Sigma J Eng Nat Sci, Vol. 44, No. 5, pp. 3270–3283, October, 2026

Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2025.00145

Research Article

Neurocomputing solution of graphene nano/microelectromechanical system using a wavelet-based neural network

Muhammad AMİR[1], Jamshaid UL RAHMAN[1,*]

[1]*Abdus Salam School of Mathematical Sciences, Government College University, Lahore, 54600, Pakistan*

ARTICLE INFO

Article history
Received: 19 February 2025
Revised: 01 June 2025
Accepted: 10 July 2025

Keywords:
Graphene Microelectromechanical Model; Morlet Wavelet-Based Neural Network; Neural Network Algorithm; Particle Swarm Optimization; Statistical Operators

ABSTRACT

Nano/microelectromechanical systems based on graphene have attracted considerable attention because of their potential applications in high-performance sensors, resonators, and actuators. However, accurately approximating their dynamic responses remains challenging due to the strong nonlinearities governing their behavior. To address these challenges, this study proposes a hybrid computational framework that integrates a Morlet wavelet neural network with particle swarm optimization and neural network algorithm for the optimization of network weights and biases. In the proposed approach, the proposed technique is employed to approximate the nonlinear system dynamics, while particle swarm optimization optimizes the network parameters by minimizing a problem-specific objective function. The effectiveness of the framework is evaluated using three graphene-based models. Its predictive accuracy and convergence behavior are evaluated through statistical performance measures, including Theil's inequality coefficient, mean squared error and mean absolute deviation. The obtained error values range from 10^{-3} to 10^{-8}, demonstrating the method's strong approximation capability and favourable convergence characteristics. The results indicate that the proposed framework can reliably capture the complex nonlinear dynamics of graphene-based systems, offering a promising computational tool for the analysis, design, and control of advanced microelectromechanical devices. The main contribution of this work lies in combining the approximation capabilities of Morlet wavelet neural networks with the global search capability of particle swarm optimization. This integration provides an effective alternative to conventional analytical approaches and standard neural network models, with the potential to improve prediction accuracy and parameter-learning efficiency in complex nanoscale systems.

Cite this article as: Amir M, Ul Rahman J. Neurocomputing solution of graphene nano/microelectromechanical system using a wavelet-based neural network. Sigma J Eng Nat Sci 2026;44(5):3270–3283.

INTRODUCTION

The field of Nano/Microelectromechanical(N/MEMS) systems has seen extensive research due to its potential applications in nano or microdevices, including switches, actuators, sensors, and bridges [1-3]. Carbon-based nanomaterials with their distinct mechanical, electrical, optical,

***Corresponding author.**
*E-mail address: jamshaid@sms.edu.pk

This paper was recommended for publication in revised form by Editor-in-Chief Ahmet Selim Dalkilic

Published by Yıldız Technical University Press, İstanbul, Turkey
© Author. This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

and chemical capabilities make them more appealing for applications including electronic materials, supercapacitors, and sensors [4-6]. Carbon nanotubes and graphene have unique mechanical characteristics such as high Young's modulus, low mass, and high thermal conductivity. Moreover, graphene can be employed as the basis for detecting frequency shifts caused by mass changes resulting from the absorption of gases in gas-sensing applications. Consequently, it is imperative to investigate the nonlinear dynamic behavior of graphene, a remarkable material that deserves to be regarded as an excellent choice for electrically actuated N/MEMS systems devices. The dynamic pull-in instability, which is dependent on certain parameters, is a useful phenomenon associated with N/MEMS systems devices. Analyzing pull-in instability is essential for these devices' dependable and effective functions. Besides pull-in analysis, another difficult issue is examining the vibratory behavior of N/MEMS due to these systems containing total energy equal to zero. The existence of strong and dominating nonlinearities is a significant additional challenge for the researchers, and ignoring these nonlinearities may result in inaccurate estimation. These nonlinearities are mostly caused by forces involved in the actuation and sensing processes. Van der Waals (VDW) attraction, Casimir force, [7, 8] electrostatic force, [9, 10] electromagnetic force, and others are the primary forces operating on N/MEMS [11, 12].

There are several analytical techniques to solve the non-linear N/MEMS model specifically such as the Homotopy perturbation technique [13, 14] is used to solve the electrically actuated MEMS system, The variational iteration approach [15, 16] investigated the nonlinear vibration in MEMS systems due to electromagnetic force at zero initial conditions, Energy balance method [17] is utilized to find the solution of nonlinear oscillators, Frequency formulation [18] analyzed the periodic solution of fractional oscillators and many more. Researchers utilized different techniques to optimize and analyze the numerous engineering problems [19-24]. In recent years, researchers have significantly focused on using machine learning approaches to solve challenging problems in a variety of engineering and science domains. Machine learning techniques outperform traditional analytical and numerical methods in tackling intricate problems involving nonlinear systems of differential equations. Recently, researchers investigated into solving nonlinear differential equations with machine learning approaches like artificial neural networks (ANNs). Non-linear differential equations can be approximated with high accuracy using artificial neural networks [25-27]. A widely used approach involves training the network to predict the solution function using a dataset of input-output pairs. While the outputs correspond to points on the solution function, the inputs represent points in the differential equation's domain. Training allows a network to learn to approximate solutions for all points in its domain. Furthermore, ANNs can simulate a system's behavior over time by learning its dynamics from data sets. This approach works especially well with differential equations that are difficult to solve analytically. Yang et al. [28] proposed a method for finding the solution of differential equations (DEs) with neural networks and optimization algorithms referred as physics-informed generative adversarial networks.

Mattheakis et al. [29] find a solution for DEs that demonstrates the dynamic properties of the system. An ANN-based loss function only uses the network's predictions rather than the output data. Raissi et al. [30] developed the idea of Physics informed neural networks (PINN), which can learn how to perform effective supervised learning challenges. Umar et al. [31] solved the predator-prey model using the MWNN technique combined with a genetic algorithm and sequential quadratic programming. The layer structure of feed-forward ANNs employing morlet wavelet activation function [32] for predicting the solution of the pantograph nonlinear equation. A novel technique of Morlet wavelet neural network [33, 34] is used to solve the non-linear models. Mattheakis et al. [35] used an ANN to solve a challenging system of nonlinear differential equations while obeying energy conservation rules. Furthermore, they ensured that the network's predictions adhere to energy conservation by constructing a symplectic neural network utilizing the hub neuron paradigm. A specific technique for finding entirely differentiable solutions was demonstrated by Piscopo et al. [36]. A surprising fact they found when examining different network designs was that even very small networks produced outstanding results. A novel method that effectively estimates unexpected functions in differential equations without requiring the solution of sensitivity equations has been studied by Hagge et al. [37].

Traditional analytical, numerical methodologies have been used to investigate these systems. However, the traditional techniques are either inadequate to model the complex nonlinear oscillations or have poor convergence and generalization. The nonlinear dynamic behavior of graphene N/MEMS is often hard to approximate with accuracy and effectiveness using suggested models. Lack of strong computational frameworks that can guarantee convergence, stability and good predictive accuracy for a broad class of N/MEMS systems by means of hybrid schemes. To be able to solve the complex nonlinear dynamics in graphene-based systems, a strong and efficient modelling algorithms are needed. For this, we designed a novel hybrid approach of the Morlet wavelet neural network hybrid with particle swarm optimization and neural network algorithm. The novelty of the proposed research emphasizes its importance, as the proposed fitness function for analysing and modelling of the complex nanotechnology system is constructed for the first time, which is based on morlet wavelet neural network. Using a number of N/MEMS validation case studies, the proposed technique has been shown to

have high agreement with the reference solutions and low error metrics.

The proposed MWNN-PSO-NNA framework has the following key contributions and distinguishing features:

- A nonlinear graphene-based nano/microelectromechanical system (N/MEMS) accurate and computing efficient solution is developed using Morlet wavelet neural network (MWNN) architecture.
- By incorporating the Particle Swarm Optimization (PSO) algorithm and refining the parameters afterwards using a neural network, the predictive accuracy and stability of the neural network is enhanced.
- The predictions of the proposed MWNN-PSO-NNA framework are compared to the corresponding exact or numerical reference solutions, if available, to assess the accuracy and reliability of the proposed framework.
- The approach is tested with a well-known statistical error measure, the mean squared error (MSE) and Theil's inequality coefficient (TIC), which quantify the agreement between the predicted and reference solutions.
- The results of the optimization are analyzed to investigate the repeatability of the optimization and to understand the robustness of the proposed approach to the variation in the initializations with 20 independent runs.

The remaining part of the paper is presented as follows: The model formulation of the graphene-based N/MEMS system is illustrated in Section 2.0. The designed methodology of the morlet wavelet neural network, optimization algorithm, and statistical measure is shown in Section 3.0. The result and discussion of the proposed technique are demonstrated in Section 4.0. The concluding remarks are presented in section 5.0.

Mathematical Formulation of the Nonlinear MEMS System

Consider an electrostatically actuated system consisting of two parallel plates. A graphene strip of area *A*' connects the movable plate to the fixed base, with an initial separation of h. As illustrated in Figure 1, the movable plate has an area A and a mass m. The constitutive relation for graphene is adopted from the studies of Lee [38] and Lu and Huang [39].

$$\sigma = E\varepsilon - D\varepsilon|\varepsilon|, \tag{1}$$

Here, the symbols σ, ε, E and D represent axial stress, axial strain, Young's modulus, and the elastic stiffness constant, respectively. The movable plate is subjected to two forces: the electrostatic force generated by the applied voltage V and the restoring force produced by the axial deformation of the graphene strip. Based on Equation (1), these forces can be expressed as follows.

$$F_r = -A'E\frac{\hat{x}}{l} + A'D\frac{\hat{x}}{l}\left|\frac{\hat{x}}{l}\right| \tag{2}$$

$$F_e = \frac{\varepsilon_o AV^2}{2(h-\hat{x})^2} \tag{3}$$

Where ε_o is the electric emissivity. The equation of motion of the movable part can be formulated using Newton's second law of motion, and is therefore a nonlinear differential equation.

$$m\frac{d^2\hat{x}}{d\hat{t}^2} + A'E\frac{\hat{x}}{l} - A'D\frac{\hat{x}}{l}\left|\frac{\hat{x}}{l}\right| = \frac{\varepsilon_o AV^2}{2(h-\hat{x})^2} \tag{4}$$

The vertical displacement whose name is $\hat{x}$, with respect to the fixed and movable parts is called $\hat{t}$. The variables that are independent of units are introduced as:

$$x = \frac{\hat{x}}{h}, t = \hat{t}\sqrt{\frac{A'E}{ml}}, \alpha = \frac{hD}{lE}, k = \frac{\varepsilon_o AlV^2}{2EA'h^3}$$

The equation of motion in dimensionless form is given by,

$$\frac{d^2x}{dt^2} + x - \alpha x|x| = \frac{k}{(1-x)^2} \tag{5}$$

$$x(0) = a, \qquad \left.\frac{dx}{dt}\right|_{t=0} = 0 \tag{6}$$

So, by using the Taylor series, Equation (5) can be represented as:

$$\frac{d^2x}{dt^2} + x - \alpha x^2 - k(1 + 2x + 3x^2 + 4x^3) \tag{7}$$

DESIGNED METHODOLOGY

The proposed computational method for nonlinear graphene nano/microelectromechanical (N/MEMS) systems involves the use of MWNN and PSO-NNA optimization. The Morlet wavelet is a widely used wavelet in neural networks, especially in time-frequency analysis. The idea in this approach is to extend the capability of the network to approximate the complex nonlinear behavior with the help of Morlet wavelet functions. During the network training, wavelet coefficients are used to enhance the accuracy. The weights and biases of the network are optimized using PSO-NNA optimization. The proposed scheme intends to reduce computational errors and boost the stability. Evaluated by means of numerical simulations and comparison with the available methods.

Design Morlet Wavelet Neural Networks

The ability of ANN-based models to give consistent solution is evolving in various applications. The approximate solution of the mathematical model of the graphene model is expressed by means of MWNNs in equation (7)

and the nth-order derivatives of the mathematical model are provided below,

$$\widehat{\omega}(t) = \sum_{j=1}^{m} \phi_j \, G(\delta_j t + \gamma_i) \tag{8}$$

$$\widehat{\omega}^{(n)}(t) = \sum_{j=1}^{m} \phi_j \, G^{(n)}(\delta_j t + \gamma_i) \tag{9}$$

For a network consisting of m neurons, [ϕ, δ, γ] the expression represents a single group of neurons associated with the unknown weights. Here, W is defined as $W = [\phi, \delta, \gamma]$.

$$\begin{gathered} \phi = [\phi_1, \phi_2, \phi_3, \ldots \phi_m],\ \delta = [\delta_1, \delta_2, \delta_3, \ldots \delta_m] \\ and\ \gamma = [\gamma_1, \gamma_2, \gamma_3, \ldots \gamma_m] \end{gathered} \tag{10}$$

The suggested function is the product of Cost and the Gaussian function is written as:

$$\hat{G}(t) = \sum_{j=1}^{m} \cos(1.75t)\, e^{0.5t^2}, \tag{11}$$

Substituting the Morlet activation function into Equation (10) yields the following expression.

$$\widehat{\omega}(t) = \sum_{j=1}^{m} \phi_j \cos\left(1.75(\delta_j t + \gamma_j\right) e^{0.5(\delta_j t + \gamma_j)^2} \tag{12}$$

$$\widehat{\omega}'(t) = \sum_{j=1}^{m} -\phi_j \delta_j e^{-0.5(\delta_j t+\gamma_j)^2} \left(\sin\left\{\frac{7}{4}(\delta_j t + \gamma_i\right\} + (\delta_j t + \gamma_j) cos\left\{\frac{7}{4}(\delta_j t + \gamma_j)\right\}\right),$$

$$\widehat{\omega}''(t) = \sum_{j=1}^{m} -\phi_j \delta_j^2 e^{-0.5(w_j t+\gamma_i)^2} \begin{pmatrix} -3.0625 cos\left\{\frac{7}{4}(\delta_j t + \gamma_j)\right\} \\ +\frac{7}{2}(\delta_j t + \gamma_j) \sin\left\{\frac{7}{4}(\delta_{u_i} t + a_{u_i})\right\} \\ +\{-1 + (\delta_j t + \gamma_j)^2 cos\left\{\frac{7}{4}(\delta_j t + \gamma_j)\right\} \end{pmatrix}$$

The fitness function can be written as,

$$E_F = E_{F1} + E_{F2}, \tag{13}$$

Here, E_{F1} and E_{F1} denote denote the fitness functions associated with the graphene-based nano/microelectromechanical system (N/MEMS), as defined below.

$$\begin{aligned} E_F = \frac{1}{N} \sum_{j=1}^{m} \left(\frac{d^2 \hat{u}}{dt_j^2} + \hat{u} - \alpha \hat{u} |\hat{u}| = \frac{k}{(1-\hat{u})^2} \right)^2 \\ + \frac{1}{2} \left((\hat{u}_0 - a)^2 + \left(\frac{d\hat{u}_0}{dt} \right)^2 \right) \end{aligned} \tag{14}$$

The fitness error functions for the governing differential equations of the graphene-based system are denoted by E_{F1} and E_{F2}. The function E_{F2} for the initial conditions, while û represents the estimated solution of the graphene system. As indicated by Equation (10), the optimal weights are obtained by minimizing the fitness error function toward zero ($E_{F2} \rightarrow 0$). Consequently, the estimated solution approaches the exact solution, such that (û(t)→ $u(t)$).

Particle Swarm Optimization

A msignificant algorithm that was inspired by bird structure in swarms is Particle swarm optimization. The main aim is to study intricate non-linear problems that are challenging to solve using conventional techniques. In the PSO, the individual and global optimal solutions are used iteratively to adjust the PSO particle locations. PSO operates by keeping a population of particles, representing candidate solutions to the optimization problem, like birds that adapt to a certain environment. The personal best of each particle and the global best of whole swarm decide the movement of particles towards optimum locations. Particles also contribute indirectly by fine tuning their positions to standards, so that the terrain can be optimized effectively.

PSO is successful in various applications, which shows its versatility and usefulness. In robotics and wireless networks, PSO is effective in optimizing the allocation of resources and the setting of network parameters [40, 41]. PSO has been an effective solution to reduce the losses in energy distribution and load management in power systems [42, 43]. PSO has been shown to be effective in a complex environment for scheduling, like job shop scheduling, where resources are scarce and need to be efficiently allocated [44, 45]. This balance will enable the algorithm to thoroughly search the solution space and see promising locations. Very helpful in multi-dimensional optimization problem with a complex and difficult terrain. PSO has been applied to many challenging scenarios including wind speed prediction and the nuclear industry [46, 47]. Initially, particles are randomly initialized, and their positions and velocities are updated at each iteration based on their individual best positions (local best) and the overall best position (global best). The PSO algorithm updates particle positions and velocities using the following equations.

$$\begin{aligned} {v_j}^z = w {v_j}^{t-1} + c_1 r_1 \left(P_{LB}^{z-1} - {Z_j}^{z-1} \right) \\ + c_2 r_2 \left(P_{GB}^{z-1} - {Z_j}^{z-1} \right) \end{aligned} \tag{14}$$

$${Z_j}^z = {Z_j}^{z-1} + {v_j}^{z-1} \tag{15}$$

v_j is the velocity vector of a particle and X_j is a position vector in the swarm. This algorithm also includes parameters like the inertia weight (V), the local and global parameters c_1,c_2 respectively. A bounded random vector r_1 and a random vector r_2, both on the domain [0, 1].

Neural Network Algorithm

The novel evolutionary approach is inspired by biological nerve systems and ANN [48] which have a new form of the NNA. NNA is a new type of neural network that combines the randomness with nonlinear optimization

problem solving in different industries and sciences, as opposed to designs based on prediction tasks like in ANN [49-53]. NNA's basic design allows for obtaining the optimal solution globally. NNA is only concerned with the size of the population, and it does not require the other factors to stop the optimization; unlike the conventional optimization it is. The NNA optimization methodology consists of 4 key elements:

Update population

The vector for population by neural network algorithm $U_t=\{u^t_1, u^t_2, u^t_3,\ldots u^t_M\}$ modify by the weight matrix $W^t = \{w^t_1, w^t_2, w^t_3,\ldots w^t_M\}$, where $w^l_t = \{w^t_{l,1}, w^t_{l,2}, w^t_{l,3},\ldots w^t_{l,M}\}$ shows the weight vector and $u^l_t=\{u^t_{l,1}, u^t_{l,2}, u^t_{l,3},\ldots u^t_{l,D}\}$ indicate the lth individual position. Moreover, D denotes the number of decision variables. The following mathematical expression illustrates the procedure for generating a new population.

$$u^t_{new,l}=\sum_{l=1}^{Q} w^t_{l,m}\times u^t_l, l=1,2,3,\ldots,Q, m=1,2,3,\ldots,Q \quad (16)$$

$$z^t_l = z^t_l + z^t_{new,l},\quad l=1,2,3,\ldots,Q \quad (17)$$

Here, t denotes the iteration number, while Q represents the population size. The weighted estimate represents the current solution of an individual, whereas the corresponding vector denotes the individual's updated solution. Furthermore, the weight vector can be expressed as follows.

$$\sum_{l=1}^{Q} w^t_{l,m}=1, 0<w^t_{l,m}<1,\ l=1,2,3\ldots,Q,\quad m=1,2,3\ldots,Q \quad (18)$$

Update weight matrix:

The weight matrix W^t plays a vital role in population generation within the NNA framework. Its dynamic behavior can be described by the following expression:

$$w^t_l = \left|w^t_l + 2\times\lambda_2\left(w^t_{obj}-w^t_l\right)\right|, l=1,2,3,\ldots,Q \quad (19)$$

Here, λ_2 represents a random number between 0 and 1, while w^t_{obj} denotes the objective or fitness vector at iteration t. It is important to note that the target vector z^t_{obj} and the objective weight vector w^t_v have corresponding indices. When z^t_{obj} matches an individual x^t_v, where $(v\in[1,Q])$, the corresponding objective weight vector w_{objtis} identical to w_{vt} at iteration t.

Bias operator

In NNA, the bias operator is a mechanism that allows the algorithm to have some degree of control over the variations of the search process in order to enhance the algorithm's exploration capability. The amount of bias is controlled by the bias factor β_1 and is adjusted by the following equation:

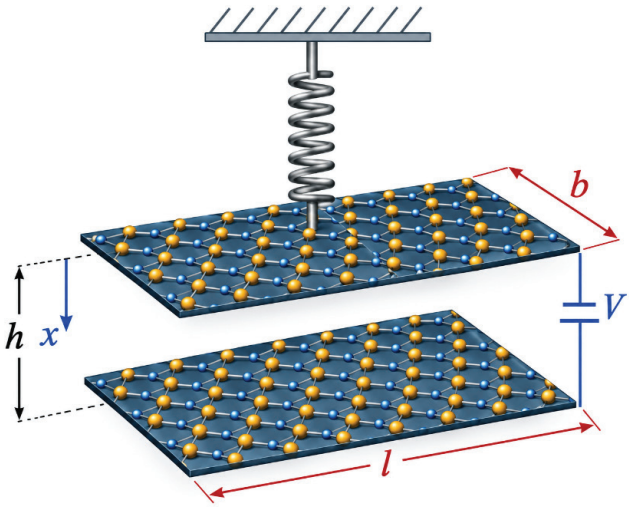


Figure 1. Model of graphene-based N/MEMS model.

$$\beta^{t+1}_1 = 0.99\beta^t_1 \quad (20)$$

The biased operator is a combination of the weight matrix and the biased population that improves the search process. The biased population operator is composed of two parts: the first is a set P and the second is a random number M_p. Assume that the maximum and minimum limits of the parameters are as shown below: The set P is made up of random integers taken from the interval given and the ceiling value β^t_1 is set by the bound given as $\beta_1\times D$. Thus, the following is a statement of the biased population.

$$z^t_{l,P(S)}=l^t_{P(S)}+\left(up(s)-l_{p(s)}\right)\times\lambda_3, S=1,2,3,\ldots,m_p \quad (21)$$

In this case the random variable λ_3 is distributed in the interval [0,1]. The bias matrix consists of two parts, one being a random number M_w and the other being a set R. The random number is the ceiling of the specified value[$\beta^t_1\times M$] and the integers R are integers chosen at random from the interval [0,M]. Thus, the bias weight matrix can be modeled with the following equation.

$$w^t_{l,R(r)}=\lambda_4,\quad r=1,2,3,\ldots,M_w \quad (22)$$

λ_4 indicate the random value between 0 and 1.

Transfer operator

The transfer operator refines the current solution by exploiting the local search capability of NNA, as expressed in Equation (23). Here, the random variable is uniformly distributed over the interval [0,1]. Similar to other optimization algorithms, NNA begins the search process with an initial population of candidate solutions.

$$u^{t+1}_j=u+2\lambda_5\left(u^t_{obj}-u^t_l\right), l=1,2,3,\ldots,M \quad (23)$$

Algorithm 1. Hybrid PSO–NNA Optimization for Neural Network Weights and Biases

Phase I: Particle Swarm Optimization

Step 1: Swarm Initialization. Generate an initial population of particles with randomly assigned positions and velocities. Specify the PSO control parameters and search boundaries.

Step 2: Objective Function Evaluation. Calculate the fitness value of each particle using the objective functions defined in Equations (12) and (13).

Step 3: Best-Solution Identification. Evaluate the fitness values and determine the personal best position of each particle and the global best solution of the entire swarm.

Step 4: Convergence Assessment. Check whether the maximum number of iterations has been reached or the fitness error satisfies $\epsilon \leq 10^{-15}$. If either condition is met, proceed to Step 7.

Step 5: Particle Update. Update the particle velocities and positions using Equations (14) and (15), respectively.

Step 6: Iterative Search. Repeat Steps 2–5 until a stopping condition is satisfied.

Step 7: Optimal Parameter Transfer. Store the best global solution and use its optimized neural network weights and biases as the initial parameter set for the NNA phase.

Phase II: Hybrid PSO–NNA Refinement

Step 8: Input and Initialization. Initialize the NNA optimization process using the best weights and biases obtained from the PSO phase.

Step 9: Fitness Evaluation. Compute the objective function using Equations (28) and (31) to assess the quality of the current neural network solution.

Step 10: NNA-Based Refinement. Apply the NNA search operators to update the weights and biases, aiming to improve the current solution and reduce the fitness error.

Step 11: Termination Check. Examine whether the maximum iteration limit has been reached or the fitness error satisfies $\epsilon \leq 10^{-15}$. If neither condition is satisfied, return to Step 9.

Step 12: Final Parameter Storage. Save the optimized weights and biases obtained from the hybrid PSO–NNA procedure and return the corresponding neural network solution.

End of Algorithm

$$u_{l,m}^{t} = l_m + (u_m - l_m) \times \lambda_6, l = 1,2,3,\dots,M, \quad m = 1,2,3,\dots,D \tag{24}$$

λ_6 represent the random value between [0,1].

Performance Evaluation Metrics

This section presents the statistical evaluation of the nonlinear MEMS model using the mean squared error, Theil's inequality coefficient and mean absolute deviation. The mathematical expressions for these three-performance metrics are provided below.

$$\mathrm{MSE} = \sum_{\mathrm{p}=1}^{q} (u_p - \hat{u}_p)^2, \tag{25}$$

$$\mathrm{T.I.C} = \frac{\sqrt{\frac{1}{q}\sum_{\mathrm{p}=1}^{q}(u_p - \hat{u}_p)^2}}{\sqrt{\frac{1}{q}\sum_{\mathrm{p}=1}^{q}(u_p)^2} + \sqrt{\frac{1}{q}\sum_{\mathrm{p}=1}^{q}(\hat{u}_p)^2}}, \tag{26}$$

$$\mathrm{MAD} = \frac{1}{q}\sum_{\mathrm{p}=1}^{q}\left|u_p - \hat{u}_p\right| \tag{27}$$

RESULTS AND DISCUSSION

A numerical evaluation of the proposed MWNN-PSO-NNA algorithm for solving the graphene N/MEMS system is presented in this section. To investigate the reliability and accuracy of the method under the different operating conditions, two different scenarios were studied, with different constant coefficient parameters in N/MEMS (1). Each scenario is one set of system conditions and enables us to explore the impact of changing the model parameters on the system dynamic response. The technique is provided and approximate solutions are calculated using this method and compared with the reference solutions obtained using the same to evaluate the performance of the method. The accuracy of the results is checked with respect to the agreement between the predicted responses and the reference responses. For this, graphical comparisons are also used, showing the behavior of the system throughout the time interval under consideration. The numerical results also provide a guidance on the

ability of the proposed method to account for the nonlinear behavior of the graphene-based nano/microelectromechanical systems. The stability and consistency of the computed solutions under both scenarios are emphasized. We will be testing the algorithm with varying parameters to get a better understanding of its usefulness in the nonlinear dynamics of graphene N/MEMS. Results of these experiments are presented separately for each scenario.

Scenario 1

$$\frac{d^2u}{dt^2} + u - 0.046(1 + 2u + 3u^2 + 4u^3) \tag{28}$$

$$u(0) = 0.2, \qquad \left.\frac{du}{dt}\right|_{t=0} = 0 \tag{29}$$

A merit function is given as,

$$E_F = \frac{1}{N}\sum_{j=1}^{m}\left(\frac{d^2\hat{u}}{dt_j^2} + \hat{u}_j - 0.046(1 + 2\hat{u}_j + 3\hat{u}_j^2 + 4\hat{u}_j^3\right)^2 + \frac{1}{2}\left((\hat{u}_0 - 0.2)^2 + \left(\frac{d\hat{u}_0}{dt}\right)^2\right) \tag{30}$$

Scenario 2

$$\frac{d^2u}{dt^2} + u - 0.01u^2 - 0.0103(1 + 2u + 3u^2 + 4u^3) \tag{31}$$

$$u(0) = 0.5, \qquad \left.\frac{du}{dt}\right|_{t=0} = 0 \tag{32}$$

Error function is demonstrated as follows:

$$E_F = \frac{1}{N}\sum_{j=1}^{m}\left(\frac{d^2\hat{u}}{dt_j^2} + \hat{u}_j - 0.01\hat{u}_j^2 - 0.0103(1 + 2\hat{u}_j + 3\hat{u}_j^2 + 4\hat{u}_j^3\right)^2 + \frac{1}{2}\left((\hat{u}_0 - 0.5)^2 + \left(\frac{d\hat{u}_0}{dt}\right)^2\right) \tag{33}$$

Scenario 3

$$\frac{d^2u}{dt^2} + u - 0.08u^2 - 0.115(1 + 2u + 3u^2 + 4u^3) \tag{34}$$

$$u(0) = 0.1, \qquad \left.\frac{du}{dt}\right|_{t=0} = 0 \tag{35}$$

The fitness function is given as follows:

$$E_F = \frac{1}{N}\sum_{j=1}^{m}\left(\frac{d^2\hat{u}}{dt_j^2} + \hat{u}_j - 0.08\hat{u}_j^2 - 0.115(1 + 2\hat{u}_j + 3\hat{u}_j^2 + 4\hat{u}_j^3\right)^2 + \frac{1}{2}\left((\hat{u}_0 - 0.1)^2 + \left(\frac{d\hat{u}_0}{dt}\right)^2\right) \tag{36}$$

The MWNN-PSO-NNA method is applied to solve the graphene-based model over 20 independent runs using different network parameter settings. In this process, the unknown network weights are optimized to obtain accurate solutions, as described below:

$$\begin{aligned}\hat{u}_1(t) = &-0.31568073\cos[\tfrac{7}{4}(0.09625538t + 0.30982734)]\, e^{-\frac{1}{2}(0.09625538t + 0.30982734)^2}\\ &- 0.39766309\cos[\tfrac{7}{4}(0.13696684t + 0.17762220)]\, e^{-\frac{1}{2}(0.13696684t + 0.17762220)^2}\\ &+ 0.34941602\cos[\tfrac{7}{4}(0.17476581t - 0.41233763)]\, e^{-\frac{1}{2}(0.17476581t - 0.41233763)^2}\\ &- 0.79761194\cos[\tfrac{7}{4}(-0.21218549t + 0.36763384)]\, e^{-\frac{1}{2}(-0.21218549t + 0.36763384)^2}\\ &- 0.74855672\cos[\tfrac{7}{4}(-0.09044882t - 0.41598823)]\, e^{-\frac{1}{2}(-0.09044882t - 0.41598823)^2}\\ &- 0.79446880\cos[\tfrac{7}{4}(0.18515875t + 0.00444848)]e^{-\frac{1}{2}(0.18515875t + 0.00444848)^2}\\ &- 0.63046444\cos[\tfrac{7}{4}(-0.14203671t - 0.80976703)]\, e^{-\frac{1}{2}(-0.14203671t - 0.80976703)^2}\\ &- 0.72305902\cos[\tfrac{7}{4}(0.18014071t - 0.13984073)]\, e^{-\frac{1}{2}(0.18014071t - 0.13984073)^2}\\ &- 0.66060531\cos[\tfrac{7}{4}(0.03621095t + 0.44418585)]\, e^{-\frac{1}{2}(0.03621095t + 0.44418585)^2}\\ &+ 0.47796952\cos[\tfrac{7}{4}(0.12173541t - 0.46772317)]\, e^{-\frac{1}{2}(0.12173541t - 0.46772317)^2}\end{aligned}$$

$$\begin{aligned}\hat{u}_2(t) = &+0.34541265\cos[\tfrac{7}{4}(-0.24346668t + 0.88214029)]e^{-\frac{1}{2}(-0.24346668t + 0.88214029)^2}\\ &- 0.33827651\cos[\tfrac{7}{4}(0.05947903t - 0.51290293)]e^{-\frac{1}{2}(0.05947903t - 0.51290293)^2}\\ &+ 0.12994626\cos[\tfrac{7}{4}(-0.16015769t + 0.31786743)]e^{-\frac{1}{2}(-0.16015769t + 0.31786743)^2}\\ &+ 0.14627845\cos[\tfrac{7}{4}(-0.30788171t - 0.12700461)]e^{-\frac{1}{2}(-0.30788171t - 0.12700461)^2}\\ &- 0.11019508\cos[\tfrac{7}{4}(0.24385736t - 0.37886755)]e^{-\frac{1}{2}(0.24385736t - 0.37886755)^2}\\ &- 0.98816953\cos[\tfrac{7}{4}(-0.35781081t - 0.38908963)]e^{-\frac{1}{2}(-0.35781081t - 0.38908963)^2}\\ &+ 0.63084757\cos[\tfrac{7}{4}(-0.10996961t - 0.24091022)]e^{-\frac{1}{2}(-0.10996961t - 0.24091022)^2}\\ &- 0.62236249\cos[\tfrac{7}{4}(0.13206141t - 0.95300717)]e^{-\frac{1}{2}(0.13206141t - 0.95300717)^2}\\ &+ 0.84292769\cos[\tfrac{7}{4}(-0.29112705t + 0.65110921)]\, e^{-\frac{1}{2}(-0.29112705t + 0.65110921)^2}\\ &+ 1.03020440\cos[\tfrac{7}{4}(-0.20102395t - 1.26008159)]\, e^{-\frac{1}{2}(-0.20102395t - 1.26008159)^2}\end{aligned}$$

$$\begin{aligned}\hat{u}_3(t) = &+0.59235013\cos[\tfrac{7}{4}(-0.07001937t + 0.91725623)]\, e^{-\frac{1}{2}(-0.07001937t + 0.91725623)^2}\\ &- 0.50205623\cos[\tfrac{7}{4}(-0.11300419t + 0.49019418)]\, e^{-\frac{1}{2}(-0.11300419t + 0.49019418)^2}\\ &+ 0.42680554\cos[\tfrac{7}{4}(-0.12935549t + 0.26277298)]\, e^{-\frac{1}{2}(-0.12935549t + 0.26277298)^2}\\ &+ 0.37257783\cos[\tfrac{7}{4}(0.14876382t + 0.82094373)]\, e^{-\frac{1}{2}(0.14876382t + 0.82094373)^2}\\ &+ 0.67131611\cos[\tfrac{7}{4}(-0.11364887t - 0.59824937)]\, e^{-\frac{1}{2}(-0.11364887t - 0.59824937)^2}\\ &+ 0.08320066\cos[\tfrac{7}{4}(0.08898507t + 0.72207993)]\, e^{-\frac{1}{2}(0.08898507t + 0.72207993)^2}\\ &- 0.91508304\cos[\tfrac{7}{4}(0.01438620t - 0.54714424)]\, e^{-\frac{1}{2}(0.01438620t - 0.54714424)^2}\\ &+ 0.43593713\cos[\tfrac{7}{4}(-0.10992117t + 0.14938902)]e^{-\frac{1}{2}(-0.10992117t + 0.14938902)^2}\\ &- 0.63177833\cos[\tfrac{7}{4}(-0.05938429t - 0.18328903)]\, e^{-\frac{1}{2}(-0.05938429t - 0.18328903)^2}\\ &- 0.66274112\cos[\tfrac{7}{4}(-0.12162378t + 0.15674049)]e^{-\frac{1}{2}(-0.12162378t + 0.15674049)^2}\end{aligned}$$

The Morlet wavelet neural network (MWNN) is developed and combined with PSO and a neural network algorithm (NNA) to create a hybrid optimization system, which is called Morlet wavelet neural network particle swarm optimization neural network algorithm (MWNN-PSO-NNA), to solve nonlinear graphene nano/microelectromechanical systems (N/MEMS). The primary goal of this hybrid approach is to enhance prediction accuracy, convergence and reliability in neural networks by optimizing the network weights and biases. This proposed framework uses

a two-stage optimization technique. In the first phase, PSO is applied to get a set of initial weights and biases which will be used as a good starting point to begin the second training phase. These initial parameter values are subsequently improved by means of NNA in the second stage in order to get closer to the nonlinear system response. This is to minimize the errors associated with the prediction and enhance the overall performance of the MWNN model by using global search followed by parameter refinement.

To examine the effectiveness of the method proposed, the governing equations of the graphene N/MEMS model are numerically solved with a network of 10 neurons. To investigate the regularity of the numerical results and the sensitivity of the optimization method to the differences in initializations, the procedure is repeated 20 times independently. The optimised weights calculated for the proposed framework are shown in Figure 2. These parameters are used to define the trained network and help in its approximation of the nonlinear response of the graphene system. The MWNN-PSO-NNA forecasts are compared with the reference solutions in Figure 3 for the same time periods. The x-axis is the time and the y-axis is the values of the solutions. The good matching of the predicted curve and the reference curves observed, shows the capability of the proposed method to simulate the system response for a range of time intervals. In a nonlinear system of graphene, for example, a slight error in prediction could affect the understanding of the dynamics of the system.

The prediction capability of the proposed framework is then analyzed with three criteria of statistical errors: mean squared error (MSE), mean absolute deviation (MAD) and Theil's inequality coefficient (TIC). These measures are additional information on the accuracy of the computed solutions. MSE is more effective than others for detecting large differences between the predicted and reference values. The MAD is a direct measure of the typical difference between the two solutions, as it is the average magnitude of absolute prediction errors. The normalized measure of the difference between the predicted and the reference responses is provided by TIC, which simplifies the evaluation of predictive accuracy. The convergence and error-distribution results are shown in Figure 4–6. These measures then can serve as a more holistic evaluation of the numerical performance of MWNN-PSO-NNA than any one of them individually. The distributions of the TIC, MAD and MSE are plotted in Figures 4-6, respectively, for the considered graphene system cases. The histograms show the distribution of the individual error values measured over the 20 independent runs, and the box plots provide a summary of the central tendency, dispersion and possible

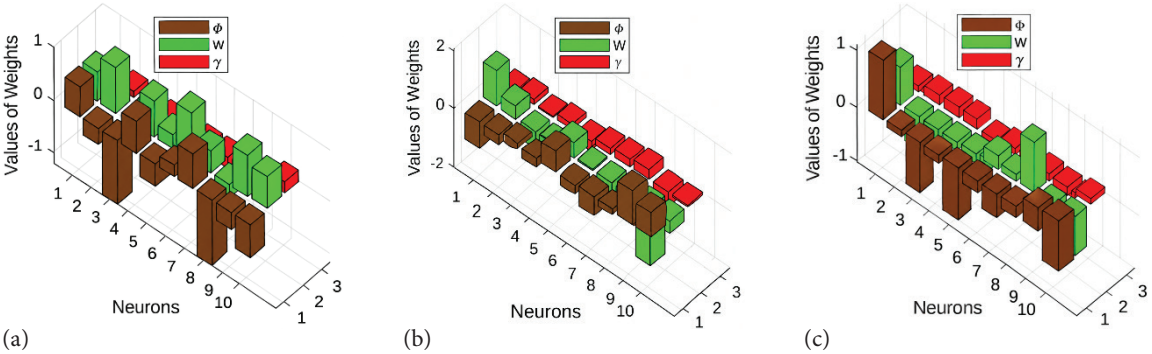


Figure 2. Best-optimzed weight plot using suggested scheme for N/MEMS system for scenarios 1-3.

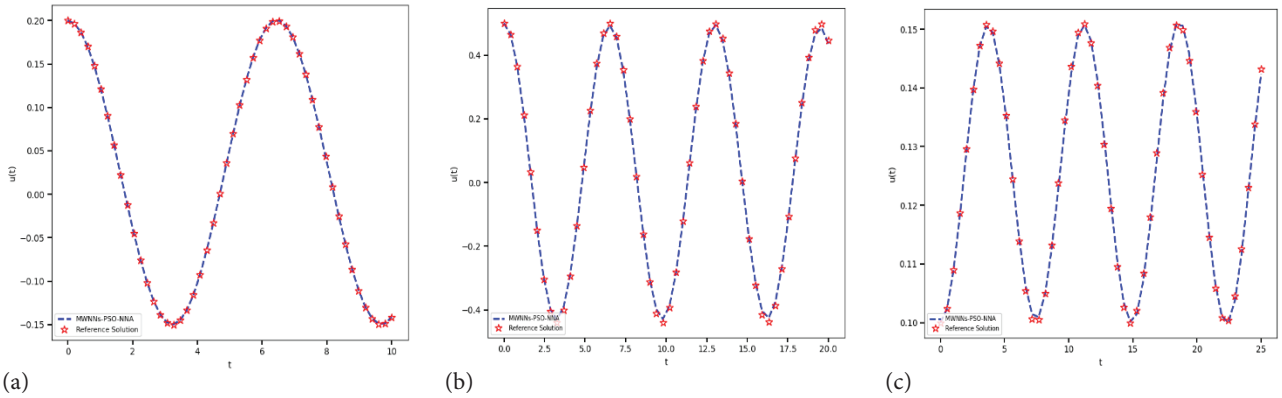
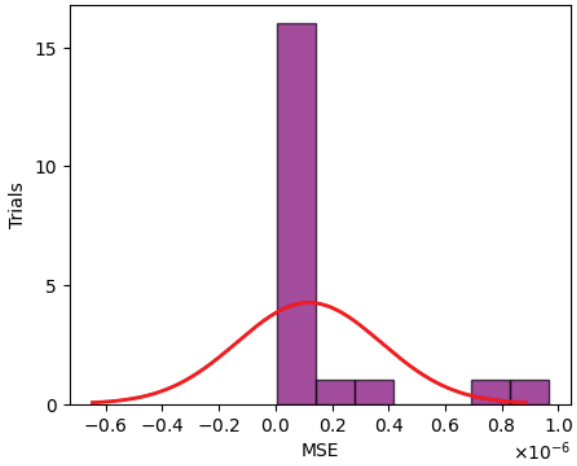
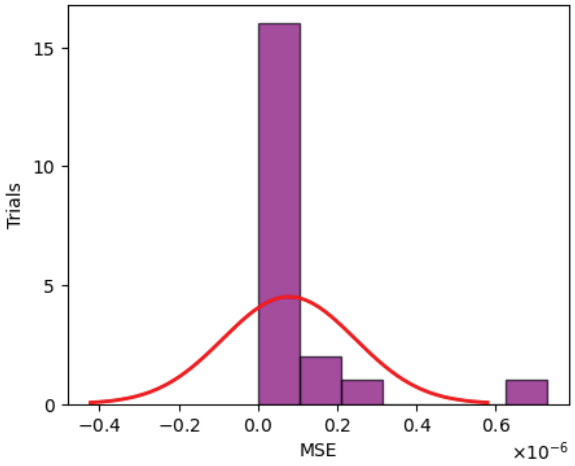


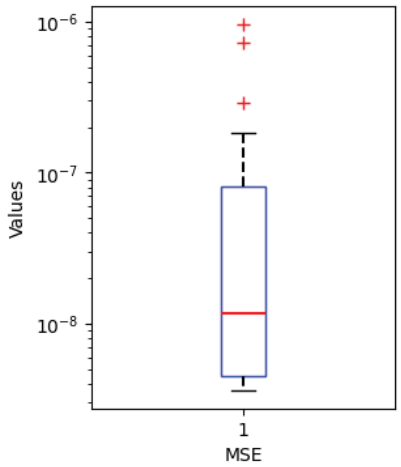
Figure 3. Comparison of MWNN-PSO-NNA results with numerical scheme for graphene model 1-3.



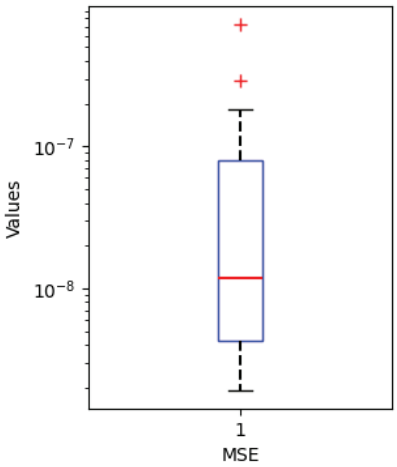
(a) HG: Scenario 1



(b) HG: Scenario 2



(c) BP: Scenario 1



(d) BP: Scenario 2

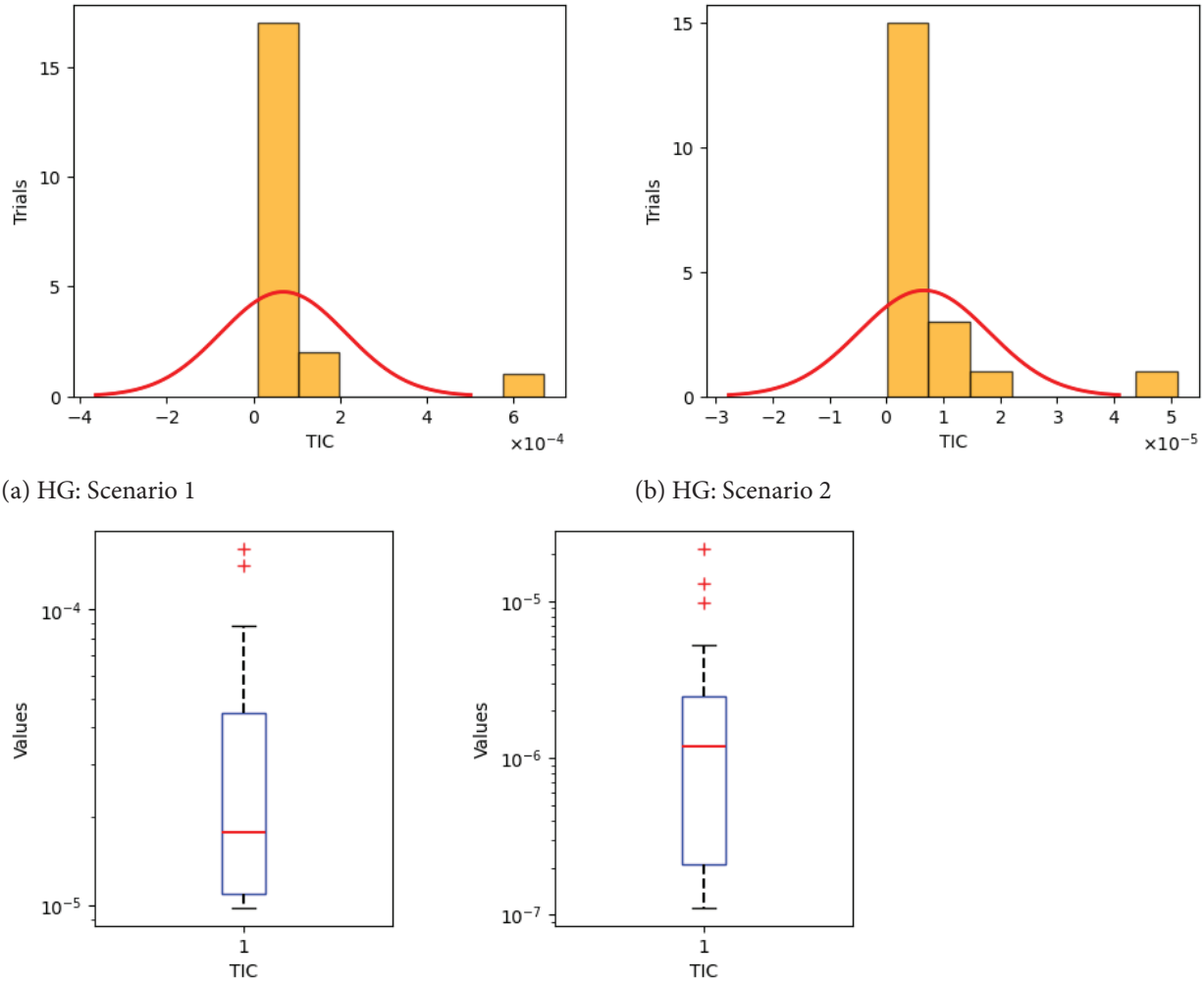
Figure 4. Convergence plot of MSE using suggested technique for N/MEMS model.

Table 1. Optimized weight values using suggested technique for N/MEMS system for scenario 1

δ_{u_i}	ϕ_{u_i}	γ_{u_i}
-0.31568073	0.30982734	0.09625538
-0.39766309	0.17762220	0.13696684
0.34941602	-0.41233763	0.17476581
-0.79761194	0.36763384	-0.21218549
-0.74855672	-0.41598823	-0.09044882
-0.79446880	0.00444848	0.18515875
-0.63046444	-0.80976703	-0.14203671
-0.72305902	-0.13984073	0.18014071
-0.66060531	0.44418585	0.03621095
0.47796952	-0.46772317	0.12173541

Table 2. Optimized weight values using suggested technique for N/MEMS system for scenario 2

δ_{u_i}	ϕ_{u_i}	γ_{u_i}
0.34541265	0.88214029	-0.24346668
-0.33827651	-0.51290293	0.05947903
0.12994626	0.31786743	-0.16015769
0.14627845	-0.12700461	-0.30788171
-0.11019508	-0.37886755	0.24385736
-0.98816953	-0.38908963	-0.35781081
0.63084757	-0.24091022	-0.10996961
-0.62236249	-0.95300717	0.13206141
0.84292769	0.65110921	-0.29112705
1.03020440	-1.26008159	-0.20102395



(a) HG: Scenario 1

(b) HG: Scenario 2

(c) BP: Scenario 1

(d) BP: Scenario 2

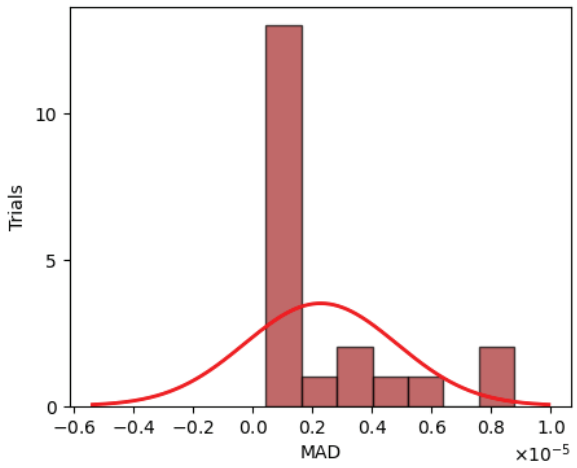
Figure 5. Convergence plot of TIC using suggested technique for N/MEMS model.

outliers for the error distributions generated. A clustering of observations in the low error range suggests that the method will have the ability to consistently approximate at low error. Likewise, the interquartile ranges are narrow, meaning that the results are consistent across independent trials. Some extreme results can be present but it does not necessarily affect the overall performance of the method, it simply means that some runs can be more sensitive to starting conditions or optimization variations. The histograms show the distribution of the errors, with a red curve indicating the fitted distribution (where applicable) to help visualise the overall pattern of errors. The range of numbers and extent of variation should be judged by the numbers given in the figures.

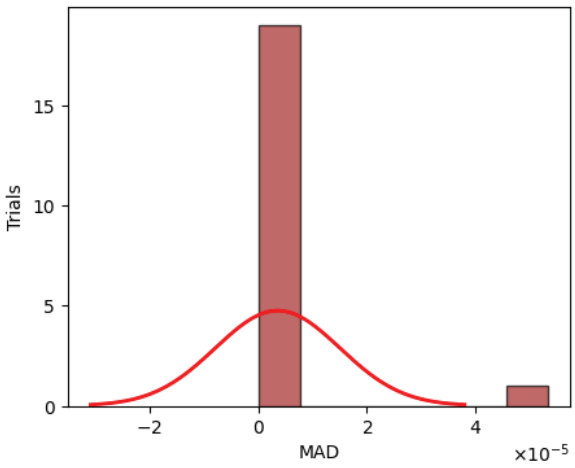
The optimized weights that are produced by applying 10 neurons are presented in Tables 1–3. The values of the

Table 3. Optimized weight values using suggested technique for N/MEMS system for scenario 3

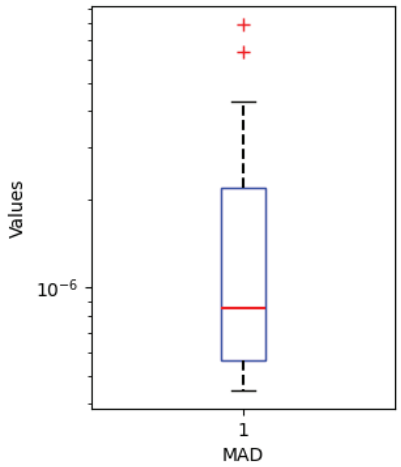
δ_{u_i}	ϕ_{u_i}	γ_{u_i}
0.59235013	0.91725623	-0.07001937
-0.50205623	0.49019418	-0.11300419
0.42680554	0.26277298	-0.12935549
0.37257783	0.82094373	0.14876382
0.67131611	-0.59824937	-0.11364887
0.08320066	0.72207993	0.08898507
-0.91508304	-0.54714424	0.01438620
0.43593713	0.14938902	-0.10992117
-0.63177833	-0.18328903	-0.05938429
-0.66274112	0.15674049	-0.12162378



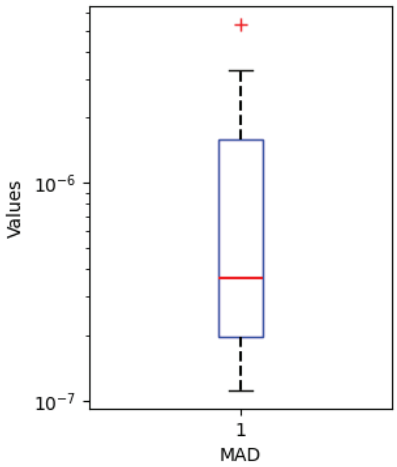
(a) HG: Scenario 1



(b) HG: Scenario 2



(c) BP: Scenario 1



(d) BP: Scenario 2

Figure 6. Convergence plot of MAD using suggested technique for N/MEMS model.

Table 4. Comparison solution of N/MEMS system using MWNN-PSO-NNA with numerical scheme and error value for scenarios 1-2

Scenarios 1, $\alpha = 0$, $k > 0$,				**Scenarios 2, $\alpha > 0$, $k > 0$,**		
t	$u(t)_{Numerical}$	$u(t)_{MWNN}$	$AE\ (u(t)_{MWNN})$	$u(t)_{Numerical}$	$u(t)_{MWNN}$	$AE\ (u(t)_{MWNN})$
0	0.100000000	0.100000000	0.000000000	0.500000000	0.500000000	0.000000000
1	0.108705079	0.108745057	0.000039978	0.302903539	0.296538232	0.006365307
2	0.128688969	0.128742633	0.000053664	-0.127678273	-0.130764100	0.003085827
3	0.145989980	0.146363491	0.000373511	-0.416772243	-0.417591398	0.000819155
4	0.149704536	0.149878020	0.000173484	-0.315076391	-0.314444876	0.000631515
5	0.137768627	0.137052948	0.000715679	0.084973875	0.091768288	0.006794413
6	0.117739686	0.116464775	0.001274911	0.439356709	0.441348733	0.001992024
7	0.102581746	0.102000318	0.000581428	0.443811315	0.443496657	0.000314659
8	0.102586398	0.103552382	0.000965984	0.093506334	0.090698505	0.002807830
9	0.118221208	0.119978207	0.001756999	-0.309456658	-0.318342188	0.008885530
10	0.139214369	0.140037291	0.000822922	-0.415696735	-0.416478040	0.000781305

parameters in these tables are associated with the trained MWNN-PSO-NNA model and they help to reproduce the reported numerical results. Table 4 shows the comparison of the proposed method with the numerical reference solution and error analysis. The agreement reported in these solutions together with the values of MSE, MAD and TIC represent a quantitative support to investigate the reliability of the proposed framework. Specifically, similar small errors for the three cases would suggest that the hybrid method is a good approximation of the governing nonlinear equations. It is also crucial to report the results of all independent runs and not just the best trial to demonstrate the robustness of the optimization procedure. Application-wise, the accurate and efficient modeling of graphene N/MEMS is important to design and analysis of miniaturized electromechanical devices. The extraordinary mechanical and electrical characteristics of graphene render it a useful material for many applications including nanoscale resonators, ultrasensitive sensors, electromechanical transducers, and components for both micro- and nanoelectromechanical systems. Here, the ability to accurately predict the response of a nonlinear system can help explore the behavior of these devices under different operating conditions and to find parameter configurations that yield desirable results. If the MWNN-PSO-NNA model is properly trained and tested, it may be used to simulate repeated calculations of the governing equations, thus minimizing the necessity of these high-computational effort calculations for preliminary design and parameter studies. Its practical suitability, however, would have to be validated over a range of operating conditions relevant to its use and compared to existing results from numerical or experimental testing.

CONCLUSION

In conclusion, the rapid advancement of artificial neural network techniques has introduced innovative techniques for solving complicated nonlinear models in a variety of engineering domains.Nano/microelectromechanical systems have attracted considerable attention in recent decades because of their small size, high reliability, and low energy consumption. It is challenging to tackle the dynamic behavior of these problems using traditional techniques. Therefore, the proposed scheme is an excellent choice for the solution of these intricate systems. We proposed an efficient scheme called morlet wavelet- based neural network combined with particle swarm optimization and a integrated with neural network algorithm to solve the graphene N/MEMS.

The key finding, limitations, and high potential of this study is demonstrated as follows:

1. The best efficiency of MWNN-PSO-NNA is determined by designing an objective function, which is based on the mean square error for the N/MEMS system.
2. To verify the accuracy of the MWNN-PSO-NNA, numerical results are compared with reference solutions for each problem in the graphene N/MEMS model.
3. In order to evaluate the efficiency of the proposed technique, twenty trials were conducted, providing ten neurons, utilizing various statistical measures.
4. Moreover, the statistical metrics, including TIC, MAD, and MSE were evaluated across twenty executions to ensure the correctness of the MWNN–PSO–NNA. Most executions demonstrated a higher level of precision for the N/MEMS models.
5. For all cases of N/MEMS systems, most of the results lie between 10^{-3} to 10^{-8}. All of these values are very small, reliable, and accurate, and demonstrate the efficiency of the proposed methodology.

One limitation of the proposed methodology is the potential loss in accuracy when addressing complicated and fractional-order models. Future research will emphasize enhancing nano/micro-oscillators by introducing more nonlinear components. Further study is needed to establish the amplitude-frequency relationship in N/MEMS. This will not only help us understand these systems better, but it will also help with implementation. Future research should explore alternative intelligent techniques to handle such challenges in nanotechnology, including quantum-based or Legendre-based neural networks, optimization algorithms, and their hybrid combinations.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors declared that the manuscript has no associated data.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

ETHICS

There are no ethical issues with the publication of this manuscript.

STATEMENT ON THE USE OF ARTIFICIAL INTELLIGENCE

Artificial intelligence was not used in the preparation of the article.

REFERENCES

[1] He JH. From micro to nano and from science to technology: nano age makes the impossible possible. Micro Nanosyst 2020;12:2–3. [CrossRef]

[2] El Hassani NEA, Baraket A, Boudjaoui S, Neto ETT, Bausells J, El Bari N, et al. Development and application of a novel electrochemical immunosensor for tetracycline screening in honey using a fully

integrated electrochemical Bio-MEMS. Biosens Bioelectron 2019;130:330–337. [CrossRef]

[3] Kumar M, Yadav S, Kumar A, Sharma NN, Akhtar J, Singh K. MEMS impedance flow cytometry designs for effective manipulation of micro entities in health care applications. Biosens Bioelectron 2019;142:111526. [CrossRef]

[4] Sedighi HM, Malikan M. Stress-driven nonlocal elasticity for nonlinear vibration characteristics of carbon/boron-nitride hetero-nanotube subject to magneto-thermal environment. Phys Scr 2020;95:055218. [CrossRef]

[5] Gong Q, Liu C, Xu Y, Ma C, Zhou J, Jiang R, et al. Nonlinear vibration control with nanocapacitive sensor for electrostatically actuated nanobeam. J Low Freq Noise Vib Act Control 2018;37:235–252. [CrossRef]

[6] Li XX, He JH. Nanoscale adhesion and attachment oscillation under the geometric potential. Part 1: the formation mechanism of nanofiber membrane in the electrospinning. Results Phys 2019;12:1405–1410. [CrossRef]

[7] Qian YH, Pan JL, Qiang Y, Wang JS. The spreading residue harmonic balance method for studying the doubly clamped beam-type N/MEMS subjected to the van der Waals attraction. J Low Freq Noise Vib Act Control 2019;38:1261–1271. [CrossRef]

[8] Mukherjee B, Sen S. Generalized closed form approximations for pull-in characteristics of fixed-fixed nano beam under the influences of van der Waals and Casimir forces. Microelectron Reliab 2019;94:32–40. [CrossRef]

[9] Ouakad HM, Sedighi HM. Static response and free vibration of MEMS arches assuming out-of-plane actuation pattern. Int J Non Linear Mech 2019;110:44–57. [CrossRef]

[10] Moory-Shirbani M, Sedighi HM, Ouakad HM, Najar F. Experimental and mathematical analysis of a piezoelectrically actuated multilayered imperfect microbeam subjected to applied electric potential. Compos Struct 2018;184:950–960. [CrossRef]

[11] He JH, Nurakhmetov D, Skrzypacz P, Wei D. Dynamic pull-in for micro-electromechanical device with a current-carrying conductor. J Low Freq Noise Vib Act Control 2021;40:1059–1066. [CrossRef]

[12] Shirbani MM, Shishesaz M, Hajnayeb A, Sedighi HM. Coupled magneto-electro-mechanical lumped parameter model for a novel vibration-based magneto-electro-elastic energy harvesting systems. Phys E Low Dimens Syst Nanostruct 2017;90:158–169. [CrossRef]

[13] Amir M, Haider JA, Ashraf A. The homotopy perturbation method for electrically actuated microbeams in MEMS systems subjected to van der Waals force and multiwalled carbon nanotubes. Acta Mech Autom 2024;18:123–128. [CrossRef]

[14] Sedighi HM, Daneshmand F. Nonlinear transversely vibrating beams by the homotopy perturbation method with an auxiliary term. J Appl Comput Mech 2014;1:1–9.

[15] He JH. Variational principle and periodic solution of the Kundu-Mukherjee-Naskar equation. Results Phys 2020;17:103031. [CrossRef]

[16] Anjum N, He JH. Analysis of nonlinear vibration of nano/microelectromechanical system switch induced by electromagnetic force under zero initial conditions. Alex Eng J 2020;59:4343–4352. [CrossRef]

[17] Fu Y, Zhang J, Wan L. Application of the energy balance method to a nonlinear oscillator arising in the microelectromechanical system (MEMS). Curr Appl Phys 2011;11:482–485. [CrossRef]

[18] He JH. The simpler, the better: analytical methods for nonlinear oscillators and fractional oscillators. J Low Freq Noise Vib Act Control 2019;38:1252–1260. [CrossRef]

[19] Sunil Kumar K, Muniamuthu S, Salunkhe S, Čep R, Abouel Nasr E. In-situ analysis on drop in thermal performance of wind turbine gearbox system: global challenges and sustainable ideas. Eng Res Express 2025. [CrossRef]

[20] Sunil Kumar K, Kamalakar V, Palani S, Muniamuthu S, Belinda CM. Design of K-type propeller optimized with lightweight aluminum material and its CFD analysis for marine-based applications. In: International Conference on Mechanical Engineering. Singapore: Springer Nature Singapore; 2023. p. 375–390. [CrossRef]

[21] Kumar KS, Surakasi R, Mohan A, Tharanisrisakthi BT, Muniamuthu S, NB G. Emerging trends and global challenges to predict drop in thermal performance of WTG gearbox. J Therm Eng 2021;10:657–669. [CrossRef]

[22] Kumar KS, Raju DBN, Arulmani J, Amirthalingam P. Design and structural analysis of liquified cryogenic tank under seismic and operating loading. Int J Mech Eng Technol 2016;7:345–366.

[23] Bishnoi D. Pressure exertion and heat dissipation analysis on uncoated and ceramic (Al2O3, TiO2 and ZrO2) coated braking pads. Mater Today Proc 2023;74:774–787. [CrossRef]

[24] Muniamuthu S, Kumar KS, Raja K, Rupesh PL. Dynamic characterization of hybrid composite based on flax/E-glass epoxy composite plates. Mater Today Proc 2022;59:1786–1791. [CrossRef]

[25] Ali AH, Amir M, Rahman JU, Raza A, Arif GE. Design of Morlet wavelet neural networks for solving the nonlinear van der Pol-Mathieu-Duffing oscillator model. Computers 2025;14:14. [CrossRef]

[26] Ul Rahman J, Makhdoom F, Ali A, Danish S. Mathematical modeling and simulation of biophysics systems using neural network. Int J Mod Phys B 2024;38:2450066. [CrossRef]

[27] Rahman JU, Danish S, Lu D. Oscillator simulation with deep neural networks. Mathematics 2024;12:959. [CrossRef]
[28] Yang L, Zhang D, Karniadakis GE. Physics-informed generative adversarial networks for stochastic differential equations. SIAM J Sci Comput 2020;42:A292-A317. [CrossRef]
[29] Mattheakis M, Sondak D, Dogra AS, Protopapas P. Hamiltonian neural networks for solving equations of motion. Phys Rev E 2022;105:065305. [CrossRef]
[30] Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J Comput Phys 2019;378:686–707. [CrossRef]
[31] Umar M, Sabir Z, Raja MAZ, Amin F, Saeed T, Sanchez YG. Design of intelligent computing solver with Morlet wavelet neural networks for nonlinear predator-prey model. Appl Soft Comput 2023;134:109975. [CrossRef]
[32] Nisar K, Sabir Z, Raja MAZ, Ibrahim AAA, Erdogan F, Haque MR, et al. Design of Morlet wavelet neural network for solving a class of singular pantograph nonlinear differential models. IEEE Access 2021;9:77845–77862. [CrossRef]
[33] Sabir Z, Wahab HA, Umar M, Sakar MG, Raja MAZ. Novel design of Morlet wavelet neural network for solving second order Lane-Emden equation. Math Comput Simul 2020;172:1–14. [CrossRef]
[34] Bhat SA, Sabir Z, Raja MAZ, Saeed T, Alshehri AM. A novel heuristic Morlet wavelet neural network procedure to solve the delay differential perturbed singular model. Knowl Based Syst 2024;292:111624. [CrossRef]
[35] Mattheakis M, Protopapas P, Sondak D, Di Giovanni M, Kaxiras E. Physical symmetries embedded in neural networks. arXiv [Preprint]. 2019;arXiv:1904.08991.
[36] Piscopo ML, Spannowsky M, Waite P. Solving differential equations with neural networks: applications to the calculation of cosmological phase transitions. Phys Rev D 2019;100:016002. [CrossRef]
[37] Hagge T, Stinis P, Yeung E, Tartakovsky AM. Solving differential equations with unknown constitutive relations as recurrent neural networks. arXiv [Preprint]. 2017;arXiv:1710.02242.
[38] Lee C, Wei X, Kysar JW, Hone J. Measurement of the elastic properties and intrinsic strength of monolayer graphene. Science 2008;321:385–388. [CrossRef]
[39] Lu Q, Huang R. Nonlinear mechanics of single-atomic-layer graphene sheets. Int J Appl Mech 2009;1:443–467. [CrossRef]
[40] Panda S, Padhy NP. Comparison of particle swarm optimization and genetic algorithm for FACTS-based controller design. Appl Soft Comput 2008;8:1418–1427. [CrossRef]
[41] Okwu MO, Tartibu LK, Okwu MO, Tartibu LK. Particle swarm optimisation. In: Metaheuristic optimization: nature-inspired algorithms swarm and computational intelligence, theory and applications. 2021. p. 5–13. [CrossRef]
[42] Yu Y, Yin S. A comparison between generic algorithm and particle swarm optimization. In: Proceedings of the 1st International Symposium on Artificial Intelligence in Medical Sciences; 2020 Sep. p. 137–139. [CrossRef]
[43] Stacey A, Jancic M, Grundy I. Particle swarm optimization with mutation. In: The 2003 Congress on Evolutionary Computation (CEC'03); 2003 Dec. Vol. 2. IEEE; 2003. p. 1425–1430. [CrossRef]
[44] Eberhart R, Kennedy J. A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science (MHS'95); 1995 Oct. IEEE; 1995. p. 39–43. [CrossRef]
[45] Toushmalani R. Gravity inversion of a fault by particle swarm optimization (PSO). Springerplus 2013;2:1–7. [CrossRef]
[46] Sarangi S, Dash PK, Bisoi R. Probabilistic prediction of wind speed using an integrated deep belief network optimized by a hybrid multi-objective particle swarm algorithm. Eng Appl Artif Intell 2023;126:107034. [CrossRef]
[47] Jiang Q, Qu W, Xiao L. Pipeline damage identification in nuclear industry using a particle swarm optimization-enhanced machine learning approach. Eng Appl Artif Intell 2024;133:108467. [CrossRef]
[48] Sadollah A, Sayyaadi H, Yadav A. A dynamic metaheuristic optimization model inspired by biological nervous systems: neural network algorithm. Appl Soft Comput 2018;71:747–782. [CrossRef]
[49] Mannan A, Shoket N, Ul Rahman J, Uwitije R. Dynamic analysis of Ebola virus disease with non-linear incidence rate using Morlet wavelet neural networks and hybrid optimization techniques. Model Earth Syst Environ 2025;11:79. [CrossRef]
[50] Amir M, Rahman JU, Ali AH, Raza A, Abduljabbar ZA, Neamah HA. Deep learning-based technique for investigating the behavior of MEMS systems with multiwalled carbon nanotubes and electrically actuated microbeams. MethodsX 2025;103337. [CrossRef]
[51] Rahman JU, Makhdoom F, Lu D. Amplifying sine unit: an oscillatory activation function for deep neural networks to recover nonlinear oscillations efficiently. arXiv [Preprint]. 2023;arXiv:2304.09759.
[52] Karadurmuş E, Göz E, Keleş C, Yüceer M. Utilizing artificial neural networks (ANN) for predictive modeling of sulfate removal from water. Sigma J Eng Nat Sci 2024;42:1866–1875. [CrossRef]
[53] Kartal F, Kaptan A. Artificial neural network and multiple regression analysis for predicting abrasive water jet cutting of Al 7068 aerospace alloy. Sigma J Eng Nat Sci 2024;42:516–528. [CrossRef]