Neuro-optimized numerical solution of non-linear problem based on Flierl–Petviashvili equation

Anas Bilal¹ · Guangmin Sun¹

Received: 27 February 2020 / Accepted: 27 May 2020 / Published online: 4 June 2020
© Springer Nature Switzerland AG 2020

Abstract

The article presents, a reliable numerical framework supported by feed-forward Artificial Neural Network (ANN) optimized with hybrid swarm intelligence technique for non-linear system of Flierl–Petviashvili (FP) problem. The universal approximation capabilities of ANN are exploited for mathematical approximation of the system in an unsupervised way based upon various performance metrics like fitness value, absolute error and execution time. The optimization of the cost function is subject to finding the appropriate weights which are highly stochastic in nature for the problem as well as its initial and boundary conditions. Therefore, hybrid approach based on Particle Swarm Optimization (PSO) and Interior Point Algorithm (IPA) is exploited for tuning of the adaptive weights in such a way that exploration is performed by PSO while the exploitation is done using IPA algorithm. The designed scheme is evaluated for standard FP problem along with its variants supported on various scenarios. The reliability, accuracy and robustness of the solvers are validated through a statistical analysis applied on two hundred independent runs.

Keywords Flierl–Petviashvili (FP) problem · Swarm intelligence · Interior point algorithm · Memetic computing

1 Introduction

The numerical framework based on derivative free methods has got much intention of the researchers and engineers in the recent decade to handle non-linear problems of the stiff nature [1, 2]. The derivative free methods can provide the solution for an unbounded domain as well on the continuous grid of time unlike the other classical numerical methods [3]. The nonlinear systems based on non-linear differential equations have various applications in applied physics, computational mathematics and engineering [4, 5]. A special class of problems like isothermal gas sphere, structure of earth and thermionic current can be distinctively formulated by various types of the equations of the Emden–Fowler type whose generalized expression is defined as [6]:

$$y''(t) + \frac{2}{t}y'(t) + \Psi u(t)\delta(y) = 0, \quad y(0) = y_0, y'(0) = 0, \quad (1)$$

A number of variants of Eq. (1) has been investigated by various researchers and applied in many applications using different formulations of $\delta(y)$ and $u(t)$ depending upon the nature and attribute of the problems [7, 8].

The Flierl–Petviashvili (FP) monopole belongs to a special class of singular non-linear Lane Emden Fowler equations as provided in Eq. (1) and written as [9, 10]:

$$y''(t) + \frac{r}{t}y'(t) - y^n - y^{n+1} = 0, \quad y(0) = \beta, y'(0) = 0, \quad (2)$$

where n is a non-linear operator. Eq. (2) can be deduced by setting the functions $u(t)$ and $\delta(y)$ and the parameter Ψ as $u(t) = 1$; $\Psi = -1$ and $\delta(y) = y^n + y^{n+1}$ in Eq. (1). A well known system of FP provided in Eq. (2) was thoroughly examined by Boyd [11] and Wazwaz [12]. They presented their results

✉ Guangmin Sun, gmsun@bjut.edu.cn | ¹Faculty of Information Technology, Beijing University of Technology, Chaoyang District, Beijing 100124, China.

in close form solution in a bounded domain without examining the computational complexity of the notorious system. Homotopy perturbation and Pade approximation has been exploited to address the first order FP problem in term of transform variable but the grid points are limited and computational budget has not been observed [13].

The need of the numerical methods arises due to fact that analytical methods provide tedious steps for complex nonlinear systems in a small definite domain and not necessarily provide a compact series solution. In this regards a fast and robust numerical approach based on shooting method has been applied for process flowsheet synthesis [14], that can handle the large domain and continuous grid point solution unlike the analytic methods. A numerical approach hybrid with Taguchi method has been presented in [15] for forgoing optimization process. These processes has infinite input continuous domains with nonlinearities, which forms it complex for analytical approaches to cope with. Similarly, some other numerical approaches are also presented [16–18] to provide the importance of numerical methods and its dominance.

The motivation for exploiting particle swarm optimization (PSO) hybrid with interior point algorithm (IPA) is due to the nature of the problem which is singular, nonlinear, dynamical and need to be address in the continuous domain of time. In this regards the methods like PSO has this capability of finding the solutions from an infinite space in a very small finite time. Moreover, the fine-tuning of the unknown parameter can be performed via local search technique like IPA that is based on langragian multipliers.

In this presented work the authors aim is to develop an alternate framework based on intelligent computing paradigm to handle numerical approximation of FP system effectively and efficiently.

Two commonly used variants of FP problem taken in this study are given as:

$$y''(t) + \frac{1}{t} y'(t) - y^n - y^{n+1} = 0, \\ y(0) = \beta, y'(0) = 0, \quad r = 1, n \geq 1, \quad (3)$$

$$y''(t) + \frac{r}{t} y(t) - y - y^2 = 0, \quad y(0) = \beta, y'(0) = 0, \quad n = 1, r \geq 1, \quad (4)$$

The special interest of research community is due to the singularity at origin $t = 0$ that's why very few analytic techniques are available to handle such problems. Although lots of classical approaches [19] has been applied to handle the above system on the discrete grid of time in a bounded domain but exploitation of

ANN may provide the solution on a continuous grid of time that motivates the author to handle such notorious problem. The extension of such continuous models can also be applied various applications of science, engineering and biotechnology as well [20–22].

The analysis depends first on employing an accurate feed-forward ANN mathematical model having strong approximation capabilities for FP problems. In this memetic computing, we aim to apply the ANN optimized with global search technique based on an intelligent method of swarm intelligence family PSO in a chaotic mechanism to get the unknowns of the ANN in an elegant way. These weights would be given as an input to the efficient local search technique based on IPA to get better convergence in a sufficiently large domain. This makes the use of an alternate approach like computational intelligence very essential to tackle problems of singularity along with non-linearity in an unbounded domain.

In what follows, the authors present the proposed methodology based on neural network mathematical modeling, creation of the fitness evaluation and the logical steps of the algorithm. In the last, the results of the simulations are presented under proposed framework.

2 Proposed methodology

This section describes details regarding the mathematical aspects of the proposed framework along with the creation of cost function. The training based on intelligent computational approach for the optimum weights of the model is also narrated.

2.1 Artificial neural network mathematical modeling

The attributes like continuity of the function, infinite domain, specified range and asymptotically no increasing and non decreasing make ANN a powerful universal approximator [23–25]. The structural flexibility can manage even a large number of training data set on continuous grid of time unlike other classical stat of art numerical techniques [26]. The algebraic sum of log-sigmoid function in the multiple hidden layers and input output units can model a nonlinear system with specified initial and boundary conditions [19, 27]. The approximate mathematical modeling of the system provided in Eq. (1) is formulated as follows:

$$\begin{cases} \hat{y}(t) = \sum_{i=1}^m a_i \frac{1}{1+\exp^{-(w_i t + b_i)}} \\ \frac{d\hat{y}(t)}{dt} = \sum_{i=1}^m a_i w_i \frac{\exp^{-(w_i t + b_i)}}{1+\exp^{-(w_i t + b_i)^2}} \\ \hat{y}''(t) = \sum_{i=1}^m a_i w_i^2 \left(\frac{2 \exp^{-2(w_i t + b_i)}}{(1+\exp^{-(w_i t + b_i)})^3} - \frac{\exp^{-(w_i t + b_i)}}{(1+\exp^{-(w_i t + b_i)})^2} \right) \end{cases} \quad (5)$$

where a_i , w_i , and b_i are bounded unknown of the solution $y(t)$, m is the number of neurons and log-sigmoid activation function is exploited for solution of DE. The differential equation neural networks (DENN) architecture is shown in Fig. 1 for system provided in Eqs. (3) and (4).

The cost function ε as unsupervised error is formed as:

$$\varepsilon = \varepsilon_1 + \varepsilon_2 \quad (6)$$

The error ε_1 of the DE is formulated as

$$\begin{cases} \varepsilon_1 = \frac{1}{N+1} \sum_{m=0}^N \left(\frac{d^2 \hat{y}_m}{dt^2} + \frac{r}{t} \frac{d\hat{y}_m}{dt} - \hat{y}_m^n - \hat{y}_m^{n+1} \right)^2, \quad t \in (0, 1) \\ N = 1/h, \quad \hat{y}_m = \hat{y}(t_m), \quad t_m = mh \end{cases} \quad (7)$$

Here $t \in (0, 1)$, $\hat{y}, t \in (0, 1)$ and $\hat{d}^2 y / dt^2$ are the models as given by the set of the equations in Eq. (5), N is the mesh interval at $t \in (x_0 = 0, x_1, x_2, \dots, x_N = 1)$ with a step size of h .

Similarly, the error ε_2 due to the initial conditions can be written as:

$$\varepsilon_2 = \frac{1}{2} \left\{ (\hat{y}(0) - \alpha)^2 + \left(\frac{d}{dt} \hat{y}(0) \right)^2 \right\} \quad (8)$$

As the value of the fitness evaluation function ε approaches zero or upto a desired level of fitness then the solution will satisfy both the ODE as well as its conditions. The fitness of the cost function is subject to the tuning and getting the optimal weights i.e. a_i , w_i , and b_i .

2.2 Learning procedure

Once the DENN architecture has been formulated, the process of finding the weights is performed via memetic computing in which partial swarm optimization (PSO) hybrid with interior point algorithm (IPA) is called as PSO-IPA.

The PSO was introduced by Eberhart and Kennedy in 1995 that fall under the umbrella of swarm intelligence [28]. Due to the stochastic nature of particles, PSO proves itself an efficient optimization algorithm for enormous applied problems of engineering and technology [29, 30]. The reliability and convergence of PSO algorithm can be guarantee by performing Monte Carlo simulations. The computational complexity in term of time of this algorithm is quite less compared with existed heuristic approached due to its implementation and simplicity of concept [31, 32].

The social behavior of the birds to get best food resource is formulated in the form mathematics to form an efficient global searching algorithm [33] that has less

Fig. 1 DENN architecture for standard FP system

probability to get stuck in the local optima. The agents in a particle represent the adaptive variables of the cost function that are randomly initialized and well scattered in a hyper dimensional space, moreover the velocity and positions are updated using the following relationships [34]:

$$\begin{cases} \mathbf{V}_i^j = w\mathbf{V}_i^{j-1} + c_1 \text{rand}_1() (\mathbf{LBP}_i^{j-1} - \mathbf{X}_i^{j-1}) + c_2 \text{rand}_2() (\mathbf{GBP}_i^{j-1} - \mathbf{X}_i^{j-1}), \\ \mathbf{X}_i^j = \mathbf{X}_i^{j-1} + \mathbf{V}_i^j, \end{cases} \quad (9)$$

where w is the linearly decreasing inertia factor, $\text{rand}_1()$ and $\text{rand}_2()$ are random function, j and $j-1$ are the current and previous visited flights, respectively. Three main components of the Eq. (9) i.e. $w\mathbf{V}_i^{j-1}$, $(\mathbf{LBP}_i^{j-1} - \mathbf{X}_i^{j-1})$ and $(\mathbf{GBP}_i^{j-1} - \mathbf{X}_i^{j-1})$ are current motion, influence of the local learning of the particles and collective intelligence of the swarm, respectively. Generally the more importance is provided to the collective intelligence factor in the initial stage of the algorithm and lesser to self intelligence while in the later stage of the flights the converse is applied. The local acceleration constant is c_1 while c_2 is the global acceleration constant. The proposed algorithm applied for weights training is of DENN network is provided below:

Step 1: A swarm of 240 particles each with a length equal to the number of a_i , w_i , and b_i is generated randomly.

Step 2: Initialize or set the parameters/values for specific and general settings as provided in Table 1 for PSO as a global optimizer and IPA on the later stage as local search scheme for fine tuning of the adaptive parameters.

Step 3: Compute the fitness value, error difference from the exact solution and budget for the cost functions provided in relations in Eqs. (6) to (8), respectively.

Step 4: The stopping criterion is formulated if any one of the condition satisfies:

- i. Mean Square Error (MSE) of 10^{-12} is achieved
- ii. Specified number of times the flights has been executed
- iii. Predefines number of function evaluation completed

In case the above mentioned criterion occurred then the unknown parameters would be passed as an input to IPA for fine tuning.

Step 5: The relation in Eq. (9) is used to update the velocity and position of the swarm keeping in view that the swarm should convergence or learn in every flight towards the better fitness value.

Step 6: The optimized parameters by the global optimizer are seeded to IPA to get even better tuning of the weights.

Step 7: A sufficient large number of independent runs are executed by repeating Step (2) to Step (6) to perform the statistical analysis that would provide the robustness and reliability of the proposed framework.

3 Simulation and results

A comprehensive scenario based simulations are provided in this section added with a number of graphs and table to show the applicability of the proposed methodology using PSO, IPA and PSO-IPA methods. The numerical results are compared with state of art Adomian Decomposition Method (ADM).

Table 1 Simulation settings for the proposed framework

PSO		IPA	
Attributes	Values	Attributes	Values
Population	200	Start point	One of the best solution from PSO
Particle length	30	No. of variable	30
Flights	2000	Iteration	1000
Local acceleration Factor	Stochastic (2.8 to 1.4)	Maximum Evaluations	50,000
Global acceleration Factor	Linear increasing (0.2 to 1.9)	Function tolerance	10^{-20}
Inertial weights	Adaptive	Constraints tolerance	10^{-20}
$V_{\max}$	04	Derivative method	Finite central difference
Population Span	(-20, 20)	X-Tolerance	10^{-20}
Velocity Span	(-4, 4)	Bounds	(-10, 10)
Fitness Limit	10^{-10}	Fitness Limit	10^{-15}

3.1 First generalization of FP problem

In the first generalization of FP problem, two cases have been considered, i.e., *Case 1*: $n = 1, r = 1$ and *Case 2*: $n = 3, r = 1$. This variant of the FP problem is given as:

$$\begin{cases} y''(t) + \frac{1}{t} y'(t) - y^n - y^{n+1} = 0, & 0 \leq t \leq 1, \quad n \geq 1 \\ y(0) = \beta, y'(0) = 0 \end{cases} \quad (10)$$

The difficulty associated with system in Eq. (10) is unavailability of the exact solution, so the series solution by ADM method [12] is taken as reference and is provided below for $n \geq 1$.

$$y(t) = \beta + \frac{\beta^n + \beta^{n+1}}{4} t^2 + \dots + \frac{(\beta^n + \beta^{n+1})(n(18n^2 - 29n + 12)\beta^{3n} + n(54n^2 - 33n + 7)\beta^{3n+1} + (18n^2 + 7n + 1)(3n\beta^{3n+2} + (n+1)\beta^{3n+3}))}{147456\beta^3} t^8 \quad (11)$$

The series solution in (11) is used to obtain the analytic result $y_{ADM}(t)$ of the given problem and the same problem is solved with proposed framework as well by taking $m = 10$ neurons in each hidden layer. The fitness evaluation function ε as given in Eq. 6 and its subsequent in Eqs. (7) and (8) for this variant is formulated as:

$$\begin{aligned} \hat{y}_1(t) = & \frac{0.840389828866272}{1 + \exp(-0.203293948447699t - 0.260728076235772)} + \frac{0.349966219563269}{1 + \exp(-0.209128661321806t - 0.609249468040922)} \\ & + \frac{0.249435916791058}{1 + \exp(-0.545172684670090t + 1.874548410059270)} + \frac{3.758806744424070}{1 + \exp(-5.270639605303650t - 7.448217094034860)} \\ & + \frac{-1.620499936645270}{1 + \exp(-4.305607941307900t - 3.364165240876380)} + \frac{-0.421952680503091}{1 + \exp(-0.243902692467155t - 3.158510006619070)} \\ & + \frac{2.153307775417280}{1 + \exp(-1.725097825881410t - 3.251141655924420)} + \frac{-1.367577517032180}{1 + \exp(-2.631579203438690t - 3.042465650211420)} \\ & + \frac{-0.100079665316700}{1 + \exp(-0.473845211060497t - 1.821811492986790)} + \frac{0.967402115920246}{1 + \exp(-1.963153505726950t - 1.765332546484340)} \end{aligned} \quad (13)$$

$$\varepsilon = \frac{1}{11} \sum_{m=0}^{10} \left(\hat{y}_m'' + \frac{1}{t} \hat{y}_m' - \hat{y}_m^n - \hat{y}_m^{n+1} \right)^2 + \frac{1}{2} \left\{ (\hat{y}(0) - \alpha)^2 + (\hat{y}'(0))^2 \right\}, \quad n \geq 1 \quad (12)$$

On an appropriate weights the value of cost function $\varepsilon \rightarrow 0$, this employ $\hat{y} \rightarrow y$, hence the solution is approximated. The optimization of fitness function provided in Eq. (12) carried out using PSO and IPA with parameter given in Table 1. The proposed solutions are obtained for PSO, IPA and PSO-IPA algorithms with $\varepsilon 4.0051 \times 10^{-04}$, 2.3914×10^{-09} , 1.1169×10^{-09} for case 1 and 1.2308×10^{-03} ,

1.2966×10^{-08} , 2.2664×10^{-08} for case 2 of FP problem provided in Eq. (10). The set of the tuned weights optimized by PSO-IPA upto ten decimal places are given in Table 2, and by using these weights in Eq. (5), the numerical solutions $\hat{y}_1$ and $\hat{y}_2$ are obtained for Case 1 and Case 2, respectively and given as:

Table 2 One of the best optimized weights using PSO-IPA Algorithm

Index	Case I: $n = 1, r = 1$			Case II: $n = 3, r = 1$		
	a_i	w_i	b_i	a_i	w_i	b_i
1	0.8403898288	-0.2032939484	-0.2607280762	7.0913542145	8.3366842673	-9.7391899849
2	0.3499662195	0.2091286613	-0.6092494680	-1.6461041469	1.3639779155	1.3641228503
3	0.2494359167	0.5451726846	1.8745484100	4.5865220574	-1.3120156437	-0.9985201706
4	3.7588067444	5.2706396053	-7.4482170940	2.9262084407	-0.4920754666	-0.5390468059
5	-1.6204999366	4.3056079413	-3.3641652408	-1.8843841855	2.5472362470	1.5837421266
6	-0.4219526805	-0.2439026924	-3.1585100066	-8.2102804459	0.5224971909	6.3751513866
7	2.1533077754	1.7250978258	-3.2511416559	6.7429673033	1.5427408146	-5.4403416008
8	-1.3675775170	2.6315792034	-3.0424656502	-3.5038530481	-4.2439825259	0.9313241751
9	-0.1000796653	-0.4738452110	-1.8218114929	-0.6824071387	0.8617849645	0.0338168368
10	0.9674021159	1.9631535057	-1.7653325464	-1.2381203527	1.2983268247	-0.7674427972

$$\begin{aligned}
\hat{y}_2(t) = & \frac{7.091354214575070}{1 + \exp^{-(8.336684267313230t - 9.739189984977530)}} + \frac{-1.646104146956380}{1 + \exp^{-(1.363977915514550t + 1.364122850345840)}} \\
& + \frac{4.586522057417810}{1 + \exp^{-(1.312015643769970t - 0.998520170664967)}} + \frac{2.926208440779080}{1 + \exp^{-(0.492075466696170t - 0.539046805919958)}} \\
& + \frac{-1.884384185558870}{1 + \exp^{-(2.547236247003520t + 1.583742126656310)}} + \frac{-8.210280445909560}{1 + \exp^{-(0.522497190958484t + 6.375151386676500)}} \\
& + \frac{6.742967303330400}{1 + \exp^{-(1.542740814682860t - 5.440341600803560)}} + \frac{-3.503853048130120}{1 + \exp^{-(4.243982525980660t + 0.931324175176035)}} \\
& + \frac{-0.682407138707965}{1 + e^{-(0.861784964512767t + 0.033816836850087)}} + \frac{-9.652272348741120}{1 + e^{-(1.298326824746130t - 0.767442797293186)}}
\end{aligned} \quad (14)$$

The numerical behavior of each case is obtained and results are provided in Fig. 2 on semi-log scale for inputs $t \in [0, 1]$ with a step of 0.1 for IPA and PSO-IPA. It has been observed from Fig. 2a, b that the proposed solutions are very much on the same track as the analytical method.

As it is merely difficult to see the difference between IPA and PSO-IPA so the results are also presented for cases 1 and 2 of the problem in Table 3. The mean absolute error (MAE) for PSO, IPA, and PSO-IPA, are 0.004828, 0.0002714 and 0.00000007054, respectively, for case 1, while for case 2 are

Fig. 2 Numerical behavior of IPA and PSO-IPA for both cases of First Variant

Table 3 Results of the proposed solution in comparison with ADM method

T	Case I: n = 1, r = 1					Case II: n = 3, r = 1				
	Numerical	Approximated				Numerical	Approximated			
	y_{ADM} [12]	y_{PSO}	y_{IPA}	$y_{PSO-IPA}$	y_{FFA}	y_{ADM} [12]	y_{PSO}	y_{IPA}	$y_{PSO-IPA}$	y_{FFA}
0.0	1.00000000	0.99997916	0.99999655	0.99999982	0.99564316	1.00000000	0.99270000	0.99903464	0.99947521	0.96743000
0.1	1.00500939	1.00485213	1.00493567	1.00500683	1.03427213	1.00502200	0.99710000	1.00399266	1.00460519	0.99875400
0.2	1.02015112	1.01975075	1.02001625	1.02014699	1.01874075	1.02035836	1.01100000	1.01918432	1.01997651	1.53432100
0.3	1.04577217	1.04519302	1.04561758	1.04576662	1.04536542	1.04686894	1.03570000	1.04547495	1.04642449	1.03545200
0.4	1.08247252	1.08177863	1.08229250	1.08245807	1.08175221	1.08616031	1.07300000	1.08443487	1.08560356	1.07302365
0.5	1.13113907	1.13033106	1.13089959	1.13109363	1.13036543	1.14088244	1.12490000	1.13861047	1.13998977	1.12490123
0.6	1.19299622	1.19201159	1.19264748	1.19287046	1.20000761	1.21521766	1.19510000	1.21196845	1.21365498	1.19510000
0.7	1.26967621	1.26839731	1.26912630	1.26937257	1.23265331	1.31563509	1.28950000	1.31087079	1.31301987	1.28325600
0.8	1.36331292	1.36154908	1.36238410	1.36265887	1.38754608	1.45200498	1.41800000	1.44611541	1.44895905	1.41806321
0.9	1.47666412	1.47412085	1.47507248	1.47538937	1.46212285	1.63918826	1.59790000	1.63684736	1.64079012	1.59791230
1.0	1.61326769	1.60954829	1.61064066	1.61100126	1.56843252	1.89923774	1.86320000	1.92166409	1.92771233	1.86321300

0.000009491, 0.0008072 and 0.0000002101, respectively. The approximation given by the proposed schemes are found to be in the close vicinity of the state of art existed numerical method. Table 3 also presents the results obtained with the standard firefly algorithm [35] as “ y_{FFA} ” for Case I and Case II respectively, it is very much evident from the table that the modern firefly method is capable enough to approximate such a difficult problem in a good range of approximation.

3.2 Second generalization of FP problem

In this subsection, discussion has been made on second generalized variant of the FP problem [11, 12].

$$\begin{cases} y''(t) + \frac{r}{t} y'(t) - y^1 - y^2 = 0, & 0 \leq t \leq 1, \quad r \geq 1 \\ y(0) = \beta, \quad y'(0) = 0 \end{cases} \quad (15)$$

The approximation reported in [12] for Eq. (10) is also taken into account by the proposed framework for $r \geq 1$ [12].

$$\begin{aligned} y_0 &= \beta, \\ y_1 &= \frac{\beta(\beta+1)}{2(r+1)} t^2, \\ y_2 &= \frac{\beta(\beta+1)(2\beta+1)}{8(r+1)(r+3)} t^4, \\ y_3 &= \frac{\beta(\beta+1)[2(3r+5)\beta(\beta+1) + (r+1)]}{48(r+1)^2(r+3)(r+5)} t^6, \\ y_4 &= \frac{\beta(\beta+1)(2\beta+1)[4(3r+10)\beta(\beta+1) + (r+1)]}{384(r+1)^2(r+3)(r+5)(r+7)} t^8, \end{aligned} \quad (16)$$

In view of (16), the solution in series form is thus given by [12]

$$\begin{aligned} y_{ADM}(t) &= \beta + \frac{\beta(\beta+1)}{2(r+1)} t^2 + \dots \dots \dots \\ &+ \frac{\beta(\beta+1)(2\beta+1)[4(3r+10)\beta(\beta+1) + (r+1)]}{384(r+1)^2(r+3)(r+5)(r+7)} t^8 \end{aligned} \quad (17)$$

The series solution in (17) is used to obtain the analytic result $y_{ADM}(t)$ of the problem provided in Eq. (15). The proposed DENN by taking the same number of neurons, unknown weights, parameter setting and grid points as discussed in the first generalized variant is also applied on Eq.(15). The fitness evaluation function ε used for this variant is provided as:

$$\begin{aligned} \varepsilon &= \frac{1}{11} \sum_{m=0}^{10} \left(\hat{y}_m'' + \frac{r}{t} \hat{y}_m' - \hat{y}_m - \hat{y}_m^2 \right)^2 \\ &+ \frac{1}{2} \left\{ (\hat{y}(0) - \beta)^2 + (\hat{y}'(0))^2 \right\}, \quad r \geq 1 \end{aligned} \quad (18)$$

The set of the tuned weights optimized by PSO-IPA upto fifteen decimal places are given in Table 4.

The numerical results in the form of series by hybrid PSO-IPA is represented as $\hat{y}_3$ and is provided below.

$$\begin{aligned} \hat{y}_3(t) &= \frac{-0.027240629414029}{1 + \exp(-(-1.666479715942230t - 2.535643168816240))} \\ &+ \frac{-1.333392133853010}{1 + \exp(-(-4.762169767821230t - 2.765778058320940))} \\ &+ \frac{0.908668667588864}{1 + \exp(-(-1.584381225915130t + 0.855683784546263))} \\ &+ \frac{1.670489196521410}{1 + \exp(-(-1.355555223164690t - 2.436376078470800))} \\ &+ \frac{-0.243661573062884}{1 + \exp(-(-0.671617130297807t + 1.577532113250530))} \\ &+ \frac{-0.394851154278195}{1 + \exp(-(-0.542657560397518t + 0.489392442404944))} \\ &+ \frac{-0.787011576613045}{1 + \exp(-(-0.647017154331779t - 0.960218472510105))} \\ &+ \frac{1.362995352053900}{1 + \exp(-(-0.502681713024216t - 7.709099205258910))} \\ &+ \frac{-1.302256373313450}{1 + \exp(-(-0.072226621372004t + 0.456401919962945))} \\ &+ \frac{2.613275251249140}{1 + \exp(-(-9.189030137352840t - 6.517000353711890))} \end{aligned} \quad (19)$$

Table 4 Best trained weights using PSO-IPA algorithm

Index	Variant with n = 1, r = 2		
	a_i	w_i	b_i
1	-0.027240629414029	-1.666479715942230	2.535643168816240
2	-1.333392133853010	4.762169767821230	-2.765778058320940
3	0.908668667588864	1.584381225915130	0.855683784546263
4	1.670489196521410	1.355555223164690	-2.436376078470800
5	-0.243661573062884	0.671617130297807	1.577532113250530
6	-0.394851154278195	0.542657560397518	0.489392442404944
7	-0.787011576613045	0.647017154331779	-0.960218472510105
8	1.362995352053900	-0.502681713024216	-7.709099205258910
9	-1.302256373313450	-0.072226621372004	0.456401919962945
10	2.613275251249140	9.189030137352840	-6.517000353711890

Fig. 3 Numerical behavior of IPA and PSO-IPA for Second Variant

Table 5 Approximate results and absolute error of proposed algorithms with numerical method

t	Numerical y_{ADM} [12]	Approximated			Absolute error		
		y_{PSO}	y_{IPA}	$y_{PSO-IPA}$	$ y_{ADM} - y_{PSO} $	$ y_{ADM} - y_{IPA} $	$ y_{ADM} - y_{PSOIPA} $
0.0	1.00000000	1.00000375	0.99999993	0.99999734	3.751E-6	7.001E-8	2.661E-6
0.1	1.00333834	1.00332448	1.00333564	1.00333417	1.392E-5	2.702E-6	4.172E-6
0.2	1.01341373	1.01339566	1.01340842	1.01340755	1.813E-5	5.313E-6	6.183E-6
0.3	1.03040958	1.03040189	1.03040397	1.03040268	7.694E-6	5.614E-6	6.904E-6
0.4	1.05463927	1.05463592	1.05463405	1.05463222	3.355E-6	5.225E-6	7.055E-6
0.5	1.08655829	1.08654816	1.08655306	1.08655109	1.016E-5	5.236E-6	7.206E-6
0.6	1.12678218	1.12676423	1.12677695	1.12677506	1.797E-5	5.237E-6	7.127E-6
0.7	1.17611137	1.17609487	1.17610767	1.17610572	1.658E-5	3.708E-6	5.658E-6
0.8	1.23556420	1.23555822	1.23556692	1.23556467	5.989E-6	2.729E-6	4.709E-7
0.9	1.30641973	1.30643281	1.30644305	1.30644052	1.317E-5	2.338E-5	2.088E-5
1.0	1.39027227	1.39034376	1.39035418	1.39035151	7.155E-5	8.197E-5	7.927E-5

The results of the stochastic solvers based on IPA and PSO-IPA are provided in Fig. 3. It is quite seen that the results based on IPA and PSO-IPA are almost consistent with the reference solution y_{ADM} .

The numerical and proposed scheme are seems to be so overlapped that difference between them is difficult to recognize. To have a close look on the results, the absolute error of the given techniques from numerical method is tabulated in Table 5.

4 Comparative study of the solvers

The comparative study of the solvers is examined by a statistical analysis of two hundred independent runs with the parameters, i.e., mean, standard deviation (STD) and minimum (MIN) values of absolute error (AE) from ADM numerical solution and the results are presented in Table 6.

The accuracy of the hybrid approach is superior than that of PSO and IPA alone, however the computational cost is slightly more that effect can be oversight with the fact that hybrid scheme show good reliability and convergence.

Furthermore the convergence of proposed framework have been further investigated by calculating the achieved value of fitness ϵ at two hundred runs of each solver. The results of fitness ϵ achieved are presented in Fig. 4a–c, respectively, for two cases of first variant and one case of second variant, respectively. The results of the Fig. 4 are drawn on semi logy scale in order to enlarge the small variations in the fitness values for all execution runs.

The results of fitness ϵ achieved are presented in Fig. 5a–c, respectively, for two cases of first variant and one case of second variant, respectively. Fig 5 presents the behavior of fitness achieved in descending order of its value attained for 200 run on semi-log scale. It is quite evident from the figure that PSO-IPA shows dominance

Table 6 A comprehensive analysis of FP problem and its variants

Scenarios	Method	Mode	AE for Inputs 't'				
			0.2	0.4	0.6	0.8	1.0
n1r1	PSO	MIN	1.71746101E-05	1.66717692E-05	3.91714725E-05	3.61087619E-05	4.50176656E-04
		Mean	2.99012262E-03	4.37016325E-03	5.49015644E-03	7.52004309E-03	1.11001280E-02
		STD	5.39011542E-03	7.38013209E-03	9.84000823E-03	1.27001768E-02	1.63002435E-02
	IPA	MIN	1.37606623E-07	1.34103585E-06	2.51685749E-06	4.89116004E-04	2.05005358E-03
		Mean	1.71746101E-05	1.66717692E-05	3.91714725E-05	3.61087619E-05	4.50176656E-04
		STD	2.99012262E-03	4.37016325E-03	5.49015644E-03	7.52004309E-03	1.11001280E-02
	PSO-IPA	MIN	5.39011542E-03	7.38013209E-03	9.84000823E-03	1.27001768E-02	1.63002435E-02
		Mean	1.37606623E-07	1.34103585E-06	2.51685749E-06	4.89116004E-04	2.05005358E-03
		STD	4.46670831E-05	6.48632993E-05	1.81078302E-04	7.22037417E-04	2.36002592E-03
n3r1	PSO	MIN	2.73103956E-05	3.74436425E-05	6.50300617E-05	8.30835755E-05	1.08026876E-04
		Mean	3.52504139E-07	8.96363528E-07	3.80603392E-05	5.41148810E-04	2.12001558E-03
		STD	4.50278042E-05	6.68447747E-05	1.91099867E-04	7.34046978E-04	2.37009912E-03
	IPA	MIN	2.49236989E-05	3.36419247E-05	5.28008954E-05	6.66809146E-05	8.67098514E-05
		Mean	3.64002004E-02	5.15000080E-02	7.56001760E-02	1.23000059E-01	2.12000090E-01
		STD	1.10000200E-01	1.44000137E-01	2.02000237E-01	3.13000225E-01	5.39000187E-01
	PSO-IPA	MIN	3.88001770E-02	4.95002161E-02	6.69001302E-02	9.68000659E-02	1.55000154E-01
		Mean	7.01036717E-04	1.12016395E-03	2.39007966E-03	4.47018755E-03	2.22189135E-04
		STD	5.41016160E-03	7.52004666E-03	1.17000259E-02	1.98000462E-02	2.20002285E-02
n1r2	PSO	MIN	6.97012706E-03	9.52009038E-03	1.38001497E-02	2.24000704E-02	3.95002383E-02
		Mean	1.83238379E-04	3.26112878E-04	1.19019081E-03	2.45002232E-03	6.21470469E-05
		STD	2.92015900E-03	4.10024050E-03	6.71010375E-03	1.17001412E-02	1.40000340E-02
	IPA	MIN	1.86019608E-03	2.55003832E-03	3.70002225E-03	6.15016742E-03	8.71017059E-03
		Mean	1.36111804E-05	3.55960309E-06	5.03528552E-06	6.12915313E-07	3.29229859E-05
		STD	1.17010594E-03	1.75015797E-03	1.84003765E-03	2.23010430E-03	2.53012872E-03
	PSO-IPA	MIN	1.51020220E-03	4.93009219E-03	5.29006885E-03	6.04015789E-03	7.08012993E-03
		Mean	6.24501046E-08	9.16763053E-08	1.55220859E-07	1.74366319E-07	7.08097925E-06
		STD	1.29326269E-05	1.28049220E-05	1.36291500E-05	1.77663591E-05	8.76187891E-05

on PSO and IPA in both cases of the first variant as well in the second variant.

The reliability of the solvers is further inspected on the basis of number of convergent runs and number of independent runs achieving a certain level of MAE whose results are tabulated in Table 7. It can be infer that even based on a stiff criteria, i.e., fitness $\epsilon \leq 10^{-06}$, average convergence rate for PSO, IPA and PSO-IPA are 0.03%, 100% and 100% for case I of variant 1, 0.00%, 4.00% and 14.00% for case II of variant 1 and 8.00%, 100% and 100% for variant 2, respectively. Similarly the mean absolute errors at different predefined levels are also presented in the table using the proposed scheme for both variants of FP problem.

Now the focus is on two other performance parameters, i.e., global mean absolute error (G_{MAE}) and mean fitness (M_F), given as:

$$G_{MAE} = \frac{1}{N} \sum_{k=1}^N \left(\frac{1}{M} \sum_{i=1}^M |y_i - \hat{y}_i^k| \right), \quad (20)$$

$$M_F = \frac{1}{R} \sum_{k=1}^R \epsilon_k$$

Here M is sample points, N represents independent flights settled for the solver, y_i is the solution using ADM for the i th input, $\hat{y}_i^k$ is the solution of the designed scheme for the i th input and k th independent run, and ϵ_k is the fitness value obtained for k th independent run. The inputs are taken from $t \in [0, 1]$ i.e., $M = 11$ and 200 runs i.e., $N = 200$ and the results are presented in Table 8 for each solvers on both of the variants. The respective values of M_F and G_{MAE} lie from 10^{-03} to 10^{-05} and 10^{-03} to 10^{-05} for PSO, 10^{-05} to 10^{-08} and 10^{-02} to 10^{-05} for IPA, 10^{-05} to 10^{-08} and 10^{-03} to 10^{-05} for PSO-IPA. Whereas, the standard deviation for M_F and G_{MAE} are of the order 10^{-03} to 10^{-04} and 10^{-01} to 10^{-02} for PSO, 10^{-04} to 10^{-08} and 10^{-02} to 10^{-05} for IPA, 10^{-05} to

Fig. 4 The values of fitness achieved in 200 independent runs of both variants using PSO, IPA and PSO-IPA

10^{-08} and 10^{-03} to 10^{-05} for PSO-IPA. It can be deduced that the values of G_{MAE} and M_F are found consistently accurate for NN tuned with PSO-IPA as compared with other two approaches based on PSO and IPA.

The computational cost of each solver is examined on a sufficient large number of runs and results are tabulated in Table 8 as mean execution time (M_{ET}) and standard deviation (STD). It is clearly evident that M_{ET} for hybrid approach PSO-IPA is relatively more as compared with PSO and IPA. However, this aspect can be overlooked on the basis of

preeminence in results by PSO-IPA in term of accuracy and convergence over other two techniques.

The statistical analysis is also taken by using 400 independent runs of methods like PSO, IPA and its hybrid approach on same performance metrics as taken in Table 8. The results of the analysis on 400-independent runs of the methods is tabulated in Table 9. It is quite evident from the table that mean execution time has been slightly increased; however mean fitness and global mean absolute error has been slightly reduced.

Fig. 5 Convergence behavior of fitness achieved in descending order for the proposed solvers

Table 7 Convergence analysis of results at various thresholds

Scenarios	Method	Number of runs with Fitness					Number of runs with MAE				
		10^{-04}	10^{-05}	10^{-06}	10^{-07}	10^{-08}	10^{-01}	10^{-02}	10^{-03}	10^{-04}	10^{-05}
n1r1	PSO	187	84	6	0	0	200	199	136	9	0
	IPA	200	200	200	200	91	200	200	200	131	0
	PSO-IPA	200	200	200	200	114	200	200	200	132	0
n3r1	PSO	2	0	0	0	0	200	0	0	0	0
	IPA	195	161	8	0	0	200	193	31	0	0
	PSO-IPA	200	187	28	0	0	200	200	75	0	0
n1r2	PSO	195	112	16	1	0	200	200	190	58	2
	IPA	200	200	200	197	162	200	200	200	200	196
	PSO-IPA	200	200	200	197	166	200	200	200	200	196

Table 8 Comparative study based on M_F , G_{MAE} and MET

Scenarios	Methods	M_F		G_{MAE}		$M_{ET}(\text{Sec})$	
		Values	STD	Values	STD	Values	STD
n1r1	PSO	1.88E-04	7.83E-04	5.17E-03	1.26E-02	135.6	31.7
	IPA	6.76E-08	5.76E-08	4.23E-05	7.23E-05	34.2	93.7
	PSO-IPA	5.51E-08	4.22E-08	1.85E-05	7.18E-05	179.4	87.1
n3r1	PSO	7.75E-03	5.17E-03	2.22E-01	1.59E-01	175.2	34.8
	IPA	6.38E-06	2.85E-04	1.16E-02	1.99E-02	39.0	36.0
	PSO-IPA	1.84E-06	1.20E-05	6.56E-03	6.17E-03	208.4	74.1
n1r2	PSO	8.61E-05	1.33E-04	1.71E-03	4.13E-03	130.4	1.3
	IPA	4.72E-08	1.42E-07	2.15E-05	4.04E-06	35.9	92.2
	PSO-IPA	4.41E-08	9.39E-08	2.67E-05	2.47E-06	157.6	7.3

Table 9 Comparative study based on M_F , G_{MAE} and MET

Scenarios	Methods	M_F		G_{MAE}		$M_{ET}(\text{Sec})$	
		Values	STD	Values	STD	Values	STD
n1r1	PSO	2.31E-04	3.98E-04	4.25E-03	2.17E-02	198.3	41.7
	IPA	5.63E-08	5.37E-08	3.67E-05	3.28E-05	54.31	53.7
	PSO-IPA	4.34E-08	4.68E-08	1.35E-05	6.48E-05	194.4	97.1
n3r1	PSO	7.34E-03	6.23E-03	1.89E-01	3.28E-01	198.4	54.8
	IPA	5.98E-06	3.87E-04	1.03E-02	3.84E-02	73.21	96.0
	PSO-IPA	2.09E-06	2.14E-05	4.89E-03	3.85E-03	292.1	44.1
n1r2	PSO	8.92E-05	4.38E-04	1.21E-03	3.45E-03	167.4	8.3
	IPA	4.86E-08	1.23E-07	1.35E-05	5.34E-06	56.43	83.5
	PSO-IPA	4.72E-08	3.48E-08	1.78E-05	3.44E-06	198.3	6.4

5 Conclusions

The following conclusions are made based on the material & methods and the numerical simulations carried out in the manuscript to prove the agenda:

- The proposed DENN architecture can provide the solution for different variants of Flierl–Petviashvili problem effectively on the continuous grid of time unlike the other analytical methods.
- The proposed methodology based on the artificial neural network, PSO and hybrid approach with IPA proves its applicability and effectiveness for non-linear complex FP systems.
- The comparison of the results of the proposed schemes is made with well known ADM and it is found that the given approximation is in a good agreement upto 5 to 7 decimal places.
- The analysis carried out for 200 runs on the performance criteria standard deviation of AE, GMAE and MF achieved are found to be dominated for PSO-IPA as compared with IPA and PSO algorithms. Generally, the relatively more accurate solutions are obtained with PSO-IPA for both of the variants of Flierl–Petviashvili equation.

- For a stiff criteria, i.e., fitness $\varepsilon \leq 10^{-06}$, average convergence rate for PSO, IPA and PSO-IPA are 0.03%, 100% and 100% for case I of variant 1, 0.00%, 4.00% and 14.00% for case II of variant 1 and 8.00%, 100% and 100% for variant 2, respectively.
- The complexity analysis in term of M_{ET} shows that PSO-IPA takes slightly more time for adaptation of weights of the networks than that of PSO and IPA, however, this can be overlooked by considering the fitness attained by hybrid approach over the other two techniques.

In future, the methods like artificial immune system, gray wolf approximation, chaotic programming etc., can be examined for not only solving FP Problem but also applied to other unsolved nonlinear system as well. The modified versions of numerical methods based on RK-45 should be developed and exploited for such rigorous problems.

Compliance with ethical standard

Conflict of interest The authors declare that they have no competing interests.

Appendix

Scenarios	Method	Mode	AE for Inputs 't'				
			0.1	0.3	0.5	0.7	0.9
n1r1	PSO	MIN	1.253431E-05	1.342568E-05	3.243547E-05	2.785436E-05	2.465737E-04
		Mean	2.569012E-03	2.364657E-03	5.897456E-03	2.465783E-03	2.465485E-03
		STD	4.352648E-03	2.465739E-03	3.576963E-03	1.874524E-02	2.465889E-02
	IPA	MIN	2.464535E-07	4.574637E-06	3.675539E-06	3.556372E-05	3.564423E-05
		Mean	3.434922E-05	3.334354E-05	7.834295E-05	7.221752E-05	9.003533E-04
		STD	5.980245E-03	8.740327E-03	1.098031E-02	1.504009E-02	2.220026E-02
	PSO-IPA	MIN	1.078023E-09	1.476026E-08	1.968002E-08	2.540035E-06	3.260049E-05
		Mean	2.752132E-07	2.682072E-06	5.033715E-06	9.782320E-04	4.100107E-03
		STD	7.901630E-05	9.820684E-05	2.594212E-04	7.942549E-04	3.260379E-03
n3r1	PSO	MIN	6.007556E-03	8.777770E-03	1.104534E-02	1.512317E-02	2.230828E-02
		Mean	1.078058E-02	1.476116E-02	1.971808E-02	2.594150E-02	3.472050E-02
		STD	4.530302E-05	6.952685E-05	1.961336E-04	1.712279E-03	6.470206E-03
	IPA	MIN	1.039400E-04	1.318488E-04	3.122221E-04	8.609359E-04	3.347089E-03
		Mean	4.240776E-02	6.027778E-02	8.664552E-02	1.381232E-01	2.343084E-01
		STD	1.207808E-01	1.587613E-01	2.217183E-01	3.389417E-01	5.737207E-01
	PSO-IPA	MIN	3.884548E-05	4.956974E-04	6.709626E-04	9.851234E-04	1.614704E-03
		Mean	8.049767E-04	1.252013E-03	2.702302E-03	5.331123E-03	3.569278E-03
		STD	4.781792E-02	6.779782E-02	9.834554E-02	1.579233E-01	2.563086E-01
n1r2	PSO	MIN	1.277509E-03	1.682814E-03	2.355185E-03	3.613418E-03	6.132209E-03
		Mean	3.902872E-02	4.989586E-02	6.828645E-02	1.009624E-01	1.615325E-01
		STD	3.725136E-03	5.352253E-03	9.412406E-03	1.703126E-02	1.756931E-02
	IPA	MIN	4.967811E-02	7.034786E-02	1.020456E-01	1.640734E-01	2.650188E-01
		Mean	1.277645E-01	1.682849E-01	2.355235E-01	3.613424E-01	6.132539E-01
		STD	4.019882E-02	5.164601E-02	7.012649E-02	1.031925E-01	1.640626E-01
	PSO-IPA	MIN	5.235338E-03	1.028235E-02	1.470247E-02	2.307142E-02	2.464944E-02
		Mean	4.967818E-02	7.034795E-02	1.020457E-01	1.640736E-01	2.650259E-01
		STD	1.277775E-01	1.682978E-01	2.355371E-01	3.613602E-01	6.133415E-01

References

- Flierl GR (1987) Isolated eddy models in geophysics. *Annu Revs Fluid Mech* 19:493–530
- Raja MAZ, Khan JA, Qureshi IM (2010) A new stochastic approach for solution of Riccati differential equation of fractional order. *Ann Math Artif Intell* 60:229–250
- Khan JA, Raja MAZ, Qureshi IM (2011) Numerical treatment of nonlinear Emden–Fowler equation using stochastic technique. *Ann Math Artif Intell* 63:185–207
- Davis HT (1962) Introduction to nonlinear differential and integral equations. Dover Publications, New York
- Chandrasekhar (1967) Introduction to the Study of Stellar Structure, Dover Publications, New York
- Shawagfeh NT (1993) Nonperturbative approximate solution for Lane–Emden equation. *J Math Phys* 34:4364–4369
- Wazwaz AM (2001) A new method for solving differential equations of the Lane–Emden type. *Appl Math Comput* 118:287–310
- Wazwaz AM (2005) Analytical solution for the time-dependent Emden–Fowler type of equations by Adomian decomposition method. *Appl Math Comput* 166:638–651
- Petviashvili VI (1981) Red spot of Jupiter and the drift soliton in a plasma. *JETP Lett* 32:619–622
- Boyd JP (1991) Monopolar and dipolar vortex solitons in two space dimensions. *Wave Motion* 13:223–241
- Boyd JP (1997) Padé approximant algorithm for solving nonlinear ordinary differential equation boundary value problems on an unbounded domain. *Comput Phys* 2:299–303
- Wazwaz AM (2006) Pade approximants and Adomian decomposition method for solving the Flierl–Petviashvili equation and its variants. *Appl Math Comput* 182:1812–1818
- Din STM, Noor MA (2010) Homotopy perturbation method and Pade approximation for solving Flierl–Petviashvili equation. *Appl Appl Math* 3(2):224–234
- Medi B, Nomvar M (2020) Developing a fast and robust numerical method for the simulation of cocurrent hollow fiber gas separation membranes for process flowsheet synthesis. *SN Appl. Sci* 2:426
- Obiko JO, Mwema FM, Shangwira H (2020) Forging optimisation process using numerical simulation and Taguchi method. *SN Appl Sci* 2:713

16. Atangana A, Araz SI (2020) New numerical method for ordinary differential equations: Newton polynomial. *J Comput Appl Math* 372:112622
17. Ghiasi V, Koushki M (2020) Numerical and artificial neural network analyses of ground surface settlement of tunnel in saturated soil. *SN Appl. Sci* 2:939
18. Kutniv MV, Datsko BY, Kunynets AV, Wloch A (2020) A new approach to constructing of explicit one-step methods of high order for singular initial value problems for nonlinear ordinary differential equations. *Appl Numer Math* 148:140–151
19. Meada AJ, Fernandez AA (1994) The numerical solution of linear ordinary differential equation by feed forward neural network. *Math Comput Model* 19:1–25
20. Khan JA, Raja MAZ, Qureshi IM (2011) Hybrid evolutionary computational approach: application to van der Pol oscillator. *Int J Phy Sci* 6:7247–7261
21. Khan JA, Raja MAZ, Qureshi IM (2011) Novel approach for van der Pol oscillator on the continuous time domain. *Chin Phys Lett* 28:1102–1105
22. Zhao HM, Chen KZ (2002) Neural network for solving systems of nonlinear equations. *Acta Electron* 30:601–604
23. Daniel RR et al (2003) Solving differential equations with unsupervised neural networks. *Chem Eng Process* 42:715–721
24. Lucie PA, Veer PVD (2001) Neural network method for solving the partial differential equations. *Neural Process Lett* 14:261–271
25. Funahashi KI (1989) On the approximate realization of continuous mappings by neural networks. *Neural Netw* 2:183
26. Hornik K, Stichcombr M, White H (1990) Universal approximation of an unknown mapping and its derivatives using multilayer feed forward networks. *Neural Netw* 3:551–560
27. Khan JA, Raja MAZ, Qureshi IM (2011) Stochastic computational approach for complex nonlinear ordinary differential equations. *Chin Phys Lett* 28:0202–0206
28. Kennedy J, Eberhart R (1995) Particle swarm optimization. *Proc IEEE Int Conf Neural Netw Perth Australia* 4:1942–1948
29. Lee KC, Jhang JY (2006) Application of particle swarm optimization algorithm to the optimization of unequally spaced antenna arrays. *J Electromagn Wave Appl* 20:2001–2012
30. Coello CA, Pulido GT, Lechuga MS (2004) Handling multiple objectives with particle swarm optimization. *IEEE Trans Evol Comput* 8:256–279
31. Mahmoud KR, El-Adawy M, Ibrahim SMM (2007) A comparison between circular and hexagonal array geometries for smart antenna systems using particle swarm optimization algorithm. *Prog Electromagn Res PIER* 72:75–90
32. Sivanandam SN, Visalakshi P (2007) Multiprocessor scheduling using hybrid particle swarm optimization with dynamically varying inertia. *Int J Comput Sci Appl* 4:95–106
33. Ong YS, Lim M, Chen X (2010) Memetic computation-past, present & future. *IEEE Comput Intell Mag* 5:24–31
34. Kennedy J, Eberhart R (2001) *Swarm intelligence*, 1st edn. Academic Press, San Diego
35. Yang X-S (2009) Firefly algorithms for multimodal optimization. In: *International symposium on stochastic algorithms*. Springer, Berlin, pp 169–178

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.