REALIZABILITY IN TENSOR BASIS NEURAL NETWORKS FOR TWO-DIMENSIONAL TURBULENT FLOWS

Anthony Man
Department of Fluids and Environment
School of Engineering
The University of Manchester
M13 9PL
anthony.man@manchester.ac.uk

Yasser Mahmoudi
Department of Fluids and Environment
School of Engineering
The University of Manchester
M13 9PL
yasser.mahmoudi@manchester.ac.uk

ABSTRACT

Tensor Basis Neural Networks (TBNNs) have become a popular machine learning model choice for improving Reynolds stress and subsequent mean flow field predictions in Reynolds-averaged Navier Stokes (RANS) approaches. However, there is still limited understanding of the target values when using TBNNs (*i.e.*, the most accurate TBNN prediction values possible), especially when a constraint to ensure non-negative eddy viscosity is enforced. This work shows that anisotropy reconstructed with the target values may not ensure realizability for all components in regions of two-dimensional flows where the constraint is applied. Hence, this indicates that the best possible TBNN prediction of anisotropy may be unable to give realizable values for all components. We show that two sets of target values can satisfy 100% accuracy and realizability in all anisotropy components if predicted by TBNNs. Therefore, it is suggested that two TBNNs may be trained – each on a set of target values to give predictions with the possibility of achieving 100% accuracy and satisfying realizability.

INTRODUCTION

Different computational fluid dynamics (CFD) approaches for simulating turbulent flows have been developed to capture various scales of turbulence, such as direct numerical simulation (DNS) and large eddy simulation (LES). However, these methods are still computationally infeasible for most industrial CFD practitioners. Reynolds-averaged Navier-Stokes (RANS) approaches are therefore still commonly used to model turbulent flows, even though they are known to predict certain flows inaccurately, including separated flows (Duraisamy et al., 2017).

To address the shortcomings in RANS, there has been growing interest in using machine learning (ML) to augment RANS models (Duraisamy et al., 2019). A particular type of model that has gained popularity is the tensor basis neural network (TBNN) proposed by Ling et al. (2016). The TBNN models Reynolds stress anisotropy using mean strain rate, mean rotation rate, and their higher order tensor products as inputs. As a result, TBNNs have been shown to model Reynolds stress and mean flow fields more accurately than typical RANS turbulence models, which usually model anisotropy as only a function of mean strain rate (Ling et al., 2016). Many modifications to the TBNN have been proposed in the literature, including ensemble TBNNs (Man et al., 2022), modular TBNNs (Man et al., 2023), the tensor basis random forest by Kaandorp and Dwight (2020), and the use of recurrent neural networks by Jiang et al. (2021).

Anisotropy predicted by a TBNN may be substituted back into the RANS equations in a process known as *a posteriori*. To mitigate ill-conditioning in the RANS equations, Wu et al. (2019) showed that the linear term in anisotropy should be treated implicitly to give effective viscosity, while the rest of the prediction should be treated as an explicit source term. To ensure a non-negative eddy viscosity, a constraint must be placed on the linear term of anisotropy.

This work shows that values of anisotropy that have been constrained may not be realizable. Anisotropy that is unrealizable may introduce normal Reynolds stresses that are negative or shear Reynolds stresses that violate Schwartz inequality in the *a posteriori* process (Durbin and Pettersson-Reif, 2011). Hence, the effect of Reynolds stress becomes a sink in the momentum equations, which results in unphysical behaviour in the flow (Pope, 2000; Durbin and Petterson-Reif, 2011). To prevent this, we show that two separate sets of predictions are required by TBNNs to satisfy accuracy and realizability of all anisotropy components in two-dimensional flow.

TENSOR BASIS NEURAL NETWORK

Tensor basis neural networks (TBNNs) are based on the general effective-viscosity hypothesis (GEVH), which shows that Reynolds stress τ_{ij} can be expressed with a finite sum of basis tensors containing mean strain rate and mean rotation rate (Pope, 1975; Ling et al., 2016). The GEVH contains four terms in two-dimensional (2D) flows:

$$\tau_{ij} = \overline{u_i' u_j'} = 2k\left(b_{ij} + \frac{1}{3}\delta_{ij}\right) \tag{1}$$

where k and δ_{ij} represent turbulent kinetic energy (TKE) and Kronecker delta, respectively. The anisotropy tensor, b_{ij} contains three of the terms:

$$b_{ij} = g_1\boldsymbol{S} + g_2(\boldsymbol{SR} - \boldsymbol{RS}) + g_3\left(\boldsymbol{S}^2 - \frac{1}{3}tr(\boldsymbol{S}^2)\delta_{ij}\right) \tag{2}$$

where $\boldsymbol{S}\,(= kS/\varepsilon)$ and $\boldsymbol{R}\,(= kR/\varepsilon)$ are the non-dimensionalized form of mean strain and mean rotation rate

tensor, respectively, and ε denotes the TKE dissipation rate. g_n $\{n = 1,2,3\}$ are scalar coefficients that must be determined to relate anisotropy b_{ij} on the left-hand side with the tensor products of $\boldsymbol{S}$ and $\boldsymbol{R}$ on the right-hand side of **Eq. (2)**. As this work focuses on the 2D GEVH, scalar coefficients g_n $\{n = 1,2,3\}$ will simply be referred to as g_n hereafter.

A TBNN can be trained to predict the g_n coefficients in a point-wise manner for any flow case. The invariants of $\boldsymbol{S}$ and $\boldsymbol{R}$ are used as inputs for predicting g_n, which are then multiplied with the $\boldsymbol{S}$ and $\boldsymbol{R}$ tensor products denoted $\boldsymbol{T}_n$ and summed in accordance with **Eq. (2)** to obtain an anisotropy b_{ij} prediction. These operations are shown schematically in **Fig. 1**. Training flow cases must have been simulated using RANS and run experimentally or computationally in a scale-resolved manner, as the $\boldsymbol{S}$ and $\boldsymbol{R}$ used in calculating the invariants and tensor products should be from RANS simulations, while target b_{ij} values deemed accurate are required to evaluate the b_{ij} prediction from the TBNN. Backpropagation is then undertaken to update the TBNN weights and biases. Thus, the aim of training a TBNN is to develop a model that takes the invariants of $\boldsymbol{S}$ and $\boldsymbol{R}$ as inputs to predict g_n coefficients that when combined with the tensor products of $\boldsymbol{S}$ and $\boldsymbol{R}$ in accordance with **Eq. (2)**, can give b_{ij} predictions on the same order of accuracy as those obtained from experiments or scale-resolved methods.

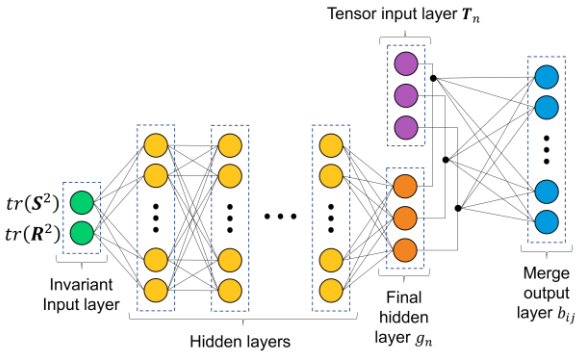


Figure 1 Two-dimensional tensor basis neural network

TARGET COEFFICIENTS

To evaluate TBNN prediction accuracy, anisotropy b_{ij} predicted by a TBNN can be readily compared with target b_{ij} from experiments or scale-resolved methods. However, it is useful to examine the target g_n that can give the target b_{ij} in **Eq. (2)**, as the mapping that a TBNN learns is the input invariants of $\boldsymbol{S}$ and $\boldsymbol{R}$ ↦ target g_n. Moreover, the predicted anisotropy b_{ij} may be substituted back into the RANS governing equations to obtain improved mean flow field predictions in a process known as *a posteriori*. Wu et al. (2019) showed that to improve conditioning of the RANS equations during this process, the linear term in predicted b_{ij} should be treated implicitly to give an effective viscosity, while the other terms should be combined and treated as an explicit source term.

To ensure a non-negative eddy viscosity, g_1 must be predicted ≤ 0 by the TBNN, and therefore target g_1 values for the purpose of TBNN training should be ≤ 0. However, it is found that target $g_1 \leq 0$ does not occur everywhere in some flow cases. Whilst the target g_1 values may be constrained so that they become ≤ 0, this has implications on the other target g_n coefficients, which can change the upper predictive performance limit (*i.e.*, most accurate possible prediction) of the TBNN. Moreover, b_{ij} given by the new upper performance limit may fall outside of the realizability bounds. This work aims to be the first investigation in the literature on these issues.

For any 2D flow case without the constraint of target $g_1 \leq 0$, **Eq. (2)** can be solved simultaneously to give the target g_n coefficients. **Eq. (2)** in its component form is as follows:

$$b_{11} = g_1 S_{11} - 2g_2 S_{12} R_{12} + \frac{1}{3} g_3 (S_{11}^2 + S_{12}^2) \quad (3a)$$

$$b_{22} = -g_1 S_{11} + 2g_2 S_{12} R_{12} + \frac{1}{3} g_3 (S_{11}^2 + S_{12}^2) \quad (3b)$$

$$b_{33} = -\frac{2}{3} g_3 (S_{11}^2 + S_{12}^2) \quad (3c)$$

$$b_{12} = g_1 S_{12} + 2g_2 S_{11} R_{12} \quad (3d)$$

where S_{ij} and R_{ij} are the components of $\boldsymbol{S}$ and $\boldsymbol{R}$, respectively in the i and j directions. Subscript 1, 2, and 3 on the b_{ij}, $\boldsymbol{S}$ and $\boldsymbol{R}$ components represent the streamwise, perpendicular, and spanwise directions, respectively. Components b_{13} and b_{23} reduce to zero. Solving **Eq. (3a-d)** simultaneously for g_n gives the following expressions (Jongen and Gatski, 1998; Man et al., 2023):

$$g_1 = \left(\frac{S_{11}}{2(S_{11}^2 + S_{12}^2)}\right)(b_{11} - b_{22}) + \left(\frac{S_{12}}{S_{11}^2 + S_{12}^2}\right) b_{12} \quad (4a)$$

$$g_2 = \frac{2S_{11} b_{12} + S_{12}(b_{22} - b_{11})}{4R_{12}(S_{11}^2 + S_{12}^2)} \quad (4b)$$

$$g_3 = \frac{3(b_{11} + b_{22})}{2(S_{11}^2 + S_{12}^2)} \quad (4c)$$

Eq. (4a-c) gives the target g_n coefficients if S and R components from RANS, and target b_{ij} from experiments or scale-resolved simulations are used on the right-hand side. These are the target g_n values for the TBNN to predict which lead to the target b_{ij} values being calculated and thus giving 100% b_{ij} prediction accuracy.

Now we examine the target g_n coefficients when the following constraint is enforced to ensure target $g_1 \leq 0$:

$$\text{target } g_{1c} = \min(0, \text{target } g_1) \quad (5)$$

where the inclusion of subscript 'c' denotes the constrained version of the target coefficient. Similar to how target g_n were determined, it is reasonable to suggest that target g_{2c} and g_{3c} may be found by substituting target g_{1c} into **Eq. (3)** and solving them simultaneously. However, it is found that one set of target g_{nc} coefficients cannot satisfy all four equations in **Eq. (3a-d)**. At locations in the flow domain where target $g_1 > 0$, **Eq. (5)** causes the first terms in **Eq. (3a)**, **(3b)** and **(3d)** to be omitted. Then by choosing two equations in **Eq. (3)** to solve for target g_{2c} and g_{3c}, it is found that target b_{ij} can only be reconstructed with 100% accuracy in two to three components. Different b_{ij} components are satisfied depending on the equations in **Eq. (3)** chosen to solve for target g_{2c} and g_{3c}. As this process determines the target g_{nc} values that would give 100% accuracy for certain b_{ij} components if predicted by a TBNN, only two to three b_{ij} components can ever be predicted with 100% accuracy by a

TBNN with the constraint and one set of g_n coefficients. The b_{ij} components that are satisfied resulting from the choice of equations in **Eq. (3)** used to solve for target g_{2c} and g_{3c} are summarized in **Table 1**, where the b_{11}, b_{22}, b_{33}, and b_{12} equations represent **Eq. (3a-d)**, respectively.

Table 1. b_{ij} equations that can be satisfied given those used to solve for target g_{2c} and target g_{3c} in Eq. (3)

Equation for solving g_{2c}	Equation for solving g_{3c}		
	b_{11} (Eq. 3a)	b_{22} (Eq. 3b)	b_{33} (Eq. 3c)
b_{11} (Eq. 3a)	-	b_{11}, b_{22}, b_{33}	b_{11}, b_{22}, b_{33}
b_{22} (Eq. 3b)	b_{11}, b_{22}, b_{33}	-	b_{11}, b_{22}, b_{33}
b_{12} (Eq. 3d)	b_{11}, b_{12}	b_{22}, b_{12}	b_{33}, b_{12}

REALIZABILITY

As the target g_{nc} coefficients cannot satisfy all b_{ij} components in **Eq. (3)** where target g_{1c} is given by **Eq. (5)**, the b_{ij} values that are reconstructed with target g_{nc} may fall outside of the realizability ranges:

$$\begin{aligned} -1/3 \leq b_{\mu\mu} \leq 2/3 \\ -1/2 \leq b_{\mu\nu} \leq 1/2 \end{aligned} \quad (6)$$

where $\mu \neq \nu$ and $\mu, \nu = \{1,2,3\}$ (Durbin and Pettersson-Reif, 2011). Unrealizable values of predicted b_{ij} substituted back into the RANS equations in the *a posteriori* process are undesirable as they can lead to unphysical flow predictions (Pope, 2000; Durbin and Pettersson-Reif, 2011). To ensure that the values of all target b_{ij} components (which are realizable) can be predicted by a TBNN, **Table 1** shows that two sets of g_{nc} coefficients can be predicted to satisfy all b_{ij} components collectively. For example, the set of coefficients obtained by solving for target g_{2c} using **Eq. (3d)** and target g_{3c} using **Eq. (3a)** may satisfy target b_{11} and b_{12}, while a second set of coefficients obtained by solving for target g_{2c} using **Eq. (3b)** and target g_{3c} using **Eq. (3a)** may satisfy target b_{22} and b_{33}. This idea is demonstrated in the following section using an example flow case.

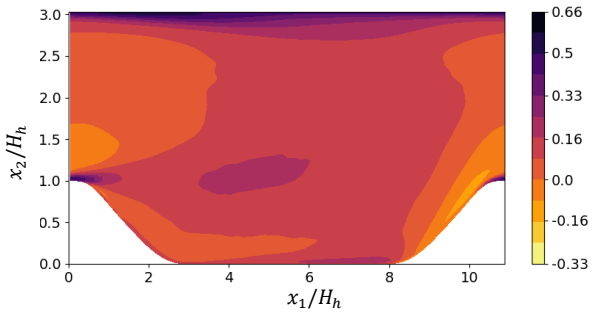
RESULTS

To show that only some target b_{ij} components can be reconstructed with 100% accuracy if only one set of target g_{nc} coefficients is predicted by TBNNs, target g_{nc} were calculated for a flow over periodic hills case at every cell centre. The Reynolds number is 5600 based on a bulk inlet velocity of 0.028 m/s and hill crest height H_h of 1m (McConkey et al., 2021). Firstly, target g_1 was calculated using **Eq. (4a)** and constrained with **Eq. (5)** to give target g_{1c}. Then, target g_{2c} and g_{3c} values were calculated by rearranging **Eq. (3d)** and **Eq. (3a)**, respectively, and the resulting anisotropy b_{ij} components were reconstructed by substituting these target g_{nc} values into **Eq. (3)**. This set of target g_{nc} values will be referred to as 'set 1', and the resulting b_{ij} fields are shown in subfigures **(a)** in **Figs. 2-5**. The same process was repeated by using target g_{2c} and g_{3c} calculated by rearranging **Eq. (3a)** and **Eq. (3b)**, respectively. This set of target g_{nc} values will be referred to as 'set 2', and the resulting b_{ij} fields are shown in subfigures **(b)** in **Figs. 2-5**.

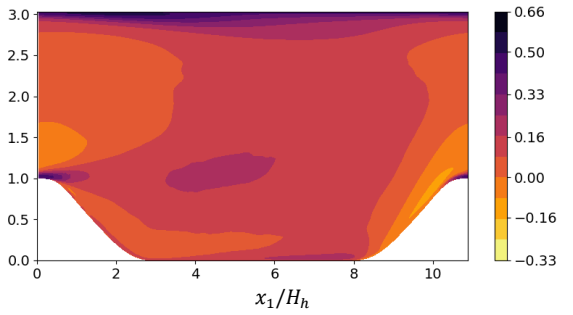
A comparison of the reconstructed b_{ij} fields from set 1 and the target b_{ij} fields from DNS in **Figs. 2-5** shows that the reconstructed b_{11} and b_{12} fields have no error in comparison to their respective target counterparts. This is because the b_{12} and b_{11} equations were used to solve for g_2 and g_3, so the values of these coefficients will have satisfied those equations. However, differences between reconstructed and target b_{22} and b_{33} which are circled in **Figs. 3** and **4** can be observed in some regions of the flow field. These coincide with where target g_1 has been constrained to equal zero. Hence, while the target g_{nc} values satisfy **Eq. (3a)** and **Eq. (3d)**, they do not satisfy **Eq. (3b)** and **Eq. (3c)**. The rest of the domain does not contain constrained coefficients, allowing all reconstructed b_{ij} components to be equal to the target b_{ij}.

The reconstructed b_{ij} fields from set 2 only shows discrepancy compared to the target b_{ij} fields in component b_{12}. This is because even though only the b_{11} equation (**Eq. 3a**) and b_{22} equation (**Eq. 3b**) were used to calculate set 2, setting target g_1 to zero in the constrained locations still conserves $b_{11} + b_{22} + b_{33} = 0$ as shown in **Eq. (3a-c)**. Hence, b_{33} is also perfectly reconstructed with set 2. Therefore, **Figs. 2-5** show that with target g_{1c}, two sets of target g_{nc} coefficients may be found that can collectively reconstruct all four target anisotropy components correctly.

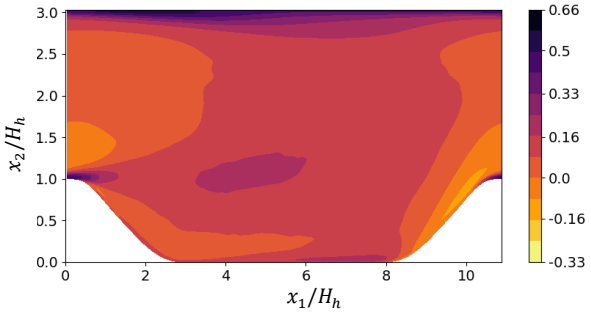
The colorbar limits in **Figs. 2-5** have been chosen such that regions with reconstructed b_{ij} values outside of the realizability limits given in **Eq. (6)** are shown in white. This indicator in **Figs. 3-5** shows that the blue circled regions where reconstructed b_{ij} (using target g_{nc}) is not equal to target b_{ij} may give unrealizable values. Hence, training TBNNs with one set of target g_{nc} values may not only lead to the TBNN predicting inaccurate b_{ij} values, but also unrealizable values in some components. With two sets of target g_{nc} values, two TBNNs can be trained in parallel, where the first aims to predict the first set and the second aims to predict the second set. Then predicted b_{ij} component values may be chosen from the TBNN that is trained on target g_{nc} values that can perfectly reconstruct the target b_{ij} component. For example, if the first set of target g_{nc} values can reconstruct target b_{12} with 100% accuracy such as the example given in **Fig. 5**, then the b_{12} predictions should be taken from the TBNN trained on this set. As the second set of target g_{nc} values lead to reconstructed target b_{22} and b_{33} with 100% accuracy, predicted b_{22} and b_{33} should be taken from the second TBNN that is trained on the second set. This ensures that all b_{ij} components have the possibility of being predicted perfectly (and thus respecting realizability) across the two TBNNs, while the TBNNs are trained to predict $g_1 \leq 0$ due to all target g_{1c} values being ≤ 0.



(a) Reconstructed b_{11} calculated with the first set of g_{nc} coefficients

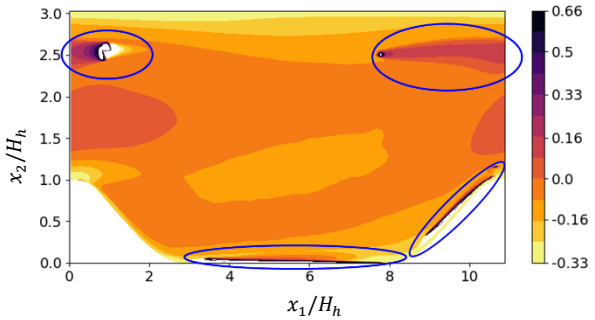


(b) Reconstructed b_{11} calculated with the second set of g_{nc} coefficients

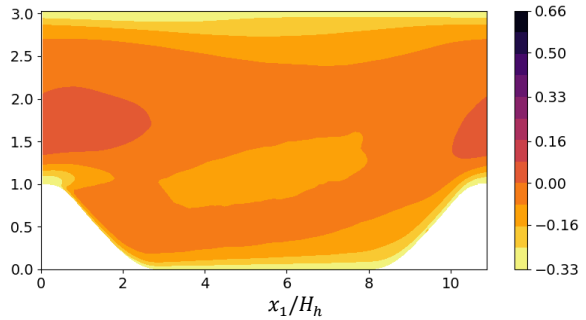


(c) Target b_{11}

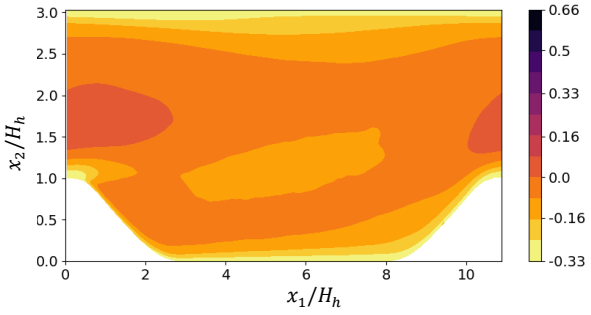
Figure 2 Contour plots of reconstructed b_{11} and target b_{11} for the periodic hills case.



(d) Reconstructed b_{22} calculated with the first set of g_{nc} coefficients

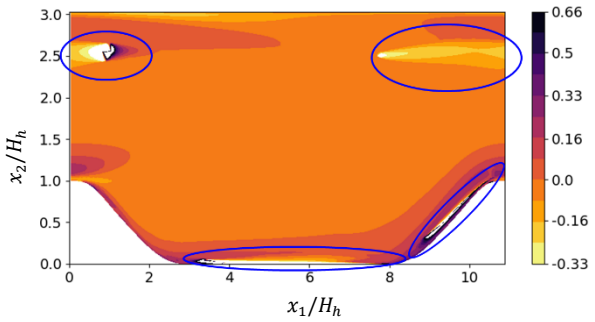


(e) Reconstructed b_{22} calculated with the second set of g_{nc} coefficients

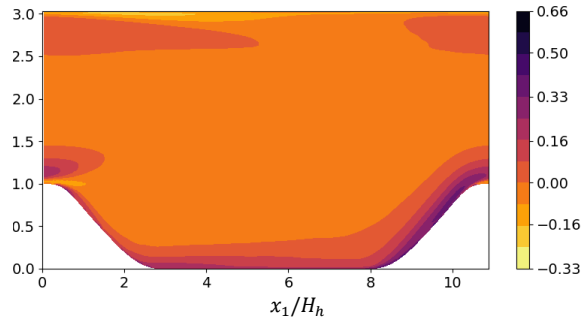


(f) Target b_{22}

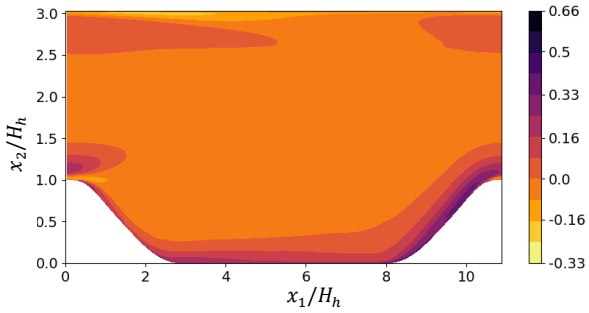
Figure 3 Contour plots of reconstructed b_{22} and target b_{22} for the periodic hills case. Regions of reconstructed b_{22} that are different to target b_{22} are circled in blue, while regions in these circles that are beyond the bounds of realizability are shown in white.



(g) Reconstructed b_{33} calculated with the first set of g_{nc} coefficients

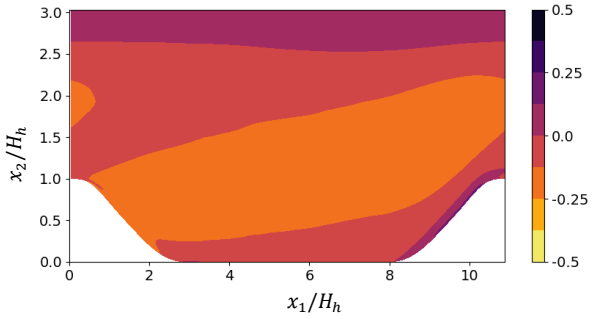


(h) Reconstructed b_{33} calculated with the second set of g_{nc} coefficients

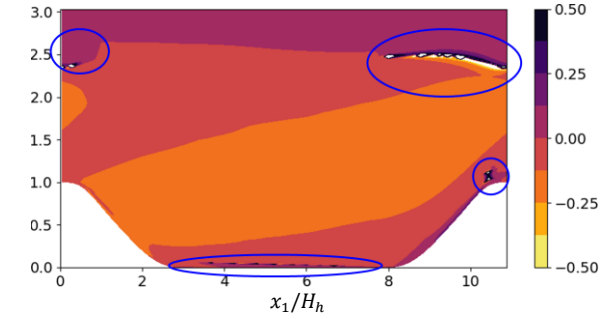


(i) Target b_{33}

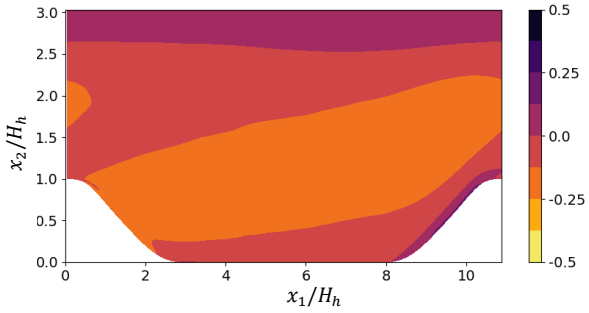
Figure 4 Contour plots of reconstructed b_{33} and target b_{33} for the periodic hills case. Regions of reconstructed b_{33} that are different to target b_{33} are circled in blue, while regions in these circles that are beyond the bounds of realizability are shown in white.



(j) Reconstructed b_{12} calculated with the first set of g_{nc} coefficients



(k) Reconstructed b_{12} calculated with the second set of g_{nc} coefficients



(l) Target b_{12}

Figure 5 Contour plots of reconstructed b_{12} and target b_{12} for the periodic hills case. Regions of reconstructed b_{12} that are different to target b_{12} are circled in blue, while regions in these circles that are beyond the bounds of realizability are shown in white.

CONCLUSION

To improve conditioning of the Reynolds-averaged Navier Stokes equations, it is recognized in the literature that the linear term in tensor-basis neural network (TBNN) anisotropy predictions should be treated implicitly. Therefore, the first coefficient g_1 that TBNNs predict should be less than zero, and the target g_1 coefficient should also be less than zero for the purpose of model training. This work has shown that when target g_1 is constrained to be less than zero, it is only possible for the TBNN to predict two to three components of anisotropy with 100% accuracy and guarantee realizability in those components for two-dimensional flows. Therefore, when this constraint is enforced, it is suggested that two TBNNs may be deployed with the possibility of collectively predicting all four anisotropy components perfectly and with realizable values.

REFERENCES

Duraisamy, K., Spalart, P.R., and Rumsey, C.L., 2017, "Status, Emerging Ideas and Future Directions of Turbulence Modeling Research in Aeronautics", NASA Technical Memorandum 219682.

Duraisamy, K., Iaccarino, G., and Xiao, H., 2019, "Turbulence Modeling in the Age of Data", *Annual Review of Fluid Mechanics*, Vol. 51, pp. 357-377.

Durbin, P.A., and Pettersson Reif, B.A., 2011, *Statistical Theory and Modeling for Turbulent Flows*. 2nd ed. John Wiley & Sons Ltd, Chichester, United Kingdom.

Jiang, C., Vinuesa, R., Chen, R., Mi, J., Laima, S., and Li, H., 2021, "An interpretable framework of data-driven turbulence modeling using deep neural networks", *Physics of Fluids*, Vol. 33, 055133.

Jongen, T., and Gatski, T.B., 1998, General explicit algebraic stress relations and best approximation for three-dimensional flows, *International Journal of Engineering Science*, Vol. 36, pp. 739–763.

Kaandorp, M.L.A., and Dwight, R.P., 2020, "Data-driven modelling of the Reynolds stress tensor using random forests with invariance", *Computers and Fluids*, Vol. 202, 104497.

Ling, J., Kurzawski, A., and Templeton, J., 2016, "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance", *Journal of Fluid Mechanics*, Vol. 807, pp. 155-166.

Man, A., Jadidi, M., Keshmiri, A., Yin, H., and Mahmoudi, Y., 2022, "Optimising a Machine Learning Model for Reynolds Averaged Turbulence Modelling of Internal Flows", *Proceedings of the 16th International Conference on Heat Transfer, Fluid Mechanics and Thermodynamics and Editorial Board of Applied Thermal Engineering*, online.

Man, A., Jadidi, M., Keshmiri, A., Yin, H., and Mahmoudi, Y., 2023, "A divide-and-conquer machine learning approach for modeling turbulent flows", *Physics of Fluids* Vol. 35, 055110.

Man, A., Jadidi, M., Keshmiri, A., Yin, H., and Mahmoudi, Y., 2023, "Non-Unique Machine Learning Mapping in Data-Driven Reynolds Averaged Turbulence Models", *arXiv*:2312.13005.

McConkey, R., Yee, E., and Lien, F-S., 2021, "A curated dataset for data-driven turbulence modelling", *Scientific Data*, Vol. 8, 255.

Pope, S.B., 1975, "A more general effective-viscosity hypothesis", *Journal of Fluid Mechanics*, Vol. 72, pp. 331-340.

Pope, S.B., 2000, *Turbulent Flows*. 1st ed. Cambridge University Press, New York, USA.

Wu, J., Xiao, H., Sun, R., and Wang, Q., 2019, "Reynolds-averaged Navier–Stokes equations with explicit data-driven Reynolds stress closure can be ill-conditioned", *Journal of Fluid Mechanics*, Vol. 869, pp. 553-586.