FIELD INVERSION AND MACHINE LEARNING STRATEGIES FOR IMPROVING RANS MODELLING IN TURBOMACHINERY

FIELD INVERSION AND MACHINE LEARNING STRATEGIES FOR IMPROVING RANS MODELLING IN TURBOMACHINERY

A. Ferrero[1], *A. Iollo*[2], *F. Larocca*[3], *M. Loffredo*[4] *and E. Menegatti*[5]

[1] DIMEAS, Politecnico di Torino, Torino, Italy, andrea_ferrero@polito.it
[2] INRIA,CNRS, IMB and Université de Bordeaux, Talence, France, angelo.iollo@inria.fr
[3] DIMEAS, Politecnico di Torino, Torino, Italy, francesco.larocca@polito.it
[4]DIMEAS, Politecnico di Torino, Torino, Italy, marianna.loffredo@studenti.polito.it
[5]Osaka, Japan, eugenio.menegatti@gmail.com

ABSTRACT

Turbulence and transition modelling are critical aspects in the prediction of the flow field in turbomachinery. Recently, several research efforts have been devoted to the use of machine learning techniques for improving Reynolds-averaged Navier-Stokes (RANS) models. In this framework, a promising technique is represented by field inversion which requires to find an optimal correction field that minimises the error between numerical predictions and experimental data. In this work, Artificial Neural Networks and Random Forests are investigated as tools to generalise the correction provided by field inversion. An approach to automatically identify the regions where the correction model should be computed is proposed: this improves the fitting and reduces the calls to the model during the predictions. Furthermore, a correction-based weighting of the database is introduced in order to improve the training performances. The potential and the issues of the methods are investigated on a high-lift gas turbine cascade at low Reynolds number.

KEYWORDS

TRANSITION, RANS, MACHINE LEARNING, FIELD INVERSION

NOMENCLATURE

a Speed of sound
d Wall distance
g Correction coefficient for the production term
M_{2s}, Re_{2s} Isentropic exit Mach and Reynolds number
$M_c, \tilde{M}_c$ Convective Mach number and its local approximation
p Pressure
R^2 Determination coefficient
$\boldsymbol{u}$ Velocity
W_L, W_H Weigths for database classes
α, β_2 Inlet and exit angle
β Field inversion correction variable
ρ Density
$\nu, \tilde{\nu}$ Molecular and modified eddy viscosity
ω Vorticity
τ_i Dimensionless maximum shear stress in an incompressible mixing layer
ζ Kinetic losses

INTRODUCTION

The simulation of turbulent flows in turbomachinery is challenging because of the presence of complex phenomena like laminar-turbulent transition and separation which deeply affect the performances of the system. The use of scale-resolving simulations like Large Eddy Simulations and Direct Numerical Simulations offers the possibility to correctly capture important physical phenomena by reducing the amount of modelling. However, the computational cost of scale-resolving simulations limits their applicability during the design process which requires the evaluation of several geometrical configurations. In this framework, several research efforts have been devoted to the development of machine learning algorithms which can be used to analyse high-fidelity data (from experiments or scale-resolving simulations) in order to obtain correction terms which can be included in Reynolds-averaged Navier-Stokes (RANS) models.
A review of the current state of the art on the simulation of turbomachinery flows with particular attention to data driven modelling was proposed by Sandberg and Michelassi (2019). Several approaches have been proposed to improve the prediction capability of RANS models. Weatheritt and Sandberg (2016) suggested the use of Gene Expression Programming to introduce a non-linear correction in the stress-strain relationship adopted in RANS closures. Zhao et al. (2020) proposed an evolution of the previous technique in which RANS simulations are integrated in the training process and which showed good results on the prediction of wake mixing in turbomachinery. Wang et al. (2017) investigated the use of physics-informed machine learning techniques to reconstruct Reynolds stress modelling discrepancies starting from DNS data. Edeling et al. (2014) studied parameter variability in the k–ε RANS model by using Bayesian estimates.
Another promising strategy is represented by field inversion and machine learning (Duraisamy et al. (2015); Tracey et al. (2015); Parish and Duraisamy (2016); Singh et al. (2017)): the method requires the solution of an optimisation problem which provides a field of corrections to the source term of the RANS model. The correction field obtained by the field inversion must be then analysed in order to identify correlations between the correction and some flow features which can be used as inputs: in this way it is possible to generalise the results and perform actual predictions on different geometries and working conditions.
The use of this procedure for developing intermittency-based transition models in turbomachinery was investigated by Ferrero et al. (2020). Yang and Xiao (2020) applied the field inversion and machine learning strategy to the improvement of transition prediction with a four equations RANS models: they proposed to solve the inversion problem by using the regularising ensemble Kalman filtering as an alternative with respect to the adjoint approach which was used in the previous implementations of field inversion.
The previously described techniques represent some examples of the research efforts that have been recently devoted to the development of data-augmented RANS models. The interested reader is suggested to refer to the review carried out by Duraisamy et al. (2019) for a more general discussion.

In this work the field inversion approach is adopted for the improvement of RANS models on low pressure gas turbine cascades. Particular attention is devoted to the generalisation of the correction field provided by the inversion procedure. Both Artificial Neural Networks (ANNs) and Random Forests (RFs) are investigated as possible regression techniques which can predict the correction value as a function of some flow features. These techniques are evaluated in terms of both fitting capability and prediction capability by performing simulations for working

conditions not included in the training database. A preliminary analysis of the correction field suggested the use of a sensor to identify the regions where the correction is necessary: in this way the training dataset is reduced and, during the predictions, the data-driven correction is applied only in a small subset of the computational domain.
Furthermore, the fitting performance of the different machine learning techniques are investigated by introducing a pre-processing step during the training: in particular, a correction-based weight is introduced in the training database in order to improve the fitting for the points where the correction is not negligible.
The prediction capability of these data-augmented RANS models is investigated on the T106c low pressure gas turbine cascade by simulating flow conditions characterised by values of Reynolds number not included in the training database.

RANS SIMULATION AND FIELD INVERSION FOR THE T106c CASCADE

In this work the attention is focused on the T106c low pressure gas turbine cascade in the working conditions investigated by Michálek et al. (2012) ($M_{2s} = 0.65$, $\alpha = 32.7°$). As the experiments confirmed, this cascade is characterised by a large open separation at low Reynolds numbers ($Re_{2s} < 10^5$) while a smaller separation followed by reattachment is obtained for higher Reynolds numbers.

Physical model

The compressible RANS equations are considered in this work. In particular, the Spalart-Allmaras (SA) closure implemented according to Allmaras and Johnson (2012) is chosen. This model was not developed for low Reynolds number flows in the transitional regime but rather for fully turbulent high Reynolds number flows. For this reason, the baseline model is expected to perform poorly when applied to the simulation of the flow field in the T106c low pressure gas turbine cascade: it represents a good starting point to verify whether the field inversion procedure can introduce significant improvements.
The model is implemented without the trip term f_{t1} and the transition delay term f_{t2}, defined by Allmaras and Johnson (2012). The effect of the term f_{t2} on low Reynolds number flows in the T106c cascade was discussed by Ferrero et al. (2019).
The experimental configuration was characterised by a very low turbulence intensity (0.9%). In order to approximate such a condition the inlet eddy viscosity was set to $\tilde{\nu}/\nu = 0.1$ where $\tilde{\nu}$ and ν represent the modified eddy viscosity and the molecular kinematic viscosity, respectively. An ideal gas with constant specific heat ratio $\gamma = 1.4$ is considered. The viscosity is assumed constant and the Prandtl number is set to $Pr = 0.72$. The turbulent Prandtl number is set to $Pr_t = 0.9$. The solid walls are assumed adiabatic.

Discretisation

The governing equations are numerically solved by the method of lines. The discontinuous Galerkin finite element discretisation is used in space while time integration is performed by means of the first order linearised implicit Euler method. The solution inside each element is represented by a modal basis obtained by the application of the modified Gram-Schmidt orthonormalisation procedure to a set of monomials defined in the physical space, following the guidelines of Bassi et al. (2012). A third order accurate scheme is adopted for the spatial dis-

cretisation. Convective and diffusive fluxes are evaluated by means of an approximate Riemann solver and a recovery-based approach according to Ferrero et al. (2015).
The linear system resulting from the implicit time discretisation is solved in parallel by the GMRES solver with the additive Schwarz preconditioner provided by the PETSc library developed by Balay et al. (2020).
The computational domain is discretised by the Gmsh tool developed by Geuzaine and Remacle (2009) with the Frontal-Delaunay for Quads algorithm .

Field inversion

The field inversion procedure is applied to the T106c cascade at two values of Reynolds number: $Re_{2s} = 8 \cdot 10^4$ and $2.5 \cdot 10^5$. The procedure allows to find an optimal correction field $g(\beta(\boldsymbol{x}))$ which multiplies the production term in the Spalart-Allmaras transport equation:

$$\frac{\partial \rho\hat{\nu}}{\partial t} + \nabla\cdot(\rho\boldsymbol{u}\hat{\nu}) = \rho\left[g(\beta)\tilde{P} - \tilde{D}\right] + \frac{1}{\sigma}\nabla\cdot(\rho(\nu+\hat{\nu})\nabla\hat{\nu}) + \frac{c_{b2}}{\sigma}\rho(\nabla\hat{\nu})^2 - \frac{1}{\sigma}(\nu+\hat{\nu})\nabla\rho\cdot\nabla\hat{\nu} \quad (1)$$

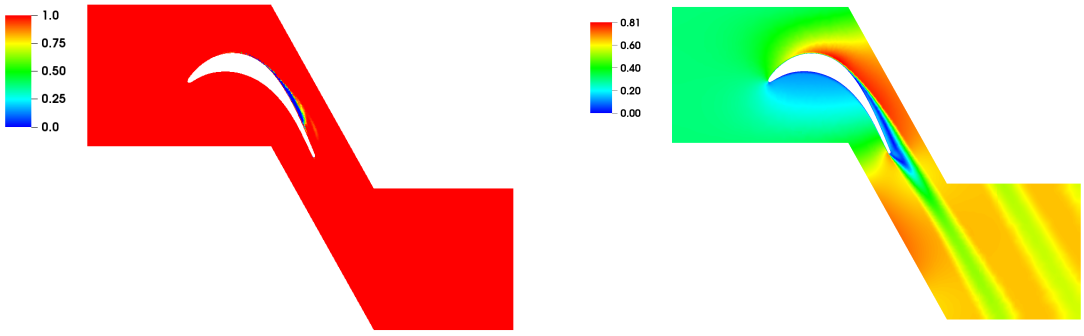
Here, ρ, $\tilde{\nu}$, $\boldsymbol{u}$, $\tilde{P}$, $\tilde{D}$ represent density, modified eddy viscosity, velocity, production and destruction terms, as defined by Allmaras and Johnson (2012). The constants c_{b2} and σ are set to the standard values reported by Allmaras and Johnson (2012).
The correction g alters the magnitude of the production term. The original SA model tends to overpredict the eddy viscosity in the T106c cascade at the considered Reynolds number and so it is not suitable to describe the laminar separation and the following transition. For this reason, the correction g is assumed to vary in the interval $[0, 1]$: in this way it acts as an intermittency function and can deactivate the turbulence model in the laminar boundary layer. On the contrary, the original turbulence model is recovered where $g = 1$.
In the original works of Parish and Duraisamy (2016) and Singh et al. (2017) the correction was chosen as $g(\beta) = \beta$ with β unlimited: this means that the optimisation procedure was free to choose any value for the correction factor. Ferrero et al. (2020) observed that a more robust approach, which is still suitable to improve the baseline model, is represented by the following choice:

$$g(\beta) = \begin{cases} 0 & \text{if } \beta <= 0 \\ 3\beta^2 - 2\beta^3 & \text{if } 0 < \beta < 1 \\ 1 & \text{if } \beta >= 1 \end{cases} \quad (2)$$

which represents a smooth approximation of the ramp function between 0 and 1.
The correction field is determined by solving an optimisation problem driven by the following goal function G:

$$G = \int_w (M_s - M_s^{exp})^2 dl + \lambda \int_\Omega (\beta - 1)^2 d\Omega \quad (3)$$

where the first integral represents the L2-norm of the error on the isentropic wall Mach number distribution (limited to the suction side of the blade) while the second integral is a Tikhonov regularisation. This last term is introduced in order to penalise unnecessary corrections and to regularise the problem. The penalisation constant λ is here set to 10^{-3}. A discussion on the choice of the penalisation constant is reported by Ferrero et al. (2020).
The optimisation problem is solved by means of the constant step gradient descent method. The size of the optimisation problem required for field inversion is related to the size of the



(a) (b)

Figure 1: Optimal correction factor g (a) and Mach number (b) in the solution provided by field inversion at $Re_{2s} = 8 \cdot 10^4$.

mesh since the optimal correction factor must be computed in each quadrature point of the domain. This makes the procedure quite expensive since each function evaluation requires a fully converged steady RANS solution. The gradient of the goal function with respect to the correction field is evaluated by means of the adjoint approach. More technical details on the application of the field inversion procedure on the T106c flow can be found in the work of Ferrero et al. (2020).

The field inversion procedure is here applied at the T106c cascade for two different values of the Reynolds number: $Re_{2s} = 8 \cdot 10^4$ and $2.5 \cdot 10^5$. The Mach field and the correction field $g(h(x))$ are reported in Figure 1 for the case at $Re_{2s} = 8 \cdot 10^4$: the results show that the correction is essentially activated only on the suction side in order to allow the laminar separation which would not be captured by the original SA model.

An example of the goal function evolution during the optimisation process is reported in Figure 2 for a configuration at $Re_{2s} = 8 \cdot 10^4$. The plot shows that after approximately 20 steps of the gradient method the goal function reaches a minimum. It is important to remember that for each step of the gradient method it is necessary to reach a steady RANS solution. This means that in this example the cost of the field inversion procedure is equivalent to approximately 20-30 RANS steady simulations for each considered working condition. However, each step of the gradient method requires the solution of a RANS which contains only small perturbations with respect to the previous step, especially after the first iterations. As a result, the RANS simulations can be performed by integrating in time with a very large CFL number and so they converge relatively quickly.

MACHINE LEARNING

The application of the field inversion procedure for a single value of the Reynolds number produces a large amount of data. The mesh used for these simulations contains 40436 elements and, since a third order accurate DG scheme is used, there are 6 degrees of freedom per equation in each element and 9 volume quadrature points in each element. In each quadrature point the correction factor and all the conservative variables (and their gradients) are available: this means that each application of the field inversion procedure gives a database with $40436 \times 9 = 363924$ points.

In this work the field inversion is independently applied for two values of Reynolds number:

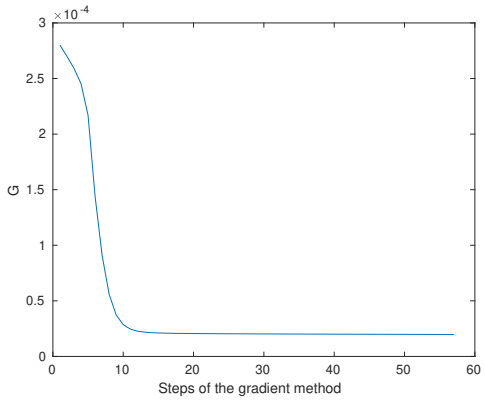


Figure 2: Goal function evolution in the field inversion procedure.

the results are collected in a global database with $2 \times 363924 = 727848$ points.
This global database is analysed by different machine learning techniques in order to identify a functional relationship between some flow features and the correction factor. This step is fundamental to generalise the correction and to perform predictions on different working conditions and different geometries which are not considered during the field inversion procedure: the goal is to express the correction as $g(\mathbf{\Phi})$, where $\mathbf{\Phi}$ is a vector of some flow features. In this way it is possible to introduce the correction in the CFD solver and perform predictions on new configurations.

Artificial Neural Networks and Random Forests

An Artificial Neural Network (ANNs) is a regression tool obtained by a sequence of layers formed by several neurons. In the architecture considered in this work each neuron receives the information from all the neurons (plus a bias) from the previous layer: the signals are averaged with different weights and the results is processed by the sigmoid activation function of the neurons. The signal coming out from the neuron is transmitted to all the neurons of the following layer. Since the output is limited in the range $[0, 1]$ also the output layer uses the sigmoid activation function. The network is trained in Matlab by the Levenberg-Marquadt algorithm with a goal function based on the mean squared error. The training algorithm requires to split the database in 3 sets: training (75%), validation (10%) and test (15%). The training set is used to compute the means squared error which drives the training. The validation set is used to avoid overfitting by stopping the training when the prediction error on the validation set starts to increase. Finally, the test set is not used during the training but it is exploited to evaluate the prediction capability of the obtained network. The ANN introduces a scaling of the inputs/output in order to work with variables in the range [-1,1]: this is automatically done by normalising with respect to the minimum and maximum values observed in the database. According to the previous analysis carried out by Ferrero et al. (2020), a network with two hidden layers and 20 neuron per layer is chosen.

A Random forest (RF) is a regression tool based on an ensemble of decision trees. Each tree is able to predict the outputs by evaluating a set of conditions on the inputs. Each tree is trained on a random subset of the full training database. The final output of the random forest is obtained by averaging the predictions of the single trees. These features give good prediction

performances and can limit the risk of overfitting. The RF model adopted in this work contains 10 trees and it is trained by means of the Python Scikit-learn library by Pedregosa et al. (2011). The depth of the tree is not limited and the minimum number of samples required to split an internal node is set to 2. In order to use the same training data factor adopted for the ANN, the database is split in two sets: a training set (75%) and a test set (25%). The validation set is not required by the considered RF training algorithm.

A sensor for model activation

A preliminary analysis of the database showed that 97% of the points are characterised by a correction factor greater than 0.99. The percentage increases to 98% if all the points with a correction factor greater than 0.9 is considered. This means that in most of the domain the correction is not active and this is confirmed by the visualisation of the correction field reported in Figure 1a for the case at $Re_{2s} = 8 \cdot 10^4$. For this reason it is useful to pre-process the database in order to focus the attention to the regions where the correction is active. As a first attempt, a reduced database was selected by choosing only the points for which the wall distance is less than a certain threshold. However, this approach gives problems during the predictions because the model trained only close to the wall gives unacceptable predictions in the wake region far from the wall. For this reason, a less arbitrary approach was investigated in order to define a criterion based on physical quantities: in this way it is possible not only to select the region used for the training in the offline phase but also to understand whether a point needs to call the correction model during the predictions.
The chosen criterion is based on setting a threshold on an estimate of the convective Mach number M_c. In the study of shear layers, the convective Mach number is defined as:

$$M_c = \frac{|u_1 - u_2|}{a_1 + a_2} \tag{4}$$

where u_1 and u_2 represent the speed on the two sides of the shear layer and a_1 and a_2 represent the corresponding speeds of sound. The approximation used to estimate M_c is based on the simplifications proposed by Paciorri and Sabetta (2003) in the framework of a compressibility correction for the SA model for free-shear flows. They proposed to correlate the eddy viscosity growth rate in a compressible mixing layer to that in an incompressible flow using local variables approximations. As a result, the following non-linear equation is obtained from which a local estimate of the convective Mach number $\tilde{M}_c$ can be computed:

$$\tilde{M}_c^2 f_2(\tilde{M}_c) = \frac{1}{4\tau_i} \frac{\tilde{\nu}|\omega|}{a^2} \tag{5}$$

where a, τ_i and $|\omega|$ represent the speed of sound, the dimensionless maximum shear stress in an incompressible mixing layer and the vorticity magnitude. The parameter τ_i is set to the constant value 0.01 according to the self-preservation hypothesis cited by Paciorri and Sabetta (2003).
The correlation f_2 is defined as:

$$f_2(\tilde{M}_c) = 0.44 \left[1/(1 + 14\tilde{M}_c^5) \right] + 0.56 \tag{6}$$

The expression "convective Mach number" could be misleading because it seems to be related to the Mach number while $\tilde{M}_c$ represents instead a measure of the local shear. However, in order to remain consistent with the previous works in the literature this expression is adopted

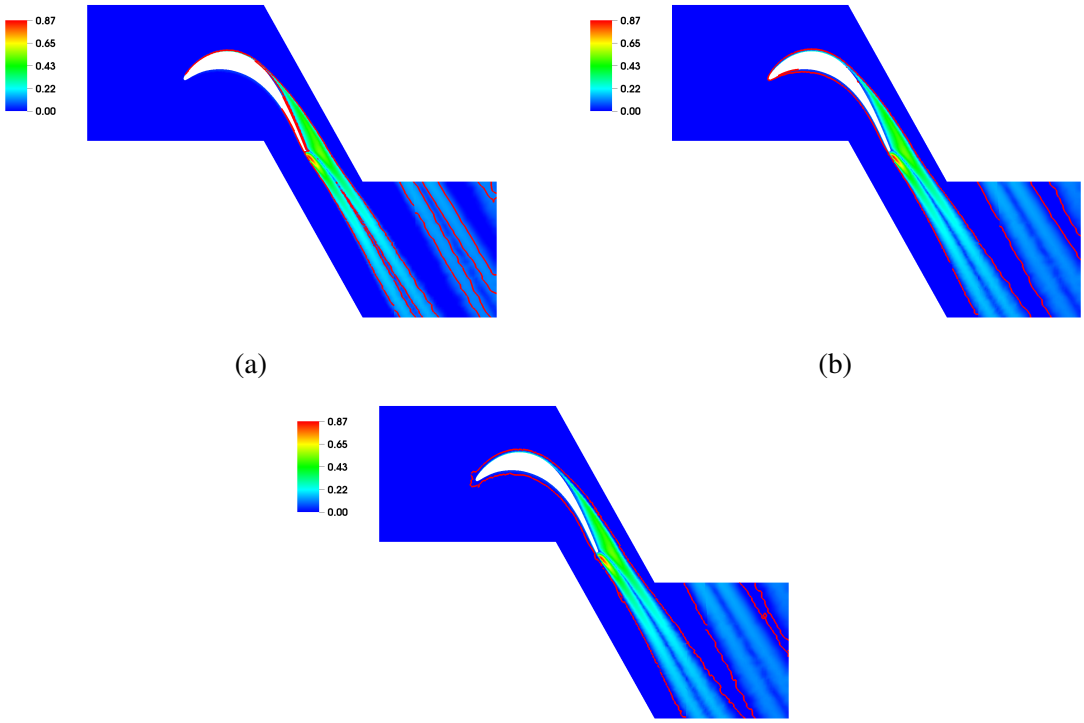


Figure 3: Convective Mach number $\tilde{M}_c$ field with isoline for $\tilde{M}_c = 0.1$ (a), $\tilde{M}_c = 0.01$ (b) and $\tilde{M}_c = 0.001$ (c).

also in this work.
The value of $\tilde{M}_c$ can be used to identify the regions where the model should be trained and then used during the predictions. In Figure 3 the $\tilde{M}_c$ field for the solution provided by field inversion at $Re_{2s} = 8 \cdot 10^4$ is reported. The isolevel for $\tilde{M}_c = 0.1$, $\tilde{M}_c = 0.01$ and $\tilde{M}_c = 0.001$ is highlighted in the plots of Figure 3a, 3b and 3c, respectively. The results show that for $\tilde{M}_c = 0.1$ only a portion of the boundary layer and only some regions of the wake are included. However, the full boundary layer and the entire wake are included for $\tilde{M}_c = 0.01$ and for $\tilde{M}_c = 0.001$. After a preliminary analysis, it was chosen to set the threshold as $\tilde{M}_c > 10^{-3}$ which means that only the inviscid external region is excluded while the near wall region and the wake region are fully included. When this threshold is applied to the database its size is reduced to the 30% of the original size.
The condition $\tilde{M}_c > 10^{-3}$ will also be evaluated at runtime during the predictions: when the condition is not satisfied the machine-learned model is not used and the correction is imposed as $g = 1$.
Finally, it is possible to observe that setting a threshold on $\tilde{M}_c$ is equivalent to setting a threshold on the term in the right hand side of Eq.5: in this way the limit condition could be directly computed from $\tilde{\nu}$, a and ω without the need to solve Eq.5 iteratively for $\tilde{M}_c$. In this work, the threshold is imposed on $\tilde{M}_c$ because of the tendency of this variable to assume values in the typical range reported in Figure 3 for the problems under investigation. However, also the right hand side of Eq.5 tends to assume values in a finite interval and so it should be equally easy to find a general threshold for this quantity.

Input selection

The choice of the inputs for the ANN and RF models is not trivial. There are some guidelines in the literature about this choice. For example, Wang et al. (2017) suggested to choose input variables which are Galilean-invariant, based on RANS computed variables and local. Furthermore, it is important to use non-dimensional variables as inputs in order to get a model which is scale-independent. Ferrero et al. (2020) proposed five input variables for the flow on the T106c cascade in the transitional regime. However, some of these variables were obtained as the ratio between quantities which can go to zero and so they assume values on a very wide range. In order to avoid numerical problems a logarithmic scale was used and small constants were added to avoid division by zero. In the present work, these poorly conditioned input variables are avoided. In particular, the following four input variables are chosen: $\tilde{\nu}/\nu$, f'_d, $\nabla p \cdot \boldsymbol{u}/\left(p|\boldsymbol{u}|/d\right)$ and $\tilde{M}_c$. The variable f'_d is a modified version of the shedding function defined by Ferrero et al. (2020). The third input is the adimensional streamwise pressure gradient which was found to be very effective in the regression step by Yang and Xiao (2020).

The scatter plots in Figure 4 show the distribution of the values assumed by the four inputs in the database. The plots do not suggest an evident trend between the correction factor and the inputs. The same conclusion is obtained by performing a linear correlation investigation. This can be done by computing the correlation coefficient CC which measures the linear dependency between two variables A and B:

$$CC(A,B) = \frac{cov(A,B)}{\sigma_A \sigma_B} \tag{7}$$

where $cov(A,B)$ is the covariance of the variables A and B and σ represents the standard deviation. The results of this linear correlation analysis are reported in Table 1. The correlation coefficients between the inputs and the correction factor are small: this means that they are not correlated by a linear relation.
In order to further investigate the relation between the chosen inputs and the correction factor, a non-linear correlation is assumed. An artificial neural network, which is capable of capturing strongly non-linear correlations, is used for a preliminary test. In particular, a network with 2 hidden layers and 20 neurons per layer is considered. First of all the network is trained by using all the four input variables. Then a leave-one-out strategy is applied, neglecting in turn one of the input variables and training the network on the remaining inputs. The results are reported in Table 2: the values of the coefficient of determination R^2 for the test portion of the database not used in training show that the small network is able to get a trend in the results while the previous linear analysis was unable to get a representative correlation. The Table shows also the reduction in the coefficient of determination ΔR^2 obtained by neglecting each of the inputs: they seem to give contributions with the same order of magnitude but the largest contribution seems to come from f'_d.

Training: sensor-based database reduction and oversampling

The use of the threshold on $\tilde{M}_c$ reduces the size of the database by excluding the external regions where the correction is set to one. Even in this way, most of the points in the database are related to corrections close to unity. Since the training of the ANN and the RF are based on the minimisation of the mean squared error, the training algorithms tend to fit the region where the correction is not active ($h \approx 1$) and do not focus on the region where strong corrections

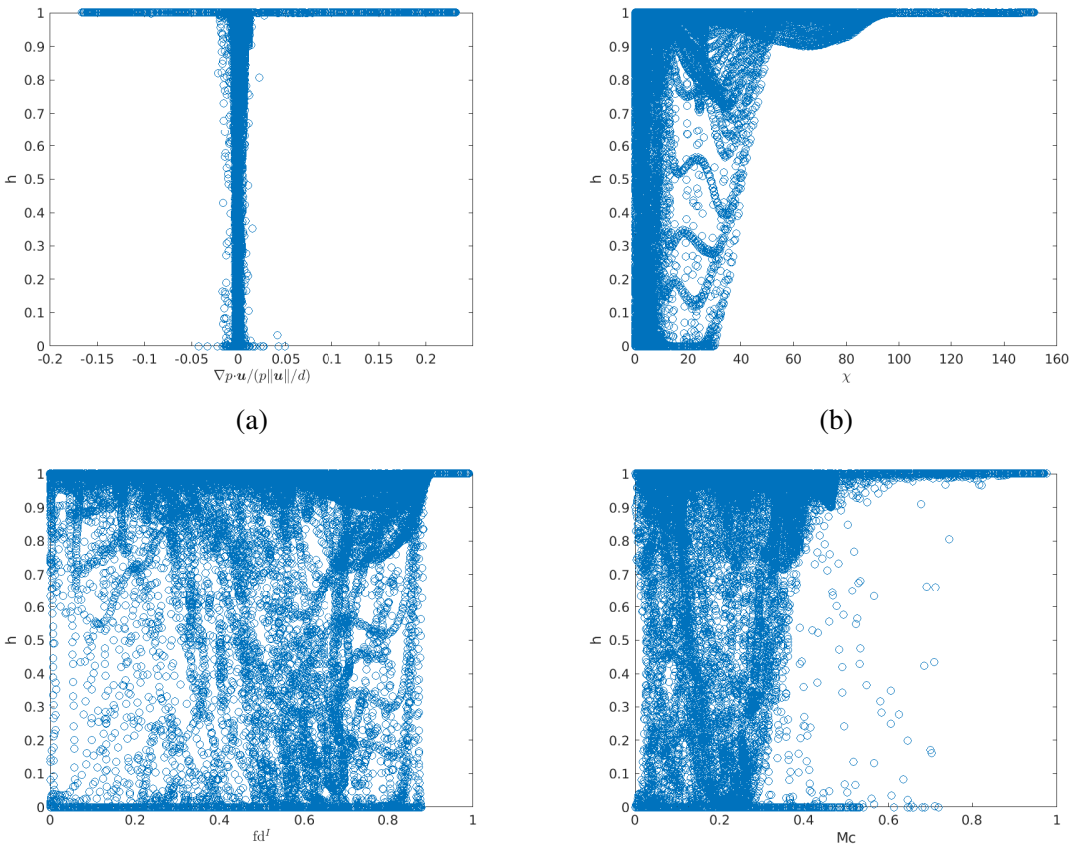


Figure 4: Scatter plots for the inputs in the database.

$CC(\nabla p \cdot \boldsymbol{u}/\left(p\lvert\boldsymbol{u}\rvert/d\right), g)$	-0.0185
$CC(\tilde{\nu}/\nu, g)$	0.0183
$CC(f'_d, g)$	0.1919
$CC(\tilde{M_c}, g)$	-0.1730

Table 1: Linear correlation coefficients.

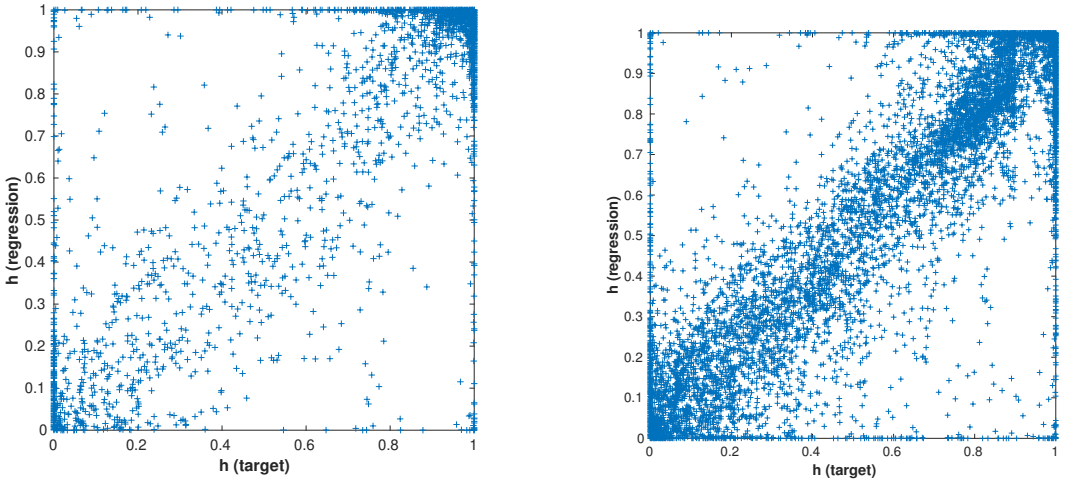
Input set	R^2	ΔR^2
$\tilde{\nu}/\nu$, f'_d, $\nabla p \cdot \boldsymbol{u}/\left(p\lvert\boldsymbol{u}\rvert/d\right)$,$\tilde{M_c}$	0.846	0.000
$\tilde{\nu}/\nu$, f'_d,$\tilde{M_c}$	0.834	0.012
f'_d, $\nabla p \cdot \boldsymbol{u}/\left(p\lvert\boldsymbol{u}\rvert/d\right)$,$\tilde{M_c}$	0.738	0.108
$\tilde{\nu}/\nu$, $\nabla p \cdot \boldsymbol{u}/\left(p\lvert\boldsymbol{u}\rvert/d\right)$,$\tilde{M_c}$	0.677	0.169
$\tilde{\nu}/\nu$, f'_d, $\nabla p \cdot \boldsymbol{u}/\left(p\lvert\boldsymbol{u}\rvert/d\right)$	0.693	0.153

Table 2: Leave-one-out analysis to quantify the contribution of the different inputs.

are applied ($h << 1$). In order to reduce this issue, the points in the database are split in two classes: low value corrections ($h < 0.9$) and high value corrections ($h >= 0.9$). Then two different weights are applied to the two classes, W_L and W_H. The weights are applied in this way: the original database is pre-processed and each point is repeated W_L or W_H times, depending on the class to which it belongs. In this way, it is possible to increase the influence of the strong corrections ($h < 0.9$) in the computation of the mean squared error which drives the training. This approach is also known as oversampling and is widely used for classification tasks in machine learning Ling and Li (1998).
In Table 3 the coefficients of determination R^2 for both the ANN and the RF are reported. The values refer to actual test predictions performed on a random subset of the database not used for training (15% of the database for ANN and 25% of the database for RF). The first two lines show that the introduction of the threshold on $\tilde{M_c}$ does not alter significantly the value of R^2. However, the size of the database is reduced by a factor 0.3 and this speeds up the training.
Then the weight W_L is increased to 5 and 10: in this way the points with the strong corrections acquire more influence during the training. The results show that the values of R^2 increases as W_L/W_H increases for both the ANN and the RF. This tendency is confirmed by the regression plots reported in Figure 5 and 6 in which the points cloud tends to the bisector for high values of W_L. These plots represent actual predictions since they are evaluated on the test subset of the data which is not used during the training.

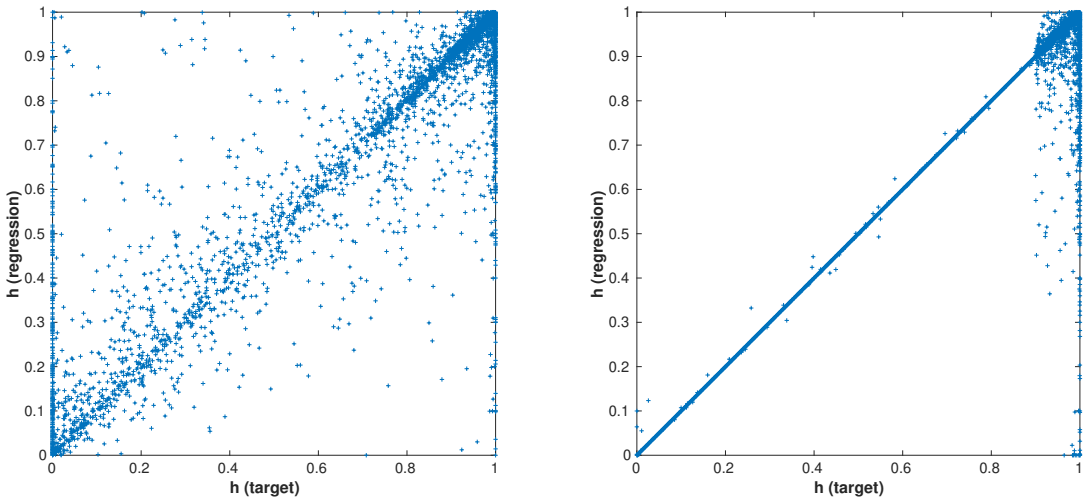
	R^2 with RF	R^2 with ANN
$\forall \tilde{M_c}, W_L = 1, W_H = 1$	0.926	0.846
$\tilde{M_c} > 10^{-3}, W_L = 1, W_H = 1$	0.922	0.843
$\tilde{M_c} > 10^{-3}, W_L = 5, W_H = 1$	0.990	0.942
$\tilde{M_c} > 10^{-3}, W_L = 10, W_H = 1$	0.995	0.940

Table 3: Effect of threshold on $\tilde{M_c}$ and data weighting on the training.



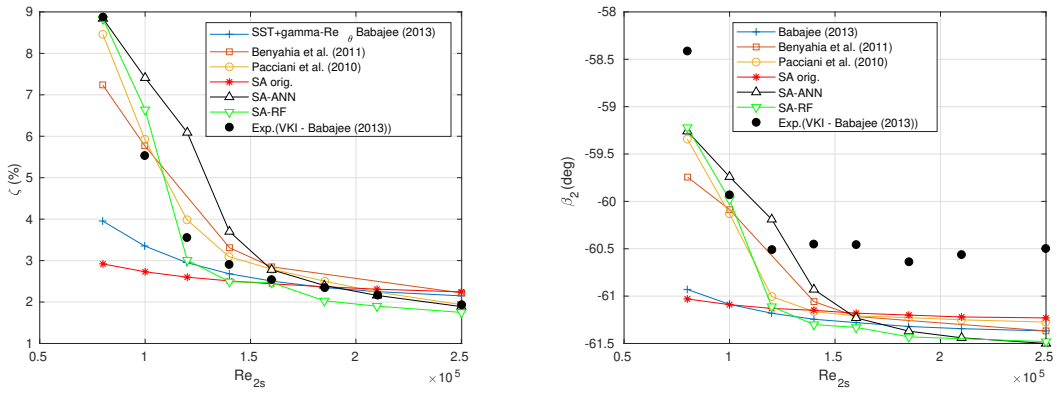
(a) (b)

Figure 5: Regression plot for ANN with full database and $W_L = W_H = 1$ (a) and reduced database with $W_L = 10$ and $W_H = 1$ (b)



(a) (b)

Figure 6: Regression plot for RF with full database and $W_L = W_H = 1$ (a) and reduced database with $W_L = 10$ and $W_H = 1$ (b)



(a) (b)

Figure 7: Mass-averaged kinetic losses (a) and exit angle (b) for different values of Reynolds number.

PREDICTIONS

The SA model augmented by the correction found by the analysis of the database is tested by simulating the flow in the T106c cascade at different values of Reynolds number: $8 \cdot 10^4$, $1 \cdot 10^5$, $1.2 \cdot 10^5$, $1.4 \cdot 10^5$, $1.6 \cdot 10^5$, $2.1 \cdot 10^5$ and $2.5 \cdot 10^5$. The mass averaged kinetic losses and exit angle are reported in Figure 7a and 7b, respectively. The plots show the experimental data and some numerical results from Babajee (2013), Pacciani et al. (2010) and Benyahia et al. (2011). It is possible to see that the SA model with the RF and ANN corrections (SA-RF and SA-ANN, respectively) perform better than the baseline SA model at low Reynolds number when the large laminar separation occurs. The results obtained by the SA-RF and SA-ANN models are in agreement for $Re_{2s} = 8 \cdot 10^4$ and $2.5 \cdot 10^5$ (which were used for the training) but they show significant differences in the actual predictions at $Re_{2s} = 1.2 \cdot 10^5$ and $1.6 \cdot 10^5$. This suggests that, even starting from the same training database, different machine learning strategies can lead to quite different results during the predictions.
While the machine-learning approach seems to improve the prediction capability in terms of losses, a large discrepancy is observed on the exit angle. This problem remains also at high values of Reynolds number for which there is no separation: in these conditions the different models should agree. It is indeed possible to observe that the numerical results obtained in this work and in other studies available in the literature seem to converge to an asymptotic value which presents an offset with respect to the experimental data. This anomaly will be investigated in the future also by testing the proposed approach on other test cases.

CONCLUSIONS

The use of ANNs and RFs for the identification of correlations between some flow features and the correction provided by field inversion is investigated. A local criterion based on the magnitude of the local shear is proposed to reduce the training data-set and to limit the use of the machine-learned model during the predictions. The training performance of both ANNs and RFs were improved by introducing a weight in the database in order to better fit the points in which the correction is not negligible: this oversampling strategy allows to focus the attention on the regions where the original RANS model fails. The final data-augmented models are tested in actual predictions for values of Reynolds numbers not considered in the training

database. The obtained results appear promising since it is possible to observe a significant improvement with respect to the baseline SA model at low values of Reynolds number.
However, it is important to keep in mind that the procedure is affected by different uncertainty contributions. First of all, the reference data are usually experimental measurements which are inevitably affected by uncertainty and this propagates through the procedure. Furthermore, the field inversion is performed by solving an optimisation problem: there are no guarantees that this problem has a unique solution and the optimisation could stop in a local minimum. Finally, the regression step tends to capture a trend in the database but it introduces an approximation, since the coefficient of determination R^2 is never unity. This means that different machine learning algorithms, like for example the ANN and the RF tested in this work, can generate different correction models: these differences can become evident when dealing with critical conditions like the lowest values of Reynolds number investigated in this work. However, the goal of the proposed procedure is not to find a universal RANS model but to suggest a procedure which allows to exploit the available reference data to obtain a thrustworthy model for a specific application. This is in line with the classical developments of RANS models: there are several RANS models in the literature which can provide quite different results in critical test cases. The generality of the machine-learned corrections can be ideally improved by adding more test cases in the training database.
Finally, there is an open-question which is shared by most machine learning approaches: when a regression shows a correlation between some flow features and the output there is no proof that it is a cause-effect relation. Future work will be devoted to introducing physical constraints in the machine-learned model and to find ways to prove the existence of a cause-effect relation between the chosen flow features and the correction.

ACKNOWLEDGEMENTS

Computational resources were provided by HPC@POLITO (http://www.hpc.polito.it).
We aknowledge the CINECA award under the ISCRA initiative, for the availability of high performance computing resources and support.

References

Allmaras, S. R. and Johnson, F. T. (2012). Modifications and clarifications for the implementation of the spalart-allmaras turbulence model. In *Seventh international conference on computational fluid dynamics (ICCFD7)*, pages 1–11.

Babajee, J. (2013). *Detailed numerical characterization of the separation-induced transition, including bursting, in a low-pressure turbine environment*. PhD thesis, Ecole Centrale de Lyon; Institut von Karman de dynamique des fluides (Rhode-Saint-Genèse, Belgique).

Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W. D., Karpeyev, D., Kaushik, D., Knepley, M. G., May, D. A., McInnes, L. C., Mills, R. T., Munson, T., Rupp, K., Sanan, P., Smith, B. F., Zampini, S., Zhang, H., and Zhang, H. (2020). PETSc users manual. Technical Report ANL-95/11 - Revision 3.14, Argonne National Laboratory.

Bassi, F., Botti, L., Colombo, A., Di Pietro, D. A., and Tesini, P. (2012). On the flexibility of agglomeration based physical space discontinuous galerkin discretizations. *Journal of Computational Physics*, 231(1):45–65.

Benyahia, A., Castillon, L., and Houdeville, R. (2011). Prediction of separation-induced transition on high lift low pressure turbine blade. In *ASME 2011 Turbo Expo: Turbine Technical Conference and Exposition*, pages 1835–1846. American Society of Mechanical Engineers.

Duraisamy, K., Iaccarino, G., and Xiao, H. (2019). Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51:357–377.

Duraisamy, K., Zhang, Z. J., and Singh, A. P. (2015). New approaches in turbulence and transition modeling using data-driven techniques. In *53rd AIAA Aerospace Sciences Meeting*, page 1284.

Edeling, W. N., Cinnella, P., Dwight, R. P., and Bijl, H. (2014). Bayesian estimates of parameter variability in the k–ε turbulence model. *Journal of Computational Physics*, 258:73–94.

Ferrero, A., Iollo, A., and Larocca, F. (2019). Rans closure approximation by artificialneural networks. In *ETC 2019-13th European Turbomachinery Conference on Turbomachinery Fluid Dynamics and Thermodynamics*.

Ferrero, A., Iollo, A., and Larocca, F. (2020). Field inversion for data-augmented rans modelling in turbomachinery flows. *Computers & Fluids*, 201:104474.

Ferrero, A., Larocca, F., and Puppo, G. (2015). A robust and adaptive recovery-based discontinuous galerkin method for the numerical solution of convection–diffusion equations. *International Journal for Numerical Methods in Fluids*, 77(2):63–91.

Geuzaine, C. and Remacle, J.-F. (2009). Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, 79(11):1309–1331.

Ling, C. X. and Li, C. (1998). Data mining for direct marketing: Problems and solutions. In *Kdd*, volume 98, pages 73–79.

Michálek, J., Monaldi, M., and Arts, T. (2012). Aerodynamic performance of a very high lift low pressure turbine airfoil (t106c) at low reynolds and high mach number with effect of free stream turbulence intensity. *Journal of Turbomachinery*, 134(6):061009.

Pacciani, R., Marconcini, M., Arnone, A., and Bertini, F. (2010). A cfd study of low reynolds number flow in high lift cascades. In *ASME Turbo Expo 2010: Power for Land, Sea, and Air*, pages 1525–1534. American Society of Mechanical Engineers Digital Collection.

Paciorri, R. and Sabetta, F. (2003). Compressibility correction for the spalart-allmaras model in free-shear flows. *Journal of Spacecraft and Rockets*, 40(3):326–331.

Parish, E. J. and Duraisamy, K. (2016). A paradigm for data-driven predictive modeling using field inversion and machine learning. *Journal of Computational Physics*, 305:758–774.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Sandberg, R. D. and Michelassi, V. (2019). The current state of high-fidelity simulations for main gas path turbomachinery components and their industrial impact. *Flow, Turbulence and Combustion*, 102(4):797–848.

Singh, A. P., Medida, S., and Duraisamy, K. (2017). Machine-learning-augmented predictive modeling of turbulent separated flows over airfoils. *AIAA Journal*, pages 2215–2227.

Tracey, B. D., Duraisamy, K., and Alonso, J. J. (2015). A machine learning strategy to assist turbulence model development. In *53rd AIAA Aerospace Sciences Meeting*, page 1287.

Wang, J.-X., Wu, J.-L., and Xiao, H. (2017). Physics-informed machine learning approach for reconstructing reynolds stress modeling discrepancies based on dns data. *Physical Review Fluids*, 2(3):034603.

Weatheritt, J. and Sandberg, R. (2016). A novel evolutionary algorithm applied to algebraic modifications of the rans stress–strain relationship. *Journal of Computational Physics*, 325:22–37.

Yang, M. and Xiao, Z. (2020). Improving the k–ω–γ–ar transition model by the field inversion and machine learning framework. *Physics of Fluids*, 32(6):064101.

Zhao, Y., Akolekar, H. D., Weatheritt, J., Michelassi, V., and Sandberg, R. D. (2020). Rans turbulence model development using cfd-driven machine learning. *Journal of Computational Physics*, page 109413.