}_i\breve{r}_i\hat{y}_i - r_i\hat{r}_i\breve{r}_i sin\breve{y}_i - 5\breve{r}_i cos\bar{y}_i + 4\hat{r}_i sin\tilde{y}_i - r_i^2\hat{r}_i\breve{r}_i cos\hat{y}_i\Big]^2 \tag{14}$$

$$e_2 = \frac{1}{3}\Big[\big(\hat{y}(0)-1\big)^2 + \big(\frac{d}{dr}\hat{y}(0)\big)^2 + \big(\frac{d^2}{dr^2}\hat{y}(0) + 1\big)^2\Big], \tag{15}$$

where $\breve{y}_i = \hat{y}(2r_i+1)$, $\bar{y}_i = \hat{y}(r_i+2)$, $\tilde{y}_i = \hat{y}(r_i-1)$, $\hat{y}_i = \hat{y}(r_i+1)$, $r_i = ih$, $r_i+1 = \hat{r}$, $r_i+2 = \bar{r}_i$, $r_i+3 = \breve{r}_i$. We executed the optimized solvers using built-in MATLAB functions by configuring the parameter values outlined in both Table 1 and Table 2. The optimized solvers, which include IPA, GA, and the hybrid GA-IPA approach, were employed to train the neural network weights, with the aim of approximating the solution for SDDEs. This process was based on our designed methodology that utilizes unsupervised neural network models for Equation (13), with Equations (14) and (15) serving as fitness functions for optimization. We employed a network configuration featuring 10 neurons and utilized a step size of $h = 0.1$. The equations described in (5-8) were employed to derive the proposed solution $\hat{y}(r)$ for Equation (13).The graphical view of optimal weights can be seen in Fig. 3 for one specific result of the design scheme with corresponding fitness is as follows:

$$\begin{aligned} \hat{y}_{IPA}(r) = &-\frac{2.6610}{1+e^{-(-0.4650r-0.3239)}} - \frac{0.6912}{1+e^{-(-0.5286r-0.0187)}} + \frac{1.1054}{1+e^{-(1.5477r-2.3054)}} \\ &+ \frac{1.3894}{1+e^{-(1.2735r-0.9973)}} - \frac{2.8831}{1+e^{-(1.0385r+1.9710)}} - \frac{2.1657}{1+e^{-(1.1401r+0.8754)}} \\ &+ \frac{1.3328}{1+e^{-(-1.2812r-3.5620)}} + \frac{4.1728}{1+e^{-(1.0066r+4.1305)}} + \frac{1.3603}{1+e^{-(0.4649r+1.6706)}} \\ &- \frac{1.0951}{1+e^{-(1.2228r+1.9011)}} \end{aligned} \tag{16}$$

$$\begin{aligned} \hat{y}_{GA}(r) = &\frac{4.3559}{1+e^{-(-1.1359r-1.9546)}} + \frac{0.1704}{1+e^{-(1.9386r+1.2346)}} - \frac{2.2973}{1+e^{-(-0.5643r-4.1828)}} \\ &- \frac{2.4648}{1+e^{-(-1.3594r+5.8062)}} + \frac{2.0275}{1+e^{-(1.1040r+1.5025)}} + \frac{1.6066}{1+e^{-(1.4821r+2.6629)}} \\ &- \frac{0.1384}{1+e^{-(0.6240r+3.5199)}} - \frac{1.4449}{1+e^{-(-0.2774r-1.1806)}} - \frac{0.0774}{1+e^{-(-0.9827r-1.0836)}} \\ &- \frac{0.3809}{1+e^{-(1.6596r+1.1717)}} \end{aligned} \tag{17}$$

$$\begin{aligned} \hat{y}_{GA-IPA}(r) = &-\frac{2.3583}{1+e^{-(1.1858r+2.9581)}} + \frac{3.1885}{1+e^{-(0.2280r-0.1316)}} - \frac{3.20128}{1+e^{-(-1.2482r+1.3988)}} \\ &- \frac{3.6279}{1+e^{-(-1.0967r+4.5850)}} - \frac{0.5610}{1+e^{-(1.6250r-0.2325)}} + \frac{1.6938}{1+e^{-(1.1239r+0.8550)}} \\ &- \frac{3.2903}{1+e^{-(0.1655r-1.4183)}} - \frac{0.1874}{1+e^{-(-1.2187r+0.0210)}} - \frac{0.5113}{1+e^{-(-1.0759r+2.0101)}} \\ &+ \frac{0.5668}{1+e^{-(-1.7507r+0.1073)}} \quad (18) \end{aligned} \tag{18}$$

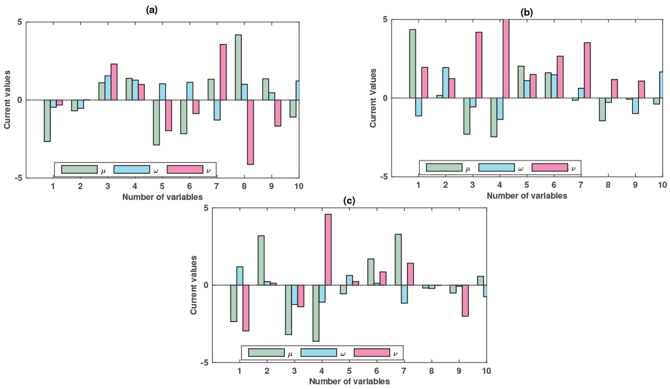

Fig. 3 Trained weights for ANNs Using IPA, GA, and GA-IPA

Table 3 Parameter settings for GA algorithm

r	yExt	IPA	GA	GA − IPA
0.0	1.000000	0.999922	0.999686	0.999988
0.1	0.995004	0.994921	0.994455	0.994984
0.2	0.980067	0.979976	0.979333	0.980017
0.3	0.955336	0.955238	0.954493	0.955226
0.4	0.921061	0.920960	0.920198	0.920851
0.5	0.877583	0.877494	0.876799	0.877237
0.6	0.825336	0.825286	0.824732	0.824827
0.7	0.764842	0.764870	0.764517	0.764149
0.8	0.696707	0.696862	0.696752	0.695824
0.9	0.621610	0.621954	0.622105	0.620545
1.0	0.540302	0.540906	0.541311	0.539075

We calculated approximate numerical solutions for neural network models using three different methods: IPA, GA, and hybridization of GA-IPA. These results are presented in Table 3, and the computations were performed for input values t belongs to [0, 1] having a step size of $h = 0.1$. We also assessed the results for the same input parameters as those outlined in Table 3. Our findings indicate that the proposed solution closely aligns with the exact solution, exhibiting a high degree of accuracy, typically within a range of 5 to 6 decimal places. Further, the graphical representation obtained by the neural networks model in Fig. 4(a) are persistently overlap the exact solution, which certified the precision of the proposed scheme. We also noticed that the Absolute Errors (AEs) for the IPA, GA, and GA-IPA methods fall within the ranges of approximately $10^{-07} \rightarrow 10^{-09}$, $10^{-06} \rightarrow 10^{-09}$, and $10^{-06} \rightarrow 10^{-10}$, respectively, as indicated in Table 4. In contrast, the Bessel Matrix Method (BMM), as reported in the literature [34], exhibits AEs within the range of approximately $10^{-04} \rightarrow 10^{-06}$.

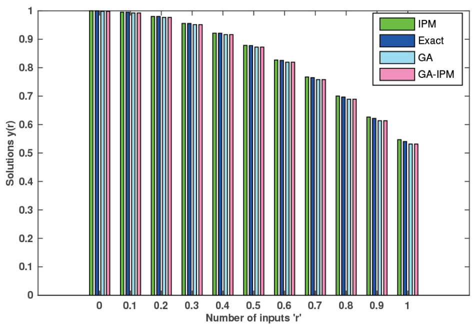

Fig. 4 Comparison of ANNs results for the Proposed Results and Exact Solution in Problem I

Table 4 Comparison of the proposed solutions in terms of AE for Problem I

r	Proposed results $\hat{y}$			BMM results in literature
	IPA	GA	GA-IPA	BMM [34]
0.0		9.88E-08	1.55E-10	0.0E+0
0.2	8.25E-09	5.38E-07	2.42E-09	2.46E-006
0.4	1.03E-08	7.45E-07	4.41E-08	1.81E-005
0.6	2.50E-09	3.64E-07	2.60E-07	5.59E-005
0.8	2.41E-08	2.03E-09	7.79E-07	1.20E-004
1.0	3.64E-07	1.02E-06	1.51E-06	2.12E-004

4.2 HOMOGENEOUS LINEAR SINGULAR DIFFERENTIAL DIFFERENCE EQUATION

Consider the IVP of homogeneous linear SDDE as [34]:

$$\begin{cases} y''(r) + \frac{2}{r}y'(r) - (4r^2 - 6)y(r) = 0 \\ y(0) = 1, y'(0) = 0 \end{cases} \tag{19}$$

The exact solution for the IVP is expressed as $y(r) = e^{r^2}$. We applied the same methodology as in the
previous problem, but the fitness function specifically designed for this problem is expressed as follows:

$$e_1 = \frac{1}{11}\sum_{i=0}^{10}[r_i \frac{d^2\hat{y}_i}{dr^2} + 2\frac{d\hat{y}_i}{dr} - r_i(4{r_i}^2 + 6)\hat{y}_i]^2 \tag{20}$$

$$e_2 = \frac{1}{2}[(\hat{y}(0) - 1)^2 + (\frac{d}{dr}\hat{y}(0))^2] \tag{21}$$

To optimize the fitness function described in Equation (20), we employed three different optimization solvers, namely IPA, GA, and GA-IPA. The results obtained from a single run of our design approach are visually represented in Figure (5). These optimized weights, which are determined through the optimization process, are then applied in Equation (19) to produce approximate solutions for Singular Differential Difference Equations (SDDEs) as follows:

$$\begin{aligned} \hat{y}_{IPA}(r) = &\frac{2.5455}{1+e^{-(3.7834r+4.3678)}} + \frac{1.9666}{1+e^{-(-0.7591r+2.5292)}} + \frac{5.6081}{1+e^{-(-2.0555r+2.4883)}} \\ &- \frac{7.9503}{1+e^{-(0.5084r+2.7617)}} + \frac{5.1209}{1+e^{-(-0.1289r+0.1671)}} + \frac{8.4514}{1+e^{-(0.7358r+0.3010)}} \\ &- \frac{8.7482}{1+e^{-(-6.0627r+9.000)}} - \frac{1.7815}{1+e^{-(0.2274r+5.3137)}} - \frac{0.4965}{1+e^{-(2.6012r+0.1351)}} \\ &+ \frac{3.9535}{1+e^{-(-0.3266r-4.5256)}} \end{aligned} \tag{22}$$

$$\begin{aligned} \hat{y}_{GA}(r) = &-\frac{3.3015}{1+e^{-(1.5249r-1.4747)}} + \frac{1.3123}{1+e^{-(1.1874r+1.2643)}} + \frac{2.5439}{1+e^{-(-1.2131r+0.4285)}} \\ &+ \frac{1.9182}{1+e^{-(2.5635r-1.6194)}} + \frac{2.0809}{1+e^{-(2.4381r-1.5711)}} + \frac{1.1448}{1+e^{-(2.7686r-2.0187)}} \\ &- \frac{2.2713}{1+e^{-(2.3078r+2.3174)}} + \frac{9.0000}{1+e^{-(6.0958r-7.3157)}} + \frac{0.7042}{1+e^{-(0.3445r+0.8880)}} \\ &- \frac{3.5931}{1+e^{-(7.1662r-8.0147)}} \end{aligned} \tag{23}$$

$$\begin{aligned} \hat{y}_{GA-IPA}(r) = &-\frac{1.3443}{1+e^{-(2.0354r-2.0421)}} + \frac{2.4395}{1+e^{-(1.4845r+1.3709)}} + \frac{2.6754}{1+e^{-(1.0329r+1.0443)}} \\ &+ \frac{2.8616}{1+e^{-(0.8075r-2.0121)}} + \frac{2.7720}{1+e^{-(0.6548r-2.0479)}} + \frac{1.9280}{1+e^{-(1.1723r-1.9858)}} \\ &- \frac{-3.2614}{1+e^{-(2.3212r+2.6150)}} + \frac{7.9969}{1+e^{-(4.5473r-7.3796)}} + \frac{1.0611}{1+e^{-(1.8126r+1.1560)}} \\ &- \frac{-0.8067}{1+e^{-(6.4985r-8.4937)}} \end{aligned} \tag{24}$$

We computed approximate numerical solutions using neural networks with IPA, GA, and their hybridiza-

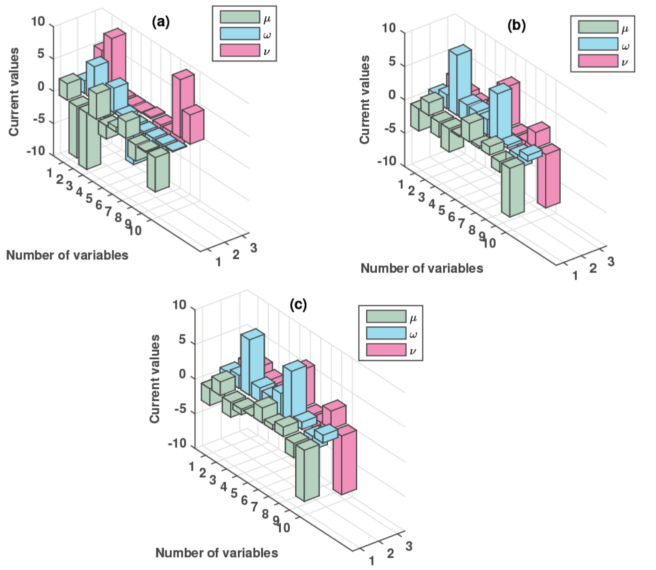

Fig. 5 Trained weights for ANNs Using IPA, GA, and GA-IPA

tion GA-IPA, and the results of each solver are presented in Table 5 for input values within the range $r \in [0, 1]$ having a step size of 0.1. Additionally, exact solutions were also computed and are displayed in Table 5 for the same inputs. The graphical representation of the approximate solutions overlapping with the exact solution is illustrated in Fig. 6(a), confirming the accuracy to a precision of 4 to 5 decimal places. In terms of Absolute Errors (AEs), a comparison was made with the Variational Iteration Method (VIM) [34], and the results are summarized in Table 6. Notably, the AEs for IPA, GA, and GA-IPA fall within the ranges of $10^{-07} \rightarrow 10^{-09}$, $10^{-05} \rightarrow 10^{-06}$, and $10^{-09} \rightarrow 10^{-10}$ respectively, and the AEs of VIM are lie in $10^{-05} \rightarrow 10^{-16}$. It's important to observe that the proposed results exhibit stability in AEs, whereas the AEs of VIM are not consistent.

Additionally, we investigated the approximate solution of Problem II using various numbers of trained weights in ANN based SDDE models. Typically, the weights assigned to neurons strike a balance between the algorithm's complexity and its accuracy. The approximate solutions were computed for different scenarios with N = 10, 15, and 30, and the results are detailed in Tables 7 and 8. According to our findings, adding more neurons leads to more accurate algorithms, but at the cost of a much higher processing effort.

Table 5 Exact and proposed results in case of problem II

r	r_{Ext}	*IPA*	*GA*	*GA − IPA*
0.1	1.000050	1.009962	1.009926	1.010036
0.2	1.040811	1.040654	1.039139	1.040796
0.5	1.284025	1.283872	1.281979	1.284007
0.7	1.632316	1.632090	1.629597	1.6322814
0.8	1.896481	1.896240	1.893389	1.896450
0.9	2.247908	2.247607	2.244200	2.247863
1.0	2.718282	2.717940	2.713940	2.718267

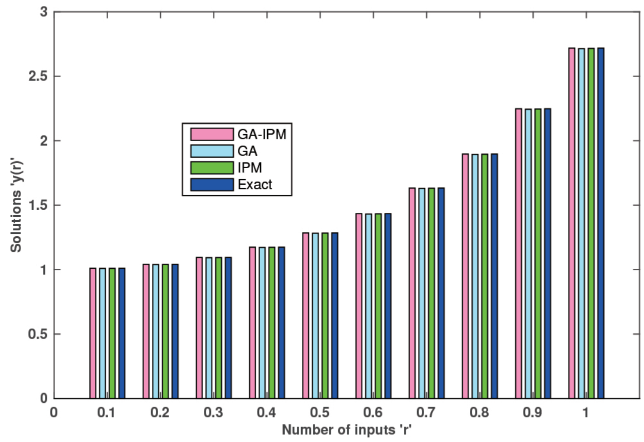

Fig. 6 Comparison of ANNs results for the Proposed Results and Exact Solution in Problem II

Table 6 Comparative studies of ANNs solutions in terms of AE in case of Problem II, N=15

t	proposed results			Results VIM in literature
	IPA	*GA*	*GA-IPA*	*VIM* [34]
0.1	1.32E-08	3.89E-07	3.89E-07	2.22E-16
0.2	3.85E-08	6.60E-07	6.60E-07	4.44E-16
0.5	3.78E-08	7.34E-07	7.34E-07	3.89E-10
0.7	7.29E-08	1.38E-06	1.38E-06	8.71E-08
0.8	8.86E-08	1.64E-06	1.64E-06	7.51E-07
0.9	1.33E-07	2.53E-06	2.53E-06	5.04E-06
1.0	1.85E-07	3.48E-06	3.48E-06	2.79E-05

Table 7 Comparison of the proposed solutions in terms of AE in case of Problem II, N=30

t	Proposed results		
	IPA	*GA*	*GA-IPA*
0.1	7.84E-09	1.50E-06	2.10E-10
0.2	2.47E-08	2.80E-06	2.31E-10
0.5	2.34E-08	4.19E-06	3.26E-10
0.7	5.10E-08	7.40E-06	1.22E-09
0.8	5.78E-08	9.56E-06	9.43E-10
0.9	9.09E-08	1.37E-05	2.05E-09
1.0	1.17E-07	1.88E-05	2.15E-10

Table 8 Comparison of the proposed solutions in terms of AE in case of Problem II, N=45

t	Proposed results		
	IPA	*GA*	*GA-IPA*
0.1	4.12E-10	8.64E-08	2.54E-17
0.2	1.33E-09	3.27E-07	2.46E-17
0.3	1.68E-09	2.90E-07	2.45E-17
0.4	1.30E-09	2.10E-07	1.78E-17
0.5	1.23E-09	2.59E-07	2.49E-17
0.6	2.15E-09	3.95E-07	5.50E-17
0.7	3.18E-09	4.85E-07	1.00E-16
0.8	3.24E-09	5.80E-07	1.48E-16

0.9	5.45E-09	8.99E-07	3.28E-16
1.0	5.92E-09	1.24E-06	5.56E-16

5. STATISTICAL ANALYSIS

We conducted a comprehensive statistical analysis to verify the reliability and convergence of our design approaches. For each solver used, we performed 100 independent runs, and from these runs, we computed various statistical measures, including the minimum (MIN), maximum (MAX), mean, mean absolute error (MAE), standard deviation (SDT), and mean square error (MSE) of the AEs associated with the proposed results.

The MAE is written as:

$$MAE = \frac{1}{11}\sum_{j=0}^{10}|y(r_j) - \hat{y}(r_j)| \tag{25}$$

whereas $r \in (r0 = 0, r_1 = 0.1, r_2 = 0.2, ..., r_{10} = 1)$. It is noted that the MAE valuesfor IPA lie around 10^{-08}, 10^{-06} and 10^{-05} , for GA the order lies around 10^{-09}, 10^{-08} and 10^{-07}, and GA-IPA lies around 10^{-09}, 10^{-07}, 10^{-05} in case of problem I. The MSE is written as:

$$MSE = \frac{1}{W}\sum_{i=1}^{W}\left(y(r) - \hat{y}_i(r)\right)^2, \tag{26}$$

whereas $r \in (r_0 = 0, r_1 = 0.1, r_2 = 0.2, ..., r_{10} = 1)$. Where i represents the initial independent run and W represents the overall independent runs of the stochastic solvers. we run our algorithm for 100 times independently for all three proposed approaches for inputs $r \in [0, 1]$ with stepsize $h = 0.1$. The results computed by 100 independent runs are provided in Table 9. We also computed the average values for these runs, and for problem I, the results for each solver are as follows: For the IPA solver, the average values are approximately: MIN: 10^{-06}, MAX: 10^{-10}, SDT: 10^{-10}, Mean: 10^{-19}, and MSE: 10^{-09}. For the GA solver, the average values are approximately: MIN: 10^{-06}, MAX: 10^{-02}, SDT: 10^{-08}, Mean: 10^{-03}, and MSE: 10^{-07}. For the hybrid solver GA-IPA, the average values are around: MIN: 10^{-11}, MAX: 1^{0-19}, SDT: 10^{-22}, Mean: 10^{-20}, and MSE: 10^{-10}. These values reflect the statistical results obtained for problem I.

Moreover, the graphical view of the results for fitness 'e_{SDDE}', MAE, and root mean square error (RMSE) can be seen in Fig. 7(a-d). It is noted from Fig. 7(a) and Fig. 7(b) that the hybridization of GA-IPA show good convergence for fitness 'eSDDE', Fig.7(c) shows that GA gives low fitness 'eSDDE', and from Fig. 7(d) GA-IPA also consistently convergent. We performed 100 independent runs to calculate the MAE for problem II. For the IPA solver, the MAE falls within the ranges of 10^{-07} to 10^{-05}, for the GA solver, it ranges from 10^{-06} to 10^{-05}, and for GA-IPA, the MAE values vary between 10^{-06} and 10^{-05}. We computed the mean values for MSE, SDT, Mean, Max and Min for each solver in Problem II. The results are as follows: For the IPA solver, the average values fall within the ranges of 10^{-06} for Min, 10^{-06} for Max, 10^{-07} for Mean, 10^{-07} for SDT, and 10^{-10} for MSE. For the GA solver, the average values are approximately: 10^{-12} for Min, 10^{-12} for Max, 10^{-17} for Mean, 10^{-12} for SDT, and 10^{-06} for MSE. For the hybrid approach GA-IPA, the average values are within the ranges of: 10^{-05} for Min, 10^{-05} for Max, 10^{-05} for Mean, 10^{-05} for SDT, and 10^{-10} for MSE. We also evaluated the proposed results $\hat{y}(r)$ against 100 independent runs for fitness 'e_{SDDE}', MAE, and RMSE, as illustrated in Fig. 8(a-d). It's evident that the hybrid approach outperforms the other two techniques in terms of achieving a lower fitness

Table 9 Results of statistical analysis on the basis of fitness 'e'

Technique	r	Problem-I				
		Min	*Max*	*Mean*	*SDT*	*MSE*
IPA	0.1	5.30E-13	2.04E-08	2.49E-16	4.38E-09	2.30E-08
	0.3	2.28E-11	4.32E-06	2.29E-11	3.60E-06	4.56E-06
	0.5	5.28E-11	7.03E-05	4.82E-09	4.73E-05	7.68E-05
	0.7	5.54E-10	8.99E-06	2..37E-07	6.41E-15	2.36E-07
	0.9	8.47E-11	2.70E-17	3.18E-06	2.72E-12	4.74E-06
GA	0.1	2.38E-05	8.82E-02	7.24E-03	9.89E-14	4.66E-07
	0.3	3.43E-05	3.71E-01	2.39E-02	3.62E-12	2.85E-06
	0.5	3.68E-06	2.60E-01	2.63E-02	7.30E-12	7.04E-06
	0.7	2.79E-04	3.20E-01	3.23E-02	2.79E-10	2.83E-05
	0.9	5.43E-04	3.45E-01	3.75E-02	8.45E-10	5.57E-05
GA-IPA	0.1	1.94E-11	9.99E-16	2.10E-18	2.22E-09	3.65E-09
	0.3	4.04E-12	2.20E-10	3.88E-14	1.31E-06	1.49E-06
	0.5	3.16E-09	8.29E-08	1.40E-11	2.54E-05	2.89E-05
	0.7	5.56E-09	3.88E-06	7.47E-10	8.84E-16	3.92E-08
	0.9	2.91E-08	6.43E-05	1.37E-08	2.37E-13	6.53E-07
Problem II						

IPA	0.1	7.78E-06	3.53E-06	4.49E-07	4.66E-07	9.57E-07
	0.3	3.43E-06	4.63E-06	2.43E-06	6.81E-07	5.75E-06
	0.5	3.71E-06	6.49E-06	2.55E-06	7.66E-07	2.32E-06
	0.7	4.13E-06	7.25E-06	3.73E-06	2.35E-06	6.39E-06
	0.9	2.40E-05	3.43E-05	3.45E-06	4.49E-06	8.18E-06
GA	0.1	3.73E-12	4.75E-12	3.25E-12	8.45E-09	3.27E-06
	0.3	2.74E-11	2.86E-11	2.53E-11	8.58E-11	4.63E-06
	0.5	2.72E-11	3.44E-11	2.73E-11	4.79E-09	5.48E-06
	0.7	6.39E-11	9.80E-11	4.64E-11	7.83E-09	8.59E-06
	0.9	2.76E-10	3.23E-10	2.87E-10	2.32E-08	2.38E-05
GA-IPA	0.1	2.73E-05	5.19E-05	2.63E-05	3.43E-10	6.73E-10
	0.3	3.43E-05	8.29E-05	3.74E-05	3.31E-10	2.18E-09
	0.5	3.29E-05	7.49E-05	3.78E-05	6.63E-10	2.54E-09
	0.7	4.54E-05	2.67E-04	5.82E-05	8.45E-10	3.34E-09
	0.9	6.68E-05	2.37E-04	6.42E-05	2.27E-09	5.78E-09

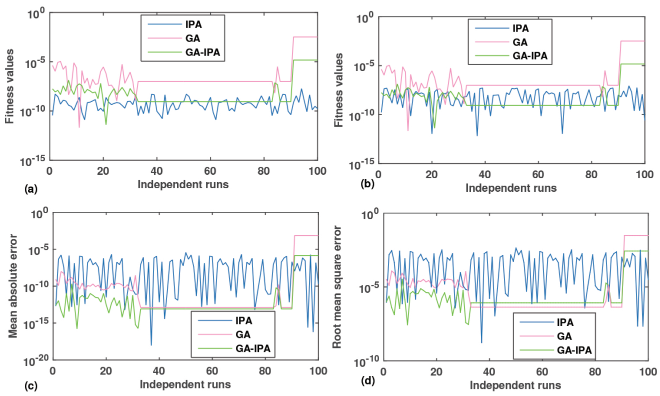

Fig. 7 Graphical view of fitness value, MAE and RMSE for 100 independent runs in case of problem I

Table 10 Exact and proposed results in case of problem II

e	% of runs with fitness			MAE	% of runs with MAE		
	IPA	*GA*	*GA-IPA*		*IPA*	*GA*	*GA-IPA*
10^{-04}	00	00	00	10^{-04}	61	10	10
10^{-05}	00	09	00	10^{-05}	03	00	01
10^{-06}	00	00	00	10^{-06}	19	00	17
10^{-07}	00	00	00	10^{-07}	13	00	66
10^{-08}	00	00	00	10^{-08}	02	00	06
10^{-09}	27	00	00	10^{-09}	01	04	00
10^{-10}	53	01	91	10^{-10}	01	17	00
10^{-11}	20	07	09	10^{-11}	00	11	00
10^{-12}	00	06	00	10^{-12}	00	01	00
10^{-13}	00	00	00	10^{-13}	00	57	00
10^{-14}	00	04	00	10^{-14}	00	00	00
10^{-15}	00	63	00	10^{-15}	00	00	00

’e_{SDDE}’, as shown in Fig. 8(a) and Fig. 8(b). IPA yields lower fitness ’e_{SDDE}’ with variations, while GA and GA-IPA consistently deliver results with low MAE, as seen in Fig. 8(c). Moreover, GA and GA- IPA also exhibit consistent convergence in terms of RMSE, as demonstrated in Fig. 8(d). Furthermore,

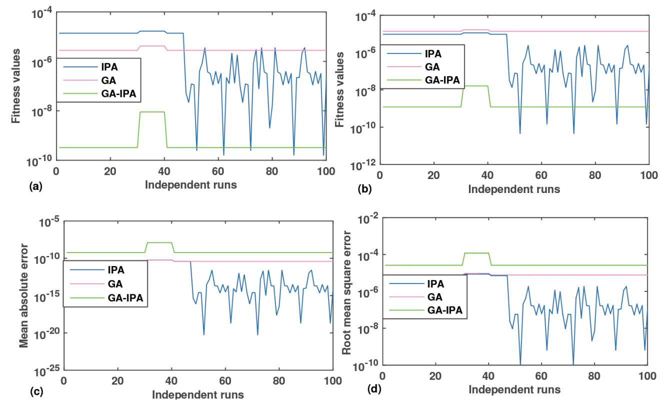

Fig. 8 Graphical view of fitness value, MAE and RMSE for 10 independent runs for problem II

we conducted 100 independent runs to determine the percentage of convergence for the fitness function ’e_{SDDE}’ and the MAE using neural network models as described in Equations (14) and (20). This was done to ensure the reliability of our proposed techniques. The results of the percentage of convergence for the three designed solvers are presented in Table 10. For problem I, we observed that the values of the fitness function ’eSDDE’ and MAE that meet the specified criterion fall within the range of 10^{-15} to 10^{-04}. From Fig. 9(a)-(b), it is evident that the hybridization of GA-IPA consistently converges in all runs of the algorithm, while the GA and IPA approaches struggle to achieve the required fitness value. The percentage of convergence for these two approaches is lower across all one hundred independent runs. The convergence analysis, conducted through statistical analysis over one hundred independent runs for all three solvers, is presented in Fig. 9 and Fig. 10. Detailed numerical outcomes are provided in Table 10 and 11. Overall, the results demonstrate that the proposed approach, utilizing ANNs based differential difference models optimized with the hybrid computing technique GA-IPA, delivers the most accurate and consistent convergent results compared to the other algorithms.

The results obtained from all three solvers demonstrate that the fitness value 'e_{SDDE}' and MAE, which satisfy the adapted criterion, fall within the ranges of 10^{-11} to 10^{-06} for problem II, as presented in Table 11. From Fig. 10(a)-(b), it is evident that the GA solver consistently achieves convergence across

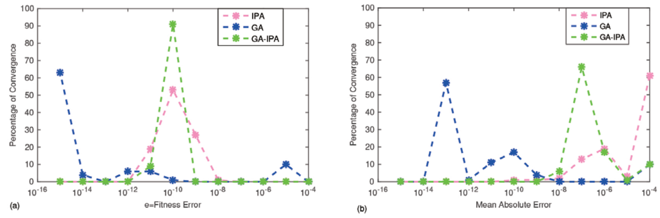

Fig. 9 Convergence analysis of fitness 'e' and MAE for problem I

Table 11 Results of Convergence analysis of SDDE for problem II

e	% of runs with fitness			MAE	% of runs with MAE		
	IPA	*GA*	*GA-IPA*		*IPA*	*GA*	*GA-IPA*
10^{-06}	047	100	000	10^{-06}	054	000	000
10^{-07}	008	000	000	10^{-07}	019	000	000
10^{-08}	029	000	000	10^{-08}	018	000	010
10^{-09}	010	000	010	10^{-09}	004	000	000
10^{-10}	001	000	090	10^{-10}	003	000	090
10^{-11}	005	000	000	10^{-11}	002	100	000

all runs of the algorithm. In contrast, the IPA and GA-IPA approaches are inadequate in achieving the required fitness value and exhibit lower convergence percentages in all one hundred independent runs. The numerical computations and simulations were carried out on an HP Premier Experience system with a CPU running at 2.40GHz, equipped with 2.00 GB of RAM, and an Intel(R) Core(TM) i3-2370M processor. The program was executed using MATLAB version R2015a.

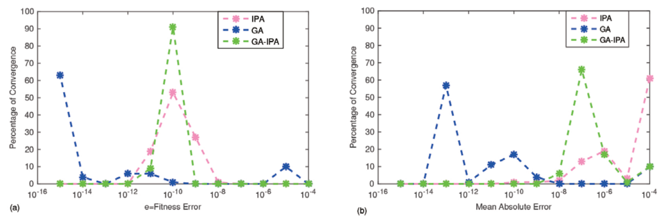

Fig. 10 Convergence analysis of fitness 'e' and MAE for problem II

6. CONCLUSION

In this section, the concluded remarks on the basis of numerical simulations and results are as follows:

- A new artificial intelligence based numerical solution is introduced to solve Initial value problems of linear singular differential difference equation by using log-sigmoid function as an activation function. We presented homogeneous and non homogeneous models of SDDEs optimized with IPA, GA and hybridization GA-IPA with the knacks of ANNs.
- Comparison of the results with the exact solution shows that the AEs are in the ranges of 10^{-07} 10^{-10}, $10^{-05}\rightarrow 10^{-08}$ and 10^{-05} 10^{-12} for IPA, GA and hybridization of GA-IPA respectively; whereas the AE of aforementioned results in literature BMM has order $10^{-04}\rightarrow 10^{-06}$ in problem I. And the results of AEs for three proposed solver are in the ranges of $10^{-05}\rightarrow 10^{-06}$, $10^{-05}\rightarrow 10^{-06}$ and $10^{-09}\rightarrow 10^{-10}$ respectively for problem II, whereas the VIM has different values of absolute error at every step size, are in the ranges of $10^{-05}\rightarrow 10^{-16}$. But AEs of proposed scheme gives consistent results.
- The numerical results and simulations of designed scheme is well-defined and predictable approach to calculate the numerical solution of linear SDDEs. The numerical results proved the accuracy of the scheme by matching the approximate results and exact solution up to 4-5 decimal places. The graphical representation of solutions obtained by proposed scheme and the exact solution also overlap each other.
- Statistical analysis of the proposed scheme has been executed based on 100 independent runs to certify the precision and reliability of the design scheme. Feed forward ANNs log-sigmoid function give significant convergence in 100 independent runs for three proposed approaches IPA, GA and hybrid combination of GA-IPA. Moreover, it is found that in hundred independent runs hybrid approach GA-IPA provides better convergence percentage i.e., 95% on the basis of fitness value

$\epsilon \leq 10^{-08}$ for problem I, whereas GA provides 95 to 98% better convergence for fitness value

$\epsilon \leq 10^{-06}$ for problem II.

The accuracy, convergence of the designed unsupervised ANN model for singular differential difference system can be improved by use of the new competency of nature inspired computing technique based on PSO and its integration with efficient local search technique. Additionally, exploitation of strength of optimization of evolutionary computing through differential evolution hybrid with local search method- ologies can be a good alternative to the improve performance of the singular differential difference system in term of precision, reliability, robustness and stability.

This research can be a good alternate with other activation functions as Mexican hat, tangent sigmoid, Wavelets hat, radial basis etc., to deal with the SDDEs.

Conflict of Interest

No conflicts of interest have been disclosed by any of the authors of the manuscript.

Data availability

This manuscript does not contain any associated data.

REFERENCES

1.Bellen, A. and Zennaro, M., (2013). "Numerical methods for delay differential equations." *Oxford university press*.

2.Anwar, N., Ahmad, I., Kiani, A.K., Shoaib, M. and Raja, M.A.Z., (2023). "Intelligent solution predictive networks for non-linear tumor-immune delayed model." *Computer Methods in Biomechanics and Biomedical Engineering*, pp.1-28.

3.Anwar, N., Ahmad, I., Kiani, A.K., Shoaib, M. and Raja, M.A.Z., (2023). "Novel intelligent Bayesian computing networks for predictive solutions of nonlinear multi-delayed tumor oncolytic virotherapy systems." *International Journal of Biomathematics*.

4.Obut, T., Cimen, E. and Cakir, M., (2023). "A novel numerical approach for solving delay differential equations arising in population dynamics." *Mathematical Modelling and Control*, **3**(3), pp.233-243.

5.Kashem, B.E. and Shihab, S., (2020). "Approximate solution of

Lane-Emden problem via modified Hermite operation matrix method." *Samarra Journal of Pure and Applied Science*, **2**(2), pp.57-67.

6.OLAYI˙WOLA, M.O. and Adegoke, A., Infinite Taylor Series Method for Solving Lane-Emden Type Equations. Cankaya University Journal of Science and Engineering, **17**(1), pp.1-10.

7.Nadeem, M. and Ahmad, H., (2019). "Variational iteration method for analytical solution of the Lane-Emden type equation with singular initial and boundary conditions." *Earthline Journal of Mathematical Sciences*, **2**(1), pp.127-142.

8.Khan, Y., (2021). "Maclaurin series method for fractal differential-difference models arising in coupled nonlinear optical waveguides." *Fractals*, **29**(01), p.2150004.

9.Ranjan, R. and Shankar Prasad, H., (2020). "A fitted finite difference scheme for solving singularly perturbed two point boundary value problems." *Information Sciences Letters*, **9**(2), p.2.

10.Devi, V., Maurya, R.K., Patel, V.K. and Singh, V.K., (2019). "Lagrange Operational Matrix Methods to Lane–Emden, Riccati's and Bessel's Equations." *International Journal of Applied and Computa- tional Mathematics*, **5**(3), pp.1-30.

11.Iqbal, M.K., Abbas, M. and Zafar, B., (2020). "New Quartic B-spline Approximations for Numerical Solution of Fourth Order Singular Boundary Value Problems." *Journal of Mathematics*, **52**(3), pp.47-63.

12.Daba, I.T. and Duressa, G.F., (2021). "Extended cubic B-spline collocation method for singularly perturbed parabolic differential-difference equation arising in computational neuroscience." *International Journal for Numerical Methods in Biomedical Engineering,* **37**(2), p.e3418.

13.Wang, Z., Zou, L. and Zhang, H., (2007). "Applying homotopy analysis method for solving differential-difference equation." *Physics Letters A*, **369**(1-2), pp.77-84.

14.Marinca, V. and Herisanu, N., (2019), "July. Optimal homotopy asymptotic method for polytrophic spheres of the Lane-Emden type equation." *In AIP Conference Proceedings* (Vol. 2116, No. 1, p. 300003). AIP Publishing LLC.

15.Arqub, O.A., Osman, M.S., Abdel-Aty, A.H., Mohamed, A.B.A. and Momani, S., (2020). "A numerical algorithm for the solutions of ABC singular Lane–Emden type models arising in astrophysics using reproducing kernel discretization method." *Mathematics*, **8**(6), p.923.

16.Asadi, M., (2019). "Exact solutions of singular IVPs Lane-Emden equation." *Communications in Non- linear Analysis*, **6**(1), pp.60-63.

17.Roul, P., Goura, V.P. and Agarwal, R., (2019). A compact finite difference method for a general class of nonlinear singular boundary value problems with Neumann and Robin boundary conditions. "*Applied Mathematics and Computation*, **350**, pp.283-304.

18.Erdogan, F., Sakar, M.G. and Saldır, O., (2020). "A finite difference method on layer-adapted mesh for singularly perturbed delay differential equations". *Applied Mathematics and Nonlinear Sciences*, **5**(1), pp.425-436.

19.Debela, H.G. and Duressa, G.F., (2020). "Accelerated fitted operator finite difference method for singularly perturbed delay differential equations with non-local boundary condition." *Journal of the Egyptian Mathematical Society*, **28**(1), pp.1-16.

20.Dizicheh, A.K., Salahshour, S., Ahmadian, A. and Baleanu, D., (2020). "A novel algorithm based on the Legendre wavelets spectral technique for solving the Lane–Emden equations." *Applied Numerical Mathematics,* **153**, pp.443-456.

21.AL-Rabahi, Z.A.A. and Hasan, Y.Q., (2020). "Numerical Solutions of Fourth-order Singular Boundary Value Problems by New Modified Adomian Decomposition Method."

22.Yahya, Q.H., Osilagun, J.A. and Adegbindin, A.O., (2019). "Approximate Analytical Solution of a Class of Singular Differential Equations with Dirichlet Boundary Conditions by the Modified Adomian Decomposition Method. International Journal of Mathematical Sciences and Optimization: Theory and Applications," **2019**(1), pp.433-442.

23.C¸ akmak, M., (2019). "Fibonacci operational matrix algorithm for solving differential equations of Lane-Emden type." *Sakarya U¨ niversitesi Fen Bilimleri Enstitu¨su¨ Dergisi*, **23**(3), pp.478-485.

24.Zhang, B., Zhao, J. and Chen, S., (2020). "The nonconforming virtual element method for fourth-order singular perturbation problem." *Advances in Computational Mathematics*, **46**(2), pp.1-23.

25.Khalid, M., Sultana, M. and Khan, F.S., (2020). "Elementary analysis of Lane-Emden equation by successive differentiation method." *Punjab University Journal of Mathematics,* **51**(6).

26.Aminikhah, H., (2018). "Solutions of the Singular IVPs of Lane-Emden type equations by combining Laplace transformation and perturbation technique." *Nonlinear Engineering*, **7**(4), pp.273-278.

27.Wei, C.F., (2019). "Application of the homotopy perturbation method for solving fractional Lane- Emden type equation." *Thermal Science*, **23**(4), pp.2237-2244.

28.Singh, R., (2019). "A modified homotopy perturbation method for nonlinear singular Lane–Emden equations arising in various physical models." *International Journal of Applied and Computational Mathematics*, **5**(3), pp.1-15.

29.Ramos, H. and Rufai, M.A., (2022). "An adaptive pair of onestep hybrid block Nystr¨om methods for singular initialvalue problems of Lane–Emden–Fowler type." *Mathematics and Computers in Simulation,* **193**, pp.497-508.

30.Seiler, W.M. and Seiß, M., (2021). "Singular initial value problems for scalar quasi-linear ordinary differential equations." *Journal of Differential Equations,* **281**, pp.258-288.

31.Daba, I.T. and Duressa, G.F., (2022). "Collocation method using artificial viscosity for time dependent singularly perturbed differential–difference equations." *Mathematics and Computers in Simulation,* **192**, pp.201-220.

32.Kabeto, M.J. and Duressa, G.F., (2021). "Robust numerical method for singularly perturbed semilinear parabolic differential difference equations." *Mathematics and Computers in Simulation*, **188**, pp.537-547.

33.Xiao, Q., Zeng, Z., Huang, T. and Lewis, F.L., (2021). "Positivity and stability of coupled differential–difference equations with timevarying delay on time scales." *Automatica*, **131**, p.109774.

34.s. Yuzbasi, A numerical approach for solving the high-order linear singular differential-difference equations," *Computers and Mathematics with Applications.* **62**(2011) 2289-2303.

35.Munawar, S., Khan, Z.A., Chaudhary, N.I., Javaid, N. and Raja, M.A.Z., (2023). "Machine intelligence aware electricity theft detection for smart metering applications." *Waves in Random and Complex Media*, pp.1-21.

36.Chaudhary, N.I., Ahmed, M., Khan, Z.A., Zubair, S., Raja, M.A.Z. and Dedovic, N., 2018. Design of normalized fractional adaptive algorithms for parameter estimation of control autoregressive autoregressive systems. Applied Mathematical Modelling, **55**, pp.698-715.

37.Anwar, N., Ahmad, I., Kiani, A.K., Naz, S., Shoaib, M. and Raja, M.A.Z., 2022. Intelligent predictive stochastic computing for nonlinear differential delay computer virus model. Waves in Random and Complex Media, pp.1-29.

38.Aslam, Z., Farooq, H.M.H., Ghayour, M., Hashmi, H., Khan, R.A.A. and Banday, A.M., 2022. A Research Study on use and Importance of the Diagnostic Radiology Reference Values. Pakistan Journal of Medical and Health Sciences, **16**(08), pp.799-799.

39.Ashraf, S., Saleem, S., Ahmed, T., Aslam, Z. and Shuaeeb, M., 2020, September. Iris and foot based sustainable biometric identification approach. In 2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM) (pp. 1-6). IEEE.

40.Yuanlei, S., Almohsen, B., Sabershahraki, M., Issakhov, A. and Raja, M.A.Z., 2021. Nanomaterial migration due to magnetic field through a porous region utilizing numerical modeling. Chemical Physics Letters, **785**, p.139162.

41.Shoaib, M., Raja, M.A.Z., Farhat, I., Shah, Z., Kumam, P. and Islam, S., 2022. Soft computing paradigm for Ferrofluid by exponentially stretched surface in the presence of magnetic dipole and heat transfer. Alexandria Engineering Journal, **61**(2), pp.1607-1623.

42.Anwar, N., Shoaib, M., Ahmad, I., Naz, S., Kiani, A.K. and Raja, M.A.Z., 2023. Intelligent comput- ing networks for nonlinear influenza-A epidemic model. International Journal of Biomathematics, **16**(04), p.2250097.

43.Shoaib, M., Anwar, N., Ahmad, I., Naz, S., Kiani, A.K. and Raja, M.A.Z., 2022. Intelligent networks knacks for numerical treatment of nonlinear multi-delays SVEIR epidemic systems with vaccination. International Journal of Modern Physics B, **36**(18), p.2250100.

44.Anwar, N., Ahmad, I., Kiani, A.K., Shoaib, M. and Raja, M.A.Z., 2023. Intelligent solution pre- dictive networks for non-linear tumor-immune delayed model. Computer Methods in Biomechanics and Biomedical Engineering, pp.1-28.

45.Anwar, N., Ahmad, I., Kiani, A.K., Shoaib, M. and Raja, M.A.Z., 2023. Novel intelligent Bayesian computing networks for predictive solutions of nonlinear multi-delayed tumor oncolytic virotherapy systems. International Journal of Biomathematics.

46.Shoaib, M., Anwar, N., Ahmad, I., Naz, S., Kiani, A.K. and Raja, M.A.Z., 2023. Neuro- computational intelligence for numerical treatment of multiple delays SEIR model of worms prop- agation in wireless sensor networks. Biomedical Signal Processing and Control, **84**, p.104797.

47.Anwar, N., Ahmad, I., Fatima, A. et al. Design of intelligent Bayesian supervised predictive net- works for nonlinear delay differential systems of avian influenza model. Eur. Phys. J. Plus **138**, 911 (2023). https://doi.org/10.1140/epjp/s13360-023-04533-w

48.Anwar, N., Ahmad, I., Raja, M.A.Z., Naz, S., Shoaib, M. and Kiani, A.K., 2022. Artificial intelli- gence knacks-based stochastic paradigm to study the dynamics of plant virus propagation model with impact of seasonality and delays. The European Physical Journal Plus, **137**(1), p.144.

49.Anwar, N., Ahmad, I., Kiani, A.K., Shoaib, M. and Raja, M.A.Z., Numerical treatment for math- ematical model of farming awareness in crop pest management. Frontiers in Applied Mathematics and Statistics, **9**, p.1208774.

50.Ahmad, I., Raja, M.A.Z., Ramos, H., Bilal, M. and Shoaib, M., 2021. Integrated neuro-evolution- based computing solver for dynamics of nonlinear corneal shape model numerically. Neural Com- puting and Applications, **33**(11), pp.5753-5769.

51.R. Fazio, A novel approach to the numerical solution of boundary value problems on infinite intervals, SIAM Journal on Numerical Analysis. **33**(1996) 1473-1483.

52.K. Liang, L. Ping, MT. Ong, RCE. Tan, A splitting moving mesh method for reactiondiusion equations of quenching type, Journal of Computational Physics. **215**(2006) 757-777.

53.MAZ. Raja, F. H. Shah, M. Tariq, I. Ahmad, Design of artificial neural network models optimized with sequential quadratic programming to study the dynamics of nonlinear Troesch problem arising in plasma physics, Neural Computing and Applications. **29** (6) (2018) 83-109.

54.I. Ahmad, F. Ahmad, MAZ Raja, H. Ilyas, N. Anwar, Z. Azad, Intelligent computing to solve fifth-order boundary value problem arising in induction motor models, Neural Computing and Applications. **29**(7) (2018) 449-466.

55.C. Buskens, H. Maurer, SQP methods for solving optimal control problems with control and state constraints: adjoint variables, sensitivity analysis and real time control, Journal of Computational and Applied Mathematics. **120**(2000) 85-108.

56.RP. Ankit, AP. Mahesh, RV. Dhaval, Variational Analysis and Sequential Quadratic Programming Approach for Robotics, Procedia Technology. **4**(2012) 636-640.

57.JW. Stephen, Primal-Dual Interior-Point Methods, Philadelphia PA SIAM ISBN. (1997) 0-89871- 382-X.

58.MH. Wright, The interior-point revolution in optimization: history, recent developments and lasting consequences, Bulletin of the American Mathematical Society. **42**(2005) 39-56.

59.N. Duvvuru, KS. Swarup, A Hybrid Interior Point Assisted Differential Evolution Algorithm for Economic Dispatch, IEEE Transactions on Power Systems. **2**(26) (2011) 541-549.

60.W. Yan, L. Wen, W. Li, CY. Chung, KP. Wong, Decomposition coordination interior point method and its application to multi-area optimal reactive power flow, International Journal of Electrical Power and Energy Systems. **1**(33) (2011) 55-60.

61.AP. Florian, JW. Stephen, Interior-Point Methods. Journal of Computational and Applied Math- ematics. 200;1-2: (124) 281-302.

62.O. Bahn, MO. Du, JL. Goffin, JP. Vial, A cutting plane method from analytic centers for stochastic programming, Mathematical Programming. **69**(1995) 45-73,.

63.VR. Slyke, R. Wets, L-shaped linear programs with applications to optimal control and stochastic linear programs, SIAM Journal on Applied Mathematics. **17**(1969) 638-663.

64. L. Goffin, A. Haurie, P. Vial, Decomposition and non differentiable optimization with the projective algorithm, Management Science. **38**(1992) 284-302.
65. L. Goffin, J. Gondzio, R. Sarkissian, P. Vial, Solving nonlinear multi commodity network flow problems by the analytic center cutting plane method, Mathematical Programming. **76**(1997) 131-154.
66. T. Back, Evolutionary algorithms in theory and practice, New York, Oxford University Press.
67. DB. Fogel, Evolutionary Computation: Towar a new philosophy of machine intelligence 3rd Edition John Wiley and Sons. New Jersey (2006).
68. BM. Ali, and M. Sezer, Hybrid Euler-Taylor matrix method for solving of generalized linear Fred- holm integro-differential difference equations. Applied Mathematics and Computation **273** (2016): 33-40.
69. AF Wang, M Xu, MK Ni, The impulsive solution for a semi-linear singularly perturbed differential- difference equation. Acta Mathematicae Applicatae Sinica, English Series 32, no. **2**(2016): 333-342.
70. LQ. Luo, XM. Zheng, Value distribution of meromorphic solutions of homogeneous and non- homogeneous complex linear differential-difference equations. Open Mathematics 14, no. **1**(2016): 970-976.
71. F. Mirzaee, B. Saeed, Solving systems of high-order linear differential-difference equations via Euler matrix method. Journal of the Egyptian Mathematical Society 23, no. **2**(2015): 286-291.
72. M. Valipour, ME. Banihabib, SM. R. Behbahani, Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir., Journal of hydrology **476**(2013): 433-441.
73. M. Valipour, Comparison of surface irrigation simulation models: full hydrodynamic, zero inertia, kinematic wave. Journal of Agricultural Science 4, no. **12**(2012): 68.
74. M. Valipour, Use of average data of 181 synoptic stations for estimation of reference crop evapotran- spiration by temperature-based methods. Water resources management, **28**, (12) (2014): 4237-4255.
75. M. Valipour, Comparative evaluation of radiation-based methods for estimation of potential evap- otranspiration. Journal of Hydrologic Engineering 20, no. **5**(2014): 04014068
76. M. Rezaei, M. Valipour, M. Valipour, Modelling evapotranspiration to increase the accuracy of the estimations based on the climatic parameters., Water Conservation Science and Engineering , **1**(3) (2016): 197-207.