Ain Shams Engineering Journal 12 (2021) 2211–2225

Contents lists available at ScienceDirect

Ain Shams Engineering Journal

journal homepage: www.sciencedirect.com

Engineering Physics and Mathematics

Design of an efficient algorithm for solution of Bratu differential equations

Ashfaq Ahmad [a], Muhammad Sulaiman [a,*], Abdulah Jeza Aljohani [b,c], Ahmad Alhindi [d,e], Hussam Alrabaiah [f,g]

[a] Department of Mathematics, Abdul Wali Khan University, Mardan 23200, KP, Pakistan
[b] Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[c] Center of Excellence in Intelligent Engineering Systems, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[d] Department of Computer Science, Umm Al-Qura University, Makkah 21955, Saudi Arabia
[e] Center of Innovation and Development in AI (CIADA), Umm Al-Qura University, Makkah 21955, Saudi Arabia
[f] College of Engineering, Al Ain University, Al Ain, United Arab Emirates
[g] Mathematics Department, Tafila Technical University, Tafila, Jordan

ARTICLE INFO

Article history:
Received 6 July 2020
Revised 11 November 2020
Accepted 21 November 2020
Available online 29 January 2021

Keywords:
Metaheuristics
Symbiotic organism search algorithm
Artificial neural networks
Bratu differential equations
Heat transfer
Fuel ignition

ABSTRACT

In this research, we have suggested a combined strategy to calculate and determine the solutions for problems originating in combustion theory and heat transfer, that are known as Bratu differential equations. We aim to suggest and test a soft computing technique using an efficient meta-heuristic the Symbiotic Organism Search (SOS) algorithm and Artificial neural network (ANN) architecture to obtain better solutions for Bratu differential equations by utilizing fewer computational resources and minimal time. We have simulated our computing approach for different cases, and we compare the outcome of our experiments with solutions obtained by the existing state-of-the-art methods. For novelty, we have found an accurate critical value of λ_c by using SOS algorithm. Values of TIC, MAD, and NSE confirm that our method is a convenient and potential candidate for handling real-application problems. We found that this ANN-SOS algorithm takes less time and is accurate in getting results of the expected standard.

1. Introduction

Bratu problems are also known as "Liouville-Gelfand-Bratu" problems named after Gelfand and nineteenth-century French mathematician Liouville [1]. Bratu problems are essential in applied mathematics, which has a large variety of applications in chemistry, including thermal reactions in different processes, Nanotechnology, and Chandrasekhar mathematical model of the evolution of the universe. Ignition problems or Bratu differential equations are essential for the analysis of systems involving heat transfer problems, which are studied in [2]. Exothermal explosions are studied in a slab, cylindrical pipe, and in symmetric geometries by using series summation technique and perturbation technique [3–5]. In thermal combustion theory, rearranging the mathematical model representing the solid fuel ignition results in an elliptic partial differential equation having characteristics of a highly non-linear Eigenvalue problem, known as Bratu differential equations. In the current paper, we have considered a one-dimensional Bratu differential equation which is given in Eq. (1) [6]:

$$y''(x) + \lambda e^{y(x)} = 0, \quad y(0) = y(1) = 0, \tag{1}$$

where $x \in [0, 1]$ and $\lambda > 0$. Eq. (1) represents a standard Bratu problem which shows modeling of the combustion problem in a numerical slab. Many researchers in [7–12], have presented interesting work about the solution of Bratu's problems. Various numerical approaches such as B-spline method [7], Adomian decomposition method (ADM) [9], finite difference method [8], weighted residual method [11] and 1-D differential transform method [10] have been applied for the solution of the Bratu's problems. Artificial Neural Networks (ANN) along with the Interior point technique has also been applied for the solution of 1-D Bratu's problems [13]. The

* Corresponding author.
E-mail addresses: msulaiman@awkum.edu.pk (M. Sulaiman), ajaljohani@kau.edu.sa (A.J. Aljohani), ahhindi@uqu.edu.sa (A. Alhindi).

Peer review under responsibility of Ain Shams University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.asej.2020.11.007
2090-4479/

Nomenclature

λ_c	Critical value of λ
f	Activation function
j	Total number of neurons
$\alpha_i, \beta_i, \omega_i$	Unknown weights
$y(\hat{x})$	Approximate series solution
E_1	Solution error of ordinary differential equation
E_2	Solution error of initial/boundary values
MV	Mutual vector
X_{best}	the best degree of adaptation
$rand(0,1)$	Vector generated from random numbers in interval $(0,1)$
BF_1	Benefit factor for first organism
BF_2	Benefit factor for second organism
X_{inew}	Updated i_{th} organism
X_{jnew}	Updated j_{th} organism
y_m	The exact solution
ANNs	Artificial Neural Networks
ODE	Ordinary Differential Equation
SOS	Symbiotic Organism Search algorithm
STD	Standard Deviation
TIC	Theil's Inequality Coefficient
MAD	Mean Absolute Deviation
NSE	Nash–Sutcliffe Efficiency
ENSE	Error in Nash–Sutcliffe efficiency
IVP	Initial Value Problem
BVP	Boundary Value Problem
ADM	Adomian Decomposition Method
CANN	Cascade Artificial Neural Networks
GA	Genetic Algorithm

exact solution of 1-D Bratu's problem has the following form in planar coordinates:

$$y(x) = -2ln\left[\frac{cosh((x-0.5)\frac{\theta}{2})}{cosh(\frac{\theta}{4})}\right], \tag{2}$$

where θ satisfies the equation

$$\theta^2 = 2\lambda cosh^2\left(\frac{\theta}{4}\right), \tag{3}$$

Eq. (1) has zero solution if the value of λ is greater than the critical value λ_c, one solution if $\lambda = \lambda_c$ and two solutions if the value of λ is less than λ_c. Differentiating Eq. (3) with respect to θ and taking $\lambda\prime(\theta) = 0$ then we get:

$$\theta = \frac{1}{2}\lambda_c cosh\left(\frac{\theta}{4}\right)sinh\left(\frac{\theta}{4}\right), \tag{4}$$

from Eq. (4), we use the value of λ_c in Eq. (3), we get

$$\frac{\theta_c}{4} = coth\left(\frac{\theta_c}{4}\right). \tag{5}$$

We have used SOS algorithm to solve Eq. (5) and obtained $\theta_c = 4.79871456103093$. Using value of θ_c in Eq. (3) we obtained $\lambda_c = 3.513830719125161$.

ANNs are capable of finding quality solutions at instantaneous points in the search space. Series solutions calculated by ANN can approximate the solution for a differential equation on the points that were not considered during simulations. Methods based on ANN for solving differential equations are more accurate than other classical numerical techniques [15]. The ANN-based mathematical models have been used for the solution of problems with initial and boundary conditions [16,17,15,18]. A two-dimensional mathematical model representing the Kirchhoff plate theory is analyzed by using a deep collocation method [19]. An artificial neural network is designed for the solution of second-order boundary value ordinary differential equations [20]. Mathematical models represented by partial differential equations are solved by an energy approach using machine learning [21]. Longitudinal waves are studied in a circular rod with magneto-electro-elastic characteristics in [22]. Another study of the propagation of surface waves with the help of the nonlinear dispersive Davey-Stewartson system and its stability is presented in [23]. Solutions of Kadomtsev–Petviashvili and modified Kadomtsev–Petviashvili dynamical equations are elaborated in [24]. In [25], a detailed study is carried out for exact solitary wave solutions by using mathematical methods for the nonlinear two-dimensional water waves of the Olver dynamical equation. Several exact and approximate techniques are developed to solve mathematical models involving partial differential equations [23,26,25,27,28,22,24,29]. ANNs based soft computing paradigms gained the attention of researchers in recent years. A plant propagation algorithm (PPA) was designed to solve design engineering problems [30]. A modified version of PPA is presented in [31]. Impacts of different crossover operators are investigated for handling multi-objective problems [32]. An improved version of the genetically adaptive multi-algorithm paradigm is studied in [33]. Plant propagation algorithm is modified and applied to constrained, unconstrained problems, and theoretical analysis are studied in [34–36]. A state-of-the-art survey is published in [61], where evolutionary algorithms are investigated in terms of decomposition and indicator functions [37,38,61]. In electrical engineering, several metaheuristics are used to solve complex optimization problems [39–42]. Unconstrained single-objective optimization problems are solved by using a hybrid of global and local search procedures [43,38]. The optimal design of heat fins is proposed in [44]. A study of temperature distribution in heat fins is carried out by using a hybrid of the Cuckoo Search (CS) algorithm and Artificial Neural Network architecture [45,46]. Neuro-fuzzy modeling is used to predict the summer precipitation in targeted metrological sites [47]. An interesting study of financial market forecasting is accomplished by the ARFIMA-LSTM technique [47]. Fractional order DPSO algorithm is used to solve the corneal model for eye surgery [48]. A novel initialization strategy is introduced in a multi-verse optimization technique, and different design engineering problems are solved in [49]. Nonlinear dusty plasma systems are analyzed with the help of NAR-RBFs neural networks [50]. A neuro-evolutionary algorithm is applied to investigate oscillatory behavior of heart beat [51]. Singular ordinary differential equations are handled by a hybrid of DPSO and artificial neural networks [52]. Fractional differential equations representing the damping materials are analysed by an efficient soft computing algorithm [52,53,61].

In [13], a hybrid algorithm of ANN and a single path following local search technique, the Interior point technique (IPT) was developed to train the unknown weights involved in the architecture of neural networks. IPT is a local search technique, and it can easily get trapped in local optima. In [54], a hybrid technique is proposed in which two metaheuristics are combined to solve Bratu,s differential equations. The unknown weights of ANNs are trained by the Genetic algorithm and the Teaching learning-based optimization (TLBO) algorithm. It is evident that CANN-GA-TLBO was slow, and it was consuming more computational resources. To address this issue, the authors of this manuscript have hybridized ANNs with a new and efficient single population-based technique known as the SOS algorithm, which is capable of balanced exploration and exploitation. The outcome of the ANN-SOS algorithm is encouraging and better than the results of state-of-the-art algorithms. The ANN-SOS algorithm is used for finding solutions for three cases of 1-D Bratu's differential equations. To further evaluate the quality of our solutions, we have calculated the values of three performance indicators: MAD, TIC, and ENSE. It is sown that the ANN-SOS technique is efficient and consumes less computing resources. A comparison of our results with well known analytical techniques like the B-Spline Method and the Adomian Decomposition Method (ADM) dictates that the ANN-SOS algorithm is fast and efficient. Four Bratu problems are considered. Problem one is an initial value problem, and the rest of the three cases are boundary value problems with different values of constant λ.

Key findings in this paper are summarised as follows:

- We have developed a new unsupervised computing paradigm, the ANN-SOS algorithm. It is fast and efficient and consumes less computational resources. A flowchart is explaining how our algorithm works is depicted in Fig. 1.
- An important real-life application from ignition problems is analyzed. These problems are named as Bratu differential equations. Four cases of Bratu problems are solved with the help of the ANN-SOS algorithm. One problem is an initial value differential equation, while the rest of the three cases are boundary value problems with distinct values of λ.
- Performance indicators, MAD, TIC and ENSE, are used to evaluate the efficiency and accuracy of the ANN-SOS algorithm.

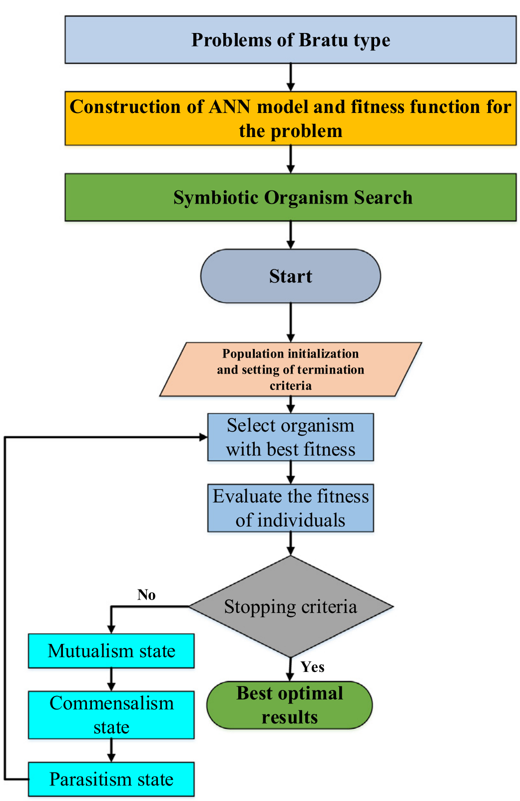


Fig. 1. Flowchart for ANN-SOS algorithm.

Table 1
Comparison of values of θ_c and λ_c obtained by iterative method and SOS algorithm.

	Iterative Method [14]	SOS
θ_c	4.798714560	4.79871456103093
λ_c	3.513830720	3.51383071912516
Error according to Eq. (3)	5.73E−09	**3.55E−15**

2. Mathematical formulation

An estimated simplified approach to the problem under consideration and its nth derivative is given in Eq. (6) [54]:

$$\hat{y}(x) = \sum_{i=1}^{j} \alpha_i f(\beta_i x + \omega_i), \tag{6}$$

$$\frac{d^n}{dx^n}\hat{y}(x) = \sum_{i=1}^{j} \alpha_i \frac{d^n}{dx^n} f(\beta_i x + \omega_i), \tag{7}$$

In Eq. (4), f is used as an activation function and the unknown weights are given as α_i, β_i and ω_i. The number of terms in series solution are j. In ANN architecture, log-sigmoid function is used as an amplifier and is given as follows,

$$f(z) = \frac{1}{1+e^{-z}}. \tag{8}$$

The approximate series solution for Bratu differential equation is given in Eq. (9),

$$\hat{y}(x) = \sum_{i=1}^{j} \alpha_i \left(\frac{1}{1+e^{-(\beta_i x + \omega_i)}} \right), \tag{9}$$

and the second derivative of $\hat{y}(x)$ is given in Eq. (10),

$$\hat{y}''(x) = \sum_{i=1}^{j} \alpha_i \beta_i^2 \left(\frac{2e^{-2(\beta_i x+\omega_i)}}{\left(1+e^{-(\beta_i x+\omega_i)}\right)^3} - \frac{e^{-(\beta_i x+\omega_i)}}{\left(1+e^{-(\beta_i x+\omega_i)}\right)^2} \right). \tag{10}$$

2.1. Fitness criteria for solutions

After calculating $\hat{y}(x)$ by Eq. (9), then a mean-squared error is calculated by putting the value of $\hat{y}(x)$ in differential equation and initial/ boundary conditions. These errors are denoted by E_1 and E_2 and are given in Eqs. (12) and (13). The minimization objective is as follows,

$$min \quad E = E_1 + E_2, \tag{11}$$

where E_1 is given by:

$$E_1 = \frac{1}{N+1} \sum_{m=0}^{N} \left(\hat{y}''_m(x) + \lambda e^{\hat{y}_m} \right)^2, \tag{12}$$

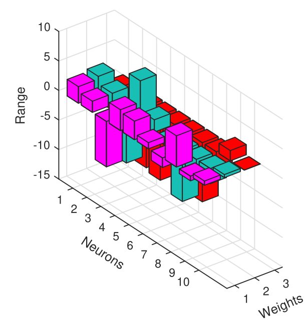
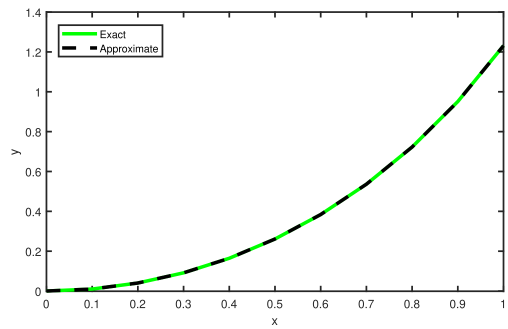
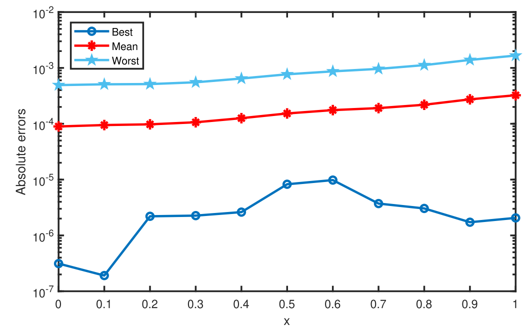


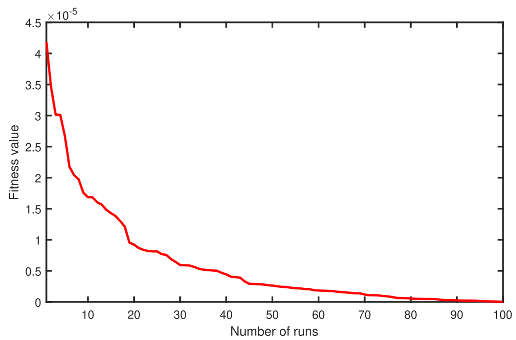
Fig. 2. Best set of weights obtained by ANN-SOS algorithm for Bratu IVP.



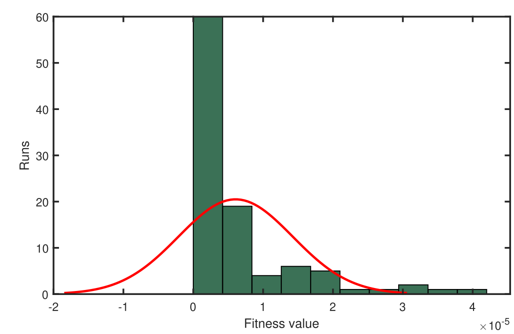
(a) The graph of ANN-SOS and exact solution.



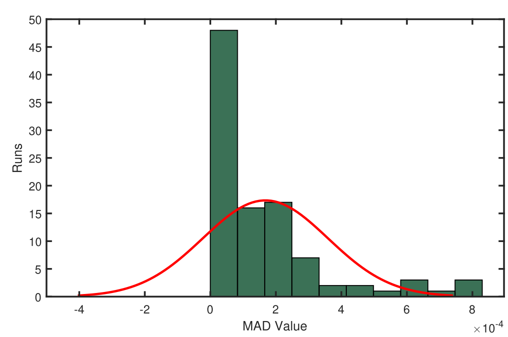
(b) Best, Mean and worst errors in solutions.



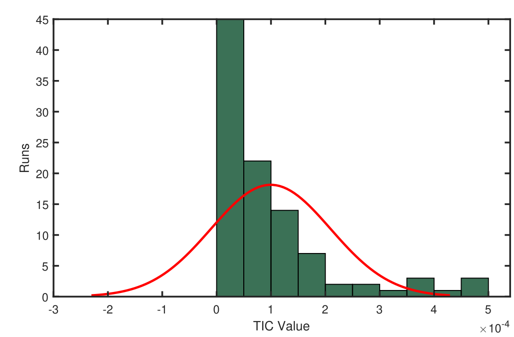
(c) Convergence of the fitness values in 100 runs.



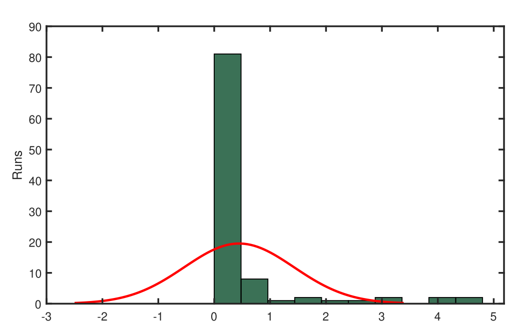
(d) Histogram of fitness values.



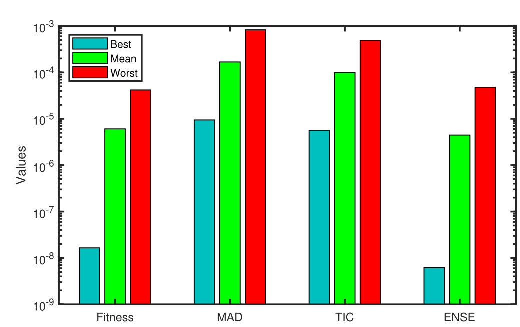
(e) Histogram of MAD values.



(f) Histogram of TIC values.



(g) Histogram of ENSE values.



(h) Best, mean and worst values of performance indicators.

Fig. 3. Results obtained by ANN-SOS for Bratu initial value problem.

where $N = \frac{1}{h}, \hat{y}_m = \hat{y}(x_m)$ and $x_m = mh$. The domain for the problem is taken from the interval $(0, 1)$ which is divided in N subintervals $(x_0 = 0, x_1), (x_2, x_3), \ldots, (x_{N-1}, x_N = 1)$ with the step size $h, \hat{y}(x)$ and $\hat{y}''(x)$ are the series solutions based on the neural networks as given in Eqs. (9) and (10).

Similarly, E_2 is defined as:

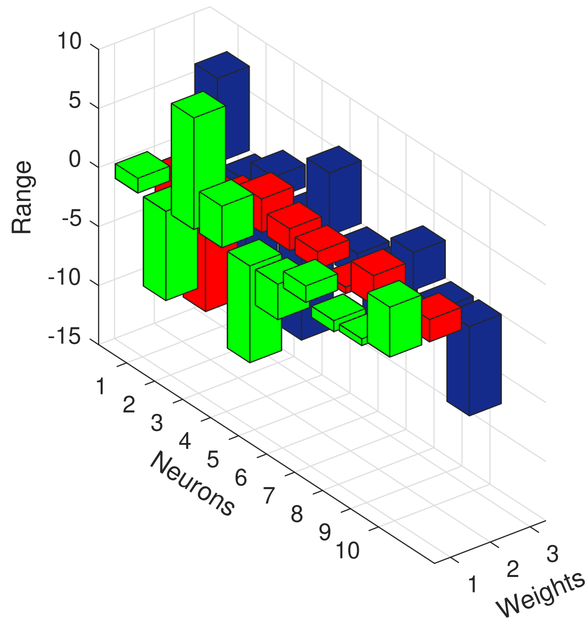


Fig. 4. Weights obtained for ANN-SOS for Bratu BVP with $\lambda = 1$.

$$E_2 = \frac{1}{2}\left((\hat{y}_0(x))^2 + (\hat{y}_1(x))^2\right). \tag{13}$$

E_1 is the error related to the differential equation while E_2 represents errors related to the boundary conditions in the Bratu problem. It is evident that for the weights α_i, β_i and ω_i in Eqs. (9) and (10) which are adjustable parameters, if E_1 and E_2 approaches to 0 for these parameters then E will also approach zero. Hence, the solution $\hat{y}(x)$ will be the best required result.

3. Optimizer for objective function

After building the neural network, we get unknown weights and minimization objective function as in Eq. (11). A well-balanced minimizer is needed to optimize the objective function and obtain the best set of weights. In our novel approach, we have chosen the Symbiotic Organism Search (SOS) algorithm to accomplish the task of optimization. We name our technique the ANN-SOS algorithm. SOS optimizer is a nature-inspired technique that simulates the process of survival of organisms in an ecosystem [55]. The SOS algorithm uses three phases; mutualism, commensalism, and parasitism. Each phase defines the states of organisms in an ecosystem. The search equations mimicking all the three phases are given in the following sections.

3.1. Mutualism state

The example of flowers and bees shows a mutualism relationship, which is beneficial for both participant organisms. This phase of SOS represents such a mutual relationship between organisms of the ecosystem. In the SOS algorithm, X_i is the organism assumed as the i_{th} member in the ecosystem. It randomly selects the other organism X_j from the ecosystem for interaction with organism X_i. Both of the organisms want to improve their survival inside the ecosystem, so they engage in a mutual relationship with one another. The new candidate solutions for organisms X_i and X_j are computed based on the mutual symbiosis between them, according to the Eq. (11)

$$X_{inew} = X_i + rand(0,1) * (X_{best} - MV * BF_1), \tag{14}$$

$$X_{jnew} = X_j + rand(0,1) * (X_{best} - MV * BF_2), \tag{15}$$

$$MV = \frac{Xi + Xj}{2}, \tag{16}$$

$rand(0,1)$ is a vector generated from random numbers. Here, the benefit factors BF_1 and BF_2 are randomly chosen either 1 or 2. The factors denote the benefit level for each organism if an organism is partially or fully getting benefits from the mutual relationship. Eq. (16) represents a vector that is known as Mutual Vector that denotes the characteristics of the relationship between the organisms X_i and X_j. In Eqs. (14) and (15), X_{best} represents the best degree of the adaptation. Therefore, X_{best} shows the best-adapted candidate solution. By using the dimensions of X_{best}, the fitness of both X_i and X_j is improved. Finally, the fitness of the current best and global best is compared, and the global best is replaced by the fittest solution.

3.2. Commensalism state

At this stage, two organisms X_i and X_j are selected from a pole of candidate solutions. X_i is privileged to have more benefit than X_j. Moreover, X_j participates in this stage on a "no profit no loss" basis. A new solution is computed by using Eq. (15). If the candidate solution X_i is improved then it is updated as follows,

$$X_{inew} = X_i + rand(-1,1) * (X_{best} - X_j), \tag{17}$$

The part of the equation, $(X_{best} - X_j)$, mimics the advantage provided by X_j to X_i, improving changes of its survival in the ecosystem.

3.3. Parasitism state

In the parasitism stage, a random candidate solution X_i is chosen as a base vector for reproduction. X_i is then modified by randomly changing its dimensions. Another solution X_j is randomly picked from a population of solutions, and finally, the fittest solution replaces the solution with low fitness.

4. Performance measures

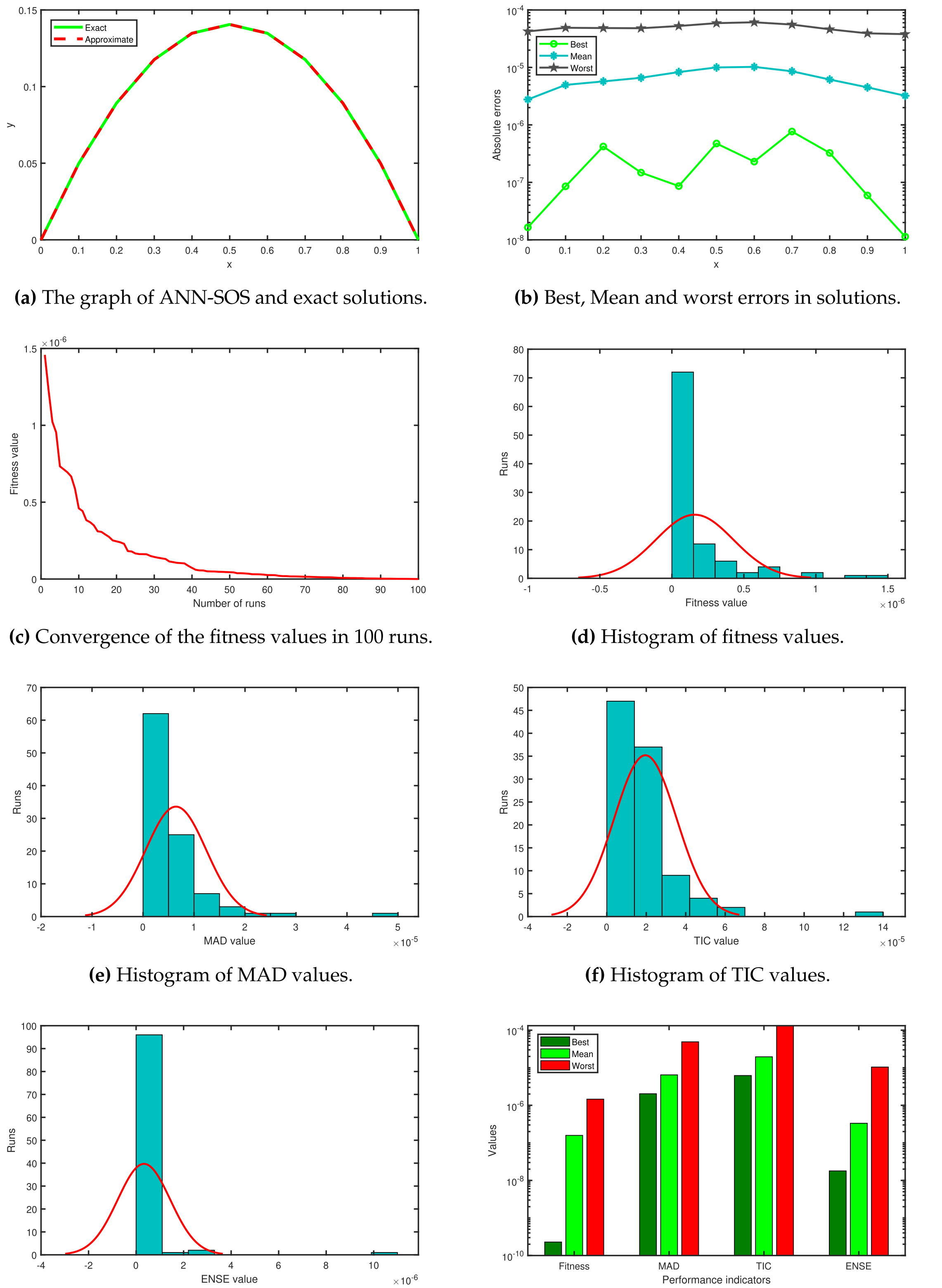
We have performed 100 simulations on all four problems to establish the stability, adaptability, and certainty of the ANN-SOS algorithm. For this purpose, we have determined the mean absolute deviation (MAD) in solutions, root-mean-square error (RMSE), error in Nash–Sutcliffe efficiency (ENSE), Theil's inequality coefficient (TIC), and Nash–Sutcliffe efficiency (NSE). The analytical definition of these indexes are provided in Eqs. (18)–(21), (see Table 1)

$$MAD = \frac{1}{n}\sum_{m=1}^{n}|y_m - \hat{y}_m|, \tag{18}$$

$$TIC = \frac{\sqrt{\frac{1}{n}\sum_{m=1}^{n}(y_m - \hat{y}_m)^2}}{\left(\sqrt{\frac{1}{n}\sum_{m=1}^{n}y_m^2} + \sqrt{\frac{1}{n}\sum_{m=1}^{n}\hat{y}_m^2}\right)}, \tag{19}$$

$$NSE = 1 - \frac{\sum_{m=1}^{n}(y_m - \hat{y}_m)^2}{\sum_{m=1}^{n}(y_m - \bar{y}_m)^2}, \quad \bar{y}_m = \frac{1}{n}\sum_{m=1}^{n}y_m, \tag{20}$$

$$ENSE = 1 - NSE. \tag{21}$$



(a) The graph of ANN-SOS and exact solutions.

(b) Best, Mean and worst errors in solutions.

(c) Convergence of the fitness values in 100 runs.

(d) Histogram of fitness values.

(e) Histogram of MAD values.

(f) Histogram of TIC values.

(g) Histogram of ENSE values.

(h) Mean, best and worst values of performance indicators.

Fig. 5. Results obtained by ANN-SOS for Bratu BVP with $\lambda = 1$.

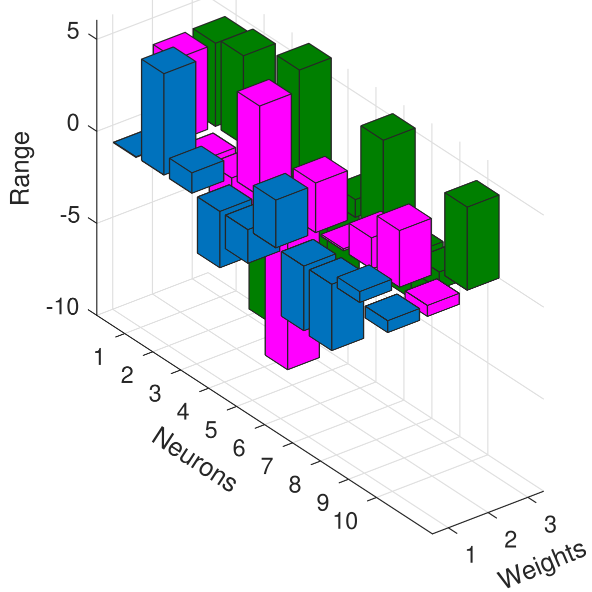


Fig. 6. Weights obtained for ANN-SOS for Bratu BVP with $\lambda = 2$.

5. Simulations and results

The proposed ANN-SOS algorithm has been implemented to solve four cases of the Bratu boundary/ initial value problems for different values of constant λ, and the results achieved by the ANN-SOS algorithm for $\lambda = 1, 2$, and 3.51 are given in Figs. 2–9 and Tables 2–16. We have compared our results with the B-spline technique and state-of-the-art solutions. In this paper, we have considered one initial value problem and three boundary value problems, which are collectively identified as Bratu differential equations.

5.1. ***Problem 01:*** *Bratu Differential Equation with Initial Values*

The Bratu initial value problem is given by:

$$y''(x) - 2e^{y(x)} = 0, \quad 0 \leqslant x \leqslant 1, \tag{22}$$

$$y(0) = y'(0) = 0, \tag{23}$$

we have solved the problem in (22) and (23) using the ANN architecture given in Eqs. (9) and (10). In each hidden layer, there are 10 neurons (10 terms in series solution) and the unknown weights are 30. The input variable x is varied over the interval $(0, 1)$ choosing a step size of $h = 1/10$, i.e., solutions are found at 11 grid points. We give the fitness function for Bratu IVP as:

$$E = \frac{1}{11}\sum_{m=0}^{10}\left(\hat{y}''_m - 2e^{\hat{y}_m}\right)^2 + \frac{1}{2}\left((y_0)^2 + (y'_0)^2\right). \tag{24}$$

The fitness function (24) is trained and optimized by the ANN-SOS algorithm. Our approach has successfully calculated the best solution with lower residual error as 1.6492×10^{-8}. The best set of weights obtained by ANN-SOS technique to minimize the fitness function are plotted in Fig. 2 and the series solution of the problem is given in Eq. (25),

$$\begin{aligned}\hat{y}(x) = {} & \frac{2.80930375218414}{1+e^{-(3.25877858055612x-4.42324022732111)}} + \frac{1.88533586713111}{1+e^{-(1.92935080260280x-6.54424119668768)}} \\ & + \frac{-7.69491236009608}{1+e^{-(-8.34008116021304x-10.4285997231487)}} + \frac{3.53291632086472}{1+e^{-(7.13418766557493x-11.0083125489669)}} \\ & + \frac{3.14205498842095}{1+e^{-(2.07492261571949x-4.56180146345720)}} + \frac{1.04196408052834}{1+e^{-(-0.817341259587510x-4.90879356410201)}} \\ & + \frac{-1.61892545895628}{1+e^{-(-8.67522044525885x-10.0243195066148)}} + \frac{5.28753700692028}{1+e^{-(-2.23065868564640x-3.01513897945583)}} \\ & + \frac{-0.793273842351156}{1+e^{-(-2.56336088924486x+1.74762681409507)}} + \frac{0.713782054972897}{1+e^{-(0.520177614554118x-0.0243667449777310)}}\end{aligned} \tag{25}$$

Exact and ANN-SOS solution of the Bratu IVP are presented in Table 2 and solutions are depicted in Fig. 3a. It is obvious that ANN-SOS techniques is accurate and efficient. The worst, best, and mean absolute errors in the results for Bratu IVP are presented in Fig. 3b. Fig. 3c shows the convergence of fitness values during 100 runs. Histograms with normal distribution fittings for fitness values, MAD, TIC and ENSE values are given in Fig. 3d,e, f and g respectively. The figures show that most of the values are less than 10^{-05} which dictates the accuracy of our algorithm. The worst, best and mean values of the performance indicators are presented in Fig. 3h. Statistical analysis of absolute errors is given in Table 4. The minimum values of absolute errors are in the range 10^{-06} to 10^{-07}, mean values are in the range 10^{-04} to 10^{-05} and standard deviation (STD) is about 10^{-04}. Statistical analysis of performance indicators is given in Table 3. The fitness values range from 10^{-05} to 10^{-08}, MAD values range from 10^{-04} to 10^{-06}, TIC values range from 10^{-04} to 10^{-06} and ENSE values range from 10^{-05} to 10^{-09}.

5.2. ***Problem 02:*** *Bratu Differential Equation with Boundary Values and λ = 1*

The Bratu BVP with $\lambda = 1$ is given as:

$$y''(x) + e^{y(x)} = 0, \quad 0 \leqslant x \leqslant 1, \tag{26}$$

$$y(0) = y(1) = 0, \tag{27}$$

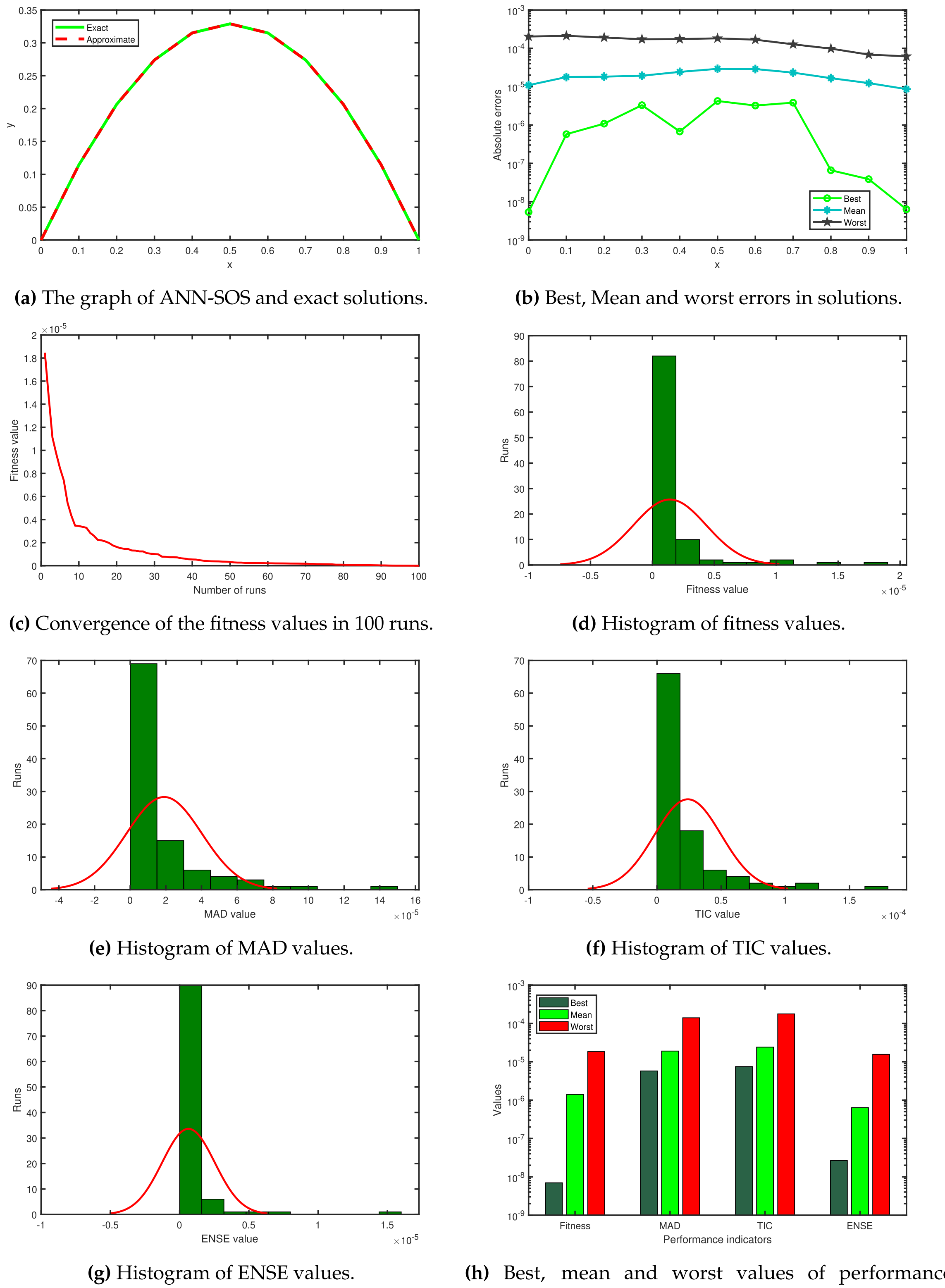
The exact solution of Bratu BVP with $\lambda = 1$ is obtained using the value of $\theta = 1.5172$ over the interval $(0, 1)$ and a step size of $h = 1/10$. The ANN-SOS algorithm is used to approximate the solution $\hat{y}(x)$ of the Bratu BVP. The fitness function for the first case of BVP is given in Eq. (28),

$$E = \frac{1}{11}\sum_{m=0}^{10}\left(\hat{y}''_m + e^{\hat{y}_m}\right)^2 + \frac{1}{2}\left((y_0)^2 + (y_1)^2\right). \tag{28}$$

Our goal is to find the weights for which the error E is miminum, i.e. $\hat{y}(x) \to y(x)$. Previous fitness values obtained by ANN using gradient descent algorithm, L-M method and conjugate gradient method are $3.20 \times 10^{-3}, 1.39 \times 10^{-4}$ and 3.95×10^{-3} respectively while ANN-SOS algorithm obtained solutions with error 2.2883×10^{-10}. Weights obtained by ANN-SOS algorithm for Bratu problem with $\lambda = 1$ are plotted in Fig. 4 and series solution for the problem is given in Eq. (29),

$$\begin{aligned}\hat{y}(x) = {} & \frac{1.24728531290468}{1+e^{-(-3.06401684752794x+7.12560604918088)}} + \frac{-7.56441140211341}{1+e^{-(-9.79059355122726x-10.4361025757102)}} \\ & + \frac{9.45007659500737}{1+e^{-(1.03536059437449x+1.86401803944560)}} + \frac{3.51460879132383}{1+e^{-(2.83342068579553x-10.4106108958789)}} \\ & + \frac{-8.19422278273552}{1+e^{-(1.86044465387359x+5.30109579770407)}} + \frac{-3.00415724602500}{1+e^{-(1.44843577295695x-2.15016843886231)}} \\ & + \frac{1.39211861821510}{1+e^{-(-0.547875973317433x-4.54733797418289)}} + \frac{-0.936704068554820}{1+e^{-(2.51380014579328x+3.18718847560574)}} \\ & + \frac{-0.570575521871031}{1+e^{-(0.177489786814245x-1.84310289784770)}} + \frac{4.22908685810153}{1+e^{-(1.88631176501465x-7.56046292101172)}}.\end{aligned} \tag{29}$$

Results obtained by ANN-SOS algorithm for Bratu problem with $\lambda = 1$ are compared with the exact solution and other analytical methods like B-spline [7], ADM [9] and ANN based L-M method [56]. Numerical solutions for Bratu BVP with $\lambda = 1$ are given in Table 5. The graph of exact and approximate solution of the Bratu



(a) The graph of ANN-SOS and exact solutions.

(b) Best, Mean and worst errors in solutions.

(c) Convergence of the fitness values in 100 runs.

(d) Histogram of fitness values.

(e) Histogram of MAD values.

(f) Histogram of TIC values.

(g) Histogram of ENSE values.

(h) Best, mean and worst values of performance indicators.

Fig. 7. Results obtained by ANN-SOS for Bratu BVP with $\lambda = 2$.

BVP with $\lambda = 1$ is given in Fig. 5a which shows that our solutions are in strong agreement with the exact solutions. The absolute errors in the solutions at each input x are given in Table 6 and our results show that ANN-SOS gives better results than other algorithms. The best, mean and worst absolute errors in the solutions are plotted in Fig. 5b. Convergence of fitness values for all 100 runs is given in Fig. 5c. Histograms with normal distribution fittings for fitness values, TIC, MAD, ENSE values are plotted in Fig. 5d, e, f and g respectively. Statistical analysis of absolute errors is given in Table 7. The minimum values of absolute errors are in the range 10^{-07} to

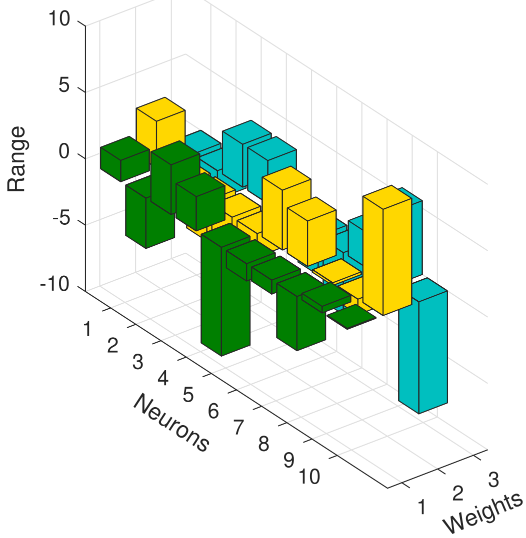


Fig. 8. Weights obtained for ANN-SOS for Bratu BVP with $\lambda = 3.51$.

10^{-8}, mean values are in the range 10^{-05} to 10^{-06} and standard deviation (STD) is about 10^{-06}. Statistical analysis of performance indicators is given in Table 8. Fitness values range from 10^{-06} to 10^{-10}, MAD values range from 10^{-05} to 10^{-06}, TIC values range from 10^{-04} to 10^{-06} and ENSE values range from 10^{-05} to 10^{-08}.

5.3. *Problem 03: Bratu Differential Equation with Boundary Values and $\lambda = 2$*

The Bratu BVP with $\lambda = 2$ is given as:

$$y''(x) + 2e^{y(x)} = 0, \quad 0 \leqslant x \leqslant 1, \tag{30}$$

$$y(0) = y(1) = 0, \tag{31}$$

The exact solution for Bratu BVP with $\lambda = 2$ can be obtained using $\theta = 2.3576$ over the interval $(0, 1)$ and step size is taken as $h = 1/10$. We have implemented the ANN-SOS algorithm to find the approximate solution $\hat{y}(x)$ of the Bratu BVP with $\lambda = 2$. The fitness function for the second case of BVP is given by:

$$E = \frac{1}{11}\sum_{m=0}^{10}\left(\hat{y}''_m + 2e^{\hat{y}_m}\right)^2 + \frac{1}{2}\left((y_0)^2 + (y_1)^2\right). \tag{32}$$

The weights obtained by ANN-SOS algorithm to minimize the fitness function for the Bratu BVP with $\lambda = 2$ are given in Fig. 6 and series solution for the problem is given in Eq. (33). The minimum fitness value obtained by the ANN-SOS algorithm for this case is 6.9844×10^{-09}.

$$\begin{aligned}\hat{y}(x) = {} & \frac{-0.0317583696308022}{1+e^{-(4.48013903492425x+4.51356342526243)}} + \frac{5.50642380672547}{1+e^{-(-1.45565879681638x+4.82204338938373)}} \\ & + \frac{1.10844521686188}{1+e^{-(-1.92243057005630x-8.73496512037616)}} + \frac{-3.06023905770815}{1+e^{-(4.89882671644400x+6.02651866456611)}} \\ & + \frac{-1.82817215674307}{1+e^{-(-8.44109852071930x-6.22450081338830)}} + \frac{2.59738183046781}{1+e^{-(2.69592019574108x+0.963688888178887)}} \\ & + \frac{-3.51468490972454}{1+e^{-(0.107881652053869x+5.17877699511452)}} + \frac{-3.60217575258556}{1+e^{-(1.69570925870392x-2.43442472472352)}} \\ & + \frac{0.591745404895704}{1+e^{-(3.08065117618813x-0.929126188225333)}} + \frac{-0.644096886462373}{1+e^{-(-0.594903863624441x+4.51470864034716)}}.\end{aligned} \tag{33}$$

Exact and ANN-SOS results are presented in Table 9 and Fig. 7a. Absolute errors in solutions obtained by ANN-SOS are compared with exact and approximate results in Table 10. The table shows that ANN-SOS gives better solution than other techniques. The mean, best and worst absolute errors are plotted in Fig. 7b. Convergence of fitness values for 100 runs is given in Fig. 7c. Histograms with normal distribution fitting for fitness values, MAD, TIC and ENSE values are plotted in Fig. 7d, e, f and g respectively. The mean, best and worst values of performance indicators are given in Fig. 7h.

Statistical analysis of absolute errors in solutions is given in Table 11. The minimum values of absolute errors in solutions are in the range 10^{-06} to 10^{-09}, mean values are in the range 10^{-05} to 10^{-06} and standard deviation (STD) is about 10^{-05}. In Table 12, statistical anlysis of performance indicators is presented. Fitness values range from 10^{-05} to 10^{-09}, MAD values range from 10^{-04} to 10^{-06}, TIC values range from 10^{-04} to 10^{-06} and ENSE values range from 10^{-05} to 10^{-08}.

5.4. *Problem 04: Bratu Differential Equation with Boundary Values and $\lambda = 3.51$*

The Bratu BVP with $\lambda = 3.51$ is given as:

$$y''(x) + 3.51e^{y(x)} = 0, \quad 0 \leqslant x \leqslant 1, \tag{34}$$

$$y(0) = y(1) = 0, \tag{35}$$

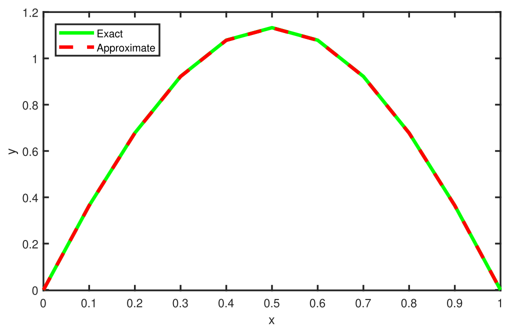
The exact solution for Bratu BVP with $\lambda = 3.51$ is obtained using $\theta = 4.6678$ over the interval $(0, 1)$ with a step size of $h = 1/10$. ANN-SOS algorithm is implemented for the approximate solution $\hat{y}(x)$ of the Bratu BVP with $\lambda = 3.51$. The fitness function for the third case of BVP is given by:

$$E = \frac{1}{11}\sum_{m=0}^{10}\left(\hat{y}''_m + 3.51e^{\hat{y}_m}\right)^2 + \frac{1}{2}\left((y_0)^2 + (y_1)^2\right). \tag{36}$$

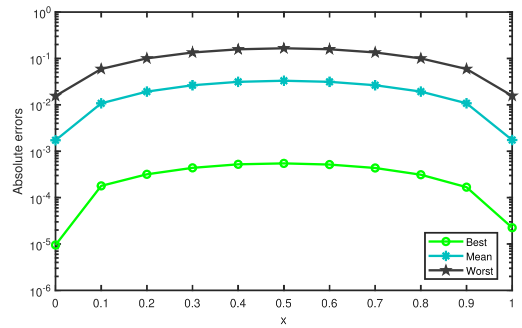
The minimum fitness value obtained by the ANN-SOS algorithm for this case is 9.8730×10^{-08}. Weights found by the ANN-SOS algorithm to minimize the fitness function are given in Fig. 8 and the series solution of the problem is given in Eq. (37).

$$\begin{aligned}\hat{y}(x) = {} & \frac{1.60868459603217}{1+e^{-(3.52294222692223x-1.04346832830829)}} + \frac{-3.79334485062983}{1+e^{-(-1.32302207966912 8x-2.13500959962196)}} \\ & + \frac{3.74597814878045}{1+e^{-(-3.02822468673060x+3.22703105107574)}} + \frac{2.58401438587289}{1+e^{-(-2.43837526010411x+3.23470235156649)}} \\ & + \frac{-8.19454097262985}{1+e^{-(-1.76266092949907x-0.710663034413324)}} + \frac{-1.30281343637084}{1+e^{-(4.50621461903119x-2.92157331861317)}} \\ & + \frac{-1.05392991013643}{1+e^{-(3.42137926339057x-4.65967442258375)}} + \frac{-4.08568323589360}{1+e^{-(-0.202272526240823x+2.66054036462423)}} \\ & + \frac{0.489637102697201}{1+e^{-(-1.16888243272632x+5.56029032511399)}} + \frac{0.0689083029401389}{1+e^{-(8.00639273072346x-8.43967448823257)}}.\end{aligned} \tag{37}$$

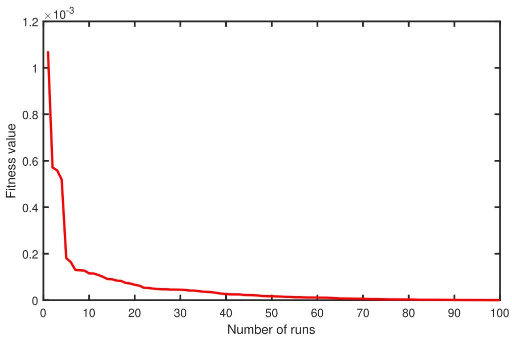
The exact and approximate solutions obtained by ANN-SOS algorithm and other techniques are given in Table 13. The graphs of exact and ANN-SOS solutions are plotted in Fig. 9a. The absolute errors in the solution calculated by ANN-SOS are compared with other techniques in Table 14. From Table 14, it is clear that ANN-SOS gives better results than other techniques. The best, mean and worst values of absolute errors in solutions obtained by ANN-SOS are given in Fig. 9b. Convergence of the fitness values is given in Fig. 9c. Histograms with normal distribution fitting for fitness values, ENSE, MAD and TIC values are plotted in Fig. 9d, e, f and g respectively. Fig. 9h shows the best, mean and worst values of fitness, MAD, TIC and ENSE. Statistical analysis of absolute errors in the solutions obtained by ANN-SOS is presented in Table 15. The minimum values of absolute errors range from 10^{-04} to 10^{-06}, mean values of absolute errors range from 10^{-02} to 10^{-03} and standard deviation (STD) range from 10^{-02} to 10^{-03}. Statistical analysis of performance indicators is presented in Table 16. Fitness, MAD, ENSE and TIC values range from 10^{-03} to $10^{08}, 10^{-01}$ to $10^{-04}, 10^{-02}$ to 10^{-04} and 10^{-01} to 10^{-06} respectively. We have solved bratu boundary value problem for $\lambda = 1, 2$ and 3.51 by training ANN using Bat algorithm [57] and PSO [58] and obtained solutions are compared with ANN-SOS as given in Table 18. The table also shows that ANN-SOS given better solutions than bat algorithm and PSO. To



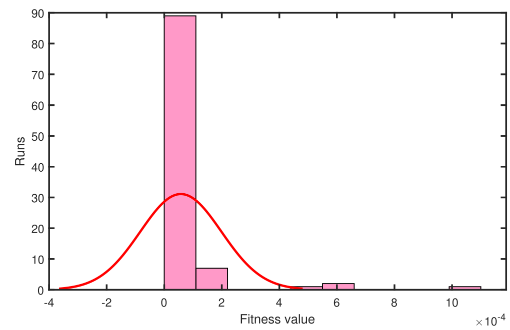
(a) The graph of ANN-SOS and exact solutions.



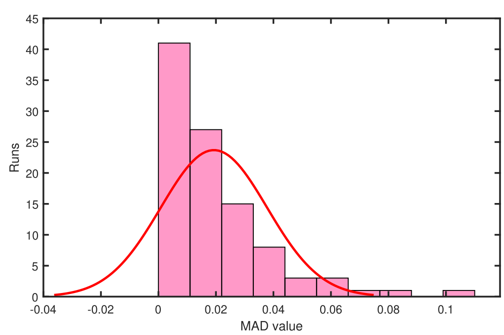
(b) Best, worst and mean errors in solutions.



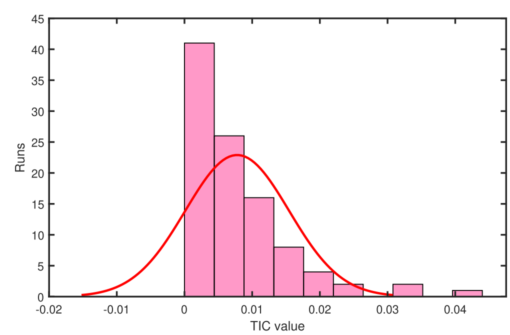
(c) Convergence of the fitness values in 100 runs.



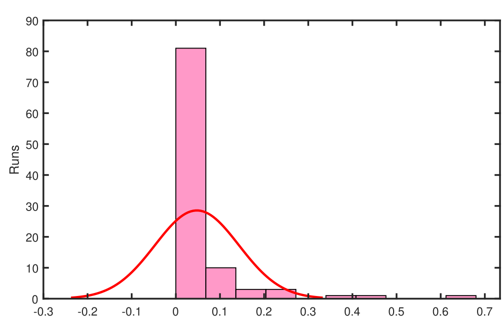
(d) Histogram of fitness values.



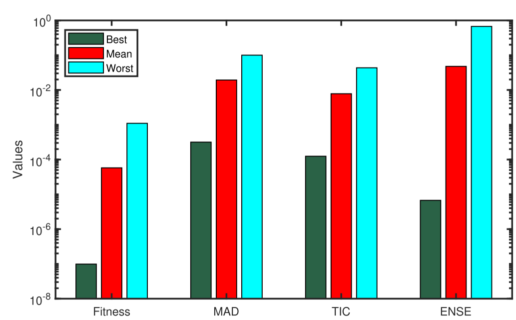
(e) Histogram of MAD values.



(f) Histogram of TIC values.



(g) Histogram of ENSE values.



(h) Best, mean and worst values of performance indicators.

Fig. 9. Results obtained by ANN-SOS for Bratu BVP with $\lambda = 3.51$.

check computational efficiency, we have compared ANN-SOS with PSO and bat algorithm as given in Table 17. Bratu problem with $\lambda = 2$ is considered to check the efficiency of the algorithms. PSO algorithm took 58.432 s and 250052 function evaluations to obtain fitness value of 1.00E-03, Bat algorithm took 45.412 s and 250050 function evaluations and reached to a fitness value of 6.11E-02 and ANN-SOS took 25.27 s and 120030 function evaluations to

Table 2
Exact and ANN-SOS solution for Bratu IVP.

x	Exact	ANN-SOS	Absolute errors
0.1	0.010016711246471	0.010016520285087	1.91E–07
0.2	0.040269546104817	0.040271993476909	2.45E–06
0.3	0.091383311852116	0.091386766303560	3.45E–06
0.4	0.164458038150111	0.164463217719673	5.18E–06
0.5	0.261168480887445	0.261176727541921	8.25E–06
0.6	0.383930338838875	0.383940466945659	1.01E–05
0.7	0.536171515135862	0.536183190285842	1.17E–05
0.8	0.722781493622688	0.722797433281814	1.59E–05
0.9	0.950884887171629	0.950903275947703	1.84E–05

Table 3
Statistical analysis of performance indicators for Bratu IVP.

	Fitness	MAD	TIC	ENSE
Best	1.65E–08	9.45E–06	5.64E–06	6.18E–09
Mean	6.06E–06	1.68E–04	9.95E–05	4.46E–06
Worst	4.19E–05	8.29E–04	4.90E–04	4.77E–05
STD	8.18E–06	1.91E–04	1.10E–04	9.81E–06

Table 4
Statistical analysis of absolute errors in solutions for Bratu IVP.

x	Min	Mean	STD
0	3.13E–07	8.94E–05	1.15E–04
0.1	1.91E–07	9.47E–05	1.20E–04
0.2	2.20E–06	9.76E–05	1.18E–04
0.3	2.26E–06	1.06E–04	1.27E–04
0.4	2.61E–06	1.26E–04	1.50E–04
0.5	8.25E–06	1.53E–04	1.76E–04
0.6	9.79E–06	1.76E–04	1.97E–04
0.7	3.72E–06	1.91E–04	2.16E–04
0.8	3.04E–06	2.19E–04	2.49E–04
0.9	1.72E–06	2.74E–04	3.09E–04
1	2.06E–06	3.24E–04	3.66E–04

Table 5
Exact and approximate solutions for Bratu BVP with $\lambda = 1$.

x	Exact	B-Spline [7]	ANN-SOS
0.1	0.0498490309241491	0.0498438	0.049847707280091
0.15	0.0708630894990087		0.0708621629772051
0.2	0.0891939678347279	0.0891845	0.089194601036884
0.25	0.104792061208615		0.104794333306934
0.3	0.1176144390042010	0.1176018	0.117617727302676
0.35	0.127625305952561		0.127628739094571
0.4	0.1347963951129060	0.1347818	0.134799204678325
0.45	0.139107283600459		0.139108995657110
0.5	0.1405456236637080	0.1405303	0.140546098551325
0.55	0.139107283600459		0.139106647199380
0.6	0.1347963951129060	0.1347818	0.134794918261672
0.65	0.127625305952561		0.127623286987649
0.7	0.1176144390042010	0.1176018	0.117612131875780
0.75	0.104792061208615		0.104789671322554
0.8	0.0891939678347279	0.0891845	0.089191712093883
0.85	0.0708630894990087		0.0708612881189368
0.9	0.0498490309241491	0.0498438	0.049848168613843

reach a fitness value of 2.15E-05 which shows that ANN-SOS is more efficient than PSO and bat algorithm.

Table 6
Absolute errors in the solutions for Bratu BVP with $\lambda = 1$.

x	B-Spline [7]	ADM [9]	ANN [56]	ANN-SOS
0.1	2.98 E–06	2.68E–03	2.75E–04	1.32E–06
0.15				9.26E–07
0.2	5.46 E–06	2.02E–03	3.29E–04	6.33E–07
0.25				2.27E–06
0.3	7.33E–06	1.52E–04	2.13E–03	3.29E–06
0.35				3.43E–06
0.4	8.50E–06	2.20E–03	1.32E–03	2.81E–06
0.45				1.71E–06
0.5	8.89E–06	3.01E–03	3.75E–04	4.75E–07
0.55				6.36E–07
0.6	8.50E–06	2.20E–03	8.63E–04	1.48E–06
0.65				2.01E–06
0.7	7.33E–06	1.52E–04	3.20E–03	2.31E–06
0.75				2.38E–06
0.8	5.46E–06	2.02E–03	1.29E–03	2.26E–06
0.85				1.80E–06
0.9	2.98E–06	2.68E–03	4.66E–06	8.62E–07

Table 7
Statistical analysis of absolute errors in solutions for Bratu BVP with $\lambda = 1$.

x	Min	Mean	STD
0	1.65E–08	2.77E–06	5.40E–06
0.1	8.53E–08	4.97E–06	6.33E–06
0.2	4.21E–07	5.71E–06	5.95E–06
0.3	1.48E–07	6.61E–06	6.03E–06
0.4	8.62E–08	8.30E–06	6.53E–06
0.5	4.75E–07	1.00E–05	7.46E–06
0.6	2.30E–07	1.02E–05	8.04E–06
0.7	7.67E–07	8.57E–06	7.35E–06
0.8	3.25E–07	6.16E–06	6.13E–06
0.9	5.92E–08	4.51E–06	5.71E–06
1	1.14E–08	3.22E–06	5.81E–06

Table 8
Statistical analysis of performance indicators for Bratu BVP with $\lambda = 1$.

	Fitness	MAD	TIC	ENSE
Best	2.29E–10	2.03E–06	6.21E–06	1.79E–08
Mean	1.58E–07	6.46E–06	1.96E–05	3.32E–07
Worst	1.46E–06	4.91E–05	1.32E–04	1.04E–05
STD	2.69E–07	5.94E–06	1.59E–05	1.11E–06

5.5. ***Problem 05:*** *System of Second Order Differential Equations with Boundary Values*

To check the efficiency of ANN-SOS for 2-dimensional differential equations, we have considered the system of differential equations [59,60]

$$y_1'' + xy_1 + xy_2 = 2, \tag{38}$$

$$y_2'' + 2xy_2 + 2xy_1 = -2, \tag{39}$$

with boundary conditions as $y_1(0) = y_1(1) = 0$ and $y_2(0) = y_2(1) = 0$. The exact solutions for the system of ODEs are $y_1 = x^2 - x$ and $y_2 = x - x^2$. We have solved the system using ANN-SOS and compared the solutions with exact solutions as given in Fig. 10. The series solutions for the system are given in Eqs. (40) and (41). The solutions obtained by ANN-SOS are very close to exact solutions which shows the efficiency of ANN-SOS algorithm.

$$\begin{aligned}\hat{y}_1(x) = {} & \frac{2.40442459290055}{1+e^{-(-2.57463864762536x-2.50249300012790)}} + \frac{0.717959958212290}{1+e^{-(4.96952889147777x+5.67482822937167)}} \\ & + \frac{-0.00819648681006459}{1+e^{-(-0.893394489795902x-0.257093990212238)}} + \frac{0.905715752252863}{1+e^{-(-0.918245684602100x-1.70722783705502)}} \\ & + \frac{-1.24989510656346}{1+e^{-(-0.109465879398657x+0.228662808771426)}} + \frac{3.12273410474153}{1+e^{-(1.84052736600937x-2.82410922658788)}} \\ & + \frac{1.73817354888999}{1+e^{-(-0.976589864644311x+0.518924754900759)}} + \frac{0.700313155192487}{1+e^{-(-0.0266996014862956x-2.09961555427663)}} \\ & + \frac{0.920130825935220}{1+e^{-(-2.53462882826823x-0.932717192696820)}} + \frac{-1.96649633452813}{1+e^{-(-2.08475347470779x+4.20794668462428)}}\end{aligned} \tag{40}$$

Table 9
Exact and approximate solutions for Bratu BVP with $\lambda = 2$.

x	Exact	B-Spline [7]	ANN-SOS
0.1	0.114415097549024	0.114393565	0.114409408571901
0.15	0.163363915345986		0.163358679573245
0.2	0.206427087404209	0.206386519	0.206420989528320
0.25	0.243346020110221		0.243337443427415
0.3	0.273890003266652	0.273834413	0.273878654436806
0.35	0.297861640828203		0.297848692592984
0.4	0.315101747408207	0.315036506	0.315088973455173
0.45	0.325493464733123		0.325482186260283
0.5	0.328965378836911	0.328896807	0.328955860889458
0.55	0.325493464733123		0.325484996904789
0.6	0.315101747408207	0.315036506	0.315093255967123
0.65	0.297861640828203		0.297852451135124
0.7	0.273890003266652	0.273834413	0.273880347831092
0.75	0.243346020110221		0.243337035757517
0.8	0.206427087404209	0.206386519	0.206420287050235
0.85	0.163363915345986		0.163360368424026
0.9	0.114415097549024	0.114393565	0.114414737859033

Table 10
Absolute errors in solution for Bratu BVP with $\lambda = 2$.

x	B-Spline [7]	ANN [56]	ANN-SOS
0.1	1.72 E−05	2.35E−03	5.69E−06
0.15			5.23E−06
0.2	3.26 E−05	1.56E−03	6.10E−06
0.25			8.57E−06
0.3	4.49E−05	3.52E−03	1.13E−05
0.35			1.29E−05
0.4	5.28E−05	4.95E−03	1.28E−05
0.45			1.12E−05
0.5	5.56E−05	4.09E−03	9.52E−06
0.55			8.46E−06
0.6	5.28E−05	5.13E−03	8.49E−06
0.65			9.18E−06
0.7	4.49E−05	3.77E−03	9.66E−06
0.75			8.98E−06
0.8	3.26E−05	1.70E−03	6.80E−06
0.85			3.54E−06
0.9	1.72E−05	1.28E−03	3.60E−07

Table 11
Statistical analysis of absolute errors in solutions for Bratu BVP with $\lambda = 2$.

x	Min	Mean	STD
0	5.37E−09	1.09E−05	2.54E−05
0.1	5.76E−07	1.78E−05	2.92E−05
0.2	1.08E−06	1.83E−05	2.45E−05
0.3	3.29E−06	1.93E−05	2.11E−05
0.4	6.77E−07	2.43E−05	2.46E−05
0.5	4.23E−06	2.92E−05	2.97E−05
0.6	3.22E−06	2.88E−05	3.03E−05
0.7	3.83E−06	2.32E−05	2.47E−05
0.8	6.60E−08	1.67E−05	1.57E−05
0.9	3.87E−08	1.24E−05	1.23E−05
1	6.34E−09	8.63E−06	1.31E−05

Table 12
Statistical analysis of performance indicators for Bratu BVP with $\lambda = 2$.

	Fitness	MAD	TIC	ENSE
Best	6.98E−09	5.77E−06	7.52E−06	2.64E−08
Mean	1.41E−06	1.91E−05	2.41E−05	6.38E−07
Worst	1.85E−05	1.40E−04	1.78E−04	1.56E−05
STD	2.95E−06	2.11E−05	2.60E−05	1.90E−06

Table 13
Exact and approximate solutions for Bratu BVP with $\lambda = 3.51$.

x	Exact	B-Spline [7]	ANN-SOS
0.1	0.364334290615029	0.357388461	0.364155174602682
0.15	0.528549826192367		0.528299150929971
0.2	0.677866771379571	0.664283874	0.677548902494709
0.25	0.809896185176297		0.809515462399274
0.3	0.922210062282960	0.902930838	0.921772041022162
0.35	1.01247780658258		1.01199129078342
0.4	1.078629288032380	1.055419782	1.078107189940600
0.45	1.11902580574817		1.11848374007722
0.5	1.132612733483520	1.107989815	1.132066255650850
0.55	1.11902580574817		1.11848854695750
0.6	1.078629288032380	1.055419782	1.078113420482010
0.65	1.01247780658258		1.01199577329851
0.7	0.922210062282960	0.902930838	0.921774890602439
0.75	0.809896185176297		0.809519231500183
0.8	0.677866771379571	0.664283874	0.677555331138821
0.85	0.528549826192367		0.528308094732500
0.9	0.364334290615029	0.357388461	0.364166335701786

Table 14
Absolute errors in solution for Bratu BVP with $\lambda = 3.51$.

x	B-Spline [7]	ANN [56]	ANN-SOS
0.1	3.84E−02	2.98E−04	1.79E−04
0.15			2.50E−04
0.2	7.48E−02	6.88E−03	3.18E−04
0.25			3.80E−04
0.3	1.06E−01	2.72E−03	4.38E−04
0.35			4.86E−04
0.4	1.27E−01	1.76E−02	5.22E−04
0.45			5.42E−04
0.5	1.35E−01	1.04E−02	5.46E−04
0.55			5.37E−04
0.6	1.27E−01	1.37E−02	5.16E−04
0.65			4.82E−04
0.7	1.06E−01	4.32E−03	4.35E−04
0.75			3.76E−04
0.8	7.48E−02	6.68E−03	3.11E−04
0.85			2.41E−04
0.9	3.84E−02	1.66E−03	1.68E−04

Table 15
Statistical analysis of absolute errors in solutions for Bratu BVP with $\lambda = 3.51$.

x	Min	Mean	STD
0	9.51E−06	1.72E−03	2.33E−03
0.1	1.79E−04	1.00E−02	1.06E−02
0.2	3.17E−04	1.93E−02	1.85E−02
0.3	4.38E−04	2.65E−02	2.52E−02
0.4	5.22E−04	3.13E−02	2.98E−02
0.5	5.46E−04	3.31E−02	3.14E−02
0.6	5.15E−04	3.13E−02	2.98E−02
0.7	4.35E−04	2.65E−02	2.53E−02
0.8	3.11E−04	1.93E−02	1.85E−02
0.9	1.67E−04	1.08E−02	1.06E−02
1	2.24E−05	1.74E−03	2.36E−03

Table 16
Statistical analysis of performance indicators for Bratu BVP with $\lambda = 3.51$.

	Fitness	MAD	TIC	ENSE
Best	9.87E−08	3.16E−04	1.24E−04	6.72E−06
Mean	5.78E−05	1.93E−02	7.80E−03	4.77E−02
Worst	1.07E−03	1.00E−01	4.34E−02	6.71E−01
STD	1.41E−04	1.85E−02	7.70E−03	9.51E−02

Table 17
Computational efficiency of algorithms.

	PSO	Bat algorithm	SOS
Fitness Value	1.00E−03	6.11E−02	**2.15E−05**
Time (seconds)	58.432	45.412	**25.27**
MFE	250051	250050	**120030**

Table 18
Solutions of bratu boundary value problem for different values of λ.

x	λ=1				λ=2				λ=3.51			
	Exact	Bat algorithm	PSO	ANN-SOS	Exact	Bat algorithm	PSO	ANN-SOS	Exact	Bat algorithm	PSO	ANN-SOS
0.1	0.0498490	0.0496347	0.0495392	0.0498477	0.1144151	0.1149388	0.1144022	0.1144094	0.3643343	0.3502382	0.3626373	0.3641552
0.2	0.0891940	0.0889685	0.0888661	0.0891946	0.2064271	0.2065309	0.2066709	0.2064210	0.6778668	0.6527166	0.6748154	0.6775489
0.3	0.1176144	0.1174020	0.1172367	0.1176177	0.2738900	0.2733599	0.2744122	0.2738787	0.9222101	0.8877109	0.9179894	0.9217720
0.4	0.1347964	0.1346911	0.1343954	0.1347992	0.3151017	0.3137300	0.3158092	0.3150890	1.0786293	1.0377094	1.0736321	1.0781072
0.5	0.1405456	0.1405570	0.1401747	0.1405461	0.3289654	0.3266813	0.3297693	0.3289559	1.1326127	1.0893993	1.1273518	1.1320663
0.6	0.1347964	0.1348358	0.1345018	0.1347949	0.3151017	0.3120234	0.3159897	0.3150933	1.0786293	1.0376901	1.0736493	1.0781134
0.7	0.1176144	0.1176614	0.1174147	0.1176121	0.2738900	0.2702861	0.2749084	0.2738803	0.9222101	0.8875961	0.9180212	0.9217749
0.8	0.0891940	0.0893255	0.0890739	0.0891917	0.2064271	0.2026107	0.2076089	0.2064203	0.6778668	0.6525708	0.6748449	0.6775553
0.9	0.0498490	0.0500267	0.0497734	0.0498482	0.1144151	0.1106154	0.1157219	0.1144147	0.3643343	0.3501246	0.3626863	0.3641663

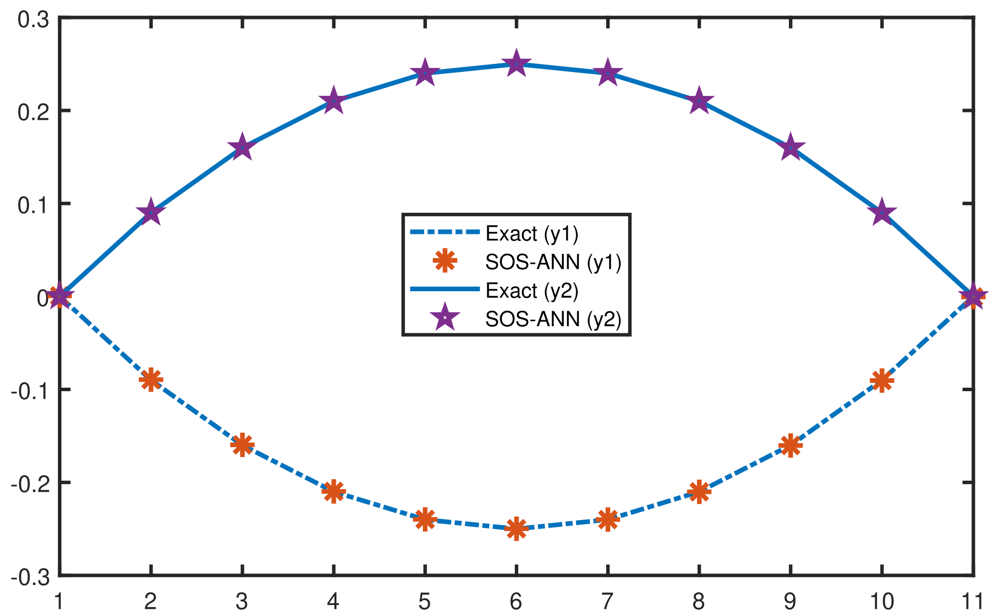


Fig. 10. Solution of the system of ODEs by ANN-SOS.

$$\hat{y}_2(x) = \frac{0.890948666418945}{1+e^{-(2.42934408305951x+2.76119652877121)}} + \frac{-0.165999663029983}{1+e^{-(4.88301177052512x-0.216151040866744)}} + \frac{-1.37143236035590}{1+e^{-(-0.826421353507985x+1.24656446486019)}} + \frac{1.53436751646672}{1+e^{-(0.0145214842427664x-2.01142009223707)}} + \frac{-9.43614706241816}{1+e^{-(1.43616159135335x-3.37978778820565)}} + \frac{-1.38649156682681}{1+e^{-(-2.39258020099330x-1.79390514570881)}} + \frac{0.392521293013865}{1+e^{-(1.66267761553255x+4.79525886880030)}} + \frac{0.804200664217380}{1+e^{-(3.21988604521199x+0.220411171477678)}} + \frac{8.03278897172594}{1+e^{-(1.16352913511213x-9.46465168600412)}} + \frac{-0.790961355694097}{1+e^{-(-1.70900217722274x-1.03455309952375)}} \quad (41)$$

Table 19
Sensitivity analysis of number of neurons in ANN.

x	Exact solution	ANN-SOS (3 neurons)	ANN-SOS (5 neurons)	ANN-SOS (10 neurons)
0.1	0.049849031	0.050207593	0.049794863	0.049847707
0.2	0.089193968	0.089578970	0.089149854	0.089194601
0.3	0.117614439	0.118011800	0.117582525	0.117617727
0.4	0.134796395	0.135192175	0.134765132	0.134799205
0.5	0.140545624	0.140941817	0.140505235	0.140546099
0.6	0.134796395	0.135208751	0.134750872	0.134794918
0.7	0.117614439	0.118056090	0.117578979	0.117612132
0.8	0.089193968	0.089657566	0.089180302	0.089191712
0.9	0.049849031	0.050302643	0.049852065	0.049848169

Table 20
Sensitivity analysis of ecosize in SOS algorithm.

x	Exact solution	ANN-SOS (ecosize = 20)	ANN-SOS (ecosize = 30)	ANN-SOS (ecosize = 50)
0.1	0.049849031	0.049868321	0.049851747	0.049847707
0.2	0.089193968	0.089200098	0.089193212	0.089194601
0.3	0.117614439	0.117610433	0.117609914	0.117617727
0.4	0.134796395	0.134778838	0.134789408	0.134799205
0.5	0.140545624	0.140510760	0.140537916	0.140546099
0.6	0.134796395	0.134743798	0.134787566	0.134794918
0.7	0.117614439	0.117547134	0.117603494	0.117612132
0.8	0.089193968	0.089116286	0.089182874	0.089191712
0.9	0.049849031	0.049764159	0.049839463	0.049848169

6. Sensitivity analysis

Sensitivity analysis of number of neurons in ANN model is given in Table 19. The table shows that when the number of neurons increases in ANN, the solution is getting better. The sensitivity analysis in terms of ecosize in SOS algorithm is given in Table 20. This table shows that the accuracy in the solutions obtained by ANN-SOS increases as ecosize increases.

7. Conclusion

We have implemented the ANN-SOS algorithm to solve one-dimensional Bratu initial and boundary value problems with different values of the parameter λ. The results obtained by the ANN-SOS algorithm are compared with other techniques. The fitness value obtained by ANN-SOS algorithm for Bratu's initial value problem is

1.6492×10^{-8} and comparison of our result with the exact solution is given in Fig. 3a. For the Bratu BVP with $\lambda = 1$, the fitness value obtained by the ANN-SOS algorithm is 2.2883×10^{-10} which is much better than ANN-based gradient descent algorithm, L-M method and conjugate gradient algorithm with fitness values $3.20 \times 10^{-3}, 1.39 \times 10^{-4}$ and 3.95×10^{-3} respectively. Table 6 shows that the ANN-SOS algorithm successfully calculated better results than analytical methods, the B-Spline method and ADM. The results for Bratu BVP with $\lambda = 2$ are given in Table 10, and it is clear from the table that the ANN-SOS algorithm shows better results than other methods. In Table 14, the absolute errors obtained by B-Spline, ANN and ANN-SOS algorithm are reported. It is evident that ANN-SOS algorithm produced accurate results. ANN-SOS algorithm quickly solved the nonlinear Bratu IVP and BVPs. It can solve higher dimensional Bratu problems and other highly nonlinear differential equations. The ANN-SOS algorithm is efficient and robust, which can be used to solve other real application problems.

Declaration of Competing Interest

None.

Acknowledgment

This project was funded by the Deanship of Scientific Research, King Abdulaziz University, Jeddah, under Grant No. (DF-232–135-1441). The author, therefore, gratefully acknowledge DSR technical and financial supports.

References

[1] Boyd JP. One-point pseudospectral collocation for the one-dimensional Bratu equation. Appl Math Comput 2011;217:5553–65.
[2] Jacobsen J, Schmitt K. The Liouville–Bratu–Gelfand problem for radial operators. J Differential Eqs 2002;184:283–98.
[3] Makinde OD. Exothermic explosions in a slab: a case study of series summation technique. Int Commun Heat Mass Transf 2004;31:1227–32.
[4] Makinde O. Strongly exothermic explosions in a cylindrical pipe: a case study of series summation technique. Mech Res Commun 2005;32:191–5.
[5] Makinde O, Osalusi E. Exothermic explosions in symmetric geometries: an exploitation of perturbation technique. Rom J Phys 2005;50:621.
[6] Raja MAZ. Solution of the one-dimensional Bratu equation arising in the fuel ignition model using ANN optimised with PSO and SQP. Connect Sci 2014;26:195–214.
[7] Caglar H, Caglar N, Özer M, Valarıstos A, Anagnostopoulos AN. B-spline method for solving Bratu's problem. Int J Comput Math 2010;87:1885–91.
[8] Buckmire R. Application of a Mickens finite-difference scheme to the cylindrical Bratu-Gelfand problem. Numer Methods Partial Differential Eqs: An Int J 2004;20:327–37.
[9] Wazwaz AM. Adomian decomposition method for a reliable treatment of the Bratu-type equations. Appl Math Comput 2005;166:652–63.
[10] Chang SH, Chang IL. A new algorithm for calculating one-dimensional differential transform of nonlinear functions. Appl Math Comput 2008;195:799–808.
[11] Aregbesola Y. Numerical solution of Bratu problem using the method of weighted residual; 2003.
[12] Boyd JP. Chebyshev polynomial expansions for simultaneous approximation of two branches of a function with application to the one-dimensional Bratu equation. Appl Math Comput 2003;143:189–200.
[13] Raja MAZ et al. Numerical treatment for solving one-dimensional Bratu problem using neural networks. Neural Comput Appl 2014;24:549–61.
[14] Sivakumar P, Jayaraman J. Efficient two-step fifth-order and its higher-order algorithms for solving nonlinear systems with applications. Axioms 2019;8:37.
[15] Shirvany Y, Hayati M, Moradian R. Numerical solution of the nonlinear Schrodinger equation by feedforward neural networks. Commun Nonlinear Sci Numer Simul 2008;13:2132–45.
[16] Shirvany Y, Hayati M, Moradian R. Multilayer perceptron neural networks with novel unsupervised training method for numerical solution of the partial differential equations. Appl Soft Comput 2009;9:20–9.
[17] Kumar M, Yadav N. Multilayer perceptrons and radial basis function neural network methods for the solution of differential equations: a survey. Comput Math Appl 2011;62:3796–811.
[18] McFall KS, Mahan JR. Artificial neural network method for solution of boundary value problems with exact satisfaction of arbitrary boundary conditions. IEEE Trans Neural Networks 2009;20:1221–33.
[19] Guo H, Zhuang X, Rabczuk T. A deep collocation method for the bending analysis of Kirchhoff plate. Comput Mater Continua 2019;59:433–56.
[20] Anitescu C, Atroshchenko E, Alajlan N, Rabczuk T. Artificial neural network methods for the solution of second order boundary value problems. Comput, Mater Continua 2019;59:345–59.
[21] Samaniego E, Anitescu C, Goswami S, Nguyen-Thanh VM, Guo H, Hamdia K, Zhuang X, Rabczuk T. An energy approach to the solution of partial differential equations in computational mechanics via machine learning: Concepts, implementation and applications. Comput Methods Appl Mech Eng 2020;362:112790.
[22] Seadawy AR, Manafian J. New soliton solution to the longitudinal wave equation in a magneto-electro-elastic circular rod. Results Phys 2018;8:1158–67.
[23] Selima ES, Seadawy AR, Yao X. The nonlinear dispersive Davey-Stewartson system for surface waves propagation in shallow water and its stability. Eur Phys J Plus 2016;131:1–16.
[24] Seadawy AR, El-Rashidy K. Dispersive solitary wave solutions of Kadomtsev-Petviashvili and modified Kadomtsev-Petviashvili dynamical equations in unmagnetized dust plasma. Results Phys 2018;8:1216–22.
[25] Seadawy AR, Jun W, et al. Mathematical methods and solitary wave solutions of three-dimensional Zakharov-Kuznetsov-Burgers equation in dusty plasma and its applications. Results Phys 2017;7:4269–77.
[26] Arnous AH, Seadawy AR, Alqahtani RT, Biswas A. Optical solitons with complex Ginzburg-Landau equation by modified simple equation method. Optik 2017;144:475–80.
[27] Seadawy AR. Two-dimensional interaction of a shear flow with a free surface in a stratified fluid and its solitary-wave solutions via mathematical methods. Eur Phys J Plus 2017;132:518.
[28] Seadawy AR. Solitary wave solutions of two-dimensional nonlinear Kadomtsev-Petviashvili dynamic equation in dust-acoustic plasmas. Pramana 2017;89:49.
[29] Seadawy AR, Alamri SZ. Mathematical methods via the nonlinear two-dimensional water waves of Olver dynamical equation and its exact solitary wave solutions. Results Phys 2018;8:286–91.
[30] Sulaiman M, Salhi A, Selamoglu BI, Kirikchi OB. A plant propagation algorithm for constrained engineering optimisation problems. Math Problems Eng 2014.
[31] Sulaiman M, Salhi A. A seed-based plant propagation algorithm: the feeding station model. Scient World J 2015;2015.
[32] Mashwani WK, Salhi A, Jan MA, Khanum RA, Sulaiman M. Impact analysis of crossovers in a multi-objective evolutionary algorithm. Sci Int 2015;27:4943–56.
[33] Khan W, Salhi A, Asif M, Adeeb R, Sulaiman M. Enhanced version of multi-algorithm genetically adaptive for multiobjective optimization. Int J Adv Comput Sci Appl 2015;6:279–87.
[34] Sulaiman M, Salhi A, Fraga ES, Mashwani WK, Rashidi MM. A novel plant propagation algorithm: modifications and implementation. Sci Int 2016;28:201–9.
[35] Sulaiman M, Salhi A, Khan A, Muhammad S, Khan W. On the theoretical analysis of the plant propagation algorithms. Math Probl Eng 2018;2018.
[36] Sulaiman M, Sulaman M, Hamdi A, Hussain ZH. The plant propagation algorithm for the optimal operation of directional over-current relays in electrical engineering. Mehran Univ Res J Eng Technol 2020;39:223–36.
[37] Mashwani WK, Zaib A, Yeniay Ö, Shah H, Tairan NM, Sulaiman M. Hybrid Constrained Evolutionary Algorithm for Numerical Optimization Problems. Hacettepe J Math Stat 2018;48:931–50.
[38] Javed H, Jan MA, Tairan N, Mashwani WK, Khanum RA, Sulaiman M, Khan HU, Shah H. On the Efficacy of Ensemble of Constraint Handling Techniques in Self-Adaptive Differential Evolution. Mathematics 2019;7:635.
[39] Sulaiman M, Ahmad S, Iqbal J, Khan A, Khan R. Optimal operation of the hybrid electricity generation system using multiverse optimization algorithm. Comput Intell Neurosci 2019;2019.
[40] Sulaiman M, Masihullah M, Hussain Z, Ahmad S, Mashwani WK, Jan MA, Khanum RA. Implementation of improved grasshopper optimization algorithm to solve economic load dispatch problems. Hacettepe J Math Stat 2019;48:1570–89.
[41] Sulaiman M, Muhammad S, Khan A, et al. Improved solutions for the optimal coordination of docrs using firefly algorithm. Complexity 2018;2018.
[42] Sulaiman M, Ahmad A, Khan A, Muhammad S. Hybridized symbiotic organism search algorithm for the optimal operation of directional overcurrent relays. Complexity 2018;2018.
[43] Khanum RA, Jan MA, Tairan N, Mashwani WK, Sulaiman M, Khan HU, Shah H. Global evolution commended by localized search for unconstrained single objective optimization. Processes 2019;7:362.
[44] Sulaiman M, Samiullah I, Hamdi A, Hussain Z. An improved whale optimization algorithm for solving multi-objective design optimization problem of PFHE. J Intell Fuzzy Syst 2019;37:3815–28.
[45] Ahmad A, Sulaiman M, Alhindi A, Aljohani AJ. Analysis of temperature profiles in longitudinal fin designs by a novel neuroevolutionary approach. IEEE Access 2020;8:113285–308.
[46] Waseem W, Sulaiman M, Islam S, Kumam P, Nawaz R, Raja MAZ, Farooq M, Shoaib M. A study of changes in temperature profile of porous fin model using cuckoo search algorithm. Alexandria Eng J 2020;59:11–24.
[47] Bukhari AH, Raja MAZ, Sulaiman M, Islam S, Shoaib M, Kumam P. Fractional neuro-sequential ARFIMA-LSTM for financial market forecasting. IEEE Access 2020;8:71326–38.

[48] Waseem W, Sulaiman M, Alhindi A, Alhakami H. A Soft Computing Approach Based on Fractional Order DPSO Algorithm Designed to Solve the Corneal Model for Eye Surgery. IEEE Access 2020;8:61576–92.
[49] Ahmad S, Sulaiman M, Kumam P, Hussain Z, Asif Jan M, Mashwani WK, et al.A novel population initialization strategy for accelerating Levy flights based multi-verse optimizer. J Intell Fuzzy Syst, pp. 1–17.
[50] Bukhari AH, Sulaiman M, Raja MAZ, Islam S, Shoaib M, Kumam P. Design of a hybrid NAR-RBFs neural network for nonlinear dusty plasma system. Alexandria Eng J 2020.
[51] Khan A, Sulaiman M, Alhakami H, Alhindi A. Analysis of Oscillatory Behavior of Heart by Using a Novel Neuroevolutionary Approach. IEEE Access 2020;8:86674–95.
[52] Waseem W, Sulaiman M, Kumam P, Shoaib M, Raja MAZ, Islam S. Investigation of singular ordinary differential equations by a neuroevolutionary approach. Plos one 2020;15:e0235829.
[53] Khan NA, Sulaiman M, Aljohani AJ, Kumam P, Alrabaiah H. Analysis of multi-phase flow through porous media for imbibition phenomena by using the LeNN-WOA-NM algorithm. IEEE Access 2020.
[54] Hassan A, Kamran M, Illahi A, Zahoor RMA, et al. Design of cascade artificial neural networks optimized with the memetic computing paradigm for solving the nonlinear Bratu system. Eur Phys J Plus 2019;134:122.
[55] Cheng MY, Prayogo D. Symbiotic organisms search: a new metaheuristic optimization algorithm. Comput Struct 2014;139:98–112.
[56] Kumar M, Yadav N. Numerical solution of Bratu's problem using multilayer perceptron neural network method. Natl Acad Sci Lett 2015;38:425–8.
[57] Yang XS. A new metaheuristic bat-inspired algorithm. In: Nature inspired cooperative strategies for optimization (NICSO 2010). Springer; 2010. p. 65–74.
[58] Kennedy J, Eberhart R. Particle swarm optimization. In: Proceedings of ICNN'95-International Conference on Neural Networks. IEEE; 1995, Vol. 4, pp. 1942–8.
[59] Öztürk Y. Numerical solution of systems of differential equations using operational matrix method with Chebyshev polynomials. J Taibah Univ Sci 2018;12:155–62.
[60] Shiralashetti S, Kumbinarasaiah S. Laguerre Wavelets Exact Parseval Frame-based Numerical Method for the Solution of System of Differential Equations. Int J Appl Comput Math 2020;6:1–16.
[61] Mashwani Wali Khan, Salhi Abdellah, jan Muhammad Asif, Sulaiman Muhammad, Khanum Rashida Adeeb, Algarni Abdulmohsen. Evolutionary Algorithms Based on Decomposition and Indicator Functions: State-of-the-art Survey. International Journal of Advanced Computer Science and Applications 2016;7(2):583–93. doi: https://doi.org/10.14569/IJACSA.2016.070274.

Ashfaq Ahmadgot his B.S degree in mathematics from Islamia College Peshawar in 2015, MPhil in Mathematics from Abdul Wali Khan University Mardan Pakistan in 2018. He is currently a Ph.D. student of Mathematics at the Abdul Wali Khan University. His research interests include Optimization algorithms, real world problems, Bio-inspired Algorithms, artificial neural networks, and energy management.

Muhammad Sulaiman received his B.Sc., from the University of Peshawar in 2004, M.Sc., and MPhil degrees in mathematics from the Quaid-e-Azam University Islamabad Pakistan, in 2007, and 2009 respectively. He got his Ph.D. degree in mathematics from the University of Essex, UK, in 2015. From 2009 to 2016, he was a Lecturer in mathematics with the Abdul Wali Khan University Mardan, Pakistan. Since Feb 2016, he has been an Assistant Professor with the Department of Mathematics, Abdul Wali Khan University Mardan, Pakistan. He is the author of two book chapters, and more than 25 research articles. His research interests include mathematical optimization techniques, global optimization, and evolutionary algorithms, Heuristics, Metaheuristics, Multi-objective Optimization, Design Engineering Optimization Problems, Structural Engineering Optimization Problems, Linear Programming, Linear and non-linear Least Squares Optimization Problems, Evolutionary Algorithms, Nature-Inspired Metaheuristics, Artificial neural networks, and differential equations. He is an Associate Editor of the journal COJ Reviews and Research, and SCIREA Journal of Mathematics.

Dr. Abdulah Jeza Aljohani received the B.Sc (Eng.) degree in electronics and communication engineering from King Abdulaziz University, Jeddah, Saudi Arabia, in 2006, and the M.Sc. degree with distinction and Ph.D. degree, awarded with no corrections, in wireless communication from the University of Southampton, Southampton, U.K., in 2010 and 2016, respectively. He is currently an Assistance Professor with the Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah, Saudi Arabia. He is also associated with the Center of Excellence in Intelligent Engineering Systems. His research interests include, convex optimization, joint source/channel coding, distributed source coding, IoT, channel coding, cooperative communications, and MIMO systems.

Ahmad Alhindi received the B.Sc. degree in computer science from Umm al-Qura University (UQU), Makkah, Saudi Arabia, in 2006, and the M.Sc. degree in computer science and the Ph.D. degree in computing and electronic systems from the University of Essex, Colchester, U.K., in 2010 and 2015, respectively. He is currently an Assistant Professor of artificial intelligence (AI) with the Computer Science Department and a researcher in CIADA, UQU. His current research interests include evolutionary multi-objective optimization and machine learning techniques. He is currently involved in AI algorithms, focusing particularly on machine learning and optimization with a willingness to implement them in the context of decision making and solving combinatorial problems in real-world projects.