Research on Engineering Structures & Materials
www.jresm.org
Research Group

Research Article

Predicting viscosity and yield surfaces for Bingham fluids in a lid-driven cavity employing a deep neural network

Eduardo Henrique Taube Cunegatto[1,a], Flavia Schwarz Franceschini Zinani[*,2,b], Sandro Jose Rigo[1,c]

[1]*Applied Computing Graduate Program, Universidade do Vale do Rio dos Sinos, São Leopoldo, Brazil*
[2]*Institute of Hydraulic Research, Universidade Federal do Rio Grande do Sul, Porto Alegre, Brazil*

Article Info

Article history:

Received 22 Nov 2024
Accepted 20 Jan 2025

Keywords:

Yield stress fluids;
Deep learning;
Computational fluid dynamics;
Artificial neural networks;
Surrogate model;
Lid-driven cavity

Abstract

Computational Fluid Dynamics (CFD) simulation of viscoplastic fluid flows is challenging. It is critical to capture the shape of the yield surfaces that limit yielded and unyielded flow regions. These are highly sensitive to numerical schemes, which usually rely on calculating velocity derivatives and require highly refined computational grids and strict convergence criteria. In this work, CFD and Deep Neural Networks (DNN) were coupled to develop a model to predict yield surface morphologies for the flow of Bingham fluids in a square cavity. Our main goal is to predict the behavior of viscoplastic fluids using a Machine Learning (ML) approach based on CFD results. Design of Experiments techniques were employed in defining the base cases, i.e., combinations of Bingham and Reynolds numbers (Bn and Re) to compose the DNN training data. The DNN predicted the viscosity fields in the problem domain given Bn and Re. These results were postprocessed using masks to create binary images. The chosen architecture was an encoder-decoder since the input and output data had different dimensions. The results of the surrogate model were adequate, giving a Mean Squared Error of 0.0015 for the training data and 0.002 for the testing data. The DNN-predicted images were consistent with those generated from CFD, corroborating the proposed technique as an excellent alternative to be implemented in more complex applications. The combination of CFD and ML is a promising alternative for predicting complex fluid behavior in diverse and challenging scenarios with faster and computationally less expensive resources.

1. Introduction

A class of non-Newtonian fluids, called yield stress or viscoplastic fluids, receives much attention for its broad applicability in engineering. These fluids have the characteristic of presenting yield stress, a minimum stress level required for the material to flow. Below the yield stress, the material behaves as a solid. This non-smooth behavior is a model for the complex behavior of some macromolecular materials. At a specific range of stresses, these materials undergo some molecular configuration changes that cause the drop of their viscosity from a high plateau (solid-like behavior) to a value much lower in the range of a fluid's viscosity [1]. The primary examples of viscoplastic fluids of engineering interest include gels, muds, creams, and molten polymers. Viscoplastic fluids are also referred to as Bingham fluids after the seminal book by E.E. Bingham [2] and the widely employed mathematical model for viscoplastic fluids, the Bingham equation [3].

*Corresponding author: flavia.zinani@ufrgs.br
[a] orcid.org/0000-0002-2692-1479; [b] orcid.org/0000-0001-5402-900X; [c] orcid.org/0000-0001-8140-5621
DOI: http://dx.doi.org/10.17515/resm2025-544ml1122rs

The flow of viscoplastic fluids is usually complex because there may be regions below and above their yield stress along the flow domain. So, the fluid moves with yielded regions (stresses above the yield stress), in which the flow is viscous, and unyielded regions (stresses below the yield stress), in which the movement is the type of rigid body motion, with negligible strain rate. The unyielded regions along the flow usually increase heat and material flow resistance, pressure drop, and pumping power requirements.

The computational modeling of viscoplastic flows using computational fluid dynamics (CFD) has the challenge of capturing the frontiers of the yielded and unyielded regions called yield surfaces. The shape of yield surfaces is sensitive to the fluid's rheology, geometry, and operation conditions. Due to the modeling of the non-smooth material behavior, which depends on local values of stresses, the CFD simulations of viscoplastic flows are computationally intense. They are memory and time-consuming and must rely on finer grids compared to simulations of Newtonian fluid flows under the same conditions.

Numerous works have addressed viscoplastic flows inside lid-driven cavities to test CFD modeling and methods. Like in Newtonian flow, a primary vortex usually occurs in the cavity's center. The shear in such a vortex is so mild that its center usually does not yield. Figure 1 shows a schematic of the flow of a viscoplastic fluid inside a lid-driven cavity, with streamlines and yielded and unyielded regions represented in white and black, respectively.

Some works that employed the lid-driven cavity benchmark with viscoplastic fluids are summarized in the sequence. The work by Frey et al. [4] presented numerical approximations via Galerkin-least-squares-type formulation in a lid-driven cavity, with different ranges of Reynolds and Bingham numbers, providing reliable results for comparing the morphology of yielded and unyielded zones. Dos Santos et al. [5] also studied the lid-driven cavity, employing a regularized viscosity model and providing a parametrical analysis of the problem. The authors discussed the criterion (stress x strain rate) for the yielded and unyielded regions, where they found that using the strain rate provides smoother yield surfaces than those originating from the stress criterion. Syrakos et al. [6] examined the inertia effects of a Bingham fluid in a lid-driven cavity by modifying the viscosity function to employ the regularization of Papanastasiou [7]. By varying the range of both Bingham and Reynolds numbers up to 100 and 5000, respectively, they found that the discretization error increases with the Bingham number due to the discontinuation of the flow around the yielded regions caused by the increase of the truncation error of the regularization. Syrakos et al. [8] investigated a transient version of the lid-driven cavity for Newtonian and Bingham fluids, where a sudden flow stoppage and its effects are analyzed. Such study presents benchmark cases, covering Reynolds and Bingham numbers from 1 to 1000 and 1 to 10, respectively, to be used in further investigations. Kefayati & Huilgol [9] studied the mixed convection in the lid-driven cavity filled with Bingham fluids by solving the model equations via the Lattice Boltzmann Method (LBM). Using such modeling, the authors could find results equivalent to those of more traditional methods, like the finite volume method (FVM). The study also presents an analysis of heat transfer, showing the influence of temperature and of the Prandtl number on flow behavior. Mahmood et al. [10] studied the lid-driven cavity with single and

double lids, also considering Bingham fluids. They found out that, for the double lid, the creation of the unyielded regions happens at the exact center of the cavity, increasing its size as the Bingham number grows. Also, for lower Bingham numbers, the behavior is more symmetric when compared to higher values. Thohura et al. [11] studied Bingham fluids in skewed lid-driven cavities. The study covers skew angles from 45° to 135°, Bingham numbers from 0 to 500, and Reynolds numbers up to 1000, providing benchmark results for the problem. Bui et al. [12] also study skewed cavities, besides square and rectangular ones, focusing on the behavior of Carbopol 940, which was modeled via a combination of Herschel-Bulkley and Papanastasiou models. While keeping the Reynolds number fixed at a value of 0.001, the authors analyzed the effect of the Oldroyd number, ranging from 0 to 50. Ferrari and Franco [13] studied squared cavities, modeling transient viscoplastic flows with the LBM method. Covering a range of Reynolds numbers up to 20000 and Bingham numbers up to 10, the authors analyzed the behavior of the flow until a steady state was reached. The results showed that the time to reach a stationary state increases as the Bingham number grows despite the Reynolds number. Ferrari and Franco [14] expanded their previous study by analyzing the steady and unsteady behavior of a viscoplastic fluid in a cubic cavity, extending the existing dataset. Once again, using the LBM method, the authors found that the three-dimensional approach appears more unstable than the bi-dimensional one, where some aspects of the flow could not be captured. The authors recommend using a Herschel-Bulkley fluid for comparison, as it is easier to compare with real fluids.

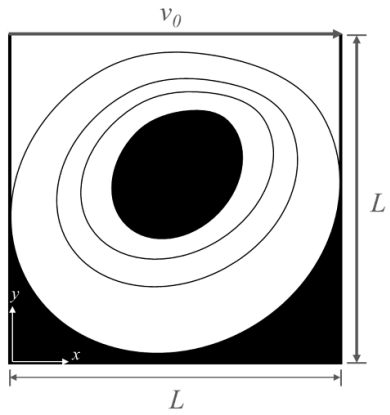


Fig. 1. A representation of viscoplastic fluid flow inside a lid-driven cavity

In recent years, deep neural networks (DNN) have become a versatile and valuable tool for optimizing results in several computational tasks by predicting complex aspects in diverse application contexts. The consistent and relevant results observed through DNN applications generated interest in their use in several fields, such as fluid dynamics. In this context, their use has been highlighted as a valuable method for modeling and predicting many properties and features in many Computational Fluid Dynamics (CFD) applications [15]. It is known that traditional CFD models can be computationally expensive and time-consuming, especially in complex geometries or refined mesh cases. In this context, surrogate models via DNNs have emerged as a promising alternative to accelerate CFD

simulations while maintaining good accuracy [16]. Thus, DNNs are an alternative to help reduce simulations' computational costs using CFD-generated data.

Many numerical methods can provide high-fidelity solutions for viscoplastic flows that may be post-processed to give the morphology of yielded and unyielded regions along the computational domain. If the shape of yield surfaces is the primary interest, such problems could be simplified if only this object of interest is solved, reducing time and computational cost in pre-processing and post-processing. Previous works have explored this concept using machine learning models trained to reproduce and generate CFD-like solutions for many applications. For example, Convolutional Neural Networks (CNN) architectures, coupled with CFD, have been used in heat transfer [17] [18] [19] [20], air/hydrofoil optimization [21] [22], ship designs [23], hemodynamics [24], prediction of porous media [25], performance of solar collectors in [26] and thermal analysis of bearings [27]. Other DNN architectures, such as fully connected with backpropagation, can be found in the works of Kim & Park [28] who applied in indoor airflow prediction, Sang et al. [29] for the prediction of wind load of buildings and Hussain et al. [30], in the modeling of mixed convection in nanofluids. However, regarding viscoplastic flows, the application of DNNs is relatively scarce in the literature. Muravleva et al. [31] used a combination of proper orthogonal decomposition (POD) and a fully connected DNN to build reduced-order models for Bingham flows in ducts. Kumar et al. [32] applied an DNN to simulate the flow of a viscoplastic fluid in a pipe, implementing physical governing equations as the loss function. Similarly, Zhang [33] developed an Augmented Lagrange Method Physics Informed Neural Network (ALM-PINN) to solve steady-state one-dimensional Poiseuille flow and bi-dimensional plane lid-driven cavity for Stokes flow of Bingham fluids. Although the current model is limited to small Bingham numbers, using such architecture, the authors could induce feasible loss functions for the network by formulating them as a linear combination of the mean squared error of the governing equations, providing a mesh-free feature, and developing a versatile network configuration.

Driven by the promising use of machine learning, the current study proposes the use of a DNN, in the form of an encoder-decoder architecture, combined with CFD data, as a surrogate model to predict viscosity contours and yield surfaces for viscoplastic flows inside a lid-driven cavity and different combinations of Bingham and Reynolds numbers. This work also includes the steps to create a custom dataset from CFD simulations, a detailed configuration of the encoder-decoder neural network, and the analysis of the model's performance. As the main objective, this study aims to introduce a versatile approach to studying viscoplastic flows in the lid-driven cavity by applying DNNs. With the excellent accuracy achieved in this work, we expect to establish a foundation for future research in this field and motivate the computational fluid dynamics community to use the capabilities of neural networks, especially when dealing with fluids of complex behavior.

2. Methodology

This section covers all the methods utilized in this work to support the steps taken to achieve the current results and for future use.

2.1. Problem Description

The system under analysis is a bi-dimensional square, representing a cavity filled with a viscoplastic fluid. The flow behavior in such fluids is characterized by the yield stress, the stress level above which the material behaves as a viscous fluid. Below the yield stress, the material would behave as a solid or, as accepted nowadays, as a highly viscous fluid. At unyielded regions, strain rates are negligible.

As presented in Figure 1, the cavity upper lid moves with tangential velocity v0, causing the flow inside the cavity. The remaining edges represent the walls of the cavity, which have length L. Two dimensionless numbers are associated with this physical problem: the Reynolds and Bingham numbers. The Reynolds number (Re) indicates the flow regime and measures the imbalance between inertial and viscous forces. The Bingham number (Bn) measures the fluid's degree of viscoplasticity relative to the system scale. As previously mentioned, Figure 1 is a schematic of the described system, which depicts a representative configuration of yielded and unyielded regions in white and black, respectively.

2.2. Mathematical Modeling

To model the problem, the flow was considered incompressible and in steady state. The balance equations of mass and momentum, Equations (1) and (2), respectively, were used.

$$\nabla \cdot \boldsymbol{v} = 0 \tag{1}$$

$$\rho[(\nabla \cdot \boldsymbol{v})\boldsymbol{v}] = -\nabla P + \nabla \cdot \boldsymbol{\tau} \tag{2}$$

in which $\boldsymbol{v}$ represents the velocity vector, P is the pressure field and ρ is the fluid`s density. In Equation (2), $\boldsymbol{\tau}$ represents the extra stress tensor, mathematically expressed by:

$$\boldsymbol{\tau} = 2\eta(\dot{\gamma})\boldsymbol{D} \tag{3}$$

where $\boldsymbol{D}$ is the strain rate tensor and $\eta(\dot{\gamma})$ represents the viscosity function. In the viscosity function, $\dot{\gamma}$ is the magnitude of the strain rate tensor, which is mathematically represented as:

$$\dot{\gamma} = \sqrt{2tr(\boldsymbol{D} \cdot \boldsymbol{D})} \tag{4}$$

where the *tr* represents the trace operator.

The viscosity function for viscoplastic liquids was modeled using the Bingham model [2], regularized as a bi-viscosity model as follows [34]:

$$\eta(\dot{\gamma}) = \tau_0 \frac{\left(2 - \frac{\dot{\gamma}}{\dot{\gamma}_c}\right)}{\dot{\gamma}_c} + k, \qquad if\ \dot{\gamma} < \dot{\gamma}_c \tag{5}$$

$$\eta(\dot{\gamma}) = \frac{\tau_0}{\dot{\gamma}} + k, \qquad if\ \dot{\gamma} > \dot{\gamma}_c \tag{6}$$

where k is a consistency parameter and τ_0 is the yield stress. The Bingham original model predicts infinite viscosity at stresses lower than τ_0 and the viscosity as in Equation (6) for stresses above τ_0. The present regularized model uses the critical shear rate, $\dot{\gamma}_c$, to give a

continuous stress behavior as a function of the shear rate avoiding infinite viscosity. Below the $\dot{\gamma}_c$, Equation (5) predicts high viscosities, and above $\dot{\gamma}_c$, Equation (6) predicts the same viscosity as the Bingham model. Equations (5) and (6) approximate the original model behavior as $\dot{\gamma}_c$ is lower. So $\dot{\gamma}_c$ must be chosen at a low value to approximate the results expected for the Bingham original model. In this work, the relation $\dot{\gamma}_c/v_0$ was chosen to be always lower than 0.05, because we have detected that the simulation results were insensitive to $\dot{\gamma}_c$for any $\dot{\gamma}_c L/v_0$ lower than 0.05. As previously mentioned, Bingham (Bn) and Reynolds (Re) are the main dimensionless numbers of this problem, which are mathematically given, respectively, by:

$$Bn = \frac{\tau_0 L}{k v_0} \tag{7}$$

$$Re = \frac{\rho L v_0}{k} \tag{8}$$

In this problem, Bn and Re numbers are varied, ranging from 1 to 10 and 1 to 250, respectively.

2.3. Numerical Modeling

The software used to solve the governing equations was ANSYS Fluent 2023R2 [34]. For the pressure-velocity scheme, COUPLED was adopted, and for pressure and momentum interpolation, the second-order model was employed. The convergence criteria for mass and momentum were set to 10^{-10}. The numerical mesh was built by dividing the domain into a grid of 200x200 finite volumes, which was considered by using the works of Neofytou [35], Dos Santos et al. [5] and Kefayati & Huilgol [9] as references, which reported that such configuration provided reliable results regarding the velocity profiles and the morphology of the yielded and unyielded surfaces of the fluid. The boundary conditions set for the problem (see Fig. 1 for reference) are prescribed velocity v_0, at the lid (upper), which is defined by the Re number; the remaining boundaries (left, right and bottom) of the cavity are considered as no-slip walls.

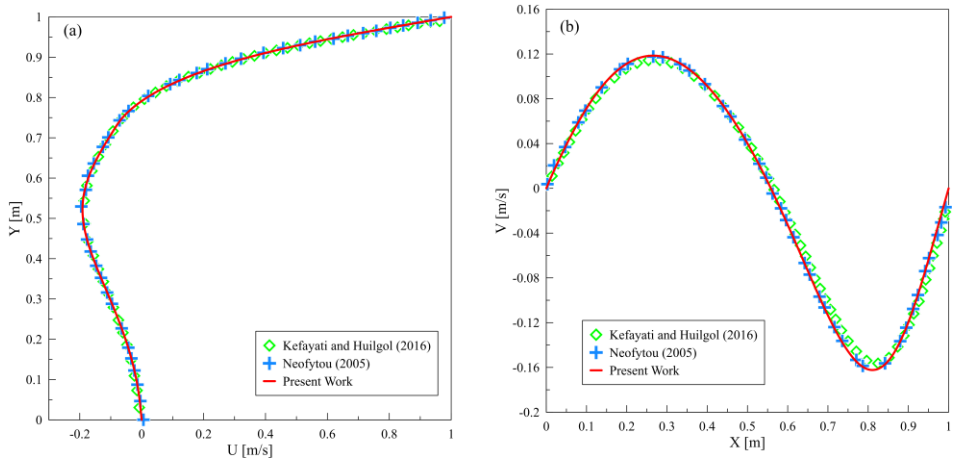


Fig. 2. Velocity profile comparison for model validation: (a) U velocity; (b) V velocity

A verification was performed to ensure that both model and mesh were adequately employed. For this, the u and v velocity profiles along the y and x axis were compared with those from [35] and [9] for Bn = 1 and Re = 100, which are shown in Figure 2. The result shown in Figure 2 shows that both the model and mesh are in accordance with the other studies.

2.4. Design of Experiments for Data Generation

The dataset used to train the artificial neural network was created using Desing of Experiments (DOE) methods with a statistical basis. According to Montgomery & Wiley [36], employing such techniques increases experiment performance and provides high-quality data. Thus, two experimental methods were used to apply the DOE: Central Composite Design (CCD) and Latin Hypercube Sampling (LHS).

The CCD compromises a set of points in the design space, which is defined by the limits of the variables. Those points include 2k factorial model points, where k is the number of factors or variables, axial points, and a central point. As pointed out by Cunegatto et al. [37] and Cunegatto et al. [38], using a CCD provides a general idea of the outputs within the design space. The values for the CCD employed in this work are shown in Table 1 below:

Table 1. Values of the CCD design space.

Variable	Levels				
	Low axial level	Low 2^k value	Central value	High 2^k value	High axial value
Bn	1	2.318	5.5	8.682	10
Re	1	37.465	125.5	213.535	250

For a problem with two variables, such as the one presented here, a Central Composite Design provides nine points. However, as mentioned by Misiulia et al. [39], a surrogate model requires a certain amount of information to understand and capture the relationship between the input and output data. Due to this and based on the suggestion of Yang & Palazzolo [27], the LHS was also employed for the data generation.

According to Stein [40], LHS is a sampling method used to produce input values for function predictions. Since the number of points generated by CCD is fixed, using LHS allows the generation of more sampling points. So, to provide enough data to train the surrogate model, a set of 111 samples was generated using the LHS, resulting in a total of 120 points. Those extra samples were generated considering the points provided by the CCD to avoid overlapping.

2.5. Deep Neural Network (DNN)

The core idea of using artificial neural networks as surrogate models as a predictive tool is to reduce time in preparing and solving traditional CFD models, that is, for subsequent analysis, since it requires the CFD data to learn the patterns. That is, for the algorithm to be able to make reliable predictions, it must learn from some data. Thus, in the current work, CFD data is used to teach the algorithm, and, consequently, the predictions are based on the data used in the learning, much like statistical models that use the available data to make predictions using an equation. In machine learning modeling, it is essential to divide the dataset (collection of CFD results in this case) into training and testing examples; the

first is used to teach the algorithm and the second, which are examples unseen during training, are used to evaluate the model. In the current study, the DNN needs to predict the viscosity contours of the flow inside the lid-driven cavity for a certain combination of Bn and Re numbers, two numerical parameters.

One key characteristic of the current study is that geometry is fixed. For any variable combination, the square cavity remains the same. The only thing that changes is the behavior of the viscosity, which is influenced by the input parameters. Knowing such information is crucial in defining the architecture of the neural network. For example, in the work of Thuerey et al. [21], many geometry configurations are studied, so the DNN must receive the geometry information to predict the desired outcome, which, in their case, is an image. Thus, the architecture employed by the authors was a Convolutional Neural Network (CNN).

As previously mentioned, two numerical parameters define the problem and are used as the input parameters for the neural network. Thus, the DNN should use the information and convert it to the output, which is the viscosity contour. Such a process can be achieved by using an architecture known as encoder-decoder. According to Minaee et al. [41], the encoder-decoder model can learn to map data from input to output using two stages: the encoder, which compresses the input data into a latent-space representation, and the decoder, which uses the latent-space representation to predict the output. The encoder-decoder architecture used in the present work is shown in Figure 3.

The neural network was built in Python computational language by using the TensorFlow [42] and Keras [43] libraries. As shown in Figure 3, each layer in the architecture is labeled as 'Dense', the denomination used in Keras library for fully connected layers, meaning that the neurons in a current layer are connected to the ones in the previous layer. The activation function used in the current work for such layers is the Rectified Linear Unit (ReLU). Also, the values under each layer represent the number of neurons within it. The architecture shows a symmetric behavior about the latent-space layer, as stated by Géron [44], in a way that the encoder gradually reduces the input data to create a compact representation in the latent-space layer. Then, the decoder rebuilt the compact representation, progressively expanding the dimension. The output, also a fully connected layer, is expanded to the size of the desired image, which in this case is 224x224x3, resulting in over 150 thousand neurons that carry the information of the image. Then, the image is recreated by reshaping the output layer into a shape of 224x224 pixels and three-color channels.

As mentioned in Section 2.4, the dataset was built by CFD simulations of the points provided by the DOE methods. Then, the viscosity images were collected and saved into a folder. A data frame containing the input variables, image type, and image directory was created to manipulate the data. Since the original shape of the image differs from 224x224 pixels, the images were resized using the OpenCV library [45].This library was also used to read the images from the folder and transform them into an array with the NumPy library [46].

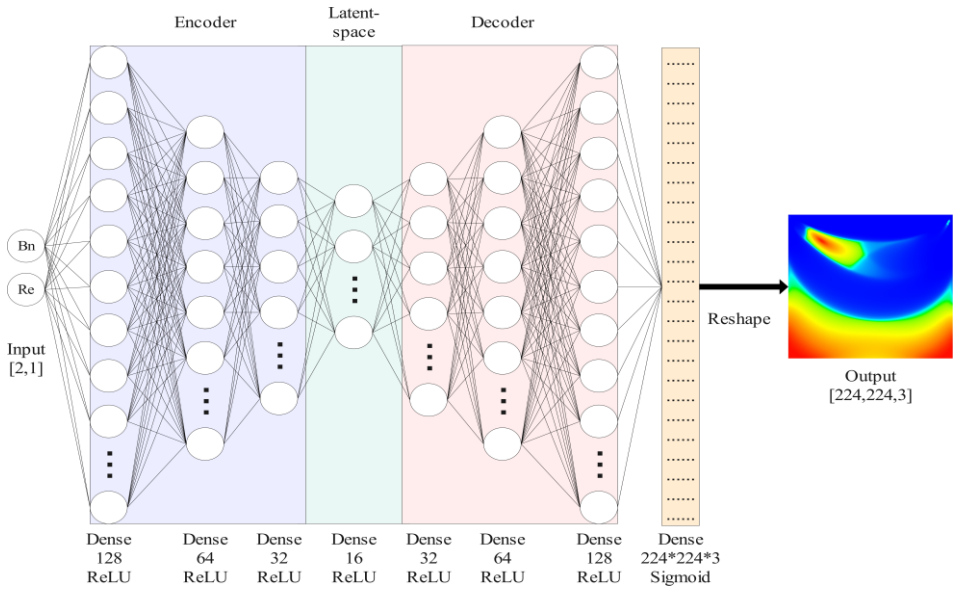


Fig. 3. Encoder-decoder neural network architecture

A crucial part of the data treatment is the data scaling. Machine learning algorithms have issues in their performance when the data have different scales [44]. For example, the Re numbers used in this work vary from 1 to 250, while the Bn number ranges from 1 to 10, which is a great difference between them. So, both input and output data were scaled to perform well. For the input values (Bn and Re), a min-max scaling was used (Eq. (9)), resulting in a range of 0 to 1 for both parameters. As for the output images, the main reason that they were transformed into an array is that each pixel value can be manipulated, allowing its scaling. Since the colors from an image vary from 0 to 255, the scaling was done by dividing each pixel value by 255, resulting in a range of 0 to 1 for the output image. This range was also the reason for using the Sigmoid activation function at the output layer of the neural network since it ranges from 0 to 1.

$$x_i' = \frac{x_i - min\ (x)}{max(x) - min\ (x)} \tag{9}$$

where x_i' is the single scaled variable, x_i is the single variable, min(x) is the minimum value of the variable x and max(x) is its maximum value. In the previous section, some hyperparameters were mentioned, such as the activation functions, the number of layers, and the number of neurons in each layer. However, a few more hyperparameters are needed to provide the complete setup of the employed neural network.

- Train/test split: The dataset was randomly divided to feed the neural network, with 80% used for training and the remaining 20% used for testing.
- Optimizer: The optimizer is an algorithm that helps the neural network adjust its weights to minimize the loss during training. The optimizer chosen in this work is Adam (Adaptive Momentum Estimation). The Adam optimizer has its hyperparameters, the momentum decay and scaling decay, which were set to 0.9 and 0.999, respectively.

- Learning rate: This hyperparameter is related to the optimizer and determines how quickly the weights are adjusted. Since a low value may cause the training process to be slow, and a high value may cause divergence, a tuning was performed, which returned 0.007 as an adequate value.
- Batch size: it defines the number of training samples used in each iteration of weight adjustment. Tuning was also performed, and a value of 8 batches was adopted.
- Number of Epochs: This defines the number of times the training data passes through the neural network during training. One epoch is complete when all the training data, i.e., batches, pass through the neural network once. In this work, 150 epochs were used.

As mentioned above, the optimizer minimizes the loss function when the neural network weights are adjusted during training. The loss function evaluates the neural network's performance, sharing a similar trait as the residuals in CFD, by comparing the predicted values with the CFD data at each epoch. Among the loss functions utilized in machine learning, the Mean Squared Error (MSE) was chosen for this study since the DNN predicts the scalar value of each pixel in the image. The MSE is mathematically expressed by Equation (10):

$$MSE = \frac{1}{N}\sum_{i,j,k=1}^{N}\left(y_{ijk} - \widehat{y_{ijk}}\right)^2 \tag{10}$$

where N represents the total number of pixels, y is the predicted value, $\hat{y}$ is the CFD value, i represents the index of the image length, j is the height index and k is the color channel index.

3. Results

This section thoroughly examines the problem as a physical phenomenon, focusing on the viscosity field and the morphology of yield and unyielded regions along the flow. In the sequence, the performance of the DNN is discussed, where comparisons between predicted and CFD viscosity contours are performed, along with the training metrics. Lastly, we explore using masks to generate the yield surfaces, i.e., the morphology of yielded and unyielded regions, using the predicted viscosity contours as a basis.

3.1. Viscosity and Yield Stress Physics

Before diving into the problem itself, a brief analysis of Re and Bn numbers is performed. The Reynolds number (Eq. (8)) is one of the most important dimensionless numbers in fluid dynamics, as it predicts the behavior of a flow by calculating the ratio between the inertial and viscous forces [47]. As the Reynolds number increases, the influence of the viscous force becomes less significant, which makes it way more challenging to hold the fluid's disturbance, resulting in more momentum in the flow. As observed, the viscosity, i.e., viscous force, is essential in defining a fluid's behavior. This property becomes more complex regarding viscoplastic fluids (Bingham plastics) because it is non-constant and dependent on yield stress. The connection between the viscous force and yield stress is found in the Bingham number (Eq. (7)), representing the ratio between yield and viscous stresses [48]. So, as the Bingham number increases, the stress levels required to yield the

material are higher. Conversely, the decrease in Bn makes the fluid yield more easily, even behaving as Newtonian if Bn = 0. So, by analyzing Equations (7) and (8), it's essential to notice that the only parameters not interchangeable between Bn and Re are τ_0 and ρ, hence why those properties are the ones that define the dimensionless numbers in this work, as mentioned in Section 2.

Returning to the lid-driven cavity problem, Figure 4 presents viscosity contours for different combinations of Re and Bn numbers. Just as a matter of convenience, the viscosity values for each image were scaled from 0 to 1 using the min-max scaling (Eq. (9)), where the red color indicates a very viscous zone, and the blue color indicates a zone with low viscosity. As observed, the main characteristic of the problem is the presence of two distinct viscous regions, which, according to Bui et al. [12], are called the dead zone at the bottom and the moving rigid zone at the upper central region of the cavity. For a low Reynolds number, the viscosity is symmetrically distributed, as the inertial forces are balanced. The moving rigid zone tends to concentrate on the left side for higher Reynolds numbers. In contrast, the dead zone is nearly unaffected by the Reynolds number.

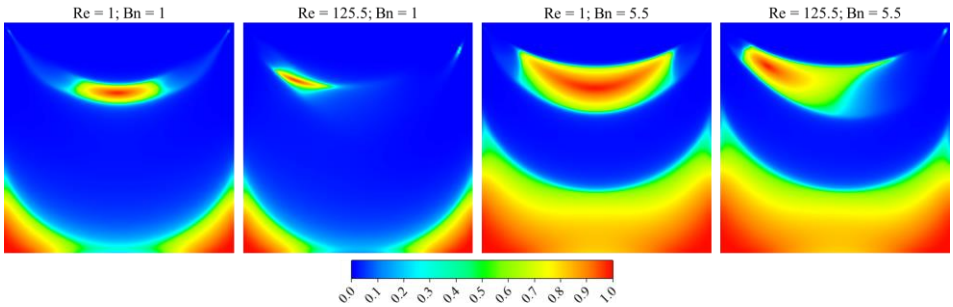


Fig. 4. Dimensionless viscosity contours for different Reynolds and Bingham numbers

Those two regions are developed due to fluid movement inside the cavity, shown in the form of velocity contours in Figure 5. As the fluid flows from left to right, it forms a vortex characterized by unyielded fluid in rigid body motion. The position of the vortex changes as the Reynolds number increases, moving to the right side of the cavity due to the increase in the momentum when the fluid hits the wall, quite the opposite of the moving rigid zone in the viscosity contours. Consequently, the vortex influences the moving rigid zone, which has its formation beside it. However, when considering the Bingham number, the flow becomes slower as the fluid is more rigid, which alters the position and size of the vortex, which is the case for the development of larger moving rigid zones. As for the dead zone, it is formed due to the low movement of the fluid at the bottom of the cavity, as it is far from the motion source (lid). In addition, its size slightly decreases from the augment of Re but increases as Bn grows.

Another important result that can be extracted from viscoplastic flows is its morphology, characterized by the unyielded and yielded regions and the flow aspect when under and above the fluid's yield stress, respectively. Generally, this information is obtained by an iso-surface, using either the critical shear rate for Herschel-Bulkley modeling or the yield stress as a threshold, where the yielded region is white and the unyielded is black. Figure 6 shows the morphology of the cases analyzed in Figures 4 and 5.

In this work, a critical shear rate of 0.05 was used as the input for the Herschel-Bulkley model and, subsequently, as the threshold of the iso-surfaces. This representation is relatable to the viscosity contours, where comparing the unyielded (black) zones with the viscous zones in the viscosity contours is possible. Despite the viscosity contours providing the overview of the property, it does not tell exactly where the fluid yields, hence why the morphology representation is commonly found in the literature, as it precisely shows that information. Due to the similarity between both viscosity contours and morphology, the same analysis regarding the formation of the unyielded zones done previously also applies here, and, in terms of computational vision, the morphology can be characterized as the mask of the viscosity contour.

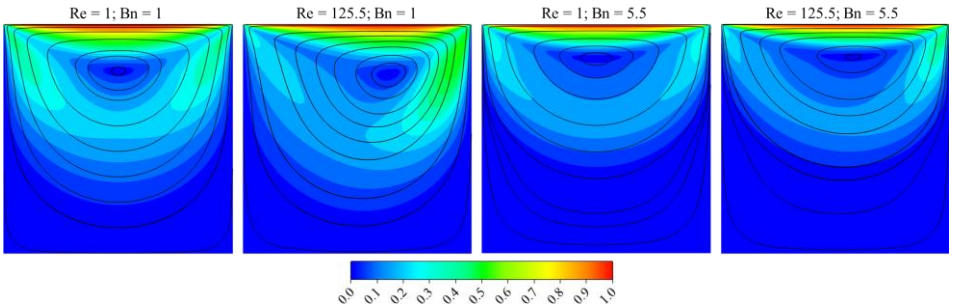


Fig. 5. Dimensionless velocity contours and streamlines for different Reynolds and Bingham numbers

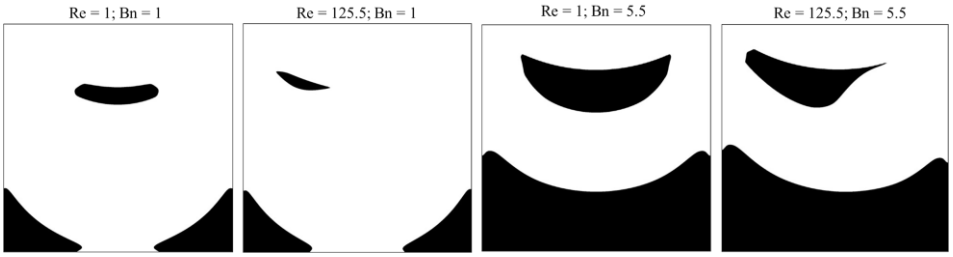


Fig. 6. Morphology of the flow for different Reynolds and Bingham numbers.

3.2. DNN Performance

As mentioned in Section 2, MSE was employed as the loss function and as a measure of the accuracy of the encoder-decoder neural network in predicting the viscosity fields. The MSE quantifies the average squared difference between predicted and CFD values, and a lower MSE means a better alignment between the data.

In the tests performed in this work, the model demonstrated satisfactory MSE values on both training and testing datasets. The MSE reached approximately 0.0015 for the training set, a value good enough to showcase the network's ability to learn and adapt to the training data. As for the testing set, the MSE was around 0.002, indicative of the model's generalization performance to unseen data. Figure 7 illustrates the training and testing loss over epochs during training to represent the neural network's learning progress visually. The x-axis represents the training epochs, while the y-axis depicts the corresponding loss (MSE) values.

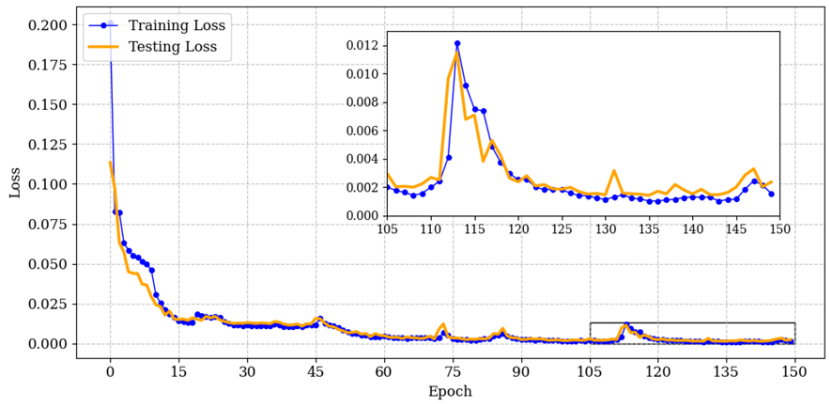


Fig. 7. Training process of the neural network

The training loss curve (blue) demonstrates a steady decrease over the initial epochs, indicative of the model learning from the training data. However, as the training progresses, the loss presents fluctuations, slowing the training a bit, although still decreasing its value. The zoomed section in Figure 7 presents a more detailed examination of the model's performance within a specific range of epochs, from 105 to 150, highlighting the network's behavior during the final training phase. The testing loss curve (orange), which evaluates the model's performance on unseen data, also presents a similar trend, affirming the generalization ability of the deep neural network.

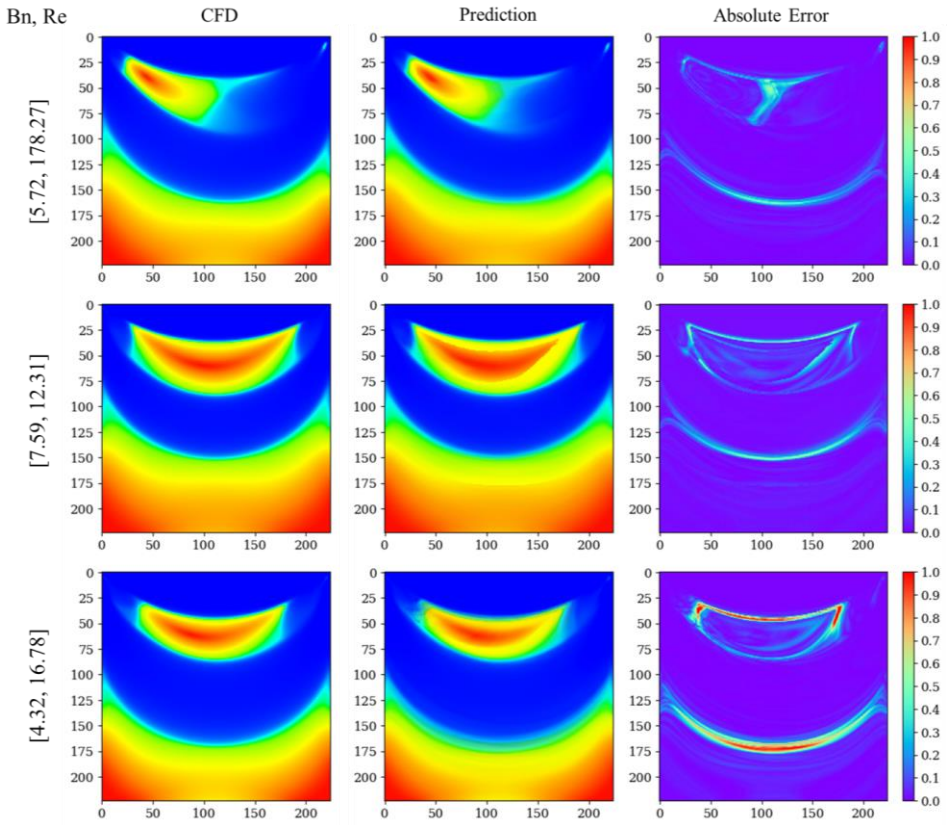
To bring a better understanding of the model's capability to predict the viscosity fields, Figure 8 shows the visual representation of the CFD fields and the corresponding prediction of the viscosity fields (scaled from 0 to 1), alongside with the difference (absolute error) between the images for five different samples of the training set.

Comparing CFD and model viscosity fields can highlight areas of agreement and discrepancy between the predicted and actual values. Simultaneously, the absolute error image portrays the spatial distribution of errors, which is the absolute difference of the pixels, accentuating regions where the model either excels or encounters difficulties. Overall, looking at the images side by side and observing the model's capability to capture and reproduce the actual phenomena is possible, providing reliable predictions.

It is important to emphasize that all the original images were resized to a smaller resolution, from 1600x1600 to 224x224, to provide quicker training for the model and reduce its computational cost. However, the tradeoff for lowering the resolution is losing information from the original image (high resolution). Thus, since the model did not account for the missed information, the predictions presented a higher discrepancy, resulting in differences in the average error plots in Figure 8, especially in the regions where the viscosity gradients are present. Overall, the results obtained by the model are adequate and provide a good overview of the viscosity field behavior for a Bingham fluid flow inside the lid-driven cavity. However, the current setting is limited, as only 120 images were used in the training, and the predictions present some errors in delicate regions, which might influence the pos-processing, especially when the flow morphology is extracted. Another limitation of the current model is that the exact (real) viscosity values for each pixel (node) are not considered, only the color value itself. Still, the morphology

can be extracted from the current results. As mentioned in the last section, the yielded and unyielded regions are affected by the viscosity field behavior, and the prediction errors might propagate into the computational domain. In this section, we use a mask, i.e., a threshold value, to separate regions on an image to capture the shapes of the yielded and unyielded regions. Since the exact values of the strain rate and stress are not present, the creation of the mask relies on defining a color threshold for the predicted viscosity fields.

From the RGB prediction image, the primary reference for the mask is the blue color, which represents a low viscosity in the viscosity fields, indicating a region where the fluid is yielded. So, the task is to find the threshold separating the blue color from the others to create a binarized image (black and white) representing the morphology. So, for this, the blue channel of the predicted viscosity field image serves as the basis for creating the binary mask. A thresholding algorithm was applied, where the intensity of the pixels was tested from 0 to 255, and a pixel intensity of 110 provided the best results. Thus, values over 110 are classified as yielded zones (white), while those below the threshold represent unyielded regions (black). Figure 9 shows the resulting mask, the predicted viscosity field, the CFD data, and the difference between them, considering the same examples in Figure 9.



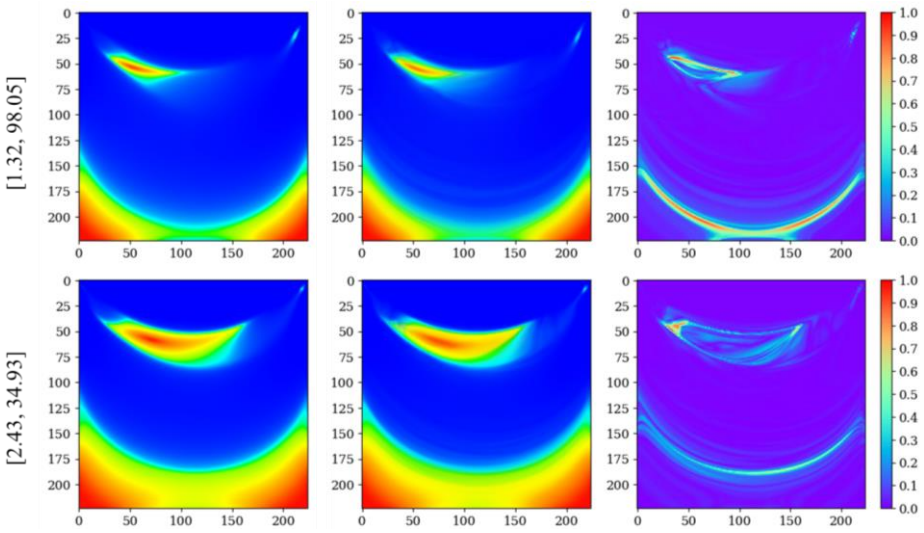


Fig. 8. Viscosity field predictions and absolute error for five testing samples

Based on the results shown in Figure 9, it is possible to confirm that the model predictions are reliable. Still, using masks to identify the yielded and unyielded zones is also adequate. The yielded zones are depicted in white, while the unyielded regions are represented in black. This binary mask offers an intuitive and easily interpretable fluid morphology predicted by the neural network. Also, despite the absolute differences presented in Figure 9, the morphology was not significantly affected, resulting in accurate representations of the real phenomena, as highlighted by the different images. However, a critical piece of information should be considered. Depending on the colors and intensity of the predicted image, the threshold to generate the mask should change. Despite being simple and proven to be a reliable way to reproduce the flow's morphology, it is essential to apply it carefully, as the physical values are not used since the model did not calculate them to generate consistent predictions.

4. Conclusions

In the present study, the lid-driven cavity problem with viscoplastic liquids was revisited by coupling Computational Fluid Dynamics (CFD) results and artificial neural networks to reproduce and predict the shape of yielded/unyielded regions for different Bingham (Bn) and Reynolds (Re) numbers. The dataset was created via Design of Experiments (DOE), where the samples were generated using the Central Composite Design and Latin Hypercube Sampling techniques. Those samples were simulated via CFD, which provided the images of the viscosity fields and the binary images of the flow's morphology for the training and testing of the artificial neural network. The chosen network architecture for the task was an encoder-decoder, which encoded the given Bn and Re and decoded them into the corresponding viscosity field.

Before the network training, the CFD modeling was evaluated, and the comparison with the literature showed enough agreement to provide the data to train the neural network. The training was evaluated by the Mean Squared Error (MSE) metric, which compared the values for each pixel between the predicted and CFD images. A further approach to the

problem was also explored by using the predicted viscosity fields to create binary images representing the flow's morphological behavior. In this approach, the use of masks to separate the yielded and unyielded regions was proposed. This involved the extraction of the adequate color channel and, later, defining a threshold intensity to separate the pixels.

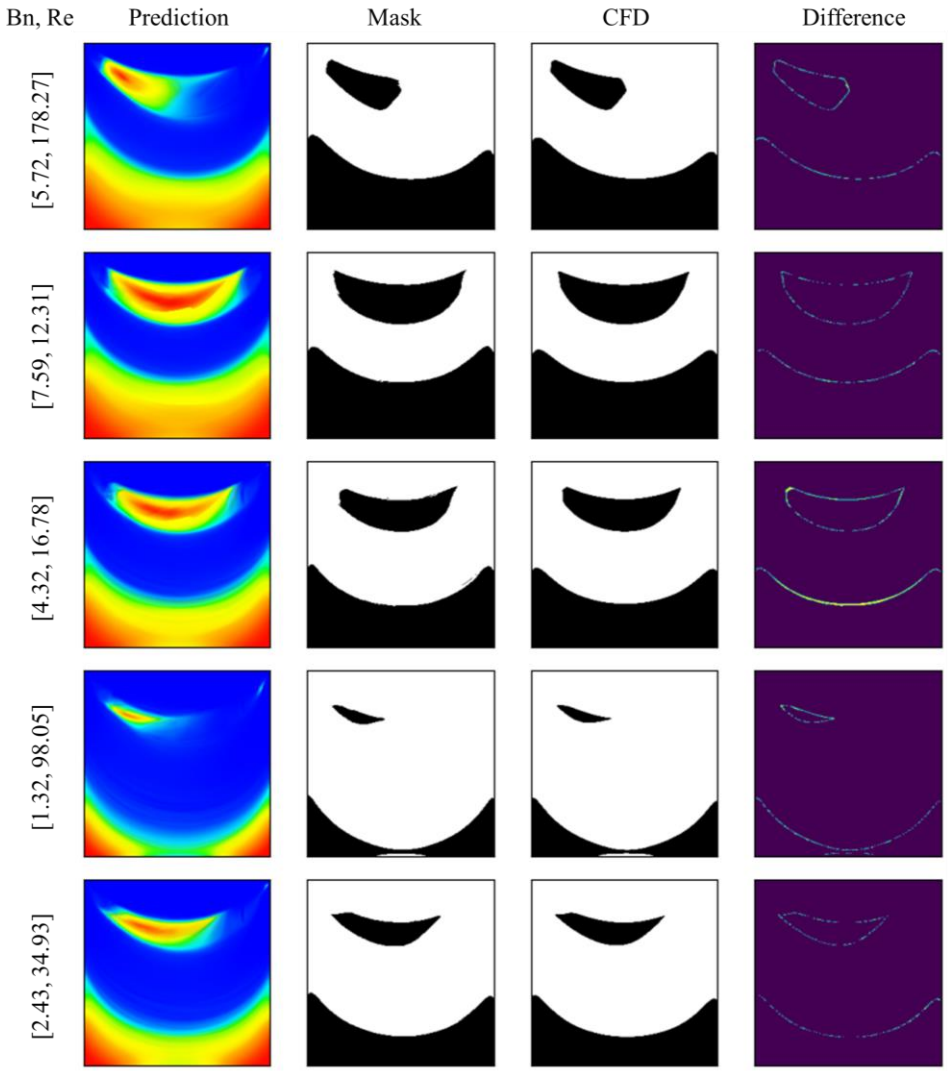


Fig. 9. Generated masks from the predicted viscosity fields for yielded and unyielded regions

As for the results of this study, they provided insights into the effectiveness of an encoder-decoder like neural network architecture in predicting viscosity fields. The quantitative analysis, measured by Mean Squared Error (MSE), showed a robust performance during training and validation, indicating the model's remarkable generalization capability. Such performance resulted in solutions that agreed with the CFD data, providing reliable predictions of viscosity fields for different flow configurations and allowing post-processing to extract the data regarding the flow's morphology. This feature was proposed

by creating binary images from the blue channel of the predicted image, which allowed a clear visual interpretation and highlighted the yielded and unyielded zones of the fluid. These binarized images are interesting in the analysis of viscoplastic flows, as they reveal crucial information about the fluid behavior. Thus, the proposed masks originating from the model's prediction provided a valid alternative for post-processing the results of the current model since the physical values were not considered. Despite the absence of such information, the resulting masks adequately represented the phenomenon, as the differences from the CFD data were minor.

It shall be reinforced that the current model was developed for a specific case test, but could be applied to other flows and geometries, such as flows in contractions and expansions. As observed in this study, the encoder-decoder architecture proved to be a versatile way to model problems with different types of input and output by transforming and associating the information provided by each source. This further opens new perspectives of fluid modeling by coupling intelligent systems with CFD to create fast and reliable solutions for engineering problems.

Acknowledgement

The authors acknowledge the financial support of the Brazilian agencies CAPES, CNPq, and FAPERGS. E.H.T. Cunegatto has a PhD scholarship funded by CAPES (Code 001). F.S.F. Zinani and S.J. Rigo are CNPq grant holders (311444/2021-0 and 305635/2021-2).

References

[1] Souza Mendes PR, Dutra ESS. Viscosity function for yield-stress liquids. Appl Rheol. 2004;14:296–302. https://doi.org/10.1515/arh-2004-0016
[2] Bingham EC. Fluidity and plasticity. New York: McGraw-Hill; 1922.
[3] Chhabra RP, Richardson JF. Non-Newtonian flow and applied rheology. Elsevier; 2008. https://doi.org/10.1016/B978-0-7506-8532-0.X0001-7
[4] Frey S, Silveira FS, Zinani F. Stabilized mixed approximations for inertial viscoplastic fluid flows. Mech Res Commun. 2010;37:145–52. https://doi.org/10.1016/j.mechrescom.2010.01.005
[5] dos Santos DDO, Frey S, Naccache MF, de Souza Mendes PR. Numerical approximations for flow of viscoplastic fluids in a lid-driven cavity. J Nonnewton Fluid Mech. 2011;166:667–79. https://doi.org/10.1016/j.jnnfm.2011.03.004
[6] Syrakos A, Georgiou GC, Alexandrou AN. Performance of the finite volume method in solving regularised Bingham flows: Inertia effects in the lid-driven cavity flow. J Nonnewton Fluid Mech. 2014;208–209:88–107. https://doi.org/10.1016/j.jnnfm.2014.03.004
[7] Papanastasiou TC. Flows of materials with yield. J Rheol (N Y N Y). 1987;31:385–404. https://doi.org/10.1122/1.549926
[8] Syrakos A, Georgiou GC, Alexandrou AN. Cessation of the lid-driven cavity flow of Newtonian and Bingham fluids. Rheol Acta. 2016;55:51–66. https://doi.org/10.1007/s00397-015-0893-4
[9] Kefayati GHR, Huilgol RR. Lattice Boltzmann method for simulation of mixed convection of a Bingham fluid in a lid-driven cavity. Int J Heat Mass Transf. 2016;103:725–43. https://doi.org/10.1016/j.ijheatmasstransfer.2016.07.102
[10] Mahmood R, Kousar N, Yaqub M, Jabeen K. Numerical simulations of the square lid-driven cavity flow of Bingham fluids using nonconforming finite elements coupled with a direct solver. Adv Math Phys. 2017;2017. https://doi.org/10.1155/2017/5210708

[11] Thohura S, Molla MM, Sarker MMA. Bingham fluid flow simulation in a lid-driven skewed cavity using the finite-volume method. Int J Comput Math. 2020;97:1212–33. https://doi.org/10.1080/00207160.2019.1613527
[12] Bui CM, Ho ANT, Nguyen XB. Flow behaviors of polymer solution in a lid-driven cavity. Polymers (Basel). 2022;14. https://doi.org/10.3390/polym14122330
[13] Ferrari MA, Franco AT. Exploring the periodic behavior of the lid-driven cavity flow filled with a Bingham fluid. J Nonnewton Fluid Mech. 2023;316. https://doi.org/10.1016/j.jnnfm.2023.105030
[14] Ferrari MA, Franco AT. The steady and unsteady regimes in a cubic lid-driven cavity with viscoplastic fluid solved with the lattice Boltzmann method. J Nonnewton Fluid Mech. 2024;325. https://doi.org/10.1016/j.jnnfm.2024.105198
[15] Le Clainche S, Ferrer E, Gibson S, Cross E, Parente A, Vinuesa R. Improving aircraft performance using machine learning: A review. Aerosp Sci Technol. 2023;138. https://doi.org/10.1016/j.ast.2023.108354
[16] Calzolari G, Liu W. Deep learning to replace, improve, or aid CFD analysis in built environment applications: A review. Build Environ. 2021;206. https://doi.org/10.1016/j.buildenv.2021.108315
[17] Hemmasian AP, Ogoke F, Akbari P, Malen J, Beuth J, Barati Farimani A. Surrogate modeling of melt pool temperature field using deep learning. Addit Manuf Lett. 2023;5. https://doi.org/10.1016/j.addlet.2023.100123
[18] Peng JZ, Liu X, Aubry N, Chen Z, Wu WT. Data-driven modeling of geometry-adaptive steady heat conduction based on convolutional neural networks. Case Stud Therm Eng. 2021;28. https://doi.org/10.1016/j.csite.2021.101651
[19] Hua Y, Yu CH, Zhao Q, Li MG, Wu WT, Wu P. Surrogate modeling of heat transfers of nanofluids in absorbent tubes with fins based on deep convolutional neural network. Int J Heat Mass Transf. 2023;202. https://doi.org/10.1016/j.ijheatmasstransfer.2022.123736
[20] Hua Y, Wang ZQ, Yuan XY, Bai Li Y, Wu WT, Aubry N. Estimation of steady-state temperature field in multichip modules using deep convolutional neural network. Therm Sci Eng Prog. 2023;40. https://doi.org/10.1016/j.tsep.2023.101755
[21] Thuerey N, Weißenow K, Prantl L, Hu X. Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows. AIAA J. 2020;58:25–36. https://doi.org/10.2514/1.J058291.
[22] Li C, Yuan P, Liu Y, Tan J, Si X, Wang S, et al. Fast flow field prediction of hydrofoils based on deep learning. Ocean Eng. 2023;281:114743. https://doi.org/10.1016/j.oceaneng.2023.114743.
[23] Huang L, Pena B, Liu Y, Anderlini E. Machine learning in sustainable ship design and operation: A review. Ocean Eng. 2022;266:112907. https://doi.org/10.1016/j.oceaneng.2022.112907.
[24] Gharleghi R, Sowmya A, Beier S. Transient wall shear stress estimation in coronary bifurcations using convolutional neural networks. Comput Methods Programs Biomed. 2022;225:107013. https://doi.org/10.1016/j.cmpb.2022.107013.
[25] Takbiri S, Kazemi M, Takbiri-Borujeni A, McIlvain J. A deep learning approach to predicting permeability of porous media. J Pet Sci Eng. 2022;211:110069. https://doi.org/10.1016/j.petrol.2021.110069.
[26] Du B, Lund PD, Wang J. Combining CFD and artificial neural network techniques to predict the thermal performance of all-glass straight evacuated tube solar collector. Energy. 2021;220:119713. https://doi.org/10.1016/j.energy.2020.119713.
[27] Yang J, Palazzolo A. Deep convolutional autoencoder augmented CFD thermal analysis of bearings with inter pad groove mixing. Int J Heat Mass Transf. 2022;188:122639. https://doi.org/10.1016/j.ijheatmasstransfer.2022.122639.

[28] Kim MH, Park HJ. Application of artificial neural networks using sequential prediction approach in indoor airflow prediction. J Build Eng. 2023;69:106319. https://doi.org/10.1016/j.jobe.2023.106319.
[29] Sang J, Pan X, Lin T, Liang W, Liu GR. A data-driven artificial neural network model for predicting wind load of buildings using GSM-CFD solver. Eur J Mech B Fluids. 2021;87:24–36. https://doi.org/10.1016/j.euromechflu.2021.01.007.
[30] Hussain SM, Mahat R, Katbar NM, Ullah I, Varun Kumar RS, Prasannakumara BC, et al. Artificial neural network modeling of mixed convection viscoelastic hybrid nanofluid across a circular cylinder with radiation effect: Case study. Case Stud Therm Eng. 2023;50:103487. https://doi.org/10.1016/j.csite.2023.103487.
[31] Muravleva E, Oseledets I, Koroteev D. Application of machine learning to viscoplastic flow modeling. Phys Fluids. 2018;30:5058127. https://doi.org/10.1063/1.5058127.
[32] Kumar A, Ridha S, Narahari M, Ilyas SU. Physics-guided deep neural network to characterize non-Newtonian fluid flow for optimal use of energy resources. Expert Syst Appl. 2021;183:115409. https://doi.org/10.1016/j.eswa.2021.115409.
[33] Zhang J. Physics-informed neural networks for Bingham fluid flow simulation coupled with an augmented Lagrange method. Appl Math. 2023;3:525–51. https://doi.org/10.3390/appliedmath3030028.
[34] ANSYS Fluent Users Guide. 2023.
[35] Neofytou P. A 3rd order upwind finite volume method for generalised Newtonian fluid flows. Adv Eng Softw. 2005;36:664–80. https://doi.org/10.1016/j.advengsoft.2005.03.011.
[36] Montgomery DC. Design and analysis of experiments. 8th ed. Hoboken: Wiley; 2013.
[37] Cunegatto EHT, Gotardo M, Zinani FSF. Numerical analysis of tube arrangements with one, two, and four degrees of freedom for heat transfer with pseudoplastic fluids. Int J Heat Mass Transf. 2023;208:124080. https://doi.org/10.1016/j.ijheatmasstransfer.2023.124080.
[38] Cunegatto EHT, Zinani FSF, Biserni C, Rocha LAO. Constructal design of passive micromixers with multiple obstacles via computational fluid dynamics. Int J Heat Mass Transf. 2023;215:124519. https://doi.org/10.1016/j.ijheatmasstransfer.2023.124519.
[39] Misiulia D, Elsayed K, Andersson AG. Geometry optimization of a deswirler for cyclone separator in terms of pressure drop using CFD and artificial neural network. Sep Purif Technol. 2017;185:10–23. https://doi.org/10.1016/j.seppur.2017.05.025.
[40] Stein M. Large sample properties of simulations using Latin hypercube sampling. Technometrics. 1987;29:143–51. https://doi.org/10.1080/00401706.1987.10488205.
[41] Minaee S, Boykov Y, Porikli F, Plaza A, Kehtarnavaz N, Terzopoulos D. Image segmentation using deep learning: A survey. 2020.
[42] Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-scale machine learning on heterogeneous distributed systems. 2015.
[43] Chollet F, et al. Keras. 2015.
[44] Géron A. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. 2nd ed. Sebastopol: O'Reilly Media; 2019.
[45] Bradski G. The OpenCV library. Dr Dobb's J Softw Tools. 2000.
[46] Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nature. 2020;585:357–62. https://doi.org/10.1038/s41586-020-2649-2.
[47] Çengel YA, Cimbala JM. Fluid mechanics: Fundamentals and applications. 3rd ed. New York: McGraw-Hill; 2014.
[48] Zamankhan P, Takayama S, Grotberg JB. Steady displacement of long gas bubbles in channels and tubes filled by a Bingham fluid. Phys Rev Fluids. 2018;3:013302. https://doi.org/10.1103/PhysRevFluids.3.013302.