Co-published by
Institute of Fluid-Flow Machinery
Polish Academy of Sciences
Committee on Thermodynamics and Combustion
Polish Academy of Sciences

Copyright©2026 by the Authors under licence CC BY-NC-ND 4.0

http://www.imp.gda.pl/archives-of-thermodynamics/

Physics-Informed Deep Learning Optimisation for Heat Transfer Analysis of Non-Similar Flow over Porous Moving Surface

Syed Shahzad Ali[a], Muhammad Imran Khan[a,b*], Mohammed Almakki[b], Ahmed Zeeshan[a,c]

[a]Department of Mathematics and Statistics, International Islamic University Islamabad, 44000, Pakistan
[b]School of Engineering, Architecture and Interior Design, Amity University Dubai, P.O. Box 345019, Dubai International Academic City, United Arab Emirates
[c]Department of Mathematics, College of Science, Korea University, 145 Aman-ro, Seongbuk-gu, Seoul 02841 Republic of Korea
[*]Corresponding author email: muhammad.phdma132@iiu.edu.pk

Received: 20.12.2025; revised: 21.06.2026; accepted: 30.06.2026

Abstract

The investigation of fluid flow and thermal behaviour near continuous moving surfaces is an active and important area of research with direct applications across numerous engineering processes. Physics-informed neural networks gain a significant role in fluid dynamics and engineering applications due to their ability to integrate physical laws directly into the learning process of neural networks. This work expands the use of physics-informed neural networks to address thermal and momentum boundary layer issues of non-Newtonian fluid. The primary concern is to investigate momentum and thermal boundary layer under tangent hyperbolic non-Newtonian fluid flow circumstances. Due to the shear thinning and shear thickening behaviour of the non-Newtonian fluid, it has wide application in various domains. The physical constraints of the governing equations are solved using physics-informed neural networks. The models are constructed and trained using TensorFlow, and the predictions produced by this method are contrasted with those derived using Richardson extrapolation and adaptive moment estimation optimisation. The adaptive moment estimation optimiser is a well-known gradient-based optimisation technique that is frequently used in fluid dynamics, particularly when governing equations are solved using machine learning, deep learning, or physics-informed neural networks. The impact of various parameters has been analysed and shown graphically. The Lorentz force opposes fluid motion, producing a retarding effect that decreases velocity. The Darcy and Forchheimer drag forces increase with decreasing porosity, resulting in substantial momentum suppression and less convective heat evacuation, which raises the local temperature.

Keywords: Physics-informed neural networks; Python programming; Non-Newtonian fluid; Inclined magnetic field; Mixed convection

Vol. 47(2026), No. 3, 111–123; doi: 10.24425/ather.2026.7508

Cite this manuscript as: Ali, S.S., Khan, M.I., Almakki, M., & Zeeshan, A. (2026). Physics-Informed Deep Learning Optimisation for Heat Transfer Analysis of Non-Similar Flow over Porous Moving Surface. *Archives od Thermodynamics*, 47(3), 111−123.

1. Introduction

Artificial intelligence is the domain of emerging computers and robots that are capable of behaving in ways that both mimic and go beyond human capabilities. Programs having artificial intelligence (AI) capabilities can interpret and analyse data to provide information or initiate activities automatically without human intervention. Machine learning is a branch of artificial intelligence that employs algorithms to automatically extract knowledge and identify trends from data, then use that knowledge to make ever-better decisions. Machine learning has a broad range of applications, including prognostication, design and speech recognition, optimisation and image processing [1−3]. A neural network is an artificial intelligence based model

Nomenclature

a – radius of the cylinder, m
b_i – bias term of network
B_0 – magnetic field strength, Wb/m^2
C_p – specific heat capacity, J/(kg K)
C_f – skin friction, Pa
Da – Darcy parameter
Ec – Eckert number
g – gravitational acceleration, m/s^2
Gr – Grashof number
k_f – fluid thermal conductivity, W/(m K)
K – permeability, m^2
L^2 – error
L_{data}, $L_{physics}$, L_{BC} – data loss, physics loss, boundary condition loss
M – magnetic parameter
Nu – Nusselt number
p – pressure, Pa
Pr – Prandtl number
q_w – wall heat flux, W/m^2
Q – heat generation and absorption parameter
Rd – thermal radiation parameter
Re – Reynolds number, Re $= Dv\rho\ /\mu$
R_f, R_θ, R_{BC} – residuals of momentum equation, energy equation and boundary condition
t – time, s
T – temperature, K
T_w – surface temperature, K
T_∞ – ambient temperature, K
u, v – velocity components, m/s
U_∞ – ambient velocity, m/s
w – weight of the network
We – Weissenberg number
x, y – horizontal and vertical coordinate axes, m

Greek symbols

α – thermal diffusivity, m^2/s
ϵ – required tolerance
μ – dynamic viscosity, Pa/s
ν – kinematic viscosity, m^2/s
ξ, η – dimensionless variables
ρ – density, kg/m^3
σ – electrical conductivity, S/m
τ_w – wall shear stress, Pa
θ – dimensionless temperature
φ – magnetic inclination angle, deg

Abbreviations and Acronyms

ADAM – adaptive moment estimation
AI – artificial intelligence
FCN – fully connected network
ML – machine learning
MSE – mean squared error
ODEs – ordinary differential equations
PDEs – partial differential equations
PINN – physics-informed neural network

that is inspired by the brain's structure and operation. It possesses enormous applications in many sectors, such as computational materials and chemical sciences [4,5], pharmaceutical sector [6], biomedical sciences [7], climate forecasting [8], robotics [9], and aerospace and mechanical engineering [10,11]. The neural network is motivated by the universal approximation theorem, which asserts that a feedforward neural network with at least one hidden layer may approximate any continuous function within some level of accuracy, especially for simple functions [12]. A neural network is one of the main supervised machine learning models, wherein input data is transformed into associated outputs. Training is performed using a given dataset of input-output pairs to modify the network parameters such that the difference between estimated and actual values is minimised. The method of convergence is through the optimisation of a loss function, usually quantified by mean squared error (MSE). This optimisation process relies on gradient-based algorithms with automatic differentiation for the efficient backpropagation of errors and the tuning of the network's performance, Baydin et al. [13].

Figure 1 shows that the artificial intelligence is a broader concept, however, machine learning is a subset of AI. Supervised learning may also be used to solve differential equations. To solve a differential equation means that one is seeking a mapping of input physical variables provided to corresponding unknown physical values, which are the estimated solution. A neural network is learned to approximate this solution in a non-linear way so that it can estimate the solution of the equation at any input point. Two major drawbacks are, however, present: first, the network has difficulty extrapolating solutions for input variables outside the range of training data. Second, the training procedure can converge failing when dealing with a large dataset.

Physics-informed neural network (PINN**)** is a machine learning framework that incorporates the laws of physics into the training section of a neural network. PINNs are designed by incorporating governing physical equations into the loss function as constraints, which helps to ensure precise output predictions even in areas with limited or no data. This section presents various strategies to apply PINNs to complex systems and problems. PINN involves minimising the residual of the governing differential equation evaluated at various points within the domain. Boundary conditions are incorporated either by announcing soft constraints with corresponding boundary data values in the minimisation process or by strictly enforcing the solution with hard constraints. To get over this restriction, more physics-based data are integrated into the classical neural network. In simulating physical and engineering problems represented by

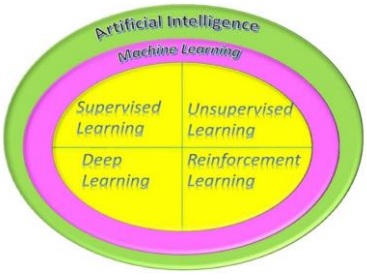


Fig. 1. Flow chart representation of artificial intelligence and machine learning.

differential equations, these methods are commonly referred to as physics-informed neural networks (PINNs). In these methods, the given set of data points, which are called collocation points, is utilised so that the learned solution obeys the governing differential equations. A new loss function, called physics loss, is added to the original loss function of the learning system. This second constraint punishes the training process, making the solution both partly driven by the initial data and partly by the physics underlying it, thus considerably limiting the solution space. With partial differential equations, little known data is needed. Therefore, PINN methods are regarded as more physics-informed than data-driven methods. Cuomo et al. [14] investigated the use of PINNs to approximate solutions to initial value problems. Bararnia and Esmaeilpour [15] examined thermal analysis of boundary layer flow with PINNs. They concluded that Pr in a heat equation is a crucial variable whose value significantly alters the number of artificial neurons and layers needed to produce the intended effects. Due to the fascinating phenomena and applications of neural networks and PINNs in the fluid mechanics domain, there are many researchers who pay attention to scrutinising an approximate solution of a fluid problem [16−21].

Non-Newtonian fluids are important in engineering and manufacturing industries because of their special physical properties, especially their viscosity, which is dependent on applied forces. These fluids find their application in manufacturing protective gear like body armour and safety helmets. Visco-plastic fluids possess dual characteristics – behaving like solids when subject to high stress and as fluids when subject to moderate stress. For instance, they minimise boundary layer thickness of airfoils, turbine blades and ship hulls for lower drag force, resulting in considerable savings in fuel consumption and improved efficiency. The first approximation theory was developed by Prandtl [22], who described a mathematical basis flow with a high Reynolds number and derived equations for a two-dimensional boundary layer. Finding a closed-form solution for a governing equation and dealing with boundary layer theory has never been a simple task. These governing equations are numerically solved using a variety of techniques, such as the shooting method, homotopy analysis methodology and finite difference method. Non-Newtonian fluids play a significant role in engineering and manufacturing industries owing to their unique physical properties. Numerous items, including safety gear and body armour, are manufactured using non-Newtonian fluids. Peng et al. [23] have discussed the use of non-Newtonian fluids. The topics of [24,25] include fluid flow and heat transfer enhancement of non-Newtonian fluid flow problems. Numerous research that uses different geometrical structures and computational methods to examine the non-Newtonian fluid model can be identified in [26−30]. Numerical investigation of heat transfer and boundary layer flow of a semi-infinite vertical plate with slip condition was made to evaluate rheological characteristics of Casson fluid [31]. Ramachandra Prasad et al. [32] examined non-Newtonian flow and heat transfer over a partial slip horizontal cylinder using an implicit finite difference method. For more interesting and advanced literature, we can view references [33−37].

In the domains of fluid dynamics, PINNs are used to solve Navier-Stokes equations, boundary layer flows, and to simulate complex phenomena like turbulence and multiphase flows. In the field of heat and mass transfer, PINNs are applied to solve problems involving conduction, convection, radiation and phase-change processes. PINNs are used in aerodynamics to forecast lift, drag, and pressure distributions with lower computing costs in airfoil and wing simulations. In order to build structures that maximise aerodynamic performance while minimising resistance, they are frequently employed in shape optimisation jobs. In aerodynamics, cylinder structure is applied to make design of aircraft and rockets.

Therefore, the main objective of this research work is to implement physics-informed neural networks (PINNs) to study non-similar boundary layer flow of tangent hyperbolic fluid over a cylinder with buoyancy effects. The author applied the adaptive moment estimation (ADAM) optimiser to evaluate an approximate solution of non-similar tangent hyperbolic fluid over a cylinder. The Python platform is implemented for programming purposes.

2. Mathematics-based system modelling

In this investigation, we considered steady, two-dimensional flow and heat transfer behaviour of a non-Newtonian tangent hyperbolic fluid along a horizontal cylinder with thermal radiation and Joule heating effects. The fluid is presumed to be incompressible and subjected to the tangent hyperbolic model, taking into consideration shear-thinning characteristics. Figure 2, shows the flow problem. Here, a represents the cylinder radius. Since ($0 \le \phi \le \pi$), $\phi = x/a$ is a y-axis angle. Also, a constant magnetic field with strength B_0 is imposed in the radial direction of the cylinder. The symbol g represents gravity, and T_w is a constant temperature.

With the above assumptions, the governing equations are as follows [31−35]:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \tag{1}$$

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \nu\left[(1-n)\left(\frac{\partial^2 u}{\partial y^2}\right) + \sqrt{2}\,n\left(\frac{\partial u}{\partial y}\right)\left(\frac{\partial^2 u}{\partial y^2}\right)\right] + -\frac{\nu}{k}u + Cu^2 + g\beta sin\left(\frac{x}{a}\right)(T - T_\infty) - \frac{\alpha B_0^2}{\rho}\sin^2\varphi\, u, \tag{2}$$

$$u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y} = \alpha\frac{\partial^2 T}{\partial y^2} - \frac{1}{\rho C_p}\frac{\partial q_x}{\partial y} + \frac{\sigma}{\rho C_p}\sin^2\varphi\, B_0^2 u^2 + + \frac{\mu}{\rho C_p}\left(\frac{\partial u}{\partial y}\right)^2 + \frac{Q_0}{\rho C_p}(T - T_\infty). \tag{3}$$

Boundary conditions to solve Eqs. (1)−(3) are defined as:

$$y = 0: \qquad u = 0, \quad v = 0, \quad T = T_w, \tag{4a}$$

$$y = \infty: \qquad u \to 0, \quad v \to 0, \quad T = T_\infty. \tag{4b}$$

Non-similarity transformations are given below:

$$\xi = \frac{x}{a}, \qquad \eta = \frac{y}{a}\sqrt[4]{\mathrm{Gr}}, \qquad f(\xi,\eta) = \frac{\psi}{\nu_f \xi \sqrt[4]{\mathrm{Gr}}},$$

$$\theta(\xi,\eta) = \frac{T - T_\infty}{T_w - T_\infty}, \qquad \mathrm{Gr} = \frac{g\beta_f (T_w - T_\infty)a^3}{{\nu_f}^2}. \tag{5}$$

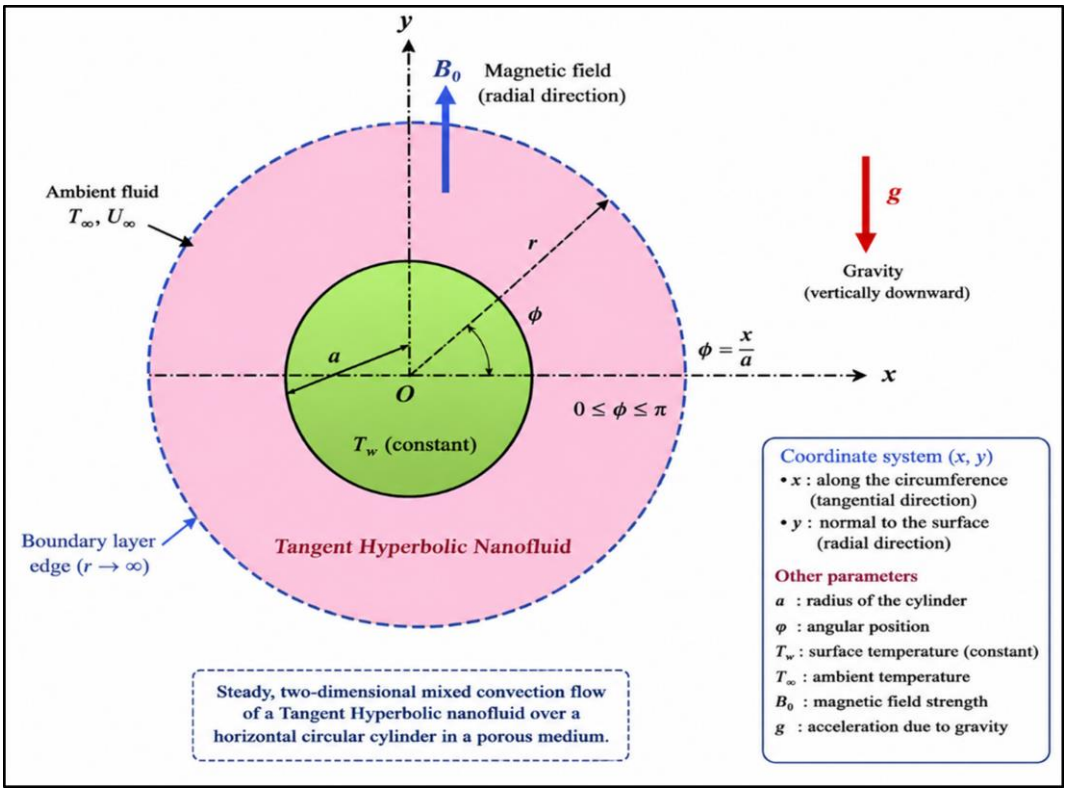


Fig. 2. Schematic flow diagram of tangent hyperbolic flow over a horizontal cylinder.

While applying the non-similarity transformation on the governing equations, the continuity equation is inherently satisfied, whereas the remaining Eqs. (2)–(3) are obtained as:

$$(1-n)f''' + n\mathrm{We}ff'' - (1+\xi\Lambda)f'^2 + \\ -\left(M\sin^2\varphi + \frac{1}{\mathrm{Da}}\right)f' + \frac{\sin\xi}{\xi}\theta = \xi\left(f'\frac{\partial f'}{\partial\xi} - f''\frac{\partial f}{\partial\xi}\right), \quad (6)$$

$$\frac{1}{\mathrm{Pr}}\left[1+\frac{4}{3}Rd\right]\theta'' + f\theta' + \sin^2\varphi J\xi^2 f'^2 + \\ + \mathrm{Ec}\xi^2 f''^2 + Q\theta = \xi\left(f'\frac{\partial\theta}{\partial\xi} - \theta'\frac{\partial f}{\partial\xi}\right). \quad (7)$$

The boundary conditions are transformed as follows:

$$\eta = 0: \qquad f' = 0, \quad f = 0, \quad \theta = 1, \quad (8a)$$

$$\eta \to \infty: \qquad f' = 0, \quad \theta = 0. \quad (8b)$$

The flow input parameters are $M = \frac{\sigma_f {B_0}^2 a^2}{\rho_f \nu_f \sqrt{\mathrm{Gr}}}$, represents magnetic field intensity, the Prandtl number is defined as $\mathrm{Pr} = \frac{\nu_f}{\alpha_f}$, $\mathrm{Da} = \frac{K\sqrt{\mathrm{Gr}}}{a^2}$ is the Darcy parameter, lambda (Λ) is equal to Ca, the radiation parameter $Rd = \frac{4\sigma^* {T_\infty}^3}{k^* k_f}$, $Q = \frac{Q_0 a^2}{(\rho C_p)_f \nu_f \mathrm{Re}}$ is the heat generation and absorption parameter, the Eckert number $\mathrm{Ec} = \frac{\mu {\nu_f}^2 \mathrm{Gr}}{a^2 C_p (T_w - T_\infty)}$, and the Weissenberg number is denoted by $\mathrm{We} = \frac{\sqrt{2}\nu\Lambda x \mathrm{Gr}^{3/4}}{a^3}$.

Now we establish two significant flow output responses in fluid and heat transfer analysis, i.e. C_f and Nu. Skin friction is the resistance offered by a fluid due to viscosity when it flows over a surface. The Nusselt number (Nu) indicates the effectiveness of convective heat transfer relative to conduction. Both parameters are vital in the optimisation of designs for fluid flow and heat exchange systems, and they are defined as:

$$C_f = \frac{2\tau_w}{\rho_f u_\infty^2}, \qquad \mathrm{Nu}_x = \frac{x q_w}{k_f (T_w - T_\infty)}, \quad (9)$$

where τ_w and q_w are:

$$\tau_w = \left[\mu\left(\frac{\partial u}{\partial y}\right) + \frac{1}{\beta c}\left(\frac{\partial u}{\partial y}\right) - \frac{1}{6\beta c^3}\left(\frac{\partial u}{\partial y}\right)^3\right]_{y=o}, \quad (10)$$

$$q_w = -k_f\left[1 + \frac{16\sigma^* T_\infty^3}{3k^* k_f}\right]\left(\frac{\partial T}{\partial y}\right)_{y=0}. \quad (11)$$

By simplifying, Eq. (9) becomes:

$$C_f \mathrm{Gr}^{-\frac{3}{4}} = (1-n)\xi f''(\xi,0) - \frac{n}{2}\mathrm{We}\xi\left(f''(\xi,0)\right)^2, \quad (12)$$

$$\frac{\mathrm{Nu}}{\sqrt[4]{\mathrm{Gr}}} = -\theta'(\xi,0). \quad (13)$$

3. Local non-similarity method

In dealing with non-similarity boundary layers, a widely used concept is the idea of local similarity. Within this concept, the right-hand sides of Eqs. (6) and (7) are presumed to be sufficiently small, essentially replacing them with approximations to zero. This keeps the computational exercise simple, generating a system of ordinary differential equations. As such, the local similarity solution, or the first level of truncation, has computa-

tional attractiveness but may provide numerical answers of questionable accuracy. The ambiguity stems from the question of whether to ignore the right-hand side of the equations when ξ is not small. To alleviate this shortcoming, Minkowycz and Sparrow [38], Sparrow and Yu [39] proposed a local non-similarity technique for finding solutions to non-similar boundary layer equations. For obtaining the local non-similarity solution of Eqs. (9)–(16), we introduce new dependent variables as addressed in [38,39].

3.1. First level of truncation

Assuming that ξ is very small ($\xi \ll 1$), derivatives of ξ are ignored, and Eqs. (6)–(8) are transformed into:

$$(1+n)f''' - (1+\xi\Lambda)f'^2 + n\mathrm{We}f''f''' + \\ -\left(M\sin^2\varphi + \frac{1}{\mathrm{Da}}\right)f' + \frac{\sin\xi}{\xi}\theta = 0, \quad (14)$$

$$\frac{1}{\mathrm{Pr}}\left[1+\frac{4}{3}Rd\right]\theta'' + f\theta' + J\sin^2\varphi\,\xi^2 f'^2 + \\ + \mathrm{Ec}\xi^2 f''^2 + Q\theta = 0, \quad (15)$$

$$\eta = 0: \quad f'(\xi,0) = f(\xi,0) = 0, \quad \theta(\xi,0) = 0, \quad (16a)$$

$$\eta \to \infty: \quad f'(\xi,\infty) = 0, \quad \theta(\xi,\infty) = 0. \quad (16b)$$

3.2. Second level of truncation

At the second level truncation, we introduce new functions:

$$g = \partial f/\partial\xi,\ h = \partial g/\partial\xi,\ \phi = \partial\theta/\partial\xi,\ \varphi = \partial\phi/\partial\xi.$$

We differentiate Eqs. (14)–(16) with respect to ξ and obtain the following system of ODEs:

$$(1+n)f''' + n\mathrm{We}f''f''' - (1+\xi\Lambda)f'^2 + \\ -\left(M\sin^2\varphi + \frac{1}{\mathrm{Da}}\right)f' + \frac{sin\xi}{\xi}\theta = \xi(f'g' - f''g), \quad (17)$$

$$\frac{1}{\mathrm{Pr}}\left[1+\frac{4}{3}Rd\right]\theta'' + f\theta' + J\sin^2\varphi\,\xi^2 f'^2 + \\ + \mathrm{Ec}\xi^2 f''^2 + Q\theta = \xi(f'\phi - \theta' g), \quad (18)$$

$$(1+n)g''' + n\mathrm{We}(fg'' + f''g) - f'g' + f''g + \\ -g'^2 - \xi\Lambda g'^2 - \Lambda f'^2 - \left(M\sin^2\varphi + \frac{1}{\mathrm{Da}}\right)g' + \\ + \frac{\sin\xi}{\xi}\phi + \theta\left(\frac{\xi\cos\xi - \sin\xi}{\xi^2}\right) = \xi(g'g' - g''g), \quad (19)$$

$$\frac{1}{\mathrm{Pr}}\left[1+\frac{4}{3}Rd\right]\phi'' + f\phi' + 2\theta' g + \mathrm{Ec}\left(\xi^2 g'^2 + 2f'^2\xi\right) + \\ + J\sin^2\varphi\left(\xi^2 g'^2 + 2f'^2\xi\right) - f'\phi + Q\phi = \xi(\phi g - g\phi'), \quad (20)$$

$$\eta = 0: \quad g'(\xi,0) = g(\xi,0) = 0, \quad \phi(\xi,0) = 1, \quad (21a)$$

$$\eta \to \infty: \quad g'(\xi,\infty) = 0, \quad \phi(\xi,\infty) = 0. \quad (21b)$$

4. Physics-informed neural network

Three parallel layers occur in an architecture of PINNs, i.e. the input layer, hidden and output layer. The intention of a fully connected network (FCN) is to collect data, process them and produce an output. We initialise the basic stages of PINN implementation by dividing a one-dimensional domain in η discrete points, or nodes, in a bid to find non-similar partial differential equations (PDEs). These nodes illustrate independent variable's oscillation between 0 and η_∞. A flow chart representation of PINN is illustrated in Fig. 3, where red nodes represent the collocated nodes and blue nodes represent the boundary nodes. The only boundary conditions needed to assemble an approximate solution are the Dirichlet and Neumann boundary conditions. The η-axis is partitioned into m equal segments and contains m+1 nodes, where η_0 is set as the first node, and η_m is a last/final node. ODEs need to be satisfied at the intermediate nodes between η_0 and η_m, which are the considered collocated nodes scattered along the η-axis.

The initial or input layer holds data (η_i) as shown in Fig. 3. We define an output function as $\hat{y}_i = [f_i(\eta), \theta_i(\eta)]$.

In this analysis, weight and bias are two learning parameters that are incorporated in the function $\hat{y}_i = F(x_i, w, b)$. In order to achieve a sufficient approximation function, such learning parameters need to be advanced in a way that $\hat{y}_i$ appropriately anticipates the desired value y_i (sample output data). The loss function determines the deviation of (y_i) and $(\hat{y}_i)$. One such method to calculate the loss function is by means of the mean

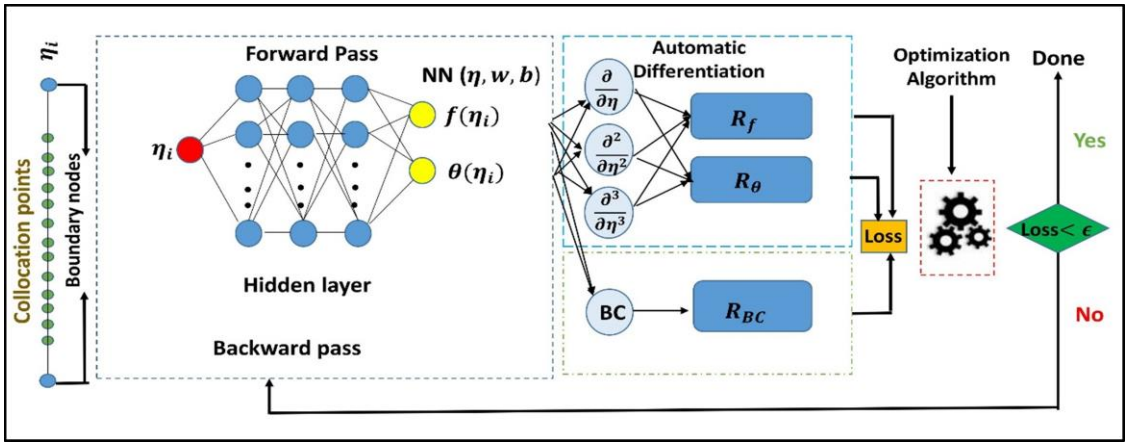


Fig. 3. Schematic flow chart of PINNs.

squared error format given by:

$$\text{Loss} = \frac{\left[\sum_{i=0}^{i=m}(y_i - \widehat{y_i})^2\right]}{m}. \quad (22)$$

The input data η_i are entered into the network as illustrated in Fig. 3. One neuron exists in the input or starting layer, which holds the given data. Following that, there are three hidden layers with an arbitrary number of neurons, and the output layer, depicted in Fig. 3, which consists of two neurons. In the architecture discussed above, each neuron is fully connected through linkages enabling information to pass between the neurons. Every link has a weight $w_{i,j}^k$, which decreases or increases the quantity of data that flows from the source node to destination nodes. Weight is utilised to calculate the contribution of every node to output outcomes, and a weighted sum of inputs is considered. A bias term is added to calculate the involvement of all nodes on predicted nodes:

$$z_i = b_i + \left(\sum_{j=1}^{j=n} a_j w_{i,j}^k\right). \quad (23)$$

The bias term (b_i) is assigned to input and hidden layers, with a value of unity. The regression analysis is employed using function $f(x) = \alpha x + \beta$, where α and β signify the weight and neural network simulation biases, respectively. An activation function is applied, which computes the output within a determined range. The tangent hyperbolic function is utilised in the present study, and the obtained z_i is forwarded through the given activation function:

$$\text{Output}(i) = \tanh(z_i). \quad (24)$$

For computation of the output z_i, data flows forward from one neuron to another. Before computing the estimated value $\widehat{y_i}$, the procedure is done layer by layer. As the procedure starts with the weight and biases being predicted, it is assumed that there will be a considerable difference between $\widehat{y_i}$ and y_i. It is required to adjust weight and bias parameters in a way that reduces the calculated loss function. Finding a variable weight and bias value that gives a minimised loss function is referred to as optimisation. We used an adaptive moment estimation (ADAM) to optimise the loss function by tuning the weights and biases. The iterative procedure proceeds until the loss function meets specified convergence criteria. Loss functions of the ordinary differential equation (ODE) type were addressed by Raissi et al. [40–42]. The total loss function is divided into three types: data loss, physics loss and boundary condition loss. For the physics loss, we define residual functions as [43–45]:

$$L_{data} = \frac{1}{N}\sum_{i=1}^{N}(\widehat{y_i} - y_i)^2. \quad (25)$$

The momentum boundary layer residuals are defined as:

$$R_f = (1+n)f''' - (1+\xi\Lambda)f'^{\mathbf{2}} + na\text{We}f''f''' + -\left(\frac{1}{\text{Da}}\right)f' + \frac{\sin\xi}{\xi}\theta. \quad (26)$$

Residuals in the thermal boundary layer are as follows:

$$R_\theta = \frac{1}{\text{Pr}}\left[1 + \frac{4}{3}Rd\right]\theta'' + f\theta' + \text{Ec}M\xi^2 f'^{\mathbf{2}} + \text{Ec}\xi^2 f''^2. \quad (27)$$

The physics loss function is given by:

$$L_{physics} = \frac{1}{N}\sum_{i=1}^{N}(R_f^2 + R_\theta^2). \quad (28)$$

The loss due to the boundary condition is:

$$L_{BC} = \frac{1}{N}\sum_{i=1}^{N}[f'(0)^2 + f(0)^2 + (\theta(0)-1)^2 + + f'(\infty)^2 + \theta(\infty)^2]. \quad (29)$$

The total loss can be calculated as:

$$Total\ loss = L_{data} + L_{physics} + L_{BC}. \quad (30)$$

5. Outcome evaluation and discussion

The solution of the problem under consideration has non-similar PDEs – Eqs. (6)–(8), which are solved through the well-known non-similarity method. The derived non-similar PDEs are reduced to ODEs as specified in Eqs. (14)–(16) through application of the first and second level of truncation. The derived system of ODEs is solved through the physics informed neural network method. We assumed 15 hidden layers and 84 neurons in a neural network architecture. The PINN prediction is evaluated at a fixed convergence tolerance 10^{-3}. An L^2-error is computed throughout the training process to assess the accuracy and validity of the PINN scheme. The comparison between PINN predictions and the numerical solution (obtained using bvp4c) is presented in Figs. 4–5. As seen in Fig. 4, there are notable differences between the numerical and PINN-projected solutions for temperature and velocity at lower iteration levels. The PINN solution for $f(\xi,\eta)$, $f'(\xi,\eta)$, $f"(\xi,\eta)$ and $\theta(\xi,\eta)$, $\theta'(\xi,\eta)$ differs significantly from the numerical solution at the start of training due to the random initialisation of the trainable parameters. This random initialisation of the trainable parameters does not satisfy the governing equations and boundary conditions; as a result, the PINN prediction deviates from the numerical solution at lower iteration levels.

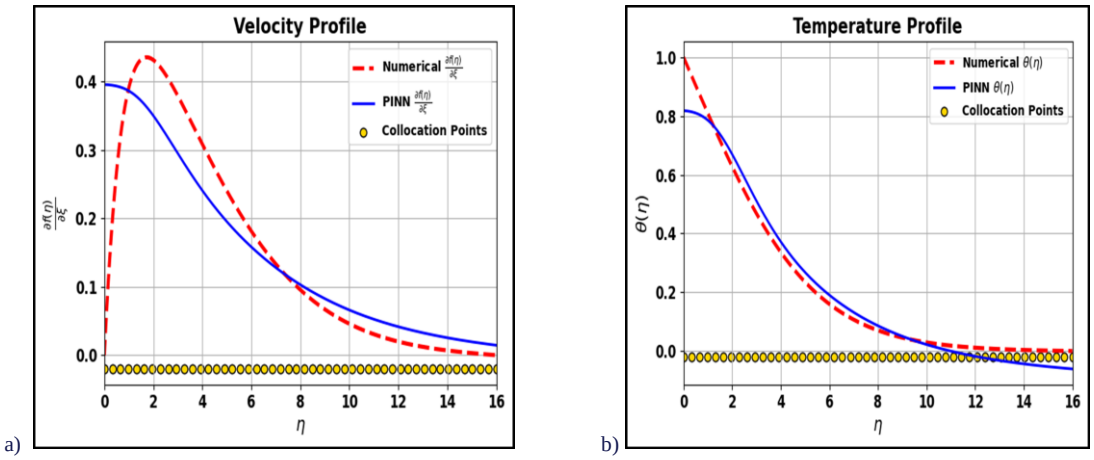
Nevertheless, an L^2-error decreases when the increased number of iterations, as depicted in Table 1.To illustrate the accuracy and stability of the solution, we increased the number of iterations to four thousand and eight hundred, as well as the number of hidden layers and neurons, as shown in Fig. 5. The comparison suggests that PINN outcome and numerical solution (achieved using bvp4c) are matched perfectly, thus demonstrating the credibility and consistency of the developed PINN approach to solve non-similar boundary layer flow of non-Newtonian fluid [46–51].

The physics loss and total loss function of the computed solution are shown in Fig. 6(a–b). The difference between the numerical results and PINN-predicted results is characterised by the L^2-error estimate. Table 1 illustrates L^2-error of the developed problem under consideration for various iteration numbers and learning rates. It is observed that when the number of iterations is increased, then the computation cost increases, but the accuracy and the precision of predicted solution increases. The accuracy of the model can also be enhanced by increasing the number of neurons while keeping a fixed number of hidden layers [52–58]. Increasing the number of neurons, L^2-error estimate

reduces, which indicates the solution reliability, as shown in Table 2. It is clear that when the loss function reduces, the network can infer the expected patterns with low accuracy. In Table 2, we consider the mesh size of 0.05. Similarly, Table 3 presents an L^2-error estimate for the mesh size of 0.02. The solution is observed to be convergent and reliable.

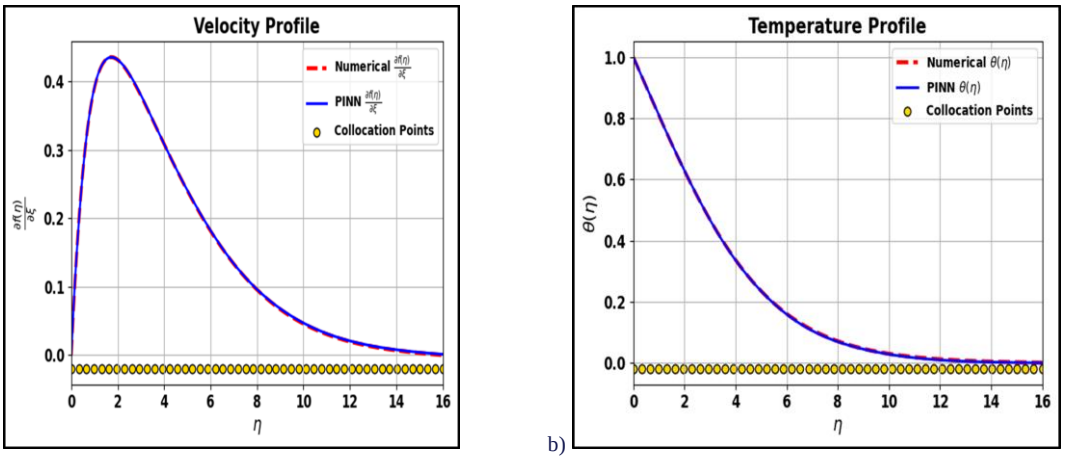
Table 4 shows L^2-error estimates for the flow response outputs, i.e. $\frac{\partial f}{\partial \eta}(\xi,\eta)$ and $\theta(\xi,\eta)$, for various values of magnetic parameter M. As evident in Table 4, the minimum errors are obtained for $M = 5$, reflecting the best accuracy of the PINN model.

Figure 7 displays the influence of magnetic field intensity M, on flow response outputs, i.e. $\theta(\xi,\eta)$ and $\frac{\partial f}{\partial \eta}(\xi,\eta)$. Figure 7a shows the effect of M on the velocity gradient, and it is noted that the velocity decreases with the increasing value of M.



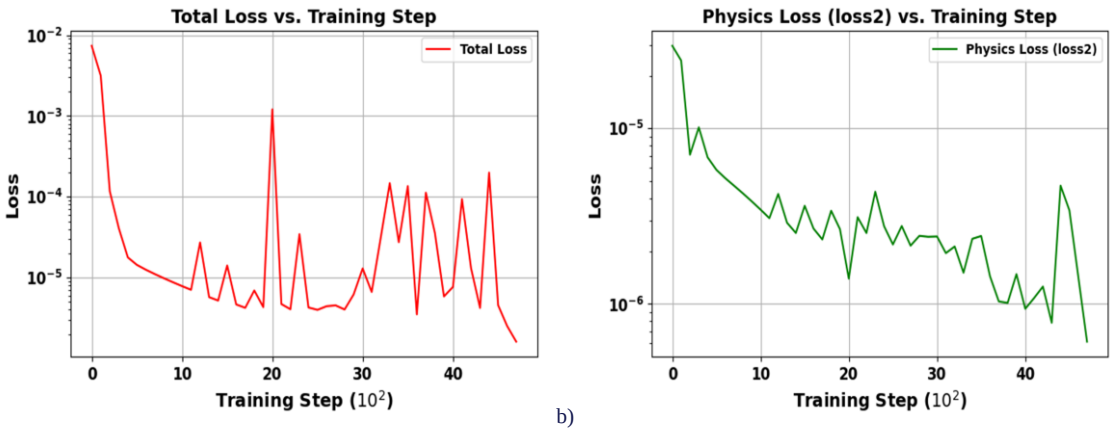
a) b)

Fig. 4. The comparison of numerical and PINN predicted solutions for velocity and temperature (low iteration number).



a) b)

Fig. 5. The comparison of numerical and PINN predicted solutions for velocity and temperature (large iteration number).



a) b)

Fig. 6. Loss function for flow response outputs.

Table 1. L^2-error between the predicted and numerical solution for different numbers of iterations and training steps.

Iteration	Training steps	L^2-error
100	10 000	0.061065
400	40 000	0.013647
800	80 000	0.007030
1200	120 000	0.004858
1600	160 000	0.006481
2000	200 000	0.004257
2400	240 000	0.009814
2800	280 000	0.003473
3200	320 000	0.003778
3600	360 000	0.008236
4000	400 000	0.004909
4400	440 000	0.002409
4800	480 000	0.001560

Table 2. L^2-error flow outputs response, i.e. (ξ,η), $\frac{\partial f}{\partial \xi}(\xi,\eta)$, $\frac{\partial^2 f}{\partial \xi^2}(\xi,\eta)$ and $\theta(\xi,\eta)$, $\frac{\partial \theta}{\partial \xi}(\xi,\eta)$ for various neuron numbers at $\eta_\infty = 5$, $\Delta = 0.05$.

Index	Hidden layers	Neurons	L^2-error of $f(\xi,\eta)$	L^2-error of $\frac{\partial f}{\partial \xi}(\xi,\eta)$	L^2-error of $\frac{\partial^2 f}{\partial \xi^2}(\xi,\eta)$	L^2-error of $\theta(\xi,\eta)$	L^2-error of $\frac{\partial \theta}{\partial \xi}(\xi,\eta)$
0	4	8	0.000866	0.001808	0.001555	0.000978	0.001327
1	4	16	0.001112	0.001209	0.000951	0.000952	0.000322
2	4	32	0.001169	0.001864	0.001947	0.001087	0.000323
3	4	64	0.000715	0.000993	0.000864	0.000563	0.000334

Table 3. L^2-error for flow outputs response, i.e. $f(\xi,\eta)$, $\frac{\partial f}{\partial \xi}(\xi,\eta)$, $\frac{\partial^2 f}{\partial \xi^2}(\xi,\eta)$ and $\theta(\xi,\eta)$, $\frac{\partial \theta}{\partial \xi}(\xi,\eta)$ for various neuron numbers at $\eta_\infty = 10$, $\Delta = 0.02$.

Index	Hidden layers	Neurons	L^2-error of $f(\xi,\eta)$	L^2-error of $\frac{\partial f}{\partial \xi}(\xi,\eta)$	L^2-error of $\frac{\partial^2 f}{\partial \xi^2}(\xi,\eta)$	L^2-error of $\theta(\xi,\eta)$	L^2-error of $\frac{\partial \theta}{\partial \xi}(\xi,\eta)$
0	4	8	0.002477	0.004453	0.003391	0.001578	0.001981
1	4	16	0.004044	0.004369	0.002940	0.001647	0.000128
2	4	32	0.011884	0.006617	0.004095	0.004044	0.001846
3	4	64	0.003274	0.002536	0.002067	0.001630	0.001027

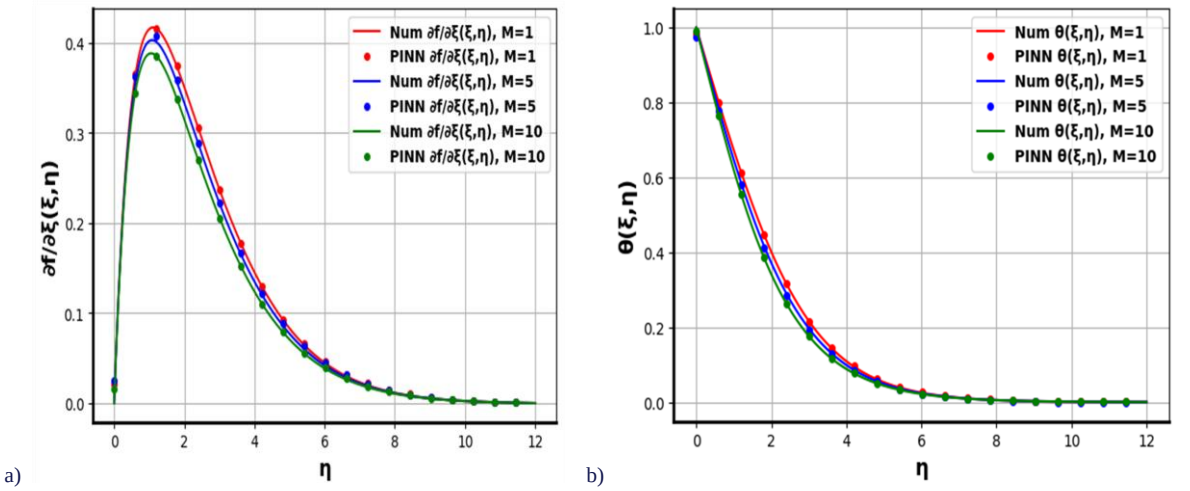
Table 4. L^2-error for flow outputs response, i.e. $\frac{\partial f}{\partial \xi}(\xi,\eta)$ and $\theta(\xi,\eta)$ for different values of M.

Index	M	L^2-error of $\frac{\partial f}{\partial \xi}(\xi,\eta)$	L^2-error of $\theta(\xi,\eta)$
0	1	0.031009	0.035468
1	5	0.006992	0.003424
2	10	0.040139	0.048676

Figure 7b indicates that the temperature profile drops more sharply as M is increased, which means that higher magnetic fields assist thermal diffusion and inhibit fluid motion. Such behaviour is representative of the traditional magnetohydrodynamic (MHD) phenomenon under which the Lorentz force acts to oppose fluid flow and alter heat transfer.

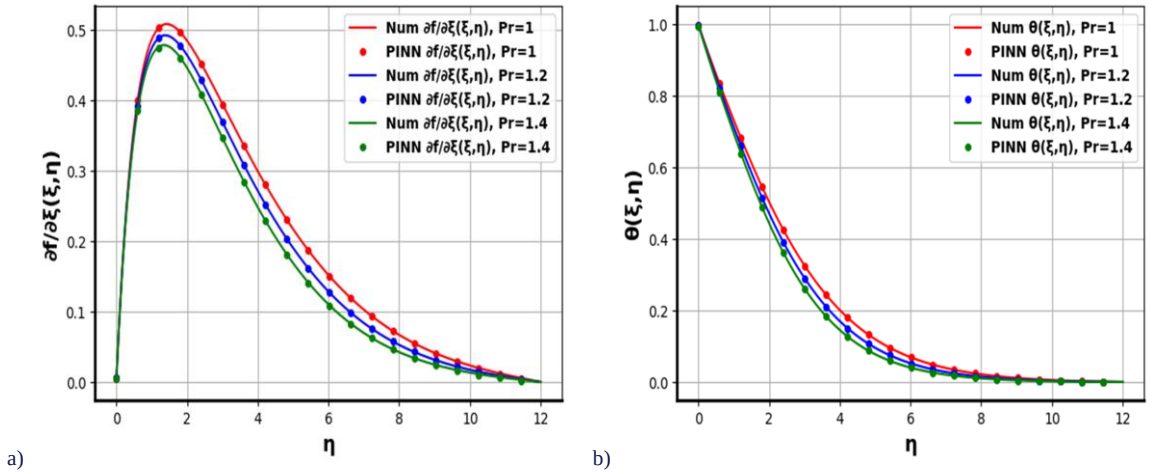
Table 5 displays the L^2-error estimates of the flow $f'(\xi,\eta)$ and $\theta(\xi,\eta)$ for various values of the Prandtl number Pr. It shows that prediction errors increase with the increasing value of Pr. Figure 8 plots the effect of Pr on both the velocity and temperature profiles. It is seen from Fig. 8a that the velocity profile diminishes with the increasing Pr, as would be expected because a greater Prandtl number represents a fluid of lower thermal diffusivity and thus a thinner velocity boundary layer is formed. Figure 8b shows that the temperature profile decays more precipitously with the greater Pr, representing a more intense thermal gradient and less heat diffusion. These trends match exactly the theoretical behaviour of boundary layer flows.

Table 6 shows L^2-errors of the velocity gradient $\frac{\partial f}{\partial \xi}(\xi,\eta)$ and $\theta(\xi,\eta)$ for various values of the radiation parameter Rd. The minimum L^2-errors for both fields occur at $Rd = 2$, which means that the PINN model is the best at higher radiation levels at $Rd = 2$. On the other hand, the maximum errors occur at $Rd = 1$, which shows a lower accuracy under a weaker radiative impact. Figure 9a shows that the velocity gradient $\frac{\partial f}{\partial \xi}(\xi,\eta)$ increases with the increase of Rd, which characterise greater fluid motion due to higher radiation of heating. Figure 9b illustrates that the temperature of fluid decreases with the increasing Rd values. All these findings provide evidence of strong effects of radiation on the flow velocity and temperature.



a) b)

Fig. 7. Flow field and temperature profile for various values of *M*.



a) b)

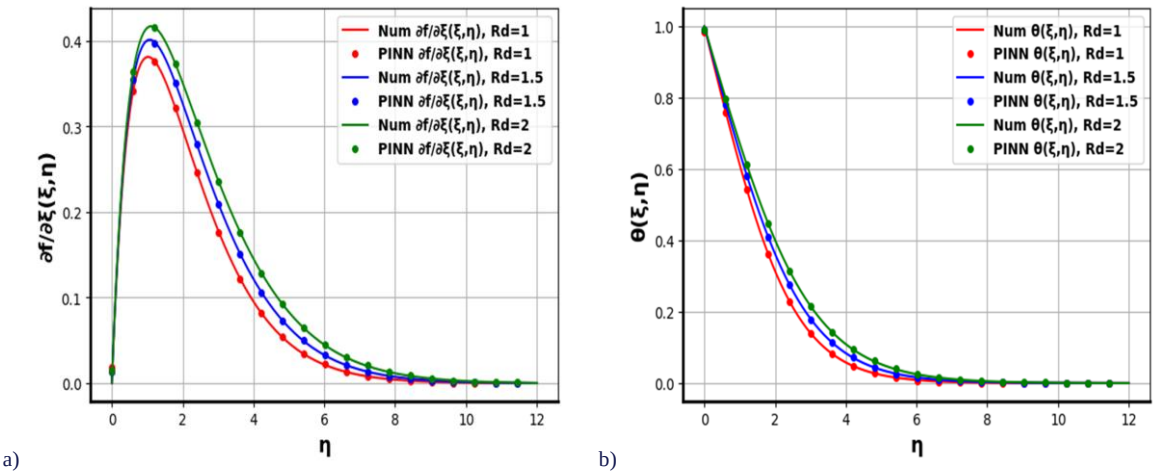
Fig. 8. Flow field and temperature profile for various values of Pr.

Table 5. L^2-error for different values of Pr.

Index	Pr	L^2-error of $\frac{\partial f}{\partial \xi}(\xi,\eta)$	L^2-error of $\theta(\xi,\eta)$
0	1.0	0.001743	0.000628
1	1.2	0.004053	0.002097
2	1.4	0.018577	0.013197

Table 6. L^2-error for different values of *Rd*.

Index	*Rd*	L^2-error of $\frac{\partial f}{\partial \xi}(\xi,\eta)$	L^2-error of $\theta(\xi,\eta)$
0	1	0.009623	0.004741
1	1.5	0.007378	0.002921
2	2.0	0.006445	0.003846



a) b)

Fig. 9. Flow field and temperature profile for various values of *Rd*.

Table 7 indicates the L^2-errors for the velocity gradient $f'(\xi,\eta)$ and temperature profile $\theta(\xi,\eta)$ for various values of parameter n. The lowest error appears at $n = 0.1$, which suggests a high accuracy of the PINN model for small values of n. Correspondingly, when the value of n increases up to 0.7, errors also increase, and it shows that the model becomes less accurate with the increased nonlinearity or complexity of flow. Figure 10a indicates that the velocity gradient increases with the increasing n; it shows the formation of a stronger momentum boundary layer. In Fig.10b, the temperature profile decreases more slowly with the increasing n, as a result of a thicker thermal boundary layer. This is in accordance with the increased convection and heat transfer due to increasing n.

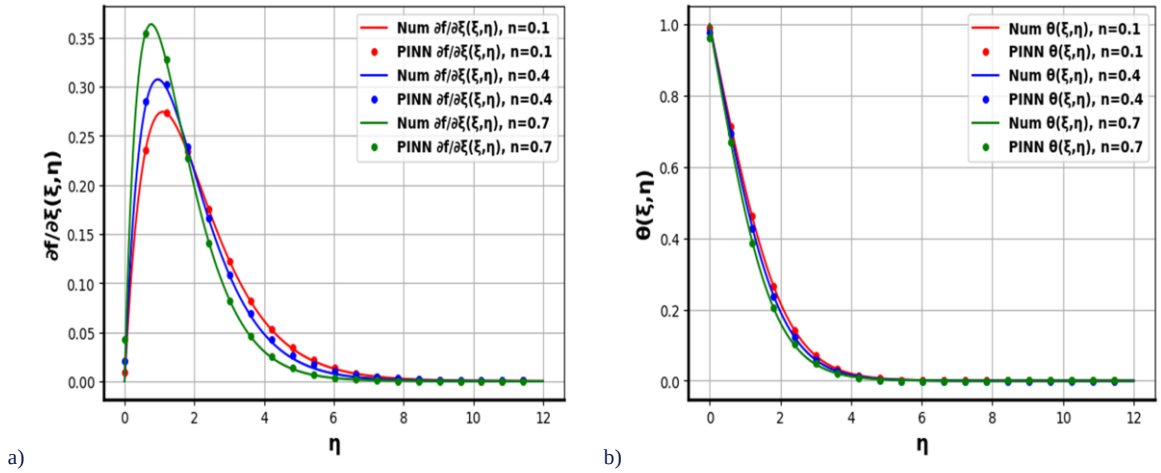
Table 8 presents L^2-errors for the velocity gradient $f'(\xi,\eta)$ and $\theta(\xi,\eta)$ for various values of Weissenberg number We in a tangent hyperbolic fluid. The largest errors are obtained at

Table 7. L^2-error for different values of n.

Index	n	L^2-error of $\frac{\partial f}{\partial \xi}(\xi,\eta)$	L^2-error of $\theta(\xi,\eta)$
0	0.1	0.007542	0.003344
1	0.4	0.008851	0.005889
2	0.7	0.026852	0.025722

Table 8. L^2-error for different values of We.

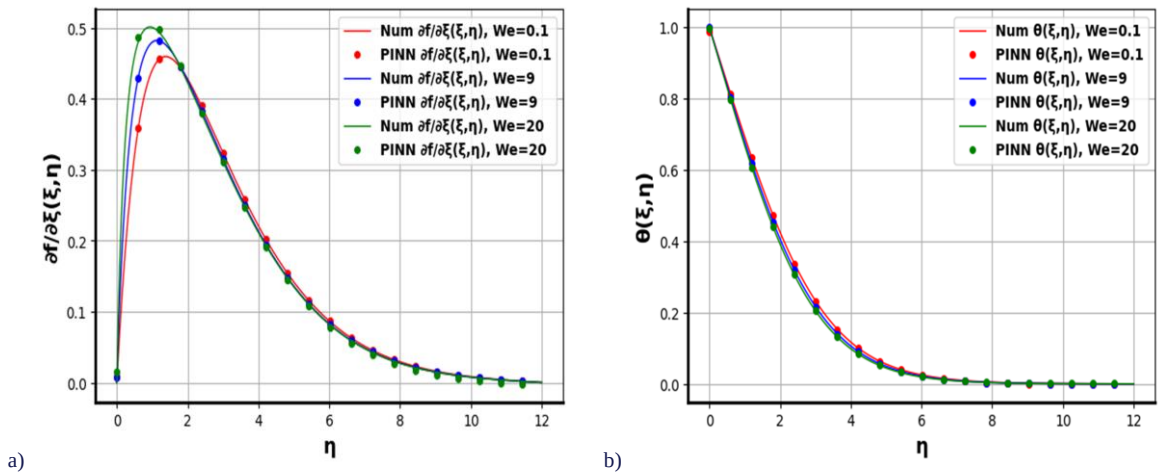
Index	We	L^2-error of $\frac{\partial f}{\partial \xi}(\xi,\eta)$	L^2-error of $\theta(\xi,\eta)$
0	0.1	0.030139	0.050238
1	9.0	0.003156	0.001225
2	20.0	0.005392	0.003613



a) b)

Fig. 10. Flow field and temperature profile for various values of n.

We = 0.1, which reflects the fact that the PINN model performs poorly under low elastic effects. With the increasing value of We, the errors decrease precipitously, reflecting a better performance of the model with moderate elasticity, while a minor error is observed at We = 20, perhaps due to higher flow complexity. Figure 11a indicates that the velocity gradient becomes stronger, developing earlier with We, which captures the fluid's increased elasticity and resistance to deformation. In Fig. 11b, the temperature profile is shown to decrease faster for large values of We. These trends verify that the tangent hyperbolic fluid is more responsive and thermally efficient with the growing We, and the PINN model well represents these variations.



a) b)

Fig. 11. Flow field and temperature profile for various values of We.

6. Conclusions

A benchmark simulation of non-similar boundary layer flow of non-Newtonian fluid with thermal radiation effects over a cylinder embedded in a porous medium is made in this study, using physics-informed neural networks. Training the networks with loss functions in charge of meeting the governing equations and boundary conditions served as the foundation for an approximate solution. It is discovered that the networks could accurately predict the solutions when their learning rate was less than 10^{-4}. L^2-error is investigated to show the accuracy and validity of the developed PINN methodology to solve non-similar boundary layer flow over a cylinder. The numerical and tabulated results are presented to ensure the accuracy of the proposed scheme. The primary considerations are as follows:

- The temperature distribution and convective flow field are significantly influenced by the buoyant force. The heat transfer and momentum boundary layer grow for increasing values of λ against a fixed value of $\eta = 12$.
- Momentum diffuses more quickly than heat at high Pr numbers, resulting in a narrower thermal boundary layer. By developing steeper temperature gradients close to heated surfaces, this accelerates the rate of heat transfer. Thus, with the improving values of Pr, the outputs $f'(\eta, \xi)$ and $\theta(\eta, \xi)$ raise.
- A high thermal radiation parameter indicates that surface convection outpaces internal conduction, leading to sharp temperature gradients within the solid and enhanced heat exchange at the surface.
- The accuracy was assessed using the L^2-error between the PINN and bvp4c solutions, which consistently decreased as the number of neurons and iterations increased. At 12 000 iterations, the L^2-error reached a minimum of $4.39 \cdot 10^{-3}$.
- At 1000 and 2000 training epochs, the PINN solution and the numerical solution were initially very different, but at 12 000 epochs, they demonstrated great agreement, signifying the importance of network size and training depth.

References

[1] Heaton, J. (2018). Review of book: Goodfellow I., Bengio Y. and Courville A.: Deep learning: MIT Press, 2016, ISBN: 0262035618. *Genetic Programming and Evolvable Machines*, 19(1), 305−307. doi: 10.1007/s10710-017-9314-z

[2] Lv, S.X., Peng, L., & Wang, L. (2018). Stacked autoencoder with echo-state regression for tourism demand forecasting using search query data. *Applied Soft Computing*, 73, 119−133. doi: 10.1016/j.asoc.2018.08.024

[3] Peng, L., Liu, S., Liu, R., & Wang, L. (2018). Effective long short-term memory with differential evolution algorithm for electricity price prediction. *Energy*, 162, 1301−1314. doi: 10.1016/j.energy.2018.05.052

[4] Agrawal, A., & Choudhary, A. (2019). Deep materials informatics: Applications of deep learning in materials science. *MRS Communications*, 9(3), 779−792. doi: 10.1557/mrc.2019.73

[5] Goh, G.B., Hodas, N.O., & Vishnu, A. (2017). Deep learning for computational chemistry. *Journal of Computational Chemistry*, 38(16), 1291−1307. doi: 10.1002/jcc.24764

[6] Chen, H., Engkvist, O., Wang, Y., Olivecrona, M., & Blaschke, T. (2018). The rise of deep learning in drug discovery. *Drug Discovery Today*, 23(6), 1241−1250. doi: 10.1016/j.drudis.2018.01.039

[7] Kelly, M.F., Parker, P.A., & Scott, R.N. (1990). The application of neural networks to myoelectric signal analysis: A preliminary study. *IEEE Transactions on Biomedical Engineering*, 37(3), 221−230. doi: 10.1109/10.52324

[8] Olaiya, F., & Adeyemo, A.B. (2012). Application of data mining techniques in weather prediction and climate change studies. *International Journal of Information Engineering and Electronic Business*, 4(1), 5–59. doi: 10.5815/ijieeb.2012.01.07

[9] Pierson, H.A., & Gashler, M.S. (2017). Deep learning in robotics: a review of recent research. *Advanced Robotics*, 31(16), 821−835. doi: 10.1080/01691864.2017.1365009

[10] Izzo, D., Märtens, M., & Pan, B. (2019). A survey on artificial intelligence trends in spacecraft guidance dynamics and control. *Astrodynamics*, 3(4), 287−299. doi: 10.1007/s42064-018-0053-6

[11] Cohen, N., Sharir, O., & Shashua, A. (2016). On the expressive power of deep learning: A tensor analysis. *Proceedings of Machine Learning Research*, 49, 698−728; *29th Annual Conference on Learning Theory*, June 23−26, New York, USA.

[12] Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5), 359−366.

[13] Baydin, A.G., Pearlmutter, B.A., Radul, A.A., & Siskind, J.M. (2018). Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(153), 1−43.

[14] Cuomo, S., Di Cola, V.S., Giampaolo, F., Rozza, G., Raissi, M., & Piccialli, F. (2022). Scientific machine learning through physics–informed neural networks: Where we are and what's next. *Journal of Scientific Computing*, 92(3), 88. doi: 10.1007/s10915-022-01939-z

[15] Barania, H., & Esmaeilpour, M. (2022). On the application of physics informed neural networks (PINN) to solve boundary layer thermal-fluid problems. *International Communications in Heat and Mass Transfer*, 132, 105890. doi: 10.1016/j.icheatmasstransfer.2022.105890

[16] Shoaib, M., Kainat, S., Alabduljabbar, A.A., Nisar, K.S., & Zahoor Raja, M.A. (2025). Integrated intelligent neuro-evolutionary computing approach to study SEIRC model representing campylobacteriosis transmission dynamics. *Applied Mathematics in Science and Engineering*, 33(1), 2441257. doi: 10.1080/27690911.2024.2441257

[17] Mao, Z., Jagtap, A.D., & Karniadakis, G.E. (2020). Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering*, 360, 112789. doi: 10.1016/j.cma.2019.112789

[18] Alahmadi, H., Iqbal, J., & Abbasi, F.M. (2025). Artificial neural network aided computing for two dimensional magnetohydrodynamic peristaltic movement of nanofluid with heat and mass transfer. *Engineering Applications of Artificial Intelligence*, 154, 110990. doi: 10.1016/j.engappai.2025.110990

[19] Hanrahan, S., Kozul, M., & Sandberg, R.D. (2023). Studying turbulent flows with physics-informed neural networks and sparse data. *International Journal of Heat and Fluid Flow*, 104, 109232. doi: 10.1016/j.ijheatfluidflow.2023.109232

[20] Ali, I., Raja, M.A.Z., Chang, C.Y., Khan, M.P., Shoaib, M., & Shu, C.M. (2025). Novel Machine Predictive Exogenous Knowledge Driven Neuro-Structures for Unsteady Squeezing Nanofluidic Model with Rotating-Oscillating Disks. *International Journal of Thermophysics*, 46(12), 182. doi: 10.1007/s10765-025-03648-9

[21] Baty, H., & Baty, L. (2023). Solving differential equations using physics informed deep learning: a hand-on tutorial with benchmark tests. *arXiv*, 2302.12260. doi: 10.48550/arXiv.2302.12260

[22] Prandtl, L. (1905). Uber Flussigkeitsbewegung bei sehr kleiner Reibung. In: *Verhandlungen des III Internationalen Mathematiker Kongresses* (pp. 485–491), 8–13 August 1904, Heidelberg, B.G. Teubner, Leipzig (in German).

[23] Peng, Y., Lv, B.H., Yuan, J.L., Ji, H.B., Sun, L., & Dong, C.C. (2014). Application and prospect of the non-Newtonian fluid in industrial field. *Materials Science Forum*, 770, 396−401. doi: 10.4028/www.scientific.net/MSF.770.396

[24] Crespí-Llorens, D., Vicente, P., & Viedma, A. (2015). Generalized Reynolds number and viscosity definitions for non-Newtonian fluid flow in ducts of non-uniform cross-section. *Experimental Thermal and Fluid Science*, 64, 125−133. doi: 10.1016/j.expthermflusci.2015.02.005

[25] Hauswirth, S.C., Bowers, C.A., Fowler, C.P., Schultz, P.B., Hauswirth, A.D., Weigand, T., & Miller, C.T. (2020). Modeling cross model non-Newtonian fluid flow in porous media. *Journal of Contaminant Hydrology*, 235, 103708. doi: 10.1016/j.jconhyd.2020.103708

[26] Sohail, M., Nazir, U., Singh, A., Tulu, A., & Khan, M.J. (2024). Finite element analysis of cross fluid model over a vertical disk suspended to a tetra hybrid nanoparticles mixture. *Scientific Reports*, 14(1), 1520. doi: 10.1038/s41598-024-51262-w

[27] Awais, M., & Salahuddin, T. (2024). Radiative magnetodydrodynamic cross fluid thermophysical model passing on parabola surface with activation energy. *Ain Shams Engineering Journal*, 15(1), 102282. doi: 10.1016/j.asej.2023.102282

[28] Khan, M.I., Rauf, A., & Shah, A. (2021). Numerical investigation of viscous effects on the nonlinear Burgers equation. *Turkish Journal of Mathematics*, 45(1), 529–539. doi: 10.3906/mat-2001-7

[29] Khan, M.I., Zeeshan, A., Arain, M.B., Alqahtani, A.S., & Malik, M.Y. (2025). Temporal stability and non-unique solution of reacting Eyring Powell flows over shrinking wedges using neural networks. *Engineering Applications of Artificial Intelligence*, 141, 109828. doi: 10.1016/j.engappai.2024.109828

[30] Zeeshan, A., Khan, M. Z., Khan, I., & Pervaiz, Z. (2023). Sensitivity analysis and numerical investigation of hybrid nanofluid in contracting and expanding channel with MHD and thermal radiation effects. In *Nanomaterials and Nanoliquids: Applications in Energy and Environment* (pp. 281–310). Springer Nature Singapore.

[31] Subba Rao, A., Ramachandra Prasad, V., Bhaskar Reddy, N., & Anwar Bég, O. (2015). Heat transfer in a Casson rheological fluid from a semi-infinite vertical plate with partial slip. *Heat Transfer – Asian Research*, 44(3), 272−291. doi: 10.1002/htj.21115

[32] Ramachandra Prasad, V., Subba Rao, A., & Anwar Bég, O. (2013). Flow and heat transfer of Casson fluid from a horizontal circular cylinder with partial slip in non-Darcy porous medium. *Journal of Applied & Computational Mathematics*, 2(2), 1000127. doi: 10.4172/2168-9679.1000127

[33] Fatima, N., Ghodhbani, R., Khalid, N., Khan, M.I., Taha, T., Ijaz, N., & Saleem, N. (2025). Machine learning-based prediction of heat transfer enhancement in carreau fluids with impact of homogeneous and heterogeneous reactions. *Multiscale and Multidisciplinary Modeling, Experiments and Design*, 8(2), 145. doi: 10.1007/s41939-025-00734-1

[34] Shaheen, S., Mahmood, T., Raja, M.A.Z., Al-Yarimi, F.A., Arain, M.B., & Hu, J. (2025). Machine learning-based investigation of activation energy for bio-convective boundary layer flow inspired by challenges in aerospace heat transfer systems. *Engineering Applications of Artificial Intelligence*, 160, 111917. doi: 10.1016/j.engappai.2025.111917

[35] Ibrahim, W., & Gizewu, T. (2020). Nonlinear mixed convection flow of a tangent hyperbolic fluid with activation energy. *Heat Transfer*, 49(5), 2427−2448. doi: 10.1002/htj.21729

[36] Jan, A., Mushtaq, M., & Hussain, M. (2024). Heat transfer enhancement of forced convection magnetized cross model ternary hybrid nanofluid flow over a stretching cylinder: non-similar analysis. *International Journal of Heat and Fluid Flow*, 106, 109302. doi: 10.1016/j.ijheatfluidflow.2024.109302

[37] Shaheen, S., Huang, H., Arain, M.B., Al-Zubaidi, A., & Tageldin, E.M. (2023). Concentration and thermal analysis of immiscible tangent hyperbolic fluid with distinct viscosity through horizontal asymmetric channel: Theoretical and observational study. *Case Studies in Thermal Engineering*, 50, 103386. doi: 10.1016/j.csite.2023.103386

[38] Minkowycz, W.J., & Sparrow, E. (1978). Numerical solution scheme for local nonsimilarity boundary-layer analysis. *Numerical Heat Transfer, Part B: Fundamentals*, 1(1), 69−85. doi: 10.1080/10407797809412161

[39] Sparrow, E.M., & Yu, H.S. (1971). Local non-similarity thermal boundary-layer solutions. *Journal of Heat Transfer*, 93(4), 328−334. doi: 10.1115/1.3449827

[40] Raissi, M., Perdikaris, P., & Karniadakis, G.E. (2017). Physics informed deep learning (part I): Data-driven solutions of nonlinear partial differential equations. *arXiv*, 1711.10561. doi: 10.48550/arXiv.1711.10561

[41] Raissi, M., Perdikaris, P., & Karniadakis, G.E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378, 686−707. doi: 10.1016/j.jcp.2018.10.045

[42] Raissi, M., Yazdani, A., & Karniadakis, G.E. (2020). Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367(6481), 1026−1030.

[43] Iqbal, J., Akbar, Y., & Alam, M.M. (2026). Thermal enhancement and energy minimization using Powell–Eyring nanofluid flow in PTSC for solar-powered aircraft: a combined RSM, ANN, and sensitivity analysis. *Journal of Thermal Analysis and Calorimetry*, 151(2), 1717−1741. doi: 10.1007/s10973-025-15104-9

[44] Rehman, A.U., & Khan, M.I. (2025). Advanced computational intelligence approach for heat transfer and entropy generation in non-Newtonian third grade magnetohydrodynamics peristaltic transport through asymmetric channels. *International Communications in Heat and Mass Transfer*, 169, 109947. doi: 10.1016/j.icheatmasstransfer.2025.109947

[45] Min, Z., Shaheen, S., Arain, M.B., & Al-Yarimi, F.A. (2026). Theoretical analysis of heat and mass transfer in a cilia-induced bi-layer Giesekus fluid flow under viscosity gradient. *International Communications in Heat and Mass Transfer*, 172, 110448. doi: 10.1016/j.icheatmasstransfer.2025.110448

[46] Khan, M.I., Asgher, Z., Zeeshan, A., & Xu, H. (2025). Integrating artificial intelligence for stability assessment in Casson hybrid nanofluid flow using LMS-BPNN. *Neural Computing and Applications*, 37(21), 16231−16252. doi: 10.1007/s00521-025-11320-4

[47] Min, Z., Shaheen, S., Arain, M.B., Zeeshan, A., & Hussein, H.S. (2026). AI-based analysis of heat transfer in Williamson nanofluids over a vertically stretching sheet. *International Journal of Hydrogen Energy*, 208, 153383. doi: 10.1016/j.ijhydene.2026.153383

[48] Akbar, Y., Li, K., Iqbal, J., Yang, X., & Mao, X. (2026). Thermodynamic analysis of nanofluid with aggregated and non-aggregated nanoparticles. *International Journal of Heat and Fluid Flow*, 118, 110205. doi: 10.1016/j.ijheatfluidflow.2025.110205

[49] Sokolov, A., Ali, R., & Turek, S. (2015). An AFC-stabilized implicit finite element method for partial differential equations on evolving-in-time surfaces. *Journal of Computational and Applied Mathematics*, 289, 101−115. doi: 10.1016/j.cam.2015.03.002

[50] Shoaib, M., & Javed, T. (2024). Sensitivity analysis of non-isothermal mixed convective heat and mass transfer in nanofluid flow over a horizontal circular cylinder with Brownian motion. *Numerical Heat Transfer, Part B: Fundamentals*, 85(8), 984−1008. doi: 10.1080/10407790.2023.2259088

[51] Aziz, T., Mahomed, F.M., Shahzad, A., & Ali, R. (2014). Travelling wave solutions for the unsteady flow of a third grade fluid induced due to impulsive motion of flat porous plate embedded in a porous medium. *Journal of Mechanics*, 30(5), 527−535. doi: 10.1017/jmech.2014.17

[52] Iqbal, J., & Akbar, Y. (2026). Sensitivity analysis of drag reduction and aerothermal heating in Hypersonic Glide Vehicles (HGVs) under magnetic fields and nanofluid coatings. *International Communications in Heat and Mass Transfer*, 171, 110032. doi: 10.1016/j.icheatmasstransfer.2025.110032

[53] Ali, R., Mitalipova, A., Hussan, S.W.U., Kurbanaliev, A., Shahzad, A., & Abduzhabbarov, M. (2025). Shape effects on steady flow and heat transfer of Cu-nanofluid over a nonlinear stretching surface with Joule heating. *Archives of Thermodynamics*, 46(3), 89−104. doi: 10.24425/ather.2025.156581

[54] Shoaib, M., Javed, T., & Iftikhar, B. (2026). Sensitivity analysis of MHD Casson hybrid nanofluid and heat transfer with Joule heating. *Multidiscipline Modeling in Materials and Structures*, 22(1), 192−217. doi: 10.1108/MMMS-02-2025-0053

[55] Akbar, Y., Iqbal, J., Turabi, Y.U.U.B., Anwar, T., & Alam, M.M. (2025). Magneto-radiative effects on thermal fields and irreversibility in a lid-driven wavy square enclosure with a central heater. *Journal of Thermal Analysis and Calorimetry*, 150(25), 21171−21184. doi: 10.1007/s10973-025-14973-4

[56] Ali, R., Iqbal, A., Abbass, T., Arshad, T., & Shahzad, A. (2024). Unsteady flow of silica nanofluid over a stretching cylinder with effects of different shapes of nanoparticles and Joule heating. *Archives of Thermodynamics*, 45(3), 115–126. doi: 10.24425/ather.2024.151222

[57] Alkinidri, M., Iqbal, J., & Abbasi, F.M. (2025). Thermal transport and irreversibility analysis for radially magnetized Carreau–Yasuda nanofluids flow over polished surfaces. *Modern Physics Letters B*, 39(29), 2550168. doi: 10.1142/S0217984925501684

[58] Iqbal, J., Abbasi, F.M., Alam, M.M., & Akbar, Y. (2025). Thermal and energy optimization of hybrid nanofluid flow in parabolic trough surface collectors (PTSCs) using statistical modeling, neural networks, and sensitivity analysis for solar-powered aircraft. *Physics of Fluids*, 37(8), 082065. doi: 10.1063/5.0279425