Modelling of New RANS Viscous Model Equations with the Use of Model Discovery Libraries and Machine Learning

Daniel James Rogers, Syed Muhammad Gillani, Adam Sohail, Gabriel Maire, Yasser El Karkouri, Hyejoon Lee, Venceslas Bignon

Group 4: OutOfBounds.Applications in Practical High-End Computing. MSc in Computational and Software Techniques in Engineering. Cranfield University. 2024

This paper presents a novel framework integrating Sparse Identification of Non-Linear Systems (SINDy) and Physics Informed Neural Networks (PINN) to predict Reynolds stresses in Poiseuille and Couette unidirectional turbulent flows. Using SINDy, the governing equations from Direct Numerical Simulation (DNS) data are derived and constrain the PINN model for accuracy. Training on DNS datasets across Reynolds Numbers (Re) of 182 to 5200, the model achieves significant improvement over traditional RANS SST-Kw predictions, with an RMSE reduction of 81.29% for unseen Channel data and 90.3% for unseen Couette data. Integrated with an automated Reynolds-Averaged Navier-Stokes (RANS) script, the model demonstrates enhanced predictive capability for unseen Reynolds Cases, achieving an RMSE improvement of 63.9% for the normal Reynolds Stresses for Re 1001 Channel Flow and 54.6% for the Re 550 Couette Flow. This research bridges data-driven techniques with traditional modelling, creating avenues for turbulence prediction in practical engineering applications.

Keywords - Machine Learning, Physics Informed Neural-Networks, Sparse Identification of Non-Linear Systems, Turbulence Modelling

Introduction

With the recent interest and research in the implementation of data-driven techniques to solve engineering problems, paired with the development of Graphical Processing Unit (GPU) computational power, new opportunities have emerged to substitute mathematical-based derivations with more innovative approaches. In the field of Computational Fluid Dynamics (CFD), the prediction of turbulent modelling for the Reynolds Average Navier Stokes (RANS) has been a major challenge that relies on accurately predicting the Reynolds Stresses (τ_{ij}) generated when implementing the Reynolds Decomposition ($u = \overline{u} + u'$) into the Navier Stokes Equations. Standard approaches to this problem rely on the application of the Boussinesq Hypothesis [Eq. (1)] which assumes a linear relation between the velocity gradients and Reynolds Stresses and applies an over-corrective term for the normal stresses. The equation consists of two terms: the turbulent Viscous Stress Term (S_{ij}), which takes into account the stress generated by the turbulent fluctuations and the Turbulent Pressure Term ($k\delta_{ij}$) which takes into account the stress generated by the isotropic assumption. This equation relies on an assumption of local turbulent isotropy, as well as the shear stresses are aligned with the normal stresses.

$$\tau_{ij}^R = -2\mu_T S_{ij} + \frac{2}{3}k\delta_{ij} \quad \textbf{(1)}$$

The Boussinesq Hypothesis relies on two additional terms, that need to be modelled with the implementation of additional equations.

This report aims to implement a new framework to predict the Reynolds Stresses for a unidirectional pressure-driven flow (Pousille or Channel) and a unidirectional shear-driven flow (Couette) travelling through infinitely long parallel plates without relying on the Boussinesq Hypothesis.

Literature Review

The requirements of adding new equations are derived on developing mathematical and empirical approaches to close this set of equations. This problem was named the RANS closure problem and extensive research has been done in this field highlighting the discovery of two-equation models such as k-epsilon (1), k-w (2) or Spalart-Almaras (3) and the SST-kw (4), being the latter one the most used in the industry. Efforts in the search of more complex models ended in the development of the Reynolds Stress Model is a seven-equation that computes individually each of the 6 individual Reynolds Stresses, not relying on this hypothesis.

One of the new approaches used in this project relied on Sparse identification of nonlinear dynamics (SINDy). SINDy is a groundbreaking method introduced by Brunton et al. (2016) (5). It employs sparse regression to unveil governing equations from complex data sets. The core idea is that the dynamics of numerous physical systems can be represented by sparse combinations of basis functions, simplifying models without losing critical dynamics. Applications of SINDy in practical areas of engineering showcase the versatility of this tool, such as Kaptanoglu et al.(2022) (8),Saeed Kandezy, et al (2024) (6) or Silva et al. (2020) (7). Applications in the field of turbulence modelling have also been developed, such as Rubini's ((9).

Machine learning techniques have shown significant potential for tackling the challenges of turbulent flow modeling and simulation. Physics-Informed Neural Networks (PINNs) incorporate governing equations like the Navier-Stokes equations into neural network architectures, enabling accurate

and physically consistent predictions (10) (11) (12). Deep learning models such as Residual Networks (ResNets) and Deep Operator Networks (DeepONets) have been applied for temporal and spatial predictions of turbulent flows (13) (14). Unsupervised, semi-supervised, and supervised learning methods have been utilised to analyse turbulence data from direct numerical simulations of canonical flows like isotropic turbulence, Rayleigh-Bénard convection, and turbulent channel flows (15) (16). Machine learning has also been integrated with traditional turbulence modeling approaches, such as using random forests and neural networks for the Reynolds stress tensor (17) (18). Additionally, reinforcement learning techniques have been employed for flow control problems (19) (20). While challenges remain in developing scalable and interpretable models (23), the integration of machine learning with turbulent flow modeling holds promise for advancing predictive capabilities, flow control, optimisation, and discovering new physical insights through data-driven approaches (15) (24).

The Data Set

All training of the models has been performed by using the DNS dataset published by Myoungkyu Lee and Robert D. Moser (26) (25) available at https://turbulence.oden.utexas.edu/. The dataset contained five Channel Flow cases ranging various turbulence ranges, characterised by Reynolds Numbers 182, 501, 1001, 1996, and 5186, along with three Couette Flow Cases featuring Reynolds numbers 93, 221, and 543. Couette data was presented for a 20PI and 100Pi streamwise length domain. Although results did not differ, both sets of datasets have been implemented. The variables of interest used throughout this project were:

- Flow Properties: kinematic viscosity, friction velocity and Friction Reynolds Number [ν, u_τ and Re_τ]
- Mean Velocity Field in the streamwise and spanwise [U and W]
- Streamwise Velocity Gradient in the normal wall distance [dU/dy]
- Reynolds Stresses Covariances [$u'u'$, $v'v'$, $w'w'$, $u'v'$, $u'w'$ and $v'w'$]
- Turbulent Kinetic Energy [$k = 0.5 * (u'u' + v'v' + w'w')$]
- Mean profile of pressure [P]

All the data was normalised, and the analysis focused on the lower half of the channel, assuming implicit symmetry. This assumption, while generally true for Channel flow, may not hold for Couette flow, prompting the adoption of a mirroring approach to increase data diversity. This mirroring of data around $y = 1$ serves as a vertical symmetry axis, allowing for data augmentation and improved training of PySINDy and Neural Network models. The mirroring approach was a trade-off between accuracy and data diversity, chosen after initial iterations with only half-channel data. It also served as an additional layer of data analysis, allowing for comparison of DNS and RANS data and assessment of the effectiveness of interpolation and mirroring methods. Adjustments were made accordingly, including additional horizontal mirroring and clarifications of normalisation methods.

Data Preprocessing and Interpolation

Discussions were initiated on both RANS and DNS data, with their respective characteristics and differences being highlighted. Challenges that needed to be addressed were revealed when comparing both datasets, specifically issues with data completeness, accuracy, and domain coverage.
To address these problems, a preprocessing step was implemented that involved mirroring the data. This step was necessary because only half the domain of the simulation was provided by the datasets. Additionally, the cubic spline method was utilised for interpolation to ensure the most accurate representation of the data. This interpolation method was chosen to minimise errors and discrepancies between the ground truth, represented by the DNS data, and the models developed using the preprocessed datasets. The interpolated data is seen in figure 1.

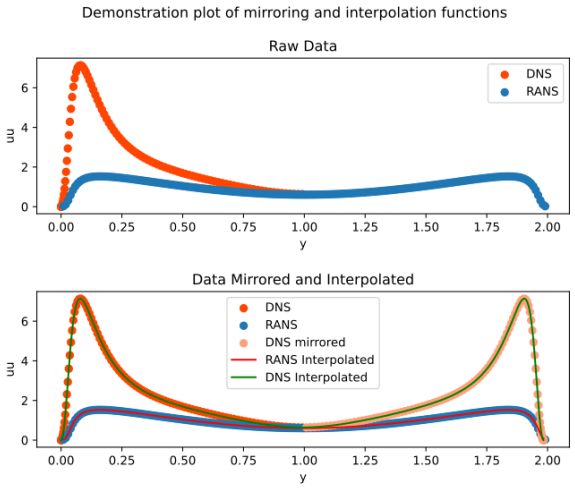

Fig. 1. Raw vs. Interpolated DNS and RANS Data

RANS Results

Test Cases RANS Modelling

To obtain the RANS predictions, the test cases were modelled as 0.05x2x0.05 block with subdivisions along the y-direction. The mesh was generated based on 3 parameters: the minimum wall distance (y_{min}), obtained through Eq. (2) and targeted y^+ of 1; the growth rate (gr), set to 1.1, and the number of cells (n), obtained with Eq. (3).

$$y = \frac{y^+ \cdot \mu}{\rho \cdot u_{tau}} = \frac{y^+ \cdot \nu}{u_{tau}} \tag{2}$$

$$n = 2 * \left[\left[\frac{log_{10}(\frac{\frac{L}{2} * gr - 1}{y} + 1)}{log_{10}(gr)} \right] - 1 \right] \tag{3}$$

For the domain boundary conditions, the top and bottom surfaces (XZ plane) were assigned a no-slip wall boundary condition, while the inlet and outlet (YX planes) and left and right (YZ) planes were assigned a periodic boundary condition. Although the mesh generation and domain boundary conditions remain the same for both Channel and Couette, the fluid boundary conditions differ from one another. On the one hand, the channel flow is characterised by a pressure gradient inputted as an x momentum source, computed with Eq. (4); whereas on the other hand, the Couette flow motion was implemented by assigning a top wall velocity of 2 m/s and static wall condition at the bottom wall.

$$\frac{\Delta P}{Lx} = \frac{2 \cdot \rho \cdot u_{tau}^2}{Ly} \tag{4}$$

Viscous Model Selection

An initial study comparing the most used viscous closure models available within the Fluent Commercial Environment was performed to establish the most accurate RANS prediction to serve as a comparison with the new results obtained by the developed models. For this project the standard K-omega (k-w) model, realisable k-epsilon (rel k-e), shear stress transport (SST) k-omega, Spalart-Almaras (SA) and Reynolds Stress Model (RSM) were compared when predicting the Channel flow with Reynolds Number of 182 and 5182. Some models such as the k-e and RSM required the implementation of wall functions and wall treatment enhancement, respectively, and they were initialised from previously converged SST k-w simulations. The convergence of the weighted average of the velocity for all models was evaluated to be within 10% of the predicted average of 1 Both the Spalarta Almaras for Reynolds 182 and RTM model for Reynolds 5200 did not fail this condition as the simulations diverged. show the convergence rate of all tested model. From all the simulations, that met the criteria and showcased in Figure 3, the velocity fields, velocity gradients, turbulent viscosity and turbulent kinetic energy were exported, and with the use of the Bousinnesq Hypothesis (Equation Eq. (1)), the Reynolds Stresses were computed. All results have been normalised to allow direct comparison with the DNS dataset; specifically, the velocity field was normalised by u_{tau}; the Reynold Stresses and turbulent kinetic energy by u_{tau}^2 and the wall distance by the inverse of the Reynolds Number.

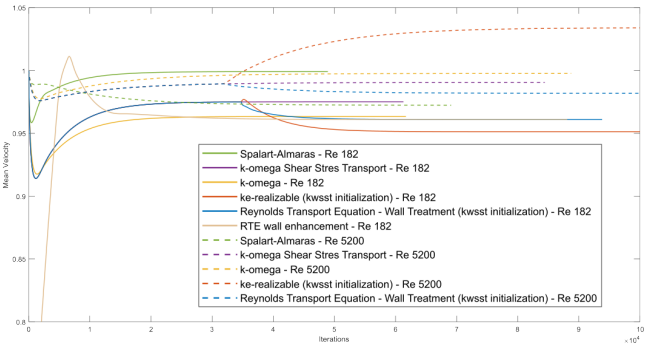

Fig. 2. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

Viscous Models Results

As all cases with independent Reynolds Numbers are being plotted individually, the results were plotted against the logarithm of the y^+. The mean streamwise velocity profiles showcase consistent behaviours for both Reynolds Cases, as displayed in Figure 3. By using the log-of-the-wall model, the velocity profile can be divided into three distinct regions: the viscous sublayer ($1 < y^+ < 30$), where $U^+ \approx y^+$; the buffer layer ($30 > y^+ > 120$), which serves as a transition region diverge; and the log-law region ($y^+ > 120$), approximated by $U^+ = \frac{1}{\kappa} \ln(y^+) + \beta^+$. The profiles show great accuracy when predicting the viscous sublayer region, but diverge from DNS in the buffer and log law region. All the models besides the RSM for the Reynolds 5200 have underestimated the velocity profile. Furthermore, the velocity field for the 5200 Reynolds Number has been predicted more accurately than the 182.

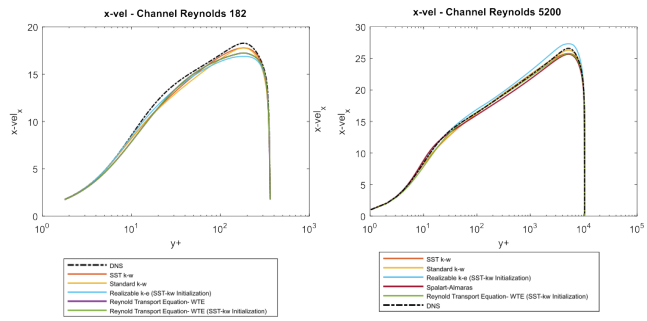

Fig. 3. Mean Velocity Profile Comparison from different RANS Viscous Models for Reynolds 182 and Reynolds 5200 Channel Flow

On the other hand, the prediction of the stresses has been displayed in Figures 6 and 5, the values of $v'w'$ and $u'w'$ have not been included as they can be assumed to be 0. Firstly, we can see a great difference between the prediction of the shear stress and the normal stresses. On the one hand, all the models besides the RSM, have provided accurate results in the viscous and log-law regions, having deviations in the buffer region. The RSM had oscillation across the entire channel width which could be attributed to the mesh resolution. On the other hand, the Normal Reynold Stresses ($u'u'$, $v'v'$ and $w'w'$) have been poorly predicted by RANS

compared to the ones computed by DNS. These stresses have been computed with equation Eq. (1) that assumed local isotropy. In these cases, the normal velocity gradients dU/dx, dV/dy and dW/dz were negligible compared to dU/dy, therefore they could be approximated to be 0, so all the contributions to the prediction come from the Turbulent Pressure Term, making the curves for the 3 normal stresses equal, behaviour that is not present in the DNS results. Finally, as the Spalart-Almaras does not have a transport equation for the turbulent kinetic energy, both the Viscous Stress Term and Turbulent Pressure Term were 0, therefore only predictions could be obtained for $u'v'$.

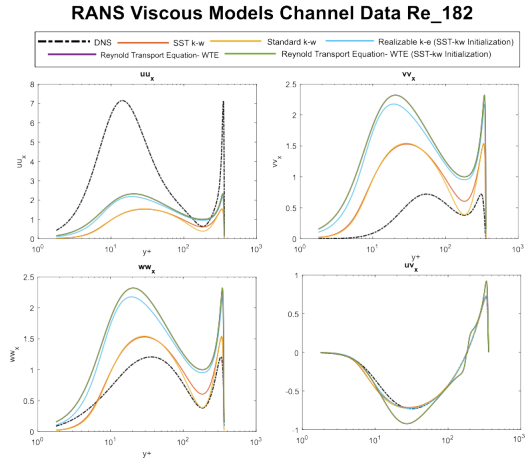

Fig. 4. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

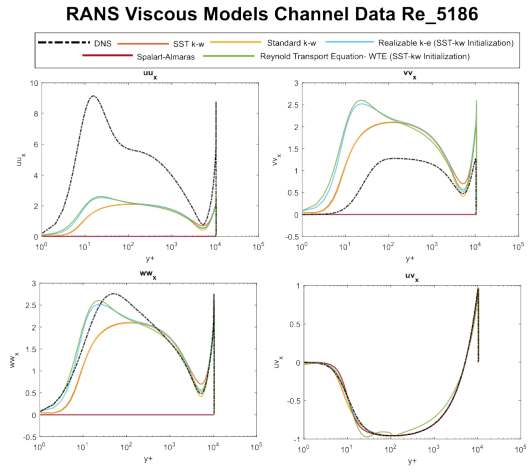

Fig. 5. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

The Root Mean Square Error (RMSE) between the RANS and DNS results have been performed and displayed in tables 1 and 2. From these results, we can observe that all the models shared similar values of errors across all variables as well as it supported the claims regarding the mismatch between the normal stresses and the accuracy in the $u'v'$ component.When comparing models it could be seen that the realisable k-e and RTM gave higher estimations of all the components of the normal stresses than both k-w models, therefore it achieved better results when predicting the normal stress in the X component and worst results for the Y and Z components. By comparing errors obtained for the turbulent kinetic energy, both the k-e and RT gave more inaccurate results than both k-w models. Finally, between the Standard and SST models, which had similar errors by only differences in the velocity and $w'w'$ prediction, it was decided to select the SST kw.

Table 1. Root Mean Square Error for Reynolds 182 Channel Flow. Models Initialised with the SST k-w models have been marked with a *

	RTM wte	realisable k-e*	standard k-w	SST k-w	RTM WTE*
U	0.681	0.860	0.624	0.449	0.681
K	0.541	0.516	0.721	0.725	0.541
uu	1.693	1.760	2.087	2.076	1.693
vv	1.022	0.940	0.547	0.575	1.022
ww	0.689	0.608	0.192	0.239	0.689
uv	0.095	0.016	0.024	0.022	0.095

Table 2. Root Mean Square Error for Reynolds 5200 Channel Flow. Models Initialised with the SST k-w models have been marked with a *

	SA	realisable k-e*	standard k-w	SST k-w	RTM WTE*
U	0.683	0.839	0.162	0.415	0.508
K	2.514	0.589	0.678	0.639	0.662
uu	2.927	1.653	1.709	1.674	1.702
vv	0.867	0.459	0.410	0.450	0.411
ww	1.264	0.103	0.116	0.172	0.091
uv	0.003	0.003	0.006	0.007	0.060

RANS Results

The three remaining Channel cases and three Couette Flows were launched with the Boundary Conditions in table 3.

Table 3. Boundary Conditions and Fluid Properties for all Channel and Couette Flows

Name	Reynolds Number	Kinematic Viscosity	Friction Velocity	Pressure Gradient	Top Wall Velocity	Bottom Wall Velocity
Channel 180	182.08	3.5E-4	0.0637	0.00406	0.0	0.0
Channel 550	543.49	1.0E-4	0.0543	0.00295	0.0	0.0
Channel 1000	1000.51	5.0E-5	0.0500	0.00250	0.0	0.0
Channel 2000	1994.75	2.3E-5	0.0459	0.00210	0.0	0.0
Channel 5200	5185.89	1.0E-8	0.0415	0.00172	0.0	0.0
Couette 93	92.91	6.7E-4	0.0619	0.0	2.0	0.0
Couette 220	219.47	2.5E-4	0.0549	0.0	2.0	0.0
Couette 500	501.37	1.0E-4	0.0501	0.0	2.0	0.0

The following results were obtained for the simulations following the same methodology and converged using the same methodology consisting of a first initialisation with a first-order upwind for the first 1000 iterations and then switched to a second-order scheme and converged to a value of $1E-8$ in all monitor values. The results obtained for all channel flows showcase the same behaviour as explained in the previous section, where the $u'v'$ has been predicted almost perfectly, however, there is a great mismatch between the predicted and actual normal stresses. On the other hand, the prediction for the Normal Reynolds Stresses from the Couette flow has displayed the same behaviour as the Channel Flow, however, it failed to accurately predict the $u'v'$, unlike the Channel data.

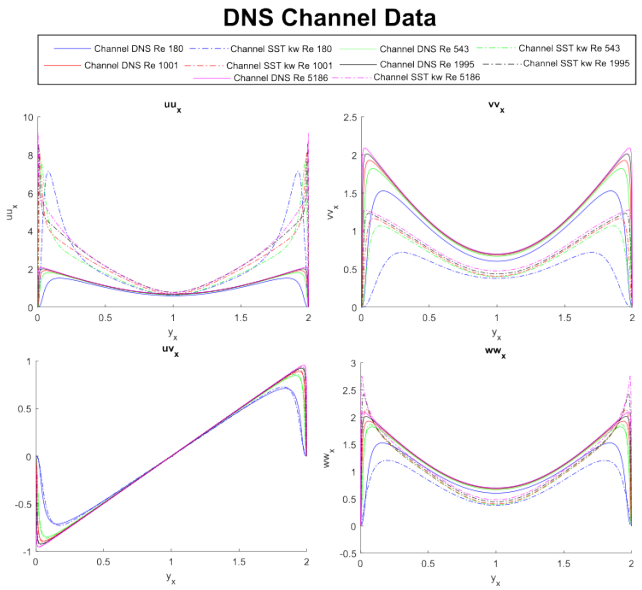

Fig. 6. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

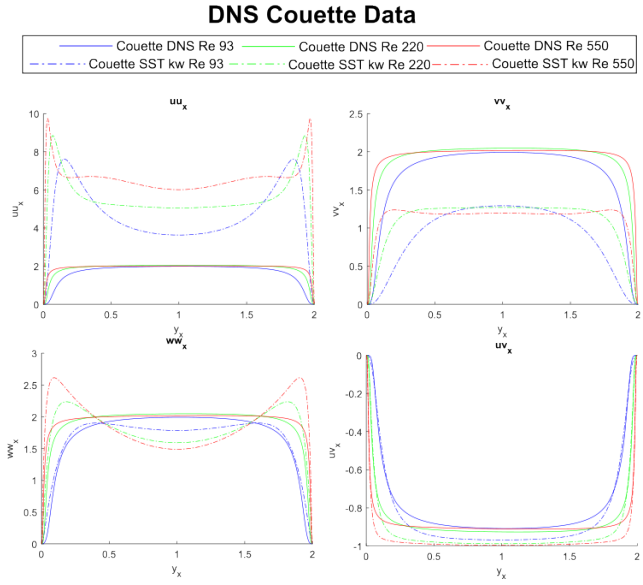

Fig. 7. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

After computing the RSME for the SST kw prediction against the DNS, they were compared against and plotted in Figure 8. From this plot, three main tendencies could be extrapolated. Firstly, It could be seen that the accuracy in the predictions increases with the Reynolds numbers, indicating that the RANS was able to predict more accurately turbulent flows, although it could also be influenced by the mesh size. On the other hand, as the Reynolds number increased for the Couette, the error increased, having the opposite behaviour from the Channel flow. Finally, for both cases, the minimum error was obtained for the $u'v'$, highlighting the differences between RANS and DNS due to the local isotropy turbulence assumption.

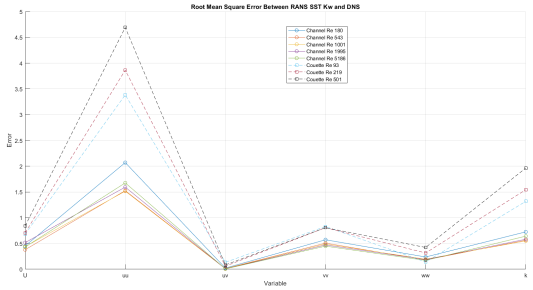

Fig. 8. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

SINDy Approach

The multi-step process used to derive a new set of governing equations for the turbulence models based on the DNS dataset involved using Sparse Identification of Nonlinear Dynamics (SINDy) models, to obtain a more accurate set of equations capable of constraining the PINN model, aiming for an improvement in the performance of an alternative PINN model based on RANS governing equations, which were particularised for unidirectional flows by crossing all zero terms. The x and y momentum equations are expressed in Eq. (5) and Eq. (6) respectively:

$$momentum_x = -\frac{dP}{dx} - \rho\frac{duv}{dy} + \nu\frac{d^2U}{dy^2} = 0 \qquad \textbf{(5)}$$

$$momentum_y = -\rho\frac{dv'v'}{dy} - \frac{dP}{dy} = 0 \qquad \textbf{(6)}$$

The training dataset for the SINDy model consisted of three channels (Reynolds 180, 1000 and 5200) cases and one Couette flow (Reynolds 220 [100 Pi]) test case. The testing dataset included two channel cases (Reynolds 550 and 2000) and two Couette cases (Reynolds 93 [20Pi and 100 Pi], 500 [20Pi and 100 Pi] and 220 [20Pi]). Both datasets used available DNS data to ensure accurate and consistent results. For feature selection, key features and parameters were manually selected to ensure accurate derivation of governing equations. A triple nested loop was used to test different combinations of parameters, including the polynomial degree, alpha, and maximum iterations necessary to run Forward Regression Orthogonal Least-Squares (FROLS), selected as it was designed to discern and retain only the most critical features, therefore ensuring that the model is both efficient and powerful, embodying a balance between simplicity and predictive accuracy.

The iterative process involved multiple training rounds and analysis to identify equations that closely matched the DNS results. This process also enabled the detection of potential issues with the RANS simulations and the refinement of the methods to improve accuracy and robustness.

Results

X Momentum Equation

Using PySINDy with FROLS optimisers, we derived two governing equations that represent the x momentum in turbulent flow scenarios. The first, simpler in structure, Eq. (7):

$$0.008\frac{dP}{dx} + 1.007\frac{duv}{dy} - \frac{d^2U}{dy^2} = 0 \quad \textbf{(7)}$$

The second, more complex one exhibiting higher-order terms, Eq. (8):

$$1.053\frac{duv}{dy} - 267.289\left(\frac{dP}{dx}\right)^2\frac{duv}{dy} - 1898.639\left(\frac{dP}{dx}\right)^3\frac{duv}{dy} + 161291.025\left(\frac{dP}{dx}\right)^3\left(\frac{duv}{dy}\right)^2 - \frac{d^2U}{dy^2} = 0 \quad \textbf{(8)}$$

Both equations demonstrated good performance in predicting x momentum, with r-squared values indicating strong correlations with the testing dataset. Table 4 provides an overview of the uncovered equations sorted by descending r-squared values.

Table 4. X Momentum PySINDy Equation Results

Equation	R2 Train	RMSE Train	R2 Test	RMSE Test
$\frac{d^2U}{dy^2} = 0.008\frac{dP}{dx} + 1.007\frac{duv}{dy}$	0.951356904	0.001599296	0.99655843	0.000661763
$\frac{d^2U}{dy^2} = 1.053\frac{duv}{dy} - 267.289\frac{dP}{dx}^2 \ldots$ $\frac{duv}{dy} - 1898.639\frac{dP}{dx}^3\frac{duv}{dy} + 161291.025\frac{dP}{dx}^3\frac{duv}{dy}^2$	0.983182652	0.000940366	0.995996848	0.000713715
$\frac{d^2U}{dy^2} = 1.057\frac{duv}{dy} - 272.974\frac{dP}{dx}^2\frac{duv}{dy} \ldots$ $-1779.846\frac{dP}{dx}^3\frac{duv}{dy} + 178958.541\frac{dP}{dx}^3\frac{duv}{dy}^2$	0.981260571	0.00099265	0.995668152	0.000742439
$\frac{d^2U}{dy^2} = 1.028\frac{duv}{dy} - 276.623\frac{dP}{dx}^2\frac{duv}{dy} \ldots$ $-1982.288\frac{dP}{dx}^3\frac{duv}{dy} - 8057.727\frac{dP}{dx}^2\frac{duv}{dy}^2$	0.981739621	0.00097988	0.995440652	0.000761685
$\frac{d^2U}{dy^2} = 1.028\frac{duv}{dy} - 103.557\frac{dP}{dx}^2\frac{duv}{dy} \ldots$ $+146597.095\frac{dP}{dx}^3\frac{duv}{dy}^2$	0.975593849	0.001132838	0.995413536	0.000763947

Comparison of Complexity and Fit

The simpler Eq. (7) is of order 1, while Eq. (8) is of order 5, indicating increased complexity. Despite this, both equations Eq. (7) and Eq. (8) yielded similar r-squared results for the testing dataset. This suggests that the pySINDy approach is adaptable, capable of uncovering various governing equations with differing complexities.

While Eq. (7) performed well in the testing dataset, it showed a lower r-squared value in the training dataset, suggesting that this equation may be more suited to Couette flow data. Eq. (7), although vastly more complex, provided a higher r-squared value in the training dataset, indicating better suitability for Channel flow data. This divergence could be attributed to the composition of the training dataset and the underlying characteristics of the different flow types.

Figure 10, represents Eq. (8) averaged for all cases, and illustrates the observed deviations and their impact. Notably, the pySINDy-derived x momentum equations exhibited oscillatory behaviour in the middle regions, contrasting with the linear patterns in DNS data. The discrepancies could be linked to varying Reynolds numbers and Couette versus Channel flow characteristics.

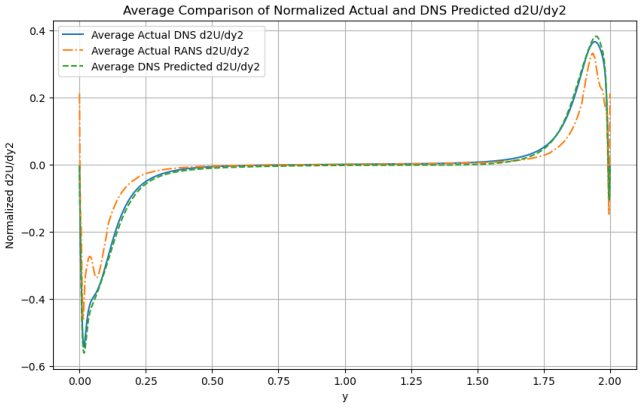

Fig. 9. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

Y Momentum Equation

For the y momentum equation, we derived the following governing equation, which aligns well with the DNS data:

$$-\frac{dv'v'}{dy} - \frac{dP}{dy} = 0 \quad \textbf{(9)}$$

Table 5 provides a list of equations sorted by r-squared values. This equation mirrors the RANS model's derived y momentum equation, indicating a solid match with the DNS data.

Table 5. Y Momentum PySINDy Equation Results

Equation	R2 Train	RMSE Train	R2 Test	RMSE Test
$\frac{d(v'v')}{dy} = -1.000\frac{dP}{dy}$	1.0000	2.202e-06	1.0000	7.185e-07
$\frac{d(v'v')}{dy} = -2451.492\frac{dP}{dy}^3 + 0.294\frac{dP}{dy}^3\mathrm{Re}_\tau$ $+1257020.483\frac{dP}{dy}^5$	0.9224	0.001616	0.9386	0.002454
$\frac{d(v'v')}{dy} = -2665.127\frac{dP}{dy}^3 + 0.131\frac{dP}{dy}^3\mathrm{Re}_\tau$ $+1442007.391\frac{dP}{dy}^5$	0.9113	0.001727	0.9293	0.002633
$\frac{d(v'v')}{dy} = -972.561\frac{dP}{dy}^3 + 0.339\frac{dP}{dy}^3\mathrm{Re}_\tau$	0.8428	0.002299	0.8561	0.003757
$\frac{d(v'v')}{dy} = -0.001\frac{dP}{dy}\mathrm{Re}_\tau + -670.648\frac{dP}{dy}^3$	0.8406	0.002315	0.8541	0.003782

Figure 9 visually demonstrates the high correlation between our derived equations and the DNS data, further validating the accuracy of our findings.

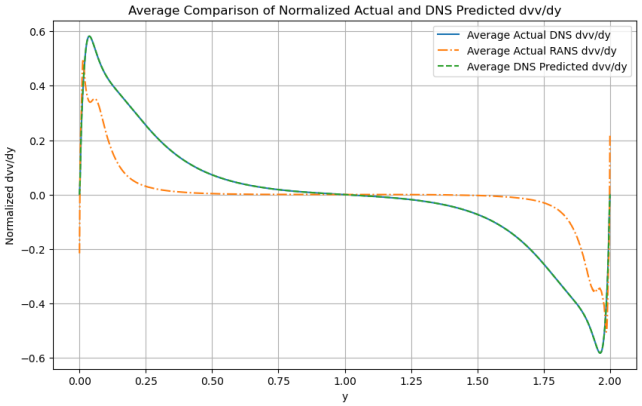

Fig. 10. Mean velocity convergence for all tested RANS Viscous Models for Channel Flow Re 182 and Re 1582.

Discussion

The results of this study highlight the importance of selecting the appropriate optimiser and features when deriving governing equations. The use of PySINDy with FROLS optimisers was effective in identifying various governing equations, ranging from simpler to more complex, providing flexibility in modelling different flow types.

The current RANS governing equations Eq. (5) and Eq. (6) guided the feature selection, which played a crucial role in the derivation of accurate equations for both x and y streamwise momentum. The proper selection of features ensured that the models captured key turbulence characteristics, even with limited training data.

Despite the limited dataset, the derived equations effectively captured the turbulent flow dynamics, indicating that PySINDy can produce robust results with the right combination of optimisers and features. The pre-processing and data smoothing further contributed to the accuracy of the derived equations, emphasising the importance of clean and well-prepared data.

Given these findings, future research should focus on exploring different optimisers and expanding feature selection to improve the robustness and generalisability of derived equations. By understanding the impact of various optimisers and feature sets, more adaptable models can be developed, capable of handling a broader range of turbulent flow scenarios. Ultimately, this approach will lead to more reliable and accurate turbulence modelling in complex systems.

The PINN Model

PINN Model Architecture

The opted architecture is a residual neural network. This architecture allows complex relationships to be captured within the data by using residual connections or skip connections as illustrated in figure 11. This facilitates the flow of information through the network and helps mitigate the vanishing gradient problem commonly encountered in deep neural networks.

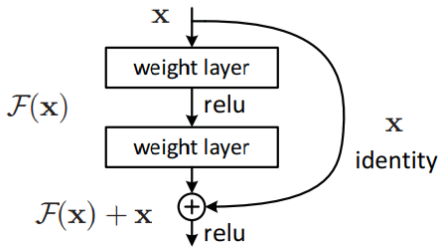

Fig. 11. A Residual Connection

After defining the architecture of the custom neural network as well as selecting the right hyperparameters, the last step involves coupling each individual custom neural network is coupled with the new governing equations obtained from SINDy through the Loss function as illustrated in equations Eq. (10) and Eq. (11) for Channel flow and Eq. (12) and Eq. (13) for Couette Flow:

$$momentum_x = -\frac{d^2U}{dy^2} + 1.057 \cdot \frac{d_u v}{dy} - 272.974 \cdot \left(\frac{dP}{dx}\right)^2 \cdot \frac{d_u v}{dy} - 1779.846 \cdot \left(\frac{dP}{dx}\right)^3 \cdot \left(\frac{d_u v}{dy}\right)^2 \quad \textbf{(10)}$$

$$momentum_y = -\frac{1}{\rho}\frac{\partial P}{\partial y} - \frac{\partial \langle u_y' \rangle^2}{\partial y} \quad \textbf{(11)}$$

$$momentum_x = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu_t \frac{\partial^2 U_x}{\partial y^2} - \frac{\partial \langle u_x' u_y' \rangle}{\partial y} \quad \textbf{(12)}$$

$$momentum_y = -\frac{\partial^2 U_x}{\partial y^2} + \frac{\partial \langle u_x' u_y' \rangle}{\partial y} - 0.001 \quad \textbf{(13)}$$

It is through the use of the loss function that physics patterns and laws can be introduce to the model. In this scenario, the loss function is seen in Eq. (14):

$$\text{MSE}(\Theta) = \text{MSE}_{\text{GT}}(\Theta) + \text{MSE}_{\text{BC}}(\Theta) + \text{MSE}_{\text{PDE}}(\Theta) \quad \textbf{(14)}$$

The model's global loss function is composed of the addition of three individual losses. The first loss($\text{MSE}_{\text{GT}}(\Theta)$)corresponds to the basic mean square error (MSE) between the predicted Reynolds Stress Tensor and the Ground Truth Reynolds Stress Tensor, Eq. (15):

$$Loss_{GT} = RST_{NN} - RST_{GT} \quad \textbf{(15)}$$

The second loss ($\text{MSE}_{\text{BC}}(\Theta)$), Eq. (16) corresponds to the boundary conditions:

$$Loss_{BC} = RST(0)_{NN} - RST(0)_{GT} \quad \textbf{(16)}$$

The third ($MSE_{PDE}(\Theta)$) loss corresponds to the physical aspect as it uses the mean square error on both the momentum equation in the x-direction and y-direction which come from the RANS equation - it is seen in Eq. (17) and Eq. (18) respectively:

$$Loss_{momentum_x} = momentum_x \tag{17}$$

$$Loss_{momentum_y} = momentum_y \tag{18}$$

This integration completes the PINN model as illustrated in Figure 12.

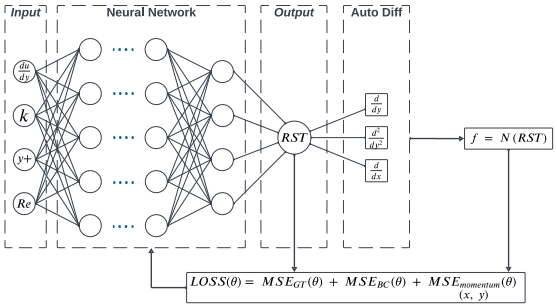

Fig. 12. The PINN mModel

Training of the Model

As it was done with the PySindy approach, the DSN cases were divided in training and testing datasets, being the former group comprised by the Channel Flows with Reynolds Numbers of 180, 500, 2000 and 5200 and Couette Flow with Reynolds Numbers 93 [20Pi and 100 Pi], 220 [20Pi and 100 Pi] and 500 [20Pi]) and the former with Channel Flow with Reynolds Number of 1000 and Couette flow 500 [100 Pi].
A total of 6 different PINN models were trained. An initial global model capable of predicting both Couette and Channel flows based on the input was created, as well as individual models for each test case. Furthermore, those three models were constrained both with RANS governing equations Eq. (5) and Eq. (6) and PySindy Equations Eq. (8) and Eq. (**??**). RANS-based equations used the original DNS dataset and PySindy were trained with the interpolated dataset generated for the equation discovery. All of the models are showcased in table 6.

Table 6. Illustration of the training setup for RANS-based PINN and PySindy-based PINN.

Version	Model	Flow Type	Rows (training)	rows (testing)
Splitted	RANS based PINN	Channel	4342	511
Splitted	RANS based PINN	Couette	891	255
Unified	RANS based PINN	Both	2811	766
Splitted	PySindy based PINN	Channel	130000	40000
Splitted	PySindy based PINN	Couette	50000	10000
Unified	PySindy based PINN	Both	120000	40000

Unified Model

The unified model was configured with the combination of the features for both RANS and PySindy equations but was not able to accurately predict either Channel and Couette flows as the model was not able to differentiate between one or another. This behaviour could be observed in both RANS based pin (Figure 13) and PySindy based PINN (14).

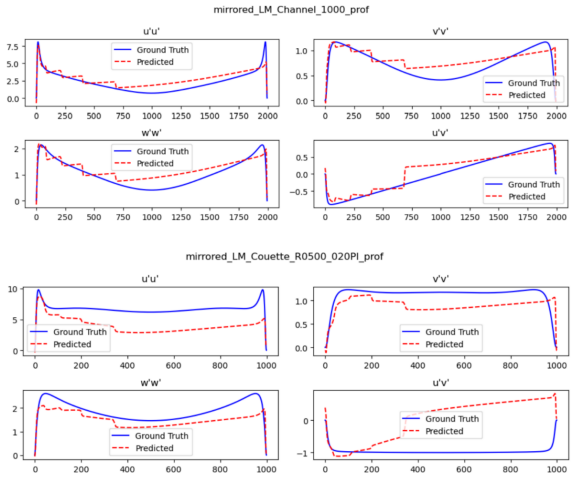

Fig. 13. Unified RANS result (20 neurons & 4 hidden layers

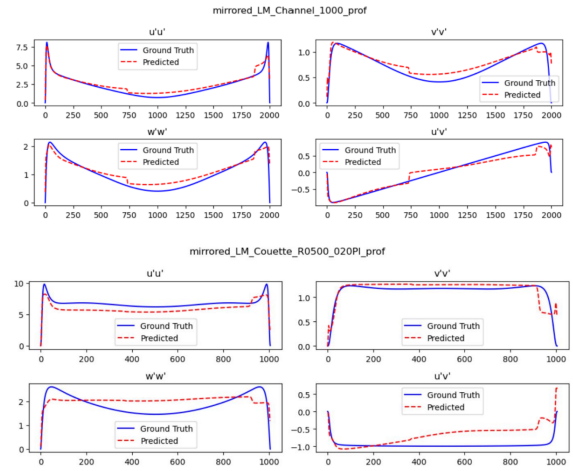

Fig. 14. Unified PySindy result (20 neurons & 4 hidden layers

Split Model

The split model was trained independently on each flow, which meant that the available data available for training the model had to be divided accordingly. The models were trained with the configuration showcased in Table 7 and were able to predict accurate results for Channel and Couette Flow.

Table 7. Model's Configuration for both RANS & PySindy based PINN.

Parameter	Value
Input	4
Hidden layer	10
Neurons	64
Output	6
activation function	Tanh
learning rate	2.5e-4
Optimiser	Adamax
epoch	2000

Qualitative approaches when comparing the new predictions showcase great improvement over the unified model. Additionally, it can be observed that both RANS-based PINN (Figure 15) and PySINDy-based PINN (Figure 16) have been able to correctly capture the intrinsic features for both flows, highlighting the prediction of the near the wall region for the Channel flow.

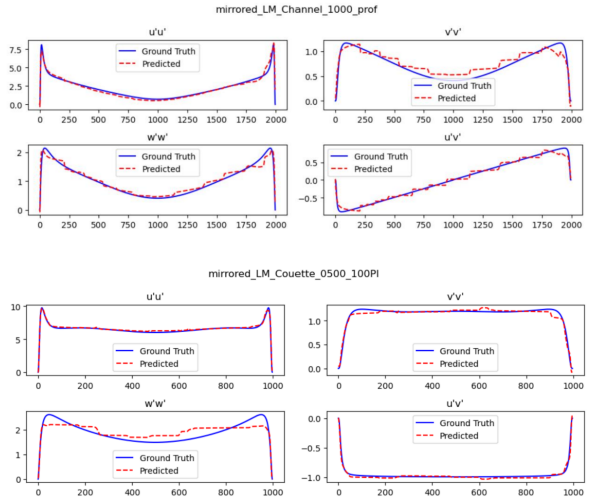

Fig. 15. Split RANS result.

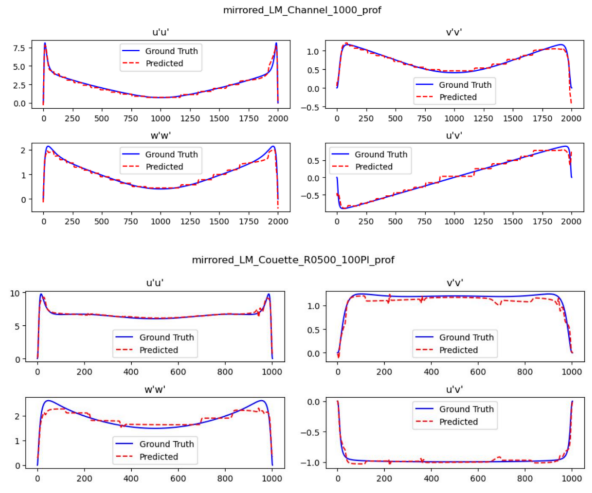

Fig. 16. Split PySindy result.

To correctly assess the quality of all split models, both the averaged Root Mean Square Error for all the predictions for the unseen test cases (both Channel and Couette) were obtained and displayed in table 8, where it can be seen that PySindy Based PINN obtained an increment in accuracy by 82% for Channel flow and 19% for Couette Flow.

Model	Flow Type	MSE
RANS	Channel	5.27e-2
RANS	Couette	1.43e-2
PySindy	Channel	9.60e-3
PySindy	Couette	1.16e-2

Table 8. Model's Results

Final Model

The final model combined the efforts of all approaches to develop software capable of predicting the Reynolds Stresses without the need for DNS inputs. By automating the execution of the CFD simulations, computation of Reynolds Stresses with the Boussinesq Hypothesis and export of both the Normalised Stresses as well as the required inputs, the model was able to enhance the current results. With the finalised model, a study consisting of predicting the stresses for all Channel and Couette cases showcased great improvements in the computations of the normal stresses. The computation of the RSME between both models compared against the DNS results, whose results are shown in Figure 19 and 20, show and increment in the accuracy in the computation of all the normal stresses however it has worsened the current prediction for the $u'v'$ component, which was predicted with great accuracy by the RANS models. When looking at the Channel Results this model has been able to overcome the business hypothesis limitations by being able to correctly predict individually each component of the normal stresses with improvements (averaged across all cases) of 72.5%, 83.7% and 32.1% for $u'u'$, $v'v'$ and $w'w'$, respectively as displayed in table XXXX.

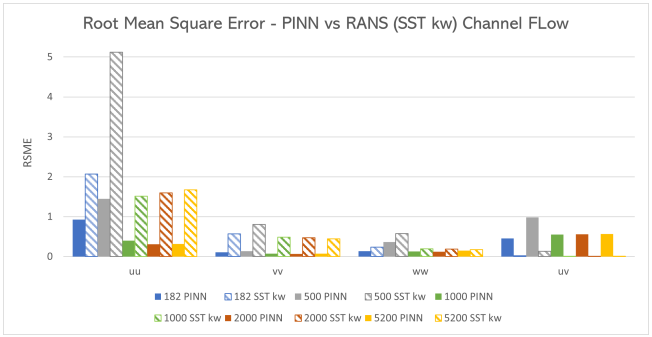

Fig. 17. The PINN mModel

Similar behaviours can be observed for the Couette flow, where results were also enhanced for the normal stresses, obtaining improvements of 69.4%, 74.2% and 18.9% for $u'u'$, $v'v'$ and $w'w'$, respectively.

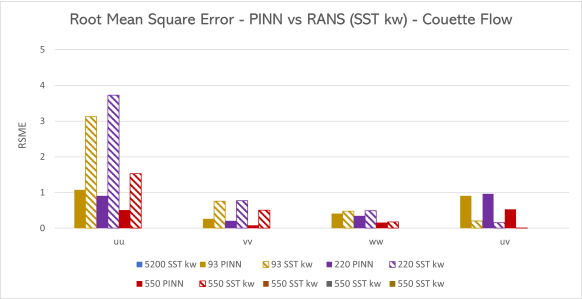

Fig. 18. The PINN mModel

Table 9. Percentage Improvements in the Root Mean Square obtained by PINN and compared to RANS

	Channel Re 182	Channel Re 500	Channel Re 1000	Channel Re 2000	Channel Re 5200	Couette Re 93	Couette Re 220	Couette Re 500
uu	55.2%	71.8%	73.5%	80.7%	81.2%	65.7%	75.6%	66.8%
vv	80.8%	83.2%	84.9%	86.3%	83.4%	65.3%	73.5%	83.9%
ww	42.7%	37.2%	33.5%	35.7%	11.6%	14.0%	29.7%	13.1%
uv	-1994.1%	-655.0%	-5620.5%	-7978.5%	-8690.0%	-345.8%	-518.8%	-4115.2%

When plotting the three predictions (DNS, RANS and PINN) for Channel 1001 (Figure 19) and Couette 500 (Figure 20), both unseen cases in the Neural Network training, great improvements in the prediction of the stresses can be observed. For the channel flow, results have shown great improvement in the prediction of all normal stress components, being able to capture the near-the-wall region behaviour characterised by a sharp gradient, which RANS was unable to capture accurately. On the other hand, the Couette flow prediction has also been improving drastically, however, discrepancies between the right and left-hand sides of the channel can be observed, indicating that the model was not able to capture symmetrical behaviour. Finally, although the results have been positive, some issues can be araised from the model, being the most critical the wall values, which do not match the boundary conditions. This can be observed clearly in the right hand side of the $v'v'$ curve for the Channel 1001 case, however this is present in all cases.

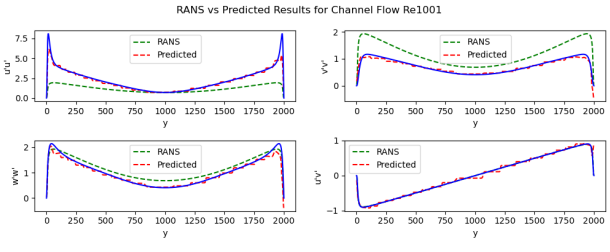

Fig. 19. PINN, RANS and DNS Comparison for Channel Flow Re 1001

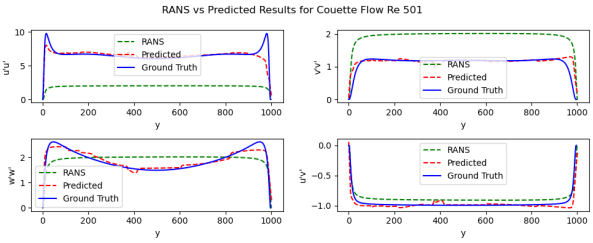

Fig. 20. PINN, RANS and DNS Comparison for Couette Flow Re 220

Finally, as a test for the model, new cases with different Reynolds number from the DNS dataset were tested, as it was suspected that the model could present overfitting due to the small amount of data available for the training. Results for Reynolds Number of 6000 and kinematic viscosity of 0.0016 were launched through the model. The results obtained, (Figure 21) show promising predictions as the near-the-wall region seems to capture the behaviour present in DNS. However, the right wall boundary condition issue can be seen in a clearer way.

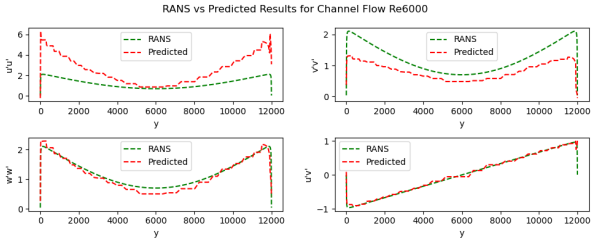

Fig. 21. PINN and RANS Comparison for Channel Flow Re 600

Conclusions

During this project, an overview regarding the strengths and weaknesses of the current state-of-the-art RANS techniques in the prediction of unidirectional flows has been stated as well as a more in depth analysis for the SST kw has been performed when comparing with the available DNS datasets. It has been stated that the current RANS results have shortcomings when predicting the normal stresses as they are still bounded by the local isotropy assumption assumed inside the Bousinnesq Hypothesis. However, the prediction of the shear stress components $u'v'$ has been accurately predicted by the SST - kw model, especially for the Channel Flow with almost perfect predictions all along the channel width.

Regarding the PySindy aproach, the new governing equations derived using PySINDy show great potential for replacing traditional RANS models in the PINN framework. Although the x momentum equation exhibits some inconsistencies for lower Reynolds numbers, the y momentum equation aligns perfectly with DNS data. The results matched the cases for two different flows: channel and couette, which have different boundary conditions, indicating the generalised applications of this equations; however, further analysis and validation will be needed to confirm these results across a broader range of flow conditions and setups.

Given the result that result was much improved when data was pre-processed than before interpolating and smoothing data, by using PySINDy, it can be considered that utilising cleaning data is playing an important role it. Furthermore, it was also highlighted that the final governing equations obtained were highly dependent on the optimiser used. Knowing and understanding of data that you will use is also crucial. Nevertheless, even though we haven't had much data to train, it has captured the turbulent model accurately.

The objectives of the Machine Learning segment were to enhance the Reynolds stress tensor using the RANS

simulation output as input, aiming to improve results within a significantly reduced timeframe compared to a DNS simulation. Initially, we opted to create a simple neural network model. Unfortunately, the results were not promising, this basic model couldn't capture the complex fluid behaviour adequately. As a result, we decided to enhance the model by incorporating the PINN model. While the results achieved with the PINN model are promising, shortcomings in terms of data and boundary conditions lead to suboptimal results, possibly even overfitting. Therefore, further work on these aspects is necessary to optimise the results, with a potential view towards commercialising the model.

Future Works

Although the model proposed was capable of improving the values of the Reynolds stresses obtained through RANS, there is still room for improvement regarding the constraining of the model. The model presented mismatches in the wall boundary condition. Furthermore, future works would involve incorporating more DNS data into the model to have a more versatile solution. Finally, future modification of this model will involve including governing equations for bi-directional and tri-directional fluids.

ACKNOWLEDGEMENTS

We would like to appreciate the efforts of Dr Jun Li, Dr Irene Moulitsas and Dr Tamás Jozsa for their support and supervision during this project.

Bibliography

1. Wilcox DC. Turbulence Modeling for CFD. DCW Industries; 1994. Wilcox, D. C. (2006), Turbulence Modeling for CFD (3rd ed.), DCW Industries, ISBN 0963605100
2. Kolmogorov, A. N. "Dissipation of Energy in the Locally Isotropic Turbulence." Proceedings: Mathematical and Physical Sciences, vol. 434, no. 1890, 1991, pp. 15–17. JSTOR, http://www.jstor.org/stable/51981. Accessed 1 May 2024.
3. P.R Spalart, Strategies for turbulence modelling and simulations, International Journal of Heat and Fluid Flow, Volume 21, Issue 3, 2000, Pages 252-263, ISSN 0142-727X,
4. Two-equation eddy-viscosity turbulence models for engineering applications.F. R. Menter. AIAA Journal 1994 32:8, 1598-1605. Available at https://arc.aiaa.org/action/showCitFormats?doi=10.2514%2F3.12149
5. S. L. Brunton, J. L. Proctor, J. N. Kutz, *Discovering governing equations from data by sparse identification of nonlinear dynamical systems*. Proceedings of the National Academy of Sciences, 113(15):3932–3937, 2016. https://browse.arxiv.org/pdf/2004.08424v1.
6. Saeed Kandezy, R.; Jiang, J.; Wu, D. On SINDy Approach to Measure-Based Detection of Nonlinear Energy Flows in Power Grids with High Penetration Inverter-Based Renewables. Energies 2024, 17, 711. https://doi.org/10.3390/en17030711
7. B. M. de Silva, K. Champion, M. Quade, J.-C. Loiseau, J. N. Kutz, S. L. Brunton, *PySINDy: A Python package for the sparse identification of nonlinear dynamical systems from data*. Journal of Open Source Software.https://arxiv.org/pdf/2111.08481.
8. A. A. Kaptanoglu, B. M. de Silva, U. Fasel, et al., *PySINDy: A comprehensive Python package for robust sparse system identification*. Journal of Open Source Software, 2022. https://www.semanticscholar.org/reader/3e89a9f1301a46d011ff04d7c05ae5134f05961a.
9. R. Rubini *Sparsification Techniques for Reduced Order Models of Turbulent Flows* https://eprints.soton.ac.uk/467602/1/Final_Thesis.pdf.
10. Raissi M, Perdikaris P, Karniadakis GE. *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*. Journal of Computational Physics. 2018;378:686-707.
11. Raissi M, Yazdani A, Karniadakis GE. *Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations*. Science. 2020;367(6481):1026-1030.
12. Mao Z, Jagtap AD, Karniadakis GE. *Physics-informed neural networks for high-speed flows*. Computer Methods in Applied Mechanics and Engineering. 2020;360:112789.
13. Guastoni L, Güemes A, Ianiro A, Discetti S, Schlatter P, Azizpour H, Vinuesa R. *Prediction of wall-bounded turbulent flows via sparse modelling*. Physical Review Fluids. 2021;6(3):034602.
14. Zhu L, Zhang W, Sun X, Liu Y, Yuan X. *Turbulence closure for high Reynolds number airfoil flows by deep neural networks*. Aerospace Science and Technology. 2021;110:106452.
15. Pandey S, Schumacher J, Sreenivasan KR. *A perspective on machine learning in turbulent flows*. International Journal of Heat and Fluid Flow. 2020;81:108544.
16. Iyer KP, Schumacher J, Yeung PK. *Fractal iso-level sets in high-Reynolds-number scalar turbulence*. Physical Review Fluids. 2020;5(4):044605.
17. Majchrzak M, Marciniak-Lukasiak K, Lukasiak P. *A Survey on the Application of Machine Learning in Turbulent Flow Simulations*. Energies. 2023;16(4):1755.
18. Fang R, Sondak D, Protopapas P, Duraisamy K. *Neural network models for the anisotropic Reynolds stress tensor in turbulent channel flow*. Journal of Turbulence. 2019;20(1):1-19.
19. Vignon A, Guastoni L, Schlatter P, Azizpour H, Vinuesa R. *Deep reinforcement learning for flow control in Rayleigh-Bénard convection*. Physical Review Fluids. 2021;6(6):064602.
20. Guastoni L, Güemes A, Ianiro A, Discetti S, Schlatter P, Azizpour H, Vinuesa R. *Perspectives on predicting and controlling turbulent flows through machine learning*. Physics of Fluids. 2021;33(6):061401.
21. Obiols-Sales O, Vishnu A, Malaya N, Chandramouli Sharan A. *CFDNet: A deep learning-based accelerator for fluid simulations*. In: Proceedings of the 34th ACM International Conference on Supercomputing. 2020:1-12.
22. Maulik R, San O, Jacob JD, Crick C. *Sub-grid scale model classification and blending through deep learning*. Journal of Fluid Mechanics. 2019;870:784-812.
23. Brunton SL, Noack BR, Koumoutsakos P. *Machine learning for fluid mechanics*. Annual Review of Fluid Mechanics. 2020;52:477-508.
24. Wang R, Kashinath K, Mustafa M, Albert A, Yu R. *Towards physics-informed deep learning for turbulent flow prediction*. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery Data Mining. 2020:1457-1466.
25. Lee M, Moser RD. Extreme-scale motions in turbulent plane Couette flows. Journal of Fluid Mechanics. 2018;842:128-145. doi:10.1017/jfm.2018.131
26. Lee, M., & Moser, R. D. (2015). Direct numerical simulation of turbulent channel flow up to Re 5200. Journal of Fluid Mechanics, 774, 395-415. doi:10.1017/jfm.2015.268