

Article

Optimization of Turbine Blade Aerodynamic Designs Using CFD and Neural Network Models

Chao Zhang *  and Matthew Janeway

Siemens Digital Industries Software, Cypress, CA 90630, USA; matt.janeway@siemens.com

* Correspondence: chao.zhang1@siemens.com

Abstract: Optimization methods have been widely applied to the aerodynamic design of gas turbine blades. While applying optimization to high-fidelity computational fluid dynamics (CFD) simulations has proven capable of improving engineering design performance, a challenge has been overcoming the prolonged run-time due to the computationally expensive CFD runs. Reduced-order models and, more recently, machine learning methods have been increasingly used in gas turbine studies to predict performance metrics and operational characteristics, model turbulence, and optimize designs. The application of machine learning methods allows for utilizing existing knowledge and datasets from different sources, such as previous experiments, CFD, low-fidelity simulations, 1D or system-level studies. The present study investigates inserting a machine learning model that utilizes such data into a high-fidelity CFD driven optimization process, and hence effectively reduces the number of required evaluations of the CFD model. Artificial Neural Network (ANN) models were trained on data from over three thousand two-dimensional (2D) CFD analyses of turbine blade cross-sections. The trained ANN models were then used as surrogates in a nested optimization process alongside a full three-dimensional Navier–Stokes CFD simulation. The much lower evaluation cost of the ANN model allows for tens of thousands of design evaluations to guide the search of the best blade profiles to be used in the more expensive, high-fidelity CFD runs, improving the progress of the optimization while reducing the required computation time. It is estimated that the current workflow achieves a five-fold reduction in computational time in comparison to an optimization process that is based on three-dimensional (3D) CFD simulations alone. The methodology is demonstrated on the NASA/General Electric Energy Efficient Engine (E3) high pressure turbine blade and found Pareto front designs with improved blade efficiency and power over the baseline. Quantitative analysis of the optimization data reveals that some design parameters in the present study are more influential than others, such as the lean angle and tip scaling factor. Examining the optimized designs also provides insight into the physics, showing that the optimized designs have a lower amount of pressure drop near the trailing edge, but have an earlier onset of pressure drop on the suction side surface when compared to the baseline design, contributing to the observed improvements in efficiency and power.



Citation: Zhang, C.; Janeway, M. Optimization of Turbine Blade Aerodynamic Designs Using CFD and Neural Network Models. *Int. J. Turbomach. Propuls. Power* **2022**, *7*, 20. <https://doi.org/10.3390/ijtp7030020>

Academic Editor: Tom Verstraete

Received: 25 April 2022

Accepted: 28 June 2022

Published: 30 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Keywords: CFD; optimization; aerodynamics; gas turbines; machine learning; neural networks



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-NC-ND) license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

The present study applies machine learning methods in a CFD-based optimization for turbine blade aerodynamics. Literature on optimization and machine learning methods used in gas turbine studies will be reviewed, followed by a summary of the motivation of the present work.

Numerical optimization has been widely used in the design and analysis of gas turbines. In some earlier studies, specific optimization algorithms have been investigated to leverage lower fidelity models to achieve fast optimization time. One study on aerodynamic wing optimization has used an approximation and model management optimization

method to incorporate low fidelity, computationally cheaper models with occasional recourse to higher fidelity, more expensive models, resulting in threefold saving in optimization time [1]. Another study adopted a similar concept, employing a trust-region approach to interleave the exact models with cheaper surrogate models during optimization iterations [2]. These methods demonstrate the possibility of obtaining optimized solutions on a limited computational budget by incorporating lower-fidelity surrogate models. In more recent years, an increasing number of optimization studies have relied on using parametric CFD models. In optimization of the coolant flow passage of the NASA C3X vane, different designs were evaluated repeatedly through CFD runs [3]. In the study of a marine high-pressure turbine [4], ten design parameters controlling multi-row film cooling designs were built into the CFD model and optimized based on a non-dominated sorting genetic algorithm. Multiple studies were also conducted on ultra-super-critical steam turbines [5,6], in which the blade aerodynamic efficiency was optimized using 2D and 3D CFD simulations driven by Siemens' Simcenter HEEDS commercial Sherpa optimization algorithm. A CFD-based co-optimization strategy was presented in [7], which demonstrates a workflow for coupling different disciplines into a nested optimization loop to conduct parallel blade aerodynamic and thermal optimizations. In addition to improving turbine blade designs, optimization has also been applied to improving the operations of gas turbine engines, for instance, to find the best valve setup parameters that reduce fuel consumption [8].

With the advancements in computer science and data storage, an increase in interest in application of machine learning methods to gas turbine designs has been observed. One area of such applications seeing increased interest has been the prediction of key performance metrics using models trained on input data gained through past simulations or experiments. In an earlier study [9], the outlet temperature and fuel mass flow rate at different operating conditions for a 255 MW single-shaft gas turbine were predicted by building a three-layer neural network model. Another application of the Artificial Neural Network (ANN) model has been seen in a turbine film cooling study to predict the instantaneous temperature distributions along the blade surface as well as the cooling effectiveness [10]. In another study of a jet engine power plant [11], a machine learning method combining physics-based and measurement-driven modeling was developed and used to conduct preventive maintenance and diagnose faults. Machine learning methods were also applied to a Viper 632-43 military turbojet engine to predict the exhaust temperature using models trained on data collected from a gas turbine simulation program [12]. Extending from predicting individual engineering metrics, machine learning has also been used to predict field quantities representing more complicated underlying physics. A method using gradient boosted trees was used to develop models of aerodynamic loads on vibrating turbine blades and demonstrated to have good agreement with detailed CFD results [13]. In another study, the turbine surface pressure distribution was predicted using transfer learning models, which transfer knowledge from a large-scale but low-fidelity dataset to a small-scale but high-fidelity dataset, shown to have a low prediction error with reduced cost [14]. Machine learning has also been applied to the prediction of operating characteristics of gas turbine engines, using real-time data of power plants to develop neural networks [15,16]. In addition to the above applications, machine learning has also been studied to develop turbulence closure models. In one study for wake mixing, a machine learning model was demonstrated to be robust across several different operating conditions when integrated into a RANS CFD model of a low-pressure turbine [17]. A review article on machine learning methods for science and engineering particularly highlighted the need for interpretable, generalizable, expandable, and certifiable machine learning techniques for safety-critical applications [18].

Recent works have also focused on using machine learning embedded into design optimization procedures. In the optimization of a centrifugal compressor impeller [19], an ANN model was first developed using CFD and FEA data from a Design of Experiment (DOE) study. Then, the ANN model was applied in an optimization procedure, which resulted in a

1% increase in isentropic efficiency and 10% reduction in the blade stress. In another study investigating a carved blade tip [20], 55 CFD runs were conducted to generate ANN meta models, which were then used in a genetic algorithm routine to optimize the blade tip shape. In a missile control surface optimization study [21], machine learning, reinforcement learning, and transfer learning were integrated into the optimization procedure and leveraged CFD in the evaluation iterations. In another study of 2D airfoil optimization [22], a deep convolutional generative adversarial network was trained and embedded as a surrogate model in an optimization framework. In still another study of a compact turbine rotor [23], machine learning models were trained and used to optimize the efficiency and torque based on a gradient-based multi-objective optimization algorithm. In addition to using machine learning models in optimization, several CFD and optimization studies have compared machine learning models with response surface models (RSM). In a study of aircyclone optimization [24], it is concluded that ANN offers an alternative and powerful approach to response surface methods for modeling the cyclone pressure drop, benchmarked against experimental data. In another study of modeling and optimizing a perforated baffle used for turbine blade passage cooling [25], both ANN and RSM methods were found capable of predicting friction factor and Nusselt number values, although the RSM method performed slightly better than ANN in that study. A more recent study on cyclone optimization also tested a RSM and several machine learning models. A GMDH-neural network model was found to be superior and chosen for the optimization process [26].

As discussed in the above literature, optimization that leverages high-fidelity CFD simulations can provide accurate and realistic optimal designs in general. While leveraging a neural network as a surrogate to replace the CFD evaluations in the optimization loop can significantly improve the computational time, the fundamental challenge is that a neural network model is a lower fidelity model compared to a CFD simulation, and therefore, it may not be as accurate as CFD in predicting certain design variants required by the optimization process. Further, neural network models may also be trained based on previously available datasets that come from different studies, such as 1D and 2D simulation data, and experimental data, all of which will result in the neural network being a reduced order model compared to 3D CFD. As such, relying solely on the neural network when the predictive accuracy is necessary can lead to errant results. The present study presents a nested optimization workflow that can leverage both neural networks and high-fidelity, 3D CFD simulations at the same time to ensure every “best” design is studied in detail by the CFD tool. The introduction of a neural network into the optimization allows for over 70% reduction in the number of CFD evaluations and thus makes significant reductions in computational cost compared to a process relying exclusively on CFD simulations. The methodology is demonstrated through an aerodynamic optimization using a rebuilt model of the NASA/General Electric E3 high pressure turbine blade [27–29].

2. Methodology

Artificial Neural Network models are used alongside 3D CFD simulations in a numerical optimization procedure to improve the aerodynamic performance of a turbine blade. ANN models are typically trained using large datasets obtained from previous experimental or numerical studies. In companies/organizations that conduct R&D on engineering designs, these datasets are usually available from previous studies. The present study proposes a methodology of using such existing knowledge from previous studies to train ANN models and then use the ANN models in an CFD-based design optimization process. A dataset from a previously published work is obtained to conduct the ANN modeling training in the present study. The overall framework of the research methodology is shown in Figure 1, which illustrates how different analysis models/tools required by the optimization process are created. First, data were obtained from a previous study [7] containing 3204 CFD design evaluations of different 2D turbine blade profiles. Performance metrics including efficiency and power were extracted as targets, with the blade geometric parameters serving as input features, to train ANN models. The ANN hyperparameters

were subsequently optimized. The blade cross-sectional profiles were parameterized using a Class-Shape Transformation (CST) method [30]. Using the 2D blade cross-sectional profile, the 3D blade CAD geometry was created with additional parameterization allowing for the variation of further design transformation, such as the thickness, twist, and lean. The blade CAD geometry was then used to build a 3D Reynolds Averaging Navier–Stokes (RANS) CFD model, which solves for the flow field in a single blade passage. Once the ANN models and the 3D CFD simulation are independently developed, they are integrated into a nested optimization procedure for repeated evaluations of different blade design variants to optimize the design parameters. In the nested optimization process, an inner optimization loop is executed using the ANN models to improve the blade cross-sectional profile. In each iteration of the global optimization, the best blade profile is selected from the inner optimization step to create a 3D CFD blade model with additional parameters such as the scaling factors, twist, and lean angles. The optimization targets improving the efficiency and power of the blade. Details of the optimization process will be discussed in a later session in this chapter.

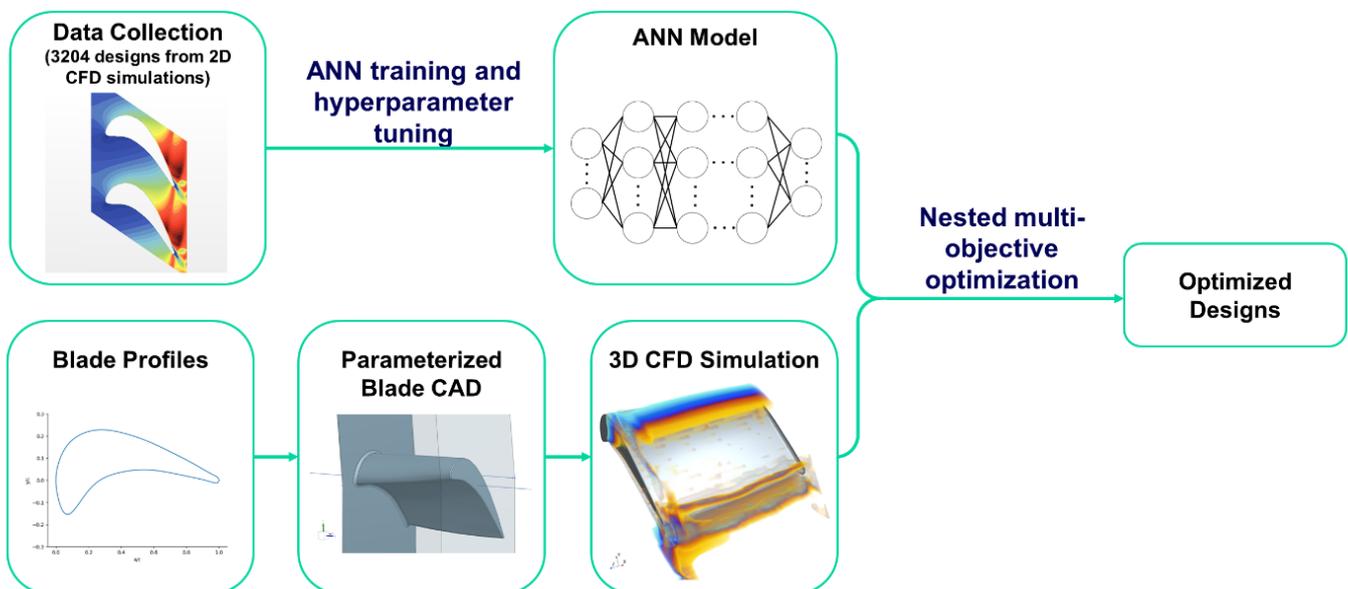


Figure 1. The overall framework of the research methodology. (2D CFD data was collected from a previous study [7].)

2.1. Artificial Neural Network

A dataset representing different design variants of 2D aerodynamic CFD results was obtained from a previous study [7] and used to train ANN models. The dataset features blade design and performance parameters for 3204 designs. The input parameters for the ANN models include 20 blade design parameters. These parameters are the weighting factors used by the CST method [30] to control variations in the blade profile. Under the CST method, these weighting factors are multiplied by Bernstein polynomials to define a shape function, and then subsequently multiplied by a class function to define the aerodynamic blade profile. An order-9 Bernstein polynomial was deemed sufficiently flexible for the purposes of the study [7] and that choice is repeated here, resulting in 10 design parameters for each of the suction and pressure sides of the blade: $\{w_{u,1}, w_{u,2}, \dots, w_{u,10}\}$ and $\{w_{l,1}, w_{l,2}, \dots, w_{l,10}\}$. These 20 parameters are used as input features in the ANN models and optimization parameters in the subsequent blade optimization process. Two performance metrics in the dataset were used as output parameters, isentropic efficiency,

and power output, for which two ANN models were developed, respectively. The efficiency and power are defined based on enthalpy quantities of the flow, as follows:

$$\eta = \frac{h_i - h_o}{h_i - h_{o,isen.}}, \quad (1)$$

$$Pow = \Delta h = h_i - h_o \quad (2)$$

The Keras API for TensorFlow [31] was used to construct the ANN models. In consideration of the different distributions of the efficiency and power data and the robustness of the model, a separate ANN model was constructed for each of these two quantities. After some preliminary tests on ANN models of 5, 6, and 7 layers, 7 and 6 layers were constructed, respectively, into the ANN models representing efficiency and power. The choice of the number of layers in ANN usually varies in different problems. It will be shown that the prediction errors are within acceptable range for the chosen number of layers, and that the errors will be further reduced by optimizing the hyperparameters of the models. The first layer of each ANN model has 20 inputs representing the blade input features (CST weights) and the last layer has one output, which is either efficiency or power. The number of neurons in the hidden layers of each ANN, along with other topological and training parameters, were optimized in a separate hyperparameter optimization process. As a starting point for that process, the hyperparameters for each ANN model were set using the values provided in the first two columns of Table 1.

Table 1. Hyperparameters of the initial and optimized ANN models.

Hyperparameters	Initial ANN for Efficiency	Initial ANN for Power	Optimized ANN for Efficiency	Optimized ANN for Power
Number of Neurons-Layer 2	20	20	5	23
Number of Neurons-Layer 3	40	40	35	15
Number of Neurons-Layer 4	20	20	53	36
Number of Neurons-Layer 5	10	5	50	42
Number of Neurons-Layer 6	5	–	62	–
Activation Function-Layer 2	relu	relu	relu	selu
Activation Function-Layer 3	relu	relu	sigmoid	softmax
Activation Function-Layer 4	relu	relu	elu	elu
Activation Function-Layer 5	relu	relu	elu	softmax
Activation Function-Layer 6	relu	–	softmax	–
Learning Rate	0.001	0.001	6.4895×10^{-5}	1.47285×10^{-3}
Number of Epochs	200	200	6123	4000

To improve the predictive performance of the ANN, the hyperparameters were optimized using Siemens commercial optimization tool HEEDS and its Sherpa algorithm [32]. A diagram of the optimization process is provided in Figure 2. The optimizer repeatedly tunes all the hyperparameters for the ANN models listed in the index column of Table 1 with an objective to minimize the model test error on a held-out testing dataset. In each optimization cycle, the dataset is randomly split into a training and testing set with a ratio of 95/5. The training set is used to train the ANN model. The training process involves a typical backpropagation procedure with an 80/20 split of the training set, over multiple epochs with TensorFlow's ADAM optimizer. After the training step is completed, the model's performance is then evaluated against the held-out test set. The evaluation error is passed back to the optimization solver to tune the hyperparameters for the next cycle. A total number of 240 iterations were executed during the optimization of each of the two ANNs. The evaluation number, 240, was based on the Sherpa algorithm best practice [32], which recommends the number of evaluations being about 10 times the number of input variables for single objective optimization problems, due to the hybrid and adaptive methods employed by the algorithm. In the ANN optimization, 12 hyperparameters must be

optimized, and thus it requires at least 120 evaluations according to the best practice. In consideration of the relatively fast run time for the ANN training process in comparison to the 3D CFD run time, that evaluation number is further doubled to 240 to provide better accuracy.

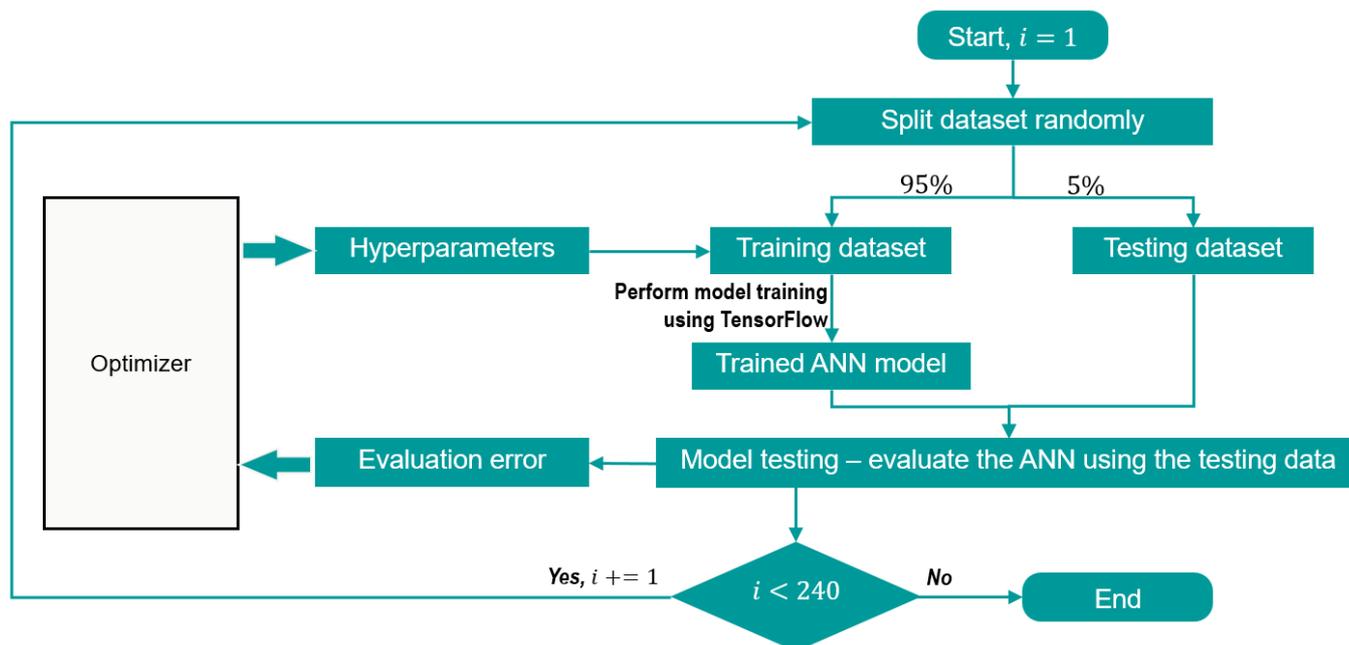


Figure 2. ANN model optimization diagram.

The optimized hyperparameters are obtained and provided in the last two columns of Table 1. After optimization, the evaluation errors for efficiency and power were reduced, respectively, from 0.33% to 0.10%, and from 0.43% to 0.10%. The performance of the initial and optimized ANN models are also examined by comparing the predicted values vs. true values from the test set, shown in Figure 3. It is observed that the optimized ANN models result in better alignment between the predicted values and the true values, as well as narrower bandwidths of the prediction errors on the histogram plots. From Figure 3, it is also observed that the bias of the optimized ANN for efficiency was greatly reduced, as the data points are evenly distributed on both sides of the line of unity slope, shown in the second subplot. This is consistent with the histograms of the prediction errors in Figure 4, which shows that both the bias and bandwidth of the errors were improved in the optimized ANN models. The optimized ANN models are used in the subsequent blade optimization process in this study. The Python scripts for training the ANN models with the optimized hyperparameters are provided in [33]. The CPU time for training an individual ANN largely depends on the number of Epochs. In the present study, the CPU times for training the ANN models representing efficiency and power with their respective optimized hyperparameters were 14.3 min and 8.5 min, respectively. The total compute times for optimizing the hyperparameters of these ANN models were, respectively, 13.07 h and 11.65 h. This optimization was carried out based on parallel execution of 10 training models at the same time using a 6-core CPU.

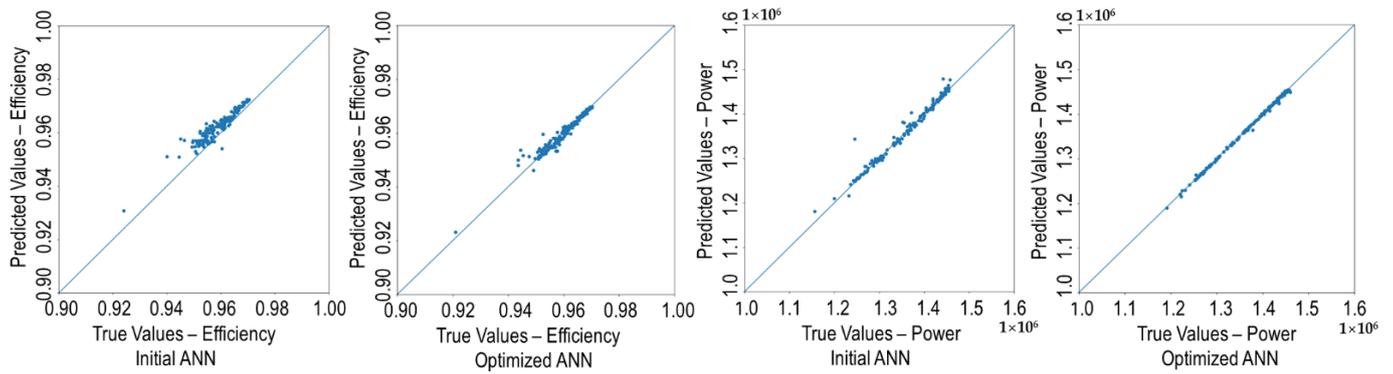


Figure 3. Predicted vs. true values of the initial and optimized ANN models.

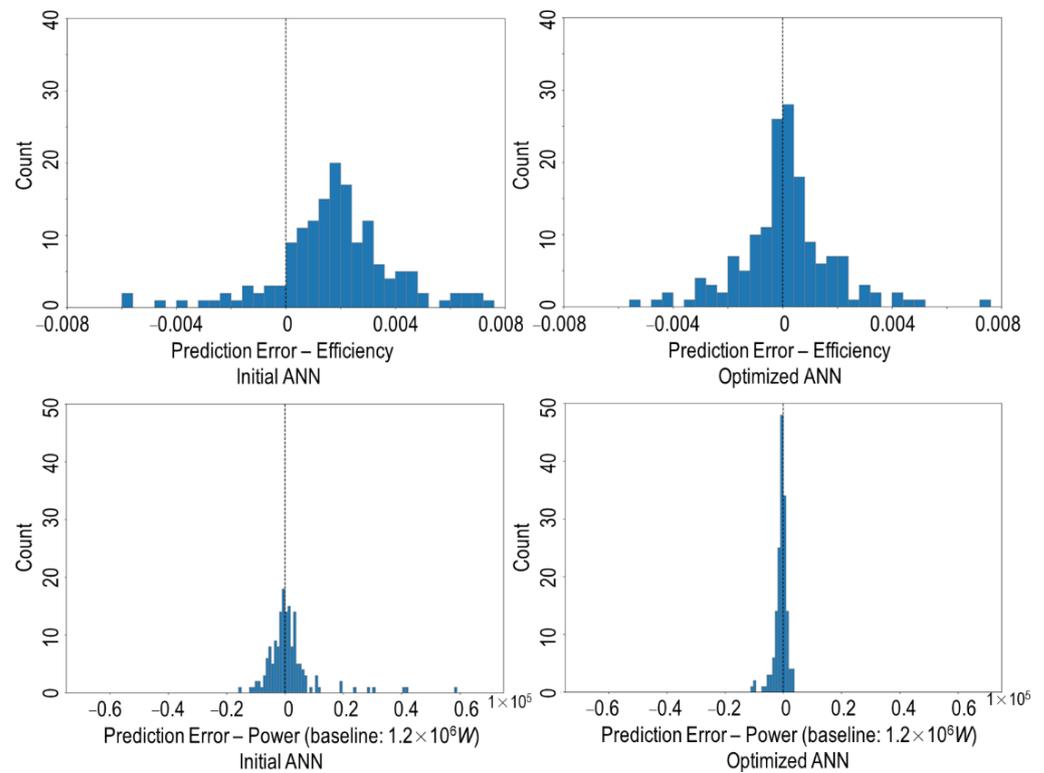


Figure 4. Histogram plots of the prediction errors of the initial and optimized ANN models.

2.2. Parametric Blade CAD and CFD Model

An initial turbine blade CAD model was built by adapting the E3 high pressure turbine blade profiles. Using the first stage rotor blade coordinates provided at three spanwise locations [27]—hub (12.693in), pitch (13.571in), and tip (14.41in)—and accounting for the incoming flow angle [28,29], a 3D CAD model was constructed by lofting these cross-sectional profiles, as shown in Figure 5a, to provide a baseline case. Applying the same philosophy of lofting cross sections and at the same time utilizing the CST method for the cross-sectional profile definition [27], a parameterized blade CAD model was created for the design optimization study. First, the CST method creates a base cross-sectional profile, which is used as the pitch profile. The CST method allows varying the base profile in design optimization through manipulating the 20 weighting parameters, $\{w_{u,1}, w_{u,2}, \dots, w_{u,10}\}$ and $\{w_{l,1}, w_{l,2}, \dots, w_{l,10}\}$. Then, the base profile is scaled to create the hub and tip profiles. In the scaling process, the twist angles and chord lengths of the hub and tip profiles are matched to their counterparts of the original E3 blade; 4 additional scaling factors, $\{\xi_{hub,l}, \xi_{hub,u}, \xi_{tip,l}, \xi_{tip,u}\}$, were applied to the lower and upper profiles

of the hub and tip sections, respectively, to allow for small variations of the thicknesses of the hub and tip sections so that they can be optimized. The ranges of these variations are defined conservatively out of practical considerations of the original E3 engine blade shape. In general, the scaling allows the hub profile to be thicker and the tip profile to be thinner. Finally, a tilt angle, θ , and a lean angle, β , were also built into the CAD model to be optimized later. Based on a reverse engineering study analyzing different engine blade geometries [34], the tilt angle can vary from 0 to 0.164° , and the lean angle can vary from -0.086° to 0.086° in the optimization study. A schematic plot of these angles is shown in Figure 5b.

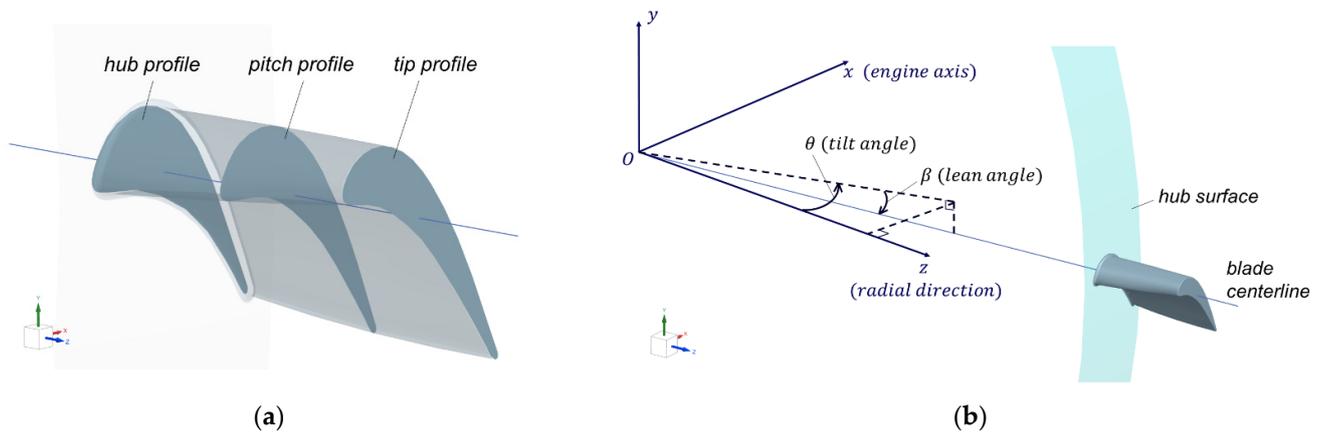


Figure 5. Blade geometry schematics: (a) blade geometry adapted from E3 profile coordinates; (b) blade angles schematic.

Using the 3D blade, a single-blade passage CFD model was developed using Siemens multi-physics package Simcenter STAR-CCM+. The continuity, momentum, and fluid energy transport equations are solved using a coupled solver (density-based) following a finite volume approach with a 2nd order upwind discretization scheme on a polyhedral grid. Menter's SST K-Omega model [35] with all y^+ treatment is used as a closure to the turbulence model. The all y^+ wall treatment adjusts the application of a turbulence wall function based on the local y^+ value of the near wall mesh cell. The CFD simulations were performed using the test conditions. Following E3 engine rotor testing conditions and 2D CFD practices from NASA studies [28,29], a total pressure of 344,777 Pa and a total temperature of 709 K were used for the inlet boundary condition, while an atmospheric pressure of 101,325 Pa was defined at the outlet. The same boundary conditions representing the engine testing conditions were also applied in the previous 2D CFD simulations in [7]. (These 2D CFD datasets were used to develop ANN models in the present work.) A grid sequencing method was used to provide initial conditions, by solving inviscid flow equations repeatedly on a set of gradually refined mesh grids. Automatic CFL number control was also applied to adjust the CFL number in response to linear solver convergence behavior during the algebraic multigrid procedure. To ensure overall solution convergence is reached, in addition to monitoring the residuals of the governing fluid equations, stopping criteria were also set based on the asymptotic values of engineering quantities, such as isentropic efficiency, power output, maximum surface temperature, and total pressure difference.

The CFD methodology has been applied in a previous study based on a 2D version of the E3 blade and validated against other experimental and numerical results of the E3 blade [7]. In the present study, a grid independence investigation has also been performed for the 3D CFD. Three sets of CFD grids with different resolutions were tested. The chosen grid features 12 million polyhedral cells with near-wall y^+ values all below 1.4, based on which the solution calculates a baseline efficiency of 96.0% and a power of 5.638×10^4 W. (Note power is calculated based on the single blade CFD model in the current study, which has a pie sector angle of 4.737°). The relative difference between the chosen grid and its next-level refined grid is 0.1% for efficiency and 0.02% for power.

2.3. Optimization Strategy

The present study adopts a nested optimization process. The nested optimization is a slight variation of a previous co-optimization strategy introduced in [7]. The nested optimization accomplishes two essential tasks. First, because an ANN model calculation consumes significantly less computing resources and time than a 3D CFD run, nesting an inner optimization based on the ANN model can effectively utilize the time while waiting for a 3D CFD run to complete. For each 3D CFD simulation run, 250 evaluations of different blade cross-sectional profiles were completed in the inner optimization loop using the ANN model. Second, the inner, ANN-driven optimization passes the best blade profile of the present optimization cycle to the 3D CFD run. This architecture effectively allows using the ANN-driven optimization to guide the 3D CFD search, and thus reduces the number of 3D CFD runs compared to an optimization process that is solely based on evaluating 3D CFD models.

The goal of the optimization is to maximize blade efficiency, η , and power output, Pow . The design variables that are being explored in optimization are the ones discussed in the previous session, $w_{u,1}, w_{u,2}, \dots, w_{u,10}, w_{l,1}, w_{l,2}, \dots, w_{l,10}, \xi_{hub,l}, \xi_{hub,u}, \xi_{tip,l}, \xi_{tip,u}, \theta$ and β . The optimization procedure repeatedly executes the ANN and CFD models to evaluate different blade design variants, while extracting performance metrics to search for better designs. As the ANN model is much less computationally expensive compared to CFD, it is leveraged in an optimization loop by itself to yield optimized cross-sectional blade profiles, which are subsequently used to construct 3D shapes for CFD simulations. Using the ANN embedded optimization to guide the search for the best blade profiles allows for reducing the number of expensive CFD evaluations. As such, a nested optimization workflow is constructed, as illustrated in Figure 6. The workflow is realized using commercial optimization software Simcenter HEEDS and its Sherpa optimization algorithm [30] is used in both Optimizer 1 and Optimizer 2. The Sherpa algorithm constantly evaluates the characteristics of the problem by using a hybrid combination of search strategies at each stage of the optimization process to best traverse the design landscape. In addition, it also adapts the tuning parameters of the search strategies to the specific region of the search. The HEEDS' Sherpa algorithm has been effectively applied to optimize gas turbine applications, as shown in studies [5–7].

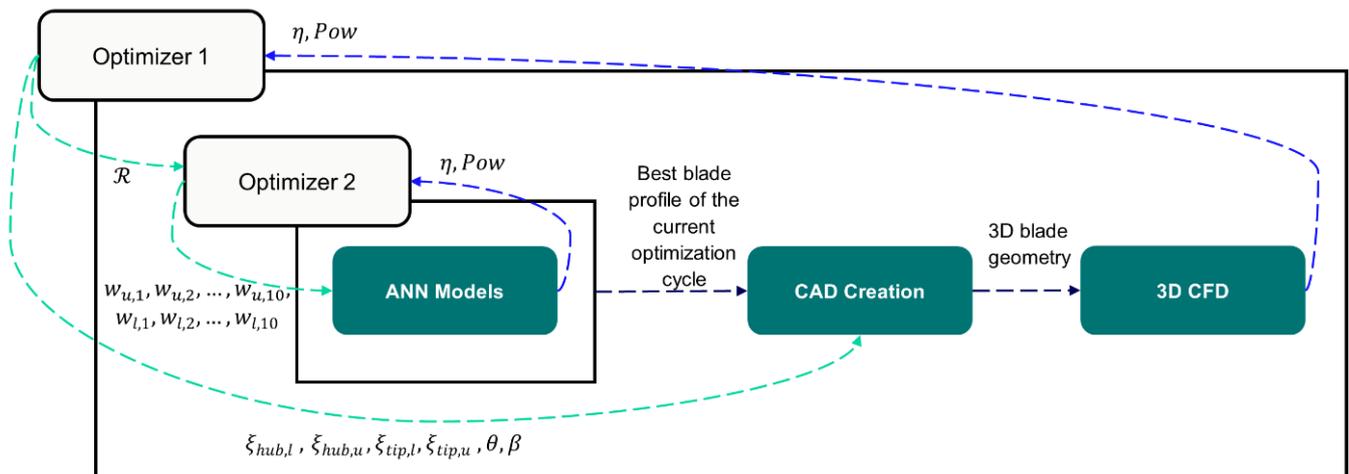


Figure 6. Optimization iteration hierarchy and data flow.

In the global optimization loop (Optimizer 1 as shown in Figure 6), a multi-objective trade-off problem is solved targeting maximizing both efficiency and power. Two constraints are set: efficiency must be greater than 95% and power must also be greater than a baseline power value. These constraints are included in the optimization performance function, based on considerations of practical gas turbine design metrics and general

performance characters of the E³ engine. The global optimization solver, Optimizer 1, affects the blade shape change by directly optimizing the section scaling factors and blade angles in the CAD. In addition, Optimizer 1 also affects an inner optimization process, which is represented by Optimizer 2. Optimization 1 controls the selection of the CST weights used as input parameters for Optimizer 2, through manipulating a random seed, \mathcal{R} , which sets the initial state for the global search strategies employed by the Sherpa algorithm. Although Optimizer 1 does not directly manipulate the CST weights, because \mathcal{R} controls their selection, it acts as a 1D encoding of the entire 20-dimensional design space for Optimizer 1 to explore. Optimizer 1 repeatedly evaluates and optimizes different design variables during the global optimization cycles; in each evaluation cycle, Optimizer 2 optimizes the CST weights using the ANN models, with an objective to maximize the efficiency and a constraint on minimum power value. Overall, 250 blade cross-sectional profiles were evaluated in each execution of Optimizer 2. It directly explores the shape profiles through manipulating the weighting parameters used in the CST method. In each evaluation cycle of Optimizer 1, the best blade profile found by Optimizer 2 is applied in the more expensive and high-fidelity CFD simulation. The performance function in each optimization is defined as the sum of normalized objectives minus the sum of normalized constraints. In a generic form, the performance function is formulated as follows:

$$P = \sum_{i=1}^{i=N_{obj}} \frac{Obj_i}{Norm_i} - \sum_{j=1}^{j=N_{cons}} \frac{10^4 \times ConsViol^2}{Norm_j^2}, \quad (3)$$

In the above formula, Obj_i represents the objective quantity, such as η and Pow ; $ConsViol$ represents the amount of violation resulted from each constraint definition ($\eta > 0.95$, $Pow > Pow_{baseline}$); the normalization factors Norm are selected using the respective η and Pow values of the baseline design.

3. Results

3.1. Optimization Results

In the present optimization study, a total of 163 blade designs have been evaluated in 3D CFD, with 40,750 2D blade profiles being evaluated by the ANNs. An efficiency vs. power plot showing all 3D blade design points is provided in Figure 7. A Pareto front is formed in the upper right region of Figure 7, which contains optimal designs with the best trade-off relations between efficiency and power. Among the optimal designs, the maximum increase in efficiency is over 3%, and the maximum increase in power is around 8%. To obtain insight into the design variable distributions of the optimal designs, two groups of optimal designs are selected—a group of top 10 high-efficiency designs, and another group of top 10 high power designs. The responses and selected design input variables (which include the two blade angles and four scaling factors) for these two groups of designs are highlighted on a parallel plot in Figure 8. The parallel plot reveals some qualitative trends in the design parameters: (1) both the high-efficiency and high power designs favor large lean angles, β ; (2) the high-efficiency designs also favor a large scaling factor for the tip upper surface, $\zeta_{tip,u}$, while the high power designs favor a small-to-medium value for $\zeta_{tip,u}$; (3) a weaker influence of the scaling factor for the hub lower surface, $\zeta_{hub,u}$, is also observed. Note that the lean angle was constrained conservatively within a smaller range in consideration of commercial turbine geometric characteristics provided in [27].

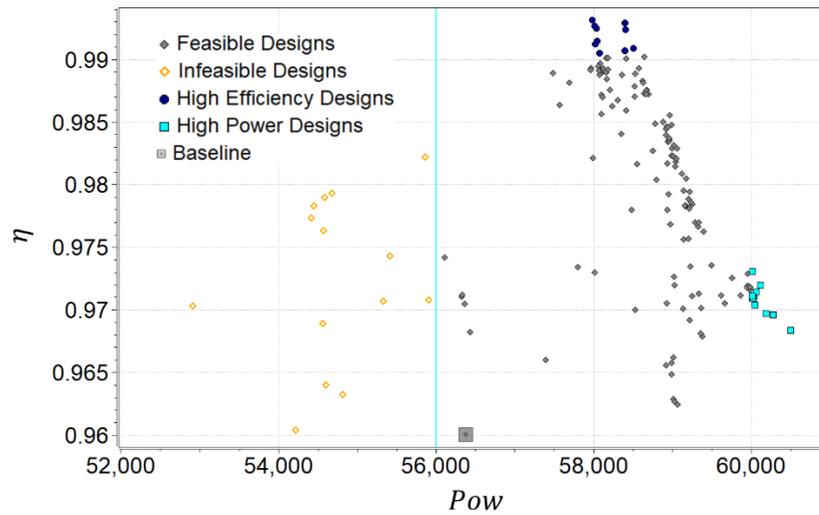


Figure 7. Efficiency vs. power for all designs points in the optimization study.

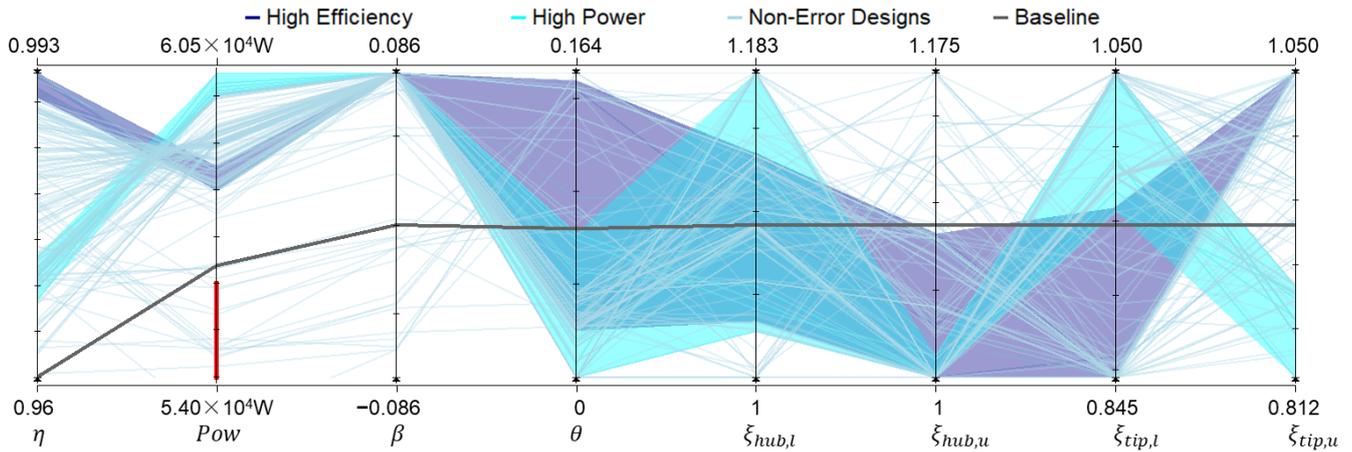


Figure 8. Parallel plot highlighting selected design parameters of different groups of designs.

A quantitative analysis has also been conducted. The Pearson correlation coefficients of the data are calculated, based on the covariance and standard deviations. The definition is given as follows:

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(\mathcal{Y}_i - \bar{\mathcal{Y}})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (\mathcal{Y}_i - \bar{\mathcal{Y}})^2}} \quad (4)$$

In the above formula, $x, \mathcal{Y} \in \{\eta, Pow, \beta, \theta, \xi_{hub,l}, \xi_{hub,u}, \xi_{tip,l}, \xi_{tip,u}\}$, ($x \neq \mathcal{Y}$). x_i and \mathcal{Y}_i represent sample values corresponding to each blade design, and n is the total number of designs, which is 163 in the present study. The Pearson correlation coefficients, the histogram distributions of selected parameters, and the data sample plots are shown on a correlation plot in Figure 9, respectively, on the upper right region, along the diagonal line joining the upper left and lower right corners, and on the lower left region. Large magnitudes of the Pearson correlation coefficients for $\{\xi_{tip,u}, \eta\}$ (1st row, last column) and $\{\beta, Pow\}$ (2nd row, 3rd column) indicate strong relationships in these two pairs of variables. The large coefficient values are confirmed by their respective data sample plots of $\xi_{tip,u}$ vs. η (last row, 1st column) and β vs. Pow (3rd row, 2nd column), in which most of the design points follow the linear regression lines closely. In addition, the correlation plot also shows that the effect of the scaling factor for the hub lower surface, $\xi_{hub,u}$, is rather mild.

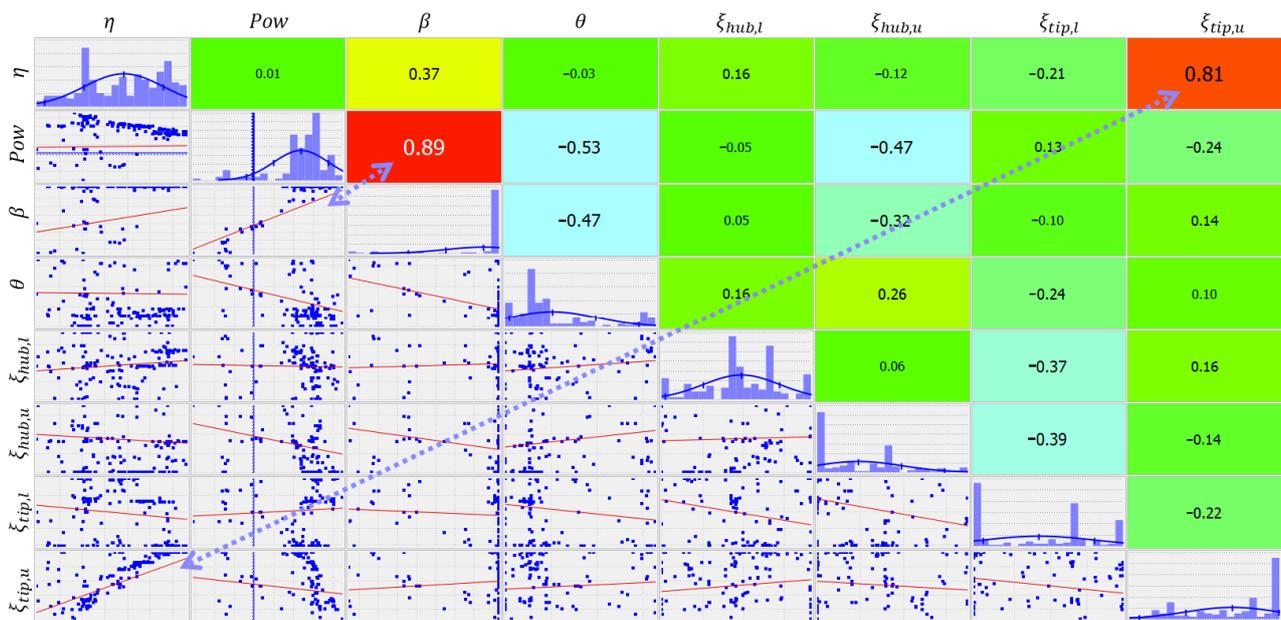


Figure 9. Correlation plot of selected design parameters of all designs. (The two most influential numbers and their corresponding plots are highlighted using the dashed arrows.)

The efficiency and power are plotted versus the two parameters with the strongest correlations, lean angle (β), and tip upper surface scaling factor ($\xi_{tip,u}$), shown in Figure 10. It is observed that (1) most of the improved designs favor larger lean angles, and (2) among the improved designs, a smaller $\xi_{tip,u}$ leads to greater power improvement, while a larger $\xi_{tip,u}$ leads to high-efficiency gain. To analyze the effects of the blade shape and the fluid dynamics, two designs are selected for further investigation—a high-efficiency design and a high power design, which will be referred to as “Design A” and “Design B,” respectively. These two designs are also marked in Figure 10 and will be discussed in the next session.

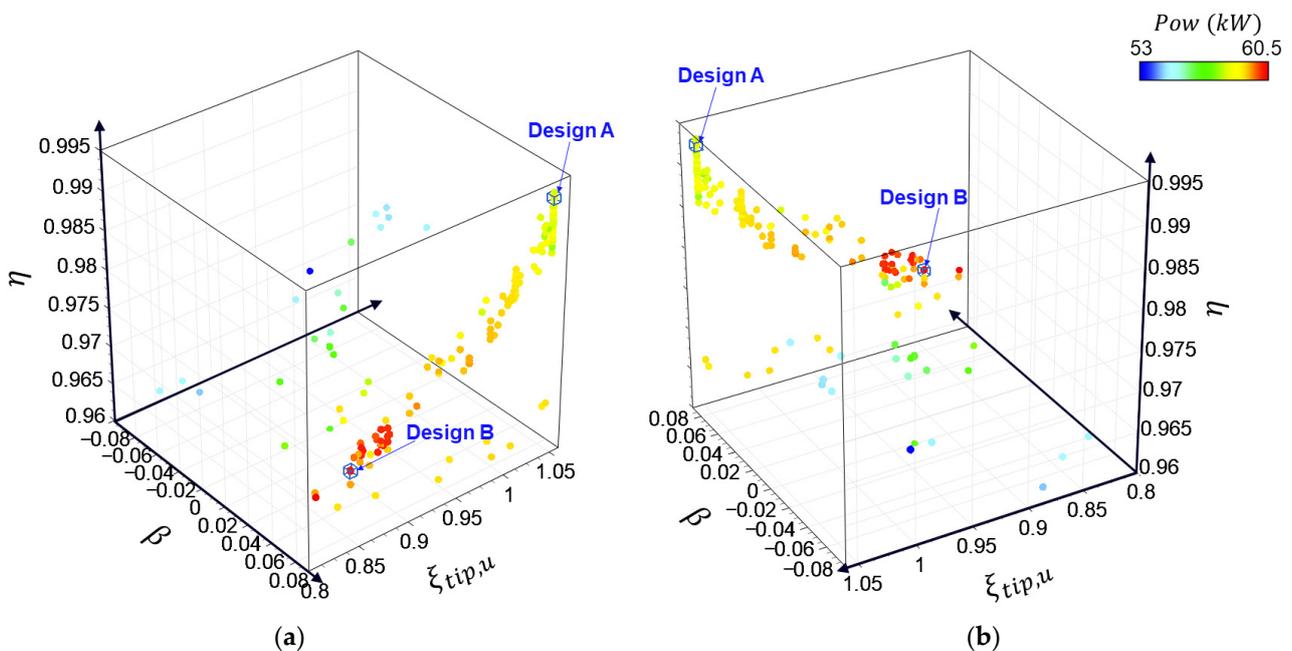


Figure 10. Two views of a 3D plot showing efficiency, power, lean angle, and tip upper surface scaling factor of all designs: (a) a normal view; and (b) a view obtained by rotating the left-side plot around the η axis by 180° .

The total run time for the optimization process was around 30 h (with six CFD evaluations running in parallel, each consuming 160 compute cores). There are 27 input variables in the global optimization problem. As illustrated in Figure 6, the integration of the inner optimization loop allows for optimizing the 20 blade profile parameters using the inexpensive ANN models. Throughout the optimization process, the ANN models were evaluated 40,750 times; in each optimization cycle, the best blade cross-sectional profile obtained from the inner ANN optimization was passed on to the 3D CFD for evaluation. As a result, the number of design variables being directly exposed in the 3D CFD runs is reduced to six. Since the required number of CFD evaluations largely depends on the number of input variables in an optimization problem, the nested optimization strategy only requires 163 evaluations of 3D CFD simulations. In comparison, if an optimization study of the same problem does not adopt the nested optimization strategy and is solely based on CFD, then all 27 design variables (rather than 6) will be evaluated based on 3D CFD runs. Based on a quick estimation, this will result in 4.5 times ($=27/6$) the number of CFD runs, which is 733 CFD runs, and will cost roughly 135 h of optimization run time if the same compute resource settings were used.

3.2. Fluid Dynamics Analysis

Two optimized designs are selected for detailed investigation of the aerodynamic results: a high-efficiency design, and a high-power design (a.k.a. Designs A and B as shown in Figure 9). They are compared with the baseline design. The CFD results show the optimized designs feature less pressure drop on the suction side near the trailing edge. A pressure coefficient is defined based on the average inlet velocity and density:

$$c_p = \frac{p}{\frac{1}{2}\rho_{in}u_{in}^2}, \quad (5)$$

The pressure coefficient distributions on the hub, mid, and tip sections are compared in the upper row of plots in Figure 11. The baseline design shows significant low pressure regions on the suction surface near the trailing edge on all three streamwise sections. In comparison, the two optimized designs saw increased minimum pressure in those areas at the hub and mid sections. An enlarged view of the pressure coefficient distribution on the mid-section near the trailing edge is provided for each design in the lower row of plots in Figure 11. The baseline design features a shock in this region, highlighted by a relatively high pressure spot next to the low pressure zone, which is present inside the dotted circle marked on Figure 11a. This shock feature is absent in the two optimized designs due to the modifications in the blade shapes in those designs. This observation is consistent with the Mach number distributions that will be discussed in a later paragraph.

To further investigate the pressure drop on the suction side near the trailing edge, volumetric renderings of the pressure coefficient scenes are shown in Figure 12, to highlight the flow in the low-pressure regime by focusing on c_p values between -0.35 and -0.1 . It is observed that in the optimized designs, lower pressure coefficients are present near the tip, but the pressure coefficient has been increased in the mid-section and nearer the hub. The overall reduction of maximum negative pressure coefficients in the mid and hub regions offsets the effects of increased tip leakage flow in the optimized designs.

Next, the pressure coefficient distributions on the blade surfaces are investigated. Scalar scenes of the pressure coefficients on the baseline, Design A, and Design B blades are shown in Figure 13. A large portion of the suction-side surface in the baseline design features a significantly negative c_p values between -0.35 and -0.23 starting around the three quarters chord. In the optimized designs, the onset of the negative pressure coefficient zone moves upstream, nearer the half chord, yet the magnitude of this zone is much less, with most areas featuring c_p values between -0.12 and 0 . To achieve a more quantitative comparison, the pressure coefficient curves along the streamwise direction on the hub, mid, and tip sections of the three blades are plotted and compared, with the plots of the blade profiles, shown in Figure 14. The earlier onsets of the lower pressure coefficient region on

the suction-side surfaces in the two optimized designs are demonstrated by the plots. The earlier onset allows for larger enclosed areas of negative c_p curves for the optimized design, contributing to the power improvements.

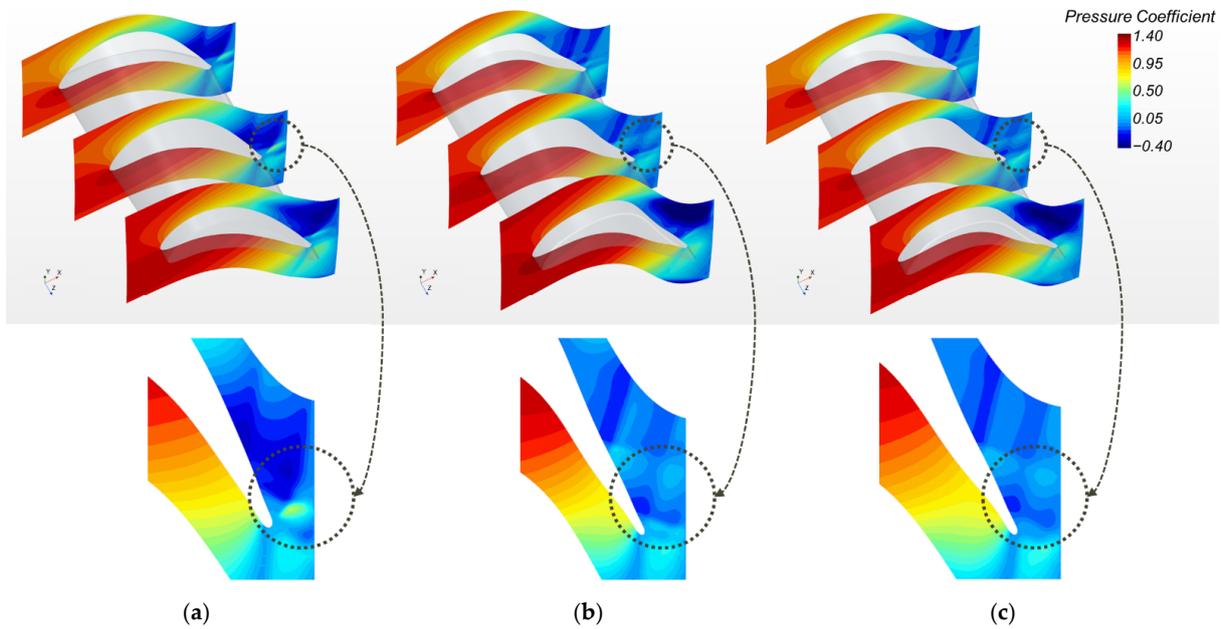


Figure 11. Static pressure distributions: (a) Baseline design; (b) Design A; and (c) Design B.

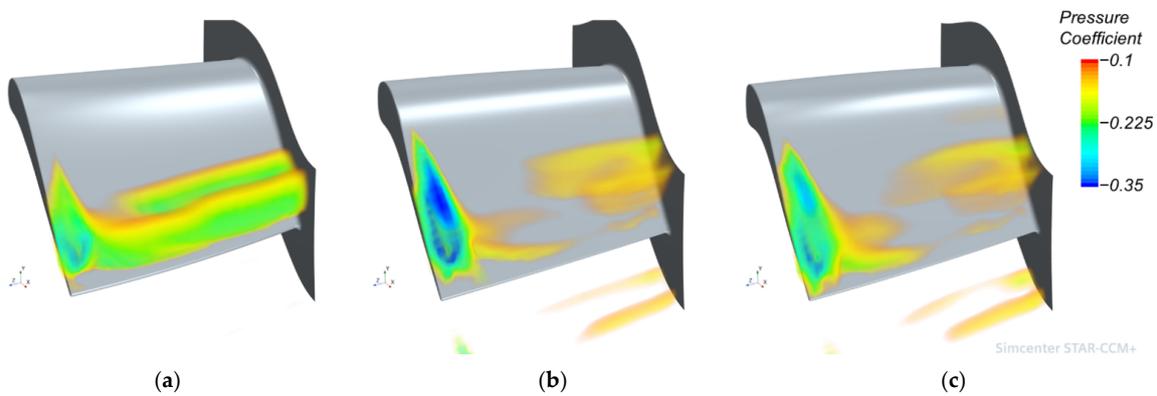


Figure 12. Static pressure distributions: (a) Baseline design; (b) Design A; and (c) Design B.

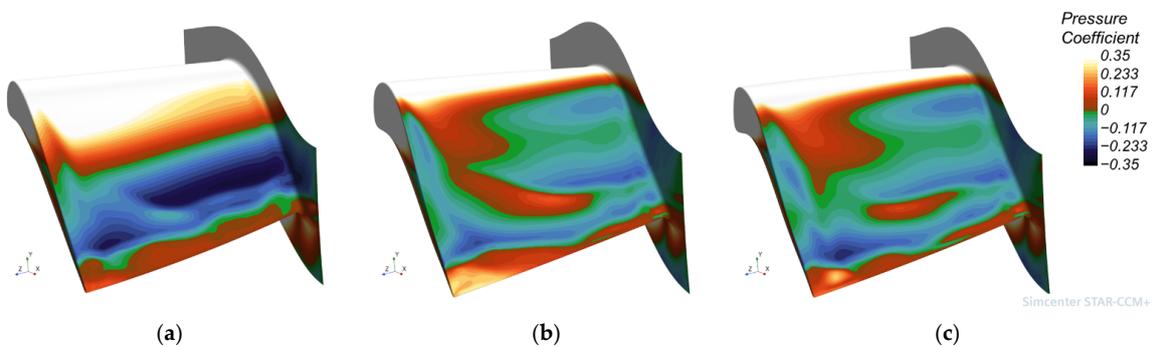


Figure 13. Pressure coefficient scenes on the surfaces of the blades: (a) Baseline design; (b) Design A; and (c) Design B.

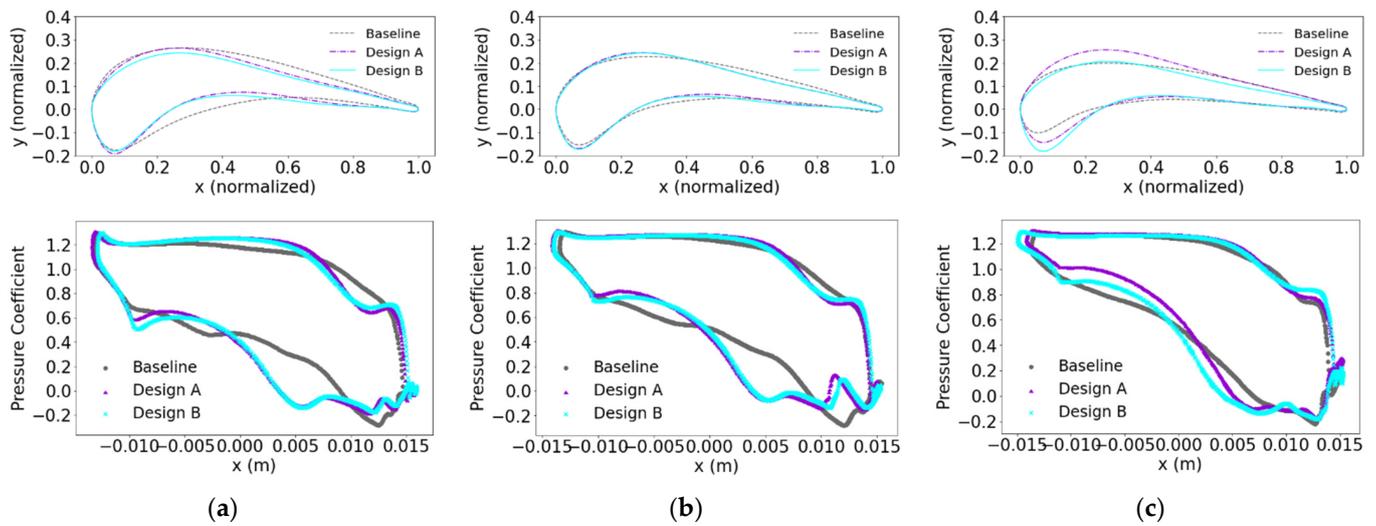


Figure 14. Blade profiles (first row) and pressure coefficients in the streamwise direction (second row): (a) hub sections comparison; (b) mid sections comparison; and (c) tip sections comparison.

To further investigate the differences in solutions, a volume rendering scene is created by focusing on visualizing changes in Mach number within the range between 0.8 and 1.2, shown in Figure 15. The goal of this plot is to reveal where sudden changes in Mach number may occur in the transonic flow regime. A consistent shock structure near the trailing edge covering the full span length of the blade is observed in the baseline design, which is consistent with the high pressure spot in the same area highlighted by the dotted circle in Figure 11a. This shock feature is missing in the two optimized designs (Designs A and B), due to the weaker gradients in those designs.

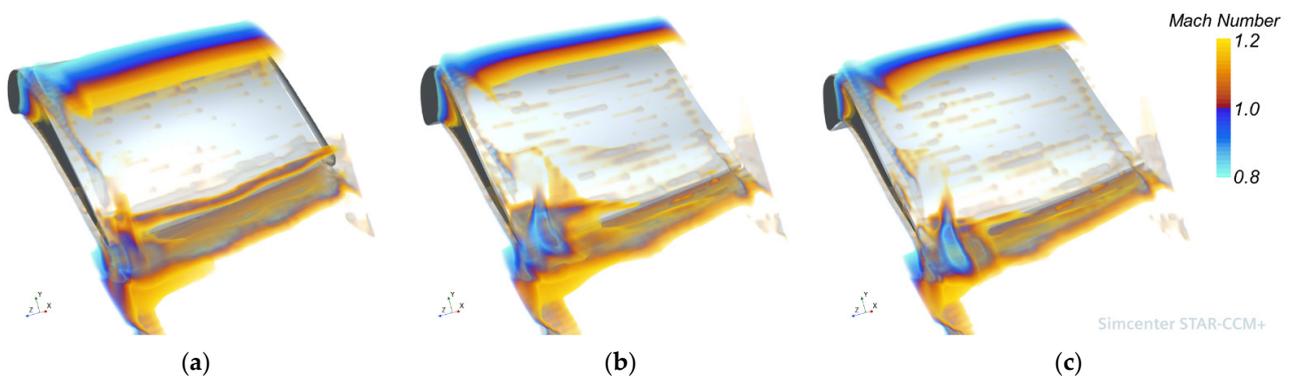


Figure 15. Mach number rendering: (a) Baseline design; (b) Design A; and (c) Design B.

A further comparison is made by plotting the streamlines in the tip leakage regions and showing pressure coefficients on spanwise cross-sections in Figure 16. It appears the two optimized designs feature slightly more chaotic Mach number distributions in the local region near the trailing edge and tip leakage. The leakage flow’s influence on the overall performance is relatively small in comparison to the previously discussed suction-side effects. It can be observed that the streamlines in the high-efficiency design (Design A) are slightly more curved, yet the induced low-pressure spots by the secondary flow effects of the tip leakage flows are weak and about the same magnitude. Overall, the difference caused by the tip leakage flow is small in these three designs. The performance improvements in the optimized designs are more influenced by other flow effects discussed earlier, such as the increased minimum pressure on the suction side near the trailing edge (Figures 11–13), and the reduction of the shock structure (Figure 15).

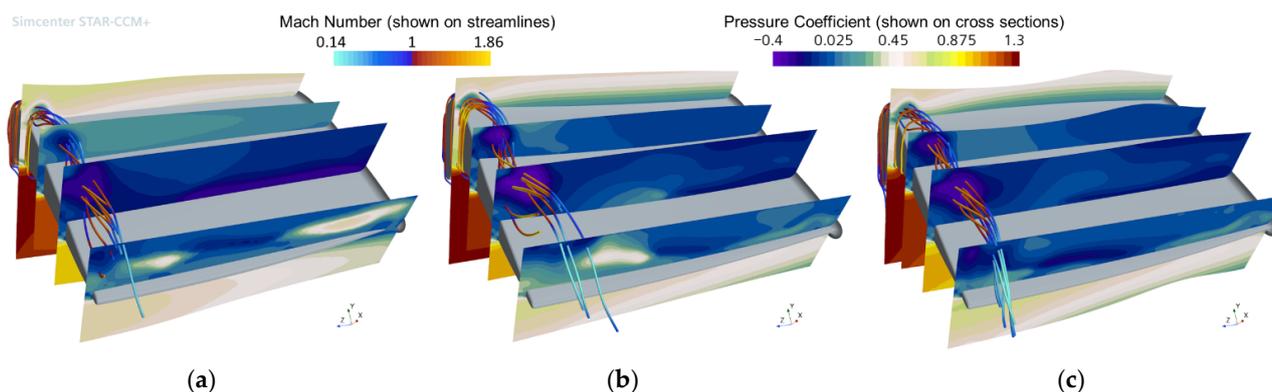


Figure 16. Tip leakage flow streamlines and pressure coefficients on spanwise cross-sections: (a) Baseline design; (b) Design A; and (c) Design B.

4. Conclusions

The present study has demonstrated an optimization workflow combining the use of neural networks and high-fidelity CFD. The neural network models were trained on over three-thousand design data points from a previous publication. The practical implication of the overall strategy is that in engineering design analysis, existing data sets, which are generally available from the previous simulation, experimental, or reduced-order studies, can be leveraged to build neural network models, which can then be used in combination with high-fidelity CFD simulations to guide optimization processes. This approach achieves a reduction in the required number of high-fidelity CFD runs, and hence reduces the computational cost while maintaining accuracy. The integration of computationally inexpensive ANN models, which were evaluated 40,750 times, allows for a relatively small number (163) of CFD evaluations in the present optimization process, resulting in a total run time of about 30 h (with 6 CFD evaluations running in parallel, each consuming 160 compute cores). It is estimated that if the nested optimization strategy based on ANN was not used, a total number of 733 CFD evaluations will be required due to the large number of design variables exposed in 3D CFD evaluations, resulting in roughly 135 h of the optimization run time if the same compute resource settings were used.

The efficacy of the methodology is demonstrated on a turbine blade aerodynamic problem. ANN models with 7 layers and 6 layers were built to represent two blade performance metrics, efficiency, and power, respectively. The hyperparameters of the ANN models were optimized, and the models were used as surrogate models along with high-fidelity CFD simulations in a nested optimization procedure to obtain optimized blade designs. Pareto front designs representing improved efficiency and power were found by the optimization procedure. A lean angle and a tip scaling factor were shown to be more favored by the optimization procedure than other parameters in the context of the chosen blade and analysis methodology used in the current study. Examining the fluid dynamics of the optimized designs vs. the baseline design reveals that the optimization (1) reduced the magnitude of the most negative pressure coefficients in the flow on the suction side near the trailing edge, and (2) altered the blade geometry that reduced the shock near the trailing edge. Both of these aspects have effects on improved efficiency and power in the optimized designs.

As a future extension of this study, the following may be considered. (1) Test the performance of other response surface methods and meta models against ANN and apply them in the nested optimization workflow. (2) The present study demonstrates a nested ANN-CFD optimization methodology, applied to an idealized turbine CFD problem as a proof-of-concept. In production-level gas turbine designs, more detailed constraints on the geometry or loading curve must be incorporated to ensure more realistic optimization outcomes. Different power output conditions of the engine may also be explored in optimization.

Author Contributions: Conceptualization, C.Z.; methodology, C.Z. and M.J.; software, C.Z. and M.J.; validation, C.Z. and M.J.; formal analysis, C.Z.; investigation, C.Z. and M.J.; resources, C.Z. and M.J.; data curation, C.Z.; writing—original draft preparation, C.Z.; writing—review and editing, C.Z. and M.J.; visualization, C.Z.; supervision, M.J.; project administration, C.Z. and M.J.; funding acquisition, M.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: This work is a part of the authors' professional development initiatives sponsored by the Simcenter Customer Support organization of Siemens Digital Industries Software. The authors wish to thank their colleagues at Siemens Digital Industries Software. Additional thanks go to Yutao He at Jet Propulsion Laboratory, Caltech, for discussions on the research methodology.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

c_p	pressure coefficient	
h	fluid enthalpy	J/kg
p	pressure	Pa
\mathcal{P}	performance function	
Pow	power	W
r	radius	m
τ	Pearson coefficient	
u	velocity	m/s
Δh	enthalpy change	J/kg
w	blade profile parameter	
η	efficiency	
ρ	density	kg/m ³
σ_c	centrifugal stress	Pa
Ω	rotating speed	rad/s

Subscripts

h	hub
i	domain inlet
l	lower surface
max	maximum quantity
o	domain outlet
s	solid
t	tip
u	upper surface

References

- Alexandrov, N.M.; Lewis, R.M.; Gumbert, G.R.; Green, L.L.; Newman, P.A. Approximation and Model Management in Aerodynamic Optimization with Variable-Fidelity Models. *J. Aircr.* **2001**, *38*, 1093–1101. [[CrossRef](#)]
- Ong, Y.S.; Nair, P.B.; Keane, A.J. Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling. *AIAA J.* **2003**, *41*, 687–696. [[CrossRef](#)]
- Mazaheri, K.; Zeinalpour, M.; Bokaei, H.R. Turbine Blade Cooling Passages Optimization Using Reduced Conjugate Heat Transfer Methodology. *Appl. Therm. Eng.* **2016**, *203*, 1228–1236. [[CrossRef](#)]
- Jiang, Y.; Lin, H.; Yue, G.; Zheng, Q.; Xu, X. Aero-Thermal Optimization on Multi-Rows Film Cooling of a Realistic Marine High Pressure Turbine Vane. *Appl. Therm. Eng.* **2016**, *111*, 537–549. [[CrossRef](#)]
- Braining, E.; Weidtmann, K.; Buehler, P.; Lautenschlaeger, A.; Braun, R. Improvement of an Automated Design Process and Investigation of the Optimization Approach and the Dependencies of the Individual Objectives. In Proceedings of the GPPS Forum 18, Global Power and Propulsion Society, GPPS-2018-0027, Montreal, QC, Canada, 7–9 May 2018.
- Braining, E.; Weidtmann, K.; Braun, R.; Rincon DL, H.; Bohn, D. Automated 2D Airfoil Optimization of Intentionally Choked Blades. In Proceedings of the Global Power and Propulsion Society, GPPS-BJ-2019-0067, Beijing, China, 16–18 September 2019.
- Zhang, C.; Hodges, J.; Janeway, M. Co-Optimization of Turbine Blade Aero and Thermal Designs Based on Computational Fluid Dynamics (CFD) Models. In Proceedings of the 5th–6th Thermal and Fluids Engineering Conference, New Orleans, LA, USA, 26–28 May 2021.

8. Pierezan, J.; Maidl, G.; Yamao, E.M.; Coelho, L.S.; Mariani, V.C. Cultural Coyote Optimization Algorithm Applied to a Heavy Duty Gas Turbine Operation. *Energy Convers. Manag.* **2019**, *199*, 111932. [[CrossRef](#)]
9. Bettocchi, R.; Pinelli, M.; Spina, P.R.; Venturini, M. Set up of a Robust Neural Network for Gas Turbine Simulation. In Proceedings of the ASME Turbo Expo 2004, Power for Land, Sea, and Air, Vienna, Austria, Jun 14–17 June 2004.
10. Li, D.; Qiu, L.; Tao, K.; Zhu, J. Artificial Intelligence Aided Design of Film Cooling Scheme on Turbine Guide Vane. *Propuls. Power Res.* **2020**, *9*, 344–354. [[CrossRef](#)]
11. Belov, S.; Nikolaev, S.; Uzhinsky, I. Hybrid Data-Driven and Physics-Based Modeling for Gas Turbine Prescriptive Analytics. *Int. J. Turbomach. Propuls. Power* **2020**, *5*, 29. [[CrossRef](#)]
12. Giorgi, M.G.D.; Quarta, M. Hybrid MultiGene Genetic Programming—Artificial Neural Networks Approach for Dynamic Performance Prediction of an Aeroengine. *Aerosp. Sci. Technol.* **2020**, *103*, 105902. [[CrossRef](#)]
13. Zhang, M.; Hao, S.; Hou, A. Study on the Intelligent Modeling of the Blade Aerodynamic Force in Compressor Based on Machine Learning. *Mathematics* **2021**, *9*, 476. [[CrossRef](#)]
14. Wang, Q.; Yang, L.; Rao, Y. Establishment of a Generalizable Model on a Small-Scale Dataset to Predict the Surface Pressure Distribution of Gas Turbine Blades. *Energy* **2021**, *214*, 118878. [[CrossRef](#)]
15. Park, Y.; Choi, M.; Kim, K.; Li, X.; Jung, C.; Na, S.; Choi, G. Prediction of Operating Characteristics for Industrial Gas Turbine Combustor Using an Optimized Artificial Neural Network. *Energy* **2020**, *213*, 118769. [[CrossRef](#)]
16. Sun, L.; Liu, T.; Xie, Y.; Zhang, D.; Xia, X. Real-Time Power Prediction Approach for Turbine Using Deep Learning Techniques. *Energy* **2021**, *233*, 121130. [[CrossRef](#)]
17. Akolekar, H.; Zhao, Y.; Sandberg, R.D.; Pacciani, R. Integration of Machine Learning and Computational Fluid Dynamics to Develop Turbulence Models for Improved Turbine Wake Mixing Prediction. In Proceedings of the ASME Turbo Expo 2020: Turbomachinery Technical Conference and Exposition, London, UK, 22–26 June 2020.
18. Brunton, S.L.; Kutz, J.N.; Manohar, K.; Aravkin, A.Y.; Morgansen, K.; Klemisch, J.; Goebel, N.; Buttrick, J.; Poskin, J.; Blom-Schieber, A.W.; et al. Data-Driven Aerospace Engineering: Reframing the Industry with Machine Learning. *AIAA J.* **2021**, *59*, 2820–2847. [[CrossRef](#)]
19. Wang, H.; Xi, G. Optimization of High Speed Centrifugal Compressor for a Micro Gas Turbine Based on CFD and FEM Analysis. In Proceedings of the ASME Turbo Expo 2010: Power for Land, Sea, and Air, Glasgow, UK, 14–18 June 2010.
20. Maral, H.; Alpman, E.; Kavurmacioglu, L.; Camici, C. A Genetic Algorithm Based Aerothermal Optimization of Tip Carving for an Axial Turbine Blade. *Int. J. Heat Mass Transf.* **2019**, *143*, 11849. [[CrossRef](#)]
21. Yan, X.; Zhu, J.; Kuang, M.; Wang, X. Aerodynamic Shape Optimization Using a Novel Optimizer Based on Machine Learning Techniques. *Aerosp. Sci. Technol.* **2019**, *86*, 826–835. [[CrossRef](#)]
22. Li, J.; Zhang, M.; Martins, J.R.R.A.; Shu, C. Efficient Aerodynamic Shape Optimization with Deep-Learning-Based Geometric Filtering. *AIAA J.* **2020**, *58*, 4243–4259. [[CrossRef](#)]
23. Wang, Y.; Liu, T.; Zhang, D.; Xie, Y. Dual-Convolutional Neural Network Based Aerodynamic Prediction and Multi-Objective Optimization of a Compact Turbine Rotor. *Aerosp. Sci. Technol.* **2021**, *116*, 106869. [[CrossRef](#)]
24. Elsayed, K.; Lacor, C. Modeling, Modeling, Analysis and Optimization of Aircyclones Using Artificial Neural Network Response Surface Methodology and CFD Simulation Approaches. *Powder Technol.* **2011**, *212*, 115–133. [[CrossRef](#)]
25. Chamoli, S. ANN and RSM Approach for Modeling and Optimization of Designing Parameters for a V down Perforated Baffle Roughened Rectangular channel. *Alex. Eng. J.* **2015**, *54*, 429–446. [[CrossRef](#)]
26. Park, D.; Cha, J.; Kim, M.; Go, G.S. Multi-objective Optimization and Comparison of Surrogate Models for Separation Performance of Cyclone Separator Based on CFD, RSM, GMDH-Neural Network, Back Propagation-ANN and Genetic Algorithm. *Eng. Appl. Comput. Fluid Mech.* **2020**, *14*, 180–201. [[CrossRef](#)]
27. Halila, E.E.; Lenahan, D.T.; Thomas, T.T. Energy Efficient Engine High Pressure Turbine Test Hardware Detailed Design Report; NASA-CR-167955; NASA Lewis Research Center: 1982. Available online: <https://ntrs.nasa.gov/citations/19850002687> (accessed on 25 April 2022).
28. Timko, L.P. Energy Efficient Engine High Pressure Component Test Performance Report; NASA-CR-168289; NASA Lewis Research Center: 1990. Available online: <https://ntrs.nasa.gov/citations/19900019237> (accessed on 25 April 2022).
29. Chen, S.-C.S. Axial-Flow Turbine Rotor Discharge-Flow Overexpansion and Limit-Loading Condition, Part 1: Computational Fluid Dynamics (CFD) Investigation; NASA/TM-2017-219506; Glenn Research Center: 2017. Available online: <https://ntrs.nasa.gov/citations/20170007293> (accessed on 25 April 2022).
30. Kulfan, B. Universal Parametric Geometry Representation Method. *J. Aircr.* **2008**, *45*, 142–158. [[CrossRef](#)]
31. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. Tensorflow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, Savannah, GA, USA, 2–4 November 2016; USENIX Association: Berkeley, CA, USA, 2016; pp. 265–283.
32. Red Cedar Technology. *SHERPA—An Efficient and Robust Optimization/Search Algorithm*; WP-1023, Rev. 05.08; Red Cedar Technology: East Lansing, MI, USA, 2021.
33. Zhang, C. A Python Program—Machine Learning Regression for Gas Turbine Blade Efficiency and Power, Version 1.0.0; Computer software; 2022. Available online: <https://github.com/Cool-Chaoz/Machine-Learning-Regression-for-Gas-Turbine-Blade-Efficiency-and-Power> (accessed on 25 April 2022).

34. Mohaghegh, K.; Sadeghi, M.H.; Abdullah, A.; Boutorabi, R. Improvement of Reverse-Engineered Turbine Blades Using Construction Geometry. *Int. J. Manuf. Technol.* **2010**, *49*, 675–687. [[CrossRef](#)]
35. Menter, F.R. Two-equation eddy-viscosity turbulence modeling for engineering applications. *AIAA J.* **1994**, *32*, 1598–1605. [[CrossRef](#)]