Frontiers of Information Technology & Electronic Engineering
www.zju.edu.cn/jzus; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Neuro-heuristic computational intelligence for solving nonlinear pantograph systems

Muhammad Asif Zahoor RAJA†‡1, Iftikhar AHMAD[2], Imtiaz KHAN[3],
Muhammed Ibrahem SYAM[4], Abdul Majid WAZWAZ[5]

([1]*Department of Electrical Engineering, COMSATs Institute of Information Technology, Attock 43200, Pakistan*)
([2]*Department of Mathematics, University of Gujrat, Gujrat 50700, Pakistan*)
([3]*Department of Mathematics, Preston University, Islamabad Campus, Kohat, Islamabad 44000, Pakistan*)
([4]*Department of Mathematical Sciences, United Arab Emirates University, Al-Ain Box 15551, UAE*)
([5]*Department of Mathematics, Saint Xavier University, Chicago, IL 60655, USA*)
†E-mail: rasifzahoor@yahoo.com; Muhammad.asif@ciit-attock.edu.pk

Received Nov. 10, 2015; Revision accepted Feb. 17, 2016; Crosschecked Mar. 14, 2017

Abstract: We present a neuro-heuristic computing platform for finding the solution for initial value problems (IVPs) of nonlinear pantograph systems based on functional differential equations (P-FDEs) of different orders. In this scheme, the strengths of feed-forward artificial neural networks (ANNs), the evolutionary computing technique mainly based on genetic algorithms (GAs), and the interior-point technique (IPT) are exploited. Two types of mathematical models of the systems are constructed with the help of ANNs by defining an unsupervised error with and without exactly satisfying the initial conditions. The design parameters of ANN models are optimized with a hybrid approach GA–IPT, where GA is used as a tool for effective global search, and IPT is incorporated for rapid local convergence. The proposed scheme is tested on three different types of IVPs of P-FDE with orders 1–3. The correctness of the scheme is established by comparison with the existing exact solutions. The accuracy and convergence of the proposed scheme are further validated through a large number of numerical experiments by taking different numbers of neurons in ANN models.

Keywords: Neural networks; Initial value problems (IVPs); Functional differential equations (FDEs); Unsupervised learning; Genetic algorithms (GAs); Interior-point technique (IPT)
http://dx.doi.org/10.1631/FITEE.1500393 **CLC number:** TP183; O175

1 Introduction

The strength of the universal function approximation capability of artificial neural networks (ANNs), optimized with local and global search techniques, has been exploited immensely to solve constrained and unconstrained optimization problems, such as model predictive control of a servomotor driven based on neuro-dynamic optimization (Peng *et al.*, 2014), real-time condition monitoring and fault diagnosis in switched reluctance motors (Uysal and Raif, 2013), effective prediction of thermo-diffusion in arbitrary binary liquid hydrocarbon mixtures (Srinivasan and Saghir, 2014), adaptive near-optimal control of a class of chaotic systems (Tang *et al.*, 2014), and viable determination of compressional wave velocity (Zoveidavianpoor, 2014). Recently, these schemes have been introduced to accurately solve a variety of problems arising in computational physics and applied mathematics (McFall, 2013; Mall and Chakraverty, 2014a; 2014b; 2014c; Chakraverty and Mall, 2014). The design and development of stochastic numerical solvers for finding the solutions

‡ Corresponding author

ORCID: Muhammad Asif Zahoor RAJA, http://orcid.org/0000-0001-9953-822X of nonlinear systems based on differential equations appear to be a promising area of research.

Stochastic platforms based on ANNs are supported by global search methods, such as genetic algorithms (GAs), particle swarm optimization (PSO), simulated annealing (SA), and pattern search (PS). Local search methods including interior-point, sequential quadratic programming, Nelder-Mead simplex direct search, and active-set algorithms have been incorporated to solve governing nonlinear systems associated with initial value problems (IVPs) and boundary value problems (BVPs) of ordinary/partial/fractional/functional differential equations. A few renewed applications solved by these models can be seen in the literature, such as the nonlinear Toresch problem (Raja, 2014a; 2014c), singular nonlinear transformed two-dimensional Bratu problems (Raja *et al.*, 2013; 2014a; 2014b), nonlinear first Painlevé initial value problem (Raja *et al.*, 2012; 2015a; 2015d), nonlinear van der Pol oscillatory problems (Khan *et al.*, 2011; 2015), nonlinear BVPs for fuel ignition model in combustion theory (Raja, 2014b; Raja and Ahmad, 2014), nonlinear algebraic and transcendental equations (Raja *et al.*, 2015b), nonlinear magnetohydrodynamic (MHD) Jeffery–Hamel flow problems (Raja and Samar, 2014a; 2014b), nonlinear BVPs of pantograph functional differential equations (P-FDEs) (Raja, 2014a), linear and nonlinear FDEs (Raja *et al.*, 2010a; 2010b), IVPs of nonlinear Riccati FDEs (Zahoor *et al.*, 2009; Raja *et al.*, 2010c; 2015c), Bagley–Torvik FDEs (Raja *et al.*, 2011a; 2011b), and thin film flow (Raja *et al.*, 2015e). Other techniques such as the exponential functions method, collocation method, and Adomian decomposition method (Evans and Raslan, 2005; Barro *et al.*, 2008; Shakourifar and Dehghan, 2008; Dehghan and Salehi, 2010a; Shakeri and Dehghan, 2010; Yusufoğlu, 2010; Yüzbaşı *et al.*, 2011; Tohidi *et al.*, 2013; Yüzbaşı and Mehmet, 2013; Pandit and Kumar, 2014) have been efficiently used in the literature. All these contributions are our motivation to explore in this domain and try to design a metaheuristic computing procedure for solving IVPs of P-FDEs.

In this paper, we develop a novel computational intelligence algorithm for approximating the solution of IVP for P-FDEs using feed-forward ANNs, evolutionary computing technique synchronized on GAs, IPT, and their hybrid combinations. Silent features of the proposed method are as follows:

1. The scheme comprises ANN mathematical models in an unsupervised manner.

2. Optimal design parameters of the models are trained using heuristic computational intelligence methods based on effective and efficient global and local search techniques.

3. Strengths and weaknesses of the proposed methods are analyzed on six different linear and nonlinear IVPs of the first, second, and third orders.

4. The validity of the results obtained is proven with available standard solutions, i.e., the exact, numerical, and analytical solutions.

The accuracy and convergence of the proposed scheme are also investigated based on a large number of numerical experiments by changing the number of neurons in neural networks modeling.

2 System models: initial value problems of pantograph functional differential equations

P-FDEs are a specific type of functional differential equations that possess proportional delays. These equations are of fundamental importance due to their functional arguments characteristics. Moreover, they play a significant role in the description of various phenomena, particularly in problems where the ordinary differential equation (ODE) models fail. The systems based on these equations occur in many applications in diverse fields including adaptive control, number theory, electrodynamics, astrophysics, nonlinear dynamical systems, probability theory on algebraic structure, quantum mechanics, cell growth, engineering, and economics (Ockendon and Tayler, 1971; Agarwal and Chow, 1986; Iserles, 1993; Derfel and Iserles, 1997; Azbelev *et al.*, 2007; Zhang *et al.*, 2008). The study in this field has been flourishing because of the significance of these equations. This has encouraged researchers to invest a considerable amount of time and effort aiming to make further progress in this domain. Many researchers have shown great interest in solving P-FDEs. Many algorithms, including analytical and numerical solvers, have been developed even recently (Saadatmandi and Dehghan, 2009; Sedaghat *et al.*, 2012).

In this study, a stochastic numerical treatment is presented for solving IVPs of P-FDEs of different orders with prescribed conditions. The generic form of P-FDEs of the first order is given as

$$\begin{aligned}&\frac{\mathrm{d}f}{\mathrm{d}t}-z(f(t),\ f(g(t)),\ t)=0,\ \ 0\le t\le 1,\\&f(0)=c_1.\end{aligned}\tag{1}$$

For second-order P-FDE, the generic form is given as

$$\begin{aligned}&\frac{\mathrm{d}^2 f}{\mathrm{d}t^2}-z\left(f(t),\frac{\mathrm{d}f}{\mathrm{d}t},f(g(t)),t\right)=0,\\&f(0)=c_1,\ \ \frac{\mathrm{d}}{\mathrm{d}t}f(0)=c_2.\end{aligned}\tag{2}$$

Accordingly, the generic form of P-FDE of the third order is given as

$$\begin{aligned}&\frac{\mathrm{d}^3 f}{\mathrm{d}t^3}-z\left(f(t),\frac{\mathrm{d}f}{\mathrm{d}t},\frac{\mathrm{d}^2 f}{\mathrm{d}t^2},f(g(t)),t\right)=0,\\&f(0)=c_1,\ \frac{\mathrm{d}}{\mathrm{d}t}f(0)=c_2,\ \frac{\mathrm{d}^2}{\mathrm{d}t^2}f(0)=c_3,\end{aligned}\tag{3}$$

where $g(t)$ is a known function, smooth or not, of the inputs, and c_1, c_2, and c_3 are the constants representing the initial conditions. The system model for the study consists of three IVPs of P-FDEs as presented in Eqs. (1)–(3).

3 Design of unsupervised artificial neural network models

A concise description of the design of unsupervised ANN models is presented here for solving IVPs of nonlinear P-FDEs. Two types of feed-forward ANN models are developed for the solution and derivative terms of the equation. Additionally, the construction of a fitness function using these networks in an unsupervised manner is given here.

3.1 Artificial neural network modeling: type 1

In this case, the mathematical model of P-FDEs is developed by exploiting the strength of feed-forward ANNs, in the form of the following continuous mapping for solution $f(t)$, and its first derivative $\mathrm{d}f/\mathrm{d}t$, second derivative $\mathrm{d}^2f/\mathrm{d}t^2$, and the nth order derivative $\mathrm{d}^nf/\mathrm{d}t^n$ are written as

$$\begin{cases}\hat f(t)=\sum\limits_{i=1}^{k}\alpha_i y(w_i t+\beta_i),\\ \dfrac{\mathrm{d}\hat f}{\mathrm{d}t}=\sum\limits_{i=1}^{k}\alpha_i\dfrac{\mathrm{d}}{\mathrm{d}t}y(w_i t+\beta_i),\\ \dfrac{\mathrm{d}^2\hat f}{\mathrm{d}t^2}=\sum\limits_{i=1}^{k}\alpha_i\dfrac{\mathrm{d}^2}{\mathrm{d}t^2}y(w_i t+\beta_i),\\ \quad\vdots\\ \dfrac{\mathrm{d}^n\hat f}{\mathrm{d}t^n}=\sum\limits_{i=1}^{k}\alpha_i\dfrac{\mathrm{d}^n}{\mathrm{d}t^n}y(w_i t+\beta_i).\end{cases}\tag{4}$$

For networks in Eq. (4), for composite functional terms in FDEs, the following neural networks mapping is incorporated:

$$\begin{cases}\hat f(g(t))=\sum\limits_{i=1}^{k}\alpha_i y(w_i(g(t))+\beta_i),\\ \dfrac{\mathrm{d}\hat f(g(t))}{\mathrm{d}t}=\sum\limits_{i=1}^{k}\alpha_i\dfrac{\mathrm{d}}{\mathrm{d}t}y(w_i(g(t))+\beta_i),\\ \dfrac{\mathrm{d}^2\hat f(g(t))}{\mathrm{d}t^2}=\sum\limits_{i=1}^{k}\alpha_i\dfrac{\mathrm{d}^2}{\mathrm{d}t^2}y(w_i(g(t))+\beta_i),\\ \quad\vdots\\ \dfrac{\mathrm{d}^n f(g(t))}{\mathrm{d}t^n}=\sum\limits_{i=1}^{k}\alpha_i\dfrac{\mathrm{d}^n}{\mathrm{d}t^n}y(w_i(g(t))+\beta_i).\end{cases}\tag{5}$$

The networks shown in the sets of Eqs. (4) and (5) are the normally used log-sigmoid $y(x)=1/(1+\mathrm{e}^{-x})$ and its respective derivatives as activation functions. Therefore, the networks shown in Eq. (4) can be written in an updated form for the first few terms:

$$\begin{cases}\hat f(t)=\sum\limits_{i=1}^{k}\alpha_i\dfrac{1}{1+\mathrm{e}^{-(w_i t+\beta_i)}},\\ \dfrac{\mathrm{d}\hat f}{\mathrm{d}t}=\sum\limits_{i=1}^{k}\alpha_i w_i\dfrac{\mathrm{e}^{-(w_i t+\beta_i)}}{\left[1+\mathrm{e}^{-(w_i t+\beta_i)}\right]^2},\\ \dfrac{\mathrm{d}^2\hat f}{\mathrm{d}t^2}=\sum\limits_{i=1}^{k}\alpha_i w_i^2\left\{\dfrac{2\mathrm{e}^{-2(w_i t+\beta_i)}}{\left[1+\mathrm{e}^{-(w_i t+\beta_i)}\right]^3}-\dfrac{\mathrm{e}^{-(w_i t+\beta_i)}}{\left[1+\mathrm{e}^{-(w_i t+\beta_i)}\right]^2}\right\},\\ \quad\vdots\end{cases}\tag{6}$$

Similarly, the networks given in Eq. (5) can be written in an updated form for the first few terms:

$$\begin{cases} \hat{f}(g(t))=\sum_{i=1}^{k}\alpha_i\dfrac{1}{1+\mathrm{e}^{-(w_i g(t)+\beta_i)}}, \\ \dfrac{\mathrm{d}\hat{f}}{\mathrm{d}t}=\sum_{i=1}^{k}\alpha_i w_i\dfrac{\mathrm{e}^{-(w_i t+\beta_i)}}{\left[1+\mathrm{e}^{-(w_i t+\beta_i)}\right]^2}\dfrac{\mathrm{d}g}{\mathrm{d}t}, \\ \dfrac{\mathrm{d}^2\hat{f}}{\mathrm{d}t^2}=\sum_{i=1}^{k}\alpha_i w_i^2\left\{\dfrac{2\mathrm{e}^{-2(w_i t+\beta_i)}}{\left[1+\mathrm{e}^{-(w_i t+\beta_i)}\right]^3}\right. \\ \qquad \left.-\dfrac{\mathrm{e}^{-(w_i t+\beta_i)}}{\left[1+\mathrm{e}^{-(w_i t+\beta_i)}\right]^2}\right\}\dfrac{\mathrm{d}g}{\mathrm{d}t} \\ \qquad +\sum_{i=1}^{k}\alpha_i w_i\dfrac{\mathrm{e}^{-(w_i t+\beta_i)}}{\left[1+\mathrm{e}^{-(w_i t+\beta_i)}\right]^2}\dfrac{\mathrm{d}^2 g}{\mathrm{d}t^2}, \\ \qquad \vdots \end{cases} \tag{7}$$

A suitable combination of the equations given in the sets of Eqs. (4) and (5) can be used to model the P-FDEs as given in the set of Eqs. (1)–(3).

Fitness function formulation: An objective or fitness function is developed by defining an unsupervised error. It is given by the sum of two mean-squared errors as

$$e=e_1+e_2, \tag{8}$$

where e_1 and e_2 are error functions associated with different orders of P-FDEs and their initial conditions, respectively. In case of a first-order P-FDE (Eq. (1)), the fitness function $e_{1\text{-FO}}$, the equations constructed by $e_{1\text{-FO}}$, and its initial condition $e_{2\text{-FO}}$, are constructed as

$$\begin{cases} e_{1\text{-FO}}=\dfrac{1}{N}\sum_{m=1}^{N}\left(\dfrac{\mathrm{d}\hat{f}}{\mathrm{d}t}-z(\hat{f}_m,\hat{f}(g_m),t_m)\right)^2, \\ N=\dfrac{1}{h},\ \hat{f}_m=\hat{f}(t_m),\ g_m=g(t_m),\ t_m=mh, \\ e_{2\text{-FO}}=(\hat{f}_0-c_1)^2. \end{cases} \tag{9}$$

However, for the second-order P-FDW as given in Eq. (2), the error functions $e_{1\text{-SO}}$ and $e_{2\text{-SO}}$ are constructed as follows:

$$\begin{cases} e_{1\text{-SO}}=\dfrac{1}{N}\sum_{m=1}^{N}\left[\dfrac{\mathrm{d}^2\hat{f}}{\mathrm{d}t^2}-z\left(\hat{f}_m,\dfrac{\mathrm{d}\hat{f}_m}{\mathrm{d}t},\hat{f}(g_m),t_m\right)\right]^2, \\ e_{2\text{-SO}}=\dfrac{1}{2}\left[(\hat{f}_0-c_1)^2+\left(\dfrac{\mathrm{d}}{\mathrm{d}t}\hat{f}_0-c_2\right)^2\right]. \end{cases} \tag{10}$$

Accordingly, in the case of the third-order P-EDE as given in Eq. (3), the unsupervised error functions $e_{1\text{-TO}}$ and $e_{2\text{-TO}}$ are formulated as

$$\begin{cases} e_{1\text{-SO}} \\ =\dfrac{1}{N}\sum_{m=1}^{N}\left[\dfrac{\mathrm{d}^3\hat{f}}{\mathrm{d}t^3}-z\left(\hat{f}_m,\dfrac{\mathrm{d}\hat{f}_m}{\mathrm{d}t},\dfrac{\mathrm{d}^2 f_m}{\mathrm{d}t^2},\hat{f}(g_m),t_m\right)\right]^2, \\ e_{2\text{-SO}} \\ =\dfrac{1}{3}\left[(\hat{f}_0-c_1)^2+\left(\dfrac{\mathrm{d}}{\mathrm{d}t}\hat{f}_0-c_2\right)^2+\left(\dfrac{\mathrm{d}^2}{\mathrm{d}t^2}\hat{f}_0-c_3\right)^2\right]. \end{cases} \tag{11}$$

3.2 Artificial neural network modeling: type 2

An alternate ANN model is also developed to exactly satisfy the initial conditions to solve the IVPs of P-FDEs. The governing equations of the updated neural networks model for solution $y(t)$, its first derivative $\mathrm{d}f/\mathrm{d}t$, second derivative $\mathrm{d}^2f/\mathrm{d}t^2$, and the nth order derivative $\mathrm{d}^nf/\mathrm{d}t^n$ are written as

$$\begin{cases} \tilde{f}(t)=P(t)+Q(t)\hat{f}(g(t)), \\ \dfrac{\mathrm{d}\tilde{f}}{\mathrm{d}t}=\dfrac{\mathrm{d}P}{\mathrm{d}t}+\dfrac{\mathrm{d}Q}{\mathrm{d}t}\hat{f}(g(t))+Q(t)\dfrac{\mathrm{d}\hat{f}(g(t))}{\mathrm{d}t}\dfrac{\mathrm{d}g}{\mathrm{d}t}, \\ \dfrac{\mathrm{d}^2\tilde{f}}{\mathrm{d}t^2}=\dfrac{\mathrm{d}^2P}{\mathrm{d}t^2}+\dfrac{\mathrm{d}^2Q}{\mathrm{d}t^2}\hat{f}(g(t)) \\ \qquad +2\dfrac{\mathrm{d}Q}{\mathrm{d}t}\dfrac{\mathrm{d}\hat{f}(g(t))}{\mathrm{d}t}\dfrac{\mathrm{d}g}{\mathrm{d}t}+Q(t)\dfrac{\mathrm{d}^2\hat{f}(g(t))}{\mathrm{d}t^2}\left(\dfrac{\mathrm{d}g}{\mathrm{d}t}\right)^2 \\ \qquad +Q(t)\dfrac{\mathrm{d}\hat{f}(g(t))}{\mathrm{d}t}\dfrac{\mathrm{d}^2g}{\mathrm{d}t^2}, \\ \qquad \vdots \\ \dfrac{\mathrm{d}^n\tilde{f}}{\mathrm{d}t^n}=\dfrac{\mathrm{d}^nP}{\mathrm{d}t^n}+\dfrac{\mathrm{d}^nQ}{\mathrm{d}t^n}\hat{f}(g(t))+Q(t)\dfrac{\mathrm{d}^n}{\mathrm{d}t^n}(\hat{f}(g(t))), \end{cases} \tag{12}$$

where $\mathrm{d}/\mathrm{d}t$, $\mathrm{d}^2/\mathrm{d}t^2$, and $\mathrm{d}^n/\mathrm{d}t^n$ are the networks shown in the sets of Eqs. (4) and (5). The updated neural network used for solving Eqs. (1)–(3) satisfies the initial conditions exactly by taking appropriate choices for functions $P(t)$ and $Q(t)$. A suitable combination of the equations given in the set in Eq. (12) can be used to model the P-FDEs as given in the sets of Eqs. (1)–(3).

Fitness function formulation: An objective function or fitness function $\varepsilon_{\mathrm{FO}}$ is developed in an unsupervised manner for Eq. (1) in the mean-squared sense as follows:

$$\varepsilon_{\mathrm{FO}}=\frac{1}{N}\sum_{m=1}^{N}\left(\frac{\mathrm{d}\tilde{f}}{\mathrm{d}t}-z(\tilde{f}_m,\tilde{f}(g_m),t_m)\right)^2, \qquad (13)$$
$$N=\frac{1}{h},\ \tilde{f}_m=\tilde{f}(t_m),\ g_m=g(t_m),\ t_m=mh,$$

where $\tilde{f}$ and its derivative are given by the set in Eq. (12). In case of the second-order P-FDE as given in Eq. (2), the fitness function $\varepsilon_{\mathrm{SO}}$ is constructed as

$$\varepsilon_{\mathrm{SO}}=\frac{1}{N}\sum_{m=1}^{N}\left[\frac{\mathrm{d}^2\tilde{f}}{\mathrm{d}t^2}-z\left(\tilde{f}_m,\frac{\mathrm{d}\tilde{f}_m}{\mathrm{d}t},\tilde{f}(g_m),t_m\right)\right]^2. \qquad (14)$$

Consequently, in case of a third-order P-EDE as given in Eq. (3), the fitness function $\varepsilon_{\mathrm{TO}}$ is written as

$$\varepsilon_{\mathrm{TO}}=\frac{1}{N}\sum_{m=1}^{N}\left[\frac{\mathrm{d}^3\tilde{f}}{\mathrm{d}t^3}-z\left(\tilde{f}_m,\frac{\mathrm{d}\tilde{f}_m}{\mathrm{d}t},\frac{\mathrm{d}^2\tilde{f}_m}{\mathrm{d}t^2},\tilde{f}(g_m),t_m\right)\right]^2. \qquad (15)$$

The weights of neural networks are available such that the objective functions given in Eqs. (8)–(15) approach zeroes, and the proposed solution will approximate the exact solution of Eqs. (1)–(3).

4 Learning process

4.1 Genetic algorithm

GAs, presented by Holland (1975), were inspired by the natural evolution of genes. GAs normally use natural evolution as an optimization mechanism to deal with problems in diverse areas of computational mathematics, engineering, and science. The performance of GA depends on factors including multiplicity of preliminary population, appropriate selection of the fittest chromosomes to the next generation, survival of high-quality genes in recombination operation, and seeding of new genetic material in mutation and the number of generations. GA is an efficient and effective global search technique which is less probable to get stuck in local minima, and more divergence-avoidable, stable, and robust compared to other mathematical solvers. The fundamental operations of GA involve crossover of the type single point, multiple points, heuristic, etc. Mutations via various functions like linear, logarithmic, adaptive feasible, and selection functions are based on the uniform method, tournament method, etc. Potential applications recently addressed with GAs can be seen in Xu *et al.* (2013), Arqub and Zaer (2014), Troiano and Cosimo (2014), and Zhang *et al.* (2014).

There are two types of parameter settings involved in GA: specific and general. The general settings involve the population size, number of generations, chromosome size, and lower and upper bounds of the adaptive parameters, whereas the specific settings involve the scaling function, selection function, crossover function, mutation function, elite count, migration direction, etc. The parameter settings in this work is provided in Table 1, achieved by performing a number of runs depending on the complexity and accuracy of the problems.

4.2 Interior-point technique

IPT is an efficient local search method used to tune the weights or unknowns of a problem. This is a derivative-based technique based on Langrage multipliers and Karush–Kuhn–Tucker (KKT) equations (Wright, 1997; Potra and Wright, 2000). There are some general and specific parameter settings involved with IPT: the specific parameters involve the type of the derivative, scaling function, number of maximum perturbations, and types of finite difference; the general setting involves the Hessian function, minimum perturbation, nonlinear constraint tolerance, fitness limit, and upper and lower bounds. The parameter settings in this work are shown in Table 2.

4.3 Hybrid approach: GA–IPT

It is well known that the global search techniques are enough to obtain the approximate solution, but at a lower level of accuracy. In contrast, local search

Table 1 Parameter setting for genetic algorithms

Parameter	Setting	Parameter	Setting
Population creation	Constraint-dependant	Population size	300
Scaling function	Rank	Chromosome size	30/45/60
Selection function	Uniform	Number of generations	700
Crossover fraction	1.25	Function tolerance	10^{-20}
Crossover function	Heuristic	StallGenLimit	100
Mutation function	Adaptive feasible	Bounds (lower, upper)	$(-20, 20)_{1\times 30}$
Elite count	4	Migration direction	Forward
Initial penalty	10	Migration interval	20
Penalty factor	100	Nonlinear constraint tolerance	10^{-20}
Migration fraction	0.2	Fitness limit	10^{-20}
Subpopulation size	10	Others	Defaults

Table 2 Parameter setting for the interior-point technique

Parameter	Setting	Parameter	Setting
Start point	Randomly from (−3.5, 3.5)	Hessian	BFGS
Derivative	Solver approximate	Minimum perturbation	10^{-8}
Subproblem algorithm	IDI factorization	X-tolerance	10^{-18}
Scaling	Objective and constraints	Nonlinear constraint tolerance	10^{-15}
Maximum perturbation	0.1	Fitness limit	10^{-15}
Finite difference types	Central differences	Bounds (lower, upper)	(−35, 35)

techniques have invariably excellent approximation time complexities, but can get stuck in local minima or cause premature convergence. However, the hybrid approach exploits the capabilities of both the global and local optimizers to obtain an accurate result. As the number of generations of GA gradually increases and the ability to bring the optimal solution decreases significantly, optimal and reliable results can hardly be obtained with very large iteration numbers. Therefore, it is required to merge efficient local search techniques to obtain better results. Hence, IPT is employed in this study to provide rapid convergence, by selecting the best global individuals of GAs as a start point in the training of neural network models of IVPs of P-FDEs. The generic workflow of the proposed methodology is given in Fig. 1. Necessary details for the procedural steps are provided in Algorithm 1.

Algorithm 1 Genetic algorithms – interior-point technique

1. Initialization: The initial population is randomly generated with real values represented. Chromosomes or individuals have the same number of elements as the number of unknown weights in the models. Initialize GAs' parameters as given in Table 1.

2. Calculate fitness: The fitness value for each individual of the population is evaluated first, by using Eq. (3) for the first ANN model and Eq. (8) for the second.

3. Ranking: Each individual of the population is ranked based on the minimum value of the respective fitness functions of the models.

4. Termination criteria: The algorithm is terminated when any of the following criteria is met:

(1) The predefined fitness value is achieved;

(2) The number of generations is completed;

(3) The stop criterion given in Table 1 is fulfilled.

5. Reproduction: Create the next generation at each cycle by using

(1) Crossover: call for the heuristic function;

(2) Mutation: call for the Gaussian function;

(3) Selection: call for the stochastic uniform function, elitism count 5, etc.

6. Hybridization: IPT is used for the refinement of results by using the best individual of GAs as the initial weights. The parameter settings used for IPT are given in Table 2.

7. Store: The final weight vector and the fitness values achieved for this run of the algorithm are stored.

5 Results and discussions

In this section, the results of the design methodology are presented for linear and nonlinear IVPs of P-FDEs. Three types of problems are taken for numerical experimentation based on different orders. Proposed solutions of the equations are determined for both types of ANN models. The effect of the

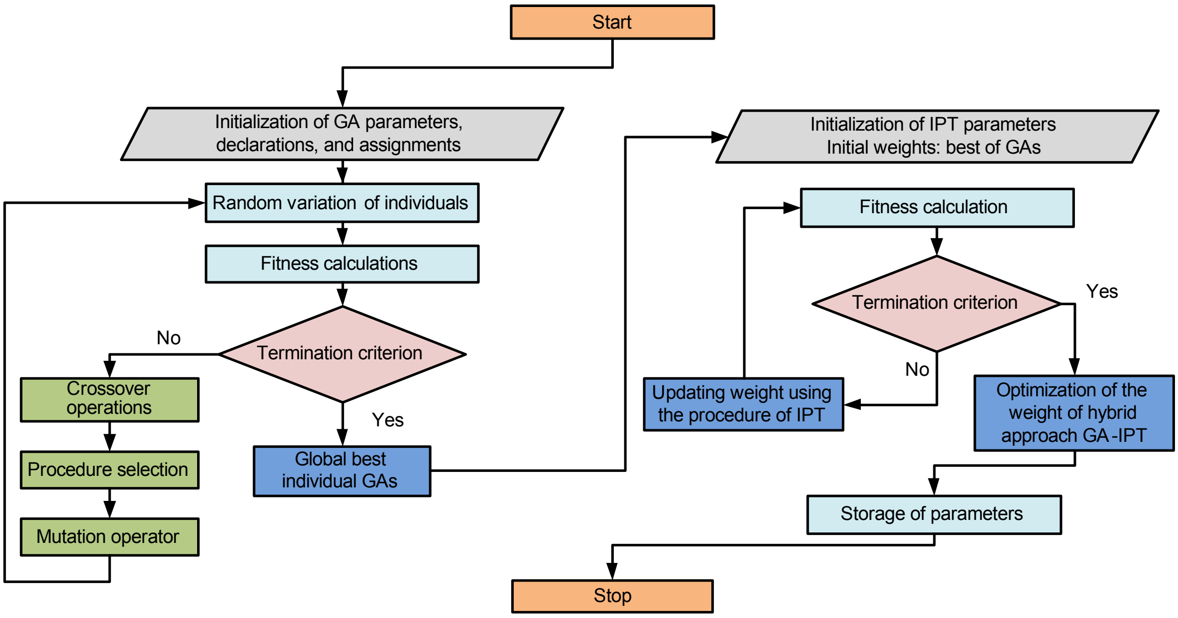
Start
Initialization of GA parameters, declarations, and assignments
Random variation of individuals
Fitness calculations
Termination criterion
No
Yes
Crossover operations
Procedure selection
Mutation operator
Global best individual GAs
Initialization of IPT parameters
Initial weights: best of GAs
Fitness calculation
Termination criterion
Yes
No
Updating weight using the procedure of IPT
Optimization of the weight of hybrid approach GA-IPT
Storage of parameters
Stop

Fig. 1 Generic flowchart of the hybrid evolutionary algorithm based on the genetic algorithm (GA) and interior-point technique (IPT)

change in the number of neurons on accuracy and convergence is also presented by a number of graphical and numerical illustrations, along with comparative studies from exact solutions as the reference.

5.1 Problem I: IVPs of first-order P-FDEs

Four different problems of first-order P-FDEs are solved by the proposed methodology presented in Section 4.3.

Example 1 In this case, IVP of the first-order P-FDE of type 1 is taken as

$$\frac{\mathrm{d}f}{\mathrm{d}t}=\frac{1}{2}\mathrm{e}^{t/2}f(t/2)+\frac{1}{2}f(t),\ \ f(0)=1, \tag{16}$$

which is derived from Eq. (1) by taking $z(f, f(\mathrm{g}), t)$= $0.5\mathrm{e}^{t/2}f(\mathrm{g}(t))+0.5f(t)$ and $g(t)=t/2$. The exact solution of Eq. (16) is given by

$$f(t)=\mathrm{e}^{t}. \tag{17}$$

The proposed design methodology is applied to solve IVP based on two types of ANN models to obtain the approximate solution. The unsupervised errors in the form of fitness functions, which are developed for this equation using N=10 and step size h=0.1, are written for both models as

$$\varepsilon_{\mathrm{FO}}=\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}\tilde{f}_m}{\mathrm{d}t}-\frac{1}{2}\mathrm{e}^{t_m/2}\tilde{f}\left(\frac{t_m}{2}\right)-\frac{1}{2}\tilde{f}_m\right]^2, \tag{18}$$

$$e_{\mathrm{FO}}=\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}\hat{f}_m}{\mathrm{d}t}-\frac{1}{2}\mathrm{e}^{t_m/2}\hat{f}\left(\frac{t_m}{2}\right)-\frac{1}{2}\hat{f}_m\right]^2 \tag{19}$$
$$+(\hat{f}_0-1)^2.$$

Now the requirement is searching for trained weights for the fitness functions given in Eqs. (18) and (19). The proposed methods based on GA, IPT, and the hybrid approach GA–IPT are applied to train unknown weights using the parameter settings as listed in Tables 1 and 2. One set of optimized weights trained by GA–IPT for the number of neurons k=10, with fitness of around 10^{-9} for both models, is used to derive the approximated solutions as given by Eqs. (20) and (21), which are shown on page 472.

The solutions presented in Eqs. (20) and (21) are provided in the extended form in Appendix, Eqs. (A1) and (A2), respectively. The proposed solutions are obtained with trained weights by GA–IPT for neuron, i.e., k=10, 20, and 30, using Eqs. (20) and (21) along with the first sets in Eqs. (7) and (12). Results are given in Table 3 for input $t\in[0, 1]$ with step size h=0.1. The exact solution determined using Eq. (17) is also given in Table 3.

Table 3 Comparison of the proposed solution of GA–IPT with the exact solution for example 1 of problem I

t	$f(t)$ exact	$\tilde{f}(t)$			$\hat{f}(t)$		
		k=10	k=15	k=20	k=10	k=15	k=20
0	1.000 000	1.000 000	1.000 000	1.000 000	1.000 002	1.000 000	1.000 000
0.1	1.105 171	1.105 171	1.105 168	1.105 171	1.105 214	1.105 093	1.105 093
0.2	1.221 403	1.221 403	1.221 399	1.221 403	1.221 440	1.221 327	1.221 327
0.3	1.349 859	1.349 859	1.349 857	1.349 859	1.349 883	1.349 776	1.349 776
0.4	1.491 825	1.491 825	1.491 823	1.491 825	1.491 858	1.491 723	1.491 723
0.5	1.648 721	1.648 721	1.648 717	1.648 722	1.648 782	1.648 609	1.648 609
0.6	1.822 119	1.822 118	1.822 114	1.822 119	1.822 196	1.822 005	1.822 005
0.7	2.013 753	2.013 752	2.013 748	2.013 753	2.013 816	2.013 629	2.013 629
0.8	2.225 541	2.225 541	2.225 538	2.225 541	2.225 582	2.225 395	2.225 395
0.9	2.459 603	2.459 603	2.459 599	2.459 604	2.459 665	2.459 439	2.459 439
1.0	2.718 282	2.718 281	2.718 275	2.718 282	2.718 391	2.718 114	2.718 114

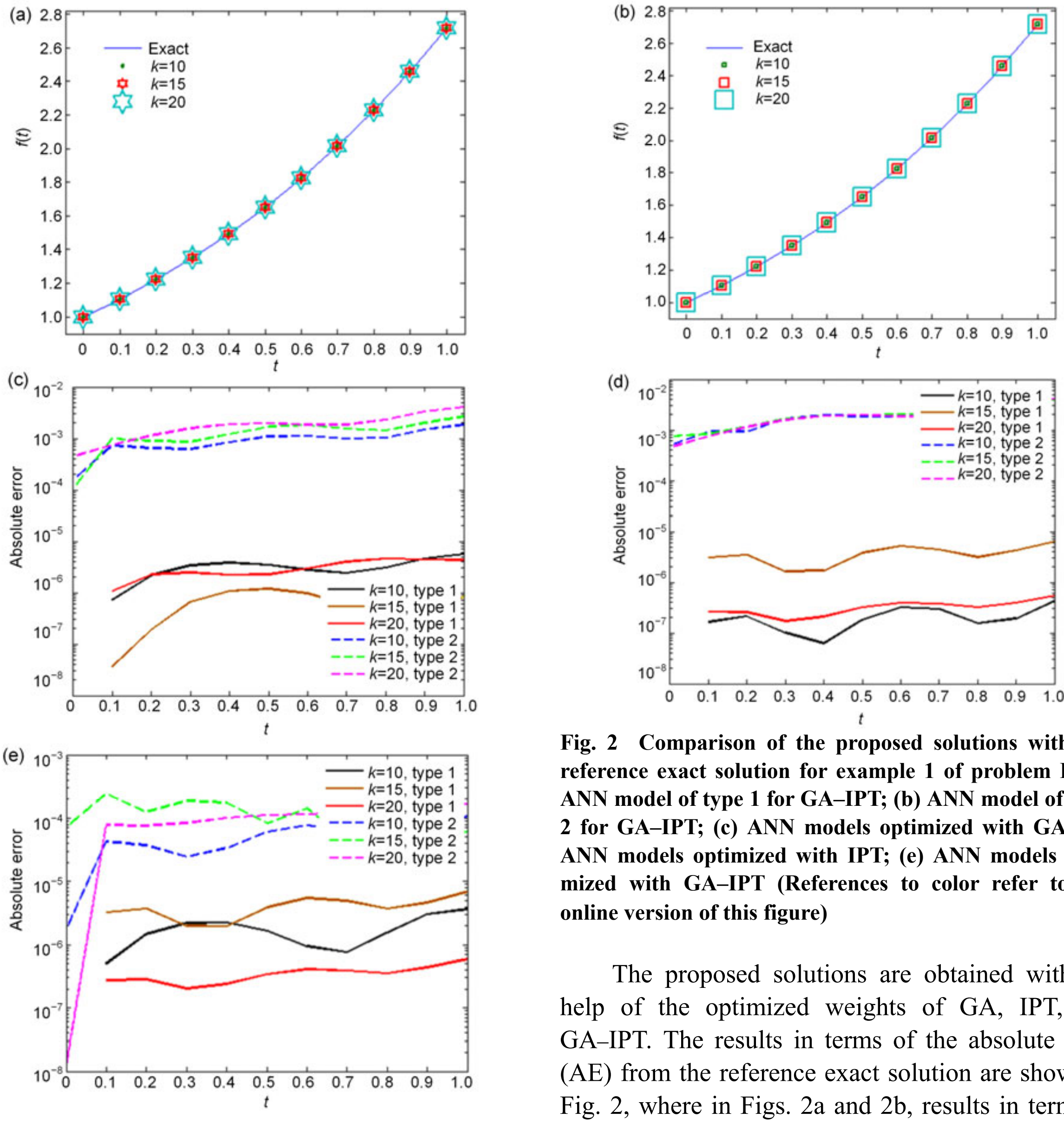

Fig. 2 Comparison of the proposed solutions with the reference exact solution for example 1 of problem I: (a) ANN model of type 1 for GA–IPT; (b) ANN model of type 2 for GA–IPT; (c) ANN models optimized with GA; (d) ANN models optimized with IPT; (e) ANN models optimized with GA–IPT (References to color refer to the online version of this figure)

The proposed solutions are obtained with the help of the optimized weights of GA, IPT, and GA–IPT. The results in terms of the absolute error (AE) from the reference exact solution are shown in Fig. 2, where in Figs. 2a and 2b, results in terms of

$$\begin{aligned}\tilde{f}(t)=\mathrm{e}^t+t^2\Bigg(&\frac{-0.0804}{1+\mathrm{e}^{-(0.1586t-1.3347)}}+\frac{0.3143}{1+\mathrm{e}^{-(0.4129t-1.0102)}}+\frac{-0.1894}{1+\mathrm{e}^{-(0.9333t-0.4303)}}+\frac{-0.3385}{1+\mathrm{e}^{-(0.0024t-0.2654)}}\\&+\frac{-0.18674}{1+\mathrm{e}^{-(0.7280t-0.4827)}}+\frac{-0.2081}{1+\mathrm{e}^{-(0.0806t-0.1575)}}+\frac{0.2899}{1+\mathrm{e}^{-(0.7425t-0.3296)}}+\frac{-0.0107}{1+\mathrm{e}^{-(-0.0654t-0.2871)}}\\&+\frac{0.0801}{1+\mathrm{e}^{-(0.9054t-0.4487)}}+\frac{0.3010}{1+\mathrm{e}^{-(-0.1943t-0.1444)}}\Bigg).\end{aligned}\tag{20}$$

$$\begin{aligned}\hat{f}(t)=&\frac{-0.0804}{1+\mathrm{e}^{-(0.1586t-1.3347)}}+\frac{0.3143}{1+\mathrm{e}^{-(0.4129t-1.0102)}}+\frac{-0.1894}{1+\mathrm{e}^{-(0.9333t-0.4303)}}+\frac{-0.3385}{1+\mathrm{e}^{-(0.0024t-0.2654)}}\\&+\frac{-0.1867}{1+\mathrm{e}^{-(0.7280t-0.4827)}}+\frac{-0.2081}{1+\mathrm{e}^{-(0.0806t-0.1575)}}+\frac{0.2899}{1+\mathrm{e}^{-(0.7425t-0.3296)}}+\frac{-0.0107}{1+\mathrm{e}^{-(-0.0654t-0.2871)}}\\&+\frac{0.0801}{1+\mathrm{e}^{-(0.9054t-0.4487)}}+\frac{0.3010}{1+\mathrm{e}^{-(-0.1943t-0.1444)}}.\end{aligned}\tag{21}$$

solution versus inputs are plotted, while in Figs. 2c, 2d, and 2e, the values of the corresponding AEs are shown by solid lines for the ANN models of type 1 and dashed lines for the ANN models of type 2 for all three algorithms. It can be inferred from Fig. 2c that for k=10, 15, and 20, the values of AE for GA lie around 10^{-6}, 10^{-6}–10^{-7}, and 10^{-6}, respectively, for the mathematical model based on Eq. (18), while for Eq. (19) they are around 10^{-3}–10^{-4}, 10^{-3}–10^{-4}, and 10^{-3}–10^{-4} for k=10, 15, and 20, respectively. Similarly, one can see from Fig. 2d that for k=10, 15, and 20, the values of AE for IPT lie around 10^{-6}, 10^{-6}–10^{-7}, and 10^{-6}, respectively, for the mathematical model based on Eq. (18), while for Eq. (19) they are around 10^{-3}–10^{-4}, 10^{-3}–10^{-4}, and 10^{-3}–10^{-4} for k=10, 15, and 20, respectively. On the other hand, the results of GA–IPT (Fig. 2e) show that for k=10, 15, and 20 the values of AE lie around 10^{-6}, 10^{-6}–10^{-7}, and 10^{-6}, respectively, for the mathematical model based on Eq. (18), while for Eq. (19) they are around 10^{-3}–10^{-4}, 10^{-3}–10^{-4}, and 10^{-3}–10^{-4} for k=10, 15, and 20, respectively.

Example 2 Consider another first-order nonlinear IVP of P-FDE:

$$\frac{\mathrm{d}f}{\mathrm{d}t}=1-2f^2(g(t)),\ f(0)=0,\tag{22}$$

which is derived from Eq. (1) by taking $z(f, f(g), t)$= $1-2f^2(g(t))$ and $g(t)=t/2$. The exact solution of Eq. (22) is

$$f(t)=\sin t.\tag{23}$$

The proposed design methodology is applied to a similar procedure to example 1; however, the fitness functions using N=10 and step size h=0.1 in this case are written for both types 1 and 2 of ANN models:

$$\varepsilon_{\mathrm{FO}}=\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}\tilde{f}}{\mathrm{d}t}-1+2\tilde{f}^2\left(\frac{t_m}{2}\right)\right]^2,\tag{24}$$

$$e_{\mathrm{FO}}=\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}\hat{f}_m}{\mathrm{d}t}-1+2\hat{f}_m^2\left(\frac{t_m}{2}\right)\right]^2+\hat{f}_0{}^2.\tag{25}$$

One set of optimized weights trained by GA–IPT for the number of neurons k=10, with fitness of around 10^{-8} for both ANN models, is used to obtain the derived solutions as given by Eqs. (26) and (27), shown on the next page.

The solutions presented in Eqs. (26) and (27) are provided in extended form in Appendix, Eqs. (A3) and (A4), respectively. Solutions are obtained using the trained weight of GA–IPT, and results are given in Table 4 for input $t\in[0, 1]$ with step size h=0.1 based on different numbers of neurons. The exact solution for the problem is also given in Table 4 for the same input parameters. The results in terms of AE are plotted in Fig. 3. In Figs. 3a and 3b, results in terms of solution versus inputs are plotted, while in Figs. 3c, 3d, and 3e, the values of the corresponding AEs are

$$\begin{aligned}\tilde{f}(t)=\sin t+t^2\bigg(&\frac{-0.9496}{1+\mathrm{e}^{-(0.3659t-0.9103)}}+\frac{-1.3470}{1+\mathrm{e}^{-(1.0241t-0.7515)}}+\frac{1.5037}{1+\mathrm{e}^{-(0.7085t-1.6240)}}+\frac{-0.0355}{1+\mathrm{e}^{-(1.3745t-0.1847)}}\\&+\frac{-0.4700}{1+\mathrm{e}^{-(-1.0889t-0.2060)}}+\frac{-0.2330}{1+\mathrm{e}^{-(-0.3844t-0.6442)}}+\frac{-0.0952}{1+\mathrm{e}^{-(-1.4891t-0.5765)}}+\frac{-0.1510}{1+\mathrm{e}^{-(-1.7607t-0.9299)}}\\&+\frac{-1.0914}{1+\mathrm{e}^{-(0.6036t-0.2050)}}+\frac{2.0344}{1+\mathrm{e}^{-(0.3910t-0.8486)}}\bigg).\end{aligned}\tag{26}$$

$$\begin{aligned}\hat{f}(t)=&\frac{0.3888}{1+\mathrm{e}^{-(1.1141t-2.2556)}}+\frac{-1.3191}{1+\mathrm{e}^{-(1.4621t-2.0752)}}+\frac{1.2244}{1+\mathrm{e}^{-(1.6263t-0.3115)}}+\frac{-0.9506}{1+\mathrm{e}^{-(-1.0389t-1.2599)}}\\&+\frac{1.2310}{1+\mathrm{e}^{-(1.9668t-1.9453)}}+\frac{-1.1211}{1+\mathrm{e}^{-(0.8882t-3.2592)}}+\frac{-0.9784}{1+\mathrm{e}^{-(0.7765t-0.9891)}}+\frac{-0.0031}{1+\mathrm{e}^{-(1.9169t-0.4591)}}\\&+\frac{-0.1830}{1+\mathrm{e}^{-(2.5442t-0.8153)}}+\frac{1.2423}{1+\mathrm{e}^{-(1.6713t-0.0094)}}.\end{aligned}\tag{27}$$

Table 4 Comparison of the proposed solution of GA–IPT with the exact solution for example 2 of problem I

t	$f(t)$ exact	$\tilde{f}(t)$			$\hat{f}(t)$		
		k=10	k=15	k=20	k=10	k=15	k=20
0	0	0	0	0	−1.62E−7	−7.20E−9	−1.62E−9
0.1	0.099 833	0.099 833	0.099 833	0.099 833	0.099 833	0.099 819	0.099 809
0.2	0.198 669	0.198 669	0.198 669	0.198 669	0.198 670	0.198 659	0.198 650
0.3	0.295 520	0.295 520	0.295 520	0.295 520	0.295 523	0.295 515	0.295 507
0.4	0.389 418	0.389 418	0.389 418	0.389 418	0.389 421	0.389 410	0.389 404
0.5	0.479 426	0.479 425	0.479 426	0.479 425	0.479 424	0.479 412	0.479 406
0.6	0.564 642	0.564 642	0.564 642	0.564 642	0.564 639	0.564 630	0.564 623
0.7	0.644 218	0.644 218	0.644 218	0.644 218	0.644 217	0.644 213	0.644 206
0.8	0.717 356	0.717 356	0.717 356	0.717 356	0.717 360	0.717 358	0.717 352
0.9	0.783 327	0.783 327	0.783 327	0.783 326	0.783 327	0.783 323	0.783 320
1.0	0.841 471	0.841 471	0.841 471	0.841 471	0.841 464	0.841 461	0.841 461

shown with solid lines for the ANN models of type 1 and dashed lines for the ANN models of type 2 for all three algorithms. It can be seen from the results that for k=10, 15, and 20, both methodologies give small values, while the ANN models with exactly satisfying initial conditions provide better accuracies.

Example 3 Consider another form of first-order IVP of P-FDEs with proportional delay in the forcing term:

$$\frac{\mathrm{d}f}{\mathrm{d}t}=-f(t)+\frac{1}{4}f(g(t))-\frac{1}{4}\mathrm{e}^{-t/2},\ f(0)=1,\tag{28}$$

which is derived from Eq. (1) by taking $z(f, f(t), t)=-f(t)+0.24f(g(t))-0.25\mathrm{e}^{-t/2}$ and $g(t)=t/2$. The exact solution of the problem is given by

$$f(t)=\mathrm{e}^{-t}.\tag{29}$$

In this case, the fitness functions based on the unsupervised error are formed using N=10 and step size h=0.1, and are given for both types 1 and 2 of ANN models:

$$\varepsilon_{\mathrm{FO}}=\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}\tilde{f}_m}{\mathrm{d}t}+\tilde{f}_m-\frac{1}{4}\tilde{f}\left(\frac{t_m}{2}\right)+\frac{1}{4}\mathrm{e}^{-\frac{t_m}{2}}\right]^2,\tag{30}$$

$$\begin{aligned}e_{\mathrm{FO}}=&\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}\hat{f}_m}{\mathrm{d}t}+f_m-\frac{1}{4}\hat{f}\left(\frac{t_m}{2}\right)+\frac{1}{4}\mathrm{e}^{-\frac{t_m}{2}}\right]^2\\&+(\hat{f}_0-1)^2.\end{aligned}\tag{31}$$

The fitness functions given in Eqs. (30) and (31) are optimized with GA, IPT, and the hybrid approach GA–IPT. The solutions derived using one set of

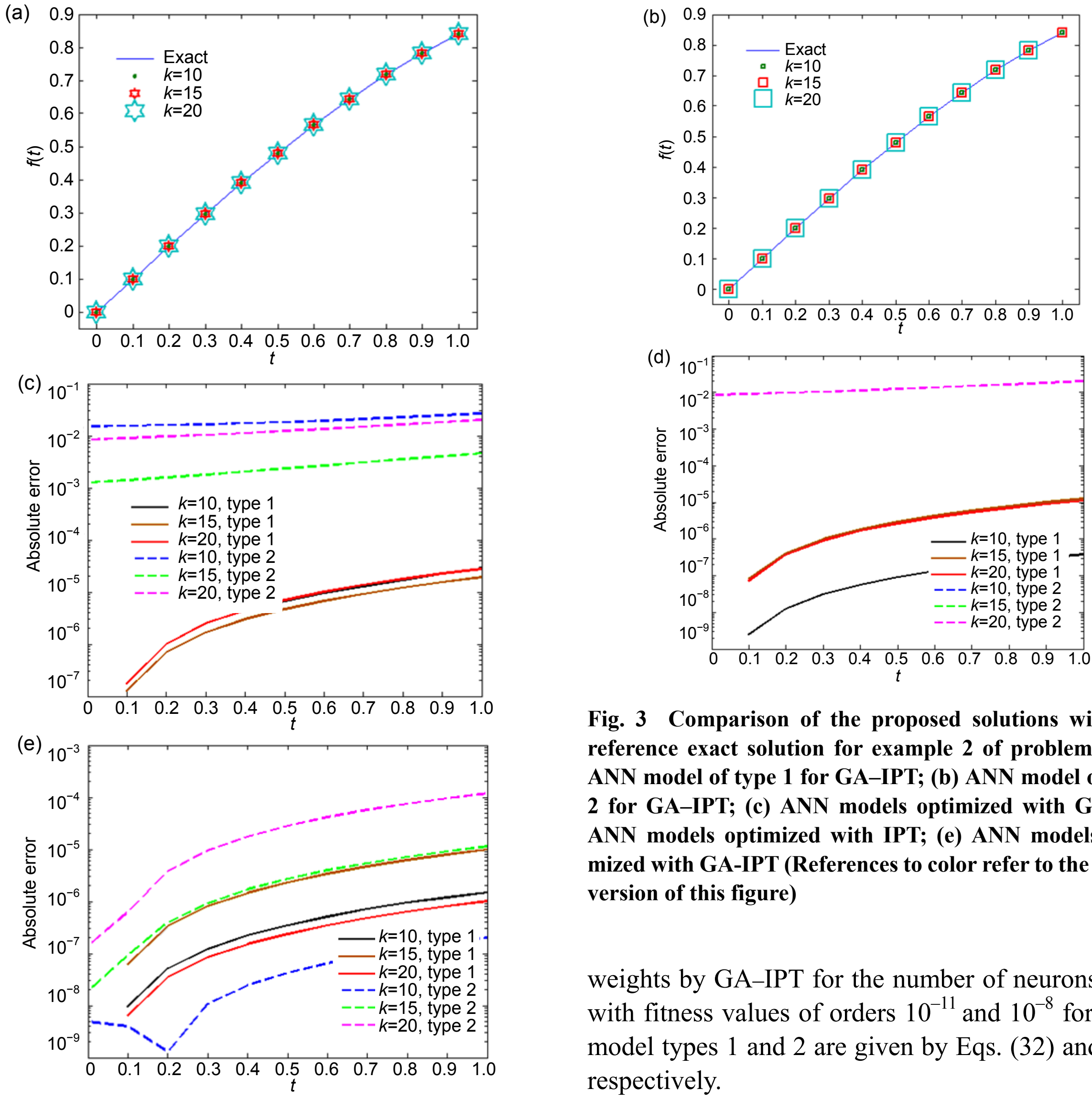

Fig. 3 Comparison of the proposed solutions with the reference exact solution for example 2 of problem I: (a) ANN model of type 1 for GA–IPT; (b) ANN model of type 2 for GA–IPT; (c) ANN models optimized with GA; (d) ANN models optimized with IPT; (e) ANN models optimized with GA-IPT (References to color refer to the online version of this figure)

weights by GA–IPT for the number of neurons k=10 with fitness values of orders 10^{-11} and 10^{-8} for ANN model types 1 and 2 are given by Eqs. (32) and (33), respectively.

$$\begin{aligned}\tilde{f}(t)={}&\mathrm{e}^{-t}+t^2\left(\frac{-1.12541}{1+\mathrm{e}^{-(-0.8641t-1.9350)}}+\frac{1.4210}{1+\mathrm{e}^{-(-1.1903t-0.3159)}}+\frac{-0.7202}{1+\mathrm{e}^{-(-0.2772t-0.1214)}}+\frac{-0.5391}{1+\mathrm{e}^{-(0.9870t-1.1302)}}\right.\\&+\frac{1.7714}{1+\mathrm{e}^{-(0.8935t-0.6601)}}+\frac{-0.7534}{1+\mathrm{e}^{-(0.0059t-0.0656)}}+\frac{-1.2487}{1+\mathrm{e}^{-(0.5566t-1.6375)}}+\frac{0.4799}{1+\mathrm{e}^{-(0.0252t-1.6337)}}\\&\left.+\frac{-0.8303}{1+\mathrm{e}^{-(-1.5402t-1.4391)}}+\frac{1.6996}{1+\mathrm{e}^{-(-0.7958t-0.4921)}}\right).\end{aligned}\tag{32}$$

$$\begin{aligned}\hat{f}(t)={}&\frac{0.0393}{1+\mathrm{e}^{-(-3.5121t-1.0534)}}+\frac{3.6192}{1+\mathrm{e}^{-(0.0334t-2.4531)}}+\frac{-0.5867}{1+\mathrm{e}^{-(-0.9518t-0.6042)}}+\frac{-2.7176}{1+\mathrm{e}^{-(2.1911t-1.7109)}}\\&+\frac{0.9885}{1+\mathrm{e}^{-(-0.3528t-0.4800)}}+\frac{-1.7338}{1+\mathrm{e}^{-(0.8103t-4.7230)}}+\frac{0.1041}{1+\mathrm{e}^{-(-0.8925t-0.2514)}}+\frac{-0.4657}{1+\mathrm{e}^{-(1.5841t-1.0946)}}\\&+\frac{-1.7886}{1+\mathrm{e}^{-(0.0901t-2.5349)}}+\frac{0.1960}{1+\mathrm{e}^{-(-1.8923t-1.0802)}}.\end{aligned}\tag{33}$$

The solutions presented in Eqs. (26) and (27) are provided in extended form in Appendix, Eqs. (A5) and (A6), respectively. The approximate solutions are also calculated with the weights of the GA–IPT algorithm for different neurons, i.e., k=10, 20, and 30. The results are listed in Table 5 for $t\in[0, 1]$ with step size h=0.1 in terms of AEs (Fig. 4). In Figs. 4a and 4b, results in terms of solution versus inputs are plotted, while in Figs. 4c, 4d, and 4e, the values of corresponding AEs are shown with solid lines for the ANN models of type 1, and dashed lines for the ANN models of type 2 for all three algorithms. The exact solution is also given in Table 5 for comparison. It is seen that the proposed results overlap with those of the standard solution, and small values of AE are achieved as well.

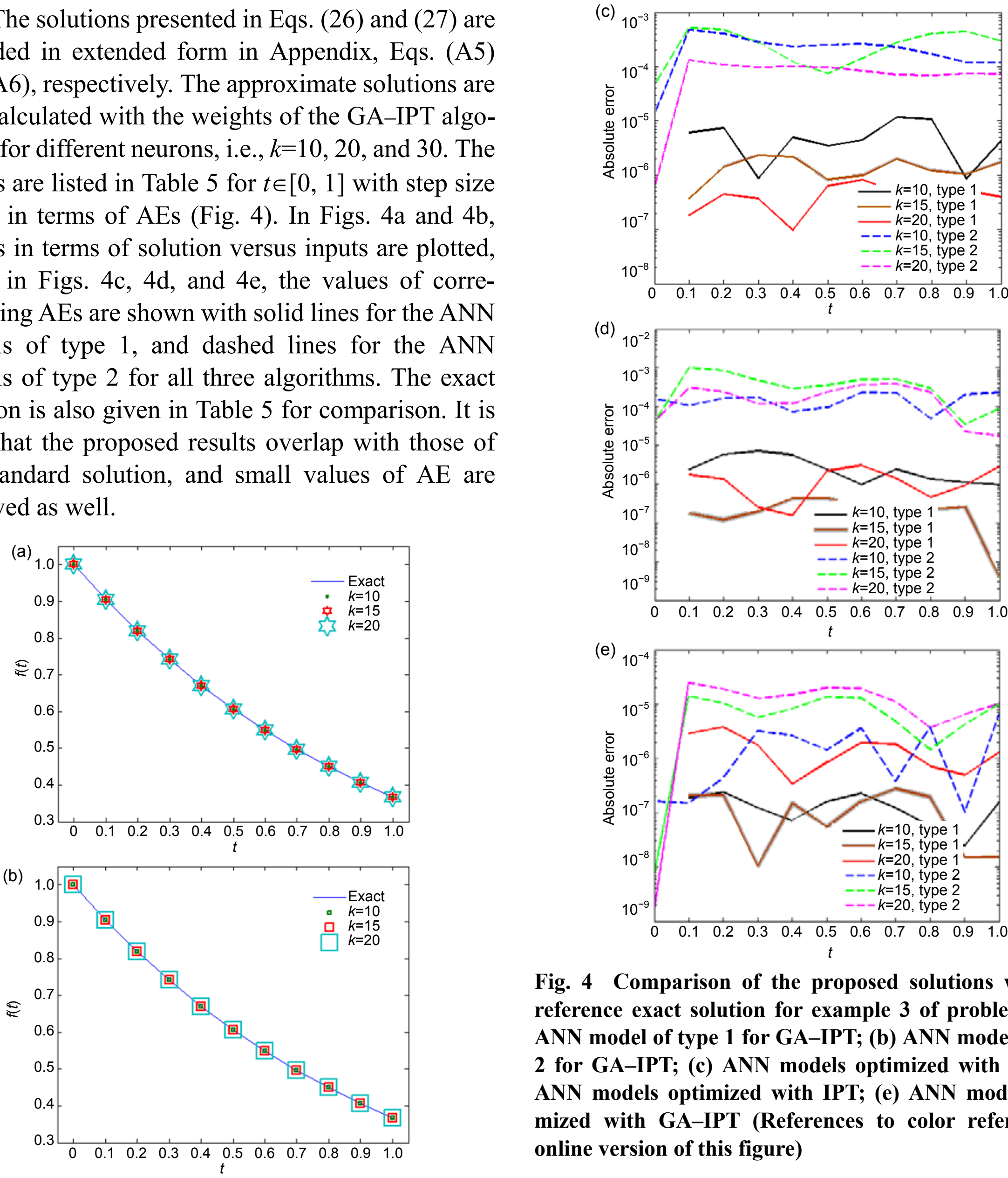

Fig. 4 Comparison of the proposed solutions with the reference exact solution for example 3 of problem I: (a) ANN model of type 1 for GA–IPT; (b) ANN model of type 2 for GA–IPT; (c) ANN models optimized with GA; (d) ANN models optimized with IPT; (e) ANN models optimized with GA–IPT (References to color refer to the online version of this figure)

Table 5 Comparison of the proposed solution of GA–IPT with the exact solution for example 3 of problem I

t	$f(t)$ exact	$\tilde{f}(t)$			$\hat{f}(t)$		
		k=10	k=15	k=20	k =10	k=15	k=20
0	1.000 000	1.000 000	1.000 000	1.000 000	1.000 000	1.000 000	1.000 000
0.1	0.904 837	0.904 837	0.904 839	0.904 836	0.904 863	0.904 863	0.904 907
0.2	0.818 731	0.818 730	0.818 732	0.818 730	0.818 752	0.818 752	0.818 789
0.3	0.740 818	0.740 818	0.740 819	0.740 818	0.740 839	0.740 839	0.740 874
0.4	0.670 320	0.670 320	0.670 321	0.670 32o	0.670 340	0.670 340	0.670 375
0.5	0.606 531	0.606 530	0.606 532	0.606 530	0.606 548	0.606 549	0.606 579
0.6	0.548 812	0.548 811	0.548 813	0.548 811	0.548 828	0.548 828	0.548 855
0.7	0.496 585	0.496 585	0.496 585	0.496 585	0.496 601	0.496 601	0.496 627
0.8	0.449 329	0.449 329	0.449 329	0.449 329	0.449 345	0.449 345	0.449 370
0.9	0.406 570	0.406 570	0.406 570	0.406 569	0.406 583	0.406 584	0.406 607
1.0	0.367 879	0.367 879	0.367 880	0.367 880	0.367 892	0.367 892	0.367 913

5.2 Problem II: second-order P-FDE

In this case, we evaluate the performance of the proposed scheme on the second-order IVP of P-FDE as

$$\begin{aligned}&\frac{\mathrm{d}^2 f}{\mathrm{d}t^2}=\frac{3}{4}f(t)+f(g(t))-t^2+2,\\ &f(0)=0,\ \frac{\mathrm{d}}{\mathrm{d}t}f(0)=0,\ g(t)=t/2,\end{aligned}\tag{34}$$

which is derived from Eq. (2) by taking function z as

$$z(f,\ f(g),t)=\frac{3}{4}f(t)+f(g(t))-t^2+2.\tag{35}$$

The exact solution of Eq. (34) is given by

$$f(t)=t^2.\tag{36}$$

To solve this problem, two fitness functions are formulated by taking N=10 and step size h=0.1, as

$$\varepsilon_{\mathrm{SO}}=\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}\tilde{f}_m}{\mathrm{d}t}+\tilde{f}_m-\frac{1}{4}\tilde{f}\left(\frac{t_m}{2}\right)+\frac{1}{4}\mathrm{e}^{-\frac{t_m}{2}}\right]^2,\tag{37}$$

$$\begin{aligned}e_{\mathrm{SO}}=&\frac{1}{10}\sum_{m=1}^{10}\left[\frac{\mathrm{d}^2\hat{f}_m}{\mathrm{d}t^2}-\frac{3}{4}\hat{f}_m-\hat{f}\left(\frac{t_m}{2}\right)+t_m^2-2\right]^2\\ &+\frac{1}{2}\left[\hat{f}_0^{\,2}+\left(\frac{\mathrm{d}}{\mathrm{d}t}\hat{f}_0\right)^2\right].\end{aligned}\tag{38}$$

To obtain the unknown parameters for fitness functions in Eqs. (37) and (38), GA, IPT, and the hybrid approach GA–IPT are applied, and the approximate solutions are expressed mathematically with one set of weights by GA–IPT for k=10 with fitness values of orders 10^{-7} and 10^{-6} as Eqs. (39) and (40), respectively, shown on the next page.

The solutions presented in Eqs. (39) and (40) are provided in extended form in Appendix, Eqs. (A7) and (A8), respectively. The proposed solutions are determined by weights trained by GA–IPT for k=10, 20, and 30, and the results with exact solutions are given in Table 6 for input $t\in[0, 1]$ with step size h=0.1. The results of AEs for both ANN models are shown in Fig. 5. In case of Figs. 5a and 5b, results in terms of solution versus inputs are plotted, while in Figs. 5c, 5d, and 5e, the values of corresponding AEs are shown with solid lines for the ANN models of type 1 and dashed lines for the ANN models of type 2 for all three algorithms. It can be seen that the given solutions overlap with the exact solutions. It is seen from Fig. 5c that for k=10, 15, and 20, the values of AE for GA are around 10^{-4}, 10^{-5}–10^{-6}, and 10^{-6}–10^{-7}, respectively, by optimizing Eq. (37), while optimizing Eq. (38) gives the values of AE as 10^{-3}–10^{-4}, 10^{-3}, and 10^{-3} for k=10, 15, and 20, respectively. Further, one can deduce from Fig. 5d that for k=10, 15, and 20, the values of AE for IPT are between 10^{-6} and 10^{-7} for type 1 model in Eq. (37), while for another model (Eq. (38)), the values of AE are also within 10^{-6}–10^{-7} for k=10, 15, and 20. Accordingly, the results of GA–IPT in Fig. 5e show that for k=10, 15, and 20, the values of AE lie in ranges 10^{-4}–10^{-5}, 10^{-5}–10^{-6}, and around 10^{-7}–10^{-8}, respectively, for the type 1 ANN model (Eq. (37)), while for the type 2 ANN model (Eq. (38)) the values of AE are 10^{-4}–10^{-8}, 10^{-4}–10^{-9}, and 10^{-4}–10^{-9} for k=10, 15, and 20, respectively.

Table 6 Comparison of the proposed solution of GA–IPT with the exact solution for problem II

t	$f(t)$ exact	$\tilde{f}(t)$			$\hat{f}(t)$		
		k=10	k=15	k=20	k=10	k=15	k=20
0	0	0	0	0	−3.46E−8	−7.55E−10	−1.56E−9
0.1	0.010 000	0.010 000	0.010 000	0.010 000	0.009 995	0.009 996	0.010 005
0.2	0.040 000	0.040 001	0.039 999	0.039 999	0.039 989	0.039 990	0.040 012
0.3	0.090 000	0.090 001	0.089 999	0.089 999	0.089 984	0.089 984	0.090 019
0.4	0.160 000	0.160 001	0.159 998	0.159 999	0.159 977	0.159 978	0.160 027
0.5	0.250 000	0.250 002	0.249 998	0.249 998	0.249 970	0.249 971	0.250 034
0.6	0.360 000	0.360 002	0.359 997	0.359 998	0.359 963	0.359 965	0.360 042
0.7	0.490 000	0.490 002	0.489 997	0.489 998	0.489 957	0.489 958	0.490 051
0.8	0.640 000	0.640 003	0.639 996	0.639 997	0.639 949	0.639 950	0.640 060
0.9	0.810 000	0.810 003	0.809 996	0.809 997	0.809 940	0.809 942	0.810 069
1.0	1.000 000	1.000 004	0.999 995	0.999 996	0.999 931	0.999 933	1.000 080

$$\begin{aligned}\tilde{f}(t)=t^2+t^2\Bigg(&\frac{0.6748}{1+\mathrm{e}^{-(0.7242t-0.0173)}}+\frac{0.5449}{1+\mathrm{e}^{-(-0.4061t-0.3663)}}+\frac{-0.4904}{1+\mathrm{e}^{-(-0.2772t-0.5385)}}+\frac{0.0372}{1+\mathrm{e}^{-(-0.7469t-0.8472)}}\\&+\frac{-0.2313}{1+\mathrm{e}^{-(0.7242t-0.0480)}}+\frac{0.1861}{1+\mathrm{e}^{-(0.7242t-1.1941)}}+\frac{-0.9102}{1+\mathrm{e}^{-(0.6772t-0.2389)}}+\frac{0.0466}{1+\mathrm{e}^{-(0.6888t-0.3597)}}\\&+\frac{0.9887}{1+\mathrm{e}^{-(-0.0827t-0.2227)}}+\frac{1.0436}{1+\mathrm{e}^{-(0.3359t-0.4375)}}\Bigg).\end{aligned} \tag{39}$$

$$\begin{aligned}\hat{f}(t)=&\frac{2.3576}{1+\mathrm{e}^{-(1.4204t-1.5204)}}+\frac{2.0152}{1+\mathrm{e}^{-(1.0376t-1.2196)}}+\frac{-1.7391}{1+\mathrm{e}^{-(2.7948t-3.3269)}}+\frac{-1.5842}{1+\mathrm{e}^{-(1.5079t-0.0606)}}\\&+\frac{3.4531}{1+\mathrm{e}^{-(0.2616t-1.2650)}}+\frac{1.4274}{1+\mathrm{e}^{-(1.4536t-3.6760)}}+\frac{-1.4325}{1+\mathrm{e}^{-(2.1884t-1.5155)}}+\frac{0.4409}{1+\mathrm{e}^{-(0.8949t-3.9208)}}\\&+\frac{4.9585}{1+\mathrm{e}^{-(1.8723t-4.1947)}}+\frac{-0.7523}{1+\mathrm{e}^{-(-0.1993t-2.2551)}}.\end{aligned} \tag{40}$$

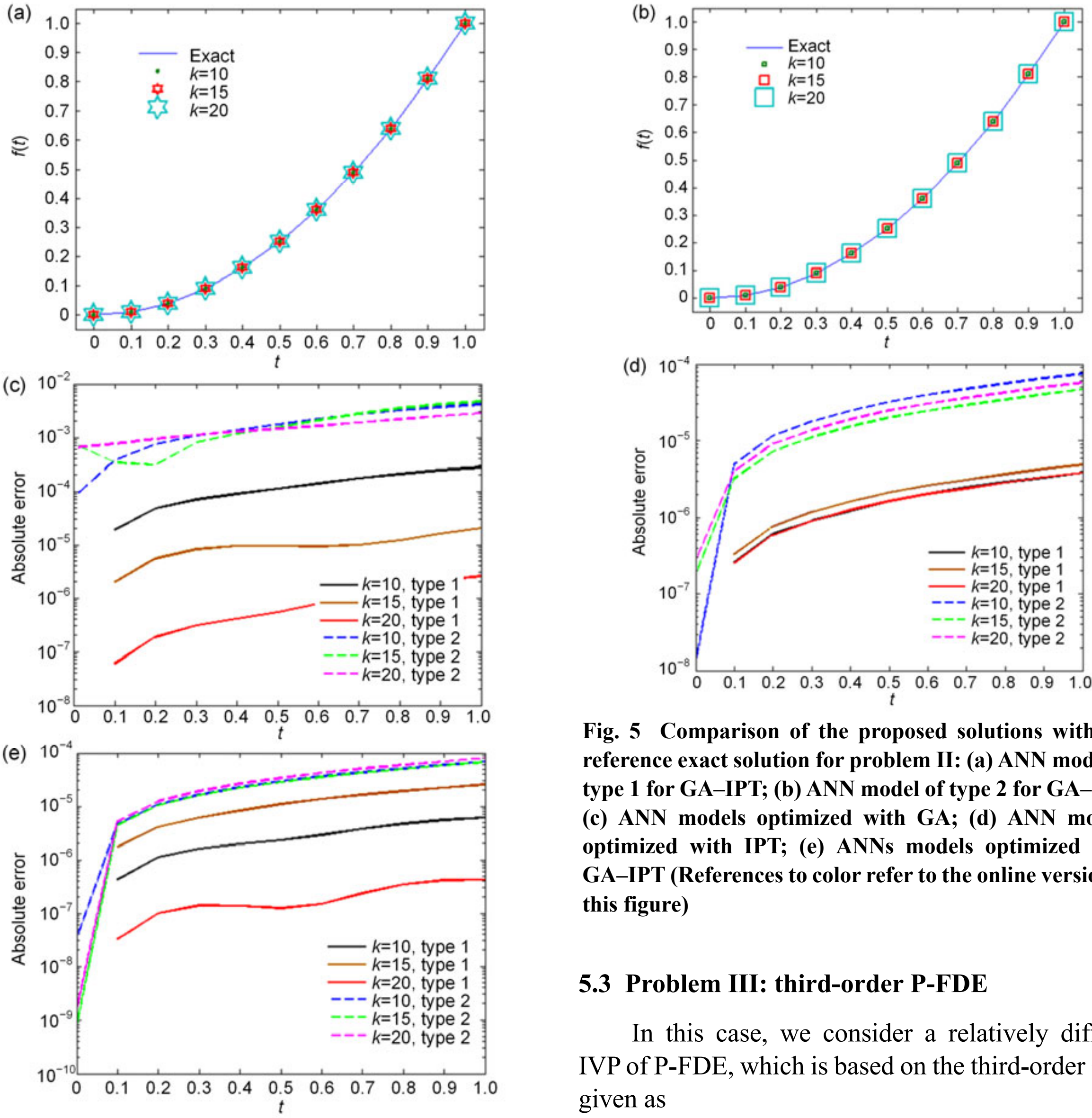

Fig. 5 Comparison of the proposed solutions with the reference exact solution for problem II: (a) ANN model of type 1 for GA–IPT; (b) ANN model of type 2 for GA–IPT; (c) ANN models optimized with GA; (d) ANN models optimized with IPT; (e) ANNs models optimized with GA–IPT (References to color refer to the online version of this figure)

5.3 Problem III: third-order P-FDE

In this case, we consider a relatively difficult IVP of P-FDE, which is based on the third-order ODE, given as

$$\frac{\mathrm{d}^3 f}{\mathrm{d}t^3}=-1+2(f(g(t)))^2,$$
$$f(0)=0,\ \frac{\mathrm{d}}{\mathrm{d}t}f(0)=1,\ \frac{\mathrm{d}^2}{\mathrm{d}t^2}f(0)=0,\ g(t)=t/2, \tag{41}$$

which is derived from Eq. (3), and function z is given as

$$z(f,f(g),t)=-1+2(f(g(t)))^2. \tag{42}$$

The exact solution of Eq. (41) is given by

$$f(t)=\sin t. \tag{43}$$

The proposed design methodology is applied to find the approximate solution of this IVP as well; however, the merit or fitness functions using N=10 and step size h=0.1 are constructed for this equation as

$$\varepsilon_{\mathrm{TO}}=\frac{1}{10}\sum_{m=1}^{10}\left\{\frac{\mathrm{d}^3\tilde{f}_m}{\mathrm{d}t^3}+1-2\left[\tilde{f}\left(\frac{t_m}{2}\right)\right]^2\right\}^2, \tag{44}$$

$$\begin{aligned}e_{\mathrm{TO}}=&\frac{1}{10}\sum_{m=1}^{10}\left\{\frac{\mathrm{d}^3\hat{f}_m}{\mathrm{d}t^3}+1-2\left[\hat{f}\left(\frac{t_m}{2}\right)\right]^2\right\}^2\\&+\frac{1}{3}\left[(\hat{f}_0)^2+\left(\frac{\mathrm{d}}{\mathrm{d}t}\hat{f}_0-1\right)^2+\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2}\hat{f}_0\right)^2\right].\end{aligned} \tag{45}$$

To obtain the weights for optimizing the fitness functions given in Eqs. (44) and (45), the strength of GA, IPT, and the hybrid approach GA–IPT is exploited. One set of learned weights by GA–IPT for k=10 for the two ANN models are written as Eqs. (46) and (47).

The solutions presented in Eqs. (46) and (47) are provided in extended form in Appendix, Eqs. (A9) and (A10), respectively. The approximate solutions are found with the adapted weights of GA–IPT for different neuron numbers, i.e., k=10, 20, and 30. The results are given in Table 7 for input $t\in[0, 1]$ with step size h=0.1. The exact solutions are shown in Table 7. The estimated solutions are calculated with the weights given by GA, IPT, and GA–IPT. The results based on AEs are shown in Fig. 6. In Figs. 6a and 6b, results in terms of the solution versus inputs are plotted, while in Figs. 6c, 6d, and 6e, the values of the corresponding AEs are shown with solid lines for the ANN models of type 1 and dashed lines for the ANN models of type 2 for all three algorithms. It is seen from the results that there is a close match between the given and the exact solutions. From the results of GA–IPT (Fig. 6e), one can infer that for k=10, 15, and 20, the values of AE lie in ranges 10^{-5}–10^{-6}, 10^{-5}–10^{-6}, and 10^{-5}–10^{-7}, respectively, for the type 1 ANN model (Eq. (44)), while for the type 2 ANN model (Eq. (45)), they are around 10^{-4}–10^{-6}, 10^{-4}–10^{-5}, and 10^{-4}–10^{-6} for k=10, 15, and 20, respectively.

6 Conclusions

The following conclusions are drawn on the basis of numerical experiments performed in this study:

1. A new heuristic computational intelligence technique is developed to solve IVPs of P-FDEs effectively by exploiting the strength of ANN models, GA, IPT, and their hybrid approach GA–IPT.

2. A comparison of the proposed approximate solutions with existing exact solutions shows that,

$$\begin{aligned}\tilde{f}(t)=\sin t+t^2\Bigg(&\frac{0.3858}{1+\mathrm{e}^{-(-0.9242t-1.0047)}}+\frac{0.6178}{1+\mathrm{e}^{-(-0.2237t-0.1588)}}+\frac{-0.4950}{1+\mathrm{e}^{-(0.3636t-0.6231)}}+\frac{0.0060}{1+\mathrm{e}^{-(0.7295t-0.6231)}}\\&+\frac{-0.6360}{1+\mathrm{e}^{-(-0.5584t-0.0036)}}+\frac{0.1211}{1+\mathrm{e}^{-(0.1745t-0.3059)}}+\frac{0.3131}{1+\mathrm{e}^{-(0.2311t-0.1747)}}+\frac{-0.3690}{1+\mathrm{e}^{-(-0.9056t-0.9262)}}\\&+\frac{0.1313}{1+\mathrm{e}^{-(-0.8040t-0.04617)}}+\frac{-0.0395}{1+\mathrm{e}^{-(0.9470t-0.1381)}}\Bigg).\end{aligned} \tag{46}$$

$$\begin{aligned}\hat{f}(t)=&\frac{0.3858}{1+\mathrm{e}^{-(-0.9242t-1.0047)}}+\frac{0.6178}{1+\mathrm{e}^{-(-0.2237t-0.1588)}}+\frac{-0.4950}{1+\mathrm{e}^{-(0.3636t-0.6231)}}+\frac{0.0060}{1+\mathrm{e}^{-(0.7295t-0.6231)}}\\&+\frac{-0.6360}{1+\mathrm{e}^{-(-0.5584t-0.0036)}}+\frac{0.1211}{1+\mathrm{e}^{-(0.1745t-0.3059)}}+\frac{0.3131}{1+\mathrm{e}^{-(0.2311t-0.1747)}}+\frac{-0.3690}{1+\mathrm{e}^{-(-0.9056t-0.9262)}}\\&+\frac{0.1313}{1+\mathrm{e}^{-(-0.8040t-0.0461)}}+\frac{-0.0395}{1+\mathrm{e}^{-(0.9470t-0.1381)}}.\end{aligned} \tag{47}$$

Table 7 Comparison of the proposed solution of GA–IPT with the exact solution for problem III

t	$f(t)$ exact	$\tilde{f}(t)$			$\hat{f}(t)$		
		k=10	k=15	k=20	k=10	k=15	k=20
0	0	0	0	0	4.88E–9	1.89E–8	1.38E–7
0.1	0.099 833	0.099 833	0.099 833	0.099 833	0.099 833	0.099 834	0.099 833
0.2	0.198 669	0.198 669	0.198 670	0.198 669	0.198 669	0.198 670	0.198 666
0.3	0.295 520	0.295 520	0.295 521	0.295 519	0.295 520	0.295 521	0.295 511
0.4	0.389 418	0.389 418	0.389 420	0.389 417	0.389 418	0.389 420	0.389 401
0.5	0.479 426	0.479 425	0.479 429	0.479 423	0.479 425	0.479 428	0.479 397
0.6	0.564 642	0.564 642	0.564 647	0.564 639	0.564 642	0.564 646	0.564 601
0.7	0.644 218	0.644 218	0.644 224	0.644 212	0.644 218	0.644 223	0.644 161
0.8	0.717 356	0.717 356	0.717 364	0.717 349	0.717 356	0.717 363	0.717 281
0.9	0.783 327	0.783 327	0.783 337	0.783 318	0.783 327	0.783 336	0.783 231
1.0	0.841 471	0.841 471	0.841 483	0.841 460	0.841 471	0.841 482	0.841 352

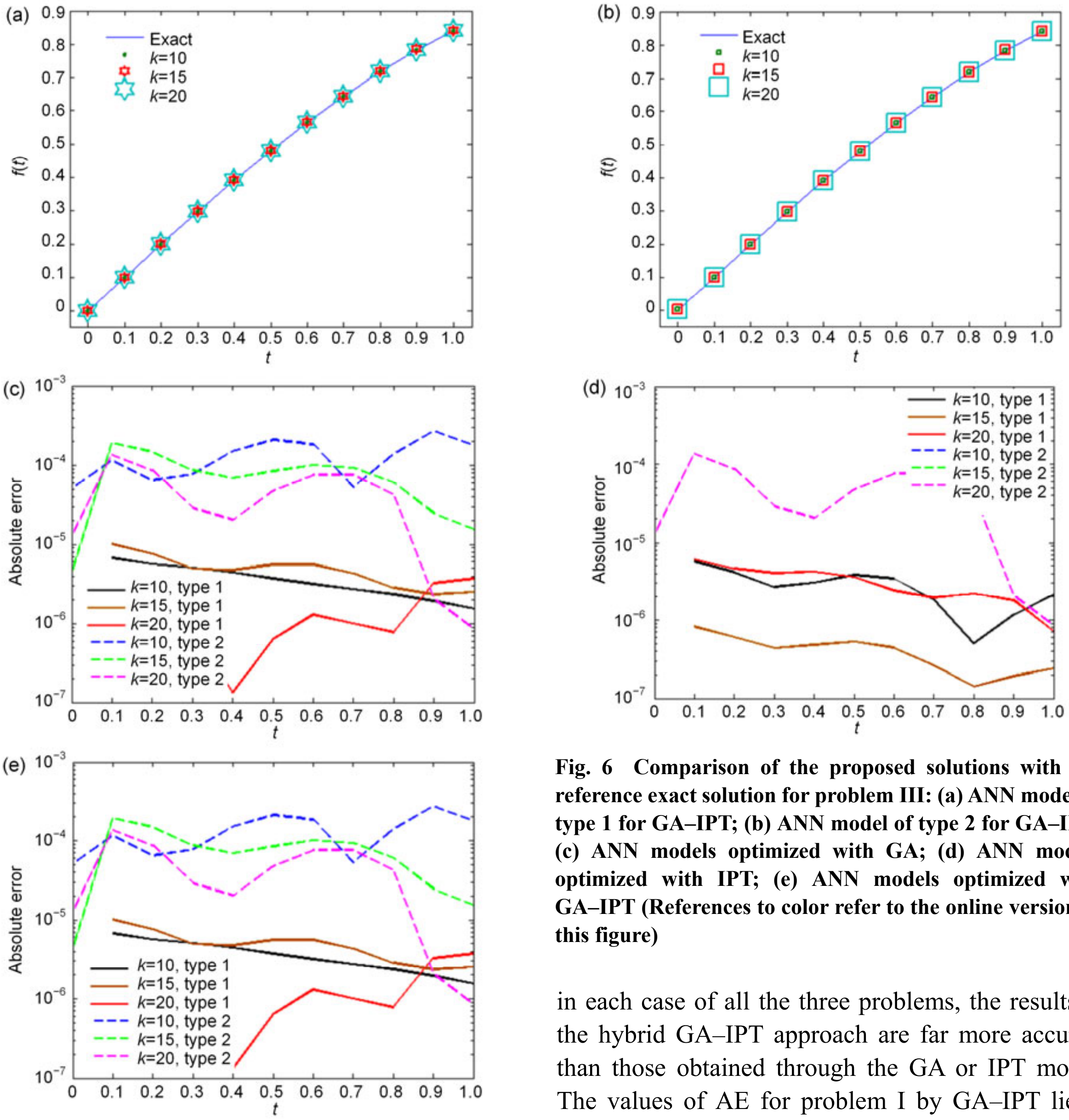

Fig. 6 Comparison of the proposed solutions with the reference exact solution for problem III: (a) ANN model of type 1 for GA–IPT; (b) ANN model of type 2 for GA–IPT; (c) ANN models optimized with GA; (d) ANN models optimized with IPT; (e) ANN models optimized with GA–IPT (References to color refer to the online version of this figure)

in each case of all the three problems, the results of the hybrid GA–IPT approach are far more accurate than those obtained through the GA or IPT model. The values of AE for problem I by GA–IPT lie in ranges 10^{-3}–10^{-7}, 10^{-5}–10^{-7}, and 10^{-4}–10^{-10} for examples 1, 2, and 3, respectively, while for problems II and III the values of AE lie in ranges 10^{-4}–10^{-8} and 10^{-4}–10^{-6}, respectively.

3. The behavior of the proposed scheme, based on the ANN model with different numbers of neurons, shows a slight gain in terms of the precision of the results with an increasing number of neurons in the modeling; however, no drastic change in terms of accuracy has been observed. On the other hand, the time of the computations has increased exponentially by increasing the number of neurons in the modeling.

4. Besides the consistent accuracy and convergence of the proposed scheme, other advantages are the simplicity of the concept, ease in implementations, provision of results on the entire continuous grid of inputs, and easily extendable methodology for different applications. All the features establish the intrinsic worth of the scheme as a good alternative, accurate, reliable, and robust computing platform for stiff nonlinear systems such as the pantograph.

Based on the presented study, the following research directions are suggested for those interested in this domain:

1. Improvement of results is possible by investigating ANN modeling either by introducing new transfer functions or by changes in the type of neural networks. Activation functions such as the radial basis function, tag-sigmoid, and modern models based on Mexican and wavelet hat neural networks should be tried in this regard.

2. One can incorporate modern optimization algorithms and their hybrid combination with the efficient local search method to obtain the design parameters of neural network models with better capabilities to improve the results. In this regard, fractional variants of the PSO algorithm, chaos optimization algorithm, evolutionary strategies, genetic programming, and differential evolution can be good alternatives.

3. Availability of better hardware platforms and modern software packages can also play a role in exploring the capability of the algorithm in a much wider search space, easily and efficiently.

References

Agarwal, R.P., Chow, Y.M., 1986. Finite difference methods for boundary-value problems of differential equations with deviating arguments. *Comput. Math. Appl.*, **12**(11): 1143-1153.
http://dx.doi.org/10.1016/0898-1221(86)90018-0

Arqub, O.A., Zaer, A.H., 2014. Numerical solution of systems of second-order boundary value problems using continuous genetic algorithm. *Inform. Sci.*, **279**:396-415.
http://dx.doi.org/10.1016/j.ins.2014.03.128

Azbelev, N.V., Maksimov, V.P., Rakhmatullina, L.F., 2007. Intoduction to the Theory of Functional Differential Equations: Methods and Applications. Hindawi Publishing Corporation, New York, USA.
http://dx.doi.org/10.1155/9789775945495

Barro, G., So, O., Ntaganda, J.M., *et al.*, 2008. A numerical method for some nonlinear differential equation models in biology. *Appl. Math. Comput.*, **200**(1):28-33.
http://dx.doi.org/10.1016/j.amc.2007.10.041

Chakraverty, S., Mall, S., 2014. Regression-based weight generation algorithm in neural network for solution of initial and boundary value problems. *Neur. Comput. Appl.*, **25**(3):585-594.
http://dx.doi.org/10.1007/s00521-013-1526-4

Dehghan, M., Salehi, R., 2010. Solution of a nonlinear time-delay model in biology via semi-analytical approaches. *Comput. Phys. Commun.*, **181**:1255-1265.
http://dx.doi.org/10.1016/j.cpc.2010.03.014

Derfel, G., Iserles, A., 1997. The pantograph equation in the complex plane. *J. Math. Anal. Appl.*, **213**(1):117-132.
http://dx.doi.org/10.1006/jmaa.1997.5483

Evans, D.J., Raslan, K.R., 2005. The Adomian decomposition method for solving delay differential equation. *Int. J. Comput. Math.*, **82**(1):49-54.
http://dx.doi.org/10.1080/00207160412331286815

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. The University of Michigan Press, Ann Arbor, USA.

Iserles, A., 1993. On the generalized pantograph functional-differential equation. *Eur. J. Appl. Math.*, **4**(1):1-38.
http://dx.doi.org/10.1017/S0956792500000966

Khan, J.A., Raja, M.A.Z., Qureshi, I.M., 2011. Novel approach for van der Pol oscillator on the continuous time domain. *Chin. Phys. Lett.*, **28**:110205.
http://dx.doi.org/10.1088/0256-307X/28/11/110205

Khan, J.A., Raja, M.A.Z., Syam, M.A., *et al.*, 2015. Design and application of nature inspired computing approach for non-linear stiff oscillatory problems. *Neur. Comput. Appl.*, **26**(7):1763-1780.
http://dx.doi.org/10.1007/s00521-015-1841-z

Mall, S., Chakraverty, S., 2014a. Chebyshev neural network based model for solving Lane–Emden type equations. *Appl. Math. Comput.*, **247**:100-114.
http://dx.doi.org/10.1016/j.amc.2014.08.085

Mall, S., Chakraverty, S., 2014b. Numerical solution of nonlinear singular initial value problems of Emden–Fowler type using Chebyshev neural network method. *Neurocomputing*, **149**(B):975-982.
http://dx.doi.org/10.1016/j.neucom.2014.07.036

McFall, K.S., 2013. Automated design parameter selection for neural networks solving coupled partial differential equations with discontinuities. *J. Franklin Inst.*, **350**(2): 300-317.
http://dx.doi.org/10.1016/j.jfranklin.2012.11.003

Ockendon, J.R., Tayler, A.B., 1971. The dynamics of a current collection system for an electric locomotive. *Proc. R. Soc. A*, **322**(1551):447-468.
http://dx.doi.org/10.1098/rspa.1971.0078

Pandit, S., Kumar, M., 2014. Haar wavelet approach for numerical solution of two parameters singularly perturbed boundary value problems. *Appl. Math. Inform. Sci.*, **8**(6): 2965-2974.

Peng, Y.G., Jun, W., Wei, W., 2014. Model predictive control of servo motor driven constant pump hydraulic system in injection molding process based on neurodynamic optimization. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **15**(2):139-146.
http://dx.doi.org/10.1631/jzus.C1300182

Potra, F.A., Wright, S.J., 2000. Interior-point methods. *J. Comput. Appl. Math.*, **124**(1-2):281-302.
http://dx.doi.org/10.1016/S0377-0427(00)00433-7

Raja, M.A.Z., 2014a. Numerical treatment for boundary value problems of pantograph functional differential equation using computational intelligence algorithms. *Appl. Soft Comput.*, **24**:806-821.
http://dx.doi.org/10.1016/j.asoc.2014.08.055

Raja, M.A.Z., 2014b. Solution of the one-dimensional Bratu equation arising in the fuel ignition model using ANN optimised with PSO and SQP. *Connect. Sci.*, **26**(3):195-214. http://dx.doi.org/10.1080/09540091.2014.907555

Raja, M.A.Z., 2014c. Stochastic numerical techniques for solving Troesch's problem. *Inform. Sci.*, **279**:860-873.
http://dx.doi.org/10.1016/j.ins.2014.04.036

Raja, M.A.Z., 2014d. Unsupervised neural networks for solving Troesch's problem. *Chin. Phys. B*, **23**(1):018903.

Raja, M.A.Z., Ahmad, S.I., 2014. Numerical treatment for solving one-dimensional Bratu problem using neural networks. *Neur. Comput. Appl.*, **24**(3):549-561.
http://dx.doi.org/10.1007/s00521-012-1261-2

Raja, M.A.Z., Samar, R., 2014a. Numerical treatment for nonlinear MHD Jeffery–Hamel problem using neural networks optimized with interior point algorithm. *Neurocomputing*, **124**:178-193.
http://dx.doi.org/10.1016/j.neucom.2013.07.013

Raja, M.A.Z., Samar, R., 2014b. Numerical treatment of nonlinear MHD Jeffery–Hamel problems using stochastic algorithms. *Comput. Fluids*, **91**:28-46.
http://dx.doi.org/10.1016/j.compfluid.2013.12.005

Raja, M.A.Z., Khan, J.A., Qureshi, I.M., 2010a. Evolutionary computational intelligence in solving the fractional differential equations. Asian Conf. on Intelligent Information and Database Systems, p.231-240.
http://dx.doi.org/10.1007/978-3-642-12145-6_24

Raja, M.A.Z., Khan, J.A., Qureshi, I.M., 2010b. Heuristic computational approach using swarm intelligence in solving fractional differential equations. Proc. 12th Annual Conf. Companion on Genetic and Evolutionary Computation, p.2023-2026.
http://dx.doi.org/10.1145/1830761.1830850

Raja, M.A.Z., Khan, J.A., Qureshi, I.M., 2010c. A new stochastic approach for solution of Riccati differential equation of fractional order. *Ann. Math. Artif. Intell.*, **60**(3):229-250.
http://dx.doi.org/10.1007/s10472-010-9222-x

Raja, M.A.Z., Khan, J.A., Qureshi, I.M., 2011a. Solution of fractional order system of Bagley-Torvik equation using evolutionary computational intelligence. *Math. Prob. Eng.*, **2011**:765075.
http://dx.doi.org/10.1155/2011/675075

Raja, M.A.Z., Khan, J.A., Qureshi, I.M., 2011b. Swarm intelligence optimized neural network for solving fractional order systems of Bagley-Tervik equation. *Eng. Intell. Syst.*, **19**(1):41-51.

Raja, M.A.Z., Khan, J.A., Ahmad, S.I., *et al.*, 2012. A new stochastic technique for Painlevé equation-I using neural network optimized with swarm intelligence. *Comput. Intell. Neur.*, **2012**:721867.
http://dx.doi.org/10.1155/2012/721867

Raja, M.A.Z., Ahmad, S.I., Samar, R., 2013. Neural network optimized with evolutionary computing technique for solving the 2-dimensional Bratu problem. *Neur. Comput. Appl.*, **23**(7):2199-2210.
http://dx.doi.org/10.1007/s00521-012-1170-4

Raja, M.A.Z., Samar, R., Rashidi, M.M., 2014a. Application of three unsupervised neural network models to singular nonlinear BVP of transformed 2D Bratu equation. *Neur. Comput. Appl.*, **25**(7):1585-1601.
http://dx.doi.org/10.1007/s00521-014-1641-x

Raja, M.A.Z., Ahmad, S.I., Samar, R., 2014b. Solution of the 2-dimensional Bratu problem using neural network, swarm intelligence and sequential quadratic programming. *Neur. Comput. Appl.*, **25**(7):1723-1739.
http://dx.doi.org/10.1007/s00521-014-1664-3

Raja, M.A.Z., Khan, J.A., Shah, S.M., *et al.*, 2015a. Comparison of three unsupervised neural network models for first Painlevé transcendent. *Neur. Comput. Appl.*, **26**(5):1055-1071. http://dx.doi.org/10.1007/s00521-014-1774-y

Raja, M.A.Z., Sabir, Z., Mahmood, N., *et al.*, 2015b. Design of stochastic solvers based on genetic algorithms for solving nonlinear equations. *Neur. Comput. Appl.*, **26**(1):1-23.
http://dx.doi.org/10.1007/s00521-014-1676-z

Raja, M.A.Z., Manzar, M.A., Samar, R., 2015c. An efficient computational intelligence approach for solving fractional order Riccati equations using ANN and SQP. *Appl. Math. Model.*, **39**(10-11):3075-3093.
http://dx.doi.org/10.1016/j.apm.2014.11.024

Raja, M.A.Z., Khan, J.A., Behloul, D., *et al.*, 2015d. Exactly satisfying initial conditions neural network models for numerical treatment of first Painlevé equation. *Appl. Soft Comput.*, **26**:244-256.
http://dx.doi.org/10.1016/j.asoc.2014.10.009

Raja, M.A.Z., Khan, J.A., Haroon, T., 2015e. Numerical treatment for thin film flow of third grade fluid using unsupervised neural networks. *J. Taiw. Inst. Chem. Eng.*, **48**:26-39. http://dx.doi.org/10.1016/j.jtice.2014.10.018

Saadatmandi, A., Dehghan, M., 2009. Variational iteration method for solving a generalized pantograph equation. *Comput. Math. Appl.*, **58**(11-12):2190-2196. http://dx.doi.org/10.1016/j.camwa.2009.03.017

Sedaghat, S., Ordokhani, Y., Dehghan, M., 2012. Numerical solution of the delay differential equations of pantograph type via Chebyshev polynomials. *Commun. Nonl. Sci. Numer. Simul.*, **17**(12):4815-4830. http://dx.doi.org/10.1016/j.cnsns.2012.05.009

Shakeri, F., Dehghan, M., 2010. Application of the decomposition method of Adomian for solving the pantograph equation of order *m*. *J. Phys. Sci.*, **65**(5):453-460. http://dx.doi.org/10.1515/zna-2010-0510

Srinivasan, S., Saghir, M.Z., 2014. Predicting thermodiffusion in an arbitrary binary liquid hydrocarbon mixtures using artificial neural networks. *Neur. Comput. Appl.*, **25**(5): 1193-1203. http://dx.doi.org/10.1007/s00521-014-1603-3

Tang, L., Ying, G., Liu, Y.J., 2014. Adaptive near optimal neural control for a class of discrete-time chaotic system. *Neur. Comput. Appl.*, **25**(5):1111-1117. http://dx.doi.org/10.1007/s00521-014-1595-z

Tohidi, E., Bhrawy, A.H., Erfani, K.A., 2013. A collocation method based on Berneoulli operational matrix for numerical solution of generalized pantograph equation. *Appl. Math. Model*, **37**(6):4283-4294. http://dx.doi.org/10.1016/j.apm.2012.09.032

Troiano, L., Cosimo, B., 2014. Genetic algorithms supporting generative design of user interfaces: examples. *Inform. Sci.*, **259**:433-451. http://dx.doi.org/10.1016/j.ins.2012.01.006

Uysal, A., Raif, B., 2013. Real-time condition monitoring and fault diagnosis in switched reluctance motors with Kohonen neural network. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **14**(12):941-952. http://dx.doi.org/10.1631/jzus.C1300085

Wright, S.J., 1997. Primal-Dual Interior-Point Methods. SIAM, Philadelphia, USA.

Xu, D.Y., Yang, S.L., Liu, R.P., 2013. A mixture of HMM, GA, and Elman network for load prediction in cloud-oriented data centers. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **14**(11):845-858. http://dx.doi.org/10.1631/jzus.C1300109

Yusufoğlu, E., 2010. An efficient algorithm for solving generalized pantograph equations with linear functional argument. *Appl. Math. Comput.*, **217**(7):3591-3595. http://dx.doi.org/10.1016/j.amc.2010.09.005

Yüzbaşı, Ş., Mehmet, S., 2013. An exponential approximation for solutions of generalized pantograph-delay differential equations. *Appl. Math. Model.*, **37**(22):9160-9173. http://dx.doi.org/10.1016/j.apm.2013.04.028

Yüzbaşı, Ş., Sahin, N., Sezer, M., 2011. A Bessel collocation method for numerical solution of generalized pantograph equations. *Numer. Meth. Part. Diff. Eq.*, **28**(4):1105-1123. http://dx.doi.org/10.1002/num.20660

Zhang, H.G., Wang, Z., Liu, D., 2008. Global asymptotic stability of recurrent neural networks with multiple time-varying delays. *IEEE Trans. Neur. Netw.*, **19**(5): 855-873. http://dx.doi.org/10.1109/TNN.2007.912319

Zhang, Y.T., Liu, C.Y., Wei, S.S., *et al.*, 2014. ECG quality assessment based on a kernel support vector machine and genetic algorithm with a feature matrix. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **15**(7):564-573. http://dx.doi.org/10.1631/jzus.C1300264

Zoveidavianpoor, M., 2014. A comparative study of artificial neural network and adaptive neurofuzzy inference system for prediction of compressional wave velocity. *Neur. Comput. Appl.*, **25**(5):1169-1176. http://dx.doi.org/10.1007/s00521-014-1604-2

Appendix: Reproduction of the results

The proposed solution determined by GA–IPT for examples 1, 2, and 3 of problem I are given in Eqs. (A1), (A3), and (A5) for the type 1 ANN model, respectively; in case of the type 2 ANN model, these solutions are given in Eqs. (A2), (A4), and (A6), respectively. In all these solutions, the values of weights are given to 15 decimal places to reproduce the results without rounding-off errors. Similarly, the proposed solution obtained by GA–IPT for types 1 and 2 of the ANN model are given in Eqs. (A7) and (A8), respectively, in case of problem II, while the respective solutions derived in case of problem III are given in Eqs. (A9) and (A10).

$$\begin{aligned}\tilde{f}(t)=\mathrm{e}^t+t^2\Bigg(&\frac{-0.0805}{1+\mathrm{e}^{-(0.1587t-1.3348)}}+\frac{0.3143}{1+\mathrm{e}^{-(0.4129t-1.0102)}}+\frac{-0.1894}{1+\mathrm{e}^{-(0.9333t-0.4304)}}+\frac{-0.3386}{1+\mathrm{e}^{-(0.0024t-0.2654)}}\\&+\frac{-0.1867}{1+\mathrm{e}^{-(0.7281t-0.4827)}}+\frac{-0.2082}{1+\mathrm{e}^{-(0.0807t-0.1575)}}+\frac{0.2900}{1+\mathrm{e}^{-(0.7426t-0.3297)}}+\frac{-0.0108}{1+\mathrm{e}^{-(-0.0655t-0.2871)}}\\&+\frac{0.0802}{1+\mathrm{e}^{-(0.9054t-0.4487)}}+\frac{0.3010}{1+\mathrm{e}^{-(-0.1944t-0.1444)}}\Bigg).\end{aligned}\tag{A1}$$

$$\begin{aligned}\hat{f}(t)=&\frac{-0.0805}{1+\mathrm{e}^{-(0.1587t-1.3348)}}+\frac{0.3143}{1+\mathrm{e}^{-(0.4129t-1.0102)}}+\frac{-0.1894}{1+\mathrm{e}^{-(0.9333t-0.4304)}}+\frac{-0.3386}{1+\mathrm{e}^{-(0.0024t-0.2654)}}\\&+\frac{-0.1867}{1+\mathrm{e}^{-(0.7281t-0.4827)}}+\frac{-0.2082}{1+\mathrm{e}^{-(0.0807t-0.1575)}}+\frac{0.2900}{1+\mathrm{e}^{-(0.7426t-0.3297)}}+\frac{-0.0108}{1+\mathrm{e}^{-(-0.0655t-0.2871)}}\\&+\frac{0.0802}{1+\mathrm{e}^{-(0.9054t-0.4487)}}+\frac{0.3010}{1+\mathrm{e}^{-(-0.1944t-0.1444)}}.\end{aligned}\tag{A2}$$

$$\begin{aligned}\tilde{f}(t)=\sin t+t^2\Bigg(&\frac{-0.9497}{1+\mathrm{e}^{-(0.3660\mathrm{t}-0.9103)}}+\frac{-1.3471}{1+\mathrm{e}^{-(1.0241\mathrm{t}-0.7515)}}+\frac{1.5038}{1+\mathrm{e}^{-(0.7086\mathrm{t}-1.6240)}}+\frac{-0.0356}{1+\mathrm{e}^{-(1.3745\mathrm{t}-0.1847)}}\\&+\frac{-0.4700}{1+\mathrm{e}^{-(-1.0890\mathrm{t}-0.2061)}}+\frac{-0.2331}{1+\mathrm{e}^{-(-0.3844\mathrm{t}-0.6442)}}+\frac{-0.0952}{1+\mathrm{e}^{-(-1.4891\mathrm{t}-0.57657)}}+\frac{-0.1511}{1+\mathrm{e}^{-(-1.7607\mathrm{t}-0.9300)}}\\&+\frac{-1.0915}{1+\mathrm{e}^{-(0.6036\mathrm{t}-0.2050)}}+\frac{-2.0344}{1+\mathrm{e}^{-(0.3911\mathrm{t}-0.8487)}}\Bigg).\end{aligned}\tag{A3}$$

$$\begin{aligned}\hat{f}(t)=&\frac{0.3888}{1+\mathrm{e}^{-(1.1142t-2.2556)}}+\frac{-1.3192}{1+\mathrm{e}^{-(1.4621t-2.0752)}}+\frac{1.2245}{1+\mathrm{e}^{-(1.6264t-0.3115)}}+\frac{-0.9506}{1+\mathrm{e}^{-(-1.03897t-1.2599)}}\\&+\frac{1.2311}{1+\mathrm{e}^{-(1.9669t-1.9453)}}+\frac{-1.1212}{1+\mathrm{e}^{-(0.8882t-3.2593)}}+\frac{-0.9785}{1+\mathrm{e}^{-(0.7765t-0.9891)}}+\frac{-0.0031}{1+\mathrm{e}^{-(1.9170t-0.4591)}}\\&+\frac{-0.1831}{1+\mathrm{e}^{-(2.5443t-0.8153)}}+\frac{1.2424}{1+\mathrm{e}^{-(1.6714t-0.0094)}}.\end{aligned}\tag{A4}$$

$$\begin{aligned}\tilde{f}(t)=\mathrm{e}^{-t}+t^2\Bigg(&\frac{-1.1254}{1+\mathrm{e}^{-(-0.8641t-1.9351)}}+\frac{1.42107}{1+\mathrm{e}^{-(-1.1904t-0.3160)}}+\frac{-0.7202}{1+\mathrm{e}^{-(-0.2773t-0.1214)}}+\frac{-0.5392}{1+\mathrm{e}^{-(0.9870t-1.1302)}}\\&+\frac{1.7714}{1+\mathrm{e}^{-(0.8936t-0.6601)}}+\frac{-0.7534}{1+\mathrm{e}^{-(0.0060t-0.0656)}}+\frac{-1.2488}{1+\mathrm{e}^{-(0.5566t-1.6376)}}+\frac{0.4800}{1+\mathrm{e}^{-(0.0252t-1.6338)}}\\&+\frac{-0.8304}{1+\mathrm{e}^{-(-1.5403t-1.4392)}}+\frac{1.6997}{1+\mathrm{e}^{-(-0.7958t-0.4921)}}\Bigg).\end{aligned}\tag{A5}$$

$$\begin{aligned}\hat{f}(t)=&\frac{0.0394}{1+\mathrm{e}^{-(-3.5121t-1.0535)}}+\frac{3.6193}{1+\mathrm{e}^{-(0.0334t-2.4532)}}+\frac{-0.5868}{1+\mathrm{e}^{-(-0.9518t-0.6043)}}+\frac{-2.7177}{1+\mathrm{e}^{-(2.1911t-1.7110)}}\\&+\frac{0.9885}{1+\mathrm{e}^{-(-0.3529t-0.4800)}}+\frac{-1.7338}{1+\mathrm{e}^{-(0.8104t-4.7231)}}+\frac{0.1042}{1+\mathrm{e}^{-(-0.8925t-0.2514)}}+\frac{-0.4658}{1+\mathrm{e}^{-(1.5842t-1.0946)}}\\&+\frac{-1.7886}{1+\mathrm{e}^{-(0.0902t-2.5350)}}+\frac{0.1960}{1+\mathrm{e}^{-(-1.8924t-1.0802)}}.\end{aligned}\tag{A6}$$

$$\begin{aligned}\tilde{f}(t)=t^2+t^2\Bigg(&\frac{0.6748}{1+\mathrm{e}^{-(0.7243t-0.0174)}}+\frac{0.5450}{1+\mathrm{e}^{-(-0.4061t-0.3664)}}+\frac{-0.4904}{1+\mathrm{e}^{-(-0.2773t-0.5386)}}+\frac{0.0373}{1+\mathrm{e}^{-(-0.7470t-0.8473)}}\\&+\frac{-0.2314}{1+\mathrm{e}^{-(0.7243t-0.0481)}}+\frac{0.1861}{1+\mathrm{e}^{-(0.7243t-1.1941)}}+\frac{-0.9102}{1+\mathrm{e}^{-(0.6773t-0.2389)}}+\frac{0.0467}{1+\mathrm{e}^{-(0.6888t-0.3597)}}\\&+\frac{0.9887}{1+\mathrm{e}^{-(-0.0828t-0.2227)}}+\frac{1.0437}{1+\mathrm{e}^{-(0.3359t-0.4376)}}\Bigg).\end{aligned}\tag{A7}$$

$$\begin{aligned}\hat{f}(t)=&\frac{2.3577}{1+\mathrm{e}^{-(1.4204t-1.5205)}}+\frac{2.0153}{1+\mathrm{e}^{-(1.0376t-1.2197)}}+\frac{-1.7392}{1+\mathrm{e}^{-(2.7949t-3.3269)}}+\frac{-1.5843}{1+\mathrm{e}^{-(1.5080t-0.0606)}}\\&+\frac{3.4531}{1+\mathrm{e}^{-(0.2616t-1.2651)}}+\frac{1.4275}{1+\mathrm{e}^{-(1.4536t-3.6760)}}+\frac{-1.4326}{1+\mathrm{e}^{-(2.1885t-1.5155)}}+\frac{0.4409}{1+\mathrm{e}^{-(0.8949t-3.9209)}}\\&+\frac{4.9585}{1+\mathrm{e}^{-(1.8724t-4.1948)}}+\frac{-0.7523}{1+\mathrm{e}^{-(-0.1993t-2.2552)}}.\end{aligned}\tag{A8}$$

$$\begin{aligned}\tilde{f}(t)=\sin t+t^2\Bigg(&\frac{0.3859}{1+\mathrm{e}^{-(-0.9243t-1.0047)}}+\frac{0.6179}{1+\mathrm{e}^{-(-0.2238t-0.1589)}}+\frac{-0.4950}{1+\mathrm{e}^{-(0.3636t-0.6232)}}+\frac{0.0061}{1+\mathrm{e}^{-(0.7295t-0.6232)}}\\&+\frac{-0.6360}{1+\mathrm{e}^{-(-0.5585t-0.0037)}}+\frac{0.1211}{1+\mathrm{e}^{-(0.1745t-0.3059)}}+\frac{0.3131}{1+\mathrm{e}^{-(0.2312t-0.1747)}}+\frac{-0.3690}{1+\mathrm{e}^{-(-0.9057t-0.9262)}}\\&+\frac{0.1313}{1+\mathrm{e}^{-(-0.8041t-0.0462)}}+\frac{-0.0395}{1+\mathrm{e}^{-(0.9470t-0.1381)}}\Bigg).\end{aligned}\tag{A9}$$

$$\begin{aligned}\hat{f}(t)=&\frac{0.3859}{1+\mathrm{e}^{-(-0.9243t-1.0047)}}+\frac{0.6179}{1+\mathrm{e}^{-(-0.2238t-0.1589)}}+\frac{-0.4950}{1+\mathrm{e}^{-(0.3636t-0.6232)}}+\frac{0.0061}{1+\mathrm{e}^{-(0.7295t-0.6232)}}\\&+\frac{-0.6360}{1+\mathrm{e}^{-(-0.5585t-0.0037)}}+\frac{0.1211}{1+\mathrm{e}^{-(0.1745t-0.3059)}}+\frac{0.3131}{1+\mathrm{e}^{-(0.2312t-0.1747)}}+\frac{-0.3690}{1+\mathrm{e}^{-(-0.9057t-0.9262)}}\\&+\frac{0.1313}{1+\mathrm{e}^{-(-0.8041t-0.0462)}}+\frac{-0.0395}{1+\mathrm{e}^{-(0.9470t-0.1381)}}.\end{aligned}\tag{A10}$$