

## Application of deep learning method to Reynolds stress models of channel flow based on reduced-order modeling of DNS data <sup>\*</sup>

Zhen Zhang <sup>1,2</sup>, Xu-dong Song <sup>3</sup>, Shu-ran Ye <sup>1,2</sup>, Yi-wei Wang <sup>1,2</sup>, Chen-guang Huang <sup>1,2</sup>, Yi-ran An <sup>3</sup>, Yao-song Chen <sup>3</sup>

1. *Key Laboratory for Mechanics in Fluid Solid Coupling Systems, Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China*
2. *College of Engineering Science, University of Chinese Academy of Sciences, Beijing 100049, China*
3. *College of Engineering, Peking University, Beijing 100871, China*

(Received November 2, 2018, Revised November 28, 2018, Accepted November 30, 2018, Published online December 17, 2018)

©China Ship Scientific Research Center 2018

**Abstract:** Recently, the methodology of deep learning is used to improve the calculation accuracy of the Reynolds-averaged Navier-Stokes (RANS) model. In this paper, a neural network is designed to predict the Reynolds stress of a channel flow of different Reynolds numbers. The rationality and the high efficiency of the neural network is validated by comparing with the results of the direct numerical simulation (DNS), the large eddy simulation (LES), and the deep neural network (DNN) of other studies. To further enhance the prediction accuracy, three methods are developed by using several algorithms and simplified models in the neural network. In the method 1, the regularization is introduced and it is found that the oscillation and the overfitting of the results are effectively prevented. In the method 2,  $y^+$  is embedded in the input variable while the combination of the invariants is simplified in the method 3. From the predicted results, it can be seen that by using the first two methods, the errors are reduced. Moreover, the method 3 shows considerable advantages in the DNS trend and the smoothness of a curve. Consequently, it is concluded that the DNNs can predict effectively the anisotropic Reynolds stress and is a promising technique of the computational fluid dynamics.

**Key words:** Deep neural network, channel flow, turbulence model, Reynolds stress

### Introduction

The turbulent flow is complex, however, its importance is increasingly recognized in view of the temporal and spatial disorders<sup>[1]</sup>. Currently, numerous methods, such as the direct numerical simulation (DNS), the large eddy simulation (LES), and the Reynolds-averaged Navier-Stokes (RANS), were used for the numerical simulations of turbulence. In the DNS, the turbulence structure is considered in all scales and the Navier-Stokes (N-S) equation is solved directly. But the DNS requires a sufficient large storage space with a high computational cost. The DNS is mainly used to explore the basic physical

mechanism of turbulence and is rarely used in practical engineering<sup>[2-3]</sup>. In comparison with the DNS, the LES filters a certain amount of pulsation but still relies on a huge amounts of mesh and sometimes, satisfactory results can not be secured, especially, near the wall. Therefore, the LES model is used for relatively simple shear and tube flows<sup>[4-5]</sup>. The RANS is the time average of the unsteady N-S equation. In this model, the turbulent pulsation of each scale might be disregarded with only consideration of the average flow expressed by the physical quantity of the average flow and the characteristic scale of the turbulence; thus, the calculation is performed easily<sup>[6-9]</sup>. The RANS model is extensively used in engineering fluid problems; however, this model cannot always provide satisfactory prediction results. The deficiency of the RANS model originates from the influence of pulsation. In the average flow fields, the eddy viscosity coefficient is isotropic, thus, the RANS model faces considerable challenges in predicting the anisotropic Reynolds stress. Currently, the main improvement direction of the RANS model lies in using the DNS

---

<sup>\*</sup> Project supported by the National Key R&D Program (Grant No. 2016YFC0301601).

**Biography:** Zhen Zhang (1992-), Female, Master, E-mail: zhangzhen2@imech.ac.cn

**Corresponding author:** Yi-wei Wang, E-mail: wangyw@imech.ac.cn

and the well-processed LES data to modify the Reynolds stress of the RANS and in extending considerations from isotropic cases to anisotropic cases. Xing evaluated several methods of the recently developed LES model based on the channel flow<sup>[10]</sup>, and provided a technical support for the calculation of the channel flow by using the LES model. These models were validated and applied to complex flows such as the cavitating flow<sup>[11-12]</sup>, in which many empirical corrections are usually required if the RANS models are adopted.

Machine learning has gained a considerable attention with its ability to perform complex interactions and to achieve excellent results in extensive applications, such as the video, the speech recognition, and the image recognition. Hinton emphasized that the multilayer neural networks enjoy excellent feature-learning abilities, and their training complexities can be effectively reduced through the layer-by-layer initialization, thereby to establish the emergence of the deep learning<sup>[13]</sup>. Under the conditions of the GPU-accelerated hardware, the deep learning is applied to numerous fields in the context of applying the big data recognition. Similarly, attempts were made to use the neural network environments to solve problems of the complex turbulent flow.

One idea of machine learning is using certain *a priori* knowledge in the algorithm that can identify patterns and structures if sufficient training data are present. Tracey et al.<sup>[14]</sup> used a hidden neural network layer to simulate the Spalart-Allmaras (SA) turbulence model and demonstrated its ability to improve the existing computational fluid dynamics (CFD) results through machine learning. The deep neural network (DNN) is a deep learning model with complex and rapid experience and flexibility developments. Ling proposed using the random forests to predict the normalized Reynolds stress anisotropy tensor; however, this research was limited to predicting fully anisotropic tensors. They also used the DNNs for the RANS models with consideration of their computational tractability in modeling a rich set of dynamics induced by turbulent flows<sup>[15-16]</sup>.

Although the DNNs were extensively used and can provide high-quality predictions in complex problems, there are numerous limitations in applying deep learning techniques to turbulence. Numerous research results show low accuracies with incomplete variable choices<sup>[17]</sup>. However, Xiao remarkably changed this problem from the perspective of physics information and proposed a data-driven, physics-informed machine learning approach, which was used to reconstruct the discrepancies in the RANS-simulated Reynolds stress. Furthermore, this method was demonstrated by observing the predicted performance from two scenarios<sup>[18-19]</sup>.

In the present study, we proceed from the limitation of the RANS model in the anisotropic flow field and use the DNS data to train the neural network, and compare the test results with the LES and DNS data. On the basis of the network selection, a regularization method is introduced to the neural networks in the method 1. As mentioned above, the choice of input variables might be inadequate. Therefore, the influence of two variable selection methods on predicting results is discussed. In the method 2,  $y^+$  is embedded in the input variable to characterize the differences between the properties of the near-wall and other regions. In the method 3, the RANS model is used in the calculation, and the 3-D channel flow can be simplified into a 2-D one with the average of the flow field, and the combination of invariants is also simplified. Finally, the three methods are compared with respect to the test errors and the fitting effects of the normalized Reynolds stress anisotropy curve.

## 1. Deep neural network

The DNN is composed of layer nodes. All neurons between the two layers are connected by weights, as illustrated in Fig. 1. The first layer is the input layer, and the last layer is the output layer, the output of one layer becomes the input of the next layer. For any  $L-1$  and  $L$  layers, the nodes of the  $L-1$  layer are connected to all nodes of the  $L$ th layer, that is, when each node of the  $L$ th layer performs calculations, the input is the weighting of all nodes of the  $L-1$  layer after the activation function<sup>[20]</sup> which can be expressed by the formula  $f$  in Fig. 2. By using multilayer transformation, the DNN can capture the complex levels of the features. The layer between the input and the output layers is called the hidden layer because the physical interpretation of its activation is unclear.

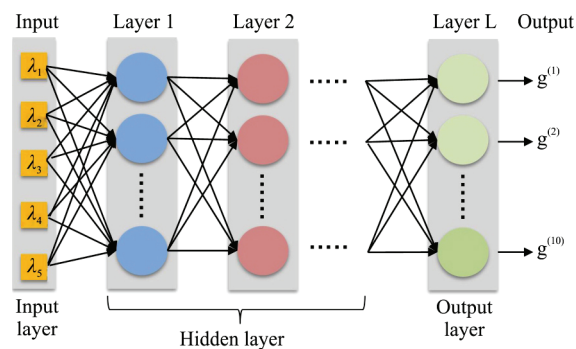


Fig. 1 (Color online) DNN structure

In this study, we program in the language of Python with calls from Tensorflow, which is Google's

open source deep learning library (<https://tensorflow.google.cn>), and we can use this framework and the Python programming language to build a large number of machine learning-based applications. There were many applications built by TensorFlow or other framework to GitHub.

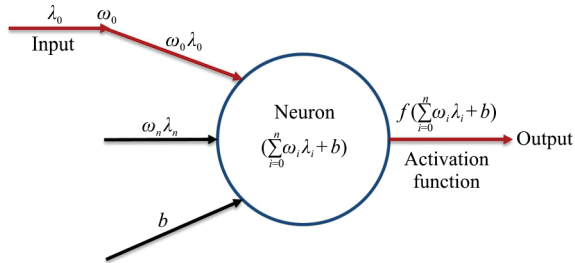


Fig. 2 (Color online) Principle of transmission and activation of neuron node in DNN structure

In the calculation process, the network structure and the training parameters must be defined and initialized first. Followed by the network input layer by reading some quantities obtained by processing the flow field data, and implementing the multilayer conversion by using the activation function. Common excitation functions are the sigmoid, the hyperbolic tangent, the rectified linear units (ReLU), the leaky ReLU, and the exponential linear unit (ELU)<sup>[21-24]</sup>. In this paper, the leaky ReLU is used as the activation function, because as Maas<sup>[24]</sup> Maas confirmed that this function converges rapidly, and the gradient solution is simple. Then, according to the error between the values of the network output layer and the real value, the weight matrix is adjusted through a backpropagation with the gradient descent until the number of iterations is reached. And some of the main parameters of the model training are given in Section 3.1.

$$\{S, R\} = \frac{k}{2\varepsilon} \{S, R\} \quad (1)$$

$$\{S, R\} = \frac{\nu}{u_\tau^2} \{S, R\} \quad (2)$$

The input information of the neural network is the average strain rate tensor  $S$  and the average rotation rate tensor  $R$  in the flow field. Pope<sup>[25]</sup> used a turbulent flow energy  $k$  and a turbulent dissipation rate  $\varepsilon$  for the dimensionless processing, as expressed in Formula (1)<sup>[25]</sup>. In the present study, the wall friction velocity  $u_\tau$  and the kinematic viscosity  $\nu$  are used for the dimensionless processing, as defined in Formula (2), and the output is the normalized Reynolds stress anisotropy tensor, as presented in Formula (3).

Pope derived the relevant integrity basis. In the most common incompressible case, the general expression of the eddy viscosity model (3) is a function of  $S$  and  $R$ , these functions can be expressed as a linear combination of 10 isotropic basis tensors<sup>[25]</sup>, as expressed in Formula (4).

$$b_{ij} = \frac{\overline{u'_i u'_j}}{2k} - \frac{1}{3} \delta_{ij} \quad (3)$$

$$\mathbf{b} = \sum_{n=1}^{10} g^{(n)}(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5) \mathbf{T}^{(n)} \quad (4)$$

The rotation invariance is a fundamental physical principle, which indicates that the physical observation of the fluid flow does not depend on a coordinate system, and any turbulent closure follows this principle, to ensure the Galilean invariance<sup>[26]</sup>. Any tensor  $\mathbf{b}$  that satisfies this principle will automatically satisfy the Galilean invariance because the network structure embeds a rotation invariant. Pope deduced the following 10 tensors  $\mathbf{T}^{(1)}, \dots, \mathbf{T}^{(10)}$  in detail<sup>[25]</sup>:

$$\mathbf{T}^{(1)} = S, \quad \mathbf{T}^{(2)} = SR - RS, \quad \mathbf{T}^{(3)} = S^2 - \frac{1}{3} I_3 \cdot Tr(S^2),$$

$$\mathbf{T}^{(4)} = R^2 - \frac{1}{3} I_3 \cdot Tr(R^2), \quad \mathbf{T}^{(5)} = RS^2 - S^2 R,$$

$$\mathbf{T}^{(6)} = R^2 S + SR^2 - \frac{2}{3} I_3 \cdot Tr(SR^2),$$

$$\mathbf{T}^{(7)} = RSR^2 - R^2 SR, \quad \mathbf{T}^{(8)} = SRS^2 - S^2 RS,$$

$$\mathbf{T}^{(9)} = R^2 S^2 + S^2 R^2 - \frac{2}{3} I_3 \cdot Tr(S^2 R^2),$$

$$\mathbf{T}^{(10)} = RS^2 R^2 - R^2 S^2 R \quad (5)$$

The five invariants  $\lambda_1, \dots, \lambda_5$  are scalar functions of  $S$  and  $R$ , and the results are as follows:

$$\lambda_1 = Tr(S^2), \quad \lambda_2 = Tr(R^2), \quad \lambda_3 = Tr(S^3),$$

$$\lambda_4 = Tr(R^2 S), \quad \lambda_5 = Tr(R^2 S^2) \quad (6)$$

## 2. Results of LES model

In order to validate the prediction precision of the normalized Reynolds stress anisotropy tensor as

obtained from the neural network, we use the LES model for the calculations of the channel flow, with the results as a comparison reference to examine the accuracy of the DNN. The sizes of the dimensionless computational domain are  $5 \times 3 \times 2.4$ . The initial and boundary setting conditions are from the turbulence.ices.utexas.edu file server. The channel flow of the friction Reynolds number  $Re_\tau = 2000$  is simulated, and the velocity  $u$  and the turbulent energy  $k$  are calculated. Results of the dimensionless velocity  $u^+$  and  $k$  obtained with the LES model and by the DNS are compared, as shown in Fig. 3, in which  $Re_\tau$ ,  $u^+$  and  $y^+$  are defined as

$$\{Re_\tau, u^+, y^+\} = \left\{ \frac{u_\tau \delta}{\nu}, \frac{u}{u_\tau}, \frac{u_\tau y}{\nu} \right\} \quad (7)$$

where  $\delta$  is the half width of the channel flow.

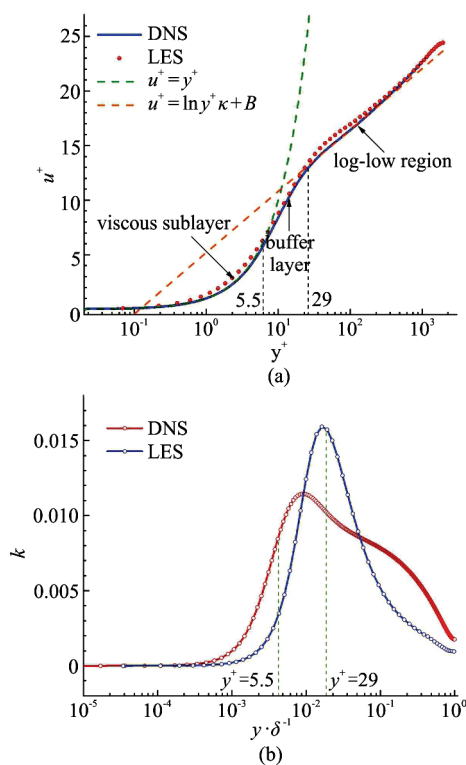


Fig. 3 (Color online) Comparison results of velocity field ( $u^+$ ) and turbulent kinetic factor ( $k$ ) between LES and DNS results. (a) Velocity field adheres to a linear relationship in the viscous sublayer and obeys the logarithmic law in the log-law region, and the middle layer is the buffer layer. (b) The variation of turbulent kinetic factor with  $y/\delta$

In Fig. 3(a), it can be seen that the velocity values in the LES and DNS results agree well with

each other. And the velocity profile adheres to a linear relationship in the viscous sublayer while  $y^+ < 5.5$ . However, in the region of  $y^+ > 29$ , the velocity obeys the logarithmic law, and the buffer layer is between the two regions. Both the viscosity and the turbulence in the buffer layer play an important role, so the properties of the region are complicated. Corresponding to Fig. 3(a), Fig. 3(b) shows that the difference of the turbulent kinetic factor between the LES and DNS results is increased in the buffer zone, and then decreases in the turbulent zone. Therefore, we may say that the current LES results are reasonable.

### 3. Model training, testing and improvement

#### 3.1 Network selection

The turbulence of the channel boundary layer is a wall-oriented and outwardly changing flow configuration, and the neural network is trained and tested on the basis of the DNS dataset of the five groups of flow fields (from turbulence.ices.utexas.edu file server,  $Re_\tau = 180, 550, 1000, 2000$  and  $5200$ ) [27]. This section focuses on the training, the testing, and the improvement of the model. And the impact and the importance of the network selection, the regularization and the variable selection are studied separately.

Table 1 Parameters of training neural networks

Parameters	Value
Training steps	$10^4, 10^5, 2 \times 10^5$
Learning rates	$2.5 \times 10^{-3}, 5 \times 10^{-5}, 2.5 \times 10^{-7}$
Activation function	Leaky ReLU
Loss function	$RMSE_0 = \sqrt{(b_{j,out} - b_{j,DNS})^2 / M}$
Optimizer	AdamOptimizer

Under different situations, the amount of data obtained through the CFD is different. Thus, different options must be considered for judging the type and the size of the network in the deep learning. Figure 1 depicts the small DNNs (DNN\_S), which are used in the present study. Five input layer parameters ( $\lambda_1 - \lambda_5$ ) are considered, with four hidden layers, 20 nodes per layer, and 10 nodes in the last layer, where  $n = 1, \dots, 10$ . The output is  $g^1, \dots, g^{10}$ , multiplied by ten tensors  $T^{(1)}, \dots, T^{(10)}$ . Finally, the anisotropic stress tensors that satisfy Formula (4) are obtained, to ensure the invariance of the Galileo network prediction. Table 1 lists several parameters of the training neural networks. The results obtained with the large network structure used by Ling (DNN\_L) are compared. The neural network input layer of DNN\_L is unchanged, then there are 8 hidden layers, 30 nodes per layer, and



**Table 2 RMSE results of RANS, LES, DNN\_L, and DNN\_S models**

Model	Duct	Flow over wavy wall	Channel-1	Channel-2
RANS	0.25	0.18	0.60	0.60
LES	-	-	0.08	0.08
DNN_L <sup>[16]</sup>	0.14	0.08	0.12	0.10
DNN_S	-	-	0.06	0.05

the final output layer contains also 10 nodes, but this large network may only rely on certain neurons on a given training set, which will result in an over-fitting of the data set. We use a small network structure, by turning off some neurons, allowing more neurons to participate in the final output.

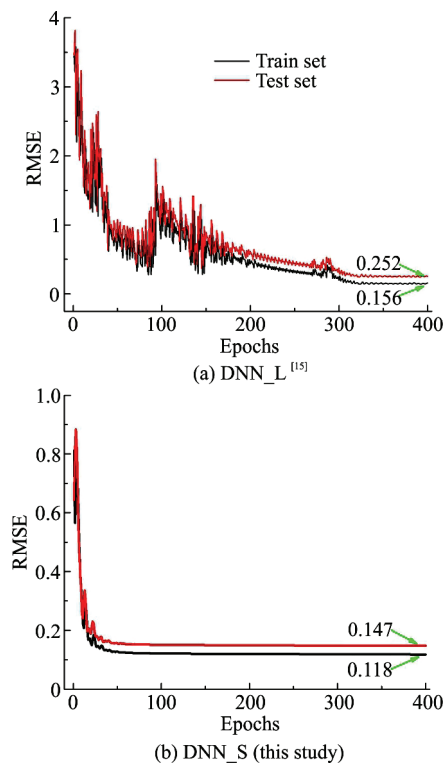


Fig. 4 (Color online) Training and testing error results of the DNN

The error results obtained by using the RANS model, the LES model, the DNN\_L, and DNN\_S models used in this study, are summarized in Table 2. The duct and the flow over the wavy wall are selected as the training and testing cases in the DNN\_L, correspondingly. Channel-1, which includes three training sets ( $Re_\tau = 180, 1\ 000$  and  $5\ 200$ ) and one test set ( $Re_\tau = 2\ 000$ ), and Channel-2, which includes two datasets ( $Re_\tau = 180, 550, 1\ 000$  and  $5\ 200$ ), are fused for the training, and one dataset ( $Re_\tau = 2\ 000$ ) is used for testing. In Table 2, the two sets of result errors (0.12 and 0.10) for the channel flow obtained by using the DNN\_L are between 0.08 (flow over a

wavy wall) and 0.14 (duct). The error results from the DNN\_S are 0.06 and 0.05. These error results are smaller than the minimum errors of the DNN\_L (0.08) and the error results of the LES and RANS simulations. Therefore, the rationality of the network is verified, with reduced error results.

Our error analysis of the training and the testing of the first 400 epochs in the two network structures (DNN\_L and DNN\_S) is demonstrated in Figs. 4(a), 4(b). With the network structure built in this study, the training efficiency is improved remarkably. With the DNN\_L, a relatively large error is obtained at first because many neurons are used, therefore, for a given training set, several neurons will participate in the forward and backward propagations for certain outputs, which only depend on a part of the neuron nodes, thus the computational efficiency will be affected with the overfitting.

From the rationality and the efficiency of the neural network, the DNN\_S is verified to be suitable for the training and the testing of the channel flow field data. The following results are based on our DNN\_S network structure.

### 3.2 Regularization

To prevent a high-frequency oscillation caused by excessive weight of several network nodes, we use  $L_2$  regularization<sup>[28]</sup>, as defined by Formula (8), where  $RMSE_0$  is the original loss function, and the rest  $\alpha \sum_{\omega} \omega^2$  is the  $L_2$  regularization term, where  $\alpha$  is the regularization coefficient, and the regular term is the sum of the squares of the parameters. The addition of the  $L_2$  regularization term after the original loss function  $RMSE_0$  is equivalent to generating a constraint on  $RMSE_0$ , that is, letting the regular term  $\|\omega\|^2$  of the  $L_2$  norm be the smallest. Each element of  $\omega$  can be small and nearly equal to 0. The task of the machine learning is to find the minimum value of the loss function through several methods, such as the gradient descent. If  $L$  is equal to the regularization term, then  $RMSE = RMSE_0 + L$ , at which point our task becomes a solution for finding the minimum value of  $RMSE_0$  under the  $L$  constraint.

$$RMSE = RMSE_0 + \alpha \sum_{\omega} \omega^2 \quad (8)$$

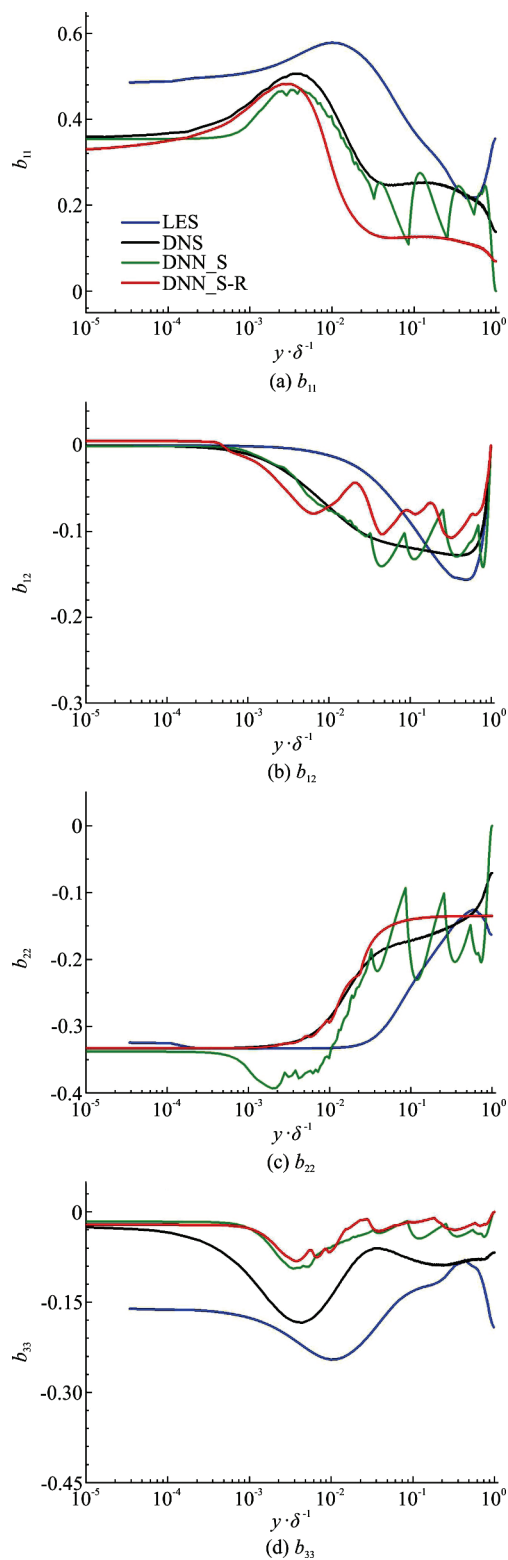


Fig. 5 (Color online) Prediction results from DNN\_S and Method 1 (DNN\_S-R (DNN combined with regularization)) and comparison results with LES and DNS results ( $Re_\tau = 2000$ ) of the channel flow

Figure 5 shows the normalized Reynolds stress

anisotropy curve of the channel flow predicted by the DNN\_S, and the DNN\_S models, combined with the regularization (DNN\_S-R) and compared with the LES and DNS results. And the results of the Reynolds stress calculated by the LES model and the DNS are also shown in Fig. 5. Here, only  $b_{11}$ ,  $b_{12}$ ,  $b_{22}$  and  $b_{33}$  are shown because  $b_{13}$  and  $b_{23}$  should be zero due to (statistical) symmetries of the flow. The DNN\_S test results are consistent with the DNS results in the trend, but with high-frequency oscillations, and the prediction results are not ideal. The oscillation presented in Fig. 5 appears nearly in the range of  $5.5 < y^+ < 29$ , which represents the buffer layer illustrated in Fig. 3. The nature of the buffer layer is complex, related with certain difficulties in the training of the model, as causes of the oscillation of the test results. After introducing the regularization, the prediction results see a smooth change, and the overfitting phenomenon is alleviated to some extent. But the smoothness of the curve is not as good as the result of the LES model.

Table 3 RMSE results of the three methods

Methods	RMSE
Network Type: DNN_S	0.06
Method 1: DNN combined with regularization (DNN_S-R)	0.10
Method 2: $y^+$ is embedded in the input invariant (DNN_S- $y^+$ )	0.05
Method 3: The combination of the invariants is simplified (DNN_S-2D)	0.04

The test error results of the three methods for predicting the normalized Reynolds stress anisotropy of the channel flow are shown in Table 3. Although the oscillation of the anisotropic Reynolds stress curve decreases after introducing the regularization, the test error increases because the error caused by the regularization is added to the supervised error result. The final error results do not represent the accuracy of predicting the anisotropic Reynolds stress. Therefore, on the basis of selecting the DNN-S and the regularization of the network, the impact of the network input variables is considered.

### 3.3 Variable selection

In addition to the regularization (the method 1), the choice of variables is crucial for the flow configuration of the channel boundary layer turbulence. The properties of the channel flow in the near-wall and in other regions are essentially different<sup>[30]</sup>, and to well distinguish the differences of the properties in the method 2,  $y^+$  is used as an input variable together with  $\lambda$ . In the method 3, the RANS model averages the time of the unsteady N-S equation, and from the

DNS data of the channel flow, we conclude that  $b_{13}$  and  $b_{23}$  are nearly equal to 0, thus, the 3-D flow may be simplified to a 2-D flow. Simultaneously, the combination of the invariants may be simplified in accordance with the Pope's derivation<sup>[25]</sup>. The input invariables of the neural network in the 2-D flow field are simplified as  $\lambda_1$ ,  $\lambda_2$ , and Formulas (4)-(6) are reduced to Formulas (9)-(11), respectively.

The test error results of the method 2 (with  $y^+$  embedded in the input invariant) and the method 3 (with the combination of the invariants simplified) for predicting the anisotropic Reynolds stress of the channel flow ( $Re_\tau = 2000$ ) are summarized in Table 2. The predicted anisotropic Reynolds stress curves are shown in Fig. 6 and compared with the LES and DNS results. The two variable selection methods have slightly improved the accuracy of the DNN\_S prediction of the anisotropic Reynolds stress but a minimal oscillation in the Method 1 remains. The result of the method 2 shows considerable advantages in the variation trend with the DNS and the smoothness of the curve.

$$\mathbf{b} = \sum_{n=0}^2 g^{(n)}(\lambda_1, \lambda_2) \mathbf{T}^{(n)} \quad (9)$$

$$\mathbf{T}^{(0)} = \frac{1}{3} \mathbf{I}_3 - \frac{1}{2} \mathbf{I}_2, \quad \mathbf{T}^{(1)} = \mathbf{S}, \quad \mathbf{T}^{(2)} = \mathbf{SR} - \mathbf{RS} \quad (10)$$

$$\lambda_1 = Tr(\mathbf{S}^2), \quad \lambda_2 = Tr(\mathbf{R}^2) \quad (11)$$

#### 4. Conclusions

In this paper, the DNN structure is constructed and improved on the basis of the type and the size of the dataset. The DNN has no universal applicability for different problems, and the rationality and the high efficiency of the neural network constructed in this study are validated.

Three methods of the DNN are developed with several algorithms and simplified models, and the impact and the importance of the network selection, the regularization and the variable selection are studied separately. From the test results of the method 1 (the DNN combined with regularization), the regularization can suppress the oscillation of the results and prevent the overfitting. In the prediction results of the method 2 ( $y^+$  is embedded in the input invariant) and the method 3 (the combination of the invariants is simplified), the errors are reduced effectively. Furthermore, the method 3 demonstrates considerable advantages in the DNS trend and in the smoothness of the curve, which may be superior in

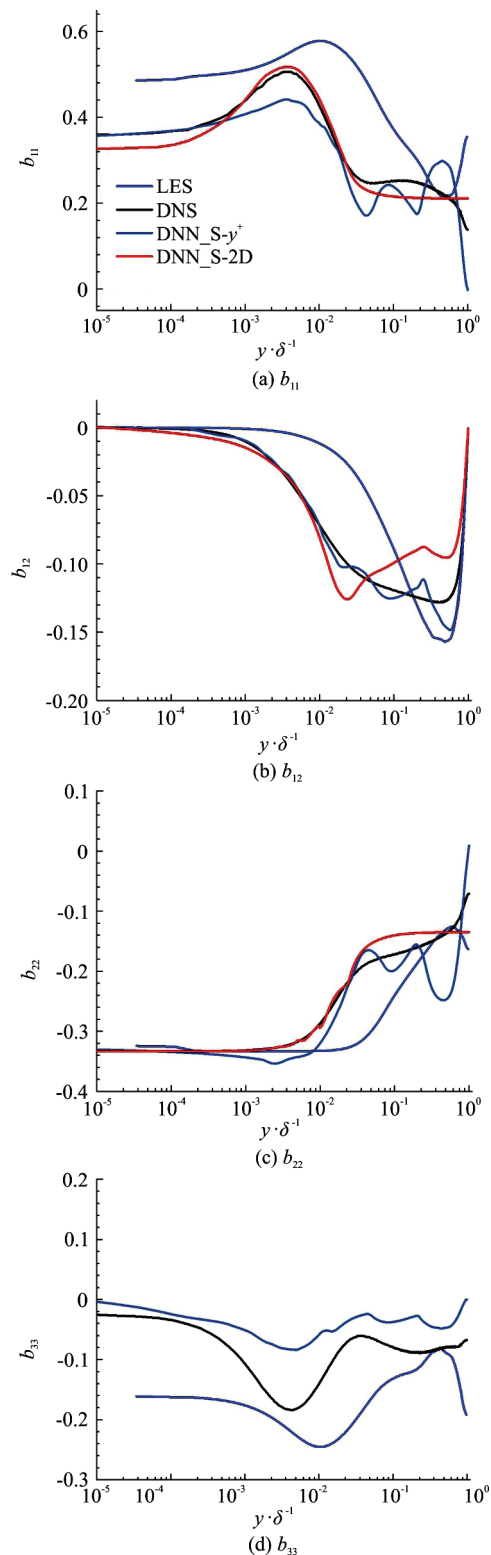


Fig. 6 (Color online) Prediction results in method 2, (DNN\_S- $y^+$  (with  $y^+$  embedded in the input invariant)), method 3 (DNN\_S-2D (with the combination of the invariants simplified)), and comparison results with LES and DNS ( $Re_\tau = 2000$ ) of the channel flow

predicting the anisotropic Reynolds stress.

From the current work, it is suggested that the prediction accuracy of the DNN results is higher than that of the LES model, however, the universal applicability of the model has not been found. Therefore, the future improvements will include applying the well-processed LES results as a supervisory volume, increasing the training datasets, and using more results with higher or lower Reynolds number as the test datasets.

In addition, the depth neural network has a favorable prediction performance in the channel flow and can also be applied to other flows. Until recently, the network structure built in this study has been used in 2-D cylinders, round tube jets and other flows and has achieved preliminary results, meanwhile, improvements are still required in future studies. And the results of this work can also be applied to the complex flows such as the cavitating flow to replace empirical corrections in the RANS models.

## References

- [1] Vandriest E. R. On turbulent flow near a wall [J]. *Journal of the Aeronautical Sciences*, 1956, 23(11): 1007-1011.
- [2] Feiereisen W. J. Numerical simulation of a compressible homogeneous, turbulent shear flow [D]. Doctoral Thesis, Stanford, USA: Stanford University, 1981.
- [3] Benzi R., Patarnello S., Santangelo P. On the statistical properties of two-dimensional decaying turbulence [J]. *Europhysics Letters*, 1987, 3(7): 811-818.
- [4] Brachet M. E., Meneguzzi M., Sulem P. L. Small-scale dynamics of high-reynolds-number two-dimensional turbulence [J]. *Physical Review Letters*, 1986, 57(6): 683-686.
- [5] Gilbert A. D. Spiral structures and spectra in two-dimensional turbulence [J]. *Journal of Fluid Mechanics*, 1988, 193: 475-497.
- [6] Pope S. B. Turbulent flows [M]. Cambridge, UK: Cambridge University Press, 2001.
- [7] Rodi W. On the simulation of turbulent flow past bluff bodies [J]. *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, 46-47: 3-19.
- [8] Speziale C. G. A review of Reynolds stress models for turbulent shear flows [C]. *20th Symposium on Naval Hydrodynamics*, Washington DC, USA, 1995.
- [9] Spalart P. R. Philosophies and fallacies in turbulence modeling [J]. *Progress in Aerospace Sciences*, 2015, 74: 1-15.
- [10] Dutta R., Xing T. Five-equation and robust three-equation method for solution verification of large eddy simulations [J]. *Journal of Hydrodynamics*, 2018, 30(1): 23-33.
- [11] Cheng H. Y., Long X. P., Liang Y. Z. et al. URANS simulations of the tip-leakage cavitating flow with verification and validation procedures [J]. *Journal of Hydrodynamics*, 2018, 30(3): 531-534.
- [12] Wang C. C., Huang B., Wang G. Y. et al. Numerical simulation of transient turbulent cavitating flows with special emphasis on shock wave dynamics considering the water/vapor compressibility [J]. *Journal of Hydrodynamics*, 2018, 30(4): 573-591.
- [13] Hinton G., Salakhutdinov R. Reducing the dimensionality of data with neural networks [J]. *Science*, 2006, 313(5786): 504-507.
- [14] Tracey B., Duraisamy K., Alonso J. Application of supervised learning to quantify uncertainties in turbulence and combustion modeling [C]. *51st AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Texas, USA, 2013, 2013-0259.
- [15] Ling J., Ruiz A., Lacaze G. et al. Uncertainty analysis and data-driven model advances for a jet-in-crossflow [J]. *Journal of Turbomachinery*, 2016, 139(2): 021008.
- [16] Ling J., Kurzawski A., Templeton J. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance [J]. *Journal of Fluid Mechanics*, 2016, 807: 155-166.
- [17] Kutz J. N. Deep learning in fluid dynamics [J]. *Journal of Fluid Mechanics*, 2017, 814: 1-4.
- [18] Wang J. X., Wu J. L., Xiao H. Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data [J]. *Physical Review Fluids*, 2017, 2(3): 1-22.
- [19] Xiao H., Wu J. L., Wang J. X. et al. Quantifying and reducing model-form uncertainties in Reynolds-averaged Navier-Stokes simulations: A data-driven, physics-informed Bayesian approach [J]. *Journal of Computational Physics*, 2016, 324(C): 115-136.
- [20] Lecun Y., Bengio Y., Hinton G. Deep learning [J]. *Nature*, 2015, 521(7553): 436.
- [21] Jarrett K., Kavukcuoglu K., Ranzato M. et al. What is the best multi-stage architecture for object recognition? [C]. *International Conference on Computer Vision (ICCV)*, Kyoto, Japan, 2009, 2146-2153.
- [22] Krizhevsky A., Sutskever I., Hinton G. E. ImageNet classification with deep convolutional neural networks [C]. *25th International Conference on Neural Information Processing Systems*, Nevada, USA, 2012, 1097-1105.
- [23] He K., Zhang X., Ren S. et al. Deep residual learning for image recognition [C]. *International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015, 1026-1034.
- [24] Maas A. L., Hannun A. Y., Ng A. Y. Rectifier nonlinearities improve neural network acoustic models [C]. *International Conference on Machine Learning (ICML)*, Atlanta, USA, 2013.
- [25] Pope S. B. A more general effective-viscosity hypothesis [J]. *Journal of Fluid Mechanics Digital Archive*, 1975, 72(2): 331-340.
- [26] Jones R., Templeton J., Ling J. Machine learning strategies for systems with invariance properties [J]. *Journal of Computational Physics*, 2016, 318(8): 22-35.
- [27] Moser R. D., Kim J., Mansour N. N. Direct numerical simulation of turbulent channel flow up to  $Re_\tau = 590$  [J]. *Physics of Fluids*, 1999, 11(4): 943-945.
- [28] Krogh A., Hertz J. A. A simple weight decay can improve generalization [C]. *International Conference on Neural Information Processing Systems*. San Mateo, California, USA, 1992, 950-957.