

Discovery of Algebraic Reynolds-Stress Models Using Sparse Symbolic Regression.

Martin Schmelzer Richard P. Dwight Paola Cinnella

This article is published (open-access). Please, use the original article instead of this one. For citation:

Schmelzer, M., Dwight, R.P. & Cinnella, P.
Discovery of Algebraic Reynolds-Stress Models Using
Sparse Symbolic Regression.
Flow Turbulence Combustion 104, 579603 (2020).
<https://doi.org/10.1007/s10494-019-00089-x>

Abstract

A novel deterministic symbolic regression method SpaRTA is introduced to infer algebraic stress models for the closure of RANS equations directly from high-fidelity LES or DNS data. The models are written as tensor polynomials and are built from a library of candidate functions. The machine-learning method is based on elastic net regularisation which promotes sparsity of the inferred models. By being data-driven the method relaxes assumptions commonly made in the process of model development. Model-discovery and cross-validation is performed for three cases of separating flows, i.e. periodic hills ($Re=10595$), converging-diverging channel ($Re=12600$) and curved backward-facing step ($Re=13700$). The predictions of the discovered models are significantly improved over the $k-\omega$ SST also for a *true* prediction of the flow over periodic hills at $Re=37000$. This study shows a systematic assessment of SpaRTA for rapid machine-learning of robust corrections for standard RANS turbulence models.

Keywords: Turbulence Modelling, Machine Learning, Sparse Regression, Symbolic Regression

1 Introduction

The capability of Computational Fluid Dynamics (CFD) to deliver reliable prediction is limited by the unsolved closure problem of turbulence modelling.

The workhorse for turbulence modelling in industry are the Reynolds-Averaged Navier-Stokes (RANS) equations using linear eddy viscosity models (LEVM) [1]. The lower computational costs compared to high-fidelity approaches, e.g. Large-Eddy (LES) or Direct Numerical Simulations (DNS), come at the price of uncertainty especially for flows with separation, adverse pressure gradients or high streamline curvature. Data-driven methods for turbulence modelling based on supervised machine learning have been introduced to leverage RANS for improved predictions [2, 3, 4]. In [5], the source terms of the Spalart-Allmaras were learnt from data using a single hidden layer neural network, which served as a first feasibility study. In [6], a factor was introduced to correct the turbulent production in the k -equation of the $k-\omega$ model. This term was found via inverse modelling and served to train a Gaussian process. While this approach has been extended and applied to industrially relevant flows such as airfoils in [7, 8] it still relies on the Boussinesq assumption. In [9], a deep neural network was trained to predict a_{ij} given input only from a baseline linear eddy viscosity simulation and thus replacing the turbulence model instead of augmenting it. The network was designed to embed Galilean invariance of the predicted a_{ij} . This concept of physics-informed machine learning was extended, e.g., in [10] using random forest regression. Despite the success of the data-driven approaches a drawback is their black box nature, which hampers the understanding of the physics of the resulting models in order to derive new modelling ideas from it.

Recently, a method has been introduced using genetic-programming (GEP) based symbolic regression to derive Explicit Algebraic Reynolds-stress Models (EARSM) directly from high-fidelity data [11, 12]. EARSM, first introduced by [13] and further developed by [14], are nonlinear extensions of LEVM and are commonly derived by projecting Reynolds-stress models (RSM) onto a set of tensorial polynomials [15, 16]. These models are numerically more robust than RSM at similar computational costs as LEVM [17], but do not show superior predictive capabilities for all kinds of flows [15]. The data-driven GEP method retains the input quantities used to derive EARSM, but replaces the commonly used projection method to find the formal structure of the model by an evolutionary process, which makes it an open-box machine learning approach. The advantage of such a data-driven method is that instead of relying on assumptions made during the development of an EARSM, a model is inferred directly from data. While such a model might not provide an universal approach for all kinds of flows as commonly aimed for in physical modelling, it serves as a pragmatic tool to correct the flow at hand. For cases exhibiting similar flow physics, e.g. separation, it has also been shown that the discovered models provide suitable corrections indicating the predictive potential of a data-driven approach.

Due to the non-deterministic nature of GEP it discovers for each run another model with a different mathematical form, e.g. other terms and/or other values for coefficients, with varying complexity. It is reported that the models using only a few nonlinear terms show a low training and prediction error as well as high numerical robustness for industrially relevant flow cases [18, 19]. Therefore, we instead introduce a new deterministic symbolic regression method SpaRTA (Sparse Regression of Turbulent Stress Anisotropy), for

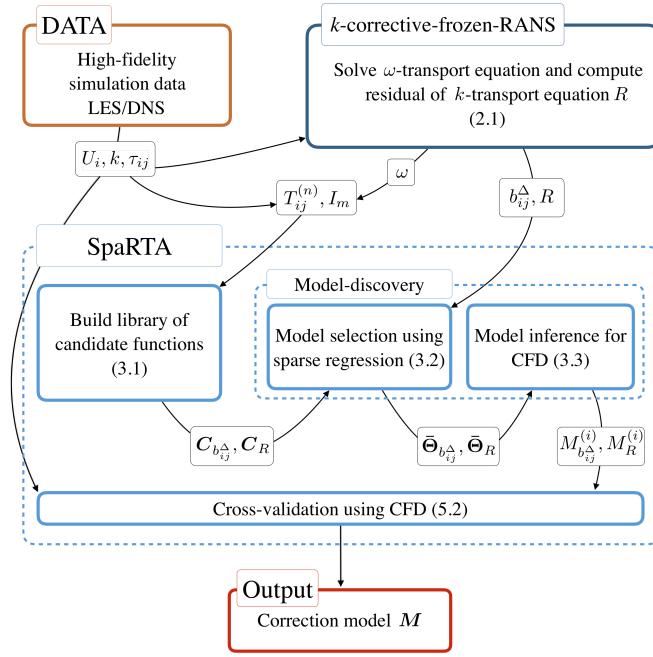


Figure 1: Technical flow diagram of SpaRTA (Sparse Regression of Turbulent Stress Anisotropy).

which we constrain the search towards sparse algebraic models using sparsity-promoting regression techniques [20, 21]. SpaRTA combines functions from a predefined library of candidates without any random recombination. It consists of four steps: (i) building a library of candidate functions, (ii) model selection using sparse-regression techniques, (iii) inference of model coefficients and (iv) cross-validation of the resulting models, see Figure 1. The first three steps are computationally very cheap also for high-dimensional problems and allow for rapid model discovery.

The present study provides several novel concepts for data-driven modelling, which are organised as follows. In Section 2 we define additive model-form error terms within the $k-\omega$ SST LEVM model and use k -corrective-frozen-RANS, which is an extension of the method introduced in [12], to compute the model-form error from high-fidelity data. The novelty in this work is that we identify not only a correction of the stress-strain relation, but also one for the turbulent transport equations and thereby achieve excellent agreement with mean-fields of high-fidelity data. We also validate that the model-form error is successfully captured by adding the two terms to the solver and performing a CFD simulation. The k -corrective-frozen-RANS does not require any iterative optimisation procedure as compared to [6] and is therefore very efficient, but also limited to full-field data. In Section 3 we introduce the steps of SpaRTA. The details of the test cases, the CFD setup and the sources of the high-fidelity data are given in Section 4. In Section 5 SpaRTA is applied to the test cases, the discovered models are presented and the best models are chosen using cross-validation. Finally, conclusions are drawn in Section 6.

2 Model-form error of RANS equations

In the following, we augment the baseline model, i.e. the linear eddy viscosity assumption and the turbulence transport equations of the $k-\omega$ SST, with additive terms accounting for the error due to the model-form. We introduce k -corrective-frozen-RANS, which is an extension of the method in [12], to extract these two types of error from high-fidelity data sources efficiently. Finally, we validate that the extracted terms reduce the error for given test cases.

2.1 Identification of additive model-form error from data

The incompressible and constant-density RANS equations read

$$\begin{aligned} \partial_i U_i &= 0, \\ U_j \partial_j U_i &= \partial_j \left[-\frac{1}{\rho} P + \nu \partial_j U_i - \tau_{ij} \right], \end{aligned} \quad (1)$$

where U_i is the mean velocity, ρ is the constant density, P is the mean pressure and ν is the kinematic viscosity. The Reynolds-stress τ_{ij} is the subject of modelling. This symmetric, second-order tensor field can be decomposed into

an anisotropic $a_{ij} = 2kb_{ij}$ and isotropic part $\frac{2}{3}k\delta_{ij}$

$$\tau_{ij} = 2k \left(b_{ij} + \frac{1}{3}\delta_{ij} \right), \quad (2)$$

in which the baseline model, $b_{ij}^o = -\frac{\nu_t}{k}S_{ij}$, forms a linear relation between anisotropy and the mean-strain rate tensor S_{ij} via the scalar eddy viscosity ν_t . Commonly, ν_t is computed using a transport model such as $k-\omega$ SST [15], in which k is the turbulent kinetic energy and ω the specific dissipation rate.

In order to extract the model-form error in these models from high-fidelity data sources, we compute the residuals of the baseline turbulence model given the data. The residual for the constitutive relation is equivalent to an additive term b_{ij}^Δ leading to an augmented constitutive relation

$$b_{ij} = -\frac{\nu_t}{k}S_{ij} + b_{ij}^\Delta. \quad (3)$$

To evaluate b_{ij}^Δ it is necessary to estimate ν_t , therefore also ω needs to be specified. In [12, 22], ω was efficiently obtained by passively solving the ω transport equation given high-fidelity data for U_i , k and b_{ij} . The associated ν_t was then used to compute b_{ij}^Δ with (3). This method is named frozen-RANS as only one equation is solved iteratively while the remaining variables are frozen. Despite the fact that b_{ij}^Δ also alters the production of turbulent kinetic energy P_k , it is not evident that solving the k equation given the data and the frozen ω should lead to the same k as present in the data. Therefore, we introduce k -corrective-frozen-RANS for which we also compute the residual of the k equation alongside the computation of the frozen ω . The residual is equivalent to an additive correction term, which we define as R , leading to an augmented $k-\omega$ SST model

$$\partial_t k + U_j \partial_j k = P_k + R - \beta^* \omega k + \partial_j [(\nu + \sigma_k \nu_t) \partial_j k], \quad (4)$$

$$\partial_t \omega + U_j \partial_j \omega = \frac{\gamma}{\nu_t} (P_k + R) - \beta \omega^2 + \partial_j [(\nu + \sigma_\omega \nu_t) \partial_j \omega] + CD_{k\omega}, \quad (5)$$

in which the production of turbulent kinetic energy is augmented by b_{ij}^Δ to $P_k = 2k(b_{ij}^o + b_{ij}^\Delta)\partial_j U_i$. The corresponding eddy viscosity is $\nu_t = \frac{a_1 k}{\max(a_1 \omega, SF_2)}$. The other standard terms of $k-\omega$ SST read

$$\begin{aligned} CD_{k\omega} &= \max \left(2\sigma_{\omega 2} \frac{1}{\omega} (\partial_i k)(\partial_i \omega), 10^{-10} \right), \\ F_1 &= \tanh \left[\left(\min \left[\max \left(\frac{\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{y^2 \omega} \right), \frac{4\sigma_{\omega 2} k}{CD_{k\omega} y^2} \right] \right)^4 \right], \\ F_2 &= \tanh \left[\left(\max \left(\frac{2\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{y^2 \omega} \right) \right)^2 \right], \\ \Phi &= F_1 \Phi_1 + (1 - F_1) \Phi_2, \end{aligned} \quad (6)$$

Table 1: Mean-squared error ϵ of reconstructed velocity U_i and Reynolds-stress τ_{ij} for different test cases with b_{ij}^Δ and R added as static fields to the solver. Normalisation with ϵ of the baseline $k-\omega$ SST results U_i^o and τ_{ij}^o . Description of cases in Section 4.

Case	$\epsilon(U_i) \cdot 10^{-5}$	$\epsilon(U_i)/\epsilon(U_i^o)$	$\epsilon(\tau_{ij}) \cdot 10^{-6}$	$\epsilon(\tau_{ij})/\epsilon(\tau_{ij}^o)$
PH ₁₀₅₉₅	1.74	0.00165	36.7	0.1495
CD ₁₂₆₀₀	31.4	0.0229	7.21	0.4781
CBFS ₁₃₇₀₀	59.6	0.22703	1.34	0.4949

in which the latter blends the coefficients $\Phi \rightarrow (\Phi_1, \Phi_2)$

$$\alpha = (5/9, 0.44), \beta = (3/40, 0.0828), \sigma_k = (0.85, 1.0), \sigma_\omega = (0.5, 0.856). \quad (7)$$

The remaining terms are $\beta^* = 0.09$, $a_1 = 0.31$ and $S = \sqrt{2S_{ij}S_{ij}}$. During the iterative computation of the frozen ω the residual of the k equation is fed back into the ω equation until convergence is achieved. In order to validate that the resulting fields compensate the model-form error, b_{ij}^Δ and R are added as static fields to a modified OpenFOAM solver [23] and a CFD simulation is performed starting from the baseline solution for the flow configurations described in Section 4, for which high-quality data is available. The mean-squared error between the high-fidelity data and the reconstructed velocity U_i as well as the Reynolds-stress τ_{ij} is low, see Table 1. Also the stream-wise velocity profiles shown in Figure 2 demonstrate that the high-fidelity mean-flow data is essentially reproduced given b_{ij}^Δ and R . The k -corrective-frozen-RANS approach requires full-field data, but is not based on an inversion procedure, e.g. using adjoint-based optimisation as in [6, 8], which makes it very cost-efficient.

2.2 Nonlinear eddy-viscosity models for b_{ij}^Δ and R

In order to discover corrections for the model-form error b_{ij}^Δ and R , we need to decide on a modelling ansatz. Within this mathematical framework the symbolic regression targets to find specific expressions as corrections models. In [13], a nonlinear generalisation of the linear eddy viscosity concept was proposed. This concept has been used in several works on data-driven turbulence modelling [2, 3]. The fundamental assumption is made that the anisotropy of the Reynolds-stress b_{ij} not only depends on the strain rate tensor $S_{ij} = \tau \frac{1}{2}(\partial_j U_i + \partial_i U_j)$ but also on the rotation rate tensor $\Omega_{ij} = \tau \frac{1}{2}(\partial_j U_i - \partial_i U_j)$ with the timescale $\tau = 1/\omega$. The Cayley-Hamilton theorem then dictates that the most general form of the anisotropic part of the Reynolds-stress can be expressed as

$$b_{ij}(S_{ij}, \Omega_{ij}) = \sum_{n=1}^N T_{ij}^{(n)} \alpha_n(I_1, \dots, I_5), \quad (8)$$

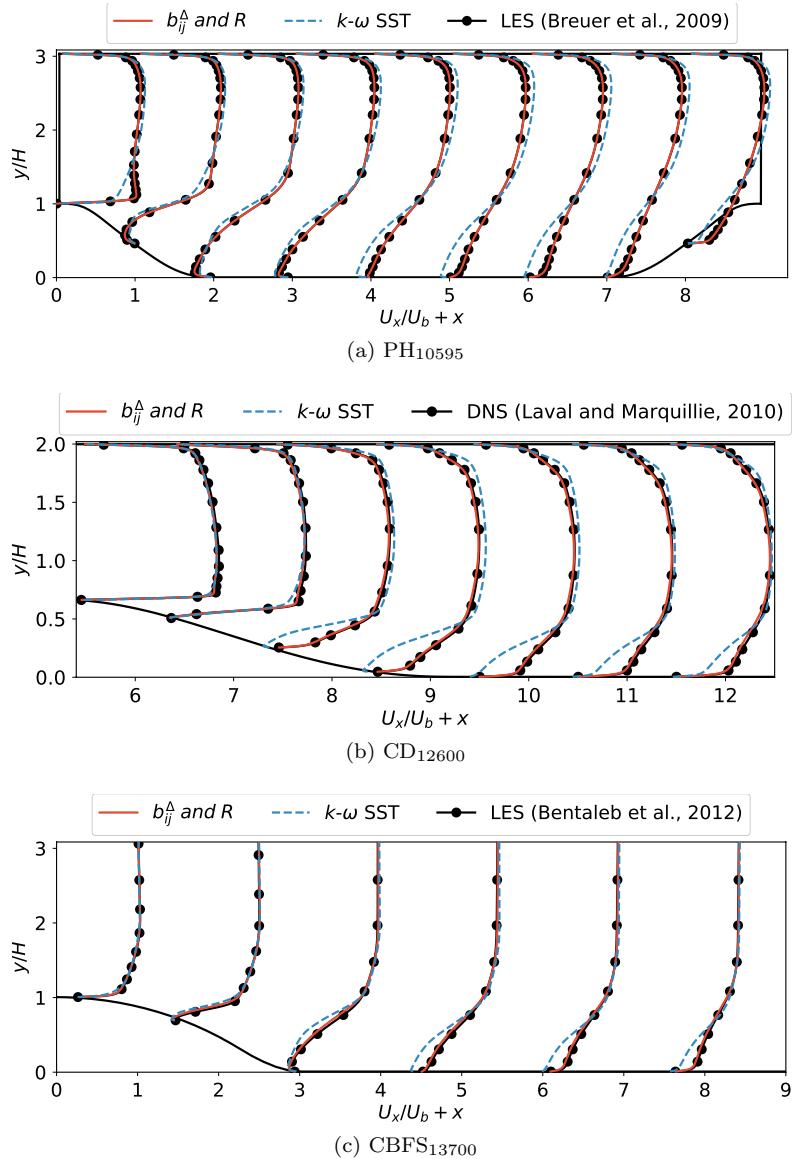


Figure 2: Stream-wise velocity component for propagated model-form error acquired using k -corrective-frozen-RANS.

with ten nonlinear base tensors $T_{ij}^{(n)}$ and five corresponding invariants I_m . Only the first four base tensors and the first two invariants are used in this work, which are

$$\begin{aligned} T_{ij}^{(1)} &= S_{ij}, \\ T_{ij}^{(2)} &= S_{ik}\Omega_{kj} - \Omega_{ik}S_{kj}, \\ T_{ij}^{(3)} &= S_{ik}S_{kj} - \frac{1}{3}\delta_{ij}S_{mn}S_{nm}, \\ T_{ij}^{(4)} &= \Omega_{ik}\Omega_{kj} - \frac{1}{3}\delta_{ij}\Omega_{mn}\Omega_{nm} \end{aligned} \quad (9)$$

$$I_1 = S_{mn}S_{nm}, I_2 = \Omega_{mn}\Omega_{nm}. \quad (10)$$

Using this set for (8) we have an ansatz, which only requires functional expressions for the coefficients α_n , to model b_{ij}^Δ . However, computing b_{ij}^Δ using (3) requires a correct k as discussed in Section 2.1. This aspect is taken into account in the modelling ansatz for R , for which we take a closer look at the eddy viscosity concept.

Both linear and nonlinear eddy viscosity models provide expressions for the anisotropy b_{ij} based on a local relation between stress and strain. Due to the restriction of this local closure only the normal stresses $\frac{2}{3}k\delta_{ij}$ can account for nonlocal effects by transport equations for the turbulent quantities using convection and diffusion terms [15, 24]. The term R provides local information to correct the transport equations. Depending on the local sign of R it either increases or decreases the net production P_k locally. Hence, it acts as an additional production or dissipation term, which can overcome the error in k . We model it in a similar way to the turbulent production

$$R = 2kb_{ij}^R\partial_j U_i, \quad (11)$$

which has the additional benefit that we can also use the framework of nonlinear eddy viscosity models to model R .

Since the general modelling framework is the same for both b_{ij}^Δ and R , a natural next step would be to combine both in order to find a single model accounting for the sources of model-form error on the level of the constitutive relation as well as within the turbulent transport equations. For example in [12] models identified using genetic programming were modified such that any additional contribution of the first base tensor $T_{ij}^{(1)}$ in (8) was added with a positive sign for the computation of P_k . This ad-hoc correction was established based on physical reasoning to avoid very low production close to walls and led to significantly improved predictions. However, in contrast to [12] we have extracted two target terms b_{ij}^Δ and R using k -corrective-frozen-RANS, which also make it possible to systematically study (i) how to obtain corrections models for each target individually and (ii) their combined effect on the predictions.

Given the polynomial model (8) and the set of base tensors (9) and invariants (10) we are now left with the task of providing suitable expressions for $\alpha_n(I_1, I_2)$

for $n = 1, \dots, 4$ to overcome the model-form error. This is the purpose of the deterministic symbolic regression technique detailed in the following section.

3 Model discovery methodology

Deterministic symbolic regression constructs a large library of nonlinear candidate functions to regress data. It identifies the relevant candidates by adopting a sparsity constraint. Two fundamental methods have been proposed: Sparse identification of nonlinear dynamics (SINDy) [20, 25] and fast function extraction (FFX) [26]. Both methods were applied in several areas of physical modelling. In the following, we introduce the steps of the model discovery methodology SpaRTA based on FFX, for which a library is constructed using a set of raw input variables and mathematical operations. The model selection uses elastic net regression. Finally, for the inference of the model coefficients the stability requirements of a CFD solver are considered. An overview of SpaRTA is given in Figure 1.

3.1 Building a library of candidate functions

The deterministic symbolic regression requires a library of candidate functions, from which a model is deduced by building a linear combination of the candidates. Hence, the library is an essential element of the entire methodology and needs to accommodate relevant candidates explaining the data. We rely on the nonlinear eddy viscosity concept and aim to find models for α_n in (8) given as primitive input features the invariants I_1 and I_2 . For the present work we focus on a library, in which the primitive input features are squared and the resulting candidates are multiplied by each other leading to a maximum degree of 6. In addition to the two invariants we also include a constant function c to the set of raw input features. The resulting vector \mathcal{B} reads

$$\mathcal{B} = \left[c, I_1, I_2, I_1^2, I_2^2, I_1^2 I_2^3, I_1^4 I_2^2, I_1 I_2^2, I_1 I_2^3, I_1 I_2^4, I_1^3 I_2, I_1^2 I_2^4, I_1^2 I_2, I_1 I_2, I_1^3 I_2^2, I_1^2 I_2^2 \right]^T \quad (12)$$

with the cardinality of \mathcal{B} , $|\mathcal{B}| = 16$.

For the library to regress models for b_{ij}^Δ each function of \mathcal{B} is multiplied with each base tensor $T_{ij}^{(n)}$, leading to the library of tensorial candidate functions

$$\mathcal{C}_{b_{ij}^\Delta} = \left[c T_{ij}^{(1)}, c T_{ij}^{(2)}, \dots, I_1^2 I_2^2 T_{ij}^{(4)} \right]^T. \quad (13)$$

In order to regress models for R the double dot product of each function in $\mathcal{C}_{b_{ij}^\Delta}$ with the mean velocity gradient tensor $\partial_j U_i$ is computed, leading to

$$\mathcal{C}_R = \left[c T_{ij}^{(1)} \partial_j U_i, \dots, I_1^2 I_2^2 T_{ij}^{(4)} \partial_j U_i \right]^T. \quad (14)$$

The two libraries $\mathcal{C}_{b_{ij}^\Delta}$ and \mathcal{C}_R are evaluated given the high-fidelity validation data for each test case and stored column-wise in matrices

$$\mathbf{C}_{b_{ij}^\Delta} = \begin{bmatrix} cT_{xx}^{(1)}|_{k=0} & cT_{xx}^{(2)}|_{k=0} & \dots & I_1^2 I_2^2 T_{xx}^{(4)}|_{k=0} \\ cT_{xy}^{(1)}|_{k=0} & cT_{xy}^{(2)}|_{k=0} & \dots & I_1^2 I_2^2 T_{xy}^{(4)}|_{k=0} \\ cT_{xz}^{(1)}|_{k=0} & cT_{xz}^{(2)}|_{k=0} & \dots & I_1^2 I_2^2 T_{xz}^{(4)}|_{k=0} \\ cT_{yy}^{(1)}|_{k=0} & cT_{yy}^{(2)}|_{k=0} & \dots & I_1^2 I_2^2 T_{yy}^{(4)}|_{k=0} \\ cT_{yz}^{(1)}|_{k=0} & cT_{yz}^{(2)}|_{k=0} & \dots & I_1^2 I_2^2 T_{yz}^{(4)}|_{k=0} \\ cT_{zz}^{(1)}|_{k=0} & cT_{zz}^{(2)}|_{k=0} & \dots & I_1^2 I_2^2 T_{zz}^{(4)}|_{k=0} \\ \vdots & \vdots & & \vdots \\ cT_{zz}^{(1)}|_{k=K} & cT_{zz}^{(2)}|_{k=K} & \dots & I_1^2 I_2^2 T_{zz}^{(4)}|_{k=K} \end{bmatrix} \in \mathbb{R}^{6K \times |\mathcal{C}_{b_{ij}^\Delta}|}, \quad (15)$$

$$\mathbf{C}_R = \begin{bmatrix} cT_{ij}^{(1)} \partial_j U_i|_{k=0} & \dots & I_1^2 I_2^2 T_{ij}^{(4)} \partial_j U_i|_{k=0} \\ \vdots & & \vdots \\ cT_{ij}^{(1)} \partial_j U_i|_{k=K} & \dots & I_1^2 I_2^2 T_{ij}^{(4)} \partial_j U_i|_{k=K} \end{bmatrix} \in \mathbb{R}^{K \times |\mathcal{C}_R|}, \quad (16)$$

in which K is the number of mesh points of the test case at hand. The corresponding target data b_{ij}^Δ and R are stacked to vectors

$$\mathbf{b}^\Delta = [b_{xx}^\Delta|_{k=0}, b_{xy}^\Delta|_{k=0}, \dots, b_{zz}^\Delta|_{k=K}]^T \in \mathbb{R}^{6K}, \quad (17)$$

$$\mathbf{R} = [R|_{k=0}, R|_{k=1}, \dots, R|_{k=K}]^T \in \mathbb{R}^K. \quad (18)$$

3.2 Model selection using sparsity-promoting regression

Given the above defined libraries the task is to form a linear model to regress the target data $\Delta = \mathbf{b}^\Delta$ or \mathbf{R} by finding the coefficient vector Θ

$$\Delta = \mathbf{C}_\Delta \Theta, \quad (19)$$

which represents a large, overdetermined system of equations. When using ordinary least-squares regression a dense coefficient vector Θ is obtained, resulting in overly complex models, which are potentially overfitting the data given the large libraries (13) and (14). Due to multi-collinearity between the candidates, \mathbf{C}_Δ can be ill-conditioned, so that the coefficients may also display large differences in magnitude expressed in a large l_1 -norm of Θ . Such models are unsuitable to be implemented in a CFD solver as they increase the numerical stiffness of the problem and impede convergence of the solution.

Following the idea of parsimonious models we constrain the search to models which optimally balance error and complexity and are not overfitting the data [25]. In principle, given a library a combinatoric study can be carried out, by performing an ordinary least-squares regression for each possible subset of candidates. Starting from each single candidate function individually, proceeding with all possible pairs up to more complex combinations. As the number of possible models grows exponentially with the number of candidates $I = 2^{|\mathcal{C}_\Delta|} - 2$

this approach becomes already infeasible for the simple libraries (13) and (14) with $|\mathcal{C}_\Delta| \approx 64$.

Hence, we follow [25, 26] and engage sparsity-promoting regularisation of the underlying least-squares optimisation problem. The model-discovery procedure is divided into two parts: (i) model selection and (ii) model inference, see Figure 1. For the first step, the model selection, we use the elastic net formulation

$$\begin{aligned} \Theta = \arg \min_{\hat{\Theta}} & \left\| C_\Delta \hat{\Theta} - \Delta \right\|_2^2 + \lambda \rho \left\| \hat{\Theta} \right\|_1 \\ & + 0.5 \lambda (1 - \rho) \left\| \hat{\Theta} \right\|_2^2, \end{aligned} \quad (20)$$

which blends the l_1 - and l_2 -norm regularisation given the mixing parameter $\rho \in [0, 1]$ and the regularisation weight λ , to promote the sparsity of Θ [26, 27]. On its own, the l_1 -norm, known as Lasso-regression, promotes sparsity by allowing only a few nonzero coefficients while shrinking the rest to zero. The l_2 -norm, known as Ridge-regression, enforces relatively small coefficients without setting them to zero, but is able to identify also correlated candidate functions instead of picking a single one. By combining both methods, the elastic net can find sparse models with a good predictive performance. Besides the mixing parameter, also the regularisation parameter λ shapes the form of the model: For a very large λ the vector Θ will only contain zeros independent of ρ . The amount of nonzero coefficients increases for smaller λ values making the discovery of sparse models possible.

Given the elastic net regularisation method we need to specify suitable combinations of the weight λ and type of the regularisation ρ , for which the optimisation problem (20) is solved. Most commonly the optimal (λ, ρ) combination is found based on a strategy to avoid overfitting of the resulting models, e.g. using cross-validation [25], for which the data is split into a training and a test set. While the optimisation problem given a grid (λ, ρ) is solved on the former, only the model with the best performance evaluated on the latter survives. For the purpose of CFD a true validation of the models can only be performed once they are implemented in a solver and applied to a test case. In order to not overcharge the role of the training data from k -corrective-frozen-RANS at this stage of the methodology, we select a wide spectrum of models varying in accuracy and complexity using (20) instead of a single one. The validation task will be performed later using a CFD solver.

Following [26] we use

$$\rho = [0.01, 0.1, 0.2, 0.5, 0.7, 0.9, 0.95, 0.99, 1.0]^T, \quad (21)$$

which ensures that we cover a substantial range of different regularisation types. The upper limit of the regularisation weight is defined as $\lambda_{\max} = \max(|C_\Delta^T \Delta|)/(K\rho)$, because for any $\lambda > \lambda_{\max}$ all elements in Θ will be equal to zero. The entire vector

$$\lambda = [\lambda_0, \dots, \lambda_{\max}]^T \quad (22)$$

is defined of having 100 entries between $\lambda_0 = \xi \lambda_{\max}$ with $\xi = 10^{-3}$ uniformly spaced using a log-scale as defined in [26]. This provides a search space $(\boldsymbol{\lambda}, \boldsymbol{\rho})$, the elastic net, which is large enough and has an appropriate resolution. At each grid point (λ_i, ρ_j) a vector $\boldsymbol{\Theta}_{\Delta}^{(i,j)}$ as a solution of (20) is found using the coordinate descent algorithm. The duration for the model selection step given the number of data points $K \sim 15000$ is of the order of a minute on a standard consumer laptop.

Solving (20) for different (λ_i, ρ_j) might produce $\boldsymbol{\Theta}_{\Delta}^{(i,j)}$ with the same abstract model form $\bar{\boldsymbol{\Theta}}$, which means that the same entries are equal to zero. As the specific values of the coefficients will be defined in the next step, the selection step of SpaRTA concludes with filtering out the set of D unique abstract model forms $\mathcal{D}_{\Delta} = \{\bar{\boldsymbol{\Theta}}_{\Delta}^d | d = 1, \dots, D\}$.

3.3 Model inference for CFD

The abstract models \mathcal{D}_{Δ} are found using standardised candidates, because the relevance of each candidate should not be determined by its magnitude during the model selection step. With the aim of defining a model with the correct units, we need to perform an additional regression using the unstandardised candidate functions for each subset determined by the abstract model forms in \mathcal{D}_{Δ} , which is the purpose of the model inference step outlined in the following.

In [25, 28, 29] this was done using ordinary least-squares regression for problems in the domains of dynamical systems and biological networks. As mentioned above, the ability of the CFD solver, in which the models will be implemented, to produce a converged solution is sensitive to large coefficients, which has been reported in [11, 12, 22]. We take this additional constraint into account by performing a Ridge regression

$$\boldsymbol{\Theta}_{\Delta}^{s,d} = \arg \min_{\hat{\boldsymbol{\Theta}}_{\Delta}^{s,d}} \left\| \mathbf{C}_{\Delta}^s \hat{\boldsymbol{\Theta}}_{\Delta}^{s,d} - \boldsymbol{\Delta} \right\|_2^2 + \lambda_r \left\| \hat{\boldsymbol{\Theta}}_{\Delta}^{s,d} \right\|_2^2, \quad (23)$$

in which λ_r is the Tikhonov-regularisation parameter. The index s denotes the submatrix of \mathbf{C}_{Δ} and the subvector of $\boldsymbol{\Theta}_{\Delta}^d$ consisting of the selected columns or elements respectively as defined in \mathcal{D}_{Δ} . The elements of $\boldsymbol{\Theta}_{\Delta}^d$ associated with the inactive candidates are zero and are not modified during this step.

By using the l_2 -norm regularisation the magnitude of the nonzero coefficients is shrunk [25, 30]. In general, low values for λ_r reduce the bias introduced through regularisation, but lead to larger coefficient values, and vice versa. Since shrinkage of the coefficients also reduces the influence of candidate functions with a lower magnitude compared to others, we need to find a trade-off between error of the model on the target data $\boldsymbol{\Delta}$ and the likelihood that the model will deliver converged solutions when used in a CFD solver. The problem of finding such an optimum is that the latter aspect can only be answered retrospectively. Recently, this problem has been addressed in [31] by embedding CFD simulations in the search for correction models guided by genetic programming. While this increases the costs of the model search drastically,

it also significantly increases the chance of delivering models with better convergence properties. Even though this procedure provides a strong indication, the identified models are also not guaranteed to converge *a priori* for any other test case outside the training set. Via testing using the cases in Section 4, we have identified $0.1 < \lambda_r < 0.01$ able to deliver coefficients in a range balancing the error on the target data Δ and the likelihood to produce converged CFD solutions.

Our efforts are based on an empirical observation, but do not guarantee a well-behaving numerical setup under all conditions. However, we have identified corrections of b_{ij}^Δ as the only contribution which can do harm to the convergence properties for the given test cases. Therefore, if a model does not converge, we further decrease the coefficients by a factor $\xi = 0.1$, for the model correcting b_{ij}^Δ only. This ad-hoc intervention is sufficient to achieve convergence for the studied cases.

Finally, the resulting coefficient vector Θ_Δ^d is used to retrieve the symbolic expression of the models by a dot product with the library of candidate functions \mathcal{C}_Δ in (13) and (14)

$$M_\Delta^d := \mathcal{C}_\Delta^T \Theta_\Delta^d, \quad (24)$$

which are implemented in the open-source finite-volume code OpenFOAM [23]. The divergence terms of the equations are discretised with linear upwinding and turbulent diffusion with 2nd order central differencing. In summary, the model discovery step of SpaRTA selects models utilising elastic net regression in (20) and further infers the coefficients of the selected models in (23). The latter process is guided by the aim to discover models complying with the restrictions of a CFD solver.

4 Test cases and high-fidelity data

In order to apply SpaRTA we need full-field data of U_i , k and τ_{ij} , which we take from LES and DNS studies conducted by other researchers. We have selected three test cases of separating flows over curved surfaces in two-dimensions with similar Reynolds-numbers. For each case fine meshes are selected, which ensure that the discretisation error is much smaller compared to the error due to turbulence modelling.

Periodic hills (PH), for which the flow is over a series of hills in a channel. Initially proposed by [32] this case has been studied both experimentally as well as numerically in detail. We use LES data from [33] for $Re = 10595$ (PH_{10595}) to apply SpaRTA and test the performance of the resulting models. In addition, we also use experimental data from [34] at a much larger $Re = 37000$ (PH_{37000}) in order to test the models outside the range of the training data. The numerical mesh consists of 120×130 cells. Cyclic boundary conditions are used at the inlet and outlet. The flow is driven by a volume forcing defined to produce a constant bulk velocity.

Converging-diverging channel (CD). A DNS study of the flow within a channel, in which an asymmetric bump is placed, exposed to an adverse pressure gradient was performed by [35] for $Re = 12600$ (CD_{12600}). The flow shows a small separation bubble on the lee-side of the bump, which is challenging for RANS to predict. The numerical mesh consists of 140×100 cells. The inlet profile was obtained from a channel-flow simulation at equivalent Re .

Curved backward-facing step (CBFS). In [36] a LES simulation of a flow over a gently-curved backward-facing step was performed at $Re = 13700$ ($CBFS_{13700}$). Similar to PH also for this flow the mean effect of separation and reattachment dynamics is the objective. The numerical mesh consists of 140×150 cells. The inlet was obtained from a fully-developed boundary layer simulation.

Despite the simple geometries, the mean effect of the separation and reattachment dynamics of a flow on a curved surface is a challenging problem for steady-RANS approaches. Especially, PH serves as an important testbed for classical and data-driven approaches for turbulence modelling, e.g. [2, 37], but also the other two have been introduced with the purpose of closure investigation.

5 Results

The method SpaRTA introduced in Section 3 is applied to the three test cases of Section 4. The models resulting from the model-discovery are presented and their mean-squared error on the training data is evaluated. In order to identify the models with the best predictive capabilities, we carry out cross-validation of the resulting models using CFD [30]: Models identified given training data of one case are used for CFD simulations of the remaining two case. For each case a single model is chosen as the best-performing one. Finally, the three resulting models are tested in a *true* prediction for the flow over periodic hills at $Re = 37000$.

5.1 Discovery of models and their error on training data

The goal of the model-discovery is to identify an ensemble of diverse models with small coefficients, varying in model-structure (complexity) and accuracy. Such an ensemble is better-suited for the cross-validation on unseen test cases, than a selection of the best models given only the training data. The sparse-regression for b_{ij}^Δ applied to the three test cases PH_{10595} , $CBFS_{13700}$ and CD_{12600} resulted in 52, 114 and 136 distinct models respectively, see Figure 3a for the results for PH_{10595} . It can be observed that, in general, an increase in complexity of a model leads to a reduction of the error. But, the bias introduced through the ridge regression of the inference step in SpaRTA, see Section 3.3, shrinks the model coefficients. If a coefficient is associated with a candidate function with a much lower magnitude compared to others, due to shrunk coefficients it becomes less relevant. The result is a staircase structure of the error: Models

Table 2: Best-predictive models with rank (index i, j, k in Figure 7) and normalised error on velocity $\epsilon(U)/\epsilon(U^o)$ for different cases.

Model	PH ₁₀₅₉₅		CD ₁₂₆₀₀		CBFS ₁₃₇₀₀	
	index i	$\epsilon(U)/\epsilon(U^o)$	index j	$\epsilon(U)/\epsilon(U^o)$	index k	$\epsilon(U)/\epsilon(U^o)$
$M^{(1)}$	(1.)	0.22287	(19.)	0.21146	-	0.30413
$M^{(2)}$	-	0.38867	(1.)	0.20828	(26.)	0.40154
$M^{(3)}$	(3.)	0.22744	-	0.22422	(1.)	0.30655

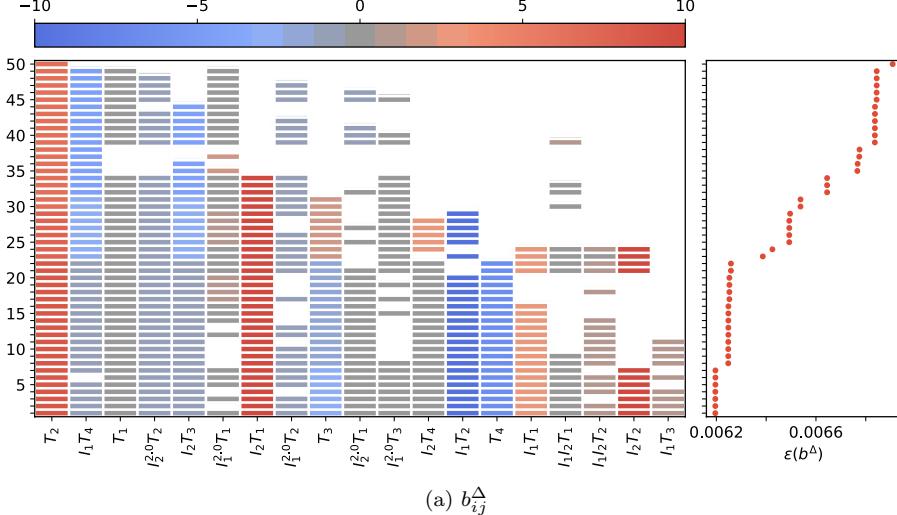
show a different form but have similar error. This pattern can also be observed for the other cases and becomes even more prominent for the models regressing R . For this target the model discovery resulted in 18 and 19 distinct model forms for CD₁₂₆₀₀ and PH₁₀₅₉₅ respectively, see Figure 3b for results of case PH₁₀₅₉₅. For CBFS₁₃₇₀₀ only three models have been found. We identify $T_{ij}^{(1)}$, $I_1 T_{ij}^{(1)}$ and $I_2 T_{ij}^{(1)}$ as the relevant candidates to regress R , and models combining all three give the lowest error per test case.

In order to reduce the redundancy within the ensemble of models regressing b_{ij}^Δ and R we select only a representative subset of models. This ensemble needs to acknowledge the hierarchical structure of diverse model-forms and their accuracy. In an ad-hoc way, we hand-select 5 models for b_{ij}^Δ and 3 for R , except for CBFS₁₃₇₀₀ only 1. The ensembles of selected models are shown in Figure 4 and Figure 5. The result is a hierarchical spectrum of models regressing the training data varying in complexity and error. Given this ensemble we study the performance of each model for predictions in the next section.

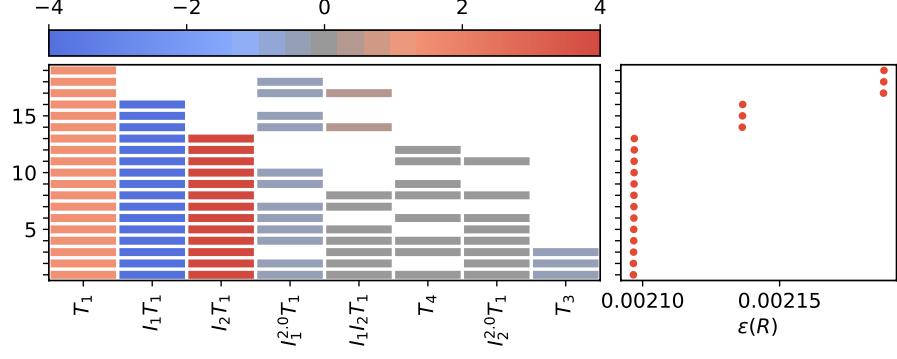
5.2 Cross-validation using CFD

Cross-validation tests how well models identified on training data perform on unseen test cases [30]. This assessment allows to determine the best-predictive models from a set. As stated above, the role of the frozen, training data should not be overcharged, so that we cross-validate using CFD. By doing so, we can assess the validity of SpaRTA as a tool for model discovery as well as the predictive performance of the identified models outside of their training set.

The selected correction models regress b_{ij}^Δ and R individually and can also be applied individually for predictions when implemented in the solver, i.e. a model correcting b_{ij}^Δ can be used without a correction of R and vice-versa. This gives us 8 models per training data for PH₁₀₅₉₅ and CD₁₂₆₀₀ and 6 for CBFS₁₃₇₀₀. Also, we can study their combined effect. With 5 models for b_{ij}^Δ and 3 for R we have additional 15 possible combinations, which makes in total 23 distinct models for the training data of PH₁₀₅₉₅ and CD₁₂₆₀₀ and 11 distinct model combinations for the training data of CBFS₁₃₇₀₀. For the cross-validation in the following, we conduct in total 35 for test cases PH₁₀₅₉₅ and CD₁₂₆₀₀ and 47 simulations for test case CBFS₁₃₇₀₀ including the baseline simulation with the uncorrected $k\text{-}\omega$ SST.

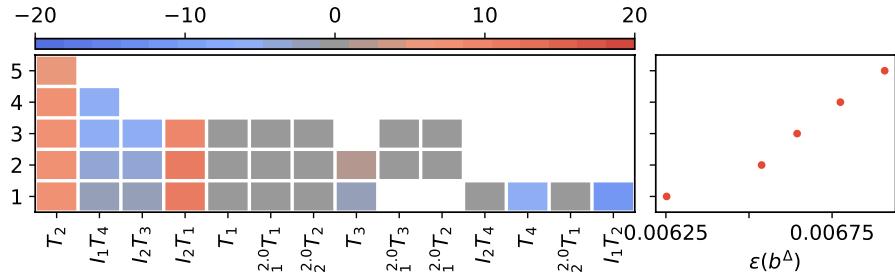
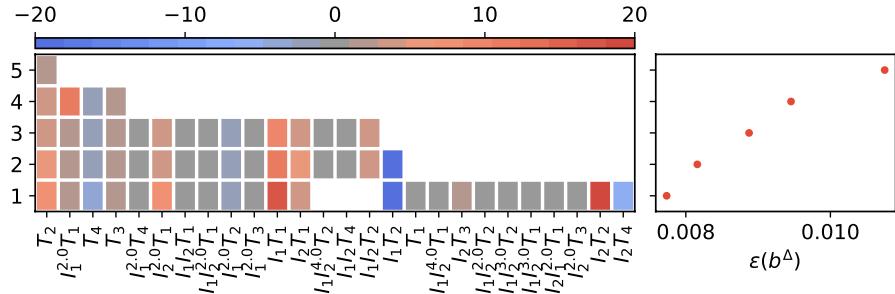
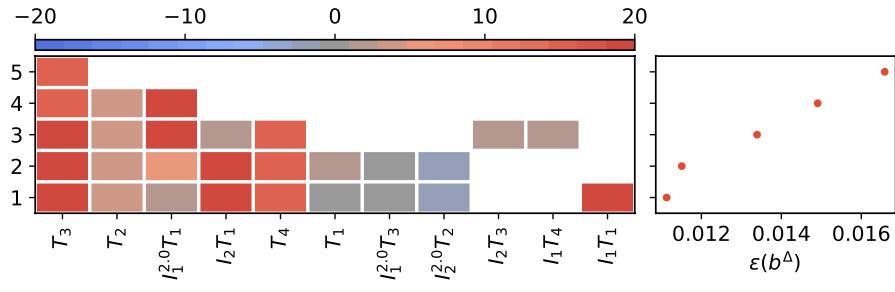


(a) b_{ij}^Δ



(b) R

Figure 3: Model-structure of all discovered models using SpaRTA and mean-squared error on training data for PH₁₀₅₉₅. The matrix (l.) shows the values of the active (coloured) candidate functions (x-axis) for each model M_i with model index i (y-axis). The mean-squared error between the frozen data b_{ij}^Δ and the model is also shown (r.).

(a) PH₁₀₅₉₅(b) CD₁₂₆₀₀(c) CBFS₁₃₇₀₀Figure 4: Selected models on frozen data b_{ij}^Δ .

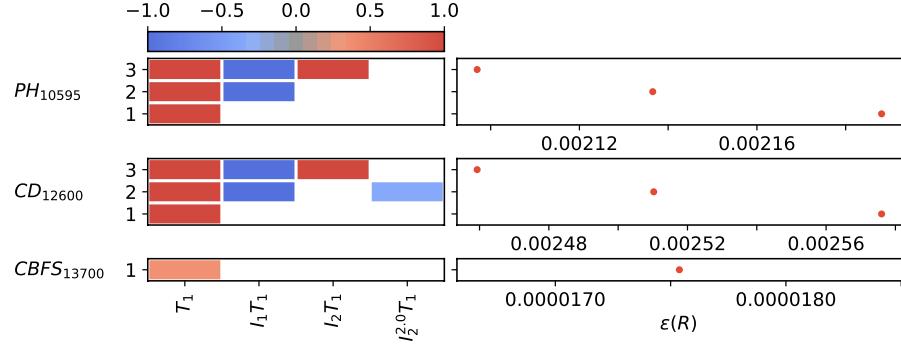


Figure 5: Selected models on frozen data R .

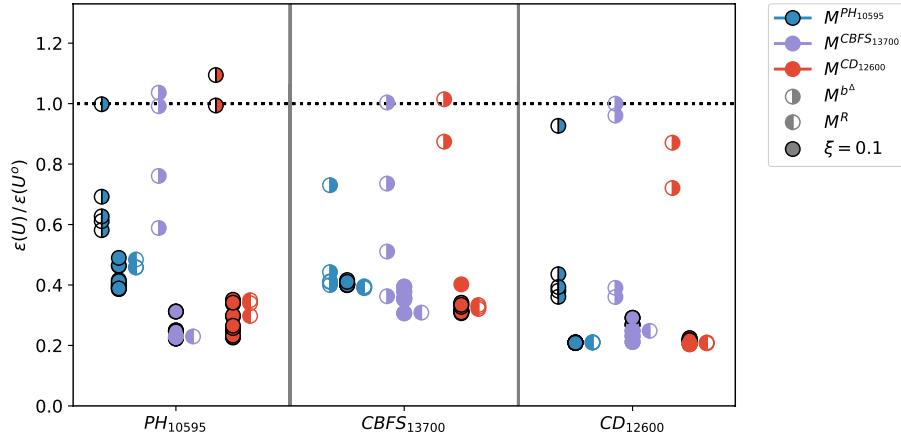


Figure 6: Mean-squared error of velocity vector of each correction model normalised by the mean-squared error of the baseline $k-\omega$ SST. The colour indicates on which high-fidelity data the models have been identified. Full circles represent simulations using both corrections, while left-/right-filled circles represent simulations using only correction for R or b_{ij}^Δ respectively.

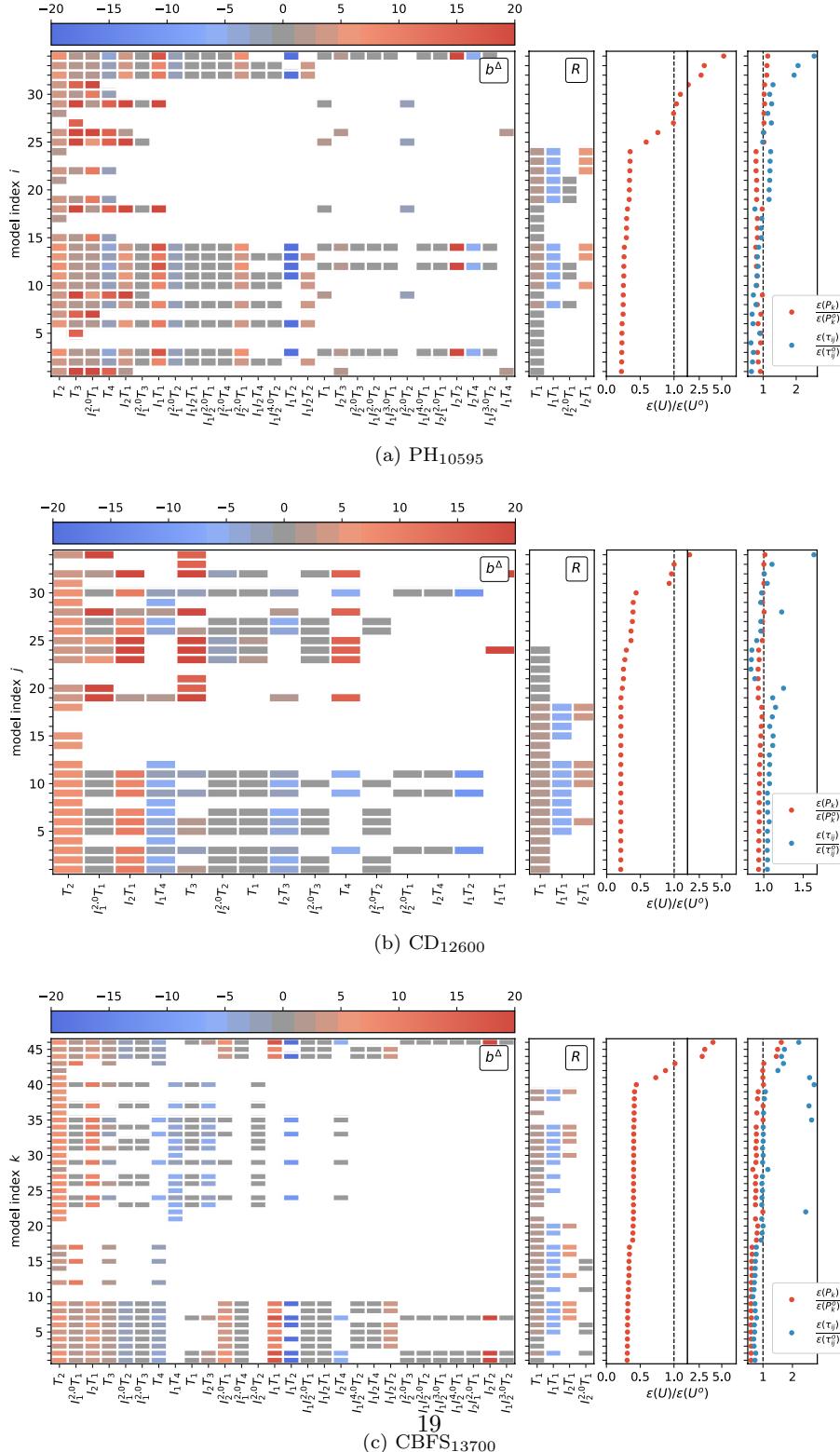


Figure 7: The two matrices ($l.$) show the models M_i for b_{ij}^Δ and R . The mean-squared error in velocity U , production P_k and Reynolds-stress τ_{ij} normalised by the mean-squared error of the baseline $k-\omega$ SST model is also shown (mid to right).

In Figure 6 the mean-squared error of each model on the velocity field $\epsilon(U)$ normalised with the mean-squared error of the baseline $\epsilon(U^o)$ is shown. The type of model, whether it is providing a correction both for b_{ij}^Δ and R or for each one individually, and from which training data it originated from, is emphasised by a unique marker form and color combination. Whether the correction for b_{ij}^Δ needs to be scaled with $\xi = 0.1$ to achieve convergence, see Section 3.3, is indicated by a black marker edge. Most of the models show a good or even substantial improvement over the baseline. But, for the set of models, only providing a correction for b_{ij}^Δ , most but not all lead to an improvement of the resulting velocity field. In contrast to that, if only a correction for R is deployed, the result is a consistent, substantial improvement across all test cases. Using both a model for b_{ij}^Δ and R leads to a further improvement, except for test case CBFS₁₃₇₀₀. Surprisingly, the best model per test case is not always identified on the associated training data. While this expectation holds for the cases CBFS₁₃₇₀₀ and CD₁₂₆₀₀ it is not true for PH₁₀₅₉₅, for which the other two training sets deliver significantly better performing models. In general, the data of CD₁₂₆₀₀ and CBFS₁₃₇₀₀ provide models, which are well performing on all test cases presented.

In Figure 7, both the error and the model structure for the correction of b_{ij}^Δ as well as for R is shown. The models are ordered according to the mean-squared error on the stream-wise velocity U . In line of the discussion of Figure 6 three groups can be identified: a few models, which lead to an increased error compared to the baseline; a small group of models per test case, which are equal or similar to the baseline; and the great majority of models, which result in an improvement. It can be observed how the error in the velocity is significantly reduced once a correction of R is used. The two other error plots in Figure 7 give an indication of the relative performance of the models compared to the baseline. The first is the mean-squared error of the total production P_k within the k equation and the second one is the mean-squared error of the Reynolds-stress τ_{ij} normalised by the baseline result. Following the rationale of correcting terms with the baseline model, improving these terms should lead to an improved velocity. For the cases CD₁₂₆₀₀ and CBFS₁₃₇₀₀, we see that the error for U and P_k reduce simultaneously for most of the models. Also the error in τ_{ij} shows a reduction when the error in U decreases, but not as significant as the error in P_k . For a group of models between model index $5 < j < 20$ for case CD₁₂₆₀₀, we see a jump in the error in P_k and τ_{ij} , while the error in the velocity is not changing compared to the neighbouring models. For case PH₁₀₅₉₅, we also observe a strong reduction of the P_k error for an active R correction. Surprisingly, for these the error in τ_{ij} increases. When the error in U is further reduced the error in τ_{ij} also decreases, but the error in P_k increases again. It can be observed, that the best models correct the velocity up to 5 times better in mean-squared error than the $k-\omega$ SST baseline model. This leaves still room for further improvement compared to the error using the frozen data sets, see Table 1. But, especially for case CBFS₁₃₇₀₀ the result is already very close to the possible correction provided by the frozen data at least for U .

Given this cross-validation assessment we select models $\mathbf{M}^{(i)} = (M_{b\Delta}^{(i)}, M_R^{(i)})^T$ based on the lowest $\epsilon(U)$ per case

$$\begin{aligned} M_{b\Delta}^{(1)} &= (24.94I_1^2 + 2.65I_2^1) T_{ij}^{(1)} + 2.96 T_{ij}^{(2)}, \\ &\quad + (2.49I_2^1 + 20.05) T_{ij}^{(3)} + (2.49I_1^1 + 14.93) T_{ij}^{(4)}, \\ M_R^{(1)} &= 0.4 T_{ij}^{(1)}, \end{aligned} \quad (25)$$

$$\begin{aligned} M_{b\Delta}^{(2)} &= T_{ij}^{(1)} (0.46I_1^2 + 11.68I_2^1 - 0.30I_2^2 + 0.37) \\ &\quad + T_{ij}^{(2)} (-12.25I_1^1 - 0.63I_2^2 + 8.23) \\ &\quad + T_{ij}^{(3)} (-1.36I_2^1 - 2.44) \\ &\quad + T_{ij}^{(4)} (-1.36I_1^1 + 0.41I_2^1 - 6.52), \\ M_R^{(2)} &= 1.4 T_{ij}^{(1)}, \end{aligned} \quad (26)$$

$$\begin{aligned} M_{b\Delta}^{(3)} &= T_1 (0.11I_1^1 I_2^1 + 0.27I_1^1 I_2^2 - 0.13I_1^1 I_2^3 + 0.07I_1^1 I_2^4 \\ &\quad + 17.48I_1^1 + 0.01I_1^2 I_2^1 + 1.251I_1^2 + 3.67I_2^1 + 7.52I_2^2 - 0.3) \\ &\quad + T_2 (0.17I_1^1 I_2^2 - 0.16I_1^1 I_2^3 - 36.25I_1^1 - 2.39I_1^2 + 19.22I_2^1 + 7.04) \\ &\quad + T_3 (-0.22I_1^2 + 1.8I_2^1 + 0.07I_2^2 + 2.65) \\ &\quad + T_4 (0.2I_1^2 - 5.23I_2^1 - 2.93), \\ M_R^{(3)} &= 0.93 T_{ij}^{(1)}, \end{aligned} \quad (27)$$

for which further details on the corresponding training data and the rank of the model on each test case are given in Table 2. Especially model $M^{(3)}$ performs very well both on CBFS₁₃₇₀₀ (rank 1.) and PH₁₀₅₉₅ (3.). While the rank of the others varies more between the test cases, they are still within the set of well-performing models with $\epsilon(U)/\epsilon(U^\circ) < 0.5$. Their predictions of stream-wise velocity U , k , the Reynolds-stress component τ_{xy} and the skin-friction coefficient C_f are shown in Figure 8 to 11 for the three test cases. As already stated for the error evaluated on the entire domain discussed above, these three models show an improvement of the spatial distribution of the predicted quantities in comparison to the baseline prediction of k - ω SST. Especially the velocity is well-captured for all three. While k is better identified compared to the baseline, we still observe a discrepancy between the predictions and the data. For PH₁₀₅₉₅ the three models do not fit the complex spatial structure especially in the shear-layer, but together encapsulate the data for most of the profiles. For CD₁₂₆₀₀ the models are underestimating k for $x < 7$ and overestimate it further downstream. For CBFS₁₃₇₀₀ the models also underestimate on the curved surface, but fit the data better than the baseline for $3 < x < 5$. The magnitude of the Reynolds-stress component τ_{xy} is underestimated on the curved surfaces of all test cases.

For PH₁₀₅₉₅ the models fail to fit the complex spatial structure especially within the separated shear-layer behind the hill and on the hill itself. The skin friction coefficient C_f and the associated separation and reattachment points are best captured by $M^{(1)}$ and $M^{(3)}$ for PH₁₀₅₉₅ and CBFS₁₃₇₀₀ and systematically under-estimated with $M^{(2)}$, i.e. a shorter recirculation zone. For CD₁₂₆₀₀, we observe a small recirculation zone as reported in the literature, but too far downstream. However, the baseline $k-\omega$ SST drastically over-predicts this zone and the model $M^{(2)}$ ignores it entirely.

Overall, the models $M^{(1)}$ and $M^{(3)}$ agree best with the data, which is in line with the global error on U in Table 2. The models are different in their form, but show similar error values and spatial structure across the test cases. Model $M^{(2)}$ tends to overestimate the magnitude of the quantities U , k and τ_{xy} and therefore predicts smaller or no separation bubbles. This model was identified using PH₁₀₅₉₅ as training data and, ignoring the specific structure for b_{ij}^Δ , has the largest coefficient for correcting R , see (26), which leads to larger k compared to the others, which is the reason for the systematic over-prediction.

In order to test how the models extrapolate to cases of larger Re , we predict the flow over periodic hills at $Re = 37000$, see Figure 12. Due to an increase of turbulence this case has a significantly shorter recirculation zone. For this *true* prediction throughout the domain the three models improve significantly compared to the baseline. Interestingly, the model $M^{(2)}$ is delivering the best fit of the data and the others tend to slightly underestimate it. Thus, taking the results of the cross-validation on the low- Re cases into account, the models show a weak Re -dependence, but overall robustness between the cases.

6 Conclusion and extension

In this work SpaRTA was introduced to discover algebraic models in order to correct the model-form error within the $k-\omega$ SST. For this novel machine learning method two additive terms, on the level of the stress-strain relation b_{ij}^Δ and within the turbulent transport equations R , were identified by means of k -corrective-frozen-RANS, for which the governing equations are evaluated given high-fidelity data of three cases of separating flows. It was validated that the computed terms are compensating the model-form error and reproduce the high-fidelity LES or DNS mean-flow data. Hence, k -corrective-frozen-RANS is a cost-efficient way to distill useful information directly from full-field data without the need of an inversion procedure.

Cross-validation of the discovered models using CFD was carried out to rank the models. While using both corrections for R as well as for b_{ij}^Δ lead to a systematic improvement of the predictions over the baseline, a correction only for R can already be enough to achieve sufficient results for the velocity field. For the best performing models on each case both the global error on U as well as the spatial structure on U , k and τ_{xy} was coherent. The models also performed well for the periodic hills flow at a much larger Re -number ($Re = 37000$). As the sparse regression is computationally inexpensive, SpaRTA allows for rapid

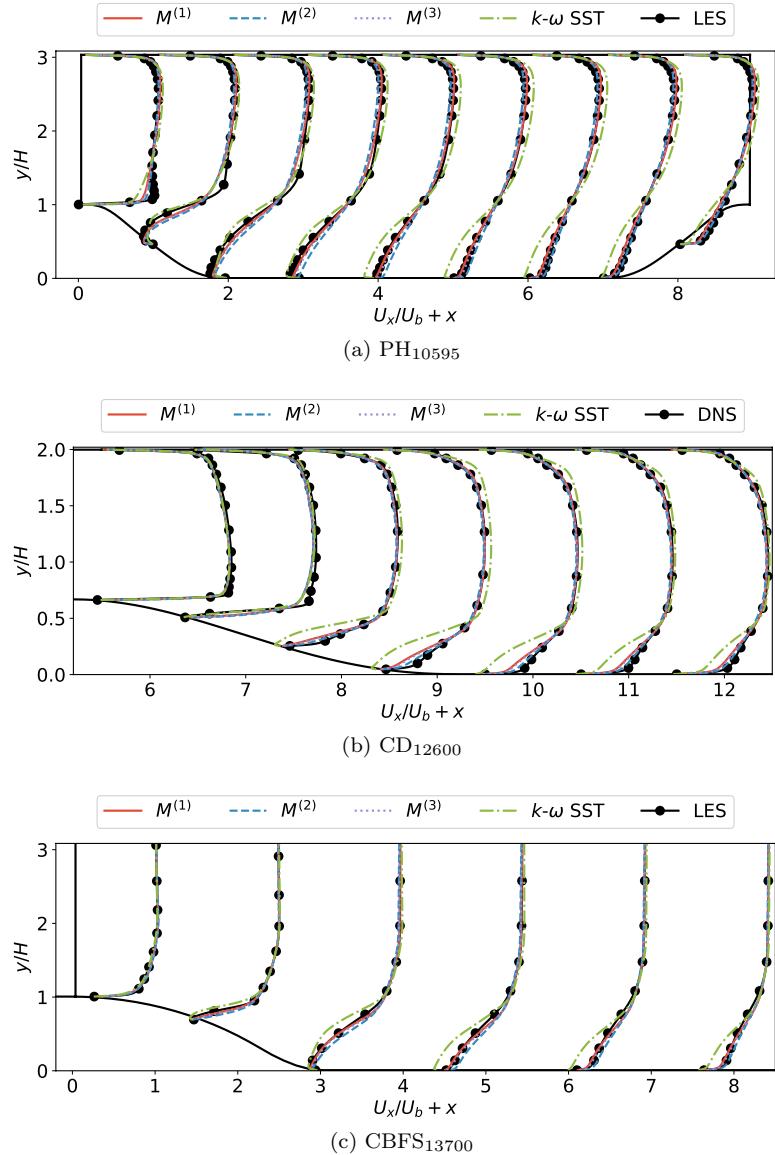


Figure 8: Predicted stream-wise velocity.

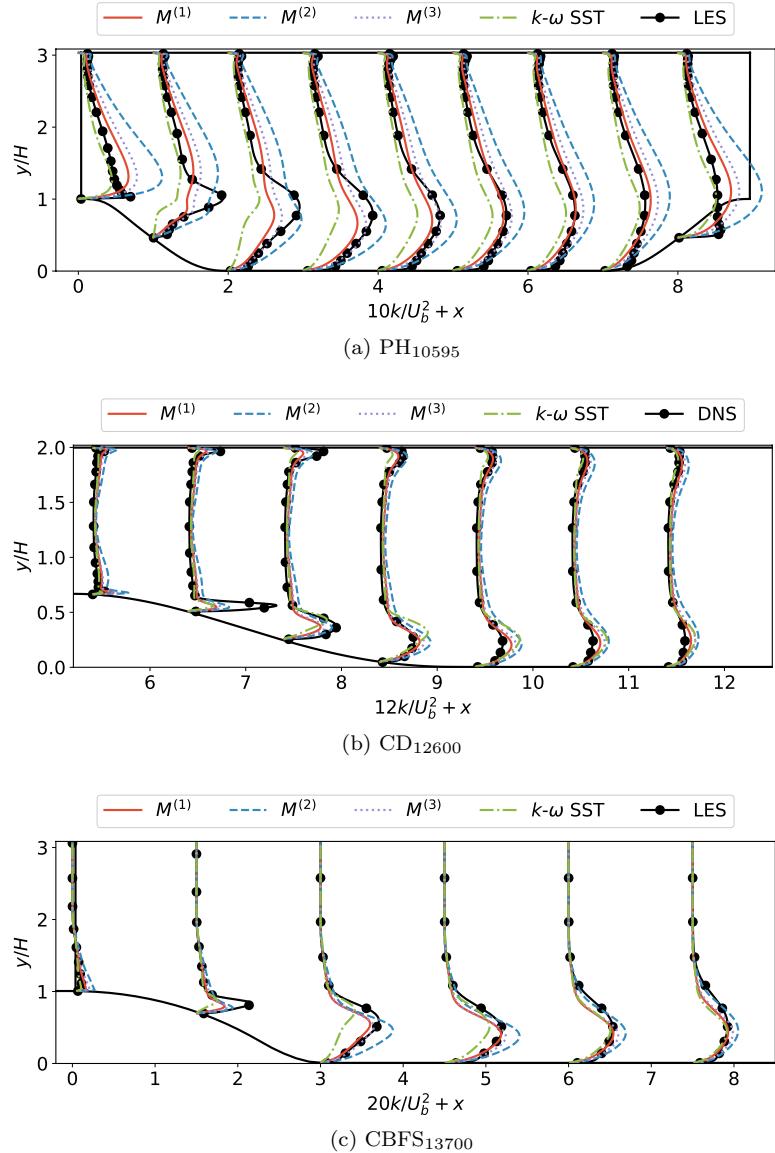


Figure 9: Predicted turbulent kinetic energy.

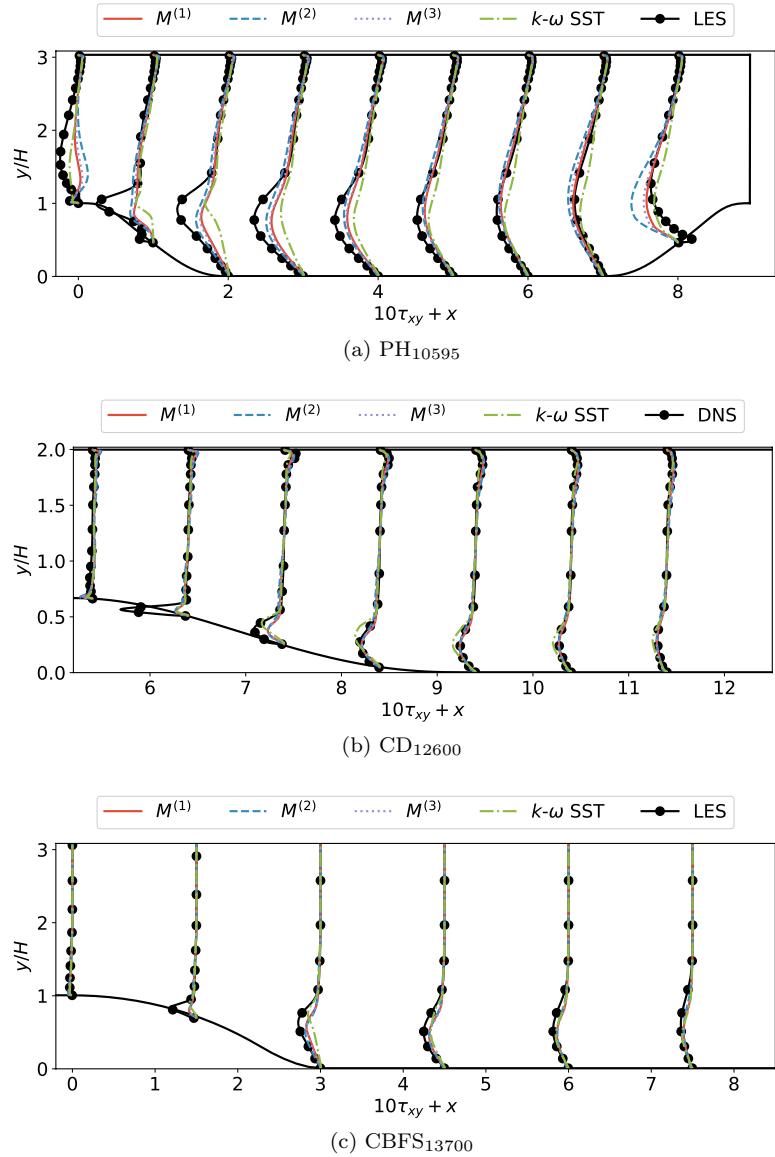


Figure 10: Predicted shear stress.

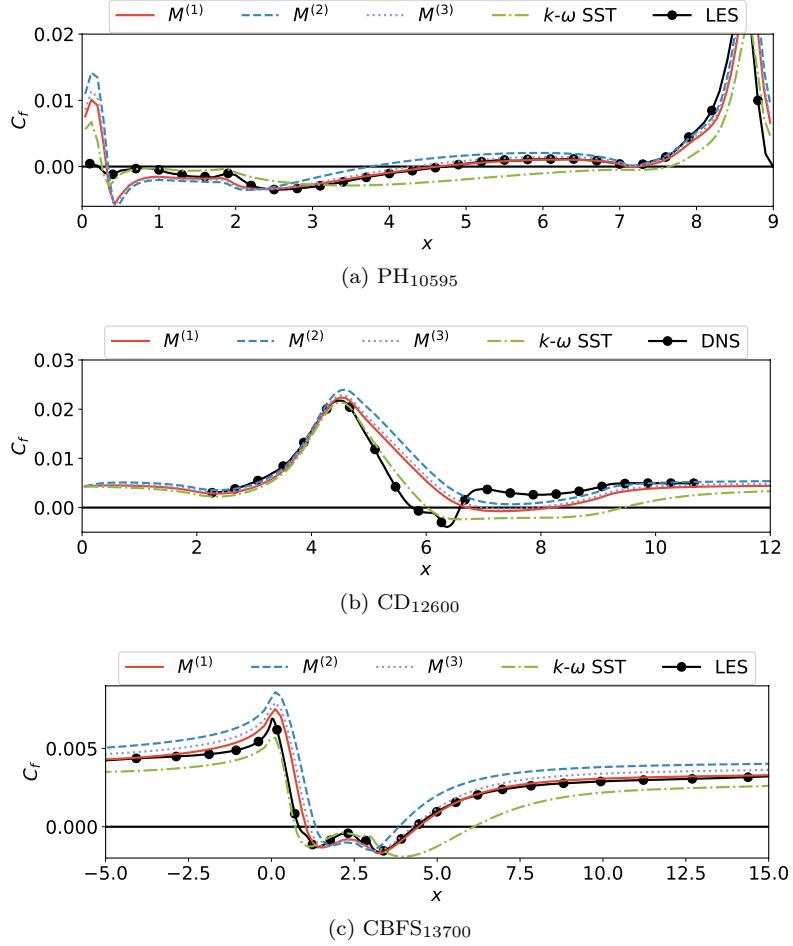


Figure 11: Predicted skin friction coefficient.

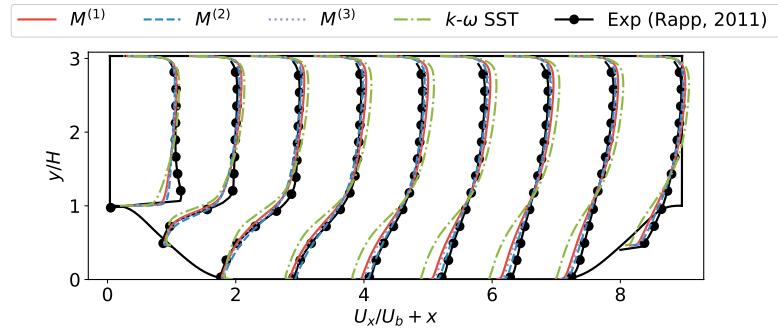


Figure 12: Flow over periodic hills at $Re = 37000$ using correction models compared to baseline $k-\omega$ SST and experimental data of [34].

discovery of robust models, i.e. a model trained for one flow may perform well for flows outside of the training range, but with similar features.

Overall, the present systematic study has shown the capabilities of SpaRTA to discover effective corrections to $k-\omega$ SST. Further work will focus on making the model-filtering and the inference step of SpaRTA more systematic and data-driven. We will also apply SpaRTA to a larger variety of flow cases in order to show its potential for rapid model discovery of corrections for industrial purposes.

acknowledgements

The authors wish to thank Richard Sandberg for sharing OpenFOAM code for comparison of implementations and Michael Breuer for providing the full-field LES and DNS simulation data for the periodic hill flow case.

Compliance with Ethical Standards

Funding This research has received funding from the European Unions Seventh Framework Programme under grant number ACP3-GA-2013-605036, UMRIDA project.

Conflict of Interest The authors declare that they have no conflict of interest.

References

- [1] J. Slotnick, A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, D. Mavriplis, CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences. Tech. Rep. March (2014). DOI 10.1017/CBO9781107415324.004
- [2] H. Xiao, P. Cinnella, Progress in Aerospace Sciences (2019). DOI 10.1016/j.paerosci.2018.10.001
- [3] K. Duraisamy, G. Iaccarino, H. Xiao, Annual Review of Fluid Mechanics **51**(1) (2019). DOI 10.1146/annurev-fluid-010518-040547
- [4] P.A. Durbin, Annual Review of Fluid Mechanics **50**(1) (2018). DOI 10.1146/annurev-fluid-122316-045020
- [5] B.D. Tracey, K. Duraisamy, J.J. Alonso, in *AIAA SciTech Forum 53rd AIAA Aerospace Sciences Meeting* (2015), January, pp. 1–23. DOI 10.2514/6.2015-1287
- [6] E.J. Parish, K. Duraisamy, Journal of Computational Physics **305** (2016). DOI 10.1016/j.jcp.2015.11.012
- [7] A.P. Singh, K. Duraisamy, Physics of Fluids **28**(045110) (2016). DOI 10.1063/1.4947045
- [8] A.P. Singh, K. Duraisamy, Z.J. Zhang, in *55th AIAA Aerospace Sciences Meeting* (2017). DOI 10.2514/6.2017-0993
- [9] J. Ling, A. Kurzawski, J. Templeton, Journal of Fluid Mechanics **807** (2016). DOI 10.1017/jfm.2016.615
- [10] J.L. Wu, H. Xiao, E. Paterson, (January) (2018). URL <http://arxiv.org/abs/1801.02762>

- [11] J. Weatheritt, R. Sandberg, Journal of Computational Physics **325**, 22 (2016). DOI 10.1016/j.jcp.2016.08.015
- [12] J. Weatheritt, R.D. Sandberg, International Journal of Heat and Fluid Flow **68** (2017). DOI 10.1016/j.ijheatfluidflow.2017.09.017
- [13] S.B. Pope, Journal of Fluid Mechanics **72**(2), 331 (1975). DOI 10.1017/S0022112075003382
- [14] T.B. Gatski, C.G. Speziale, Journal of Fluid Mechanics **254**, 59 (1993). DOI 10.1017/S0022112093002034
- [15] M. Leschziner, *Statistical Turbulence Modelling for Fluid Dynamics - Demystified: An Introductory Text for Graduate Engineering Students* (Imperial College Press, 2015)
- [16] S.B. Pope, *Turbulent Flows* (Cambridge University Press, 2000)
- [17] S. Wallin, Engineering turbulence modelling for CFD with a focus on explicit algebraic Reynolds stress models by. Phd thesis, Royal Institute of Technology Stockholm (2000)
- [18] J. Weatheritt, R.D. Sandberg, in *Conference: 11th International ERCOFTAC Symposium on Engineering Turbulence Modelling and Measurements*, vol. 2 (2017), vol. 2, pp. 2–7
- [19] H.D. Akolekar, J. Weatheritt, N. Hutchins, R.D. Sandberg, G. Laskowski, V. Michelassi, in *Proceedings of ASME Turbo Expo 2018* (Oslo, Norway, 2018), pp. 1–13
- [20] S.L. Brunton, J.L. Proctor, J.N. Kutz, Proceedings of the National Academy of Sciences **113**(15) (2016). DOI 10.1073/pnas.1517384113
- [21] S.H. Rudy, S.L. Brunton, J.L. Proctor, J.N. Kutz, Science Advances **3** (2017). DOI 10.1126/sciadv.1602614
- [22] J. Weatheritt, R.D. Sandberg, Journal of Ship Research (2019). DOI 10.5957/josr.09180053
- [23] H.G. Weller, G. Tabor, H. Jasak, C. Fureby, Computers in Physics **12**(6), 620 (1998). DOI 10.1063/1.168744
- [24] D.C. Wilcox, *Turbulence Modeling for CFD*, 3rd edn. (DCW Industries, Inc., 2006)
- [25] S.L. Brunton, J.N. Kutz, *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control* (Cambridge University Press, Cambridge, 2019). DOI 10.1017/9781108380690
- [26] T. McConaghy, in *Genetic Programming Theory and Practice IX. Genetic and Evolutionary Computation*. (Springer, New York, NY, 2011)
- [27] H. Zou, T. Hastie, Journal of the Royal Statistical Society. Series B: Statistical Methodology **67**(2), 301 (2005). DOI 10.1111/j.1467-9868.2005.00503.x
- [28] M. Quade, M. Abel, J. Nathan Kutz, S.L. Brunton, Chaos **28**(6) (2018). DOI 10.1063/1.5027470
- [29] N.M. Mangan, J.N. Kutz, S.L. Brunton, J.L. Proctor, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **473**(2203) (2017). DOI 10.1098/rspa.2017.0009

- [30] Bishop, Christopher M, *Pattern Recognition and Machine Learning* (Springer, 2006)
- [31] Y. Zhao, H.D. Akolekar, J. Weatheritt, V. Michelassi, R.D. Sandberg, (2019)
- [32] C.P. Mellen, J. Fröhlich, W. Rodi, in *16th IMACS World Congress*, (2000)
- [33] M. Breuer, N. Peller, C. Rapp, M. Manhart, *Computers and Fluids* **38**(2), 433 (2009). DOI 10.1016/j.compfluid.2008.05.002
- [34] C. Rapp, M. Manhart, *Experiments in Fluids* **51**(1), 247 (2011). DOI 10.1007/s00348-011-1045-y
- [35] J.P. Laval, M. Marquillie, in *Progress in Wall Turbulence: Understanding and Modeling*, vol. 14, ed. by M. Stanislas, J. Jimenez, I. Marusic (ERCOFTAC Series, 2011), vol. 14, pp. 203–209. DOI 10.1007/978-90-481-9603-6
- [36] Y. Bentaleb, S. Lardeau, M.A. Leschziner, *Journal of Turbulence* **13**(4) (2012). DOI 10.1080/14685248.2011.637923
- [37] S. Jakirlic, Extended excerpt related to the test case: "Flow over a periodical arrangement of 2D hills". Tech. Rep. June (2012)