IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 July 2023, accepted 28 August 2023, date of publication 7 September 2023, date of current version 8 November 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3312708*

RESEARCH ARTICLE

# An Artificial Neural Network Model for Predicting the Nanofluids' Optimal Velocity and Skin Friction in a Convergent Channel

**UPENDAR MENDU[1] AND VEMAN SAI PRABHATH MENDU[2]**

[1]Department of Mathematics, Faculty of Science and Technology, ICFAI Foundation for Higher Education, Hyderabad 501203, India
[2]Department of Communication and Information Systems, University of Central Missori, Warrensberg, MO 64093, USA

Corresponding author: Upendar Mendu (drumendu@ifheindia.org)

This work was supported by the ICFAI Foundation for Higher Education.

**ABSTRACT** This work purveys an artificial neural network (ANN) model for the steady two-dimensional viscous incompressible radial flow of $Au - Water$ (a mixture composed of water as base fluid and nanoparticles of Gold (Au)) and $Ag - Water$ (a mixture composed by water as base fluid and nanoparticles of Silver (Ag)) nanofluids between plane walls which converge in the presence of the MHD effect. The governing partial differential equations of the present physics and their appropriate boundary conditions are initially cast into dimensionless forms to reduce into the ordinary differential equation. The resulting equation thus formed is worked out by adopting the superior numerical simulation, namely, successive linearization method (SLM) to get the accurate solution, and used to generate the data set of size $1502 \times 4$. A great agreement of the numerical simulation is achieved with the existing results. Two ANN models (Model 1 and Model 2) with two hidden layers were proposed. In Model 1 first hidden layer consists of 5 neurons and the second hidden layer consists of 3 neurons, whereas Model 2 consists of 10 neurons in the first hidden layer and 5 neurons in the second hidden layer. The results showed that the proposed ANN model can predict the velocity with high accuracy, and helps to predict the velocity/skin friction for any combination of the values of the proposed fluid's physical parameters. An R-squared value reaching 0.99 can be obtained using an artificial neural network with the Stochastic Gradient Descent (SGD) optimizer, with a momentum of 0.8, and a learning rate of 0.02.

**INDEX TERMS** Artificial neural networks (ANNs), convergent channel, MHD, nanofluids, skin friction.

## NOMENCLATURE

| | |
|---|---|
| $\alpha$ | Reference angle ($Rd$). |
| $\eta$ | Non-dimensional parameter of angle. |
| $\hat{y}$ | Predicted value of the target variable. |
| $\mu$ | Dynamic viscosity of the fluid ($Pa\,s$). |
| $\mu_{nf}$ | Dynamic viscosity of the nanofluid ($Pa\,s$). |
| $\overline{J}$ | Current density $\left(A\,m^{-2}\right)$. |
| $\overline{V}$ | Dimensional velocity vector ($MS^{-1}$). |
| $\phi$ | Solid volume fraction of Nanoparticles. |
| $\rho$ | Density of the fluid ($kg\,m^{-3}$). |
| $\rho_f$ | Density of the base fluid ($kg\,m^{-3}$). |
| $\rho_{nf}$ | Density of the nanofluid ($kg\,m^{-3}$). |
| $\rho_s$ | Density of the nano solid particles ($kg\,m^{-3}$). |
| $\sigma$ | Conductivity of the fluid (Siemens per meter). |
| $\theta$ | Plane angle ($Rd$). |
| $\varphi$ | Activation function. |
| $a^{[l]}$ | Activation in layer $l$. |
| $A_{i-1}$ | Coefficient matrix. |
| $A_{ij}$ | (i, j) element of the coefficient matrix. |
| $Ag$ | Solver (Argentum). |
| $Au$ | Gold (Aurum). |
| $B$ | Non-dimensional magnetic field. |
| $b^{[l]}$ | Bias in layer $l$. |
| $D$ | Differential operator. |
| $d\,a^{[l]}$ | Differentiation of Loss function with respect to activation $a^{[l]}$. |
| $d\,W^{[l]}$ | Differentiation of Loss function with respect to weight $W^{[l]}$. |

The associate editor coordinating the review of this manuscript and approving it for publication was Vivek Kumar Sehgal.

© 2023 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

| | |
|---|---|
| $d\,b^{[l]}$ | Differentiation of Loss function with respect to bias $b^{[l]}$. |
| $L$ | Final layer number of the neural network. |
| $l$ | Layer number of the neural network ($l = 0, 1, 2, \ldots, L$). |
| $M$ | Magnetic parameter number. |
| $MHD$ | Magneto-hydrodynamics. |
| $p$ | Fluid pressure (Pa). |
| $R^2$ | Coefficient of regression. |
| $Re$ | Reynolds number. |
| $u$ | Dimensional velocity component of $\overline{V}$ along r-direction ($m\,s^{-1}$). |
| $u_{max}$ | Center line rate of movement of the fluid ($m\,s^{-1}$). |
| $u_w$ | Velocity of the fluid on the surface ($m\,s^{-1}$). |
| $v$ | Dimensional velocity component of $\overline{V}$ along $\theta$ direction ($m\,s^{-1}$). |
| $w$ | Dimensional velocity component of $\overline{V}$ along $z$ direction ($MS^{-1}$). |
| $W^{[l]}$ | Weight in layer $l$. |
| $y_i$ | Original (Experimental/Simulated) value of the target variable. |

## I. INTRODUCTION

The steady two-dimensional viscous incompressible radial flow between converge/diverge plane walls, is one of the bulk-appropriate manifestations in Bio-mechanical engineering, applied mathematics, civil, environmental, and fluid mechanics. Reference [1] first coined the exploration on this subject, and he has named it to be the initiating point for several researchers by other authors; notably, [2], [3], [4], [5], [6], [7], [8]. In the pioneering works of [1] and [2], they derived an exact solution to the proposed mathematical model of viscous incompressible fluid flow through non-parallel walls. Further, various researchers [9], [10], [11], [12], [13], [14] conducted the analytical and numerical treatment of the Jeffery - Hamal flow.

Artificial neural networks (ANNs) got inspiration from the principles of neuronal organization in biological neural networks. These networks learn tasks by running on the data sets [15]. The demand for neural networks has been increasing over the last two decades because of their capability to learn complex patterns and relationships within the data in classification, pattern recognition, regression, and forecast [16], [17], [18], [19], [20], [21]. Neural networks have become all the range because of their advances in processing power and the high accuracy of traditional statistical methods. The pivotal attribute of artificial learning is the use of artificial neural networks, which have multiple layers of interconnected nodes. These networks can learn complex representations of data by discovering hierarchical patterns and features in the data. Artificial learning algorithms can automatically learn and improve from data without the need for manual feature engineering. Artificial neural networks have a higher accuracy of predicting non-linear outcomes, compared to traditional linear statistical models [22]. Artificial neural networks consist of hidden layers between input and output layers. Each hidden layer consists of some neurons. After all, ANNs have not been used for predicting the effects of physical and fluid parameters on the skin friction of nanofluid, efficiency, and output power. The ANN models could yield high predictive capabilities that can be used rather than conducting experiments that are time, resource consuming, and costly.

With inspiration from neural networks, the authors intend to develop a data set using a semi-analytic mathematical method on the governing equations to implement the ANNs to find the impact of the physical parameters on the velocity field.

## II. MATHEMATICAL FORMULATION

Consider the steady two-dimensional radial motion of incompressible $Au-Water$ and $Ag-Water$ nanofluids within the inclined plane walls in the presence of the magnetic field. The representative plane is perpendicular to the inclined walls. The co-ordinates are $r$, $\theta$ where $r$ is measured from origin ($O$), the crossing of the walls, and $\theta$ is evaluated from the midway strip $OX$ (see Fig. (1)). The walls of the channel cut the representative region in the lines $OV$, $OW$ ($\theta = \pm\alpha$). The coordinate perpendicular to this surface is $h$. The flow diverges from/converges to the line represented by $\theta = 0$. The velocities in the directions of $r$, $\theta$, $h$ are $u$, $v$, $w$. In view of these cogitations along with boundary layer approximations, the equations of the present physics are given below.

$$\nabla.\overline{V} = 0 \tag{1}$$

$$\rho(\overline{V}.\nabla)\overline{V} = -\nabla p + \mu\nabla^2\overline{V} + \sigma\overline{J}\times\overline{B} \tag{2}$$

where $\overline{V} = u(r,\theta)i_r + v(r,\theta)i_\theta + w(r,\theta)i_h$ is the vector of the velocity, $\overline{J}$ is current density, $\overline{B}$ is the magnetic field, $\sigma$ be the conductivity of the fluid, p is the fluid pressure,$\rho$ is the density of the fluid, $\mu$ is the dynamical viscosity of the fluid.

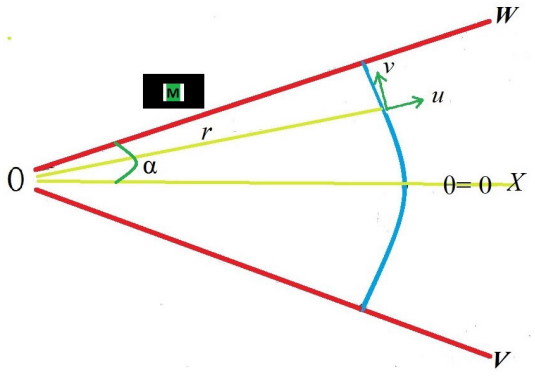


**FIGURE 1.** Physical representation.

Since the flow is radial, it is evident that $v = w = 0$ and the velocity component along r is $u(r,\theta)$ (see [1], [3]). The governing equations for the Jeffery – Hammel flow for

Nanofluids in polar co-ordinates follow as [9]

$$\rho_{nf}\left(\frac{\partial u}{\partial r}+\frac{u}{r}\right)=0 \tag{3}$$

$$\left.\begin{aligned}\rho_{nf}\,u\,\frac{\partial u}{\partial r}&=-\frac{\partial p}{\partial r}-\sigma B^2\frac{u}{r^2}\\&+\mu_{nf}\left(\frac{\partial^2 u}{\partial r^2}+\frac{1}{r}\frac{\partial u}{\partial r}+\frac{1}{r^2}\frac{\partial^2 u}{\partial\theta^2}-\frac{u}{r^2}\right)\end{aligned}\right\} \tag{4}$$

$$\frac{1}{r}\frac{\partial p}{\partial\theta}=\frac{2\mu_{nf}}{r^2}\frac{\partial u}{\partial\theta} \tag{5}$$

where $u(r,\theta)$ is the $r$ - direction component of the velocity.

The effective dynamic viscosity of the nanofluid is given by ( [23]) as

$$\mu_{nf}=\frac{\mu_f}{(1-\phi)^{2.5}} \tag{6}$$

where $\phi$ is the solid volume fraction of Nanoparticles. The effective density of the Nanofluids is given as

$$\rho_{nf}=(1-\phi)\rho_f+\phi\rho_s \tag{7}$$

here, the subscripts $nf$,$f$, and s represent the nanofluid's thermo physical properties, base fluid, and nano solid particles, respectively.

The boundary conditions are:

$$\begin{aligned}&u=u_{max},\ \frac{\partial u}{\partial\theta}=0,\quad at\quad \theta=0,\\&u=u_w=0,\quad at\quad \theta=\alpha\end{aligned} \tag{8}$$

where $u_{max}$ being the center line rate of movement.

In view of Eq. (3) and the assumption on directions of the fluid particles in the flow considered to be the straight lines passing through a point in a plane aggregate the streamlines, this results in the variation of the velocity from line to line, hence physically the velocity is to be considered as a function of the polar angle $\theta$ and further $\frac{\partial u}{\partial\theta}=0$ at $\theta=0$ yields the velocity component $u(r,\theta)$ to

$$u(r,\theta)=\frac{f(\theta)}{r} \tag{9}$$

where $f(\theta)$ is an arbitrary function of $\theta$. Making use of dimension analysis

$$f(\eta)=\frac{f(\theta)}{u_{max}},\quad \eta=\frac{\theta}{\alpha} \tag{10}$$

In view of the Equations (9) and (10) and get rid of the pressure $p$ from Eqs. (4) and (5) the governing equations (3) - (5) reduce to

$$\left.\begin{aligned}f'''+2\alpha\,Re((1-\phi)+\phi\,\frac{\rho_s}{\rho_f})(1-\phi)^{2.5}ff'+\\(4\ -\ M\,(1-\phi)^{2.5})\alpha^2 f'=\ \ 0\end{aligned}\right\} \tag{11}$$

$$Re=\frac{\alpha u_{max}}{\upsilon_f},\quad M=\frac{B^2\sigma}{\mu_f} \tag{12}$$

The channel

$$\begin{aligned}&\text{diverges}\quad \text{if}:\alpha>0\quad\text{and}\quad U_{max}>0,\\&\text{Converges}\quad \text{if}:\alpha<0\quad\text{and}\quad U_{max}<0\end{aligned} \tag{13}$$

The Eq. (8) in terms of $f(\eta)$ are:

$$f(0)=1,\quad f'(0)=0,\quad f(1)=0 \tag{14}$$

### A. SUCCESSIVE LINEARISATION METHOD (SLM)

To investigate the successive linearization application on Eqs. (11) we suppose that the solution $f(\eta)$ can be enlarged as

$$f(\eta)=f_i(\eta)+\sum_{m=o}^{i-1}f_m(\eta)\quad i=1,2,3\ldots \tag{15}$$

where $f_i$'s are unknown functions and $f_m$'s are approximations that are obtained by recursively solving the linear part of the equation that results from substituting (15) in the governing equation (11) and neglecting the nonlinear term $ff_i'$ gives

$$f_i'''+a_{1,i-1}f_i'+a_{2,i-1}f_i=r_{1,i-1} \tag{16}$$

where the coefficient parameters $a_{k,i-1}$, $(k=1,2)$, $r_{1,i-1}$, are defined as

$$\begin{aligned}a_{1,i-1}&=2\,Re\,\alpha\,A_1\,A_2\sum f_m+(4-A_2\,M)\,\alpha^2,\\a_{2,i-1}&=2\,Re\,\alpha\,A_1\,A_2\sum f_m',\\r_{1,i-1}&=-\sum f_m'''-2\,Re\,\alpha\,A_1\,A_2\sum f_m\sum f_m'\\&\quad-(4-A_2\,M)\,\alpha^2\sum f_m'\\A_1&=\left[(1-\phi)+\phi\frac{\rho_s}{\rho_f}\right]\text{ and }A_2=(1-\phi)^{2.5}\end{aligned}$$

When every solution of $F_i$, $(i\geqslant 1)$ has been set up from iteratively working out the equation (16) the approximate solution for $f(\eta)$ is obtained as

$$f(\eta)\approx\sum_{m=0}^{\ell}F_m(\eta) \tag{17}$$

where $\ell$ is the order of SLM approximation. Equation (17), is obtained by assuming that $F_i$ becomes increasingly smaller as $i$ becomes large, that is

$$\lim_{i\to\infty}F_i=0, \tag{18}$$

Afterward, knowing the coefficients and right-hand side terms of Eq. (16) at the $(i-1)^{th}$ (previous) iteration, the Eq. (16) can be trivially solved by making use of the numerical methods, like, FDM, FEM, etc. In the present work, the Eq. (16) is solved using the spectral linearization method. The background of this method is approximating the unrevealed functions by the Chebyshev interpolating polynomials so as they are accumulated at the Gauss - Lobatto points elucidated as

$$\eta_j=cos\left(\frac{\xi_j}{N}\right),\quad j=0,1,2,3\ldots N \tag{19}$$

where $N$ is the collocation points number. The derivative of $F_r$ at the accumulated points are constituted by way of

$$\frac{d^s F_r}{d\eta^s} = \sum_{k=0}^{N} d_{kf}^s F_r(\eta_k) = D^s F, \quad j = 0, 1, \ldots, N \tag{20}$$

where $D^s = (\frac{2\Im}{L})^s$ and $\Im$ is the spectral differentiation matrix of Chebyshev (see, e.g., ( [24], [25]), and $F$, the vector function, is given by $F = [F(\eta_0), F(\eta_1), \ldots, F(\eta_N)]^T$, where $s$ is the derivative order, and $\mathbf{D}$ is the $(N+1) \times (N+1)$ matrix. Use Eqs. (19) and (20) in the scheme of the Spectral Linearization Method brings in the following matrix equation

$$A_{i-1} F_i = R_{i-1} \tag{21}$$

and the corresponding boundary conditions are reduced to

$$\sum_{k=0}^{N} F_i(\xi_N) = 0, \sum_{k=0}^{N} D_{NK} F_i(\xi_k) = 0, F_i(\xi_o) = 0 \tag{22}$$

where

$$A_{11} = D^3 + a_{1,i-1} D + a_{2,i-1}, A_{12} = 0, A_{13} = 0, \tag{23}$$

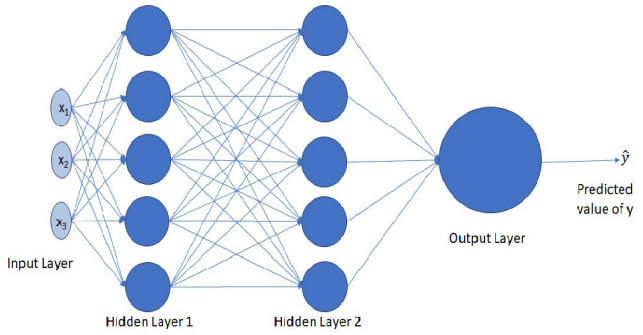


**FIGURE 2.** The architecture of artificial neural networks (ANNs). The input layer has predictive factors, while the output layer has predicted velocity. The hidden layers have a number of neurons.

## III. ARTIFICIAL NEURAL NETWORK MODELS AND STATISTICAL ANALYSIS

A fully connected feedforward ANN was used with two hidden layers as shown in Fig. (2). Three features were included in the input layer which were normalized Reynolds number, normalized angle of inclination, and normalized magnetic parameter. Two different ANNs, namely "Model 1" and "Model 2" were constructed with different numbers of neurons in hidden layers. Model 1 consists of two hidden layers with 5 and 3 neurons each. Whereas Model 2 consists of ten neurons in the first hidden layer and 5 neurons in the second hidden layer. In this ANN's Stochastic Gradient Descent (SGD) optimizer was used with learning rate = 0.02 and with momentum = 0.8. And 1000 epochs were used. The skin friction was chosen in the output layer. However, the current method of predicting the optimal skin friction was validated by comparing the simulated values with predicted values. The predicted performance was very close to the simulated and experimental performances as shown in Section (IV-B2).

In this present work Stochastic Gradient Descent optimization is used. Adaptive Learning Rate Method (Adam) is a fast converging optimizer, but sometimes leads to worse optimized values. i.e., sometimes it approaches local minima. Whereas the SGD may be a slower optimizer than the Adam, but converges to more appropriately than Adam, i.e., SGD converges to global optimization value.

### A. IN LAYER L

The number of layers (including the input and output layers) of a NN was denoted by $l$, runs from 1 to $L$. Where $l = 0$ represents the input layer, and $l = L$ represents the output layer. In each layer, the key components needed to implement a neural network are forward propagation and backward propagation.

### B. THE BASIC BUILDING BLOCKS OF FORWARD PROPAGATION ARE

Input: $a^{[l-1]}$

$$Z^{[l]} = W^{[l]} * a^{[l-1]} + b^{[l]}$$

Output: $a^{[l]} = \varphi\left(Z^{[l]}\right)$

### C. THE BASIC BUILDING BLOCKS OF BACKWARD PROPAGATION ARE

Input: $da^{[l]} \rightarrow cache\left(Z^{[l]}\right)$

Output: $da^{[l-1]} \rightarrow dW^{[l]}, db^{[l]}$

where $a^{[l]}$ is the activation in layer $l$, $W^{[l]}$, $b^{[l]}$ are the weight and bias vectors in layer $l$ of sizes $(l, l-1)$ and $(l, 1)$, respectively, $\varphi$ is the activation function. It is a non-linear function, like *Sigmoid*, *ReLU*, *LeakyReLU*, *Tanh*. In this work, the Hyperbolic Tangent function (*Tanh*) is used as the activation function for the first and second hidden layers, while it is a linear function for the output layer.

The summary of the procedure in layer $l$ is given in Fig.(3) as a flowchart.

The model's structure is given below

Input → Linear → Tanh → Linear → Tanh → Linear → Linear → Output

The data set included 1502 samples with features $Re, \alpha, M, f$. They were randomly split into 70% for training sets, 15% for validation sets (hold-out cross-validation sets or development sets), and 15% for test sets. Here the development sets and the test sets were from the same distribution. The ANN model with 1000 epochs, and with SGD optimizer which used the learning rate = 0.02 and

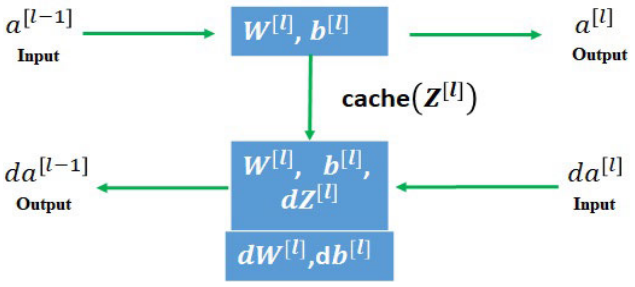


**FIGURE 3.** Basic blocks in layer *l*.

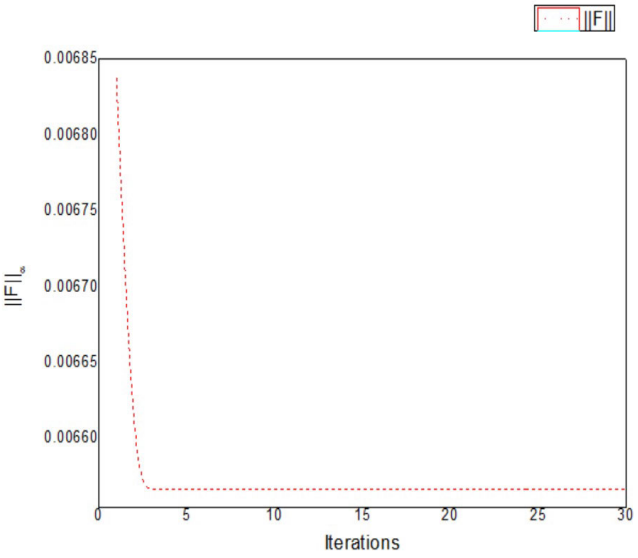


**FIGURE 4.** Error norm.

momentum 0.8, was kept on training on the training sets and evaluated solution by minimizing the objective function, i.e., the mean squared error function between the original values $y_i$ and predicted values $\hat{y}$ as follows

$$Loss = \sqrt{\frac{\sum_{i=1}^{m} (y_i - \hat{y}_i)}{m}} \tag{24}$$

The loss vs. epochs for the training sets is shown in Fig. (9). Then the development sets or hold-out cross-validation sets were used to see which of many different models perform best on these sets. For this, different learning rates and different momentum values were considered and finally, two models were finalized, namely "Model 1" and "Model 2". Then these models were evaluated on the test sets in order to get an unbiased estimate of how well these algorithms were done. From the data set it is seen that the train set error (MSE) is 0.000067, and the error of the development set is 0.000049. Since these errors were very near to zero, it has been concluded that there is no bias and variance problem for the present algorithm.

Further, to evaluate the performance of this neural network model, with the use of statistics, R squared $(R^2)$ was evaluated. $R^2$ is the square of the correlation between predicted $(\hat{y}_i)$ and true $(y_i)$ data. It ranges from 0 to 1.

**TABLE 1.** Rate of convergence of the method: $f(\eta)$ values for different iterations ($i$), for$M = 0$, $Re = 50$, $\phi = 0$ and $\alpha = -5$..

| $\eta$ | $f(\eta), i = 1$ | $f(\eta), i = 2$ | $f(\eta), i = 3$ | $f(\eta), i = 4$ |
|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 1 |
| 0.02447 | 0.999667 | 0.999663 | 0.999663 | 0.999663 |
| 0.09549 | 0.994899 | 0.994846 | 0.994845 | 0.994845 |
| 0.20611 | 0.975666 | 0.975419 | 0.975419 | 0.975419 |
| 0.34549 | 0.927797 | 0.927113 | 0.927113 | 0.927113 |
| 0.5 | 0.834599 | 0.833309 | 0.833309 | 0.833309 |
| 0.65451 | 0.682055 | 0.680431 | 0.680431 | 0.680431 |
| 0.79389 | 0.473534 | 0.472228 | 0.472228 | 0.472228 |
| 0.90451 | 0.246591 | 0.245946 | 0.245946 | 0.245946 |
| 0.97553 | 0.067877 | 0.067717 | 0.067717 | 0.067717 |
| 1 | 0 | 0 | 0 | 0 |

**TABLE 2.** The comparison between the NM, HPM of ( [12]) and SLM (present) solution for $f(0.5)$, where $\alpha = -5$, $Re = 180$, $M = 0$ and $\phi = 0$.

| | NM | HPM | SLM |
|---|---|---|---|
| $f(0.5)$ | 0.936032 | 0.936028 | 0.9360317189 |
| $f''(0.5)$ | −2.221403 | −2.221530 | 2.2214029821 |

**TABLE 3.** Effect of various parameters on skin friction $f''(0)$ for convergent channel.

| | | | | $Ag - H_2O$ | $Au - H_2O$ |
|---|---|---|---|---|---|
| M | Re | $\alpha$ | $\phi$ | $f''(0)$ | $(f''(0)$ |
| 0 | 100 | −5 | 0.01 | −0.68656 | −0.64402 |
| 20 | 100 | −5 | 0.01 | −0.68166 | −0.63978 |
| 50 | 100 | −5 | 0.01 | −0.67446 | −0.63356 |
| 100 | 100 | −5 | 0.01 | −0.66287 | −0.62354 |
| 50 | 50 | −5 | 0.01 | −1.08770 | −1.04165 |
| 50 | 100 | −5 | 0.01 | −0.67446 | −0.63356 |
| 50 | 200 | −5 | 0.01 | −0.42962 | −0.42417 |
| 50 | 100 | 0 | 0.01 | −1.99999 | −1.99999 |
| 50 | 100 | −3 | 0.01 | −0.99189 | −0.94353 |
| 50 | 100 | −5 | 0.01 | −0.67446 | −0.63356 |
| 50 | 100 | −7.5 | 0.01 | −0.48813 | −0.466147 |
| 50 | 100 | −5 | 0 | −0.71065 | −0.71065 |
| 50 | 100 | −5 | 0.01 | −0.67446 | −0.63356 |
| 50 | 100 | −5 | 0.02 | −0.64381 | −0.57602 |

**TABLE 4.** Thermo-physical properties of water and nanoparticles.

| Properties | $\rho(kg/m^3)$ | $\rho(J/KgK)$ | $k(w/mk)$ |
|---|---|---|---|
| Pure water | 997.1 | 4179 | 0.613 |
| $Ag$ | 10500 | 235 | 429 |
| $Au$ | 19282 | 129 | 310 |

$R^2$ equals one for highly correlated data and zero for non-correlated data. $R^2$ is given by:

$$R^2 = 1 - \sum_{i=1}^{m} \frac{(y_i - \hat{y}_i)^2}{(y_i - \overline{y_i})^2} \tag{25}$$

where $\overline{y}$ is the average of $y_i$'s

For the present algorithm, the evaluated $R^2$ values for the training sets, development sets, and test sets are 0.975959902044879, 0.9811696408944293, and 0.993198761285178, respectively.

## IV. RESULTS AND DISCUSSION

### A. RESULTS OF NUMERICAL SIMULATION

The governing equation of the current physics, which is the nonlinear differential equation (11) with boundary conditions

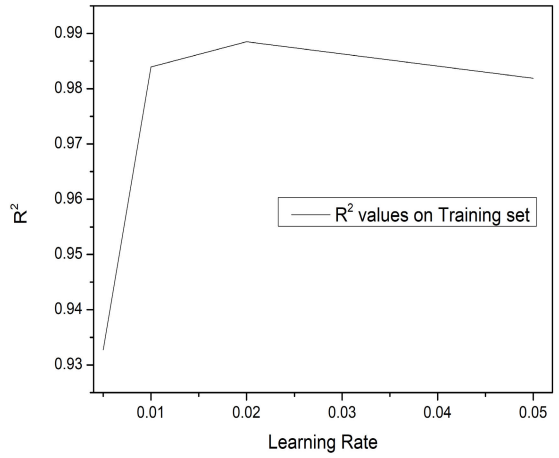


**FIGURE 5.** $R^2$ values for the different values of learning rate.

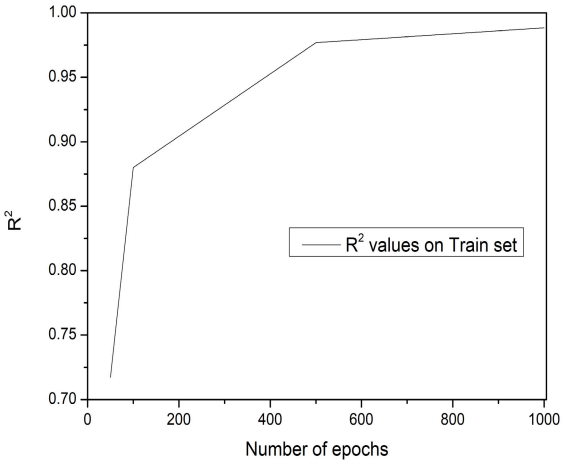


**FIGURE 6.** $R^2$ values for the epochs.

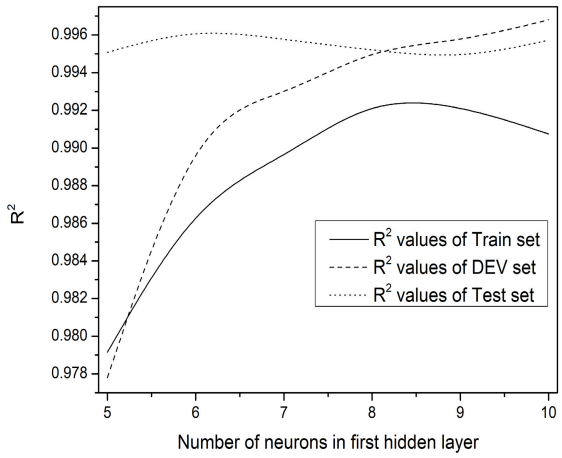


**FIGURE 7.** $R^2$ values for the different neurons in layer 1.

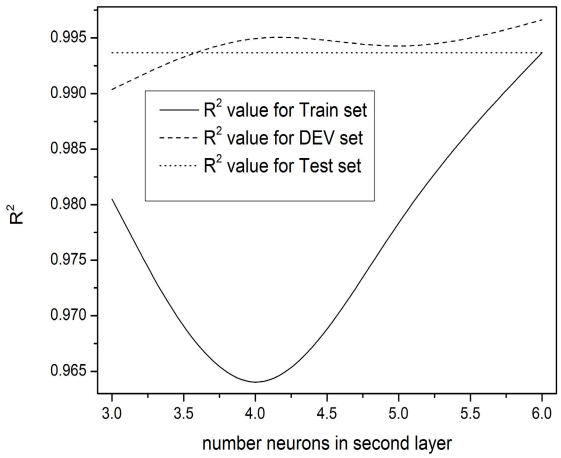


**FIGURE 8.** $R^2$ values for the different neurons in layer 2.

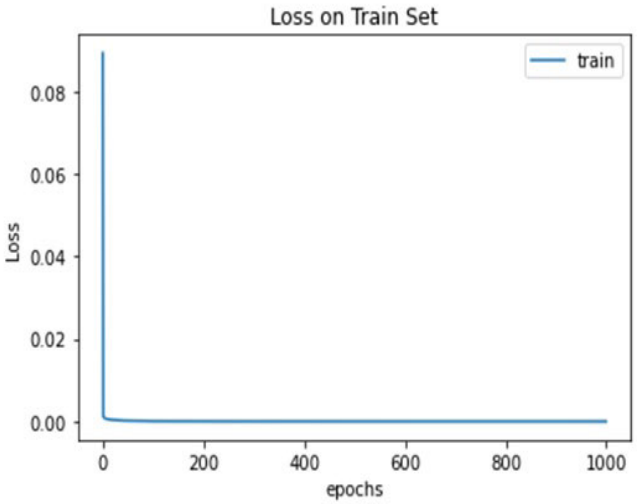


**FIGURE 9.** Model 1: Loss function on Train set.

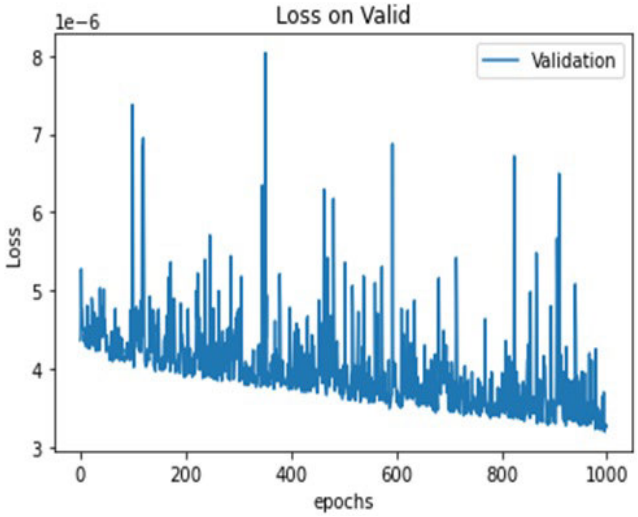


**FIGURE 10.** Model 1: Loss function on development set.

(14) doesn't have an analytical solution. Hence, the Eq. (11) was worked out numerically making use of the successive linearization method (SLM). A 101 × 101 grid-mesh was used to obtain the results which were found to be accurate in producing the solutions consistent to six decimal places.

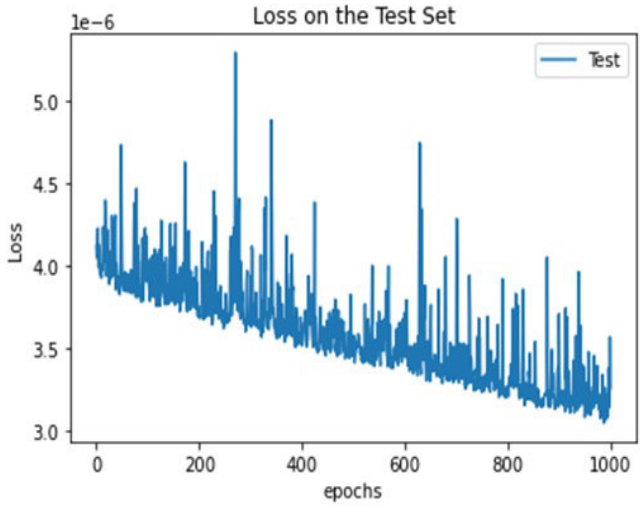


**FIGURE 11.** Model 1: Loss function on test set.

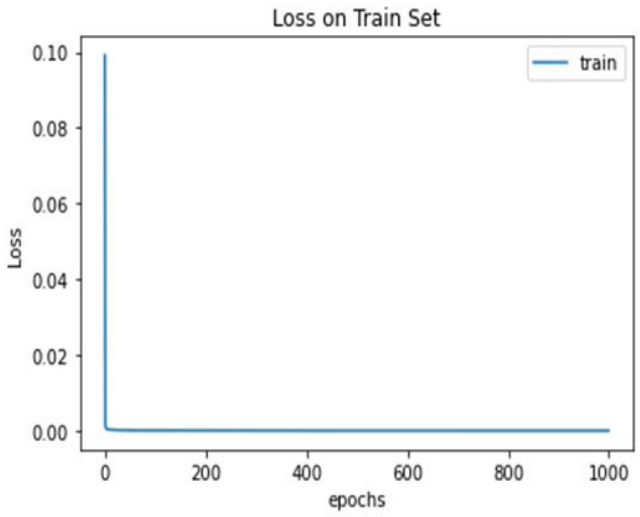


**FIGURE 12.** Model 2: Loss function on Train set.

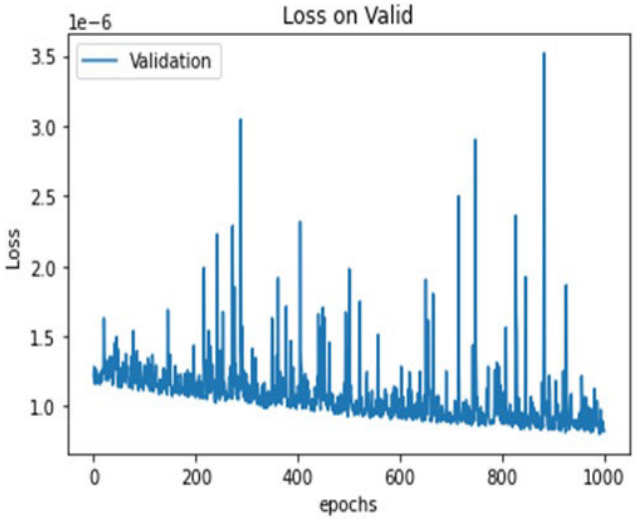


**FIGURE 13.** Model 2: Loss function on development set.

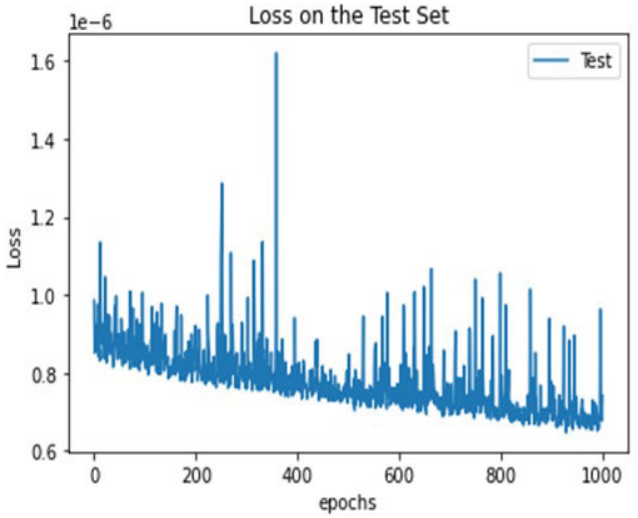


**FIGURE 14.** Model 2: Loss function on test set.

#### 1) RATE OF CONVERGENCE

The convergence rate of the present method has been provided in Table (1), and from this table, it can be adequate and confirms that the full convergence is achieved by the third order of SLM approximation. Further, the present results are validated with the results reported in [12] for $\alpha = -5$ and $Re = 180$ in Table (2). From Table (2) we can conclude that the results obtained by the present method are more accurate. The thermophysical properties of the base fluid water and nanoparticles are given in Table (4). The error norm of the solution is evaluated to investigate the convergence of the present numerical scheme. For this, the difference between the solution in successive iterations is considered, and we ensure the convergence of the method when the difference does not change even further iteration. The error norms of the solution are defined as

$$\|F\|_\infty = \max_{0\le i\le M_i} \|F_{r+1,i} - F_{r,i}\|_\infty \tag{26}$$

### B. RESULTS OF ANN MODELS

The ANNs were trained to evaluate $R^2$ values of the target variable for different values of learning rate (see Fig. (5)) and for different values of epochs (see Fig. (6)). Hence, the best values for learning rate and epochs were 0.02 and 1000, respectively.

The ANN was trained to evaluate $R^2$ values of the target variable on train, validation, and test sets for different numbers of neurons in the first hidden layer and also, in the second hidden layer. First, the algorithm learned for the various number of neurons, varies from 5 to 10 in the first hidden layer, by fixing the number of neurons as five in

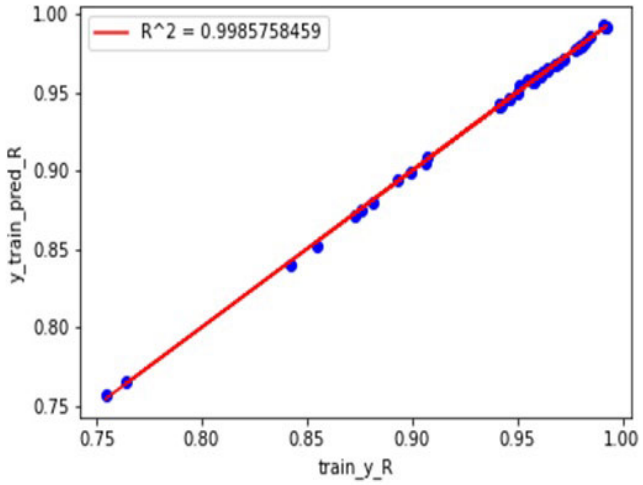


**FIGURE 15.** Model 1: $R^2$ value for the training set.

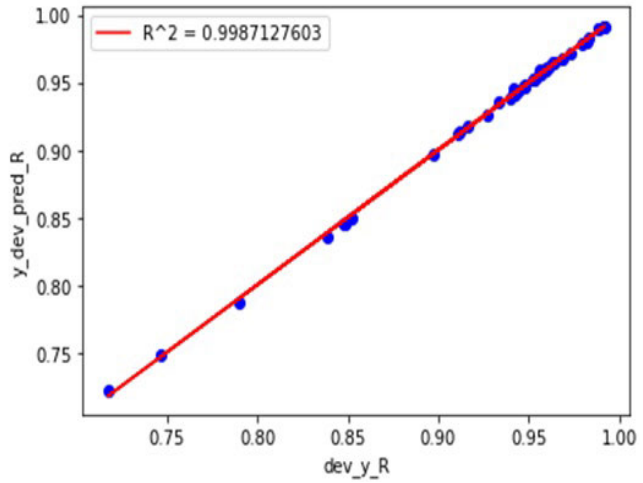


**FIGURE 16.** Model 1: $R^2$ value for the development set.

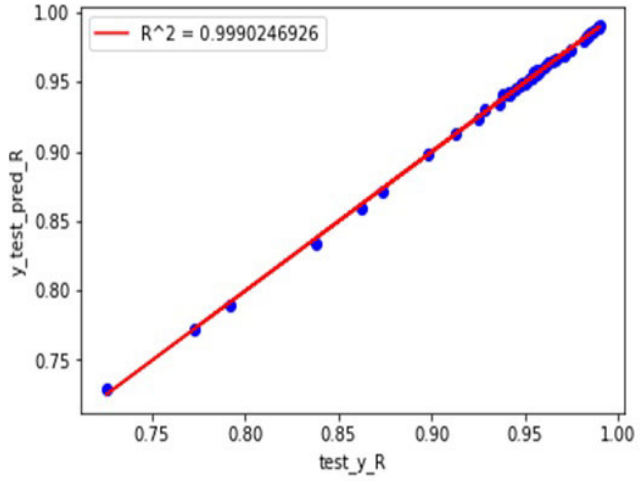


**FIGURE 17.** Model 1: $R^2$ value for the test set.

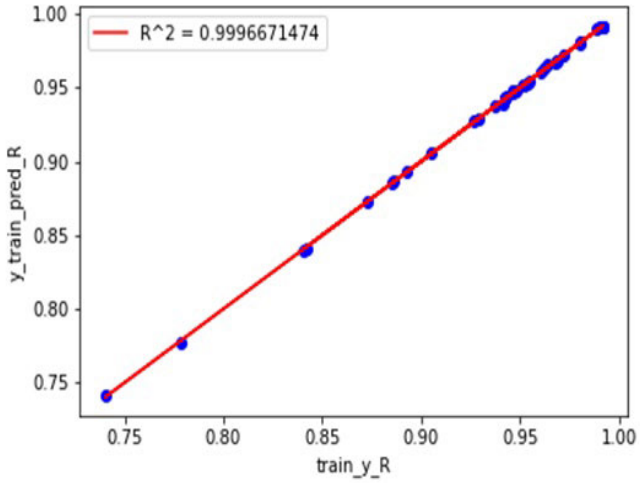


**FIGURE 18.** Model 2: $R^2$ value for the training set.

the second hidden layer (see Fig. (7)). Fig. (7) is evident to observe the performance of the algorithm from the training set → development set → test set. Further, the algorithm was learned for various numbers of neurons in the second layer, varying from 3 to 6, while fixing the number of neurons in the first layer five (see Fig.(8)). The performance of the algorithm can be seen in Fig. (8).

#### 1) ANN MODEL WITH 5 AND 3 NEURONS IN THE MODEL 1

The performance of Model 1 with five neurons in the first hidden layer and 3 neurons in the second hidden layer on the training, development, and test sets is shown in Figs. (9) - (11). Fig. (9) depicts the loss on the training set for 1000 epochs. As the number of epochs increases the loss is reduced to zero. i.e., the predicted values ($\hat{y}$) coincide with the original values ($y_i$). Also, this can be validated with the value $R^2 = 0.9985758459$ obtained from the training set for 1000 epochs. Fig. (10) shows the loss on the development set (validation set) for 1000 epochs. This figure depicts that the loss lies in (0.000003, 0.000004). As expected the loss decreases for the increasing values of epochs. Fig. (11) shows the loss on the test set for 1000 epochs. From this figure, it is evident that the loss lies in the interval (0.000003, 0.0000035). It is evident from Figs. (15) - (17) that the model's $R^2$ value for training set is 0.9985758459, whereas it's 0.9987127603 for validation set and 0.9990246926 for test set. These values of $R^2$ for the train, validation, and test sets explain that the performance of Model 1. Because $R^2 = 0$ shows that there is no correlation between the original and validated values of the target variable, and $R^2 = 1$ shows the perfect correlation between the original and expected values of the target variable.

#### 2) ANN MODEL WITH 10 AND 5 NEURONS IN THE MODEL 2

The performance of Model 2 with ten neurons in the first hidden layer and five neurons in the second hidden layer on the training, development, and test sets is shown in Figs. (12) - (14). Fig. (12) depicts the loss on the training

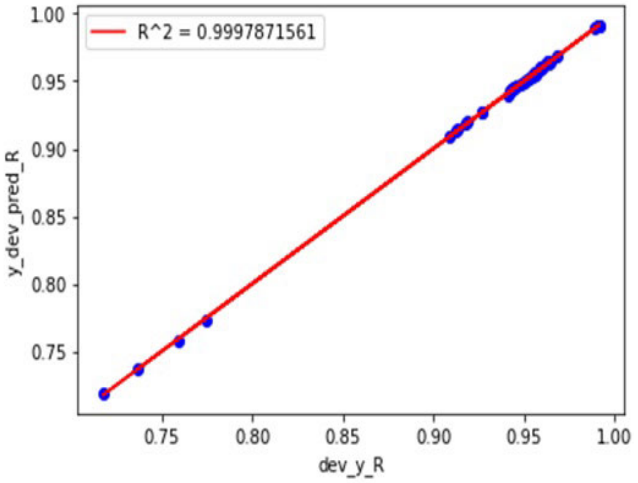


**FIGURE 19.** Model 2: $R^2$ value for the development set.

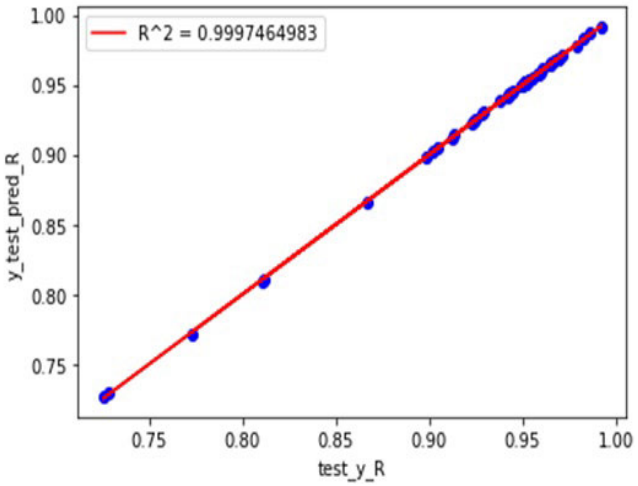


**FIGURE 20.** Model 2: $R^2$ value for the test set.

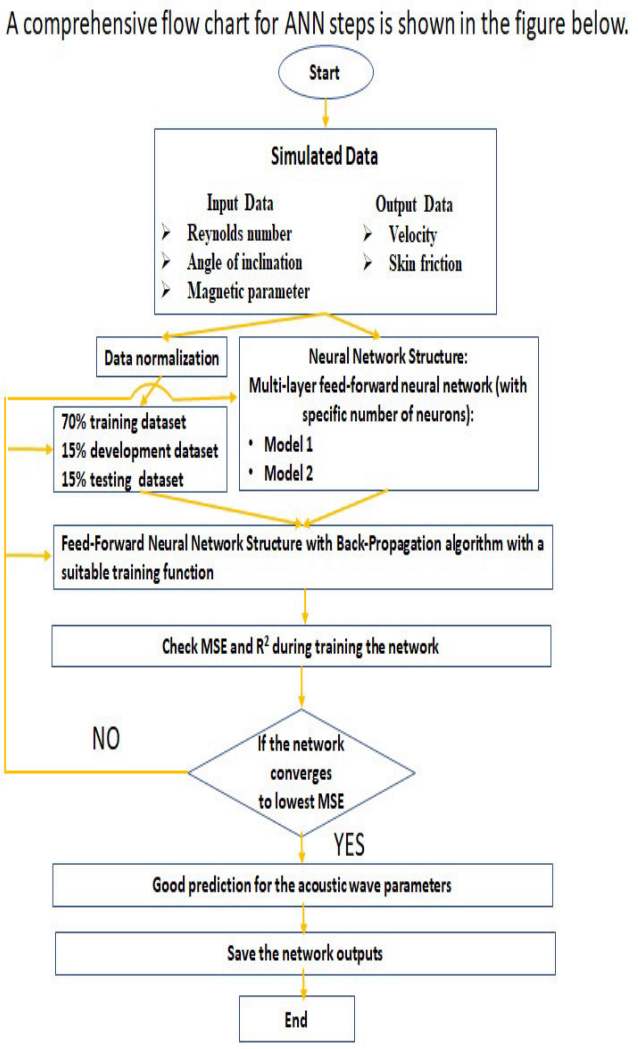


**FIGURE 21.** A comprehensive flow chart for ANN steps.

set for 1000 epochs. As the number of epochs increases the loss function decreases to zero. From the first epoch onwards the loss remains the same, till the 1000th epoch. This loss on the train set in model 2 is the same as the loss in model 1. The $R^2 = 0.9996671474$ from model 2 is better than the $R^2 = 0.9985758459$ from model 1. Fig. (13) shows the loss on the development set for 1000 epochs. This figure shows that, as the number of epochs increases the loss function decreases. This figure depicts that the loss lies between (0.000001, 0.0000015). This loss is smaller than the loss obtained in model 1 with 5, 3 neurons in hidden layers. Fig. (14) shows the loss on the test set for 1000 epochs. This figure portrays that as the number of epochs increases the loss function decreases. It is evident from Figs. (18) - (20) that the model's $R^2$ value for training set is 0.9996671474, whereas it's 0.9997871561 for validation set and 0.9997464983 for test set.

We can conclude that the model 2 performed extremely well to the model 1 on the data set.

## V. CONCLUSION

In the present work, the governing equations are solved using the superior numerical method of successive linearization. Using this simulation a data set of size $1502 \times 3$ is generated for a convergent channel for the target variable skin friction and velocity. Two ANN models were built to fit the target variables and these models were optimized by stochastic descent gradient (SGD). From these ANN models the following conclusions are made.

- Artificial neural networks can accurately fit the target functions and variables. The mean square errors (MSE) between the predicted values and true values for the training data set, developing data set, and testing data set through Model 2 are 0.0000011206, 0.0000008127, and 0.0000007536, respectively.
- These MSE values showed that the ANNs accurately fit the target functions.
- The results showed that the use of Model 2 (10 and 5 neurons in the first and second hidden layers,

respectively) was better than the use of Model 1 (5 and 3 neurons in the first and second hidden layers, respectively).

- The high predictability of ANNs suggests that neural networks can be adopted in future research for modeling the described flow in the given geometry minimizing the need for conducting time-consuming and costly experiments.
- For the increasing values of magnetic parameter the skin friction increases in both the nanofluids.
- Skin friction increases for the increasing values of Raynold number in both the nanofluids.
- Skin friction increases in both the nanofluids for the increasing values of angle of inclination ($\alpha$) and volume fraction rate ($\phi$)

## REFERENCES

[1] G. B. Jeffery, "The two-dimensional steady motion of a viscous fluid," *Philos. Mag. Lett.*, vol. 29, no. 6, pp. 455–465, Apr. 1915.

[2] I. R. Petroudi, D. D. Ganji, M. K. Nejad, J. Rahimi, E. Rahimi, and A. Rahimifar, "Transverse magnetic field on Jeffery–Hamel problem with Cu-water nanofluid between two non-parallel plane walls by using collocation method," *Case Stud. Thermal Eng.*, vol. 4, pp. 193–465, Nov. 2014.

[3] L. R. Head, "The steady two-dimensional radial flow of viscous fluid between two inclined plane walls," *Proc. Roy. Soc. London. Ser. A, Math. Phys. Sci.*, vol. 175, pp. 436–464, Jul. 1940.

[4] D. W. Lexi, "Note on the divergent flow of fluid," *Philos. Mag. Lett. J. Sci.*, vol. 18, no. 121, pp. 759–777, 1934.

[5] J. E. Green, "The second Goldstein lecture. Modern developments in fluid dynamics—An addendum," *Aeronaut. J.*, vol. 96, no. 953, pp. 69–86, Mar. 1992.

[6] W. Axford, "The magneto hydrodynamic Jeffery–Hamel problem for a weakly conducting fluid," *Quart. J. Mech. Appl. Math.*, vol. 14, pp. 335–351, Aug. 1961.

[7] L. Fraenkel and H. B. Squire, "Laminar flow in symmetrical channels with slightly curved walls, I. On the Jeffery–Hamel solutions for flow between plane walls," *Proc. Roy. Soc. London. Ser. A, Math. Phys. Sci.*, vol. 267, pp. 119–138, Apr. 1962.

[8] U. Mendu, "An application of spectral linearisation method on the steady two-dimensional radial flow of Au-water and Ag-water nanofluids between two inclined plane walls," *Int. J. Math. Model. Numer. Optim.*, vol. 13, no. 1, pp. 49–63, 2023.

[9] S. S. Motsa, P. Sibanda, F. G. Awad, and S. Shateyi, "A new spectral-homotopy analysis method for the MHD Jeffery–Hamel problem," *Comput. Fluids*, vol. 39, no. 7, pp. 1219–1225, Aug. 2010.

[10] R. V. Surendra Mani and U. Mendu, "Application of successive linearization method on steady radial flow of nanofluids between inclined plane walls," *J. Nanofluids*, vol. 11, no. 6, pp. 952–961, Dec. 2022.

[11] S. M. Moghimi, D. Ganji, H. Bararnia, H. Hosseini, and M. Jalaal, "Homotopy perturbation method for nonlinear MHD Jeffery–Hamel problem," *Adv. Eng. Soft.*, vol. 61, no. 8, pp. 108–113, Apr. 2011.

[12] Z. Z. Ganji, D. D. Ganji, and M. Esmaeilpour, "Study on nonlinear Jeffery–Hamel flow by he's semi-analytical methods and comparison with numerical results," *Comput. Math. Appl.*, vol. 58, nos. 11–12, pp. 2107–2116, Dec. 2009.

[13] A. A. Khidir, "A new spectral-homotopy perturbation method and its application to Jeffery–Hamel nanofluid flow with high magnetic field," *J. Comput. Methods Phys.*, vol. 2013, Dec. 2013, Art. no. 939143, doi: 10.1155/2013/939143.

[14] I. R. Petroudi, D. D. Ganji, M. K. Nejad, J. Rahimi, E. Rahimi, and A. Rahimifar, "Transverse magnetic field on Jeffery–Hamel problem with Cu-water nanofluid between two non parallel plane walls by using collocation method," *Case Stud. Thermal Eng.*, vol. 4, pp. 193–201, Nov. 2014.

[15] W. S. McCulloch and W. Pitts, "A logical calculus of ideas immanent in nervous activity," *Bull. Math. Biol.*, vol. 52, nos. 1–2, pp. 99–115, 1990.

[16] M. Rezaeianzadeh, H. Tabari, A. A. Yazdi, S. Isik, and L. Kalin, "Flood flow forecasting using ANN, ANFIS and regression models," *Neural. Comput. Applic.*, vol. 25, pp. 25–37, Jul. 2014.

[17] S. Fidan, H. Oktay, S. Polat, and S. Ozturk, "An artificial neural network model to predict the thermal properties of concrete using different neurons and activation functions," *Adv. Mater. Sci. Eng.*, vol. 2019, pp. 1–13, Apr. 2019, doi: 10.1155/2019/3831813.

[18] A. Berber, M. Gürdal, and K. Bagirsakci, "Prediction of heat transfer in a circular tube with aluminum and Cr-Ni alloy pins using artificial neural network," *Exp. Heat Transf.*, vol. 34, no. 6, pp. 547–563, 2021.

[19] M. Gürdal, "Artificial intelligence approach for energy and entropy analyses of $NiFe_2O_4/H_2O$ nanofluid flow in a tube with vortex generator," *Eng. Anal. Boundary Elements*, vol. 152, pp. 277–292, Jul. 2023.

[20] J. Raza, M. Raza, T. Mustaq, and M. I. Qureshi, "Supervised machine learning techniques for optimization of heat transfer rate of Cu–$H_2O$ nanofluid flow over a radial porous fin," *Multidiscip. Model. Mater.*, vol. 19, no. 4, pp. 680–706, 2023.

[21] M. C. Louay, M. Pedro, and Z. Gregory, "Heat transfer and friction in helically-finned tubes using artificial neural networks," *Adv. Multiphase Flow Heat Transf.*, vol. 2, no. 1, pp. 62–106, 2009.

[22] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.

[23] H. C. Brinkman, "The viscosity of concentrated suspensions and solutions," *J. Chem. Phys.*, vol. 20, no. 4, p. 571, Apr. 1952.

[24] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics*. Berlin, Germany: Springer, 1988.

[25] L. N. Trefethen, *Spectral Methods in MATLAB*. Philadelphia, PA, USA: SIAM, 2000.

**UPENDAR MENDU** received the B.Sc. degree from Kakathiya University, Telangana, India, in 1996, and the M.Sc. degree in applied mathematics and the Ph.D. degree in mathematics [computational fluid dynamics (CFD)] from the National Institute of Technology, Warangal, Telangana, in 1998 and 2014, respectively. He is currently an Assistant Professor with the Department of Mathematics, ICFAI Deemed to be University (ICFAI Foundation for Higher Education), Hyderabad, India. His current research interests include nanofluids, data science, machine learning, computational fluid mechanics, convective heat, and mass transfer.

**VEMAN SAI PRABHATH MENDU** received the B.Tech. degree in electronics and communications engineering from the Indian Institute of Information Technology (IIIT), Chittoor, Andhra Pradesh, India, in 2021, the M.S. degree in big data analytics and information technology from the University of Central Missouri, Warrensburg, MO, USA, in 2023. His current research interests include data science, machine learning, deep learning, computer vision, and natural language processing.