The Aeronautical Journal (2024), 1–22
doi:10.1017/aer.2024.123

RESEARCH ARTICLE

Involving non-equilibrium training dataset in data-driven turbulence modeling for turbomachinery

J.L. Du[1], W.S. Liu[2,3], L. Fang[1] and T.W. Bao[4]

[1]Laboratory of Complex Systems, Ecole Centrale de Pékin, Beihang University, Beijing, China
[2]School of Energy and Power Engineering, Beihang University, Beijing, China
[3]College of Engineering, Peking University, Beijing, China
[4]BSS TurboTech, Beijing, China
Corresponding authors: L. Fang; Email: le.fang@buaa.edu.cn, T.W. Bao; Email: tianwei.bao@bssturbotechltd.cn

Received: 10 December 2023; **Revised:** 19 September 2024; **Accepted:** 9 October 2024

Keywords: k − ω SST model; machine learning; non-equilibrium turbulence; turbine cascade

Abstract

In order to improve the performance of $k - \omega$ SST model in turbomachinery, previous studies have used the machine-learning (ML) technique to obtain turbulence models (for example, the ML-RANS EQ model). However, these models do not lead to satisfactory results in complex flows in turbomachinery. In this study, we use non-equilibrium training dataset to obtain a new turbulence model (i.e., the ML-RANS TR-NE-EQ model). Calculations in various cases of turbine cascade flows show that ML-RANS TR-NE-EQ model performs obviously better than ML-RANS EQ model as well as $k - \omega$ SST model.

Nomenclature

$\boldsymbol{a}$	anisotropy Reynolds stress tensor
C_ϵ	dissipation coefficient
$C_{p_{\text{static}}}$	pressure coefficient
$C^{\text{loss}}_{p_{\text{total}}}$	total pressue loss coefficient
d	wall distance
d_b	diameter of cylindrical bar in turbine cascades
h	channel half width in high-fidelity training data
i, j	index
$\boldsymbol{I}$	unit tensor
k	turbulence variable in the $k - \omega$ SST model
l	chord in turbine cascades
l_{ax}	axial chord in turbine cascades
l_b	distance of cylindrical bar from the leading edge in turbine cascades
L_x	streamwise location in high-fidelity training data
$\mathcal{L}$	integral length scale
m	number of layers in the neural network
n	number of nodes in the neural network
N	number of high-fidelity training data
p	pressure
p_{total}	total pressure
$p^{\text{inlet}}_{\text{total}}$	total pressure at inlet
q_1	first input feature for the practical ML-RANS model
q_2	second input feature for the practical ML-RANS model
Re	Reynolds number
Re_m	Reynolds number based on bulk mean velocity

Re_λ	Taylor-scale Reynolds number
Re_τ	Reynolds number based on friction velocity
$\boldsymbol{S}$	mean strain rate
$\boldsymbol{S_h}$	mean strain rate in high-fidelity training data
S_k	skewness of longitudinal velocity derivative
t	pitch of turbine cascades
u_τ	friction velocity
U_i	component of time-averaged velocity
U_m	bulk mean velocity
U^+	dimensionless time-averaged velocity
$\mathcal{U}^2$	turbulent kinetic energy
$w_{i,j}$	one weight of the neural network
y	pitchwise coordinate in turbine cascades
d^+	dimensionless wall distance
$d^+\|_{wall}$	dimensionless wall distance of the first-level mesh near wall
z	Streamwise coordinate in turbine cascades

Greek symbols

$\beta_s, \beta_1, \beta_2$	three kinds of angle in experiment of turbine cascades
Δx^+	homogeneous streamwise mesh size
Δy^+	first-level normal mesh size from the wall
Δz^+	homogeneous spanwise mesh size
ϵ	turbulence dissipation
λ	L1 regularisation coefficient
ν	kinematic viscosity
ν_t	eddy viscosity
ν_t^m	optimal eddy viscosity calculated with high-fidelity training data
ν_t^*	initial learning target of the ML-RANS framework
ν_t^0	target variable for the practical ML-RANS model
$\nu_t^{0,i}$	one target variable for the practical ML-RANS model in high-fidelity training data
$\widehat{\nu_t^{0,i}}$	one target variable predicted by the neural network
ξ_k	dimensionless cross diffusion of k and ω
$\boldsymbol{\tau}$	Reynolds stress
$\boldsymbol{\tau}_h$	Reynolds stress in high-fidelity training data
ω	turbulence variable in the $k-\omega$ SST model
$\boldsymbol{\Omega}$	rotation rate

Abbreviations

CFD	computational fluid dynamics
DNS	direct numerical simulation
EQ	equilibrium
FIML	field inversion machine learning
LES	large eddy simulation
M1	method of calculating (k, ω) with their original physical definition
M2	method of calculating (k, ω) with $k-\omega$ SST model
ML	machine learning
ML-RANS	iterative machine-learning framework for RANS turbulence modeling
NE	non-equilibrium
PSB	periodic suction and blowing disturbance
RANS	Reynolds-averaged Naviers-Stokes
SST	shear stress transport
TR	transition

1.0 Introduction

In recent decades, numerical simulation methods have been widely applied and have generated a vast amount of flow data. It is still expensive to employ high-fidelity methods such as direct numerical simulation (DNS) and well-resolved large eddy simulation (LES) for complex flows. Thus, the Reynolds-averaged Navier-Stokes (RANS) simulation remains a prevalent method in engineering applications. RANS equations incorporate a Reynolds stress tensor term that necessitates closure through a turbulence model. Towards the end of the 20th century, NASA researchers [1] conducted a comparative study on popular eddy viscosity models including the $k-\epsilon$ model [2], $k-\omega$ model [3], $k-\omega$ SST model [4, 5], and the Spalart-Allmaras model. They found that the SST model demonstrated the best overall performance because it not only matched the superior performance of other models in simulating simple flows but also excelled in handling complex flows with adverse pressure gradient phenomena. Recently, Chipongo et al. [6] compared the simulation outcomes of the realisable $k-\epsilon$ model, $k-\omega$ SST model and the Reynolds stress model (RSM), which directly solves the Reynolds stress transport equations, in spatially varied flows. Taking into account both cost and accuracy, they discovered that these three models were comparably effective. Consequently, this study adopted the SST model as the baseline model.

Despite the prevailing use of the Boussinesq eddy viscosity assumption, there are two issues worthy of attention regarding two-equation models like the SST model. One issue is that deriving turbulence models using theoretical knowledge and observational data derived from equilibrium flows inherently renders these models unsuitable for numerical simulations of non-equilibrium flows at a fundamental level. Fang et al. [7] quantitatively described non-equilibrium phenomena in compressor flows using the equation of Lagrangian velocity gradient correlation, revealing that non-equilibrium time-scales are comparable to the main flow time-scales. This underscores the prevalence of non-equilibrium phenomena in turbomachinery. However, as Wilcox [8] explains, the scalar attribute and initial definition of turbulence viscosity derive from the rigorous theoretical analysis of the Richardson-Kolmogorov energy cascade. Calibration of turbulence model coefficients heavily relies on decay homogeneous isotropic turbulence and zero-pressure-gradient boundary layers, leading Spalart [9] to advocate for a reduction in the proportion of cases similar to homogeneous isotropic turbulence, aiming to improve the performance of turbulence models in more practical cases such as jets, wakes, boundary layers, among others.

The other issue is that turbulence variables in eddy viscosity models may not adhere to their initial definitions. As an example, in the $k-\omega$ models, Wilcox [8] introduce a stress limiter in the calculation formula for turbulene viscosity, while Menter [4, 5] devises a blending function for the SST model. These artificial or empirical enhancements, although improving the predictive accuracy of the $k-\omega$ models, cause the turbulence viscosity and the eddy viscosity model to deviate from its physical underpinnings. Furthermore, the modeled transport equations are not rigorously derived. The closure approximations used in the transport equations for k and ϵ inevitably reduce the accuracy of the solution, and excessive approximations can even render the transport equations overly dependent on the turbulence data required for modeling. Additionally, the ω equation is derived via dimensional analysis based on the variation process of specific dissipation rate. Hence, the necessary modeling of the transport equations within eddy viscosity models constitutes a systematic factor contributing to the deviation of turbulence variables from their physical definitions. In Spalart's perspective [9], attempting to obtain accurate turbulence variables from turbulence models represents a fallacy in turbulence modeling, given his observation that the best $k-\epsilon$ model arguably possesses an inaccurate k, which significantly complicates the use of DNS data for modeling purposes.

Machine learning (ML) methodologies have been increasingly employed to enhance the predictive capabilities of computational fluid dynamics (CFD), particularly concerning the improvement and modification of established turbulence models [10, 11]. A notable strategy involves leveraging ML techniques to ascertain a generalised formulation of the Reynolds stresses. This stress tensor can be represented as a function of the strain rate ($\boldsymbol{S}$) and rotation rate ($\boldsymbol{\Omega}$), formulated as a linear combination of ten tensorial basis where the coefficients are functions of five scalar invariants [12]. Ling et al. [13] pioneeringly

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

introduced a framework entitled Tensor Basis Neural Networks (TBNN), which maps the local scalar features to the Reynolds stress anisotropy tensor. Following this groundbreaking work, Pope's seminal analysis [12] has been widely embraced by the majority of research endeavours in the same domain and integrated with diverse methodologies, including the ensemble Kalman method [14], field inversion techniques [15], symbolic regression approaches [16–18], and random forest algorithms [19, 20], among others.

Another route pursued in the literature involves the application of ML to the transport equations of turbulence models. Notably, Parish and Duraisamy [21] devised a framework known as field inversion machine learning (FIML). Within this paradigm, the transport equations are adjusted by multiplication with a spatially variant factor inferred through field inversion techniques, while a machine learning model is utilised to deduce regression functions characterising this factor based on selected input features. This methodology has been successfully implemented across various flow regimes, including turbomachinery flows [22], aerofoil aerodynamics [23], boundary layer transition processes [24] and three-dimensional separation flows [25]. Although FIML delivers commendably accurate mean flow fields, it may occasionally produce physically unreasonable Reynolds stresses. Moreover, Zhu et al. [26] trained a radial basis function neural network model to replace the conventional Spalart-Allmaras model equations. Concurrently, Zhang et al. [27] constructed a novel source term within the ϵ equation of the $k - \epsilon$ model through deep learning techniques, thereby significantly reducing the overall field error in simulation outputs.

A more cost-effective strategy centers on optimising the turbulence viscosity. Wu et al. [19, 28] proposed decomposing the anisotropy stress tensor into linear and nonlinear components and computing the optimal eddy viscosity via pointwise least-square approximation. This approach has gained traction among numerous researchers [29–31] and has been empirically shown to enhance the conditioning of RANS equations. More recently, Sun et al. [32] developed an eddy viscosity model, trained on an aerofoil turbulence dataset, which was subsequently tested under high Reynolds number conditions, demonstrating promising results. Liu et al. [29] have introduced an iterative machine learning framework for turbulence modeling, wherein a neural network is trained solely on DNS data of equilibrium channel flow [33, 34] to emulate the eddy viscosity computation. This novel approach gives rise to a data-driven turbulence model, denoted as ML-RANS EQ, which demonstrates enhanced performance over the baseline model in scenarios involving separated flow phenomena, such as the two-dimensional periodic hill [29] and the three-dimensional turbine cascade [35]. Acknowledging its potential for generalisability, we adopt this framework in the present study, with a introduction provided in sec:theory-1.

Based on Spalart's statement [9], turbulence models fail to provide precise estimates of turbulence variables. Our previous work [35] illustrates this point, showing that in RANS simulations, the values of k and ω can significantly differ from reality. In the same work [35], we classified the computation methods for turbulence variables in the training phase of machine learning into two categories. One category (M1) involves directly calculating the high-fidelity turbulence variables according to their original physical definitions. The other category (M2) involves solving the transport equations of the turbulence model based on the frozen mean flow field of high-fidelity data. In the prediction phase, all the input features of neural network are constructed by performing the baseline RANS solver. As a result, the M1 method establishes the consistency of the ML model with the DNS field, but does not guarantee consistency with the RANS or LES environment [36], whereas the M2 method ensures the opposite. As pointed out by Wu et al. [28], even small discrepancies in the Reynolds stress tensor lead to dramatic high error in the mean flow field due to the ill-conditioning nature of RANS equations. Consequnetly, earlier works noted that ML models based on the M1 method often give poor predictions of the mean flow field in a-posteriori tests [37–39]. Since 1998, Parneix et al. [40] introduced the M2 method specifically designed to process and derive turbulence variables from DNS datasets. Amidst the burgeoning field of ML-informed turbulence modeling, several scholars [29, 37, 41, 42] have advocated for or applied the M2 method to produce turbulence variables critical for machine learning model training. Our preceding research [35] compared the turbulence variables based on M1 method and M2 method, and evidenced M2 method's merits in terms of improved accuracy, leading to its adoption in this investigation.

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

The aforementioned quantitative investigation by Fang et al. [7] has evidenced non-equilibrium phenomena within the regions of corner separation, wake, and boundary layers in compressors. To enhance the precision of simulating such corner separation flows, both velocity helicity reflecting nonlinear turbulence energy backscatter processes [43, 44] and kinematic vorticity accounting for fluid rotation and deformation [45] have been integrated into the transport equations of conventional turbulence models. Sun [46] showed that employing the SST model augmented with velocity helicity correction, together with a good grid, effectively captures compressor flow characteristics with reasonable fidelity, thereby yielding credible predictions of compressor performance and stage matching. Moreover, Spalart et al. [9] highlighted that certain canonical flows deviate from the notion of equilibrium conditions. Flows such as normally strained flows, boundary layers experiencing adverse or favourable pressure gradients, steady and unsteady backward-facing step flows, amongst others, have been utilised by Klein et al. [47] to assess turbulence models' performance under basic equilibrium regimes. Notably, Liu et al. [48] scrutinised and characterised the non-equilibrium turbulence zone within a spatially transitional channel flow. Xiao et al. [49] contend that flows over periodically varying hill slopes exemplify strong non-local and non-equilibrium effects, thus conducting direct numerical simulations to establish a database for data-driven turbulence modeling. In complement to embedding non-equilibrium effects within the turbulence modeling framework, as exemplified by Wu and Zhang's work [42], the customisation of training datasets is pivotal to enhancing the efficacy of data-driven turbulence models [13]. For example, incorporating diverse geometries facilitates the generalisability of machine learning (ML)-based turbulence models [16, 50]. In this paper, our objective is to engage a training dataset including non-equlibrium flows in data-driven turbulence modeling to refine the accuracy of turbulence models applied in turbomachinery contexts.

The remainder of the paper is organised as follows. In sec:theory, we will recall briefly the theoretical basis of the ML-RANS framework [29, 35] and non-equilibrium flows. sec:modeling contains details of a new training set and a brief introduction to the training process. The new ML-RANS model has been applied to some in-house cascade test configurations with the previous ML-RANS model and the baseline RANS model. Comparing their simulation results with experimental data, an *a-posteriori* evaluation of the new ML-RANS model will be presented in sec:application. sec:conclusion is our conclusion and perspectives.

2.0 Theoretical basis

Within this section, we initially present the iterative machine-learning framework for RANS turbulence modeling, herein referred to as ML-RANS, as proposed by Liu et al. previously [29, 51, 52]. This framework aims at replacing the conventional calculation formula for turbulence viscosity in the baseline $k-\omega$ SST model using a neural network. Note that all theories and calculations, including *a-priori* and *a-posteriori* tests, are 3D. However, statistical average in homogeneous directions can reduce the data dimensions for training and turbulence modeling. Conclusively, we review some research on non-equilibrium turbulence that motivate the incorporation of effects pertaining to three flow regimes within the ML-RANS framework.

2.1 ML-RANS framework

The ML-RANS framework consists of two independent phases: training phase and prediction. Throughout the ML-RANS modeling process, neural networks, input features and target variables play integral roles. Liu et al. [29] employed the TensorFlow platform [53] to construct an artificial neural network. In the training phase, M2 method described in 1 is first adopted for the construction of machine learning dataset. The machine learning regression algorithms, along with prescribed input features and target variables, are used to determine the weights and biases of the neural network, resulting in a machine learning model. During the prediction phase, this machine learning model is coupled with a baseline turbulence model. It first computes the input features using numerical solutions of both transport

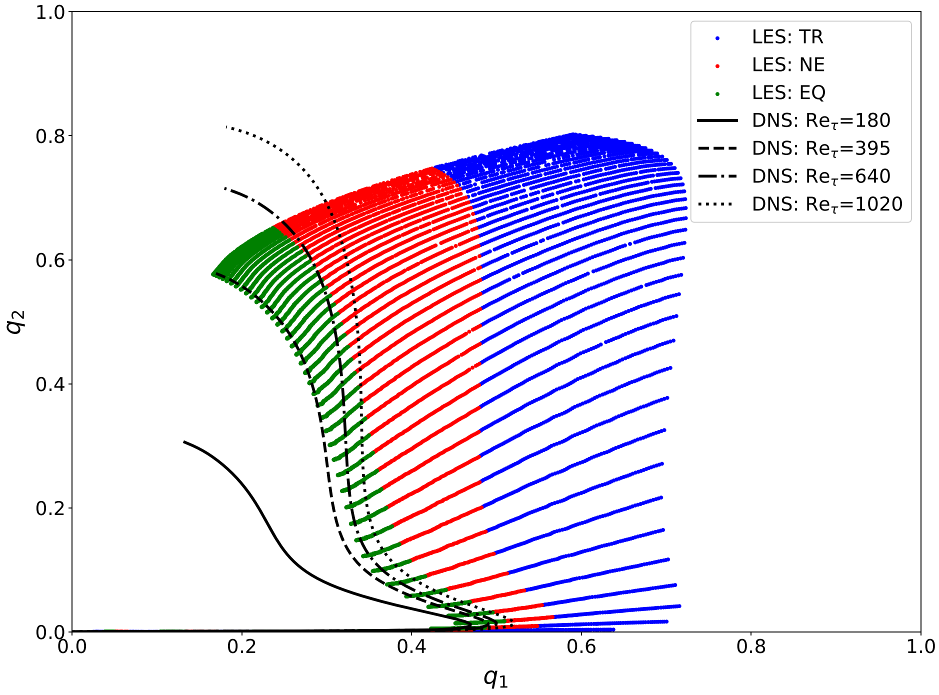

Figure 1. *Comparisons of* (q_1, q_2) *in the transitional channel flow in transition region (TR), non-equilibrium region (NE) and equilibrium region (EQ) respectively [35].*

equations in the baseline turbulence model and RANS equations. The comparison of turbulent quantities (k, ω, ...) obtained with DNS/LES with those of RANS formulation can refer to previous studies, for example Figs. 1 and 2 in Ref. [35]. Subsequently, these input features are mapped onto the target variable to generate turbulence viscosity. This framework exhibits two notable attributes: consistency in the computation of input features across both phases and the evolution of the turbulence viscosity within the prediction phase.

In the ML-RANS framework, the target variable refers to the object of turbulence modeling, which involves leveraging machine learning techniques to repesent Reynolds stresses, either as mathematical expressions or through artificial neural networks, derived from high-fidelity flow data. However, certain researchers [54, 55] have observed discrepancies between the resolved mean velocity when employing the Reynolds stresses from high-fidelity data to close RANS equations and the mean velocity in the original high-fidelity data. Wu et al. [28] highlighted that the RANS equations coupled with Reynolds stress closure models could be ill-conditioned, where minor a-priori errors in modeled terms, such as the Reynolds stresses, can escalate into significant a-posteriori errors during the solution of the RANS equations. To tackle this issue, Wu et al. [28] partitioned the Reynolds stresses into an implicitly treatment (or linear) component involving ν_t and a nonlinear part, i.e.,

$$\boldsymbol{a} = \boldsymbol{\tau} - \frac{1}{3}tr(\boldsymbol{\tau})\boldsymbol{I} = 2\nu_t \boldsymbol{S} + \boldsymbol{a}_\perp. \tag{1}$$

The linear portion corresponds to one aspect of Pope's proposed mathematical representation of the Reynolds stress [12], which adheres to the Boussinesq eddy viscosity hypothesis. From the high-fidelity dataset, the Reynolds stress tensor $\tau_{\boldsymbol{h}}$ and the mean shear stress tensor $\boldsymbol{S_h}$ are extracted to calculate ν_t by minimising the nonlinear part, resulting in

$$\nu_t = \left| \frac{\tau_{h_{ij}} S_{h_{ij}}}{2 S_{h_{ij}} S_{h_{ij}}} \right|. \tag{2}$$

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

Table 1. *The input features and the target variable for the practical ML-RANS model [51, 52]*

Variable	Description	Definition	Normalisation	Actual form
q_1	Turbulence intensity	k	$\frac{1}{2}U_iU_i$	$\frac{25k}{25k+0.5U_iU_i}$
q_2	Estimated eddy viscosity	$\frac{k}{\omega}$	ν	$\frac{k}{k+50\nu\omega}$
ν_t^0	Authentic eddy viscosity	ν_t	$\frac{k}{\omega}$	$\frac{5\nu_t}{5\nu_t+3\frac{k}{\omega}}$

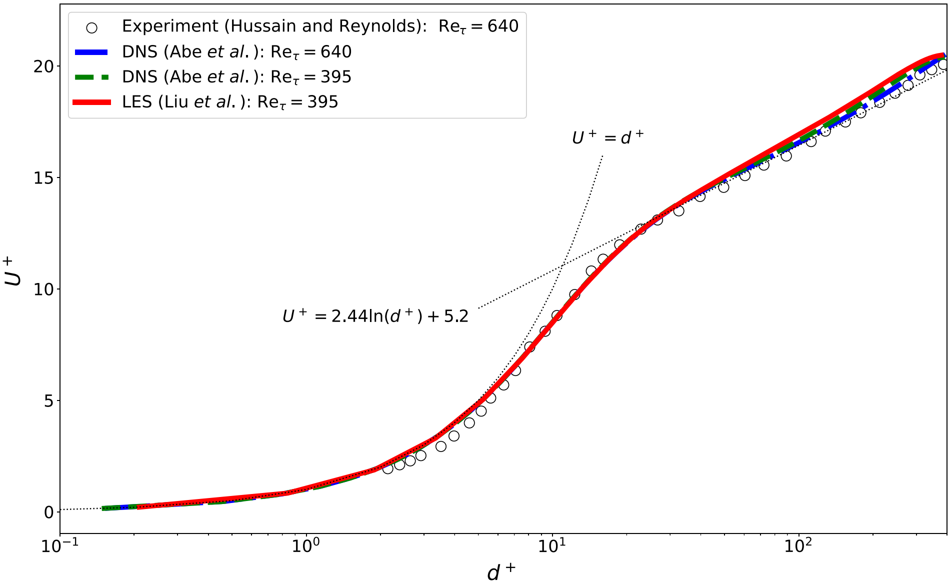

Figure 2. *Mean velocity profiles by use of different high-fidelity methods.*

Their analysis showed that enhancing the implicit treatment of the Reynolds stress, for instance, by adding an increment to the optimal ν_t while correspondingly subtracting from the nonlinear part, can further improve the conditioning of the model, thereby reducing the sensitivity of the RANS equation solution to the unclosed terms [28]. In earlier work, Wu et al. [19] had independently trained machine learning models for the linear and nonlinear components of the Reynolds stress, culminating in a well-conditioned and accurate Reynolds stress model. Liu et al. [29, 51, 52], building upon this, took a step further by focusing solely on the turbulence viscosity as the target variable, bypassing the nonlinear part of the Reynolds stress altogether. To enhance the effectiveness of the machine learning regression, they first nondimensionalised the turbulence viscosity using the traditional definition k/ω from the $k-\omega$ model [3, 8], followed by normalisation to arrive at the actual form of the target variable presented in Table 1.

As for the input features, Liu et al. [29] initially derived from Pope's analysis [12] an essential input feature (i.e., $k/\nu\omega$) related to the target variable. Based on empirical evidence, Liu et al. bypassed tensor invariances in the input features to stabilise simulations and introduced five supplementary input features that included turbulence intensity, wall distance, cross-diffusion of k and ω, along with two SST model variables specifically capturing properties of the viscous sublayer and the turbulent region [29]. In subsequent research [51, 52], they streamlined their approach by retaining only the first two input features and subjected them to normalisation based on their initial design, as detailed in Table 1. Notably,

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

our previous study [35] has demonstrated the ability to identify distinct regimes of turbulence development exclusively using these two dimensionless and normalised input features (q_1, q_2), as illustrated in Fig. 1. In previous practices, DNS datasets [33, 34] and RANS calculations [29, 51, 52] were used, respectively.

2.2 Non-equilibrium flows

The property of multi-scale energy transfer is a good starting point for understanding turbulence. The well-known Richardson-Kolmogorov energy cascade indicates an equilibrium dissipation law, that can be depicted as a balance of high Reynolds turbulence between the forward energy transfer and the dissipation.

$$\epsilon = \frac{C_\epsilon \mathcal{U}^3}{\mathcal{L}}. \tag{3}$$

where $\mathcal{U}^2$ is turbulent kinetic energy, $\mathcal{L}$ is the integral length scale and the dissipation coefficient C_ϵ is constant. In fact, in real flows there exist various dissipation laws, including the equilibrium or quasi-equilibrium laws and the non-equilibrium laws, as Vassilicos [56] introduced. It was discovered that there exists a non-equilibrium dissipation law between C_ϵ and Taylor-scale Reynolds number Re_λ, which is approximately $C_\epsilon \sim \mathrm{Re}_\lambda^{-1}$, or rigorously by theoretical analysis [57],

$$C_\epsilon \sim \mathrm{Re}_\lambda^{-\frac{15}{14}}. \tag{4}$$

It denotes the imbalance between energy production and dissipation, which mainly exists with a spatially inhomogeneous geometry, even when the flow is a fully developed turbulence. The non-equilibrium canonical flows discussed in sec:intro largely conform to this dissipation rate. Researchers have actively explored the integration of non-equilibrium phenomena within turbulence models by leveraging the ratio between the production and dissipation of turbulence kinetic energy [42, 58]. Recently, it was found that for flows whose transfer spectrum is out-of-equilibrium, there also exists another non-equilibrium dissipation law

$$C_\epsilon \sim \mathrm{Re}_\lambda^{-2}, \tag{5}$$

which is usually rather evident in short-time evolution by comparing to the $-15/14$ scaling [59–62] and is closely related to the self-organisation procedure in a flow that is (temporally or spacially) not fully developed. During the evolution of turbulent flows, this novel scaling law typically implies the presence of turbulence energy backscatter (i.e., the transfer of energy from smaller to larger scales). Despite their appearance prior to the establishment of such a scaling law, the conventional turbulence modeling efforts [43, 44] described in sec:intro, aiming at enhancing the Spalart-Allmaras and SST models by incorporating helicity, can be regarded as early attempts to account for this out-of-equilibrium in third-order statistical quantities. However, until now, there has not yet been a data-driven turbulence modeling that specifically addresses the -2 scaling behaviour associated with energy backscatter phenomena.

In previous studies we showed that a spatially transitional channel flow can perfectly involve all the known states of energy transfer of turbulence evolution, including both the $-15/14$ and -2 scalings of non-equilibrium flows [59]. On another hand, we also illustrated that the flows in turbomachinery are almost non-equilibrium everywhere [7]. In the present paper we then use this dataset, aiming at involving various flows in the training procedure, and hoping to improve the performance of turbulence models in turbomachinery.

3.0 Turbulence modeling

According to earlier research [16, 50], the test flow's shape, pressure gradient, and other components should be present in the training flow as well. Although its training set was created using the canonical channel flow, the ML-RANS EQ model performed slightly better than its baseline model in the complex

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

Table 2. *Information of LES on the spatially transitional channel flow [48]*

Variable	Value	Description
Re_m	7,000	$\mathrm{Re}_m = U_m h/\nu$, with U_m bulk mean velocity
Re_τ	395	$\mathrm{Re}_\tau = u_\tau h/\nu$ where u_τ friction velocity
Δx^+	77	Homogeneous streamwise mesh size
Δy^+	0.2	First-level normal mesh size from the wall
Δz^+	38	Homogeneous spanwise mesh size

geometry (see more details in the papers of Liu et al. [29] and Fang et al. [35]), indicating good generalisation of the present ML-RANS framework. Using a spatially transitional channel flow database [48] with transition region (TR), non-equilibrium region (NE), and equilibrium region (EQ), we discovered three essential constraints of this training set (i.e., sampling in one dimension, geometry of the straight wall, and turbulence in equilibrium) in the previous study [35]. Moreover, we specified the issue of expansibility from EQ to TR/NE in both *a-priori* and *a-posteriori* analysis. In the present study, we create the ML-RANS TR-NE-EQ model by fusing the aforementioned transitional flow database [48] with the present ML-RANS framework [51, 52].

3.1 Training set

A training set based on the spatially transitional channel flow [48] will be introduced and evaluated in this subsection. The channel has a $64\pi h \times 2h \times 2\pi h$ domain with h channel half-width with a spatial resolution $1,024 \times 96 \times 64$ in x, y, z directions, including a PSB (periodic suction and blowing disturbance) region in the streamwise interval $L_x \in [\pi h, 2\pi h]$ and a fringe region [63] in the streamwise interval $L_x \in [56\pi h, 64\pi h]$. The existence of two regions contributes to the generation of the spatial transition. The commonly used dynamic Smagorinsky model was selected as the subgrid-scale model for LES of the channel flow. No-slip boundary conditions are imposed at wall (excluding the PSB region), whereas periodic boundary conditions are utilised along the spanwise direction with the LES. Laminar flow distribution is set at inlet and zero-gradient is set at outlet. Furthermore, the grid spacings and Reynolds numbers are documented in Table 2 for more details. More details refer to Refs [48, 64, 65].

The mean flow profile deviates from its parabolic shape within the range $L_x > 12\pi$ with the maximum value of the mean velocity reaching approximately 1.14 at $L_x \approx 23.4\pi$. This confirms the typical morphology of a turbulent velocity profile beyond $L_x > 23.4\pi$, which is in good agreement with DNS results from Moser et al. [34] and those from Abe et al. [33] at the same friction Reynolds number, Re_τ. However, the flow had not yet been fully developed until nearly $L_x = 31\pi$, at which the profile of the total shear stress was a horizontal straight line. In the interval, $L_x \in [23.4\pi h, 31\pi h]$, the dissipation coefficient C_ϵ increased weakly to constant, and the skewness of longitudinal velocity derivative S_k decreased slightly to constant. In addition, C_ϵ was inversely proportional to the integral scale Reynolds number at a low Reynolds number. The phenomena were also observed in the non-equilibrium study of grid turbulence [66]. Consequently, the segment $[23.4\pi h, 31\pi h]$ along the streamwise direction was classified as NE, while $L_x < 23.4\pi$ was characterised as TR and $L_x > 31\pi$ was considered as EQ. The near-wall mean velocity profile at EQ phase is compared with DNS results of Abe et al. [33], as well as the experiment of Hussain and Reynolds [67]. Pope reviewed the classical wall law, including the linear relation $U^+ = d^+$ in the viscous sublayer ($d^+ < 5$) and the logarithmic law

$$U^+ = \frac{1}{\kappa}\ln(d^+) + C, \tag{6}$$

due to von Karman for $d^+ > 30$, where $\kappa \approx 0.41$ represents the Karman constant and C was taken as 5.2 despite dependance on Reynolds number [33, 68]. As shown in Fig. 2, all three simulations and the experiment agree with the linear relation in the viscous sublayer. In the logarithmic region, the DNS result at $\mathrm{Re}_\tau = 640$ closely matches both the theoretical formula and corresponding experimental

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

Table 3. Hyperparameters of the neural network for the ML-RANS TR-NE-EQ model

Hyperparameter	Value/Description
Number of hidden layers	3
Number of nodes per layer	24
Activation function	tanh
Optimiser	Adam optimiser
Learning rate	0.002
L1 regularisation coefficient	10^{-5}

findings at the same Reynolds number, thereby substantiating the accuracy of the DNS programme employed by Abe *et al.* At $\mathrm{Re}_\tau = 395$, the LES result aligns closely with the DNS result, which indicates that the LES result remains highly credible. For in-depth information, see the paper published by Liu et al. [48].

In TR, positive S_k denotes the energy backscatter signal [69], which also occurs after altering the sign of velocities in time-reversed turbulence [59] where the large-scale dynamics cannot feel the effect of energy backscatter as concurrently as small-scale dynamics does. This non-equilibrium behaviour, characterised by a -2 scaling law, continues to impact the channel flow until the large-scale structures within NE are notably affected. Subsequently, the flows begins to conform to $-15/14$ scaling law. The flow eventually transforms into equilibrium turbulence in EQ after the rapid and slow non-equilibrium procedures. The physical information of transition, non-equilibrium and equilibrium turbulence is retrieved from the flow data in the two-dimensional mid-span section of the spatially transitional channel flow in the current study.

3.2 Modeling process

During the training phase, Google's TensorFlow [53] was adopted as the platform for machine learning. The training dataset was separated randomly into the training part and the validation one in a ratio of four to one. According to training effect, the hyperparameters in Table 3 were adopted to construct a feedforward artificial neural network (i.e., multi-layer perception). For sake of prevention from overfitting [70], L1 regularisation was adopted to minimise the weight of each layer in the neural network. The total loss was thus defined as

$$\mathrm{Loss} = \frac{1}{N}\sum_{i=1}^{N}\left(\widehat{\nu_t^{0,i}} - \nu_t^{0,i}\right)^2 + \lambda \sum_{i,j}^{m,n} |w_{i,j}|. \tag{7}$$

where $\nu_t^{0,i}$ is the target variable in the training data, $\widehat{\nu_t^{0,i}}$ is the value predicted by the neural network, and $w_{i,j}$ is the weight in the neural network. λ represents the L1 regularisation coefficient. N is the number of training data, m and n are respectively the numbers of layers and nodes in the neural network. The first part of the loss is the mean square error (MSE) from the training data, and the second part is the L1 regularisation penalty. As seen in Fig. 3, the evolution of errors indicates that our model is rapidly established once the training process starts. To estimate visually the model result, we plot the profiles of the target variable ν_t^0 at different streamwise locations in Fig. 4, and the relation between the target variable and both input features is shown in Fig. 5. It can be observed that the training data and ML predictions are all in good agreement.

Following training, a comprehensive data-driven turbulence model is created by combining the machine learning model with the predictor library [51]. Figure 6 illustrates that the target variable ν_t^0 is pretty independent of the input features where q_1 is large for the ML-RANS EQ model, while it decreases obviously as q_2 increases in the same region for the ML-RANS TR-NE-EQ model.

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

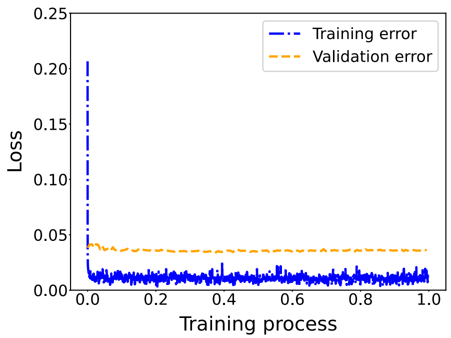

Figure 3. *Evolution of loss in the training process. The x-axis represents the ratio of current training step and total steps.*

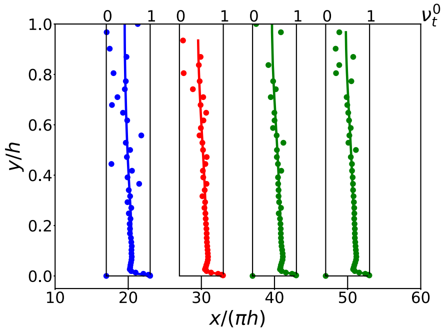

Figure 4. *Profiles of the target variable ν_t^0 at $x/(\pi h) = 20, 30, 40, 50$ in TR (blue), NE (red) and EQ (green) regions: comparison of ML predictions (lines) and a snapshot of training dataset (points).*

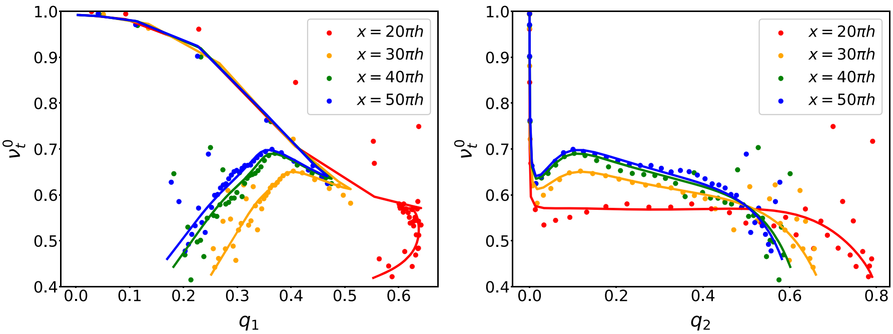

Figure 5. *Relation between the target variable ν_t^0 and two input features (q_1, q_2) at four different streamwise locations: comparison of ML predictions (line) and a snapshot of training dataset (point).*

3.3 A-posteriori test

Having implemented the novel ML-RANS TR-NE-EQ model in OpenFOAM, we applied it into the spatially transitional channel flow used during training, comparing its performance against the earlier ML-RANS EQ model and the baseline model. Inspired by Zhao's doctoral dissertation [71] where several turbulence models were utilised for RANS simulations of this transitional channel flow, this study followed suit by removing the fringe area close to the outlet. A computationally economical mesh configuration of 440 × 50 × 32 was established across the computational domain of $55\pi h \times 2h \times 2\pi h$,

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

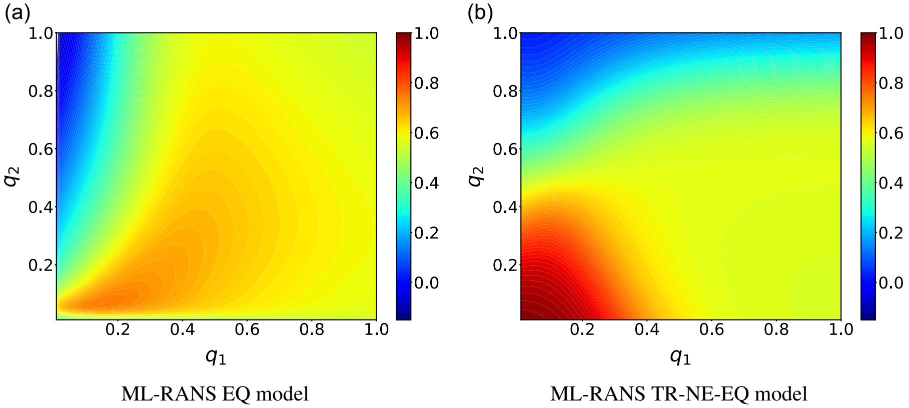

Figure 6. *Mapping from (q_1, q_2) to ν_t^0 in two ML-RANS models.*

ensuring effective refinement near the top and bottom walls, particularly with the first cell layer meeting the requirement of $d^+ < 1$.

Consistent with LES conducted by Liu et al. [48], the transition is facilitated by the PSB region near the inlet; thus, for the wall segment where the streamwise coordinates L_x lie within $[\pi h, 2\pi h]$, we also designed the PSB boundary condition for the velocity field while imposing a zero-gradient condition for other physical quantities. For the remaining walls, we set the no-slip condition with low-Reynolds number corrections. In the meantime, laminar flow inlet and zero-gradient outlet were configured accordingly. Considering that the velocity boundary values vary over time in the PSB region, we conducted unsteady RANS simulations using the baseline turbulence model, ML-RANS EQ model, and the ML-RANS TR-NE-EQ model, each assisted by the PISO algorithm. To maintain numerical stability, the time step was constrained to ensure a Courant number below 1.

For incompressible flows, the bulk mean velocity remains constant, and the transition from laminar to turbulence can be reflected through variations in the maximum value of the mean velocity. Upon comparing the simulation outputs generated by the three turbulence models against the maximum value of the mean velocity from the LES data, it is observed that the ML-RANS TR-NE-EQ model delivers superior performance among three RANS turbulence models, as shown in Fig. 7. Firstly, both ML-RANS models effectively avoid the occurrence of the local velocity peak within the PSB region simulated by the baseline model. Subsequently, following the PSB region, the maximum value of the mean velocity computed by all three models begin to decline, which indicates that the flow starts to deviate from the laminar. However, the ML-RANS TR-NE-EQ model exhibits a notably delayed initiation of flow transition, as its calculated mean velocity maximum doesn't experience a sharp drop until reaching $L_x = 10\pi h$, aligning closely with the LES-derived transitional point $L_x = 12\pi h$. The RANS simulation adopting the ML-RANS TR-NE-EQ model presents remarkable agreement with the LES results in the NE region. Within the EQ region where the large-scale non-equilibrium is virtually absent, the maximum value of the mean velocity calculated by all four numerical simulations converge. Among three RANS models, the ML-RANS TR-NE-EQ model's prediction shows a closer resemblance to the LES result than that of the baseline model. In summary, the ML-RANS TR-NE-EQ model provides the best simulation of the transitional channel flow among the three RANS turbulence models.

4.0 Application in the turbine cascades

We apply two practical ML-RANS models and their baseline turbulence model (i.e., $k - \omega$ SST model) to some turbine cascade flow cases which had been the subject of in-house experimental measurements, respectively. Every cascade in the experiment consists of the same eight blades, which stand in for the mid-span region of a low-pressure turbine rotor blade. Several cylindrical bars are fitted as wake

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

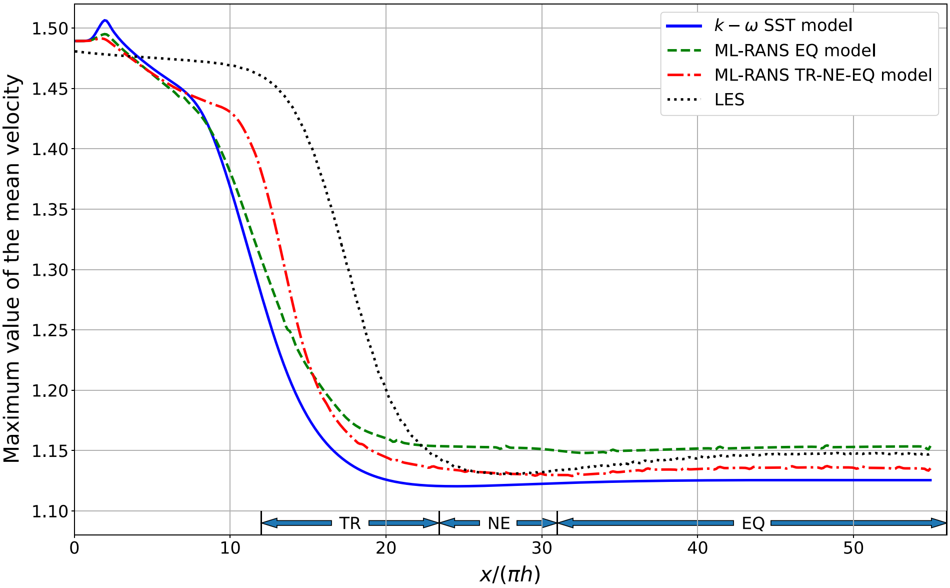

Figure 7. *Evolution of maximum value of the mean velocity along streamwise direction.*

generators in front of these blades. The total pressure, total temperature and wall static pressure in the cascade inlet are all monitored for the flow conditions. Certain aerodynamic parameters are monitored using an L-shaped five-hole probe and wall static pressure holes at a distance of 0.4 axial chord (l_{ax}) from the blade's trailing edge (e.g., total pressure loss coefficient and Mach number). In the present study, we analyse 15 cases with five various blade profiles (denoted as Blade 1, Blade 2, . . ., respectively) and three different angles of attack (0°, −15°, and 15°, respectively), under similar outlet Reynolds number (i.e., 770,000) and outlet Mach number (i.e., 0.75). Figure 8 illustrates a few cascade case configurations by using randomly Blade 2 and zero angle-of-attack as an example. In the figure the star symbols located $0.4l_{ax}$ before the leading edge are marked, corresponding to the observation positions of spanwise near-wall behaviour in simulation, which will be represented later. More experimental details can refer to our previous study [35].

All the computational domains containing five different blades are divided by an H–O–H structured mesh of 193 points (for Blade 3) or 185 points (for other four blades) in the streamwise direction and 84 points in the orthogonal direction, with the first node near the blade satisfying $d^+|_{wall} < 1$. The length of the inlet region is half of the axial chord length, while the outlet region has one axial chord length. Figure 9 shows the mesh about Blade 2. Mesh independency has been accordingly verified to guaranty the rationality of simulation. The $k-\omega$ SST model and two ML-RANS models are applied in the open-source code OpenFOAM. In practice, we set uniform values of total pressure, total temperature and flow angle at the inlet, a static pressure at the outlet, and no-slip condition at the wall with low Reynolds number corrections.

The comparison on near-wall effects between ML-RANS models and the classic $k-\omega$ SST model is shown in Figs. 10 and 11. In front of bars and blades, the endwall of the turbine cascade resembles a channel. The near-wall results observed here are not influenced by the wake. The mean velocity profile in Fig. 10(a) by using ML-RANS TR-NE-EQ model (i.e., $U^+ = 2.46\ln(d^+) + 5.92$) is the closest to the logarithmic law of the wall (see Equation 6), by comparing to the other two curves. In addition, the eddy viscosity shown in Fig. 10(b) also differs. At locations on the blade surface away from the endwalls, two measurement points were selected near the leading edge and trailing edge, respectively, to compare the mean velocity profiles along the blade surface normal direction from different models. Given that molecular viscosity dominates the flow within the viscous sublayer, the RANS equations necessarily

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

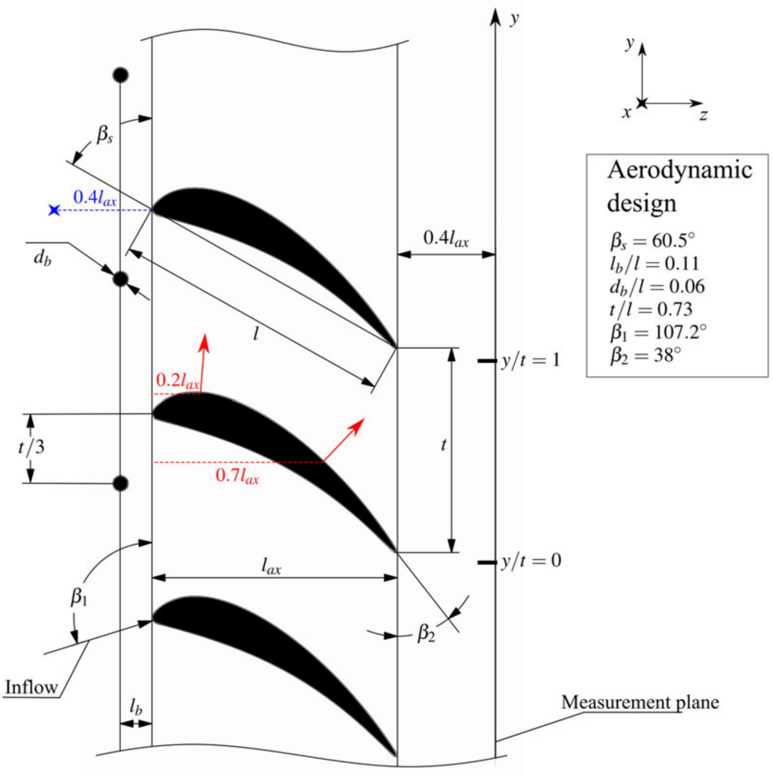

Figure 8. *Sketch of cascade design of LPT: near-wall behaviours in simulations are observed both at the blue star (endwall, measurement along the x axis) and at red arrows (blade).*

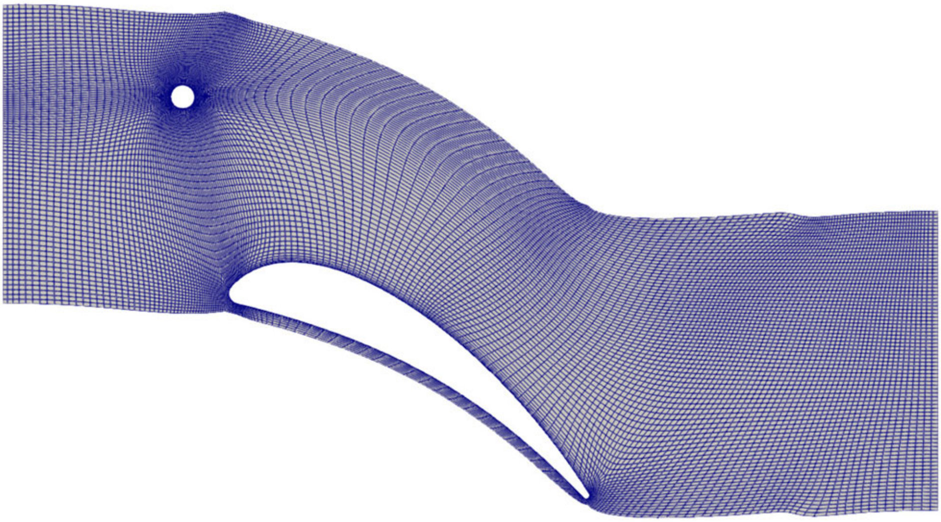

Figure 9. *Mesh of the channel with Blade 2 used in numerical simulations.*

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

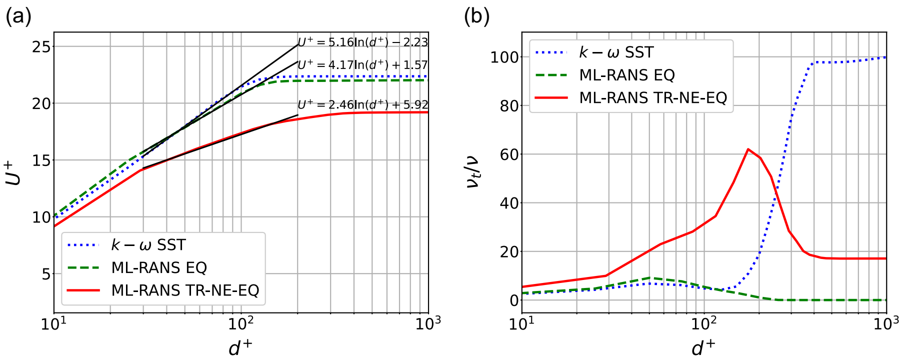

Figure 10. *Comparison on results near endwall among the $k-\omega$ SST model and the two ML-RANS approaches, at the location marked as the blue star in Fig. 8: (a) Mean velocity, (b) Eddy viscosity.*

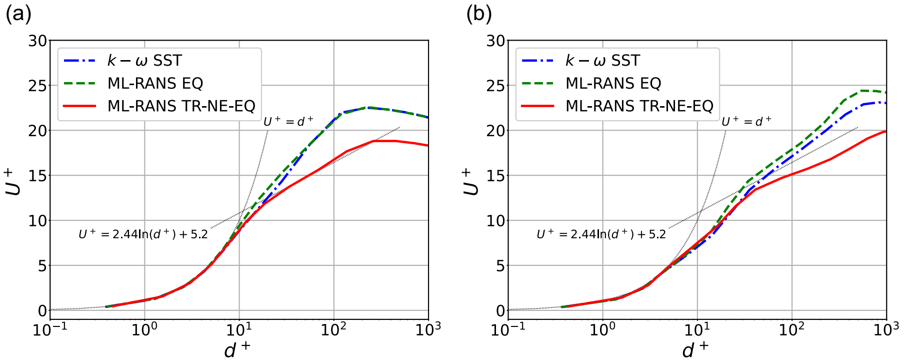

Figure 11. *Comparison on mean velocities near blade among the $k-\omega$ SST model and the two ML-RANS approaches, at the locations marked as the arrows with different distances from the leading edge in Fig. 8: (a) $0.2l_{ax}$, (b) $0.7l_{ax}$.*

simplify to the linear relation, as calculated by all three models. In the logarithmic region, however, there were slight differences in performance among the three models near both the leading and trailing edges of the blade. Similar to Fig. 10, the $k-\omega$ SST model and the ML-RANS EQ model provided distinct logarithmic laws that deviated from the von Kármán's formula. In contrast, the ML-RANS TR-NE-EQ model approximated the standard logarithmic law closely near the leading edge but showed a obvious difference only in the constant term C of the Equation (6) near the trailing edge.

A-posteriori results in Fig. 12 show that both the two ML-RANS approaches show good predictions for the static pressure coefficient (Equation 8) around the blade. In the previous study [35], we pointed out that the blade downstream without separation is almost EQ according to Fang et al. [7] and that the bar wake with obvious separation is mostly NE/TR according to Tao and Zhou [72]. We take the case of Blade 2 and zero angle-of-attack as an example. We comment that this choice does not mean that Blade 2 is the best case. The global performance of all cases will be shown later. The Mach number calculated by the ML-RANS TR-NE-EQ model in Fig. 13 is the smallest among three simulations both in the bar wake and the blade downstream, indicating that its bar wake is the longest and that its blade downstream is the most obvious, while the ML-RANS EQ model yields the similar effect to those of the $k-\omega$ SST model. Eddy viscosity is crucial for solving the Reynolds equation in the eddy viscosity model. When comparing the scalar in the bar wake, it can be observed that the ML-RANS TR-NE-EQ model exports far less eddy viscosity than the ML-RANS EQ model and their baseline model (see Fig. 14).

Furthermore, the ML-RANS EQ and the baseline model both depend on equilibrium turbulence and two ML-RANS models were born in the same data-driven turbulence modeling framework [29],

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

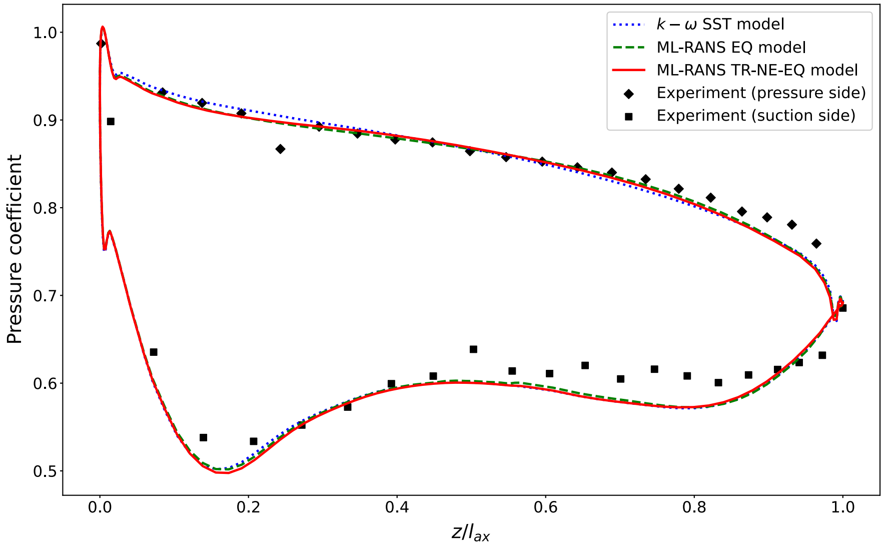

Figure 12. *Comparison on* a-posteriori *results on static pressure coefficient around the blade.*

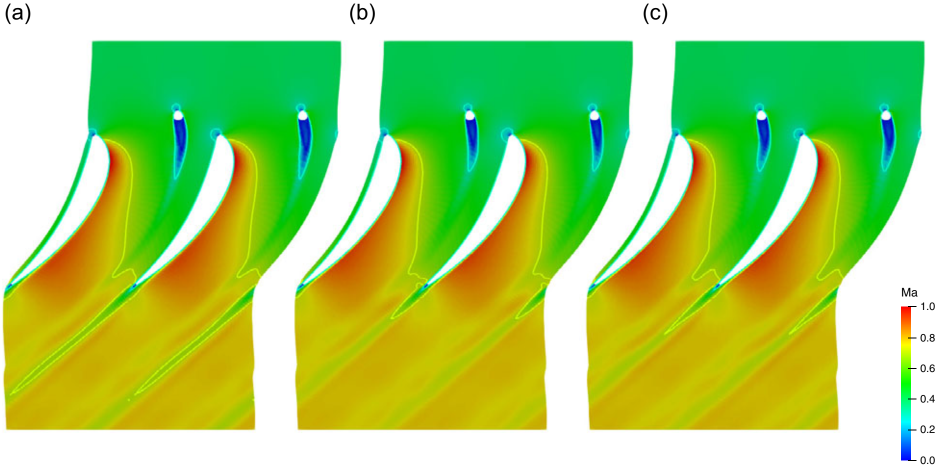

Figure 13. *Mach number (Ma) calculated by (a) ML-RANS TR-NE-EQ model, (b) $k-\omega$ SST model and (c) ML-RANS EQ model at mid-span section: 0.3 at cyan contour and 0.7 at yellow contour.*

where the machine learning regression mapping partially replaces the physics-informed equation of ν_t. Thus, analysing discrepancies in turbulence viscosity outputs from two ML-RANS models can help us understand influence of non-equilibrium flows on turbulence modeling. One practical strategy involves indicating the TR, NE and EQ regimes, especially the TR/NE zones. Nevertheless, based on prior research [7, 35, 48, 60], the quantitative criterion needs further determination to accurately identify non-equilibrium flow state Inspired by the successful partition of three regions, the (q_1, q_2) diagram is utilised to identify the discrepancy of two ML-RANS models in the bar wake satisfying $\nu_t > 0.0015\text{m}^2/\text{s}$. Two ML-RANS models in Fig. 15 predicate that the closer the position is to the bar, the greater the couple (q_1, q_2), which denotes that the bar wake is located in TR/NE regimes according to Fig. 1. Reviewing Fig. 6, due to diversity of training data, the ML-RANS TR-NE-EQ model predicts more changeful target variable than ML-RANS EQ model in TR/NE regions, which indicates that the new model can deal

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

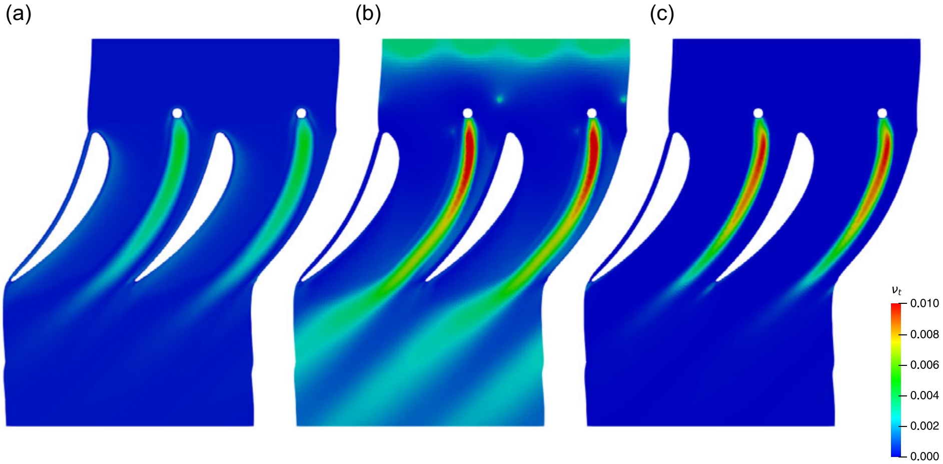

Figure 14. *Eddy viscosity (ν_t) calculated by (a) ML-RANS TR-NE-EQ model, (b) $k-\omega$ SST model and (c) ML-RANS EQ model at at mid-span section.*

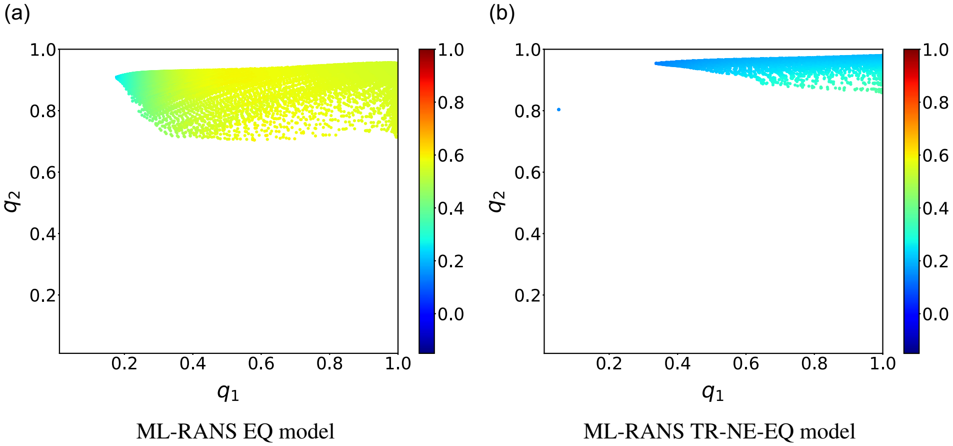

Figure 15. *Mapping from (q_1, q_2) to ν_t^0 of two ML-RANS models in the bar wake ($\nu_t > 0.0015\text{m}^2/\text{s}$).*

with different flow phases without requiring manual adjustments. As a result, the ML-RANS TR-NE-EQ model offers lower turbulence viscosity than two other models in the bar wake. Note that these results are predicted by extrapolation instead of interpolation. Usually the extrapolation of ML is poor, but it is unavoidable when we employ ML-RANS models to real engineering applications, in particular for those research and development projects considering off-design conditions, which has attracted gradually more attentions in previous years. Thus we believe that the present attempt by using extrapolation is important and necessary for real engineering applications.

$$C_{p_{\text{static}}} = \frac{p}{p_{\text{total}}^{\text{inlet}}}, \tag{8}$$

In order to evaluate the ML-RANS TR-NE-EQ model, we compare the simulation results of various turbulence models with the corresponding experimental results in the blade downstream. Figure 16 illustrates the difference in the total pressure loss coefficients (Equation 9). In the five simulation results, the first peak (i.e., $y \approx 0.2t$) in the y axis direction in the figure corresponds to the wake of the bar, whereas the second peak (i.e., $y \approx 0.5t$) corresponds to the wake of the blade. Because of

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

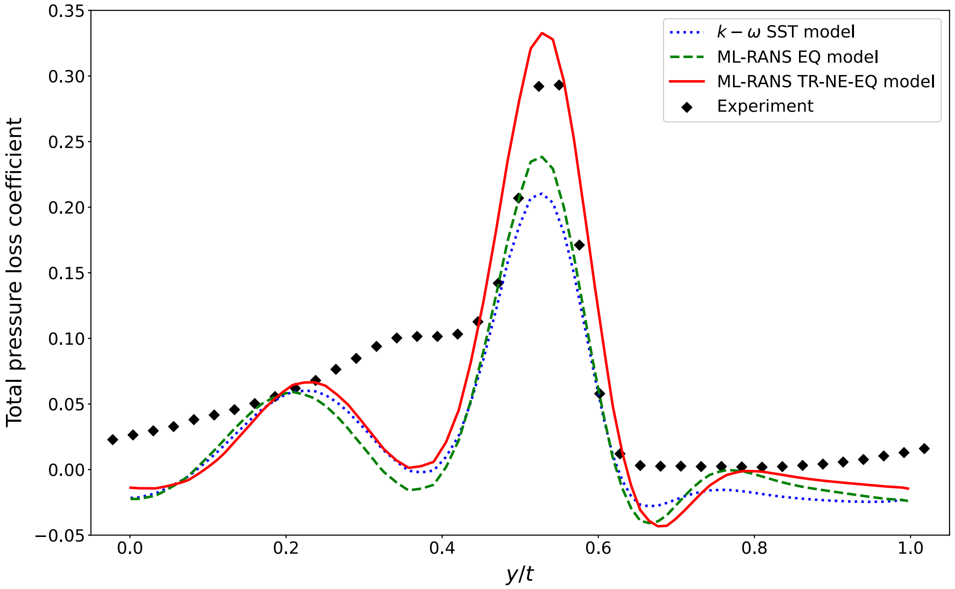

Figure 16. *Total pressure loss coefficient at measurement plane on the blade mid-span section.*

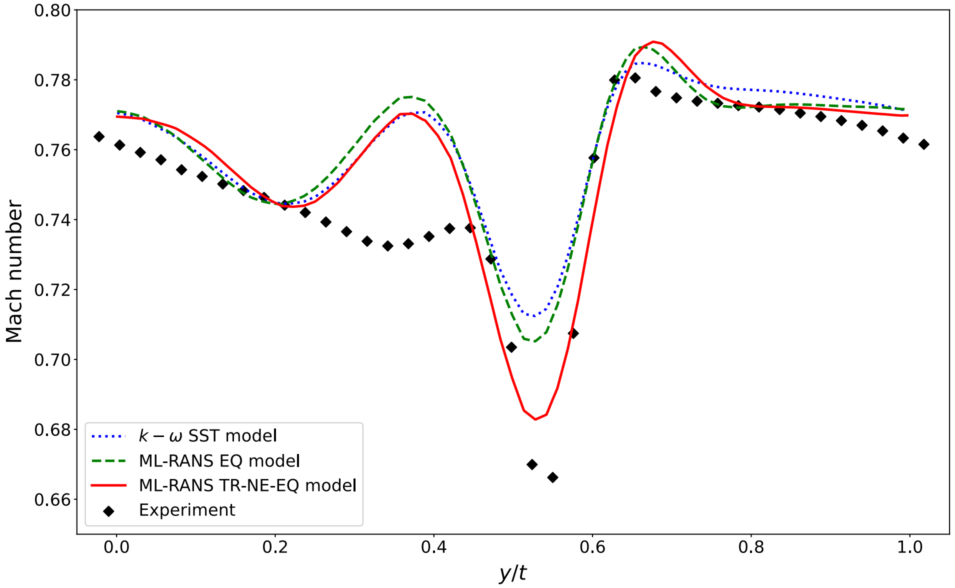

Figure 17. *Mach number at measurement plane on the blade mid-span section.*

the substantial interaction between the bar wake and the blade pressure side boundary layer, the first peak (i.e., $y \approx 0.35t$) in the experiment is closer to the second peak and is less noticeable. Such a phenomenon is captured by neither the conventional nor the ML-RANS model. However, the ML-RANS TR-NE-EQ model predictes rather accurately the total pressure loss in the blade wake, while the ML-RANS EQ model offers a modest improvement. The aforementioned phenomena are also observed in the prediction for the Mach number (see Fig. 17) as long as we pay attention to the valley instead of the peak. For each of the 15 cases, we continue to observe the discrepancy between the experimental result and the simulation results from various turbulence models on the measure plane behind the cascade.

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

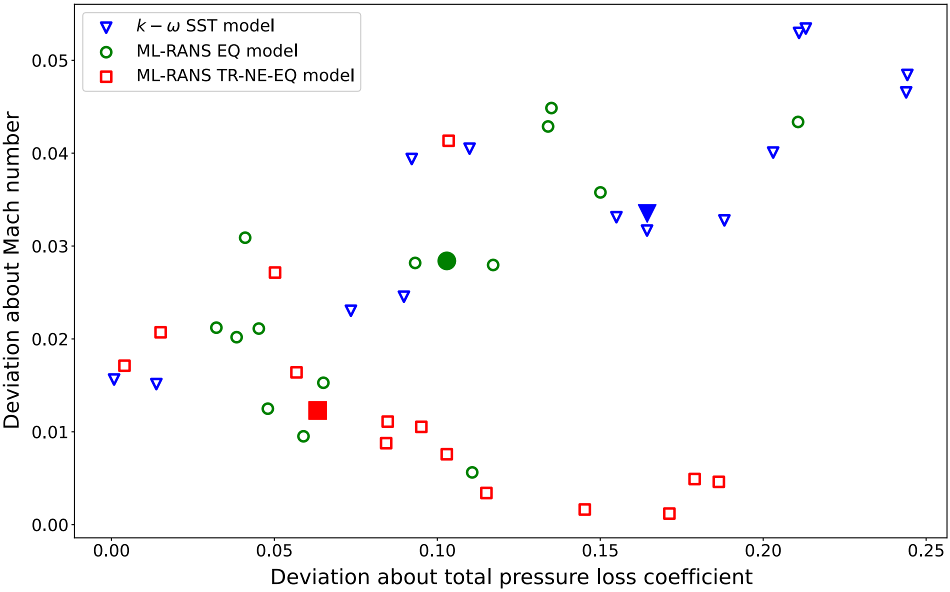

Figure 18. *Deviation of the second peak about the total pressure loss coefficient and the Mach number. The solid and larger symbols represent the result in the selected case (Blade 2 with zero angle-of-attack).*

Equation (10) is utilised to calculate the deviation of the second peak (i.e., the blade wake) about the total pressure loss coefficient and the second valley (i.e., the blade wake) about the Mach number. The ML-RANS TR-NE-EQ model performs better than others, in particular captures well the valley of Mach number in the blade wake (see Fig. 18). The solid and larger symbols represent the result in the selected case, i.e., Blade 2 with zero angle-of-attack, but for other cases the trend is exactly similar: the ML-RANS TR-NE-EQ model performs better than other models, in the sense of a global consideration on both Mach number and pressure loss coefficient.

$$C^{\mathrm{loss}}_{p_{\mathrm{total}}} = \frac{p^{\mathrm{inlet}}_{\mathrm{total}} - p_{\mathrm{total}}}{p_{\mathrm{total}} - p}, \tag{9}$$

$$\mathrm{Deviation} = \frac{|\mathrm{Simulation\ result} - \mathrm{Experimental\ result}|}{|\mathrm{Simulation\ result}| + |\mathrm{Experimental\ result}|}. \tag{10}$$

In consequence, the ML-RANS TR-NE-EQ model cannot only identify the TR/NE in the bar wake with (q_1, q_2) like the ML-RANS EQ model but also transform them to the significantly different eddy viscosity from one calculated by the equilibrium model. The discrepancy is then transferred to the blade wake on account of the interaction between the bar wake and the boundary layer in the blade pressure side. The model improvement is in particular obvious in the region of blade wake by comparing to experiment.

5.0 Conclusions

In previous studies, the ML-RANS framework has presented better prediction for flow statistics such as the total pressure loss coefficient downstream of the blade [29, 35]. However, results were still quite different comparing to experiments, which might stem from the fact that these ML-RANS models were trained by using equilibrium turbulent flow dataset (i.e., ML-RANS EQ model). Considering the non-negligible non-equilibriurm effect of energy transfer in real turbomachinery [7], in the present paper we

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

then aim at improving the ML-RANS models by involving non-equilibrium training dataset (i.e., ML-RANS TR-NE-EQ model). Comparing with ML-RANS EQ model and the baseline RANS model (i.e., $k - \omega$ SST model), we show that ML-RANS TR-NE-EQ model with non-equilibrium features predicts better statistics, and shows good agreement with experiments.

Acknowledgements. We would like to express our gratitude to O. Jung and J. Pavillet for their supportive guide to the project. This work is supported by National Natural Science Foundation of China (Project Approval No. 12372214, U2341231), BSS Project (PO_20201112_BUAA PhD Project) and the Science Center for Gas Turbine Project (Project No. P2022-C-III-001-001).

References

[1] Bardina, J., Huang, P. and Coakley, T. Turbulence modeling validation, *28th Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, June 1997.

[2] Launder, B.E. and Sharma, B.I. Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc, *Lett. Heat Mass Transfer*, 1974, **1,** (2), pp 131–137.

[3] Wilcox, D.C. Reassessment of the scale-determining equation for advanced turbulence models, *AIAA J.*, 1988, **26,** (11), pp 1299–1310.

[4] Menter, F.R. Zonal two equation k-w turbulence models for aerodynamic flows, 23rd Fluid Dynamics, Plasmadynamics and Lasers Conference, American Institute of Aeronautics and Astronautics, 1993.

[5] Menter, F.R. Two-equation eddy-viscosity turbulence models for engineering applications, *AIAA J.*, 1994, **32,** (8), pp 1598–1605.

[6] Chipongo, K., Khiadani, M. and Sookhak Lari, K. Comparison and verification of turbulence Reynolds-averaged Navier–Stokes closures to model spatially varied flows, *Sci. Rep.*, 2020, **10,** (1), p 19059.

[7] Fang, L., Zhao, H.K., Lu, L.P., Liu, Y.W. and Yan, H. Quantitative description of non-equilibrium turbulent phenomena in compressors, *Aerospace Sci. Technol.*, 2017, **71**, pp 78–89.

[8] Wilcox, D.C. *Turbulence Modeling for CFD*, 3rd ed, DCW Industries, 2006, La Canada, CA.

[9] Spalart, P.R. Philosophies and fallacies in turbulence modeling, *Prog. Aerospace Sci.*, 2015, **74**, pp 1–15.

[10] Park, J. and Choi, H. Toward neural-network-based large eddy simulation: Application to turbulent channel flow, *J. Fluid Mech.*, 2021, **914**, p A16.

[11] Kurz, M., Offenhäuser, P. and Beck, A. Deep reinforcement learning for turbulence modeling in large eddy simulations, *Int. J. Heat Fluid Flow*, 2023, **99**, pp 109094.

[12] Pope, S.B. A more general effective-viscosity hypothesis, *J. Fluid Mech.*, 1975, **72,** (02), p 331.

[13] Ling, J., Kurzawski, A. and Templeton, J. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance, *J. Fluid Mech.*, 2016, **807**, pp 155–166.

[14] Zhang, X.L., Xiao, H., Luo, X.D. and He, G.W. Ensemble kalman method for learning turbulence models from indirect observation data, *J. Fluid Mech.*, 2022, **949**, p A26.

[15] Mandler, H. and Weigand, B. A realizable and scale-consistent data-driven non-linear eddy viscosity modeling framework for arbitrary regression algorithms, *Int. J. Heat Fluid Flow*, 2022, **97**, p 109018.

[16] Weatheritt, J. and Sandberg, R. A novel evolutionary algorithm applied to algebraic modifications of the RANS stress–strain relationship, *J. Comput. Phys.*, 2016, **325**, pp 22–37.

[17] Zhao, Y.M., Akolekar, H.D., Weatheritt, J., Michelassi, V. and Sandberg, R.D. RANS turbulence model development using CFD-driven machine learning, *J. Comput. Phys.*, 2020, **411**, p 109413.

[18] Lav, C., Banko, A.J., Waschkowski, F., Zhao, Y.M., Elkins, C.J., Eaton, J.K. and Sandberg, R.D. A coupled framework for symbolic turbulence models from deep-learning, *Int. J. Heat Fluid Flow*, 2023, **101**, p 109140.

[19] Wu, J.L., Xiao, H. and Paterson, E. Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework, *Phys. Rev. Fluids*, 2018, **3,** (7), p 074602.

[20] Brener, B.P., Cruz, M.A., Macedo, M.S.S. and Thompson, R.L. A highly accurate strategy for data-driven turbulence modeling, *Comput. Appl. Math.*, 2024, **43,** (1), p 59.

[21] Parish, E.J. and Duraisamy, K. A paradigm for data-driven predictive modeling using field inversion and machine learning, *J. Comput. Phys.*, 2016, **305**, pp 758–774.

[22] Ferrero, A., Iollo, A. and Larocca, F. Field inversion for data-augmented rans modelling in turbomachinery flows, *Comput. Fluids*, 2020, **201**, p 104474.

[23] Yan, C.Y., Li, H.R., Zhang, Y.F. and Chen, H.X. Data-driven turbulence modeling in separated flows considering physical mechanism analysis, *Int. J. Heat Fluid Flow*, 2022, **96**, p 109004.

[24] Duraisamy, K., Zhang, Z.J. and Singh, A.P. New approaches in turbulence and transition modeling using data-driven techniques, 53rd AIAA Aerospace Sciences Meeting, 2015, p 1284.

[25] Yan, C.Y., Zhang, Y. and Chen, H. Data augmented turbulence modeling for three-dimensional separation flows, *Phys. Fluids*, 2022, **34,** (7), p 075101.

[26] Zhu, L.Y., Zhang, W.W., Kou, J.Q. and Liu, Y.L. Machine learning methods for turbulence modeling in subsonic flows around airfoils, *Phys. Fluids*, 2019, **31,** (1), p 015105.

[27] Zhang, S.M., Li, H.W., You, R.Q., Kong, T.L. and Tao, Z. A construction and training data correction method for deep learning turbulence model of Reynolds averaged Navier–Stokes equations, *AIP Adv.*, 2022, **12,** (6), p 065002.

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

[28] Wu, J.L., Xiao, H., Sun, R. and Wang, Q.Q. Reynolds-averaged navier–stokes equations with explicit data-driven reynolds stress closure can be ill-conditioned, *J. Fluid Mech.*, 2019, **869**, pp 553–586.

[29] Liu, W.S., Fang, J., Rolfo, S., Moulinec, C. and Emerson, D.R. An iterative machine-learning framework for rans turbulence modeling, *Int. J. Heat Fluid Flow*, 2021, **90**, p 108822.

[30] Brener, B.P., Cruz, M.A., Thompson, R.L. and Anjos, R.P. Conditioning and accurate solutions of reynolds average navier–stokes equations with data-driven turbulence closures, *J. Fluid Mech.*, 2021, **915**, p A110.

[31] Spencer, R., Przytarski, P., Adami, P., Grothe, P. and Wheeler, A.P.S. Importance of non-equilibrium modelling for compressors, *J. Turbomach.*, 2022, pp 1–16.

[32] Sun, X.X., Cao, W.B., Liu, Y.L., Zhu, L.Y. and Zhang, W.W. High Reynolds number airfoil turbulence modeling method based on machine learning technique, *Comput. Fluids*, 2022, **236**, p 105298.

[33] Abe, H., Kawamura, H. and Matsuo, Y. Direct numerical simulation of a fully developed turbulent channel flow with respect to the Reynolds number dependence, *J. Fluids Eng.*, 2001, **123,** (2), pp 382–393.

[34] Moser, R.D., Kim, J. and Mansour, N.N. Direct numerical simulation of turbulent channel flow up to $\mathrm{Re}_\tau = 590$, *Phys. Fluids*, 1999, **11,** (4), pp 943–945.

[35] Fang, L., Bao, T.W., Xu, W.Q., Zhou, Z.D., Du, J.L. and Jin, Y.. Data driven turbulence modeling in turbomachinery — an applicability study, *Comput. Fluids*, 2022, **238**, p 105354.

[36] Duraisamy, K. Perspectives on machine learning-augmented reynolds-averaged and large eddy simulation models of turbulence, *Phys. Rev. Fluids*, 2021, **6,** (5), p 050504.

[37] Schmelzer, M., Dwight, R.P. and Cinnella, P. Discovery of algebraic reynolds-stress models using sparse symbolic regression, *Flow Turbul. Combust.*, 2020, **104**, pp 579–603.

[38] Akolekar, H.D., Weatheritt, J., Hutchins, N., Sandberg, R.D., Laskowski, G. and Michelassi, V. Development and use of machine-learnt algebraic reynolds stress models for enhanced prediction of wake mixing in low-pressure turbines, *J. Turbomach.*, 2019, **141,** (4).

[39] Mandler, H. and Weigand, B. On frozen-rans approaches in data-driven turbulence modeling: Practical relevance of turbulent scale consistency during closure inference and application, *Int. J. Heat Fluid Flow*, 2022, **97**, p 109017.

[40] Parneix, S., Laurence, D. and Durbin, P.A. A procedure for using dns databases, *J. Fluids Eng.*, 1998, **120,** (1), pp 40–47.

[41] Weatheritt, J. and Sandberg, R.D. The development of algebraic stress models using a novel evolutionary algorithm, *Int. J. Heat Fluid Flow*, 2017, **68**, pp 298–318.

[42] Wu, C.Y. and Zhang, Y.F. Enhancing the shear-stress-transport turbulence model with symbolic regression: A generalizable and interpretable data-driven approach, *Phys. Rev. Fluids*, 2023, **8,** (8), p 084604.

[43] Liu, Y.W., Lu, L.P., Fang, L. and Gao, F. Modification of spalart–allmaras model with consideration of turbulence energy backscatter using velocity helicity, *Phys. Lett. A*, 2011, **375,** (24), pp 2377–2381.

[44] Liu, Y.W., Tang, Y.M., Scillitoe, A.D. and Tucker, P.G. Modification of shear stress transport turbulence model using helicity for predicting corner separation flow in a linear compressor cascade, *J. Turbomach.*, 2020, **142,** (2), p 021004.

[45] Liu, Y.W., Yan, H., Fang, L., Lu, L.P., Li, Q.S. and Shao, L. Modified k-ω model using kinematic vorticity for corner separation in compressor cascades, *Sci. China Technol. Sci.*, 2016, **59,** (5), pp 795–806.

[46] Sun, W. Assessment of advanced rans turbulence models for prediction of complex flows in compressors, *Chin. J. Aeronaut.*, 2023, **36,** (9), pp 162–177.

[47] Klein, T.S., Craft, T.J. and Iacovides, H. Assessment of the performance of different classes of turbulence models in a wide range of non-equilibrium flows, *Int. J. Heat Fluid Flow*, 2015, **51**, pp 229–256.

[48] Liu, F., Lu, L.P. and Fang, L. Non-equilibrium turbulent phenomena in transitional channel flows, *J. Turbul.*, 2018, **19,** (9), pp 731–753.

[49] Xiao, H., Wu, J.L., Laizet, S. and Duan, L. Flows over periodic hills of parameterized geometries: A dataset for data-driven turbulence modeling from direct simulations, *Comput. Fluids*, 2020, **200**, p 104431.

[50] Wang, J.X., Wu, J.L. and Xiao, H. Physics-informed machine learning approach for reconstructing reynolds stress modeling discrepancies based on DNS data, *Phys. Rev. Fluids*, 2017, **2,** (3), p 034603.

[51] Liu, W.S., Song, Z.M. and Fang, J. Nnpred: A predictor library to deploy neural networks in computational fluid dynamics software, arXiv preprint arXiv:2209.12339, September 2022.

[52] Liu, W.S., Fang, J., Rolfo, S., Moulinec, C. and Emerson, D.R. On the improvement of the extrapolation capability of an iterative machine-learning based RANS framework, *Comput. Fluids*, 2023, **256**, p 105864.

[53] Abadi, M., Barham, P., Chen, J.M., Chen, Z.F., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y. and Zheng, X.Q. Tensorflow: A system for large-scale machine learning, Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, OSDI'16, USA, 2016, USENIX Association, pp 265–283.

[54] Thompson, R.L., Sampaio, L.E.B., de Bragança Alves, F.A.V., Thais, L. and Mompean, G. A methodology to evaluate statistical errors in dns data of plane channel flows, *Comput. Fluids*, 2016, **130**, pp 1–7.

[55] Poroseva, S.V., Colmenares, F.J.D. and Murman, S.M. On the accuracy of rans simulations with dns data, *Phys. Fluids*, 2016, **28,** (11), p 115102.

[56] Vassilicos, J.C. Dissipation in turbulent flows, *Ann. Rev. Fluid Mech.*, 2015, **47,** (1), pp 95–114.

[57] Bos, W.J.T. and Rubinstein, R. Dissipation in unsteady turbulence, *Phys. Rev. Fluids*, 2017, **2,** (2), p 022601.

[58] Li, H.R., Zhang, Y.F. and Chen, H.X. Aerodynamic prediction of iced airfoils based on modified three-equation turbulence model, *AIAA J.*, 2020, **58,** (9), pp 3863–3876.

[59] Liu, F., Lu, L.P., Bos, W.J.T. and Fang, L. Assessing the nonequilibrium of decaying turbulence with reversed initial fields, *Phys. Rev. Fluids*, 2019, **4,** (8), p 084603.

https://doi.org/10.1017/aer.2024.123 Published online by Cambridge University Press

[60] Fang, L. and Bos, W.J.T. An EDQNM study of the dissipation rate in isotropic non-equilibrium turbulence, *J. Turbul.*, 2023, **24**, pp 217–324.
[61] Shao, X., Fang, J. and Fang, L. Non-equilibrium dissipation laws in a minimal two-scale wake model, *Phys. Fluids*, 2023, **35**, p 085105.
[62] Araki, R. and Bos, W.J.T. Inertial range scaling of inhomogeneous turbulence, *J. Fluid Mech.*, 2024, **978**, p A9.
[63] Ma, B., Van Doorne, C.W.H., Zhang, Z. and Nieuwstadt, F.T.M. On the spatial evolution of a wall-imposed periodic disturbance in pipe poiseuille flow at $re = 3000$. part 1. subcritical disturbance, *J. Fluid Mech.*, 1999, **398**, pp 181–224.
[64] Xu, C., Zhang, Z., den Toonder, J.M.J. and Nieuwstadt, F.T.M. Origin of high kurtosis levels in the viscous sublayer. direct numerical simulation and experiment, *Phys. Fluids*, 1996, **8,** (7), pp 1938–1944.
[65] Liu, F. Researches on the Non-Equilibrium Properties of Turbulence, PhD thesis, Beihang University, Beijing, China, 2019.
[66] Hearst, R.J. and Lavoie, P. Velocity derivative skewness in fractal-generated, non-equilibrium grid turbulence, *Phys. Fluids*, 2015, **27,** (7), p 071701.
[67] Hussain, A.K.M.F. and Reynolds, W.C. Measurements in fully developed turbulent channel flow, *J. Fluids Eng.*, 1975, **97,** (4), pp 568–578.
[68] Pope, S.B. *Turbulent Flows*, Cambridge University Press, New York, 2000.
[69] Zhao, H.K., Liu, Y.W., Shao, L., Fang, L. and Dong, M. Existence of positive skewness of velocity gradient in early transition, *Phys. Rev. Fluids*, 2021, **6,** (10), p 104608.
[70] Tibshirani, R. Regression shrinkage and selection via the lasso, *J. R. Stat. Soc. Ser. B (Methodolog.)*, 1996, **58,** (1), pp 267–288.
[71] Zhao, H.K. Analysis of Nonequilibrium Turbulence Phenomenon and Turbulence Modeling for Compressor Internal Model Flow, PhD thesis, Beihang University, Beijing, China, 2021.
[72] Tao, S.C. and Zhou, Y. Turbulent flows around side-by-side cylinders with regular and multiscale arrangements, *Phys. Rev. Fluids*, 2019, **4,** (12), p 124602.

Cite this article: Du J.L., Liu W.S., Fang L. and Bao T.W. Involving non-equilibrium training dataset in data-driven turbulence modeling for turbomachinery. *The Aeronautical Journal*, https://doi.org/10.1017/aer.2024.123