POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Data Driven Approach to Turbomachinery Blade Design

Master of Science in Aeronautical Engineering - Ingegneria Aeronautica

Author: Antonio Pucciarelli

Advisor: Prof. Paolo Gaetani

Promoter: Prof. Sergio Lavagnoli

Academic year: 2022-2023

---

1. Introduction

In the field of turbomachinery design, current models are often slow, repetitive, and heavily reliant on computer simulations, often overlooking valuable existing knowledge [1]. This work introduces a groundbreaking methodology that seamlessly integrates machine learning techniques into the blade design process.

2. Significance and Objectives

The primary significance of this work lies in its potential to revolutionize turbomachinery design by eliminating the need for time-consuming CFD simulations. It provides designers with a clear understanding of blade loading limits and the intricate correlations between loading distribution and blade geometry, thereby shedding light on important physical limitations in the design process.

2.1. Efficiency Enhancement

By harnessing machine learning, this research facilitates a quicker and more autonomous blade design process, breaking free from the constraints of lengthy simulations and conserving both time and resources.

2.2. In-Depth Understanding

This work offers designers profound insights into blade loading limits, emphasizing the significance of making data-driven design decisions.

2.3. Data Processing

The study elucidates the process of data collection and analysis, providing clarity on the integration of artificial intelligence into turbomachinery design.

3. Practical Implications

This research not only introduces a novel approach to turbomachinery blade design but also lays the foundation for more effective design strategies. It empowers designers with tools to optimize efficiency and accuracy in the design process.

4. Problem Framing

The following sections provide a summary of the work. The work aims to generate a database, which will then be utilized by a machine learning algorithm for the generation of a mathematical function, $\hat{f}$, that facilitates blade design [2]. This design function, $\hat{f}$, takes as input a desired loading distribution and a desired flow deflection

made by the blade, and it outputs the blade geometry that achieves this flow behavior. This research begins by framing the problem effectively.

4.1. Dimensionality

Due to the multidimensional characteristics of the problem, it is necessary to reduce the field of study to its minimum extent [1]. Dimensionality reduction enables the use of fewer parameters for problem description and provides better control over important correlations between flow properties and geometric features in the dataset.

4.1.1 Macroscopic Flow Properties

A way of describing the macroscopic working conditions of the flow is to define:

- α_1 : inlet flow angle
- α_2 : outlet flow angle
- M_2 : outlet Mach number
- Re: Reynolds number of the flow

These four parameters, represented in Figure 1, are the minimum requirements for the definition of a working condition or duty of the blade. As a result, these parameters will be identified as the aerodynamic duty of the blade.

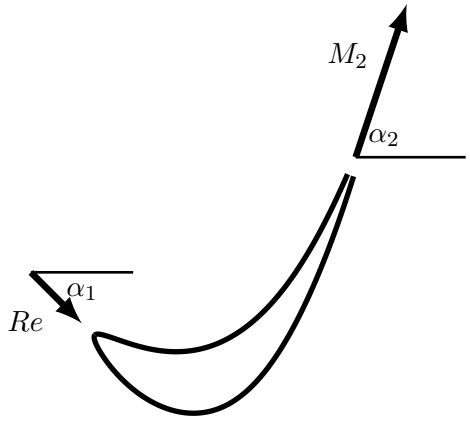

Figure 1: Aerodynamic duty parameters.

4.1.2 Local Flow Properties

The loading distribution is defined by the surface fraction, $\frac{S}{S_{TOT}}$, and the Mach fraction number, $\frac{M}{M_{TE}}$. The load distribution is determined using the surface fraction, $\frac{S}{S_{TOT}}$, instead of the chord because it provides a better definition of the local loading distribution in critical regions, such as the leading edge, and has a more direct correlation with boundary layer behavior since its properties are defined by the path length traveled by the flow.

The loading distribution is parametrized by:

- Leading edge Mach fraction (M_{LE}): Defines the load at the leading edge on the suction side of the blade.
- Peak Mach fraction (M_{PEAK}): Represents the highest Mach value over the blade, found on the suction side.
- Pressure Mach number (M_{PRESS}): Provides a dual descriptor, indicating the leading edge load on the suction side and the Mach fraction before it rises to reach the trailing edge on the pressure side.
- Surface fraction position (S_{PEAK}): Identifies the position where the peak Mach fraction is located along the load distribution.

These parameters are crucial in defining the local load distribution along the blade. The loading distribution is constructed using a set of Bezier splines controlled by the parameters mentioned above.

Trial & Error Because the many possible loading distributions. A large investigation was made on suitable loading patterns. These patters are the ones which guarantee a geometry which fullfils the macroscopic and local flow properties [1]. The reason of this investigation relies on the fact that the database has to contain only information which are useful for the aim of the work. Studying a loading distribution which does not have any geometry associated to the target loading distribution, it is completely unnecessary. Fine-tuning the blade's loading distribution is crucial to meeting both manufacturing requirements and aerodynamic style constraints. A trial-and-error approach, particularly at the leading edge, enhances blade convergence and exit flow angle prediction, optimizing results while considering manufacturing limitations.

Figure 2, Figure 3, Figure 4 and Figure 5 represent all the possible loading variation.

4.1.3 Blade Geometry

The blade geometry - consisting of the camberline, suction side and pressure side - is defined

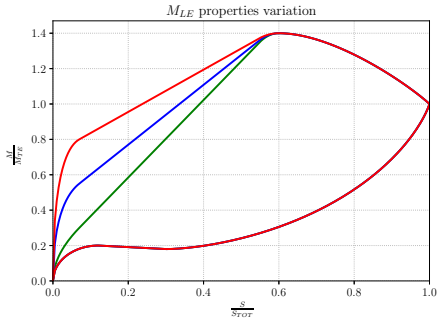

Figure 2: $\frac{M}{M_{TE}}$ *vs* $\frac{S}{S_{TOT}}$. M_{LE} variation.

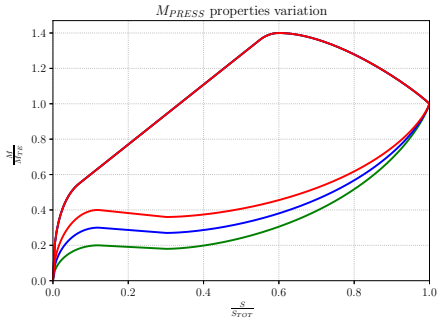

Figure 3: $\frac{M}{M_{TE}}$ *vs* $\frac{S}{S_{TOT}}$. M_{PRESS} variatio

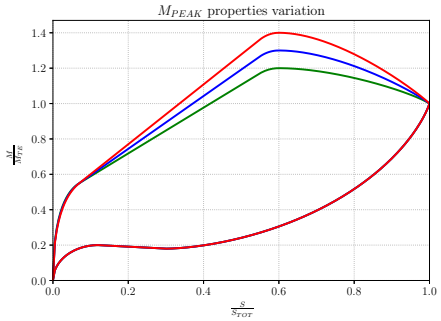

Figure 4: $\frac{M}{M_{TE}}$ *vs* $\frac{S}{S_{TOT}}$. M_{PEAK} variation.

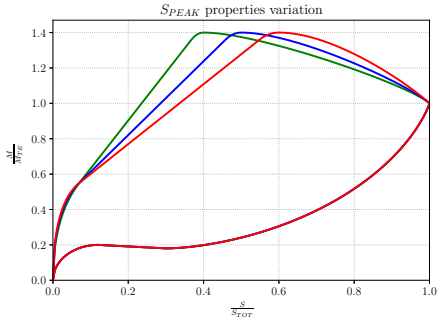

Figure 5: $\frac{M}{M_{TE}}$ *vs* $\frac{S}{S_{TOT}}$. S_{PEAK} variation.

through meticulous parametrization, following Kulfan's approach.
The camberline is parameterized by the stagger

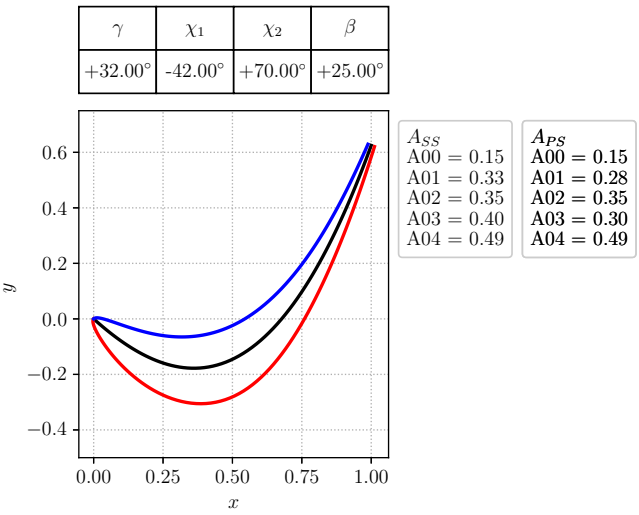

Figure 6: Kulfan parametrized blade.

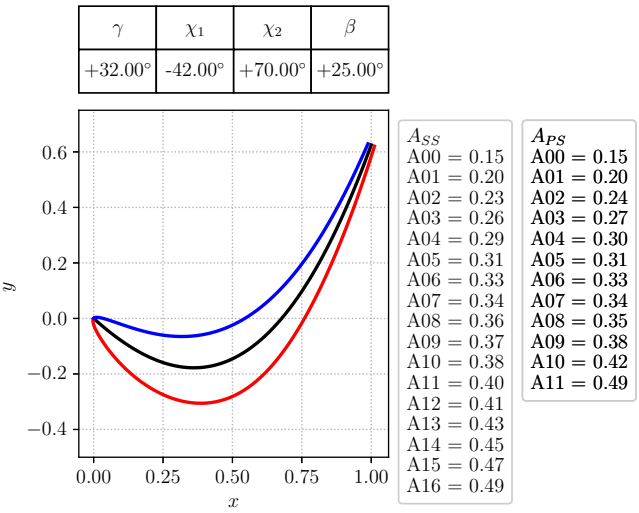

Figure 7: Figure 6 scaled blade with a higher parametrization.

angle (γ), the metal inlet angle (χ_1), and the metal outlet angle (χ_2). The suction side and the pressure side of the blade are defined using a set of N parameters, A_i. Scaling a blade can be accomplished using additional parameters by solving a linear system of equations.
Kulfan's parametrization [3] preserves critical geometric properties, including the leading-edge radius (R_{LE}) and the trailing-edge angle (β), which have a significant impact on blade characteristics and flow properties.
Figure 6 and Figure 7 show the Kulfan parametrized blade and the same blade scaled

with a higher parametrization.

4.2. Database Generation

The database is created using an in-house code designed for blade optimization and analysis, appropriately named `datablade`. The program's structure is modular, enabling flexible combinations of blocks tailored to specific objectives.

4.2.1 Configuration File

`datablade` optimization relies on a configuration file, formatted in `.json` for enhanced readability and ease of use. This file serves as the blueprint for blade optimization, initializing parameters such as the initial guess for blade geometry, geometrical constraints setup, and loading properties.

4.2.2 Optimization

The blade optimization plays a pivotal role, employing a classic gradient-free method known as the Simplex method. This method, while seeking local minima, delivers satisfactory results when guided by suitable initial guesses, dimensionality adaptation strategies, and a robust cost function.

Dimensionality Adaptation Optimizing blades involves solving multidimensional optimization problems with complex solutions. `datablade` adopts an intelligent strategy that adjusts the problem's dimensionality. Initially, optimization commences with fewer parameters to reduce the design space. As convergence is achieved, the degree of freedom is incrementally increased, leading to improved solutions. Empirical tests have validated this approach, demonstrating its effectiveness in constructing the database.

Cost Function The core of blade optimization revolves around the management of two key errors: load distribution and flow exit angle. These errors are quantified using the root mean squared error ($RMSE$) and the flow exit angle error ($\Delta\alpha_2$). The cost function, which combines these properties, sets criteria for optimization convergence and establishes acceptance thresholds for optimized blades. The function balances the significance of the angle error ($\Delta\alpha_2$) relative to $RMSE$ through the use of scaling and threshold values.

$$RMSE = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(\frac{M_{real}}{M_{TE,real}}\bigg|_i - \frac{M_{target}}{M_{TE,target}}\bigg|_i \right)^2} \quad (1)$$

$$\Delta\alpha_2 = \left| \alpha_{2,real} - \alpha_{2,target} \right| \quad (2)$$

$$cost = RMSE \cdot \left[1 + 0.04 \cdot \Big(max\Big(0,\ \Delta\alpha_2 - 1.0\Big) \Big)^{2.0} \right] \quad (3)$$

4.2.3 Optimizer

The `datablade` optimizer combines the capabilities of the `MISES` software [4] for flow properties computation with the `scipy` module using the Simplex method [5], providing robust performance and efficiency.

4.2.4 Domain

Table 1: Domain boundaries and discretization.

Variable	Min	Max	Points
α_1	$-50°$	$-20°$	3
α_2	$65°$	$72.5°$	4
M_2	0.4	0.7	3
Re	$6 \cdot 10^5$	$6 \cdot 10^5$	1
$\frac{M_P}{M_{TE}}$	1.2	1.4	3
$\frac{L_P}{L_{surf}}$	0.5	0.6	3
$\frac{M_{LE}}{M_{TE}} \frac{M_2}{M_1}$	1.2	1.8	3
$\frac{M_{PS}}{M_{TE}} \frac{M_2}{M_{1,ax}}$	0.8	1.2	3

The database is created by optimizing blades with different aerodynamic styles and aerodynamic duty properties. The dataset points are presented in Table 1.
For the high-pressure turbine study, a fixed Reynolds number ($Re = 6 \times 10^5$) is employed due to the Reynolds independence of the flow under investigation.

An innovative optimization strategy unfolds as an additional acceleration layer for computing the initial guess ($\boldsymbol{x}_0$) used in the optimizer. This strategy involves linearizing the inner domain using data from outer domain points. The outer points are optimized first to provide a suitable initial guess for inner domain points, thereby streamlining the overall convergence speed of the database [1].

Outer Points Generating the corner points of the domain is the initial and time-consuming step in database creation. These corner points serve as a foundation, expediting the optimization process for inner domain points. `datablade` optimizes 128 blades to establish these corner points.
Figure 8 illustrates the boundary points of the study domain.

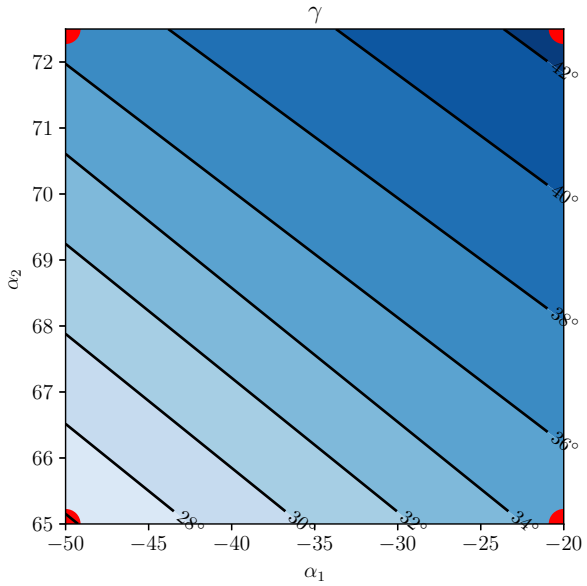

Figure 8: Linear interpolation of the stagger angle, γ, with respect to α_1 and α_2 from the corner points in red.

Inner Points After generating the corner points, the initial guess for inner points, denoted as $\boldsymbol{x}_0$, is computed using linear interpolation. This method simplifies the estimation of blade parameters across the entire inner domain. A total of 2916 blades will undergo optimization for inner points, culminating in the creation of the comprehensive database.

Figure 9 displays the initial guess, γ, obtained through linear interpolation of the domain.

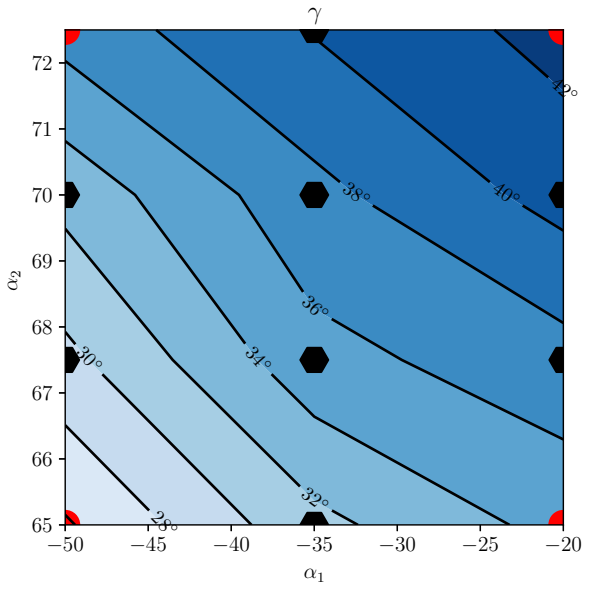

Figure 9: Complete γ variation with respect to α_1 & α_2. Corner points in red and inner points in black.

4.2.5 Optimized Data

Table 2: Data properties inside the optimized database.

	count	μ	σ	*min*	*max*
cost	2916	0.01	0.006	0.005	0.04
$\boldsymbol{\Delta\alpha_2}$	2916	0.96°	0.542°	0.0003°	2.37°

The database contains Kulfan's parameters that define blade geometry. Key blades of interest for initial analysis are categorized based on their loading distribution and exit flow angle errors:

- Blades with low errors in both the loading distribution and the exit flow angle. Figure 10 illustrates this blade setup.
- Blades with low errors in the loading distribution but high errors in the exit flow angle. Figure 11 represents this blade behavior.
- Blades with high errors in both the loading distribution and the exit flow angle. Figure 12 depicts this blade family.

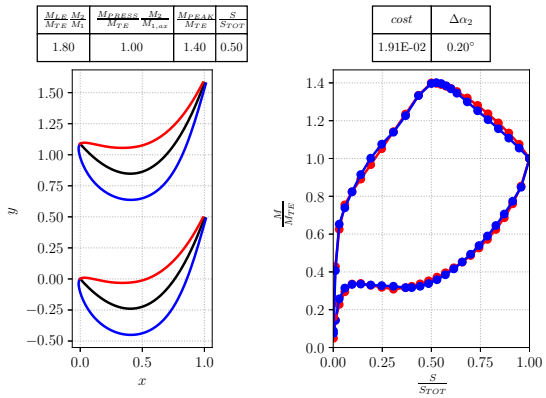

Figure 10: Blade with good performances on the load distribution and on the exit angle error.

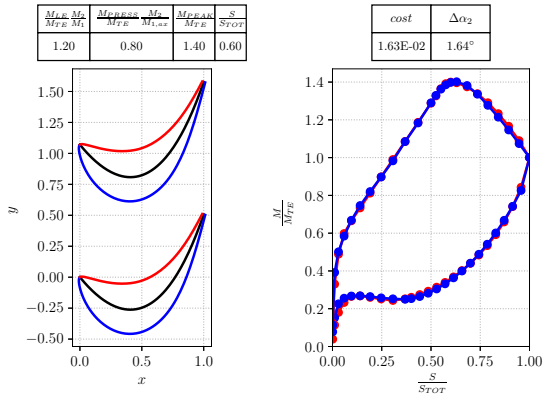

Figure 11: Blade with good performances on the load distribution but low performance on the exit angle error.

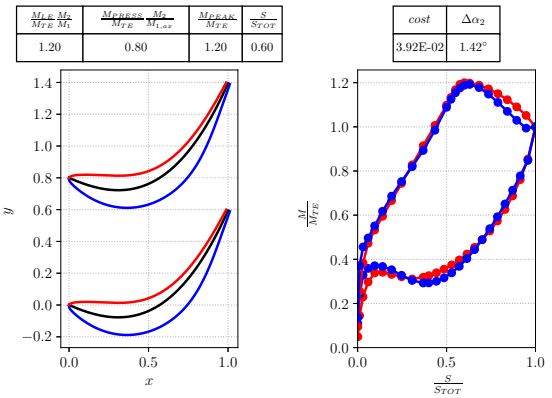

Figure 12: Blade with poor performances on the load distribution and on the exit angle error.

Table 2 provides an overview of the complete database after optimization, showcasing acceptable mean values for cost and $\Delta\alpha_2$. A significant portion of the blades demonstrates a cost below 2.75%, indicating convergence towards the desired aerodynamic style. However, it is worth noting that over 25% of the database exhibits an exit flow angle error, $\Delta\alpha_2$, exceeding 1°, underscoring the need for correction.
Some blades perform poorly and are excluded from the machine learning analysis due to significant discrepancies in loading distribution. These blades lack correlation with the desired aerodynamic style.

4.2.6 Error Distribution

Certain blades within the database exhibit significant errors in the loading distribution ($RMSE$). These infeasible designs, characterized by erroneous input data, cannot be rectified by machine learning algorithms. As a result, infeasible designs are excluded from the database to ensure the availability of meaningful data for interpolation. A threshold of 2.75% is used as filter for refining the database.
Figure 13 and Figure 14 illustrate the error distribution on a *slice* of the domain.

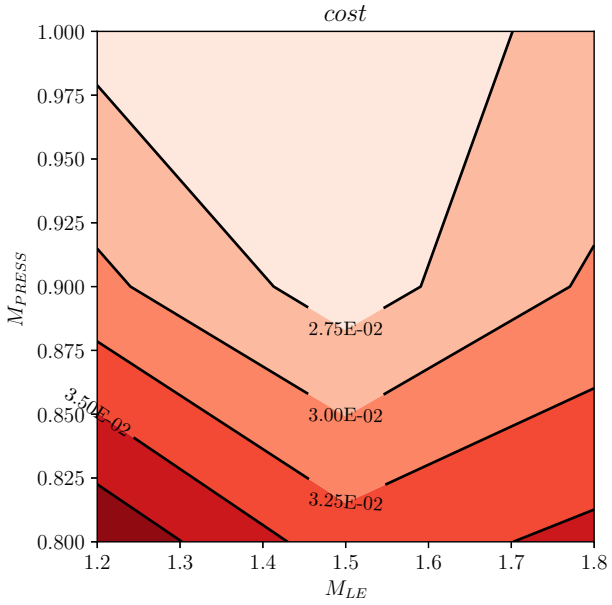

Figure 13: *cost* distribution.

4.2.7 Filtering

To refine the database for interpolation purposes, data filtering is implemented. Blades with

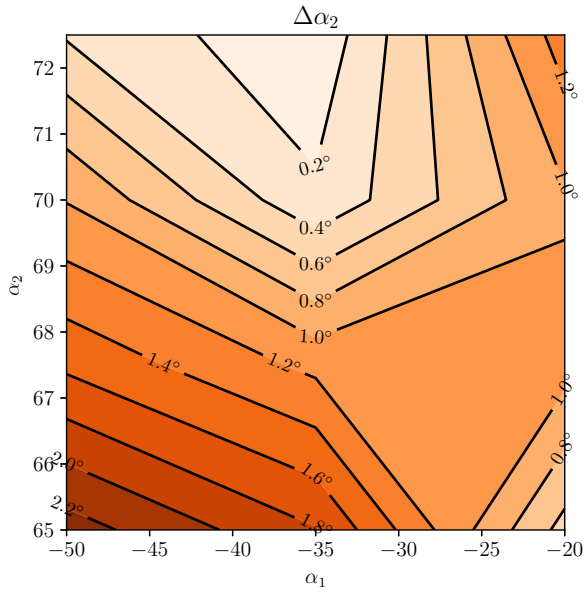

Figure 14: $\Delta\alpha_2$ behavior in the domain.

unacceptable errors in blade loading are omitted, resulting in a slight reduction in the dataset. The infeasible designs are typically located at the domain boundaries, justifying their exclusion. The filtering process primarily relies on the blade's cost properties, as it predominantly reflects the $RMSE$.
Blades with low $RMSE$ but high $\Delta\alpha_2$ will be subsequently filtered numerically by the machine learning algorithm through a correction of the blade design function, $\hat{f}$.
Table 3 displays the database properties following the filtering operation.

Table 3: Data properties inside the optimized database.

	count	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$	***min***	***max***
cost	2787	0.015	0.004	0.005	0.027
$\boldsymbol{\Delta\alpha_2}$	2787	0.846°	0.421°	0.001°	2.373°

5. Machine Learning

The heart of the machine learning algorithm is constructed upon regression techniques, specifically tailored for domain interpolation. This algorithm excels in the remarkable task of predicting blade geometries based on inputs of aerodynamic duty and aerodynamic style, mirroring the parameterization of the database's blades.
The domains of study are as follows:

- $\mathcal{X}$: This dataset stores the aerodynamic style and aerodynamic duty properties, organized as $\mathcal{X} \in \mathbb{R}^{2787\times 8}$, representing 2787 filtered blades and 8 parameters responsible for defining the aerodynamic style and duty. The $\boldsymbol{x}$ vector holds the aerodynamic style and aerodynamic duty of the blade and is a general element within $\mathcal{X}$.
- $\mathcal{Y}$: This dataset stores the camberline and Kulfan parameters, essential for characterizing each blade. It is structured as $\mathcal{Y} \in \mathbb{R}^{2787\times 42}$, indicating 2787 blades and 42 parameters, including Kulfan parameters and pitch. The $\boldsymbol{y}$ vector encompasses the blade geometry properties and is a general element within $\mathcal{Y}$.

The machine learning algorithm is tasked with approximating a mapping function, f, which bridges the two domains, $\mathcal{X}$ and $\mathcal{Y}$, as depicted in Equation 4.
The subsequent endeavor revolves around computing $\hat{f}$, a numerical approximation of f. This new function, $\hat{f}$, plays a pivotal role in defining the underlying problem.

$$f_{(\boldsymbol{x})} = \boldsymbol{y} \tag{4}$$

$$\hat{f}_{(\boldsymbol{x})} \approx \boldsymbol{y} \tag{5}$$

5.1. Radial Basis Function

To achieve the research goals, the selection of a suitable regression algorithm is crucial. In this study, radial basis functions (RBFs) are chosen as the kernel for the machine learning model. RBFs, known for their robustness, prove to be a powerful tool for data interpolation [6]. They effectively handle overfitting issues and enable perfect interpolation of training points $(\boldsymbol{x}^*, \boldsymbol{y}^*)$. The kernel relies on parameters such as length scale and the Euclidean norm between domain points, which contribute to its effectiveness in minimizing the influence of distant data points. The utilization of Gaussian functions to approximate the domain is another significant aspect, ensuring the smooth transition of geometrical properties of the blade, including camberline and Kulfan parameters.

5.2. Training & Testing

With the foundational elements in place, the work progresses to establish a system for computing $\hat{f}$. This system is characterized by a linear system of equations, representing linear combinations of radial basis functions. These equations are weighted by parameters referred to as weights, denoted as $\boldsymbol{w}$, which are essential for fitting the radial basis functions to the database. Once $\boldsymbol{w}$ is computed, the chapter delves into evaluating the quality of weights using the test set $(\mathcal{X}^{**}, \mathcal{Y}^{**})$.

Figure 15, Figure 16, and Figure 17 display blades generated from $\hat{f}$. It can be observed that the error in $RMSE$ consistently remains below $3 \cdot 10^{-2}$, and $\Delta\alpha_2$ is consistently lower than 0.5°. These results are deemed acceptable for the purpose of the design tool generated by this work.

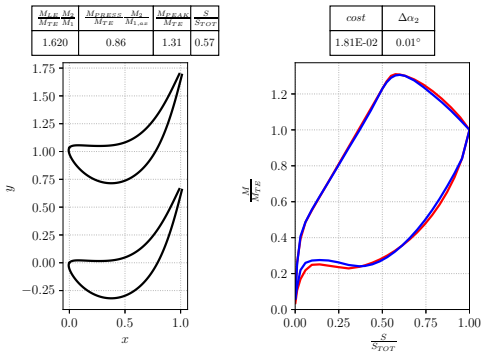

Figure 15: Blade generated by $\hat{f}$ with $\alpha_1 = -34.04°$, $\alpha_2 = 70.15°$ and $M_2 = 0.57$.

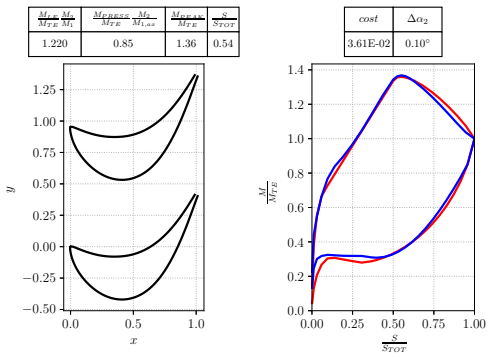

Figure 16: Blade generated by $\hat{f}$ with $\alpha_1 = -49.16°$, $\alpha_2 = 65.16°$ and $M_2 = 0.67$.

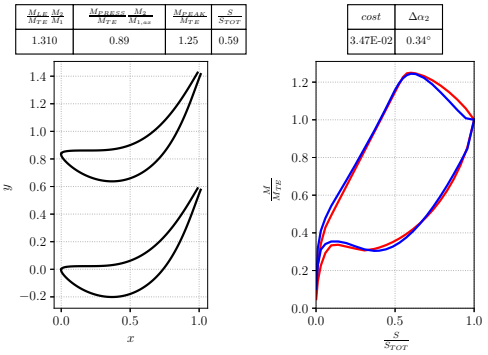

Figure 17: Blade generated by $\hat{f}$ with $\alpha_1 = -20.07°$, $\alpha_2 = 65.16°$ and $M_2 = 0.60$.

6. PCA

The work introduces Principal Component Analysis (PCA) as a robust technique for simplifying complex datasets [6]. In this context, PCA operates solely on the $\mathcal{Y}$ dataset.

PCA commences with a comprehensive analysis of correlations among various blade parameters:

- γ, χ_1, and χ_2 for camberline
- A_{suct} for suction-side parametrization
- A_{press} for pressure-side parametrization
- Pitch

This analysis uncovers the principal correlation directions, essentially representing the eigenvectors that define the $\mathcal{Y}$ dataset. Subsequently, the analysis computes additional modal directions, each signifying the dataset's coverage importance, with higher variance equating to broader coverage.

Notably, the work demonstrates that only three modes can effectively encompass over 95% of the entire dataset, as evidenced in Figure 18.

6.1. Modal Analysis

The exploration continues with modal analysis, which reveals the significance of each mode within the $\mathcal{Y}$ dataset. Modal analysis further refines the research domain, concentrating exclusively on the $\mathcal{Y}$ dataset.

Figure 19 introduces the first mode, emphasizing its substantial influence on camberline variation and the resulting suction-side load distribution.

Figure 20 delves into the second mode, primarily highlighting blade thickness variation and its implications on pressure-side load distribution.

Figure 21 unveils the third mode, highlighting changes in the position of the peak Mach number

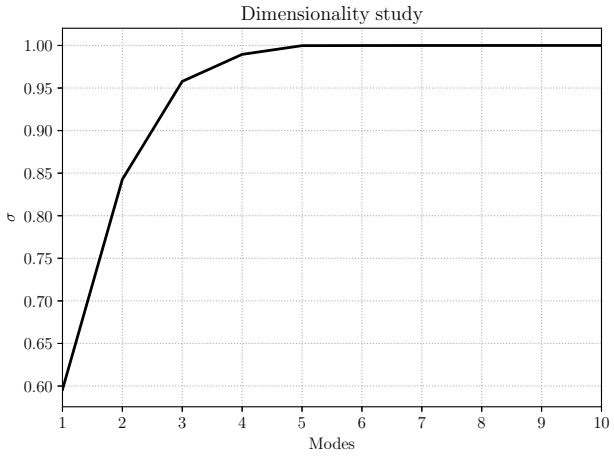

Figure 18: Principal Components of $\mathcal{Y}$ dataset.

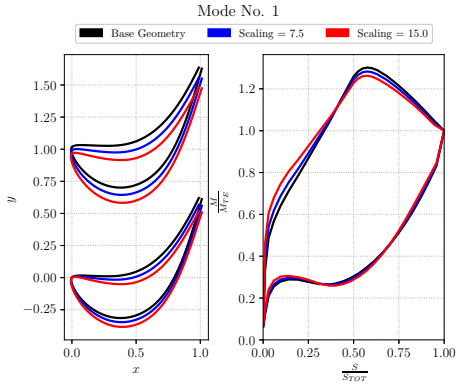

Figure 19: Mode No. 1 with the respective modal loading distribution.

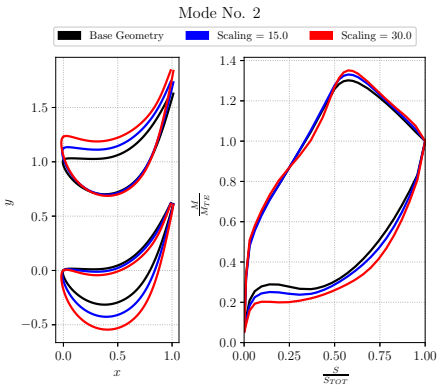

Figure 20: Mode No. 2 with the respective modal loading distribution.

on the suction side and significant alterations in pressure-side Mach distribution.
Additionally, the research emphasizes distinctive modes, including the 10th mode, the 30th mode, and the 42nd mode, each providing unique insights into physical properties.

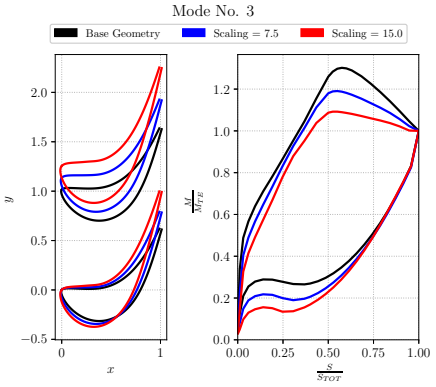

Figure 21: Mode No. 3 with the respective modal loading distribution.

Figure 22 elucidates the impact of the 10th mode, primarily influencing pitch and subsequently altering load distribution over the blade.

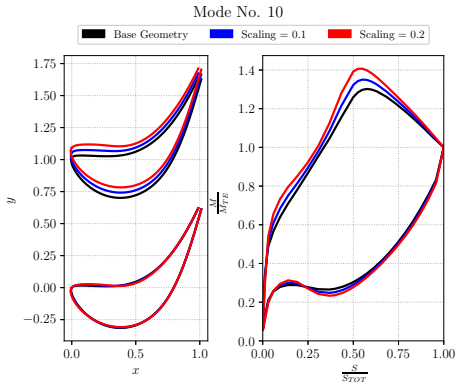

Figure 22: Mode No. 10 with the respective modal loading distribution.

Figure 23 illustrates the consequences of a low-variance mode, indicating that avoiding modes with low variance does not significantly compromise geometry representation accuracy.

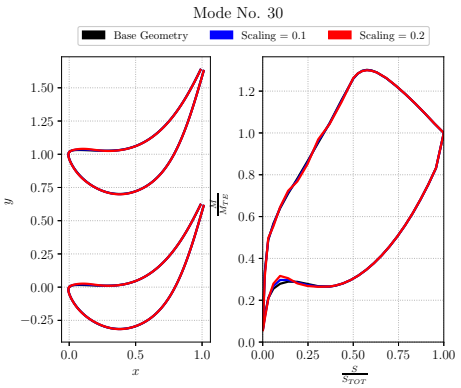

Figure 23: Mode No. 30 with the respective modal loading distribution.

Figure 24 introduces the 42nd mode, characterized as a wobbling noise around the blade, highlighting its minimal importance in the blade domain representation.

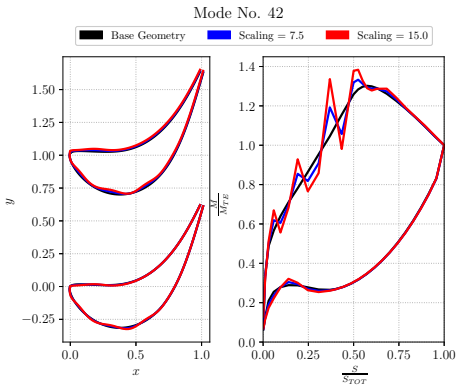

Figure 24: Mode No. 42 with the respective modal loading distribution.

7. Conclusions

7.1. Key Insights and Achievements

The primary achievement of this research lies in the successful validation of the proposed methodology. It demonstrates its effectiveness in blade design, providing both speed and the capability to deconstruct the design space into fundamental geometric modes.

7.2. Implications and Significance

These identified modes hold the promise of crafting blades with enhanced efficiency, utilizing a reduced set of parameters compared to traditional blade parametrization methods. Moreover, by establishing a tangible correlation between blade geometry and parametrized loading distribution, the model empowers designers and imparts a comprehensive understanding of the turbomachinery design space and its inherent constraints.

7.3. Recognizing Boundaries

It is imperative to acknowledge that the model's boundaries are intricately entwined with the laws of physics. The sensitivity of database generation to loading parametrization underscores the model's intimate connection to the realm of physics. Furthermore, the quality of the input database emerges as a pivotal factor that profoundly influences the quality of the resultant blades. Thus, ensuring a high-quality database remains a paramount consideration for optimizing results.

7.4. Forging Pathways Forward

Future research endeavors may explore how blade geometry evolves in response to variations in the Reynolds number. Additionally, investigating how blade geometry adapts under diverse loading conditions holds significant promise for advancing this innovative methodology.

7.5. Concluding Remarks

The practicality of this model within the industry is evident, as it serves as an initial design layer for blade generation. By harmoniously intertwining the realms of physics and machine learning through data integration, it paves the way for a transformative approach to designing turbomachinery systems. This efficient tool streamlines the design process, making it accessible to designers from various domains. Its profound significance lies in its ability to reshape the design landscape, ushering in a faster, more intuitive era for turbomachinery system design.

References

[1] Christopher J Clark. A step towards an intelligent aerodynamic design process. In *Turbo Expo: Power for Land, Sea, and Air*, volume 58578, page V02CT41A033. American Society of Mechanical Engineers, 2019.

[2] Jeremy Howard and Sylvain Gugger. *Deep Learning for Coders with fastai and PyTorch*. O'Reilly Media, 2020.

[3] Brenda M Kulfan. Universal parametric geometry representation method. *Journal of aircraft*, 45(1):142–158, 2008.

[4] Mark Drela and Harold Youngren. A user's guide to mises 2.53. *Massachusetts Institute of Technology, Cambridge, MA*, 1998.

[5] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[6] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 2022.