



POLYTECH<sup>®</sup>  
LYON



4TH YEAR INTERNSHIP REPORT PRESENTED BY

**Romain URBANIAK**

MECHANICAL ENGINEER  
YEAR 2024-2025

---

**Data-Driven Model-Order Reduction of  
Optimized Wing Geometries of  
Composite Aircraft**

---



東北大学 流体科学研究所  
Institute of Fluid Science. Tohoku University

*Autor:*

Romain URBANIAK

*Supervisor:*

Professor Abe

## Remerciements

First and foremost, I would like to express my sincere gratitude to Professor Yoshiaki Abe, my internship supervisor at the Department of Mechanical Engineering at Tohoku University, for welcoming me into his team and for his guidance throughout this experience. His expertise, valuable advice, and availability were crucial to the success of this project.

I would also like to thank all the members of the laboratory for their warm welcome, kindness, and daily support. Special thanks to Liu-san for his clear technical explanations and help.

I would like to extend my thanks to Professor Jean-Yves Cavaille, professor at INSA Lyon and head of the joint laboratory between Lyon 1 and Tohoku University, ELYTMAX, without whom this internship would not have been possible, and who provided me with invaluable assistance.

Lastly, I would like to express my heartfelt thanks to my family and loved ones for their constant support and trust throughout my journey.

# Contents

<b>1 Introduction</b>	<b>3</b>
1.1 Context . . . . .	3
1.2 Objectifs . . . . .	4
<b>2 Reduction of Dimension and ROM</b>	<b>5</b>
2.1 Proper Orthogonal Decomposition . . . . .	5
2.1.1 Mathematics basics . . . . .	5
2.1.2 Experimental analysis . . . . .	6
2.2 Self Organizing Maps . . . . .	10
2.2.1 Theoretical approach . . . . .	10
2.2.2 Experimental approach . . . . .	10
2.3 Data-Driven ROM . . . . .	14
2.4 Exploring Further: Towards Inverse Optimization . . . . .	14
2.4.1 Inverse Optimization Framework . . . . .	14
<b>3 POD based ROM</b>	<b>16</b>
3.1 Comparison Setup . . . . .	16
3.2 CFD Trend . . . . .	17
3.2.1 Qualitative analysis . . . . .	17
3.2.2 Quantitative analysis . . . . .	19
3.3 CSM behavior . . . . .	21
3.3.1 Qualitative analysis . . . . .	21
3.4 POD-reconstruction Conclusion . . . . .	22
<b>4 L1 based ROM</b>	<b>23</b>
4.1 Geometrical Comparison . . . . .	23
4.2 CFD behavior . . . . .	24
4.2.1 Qualitative comparison . . . . .	24
4.2.2 Quantitative analysis . . . . .	26
4.3 CSM behavior . . . . .	27
4.4 L1-reconstruction conclusion . . . . .	27
<b>5 General conclusion</b>	<b>28</b>
<b>6 Future perspective of the internship</b>	<b>29</b>
6.1 Context . . . . .	29
6.1.1 Partitioned Method – Block Gauss-Seidel (BGS) . . . . .	29
6.1.2 Fully Partitioned Method . . . . .	29
6.2 Using POD to Estimate Structural Displacement . . . . .	30
<b>Appendices</b>	<b>33</b>
<b>A Python Programs</b>	<b>33</b>
A.1 POD Program . . . . .	33
A.2 SOM Program . . . . .	34
<b>B Geometrical Parameter Error</b>	<b>35</b>
<b>C SOM Heatmaps for Geometrical Parameters</b>	<b>36</b>
C.1 c1 . . . . .	36
C.2 c2 . . . . .	37
C.3 c3 . . . . .	38
C.4 l2 . . . . .	39
C.5 theta . . . . .	40

## List of Abbreviations

AR	Aspect Ratio	$X_{std}$	Standardized deviation dataset
POD	Proper Orthogonal Decomposition	$\alpha_i$	Modal coefficient of mode $i$
CFD	Computational Fluid Dynamics	$U_i$	Mode $i$
CSM	Computational Structural Mechanics	ROM	Reduced Order Model
SVD	Singular Value Decomposition	BMU	Best Matching Unit
SOM	Self-Organizing Maps	Cd	Drag Coefficient
X	Dataset	n	Sample number
$\bar{X}$	Average dataset		

## 1 Introduction

Aviation is one of the most polluting sectors and contributes significantly to global greenhouse gas emissions. In today's environmental context, reducing its impact has become an urgent and essential challenge. To support the development of more efficient and sustainable aircraft, engineers increasingly rely on numerical simulations. These tools enable the exploration of innovative aerodynamic configurations and structural designs without the cost and risk of full-scale physical testing. They also offer a controlled environment for evaluating performance under various flight conditions.

Numerical simulations have become an unreplaceable step in new designs approaches, particularly in fluid mechanics and aerodynamics, where testing new technologies on real-life aircraft is too costly. But one drawback is that these simulations generate massive datasets due to the complexity and the richness of flow fields such as pressure, velocity fields... As environmental constraints increase and performance demands grow, this issue will only become more critical and we need some solutions.

In parallel as said in Okamoto and al [1], Carbon Fiber Reinforced Plastics (CFRP) are increasingly used in primary structures due to their high specific strength and stiffness. However, their anisotropic and heterogeneous nature due to their resin, fiber, and laminate composition makes behavior unpredictable. As a result, CFRP structures demand extensive and costly certification testing, adding to development challenges.

To address these limitations, data-driven strategies are needed to reduce computational cost without losing accuracy. Some methods allow to look into dimensionality reduction approach by extracting dominant features, aerodynamic mode identification, and optimized structure design as explained in Renganathan and al [2].

### 1.1 Context

An efficient approach to explore such wing configurations is by using multi-objectives optimization algorithms. These algorithms iteratively adjust wing design parameters such as geometry, structural layout, or material distribution to minimize or maximize selected objective functions [3]. However, as mentioned those algorithms generate massive datasets due to the large number of design variables (wing dimensions, material characteristics...) and output studied variables field (pressure, velocity...) .

Talking about design variables, our study will focusing on 6 spatial parameters that fully describe the wing geometry as :

- $\theta$  : sweep angle
- $l_1$  : span-wise length
- $l_2$  : span-intermediate length
- $c_1$  : chord-base width
- $c_2$  : chord-intermediate width
- $c_3$  : chord-tip width

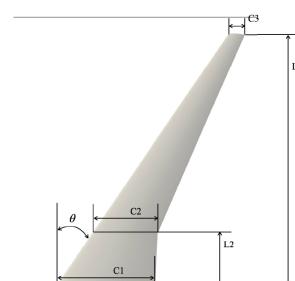


Figure 1: Geometrical features of the studied wing

On this wing configuration 1, an ongoing study conducted by this very laboratory applied two optimization algorithms to explore the trade-off between two aerodynamic objective functions : the drag coefficient ( $C_d$ ) and the structural weight as studied in[4]. The algorithms used were Bayesian Optimization and the Non-dominated Sorting Genetic Algorithm II (NSGA-II) as described in[5, 6].

This process resulted in two distinct optimization datasets, visualized in the following plot :

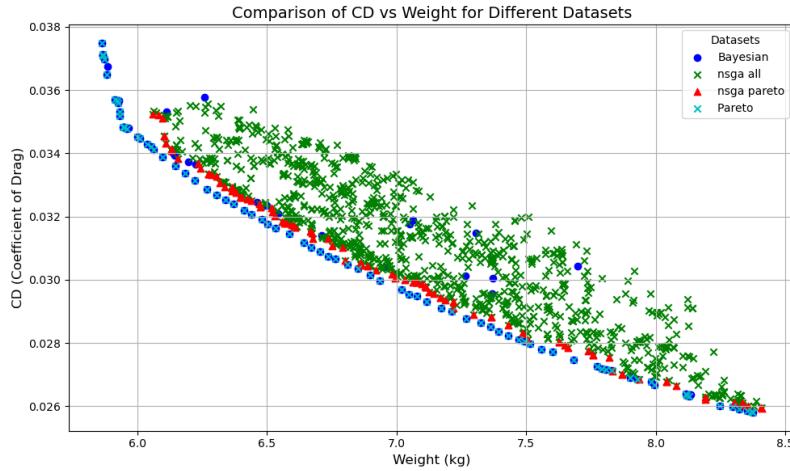


Figure 2:  $C_d$  vs. Weight

Figure 2 shows two optimisation datasets : NSGA-II optimisation dataset in red and green, and Bayesian Optimisation dataset in light and dark blue. On this plot, we can observe two distinct trends (marked by red and light blue crosses), each corresponding to the Pareto front obtained from both optimization methods.

The Pareto front as explained in [7] represents a set of non-dominated solutions, where any improvement in one objective function leads to a loss in the other. These fronts illustrate the trade-off between  $c_d$  and structural weight captured by each algorithm. In this case, the Bayesian Pareto front consists of 76 samples, each representing a distinct wing configuration with its own set of geometric parameters and associated physical features (dataset about 76\*15 datas).

This is precisely where this study intervenes : we aim to reduce the dimensionality of the Bayesian Pareto front using data-driven techniques.

**To what extent can we reconstruct reduced order models that lower computational cost while accurately keeping essential physical behaviors ?**

## 1.2 Objectifs

- Reduce the dimensionality of the dataset using data-driven techniques with the aim of reducing computational complexity.  

$$[\theta, c_1, c_2, c_3, l_2, l_1] \xrightarrow{\text{reduction}} [?]$$
- Develop a ROM (reduced-order model) that captures the essential physical behavior of the wing configurations, allowing faster simulations and further design iterations.
- Validate the reduced model by comparing its aeroelastic behavior with the original high-fidelity dataset.

## 2 Reduction of Dimension and ROM

### 2.1 Proper Orthogonal Decomposition

#### 2.1.1 Mathematics basics

As described in Podvin (2011) [8], Proper Orthogonal Decomposition (POD) is particularly useful in fluid dynamics analysing large and complex datasets, as it allows the identification of dominant mathematical structures called POD Modes. The purpose of the POD is to search for an optimal basis (set) of orthonormal functions  $U$  (POD modes) that capture most of the variance dataset. These modes represent the most significant features of the system's behavior.

Once these modes are identified, the original sample data can be approximated by a linear combination of the modes  $\mathbf{U}$  and their associated modal coefficients  $\alpha$  as it follows :

$$\mathbf{X} = \sum_{i=0}^{n_{\text{modes}}} \alpha_i \mathbf{U}_i$$

The mathematical formulation of finding  $\mathbf{U}$  and  $\alpha$  can be summarized as follows :

Given a dataset  $\mathbf{X} \in R^{m \times n}$ , where  $m$  is the number of samples and  $n$  is the number of features (or spatial locations) :

$$\mathbf{X} = \begin{bmatrix} x_1^1 & \cdots & x_n^1 \\ \vdots & \ddots & \vdots \\ x_1^m & \cdots & x_n^m \end{bmatrix}$$

- Normalize the data:

$$\mathbf{X}_{\text{std}} = \frac{\mathbf{X}_{\text{centered}}}{\sigma}$$

- Compute the covariance matrix :

$$\mathbf{C} = \frac{1}{m} \mathbf{X}_{\text{std}} \mathbf{X}_{\text{std}}^T$$

- Apply Singular Value Decomposition (SVD) :

$$\mathbf{C} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^T$$

where  $\mathbf{U}$  contains the eigenvectors (the POD modes), and  $\mathbf{\Lambda}$  is the diagonal matrix of eigenvalues  $\lambda_i$  indicating the energy associated with each mode.

- Find the temporal coefficients :

$$\boldsymbol{\alpha} = \mathbf{X} \mathbf{U}$$

It is also worth noting that the energy of the POD modes  $E_i$  depends on the percentage of the eigenvalue  $\lambda_i$  relative to the total sum of eigenvalues:

$$E_i = \frac{\lambda_i}{\sum_i \lambda_i}$$

To put in a nutshell, Proper Orthogonal Decomposition (POD) enables a reduced-order representation of complex datasets by decomposing it into a set of orthogonal modes  $U_i$  using SVD that captures the principal directions of variability  $\lambda_i$ .

The accuracy and reliability of this method will be illustrated in next section.

### 2.1.2 Experimental analysis

Now that the POD approach has been introduced, we implement it on python ?? and apply it to our Pareto front dataset (which consists of 76 samples characterized by 6 geometric parameters). The goal is to extract the dominant structures (modes) from this high-dimensional dataset. These modes may potentially allow us to reconstruct an approximate version of the original system using less variables. [9]

$$[\theta, c_1, c_2, c_3, l_2, l_1] \xrightarrow{\text{POD}} [?]$$

This gives us the main "direction" of the dataset based on the variance analysis is the spatial representation of each mode.

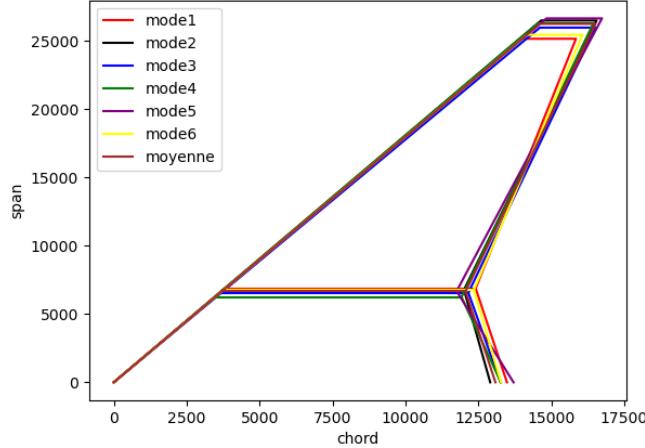


Figure 3: POD mode spatial representation

In our case 3, the POD extracts six different modes, each corresponding to a different planform geometry ( $[\theta, c_1, c_2, c_3, l_2, l_1]$ ) that approximates most of the dataset. Each mode obtained through the POD decomposition represents a distinct spatial pattern that captures dominant geometric variations, but not with perfect accuracy. In fact, achieving an exact reconstruction would require using all 6 modes, which is like using the whole dataset. To better understand the importance of the mode or the amount of information each mode contains across the dataset, we need to plot POD mode's energy.

The following graph 4 illustrates the Cumulative Contribution Ratio the proportion of total variance  $\lambda_i$  captured by the modes.

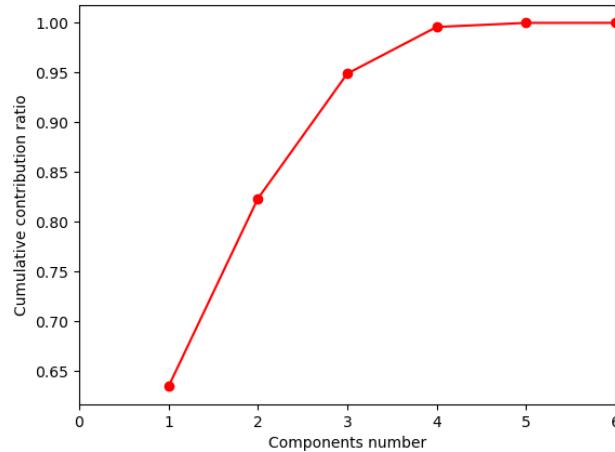


Figure 4: Cumulative components ratio of POD modes

It is observed that that the first mode alone capture 65 % of the dataset energy (variance). When Mode 2 is added, the recovery reaches 83 % accuracy meaning that 83 % of the data is in the "direction" of mode 1+2. Usually, we take the amount of mode needed to get a 95% accurate recovery - [8] which would be requiring 3 modes in this case. By looking at this graph, we can assume that reducing the dataset's dimension from 6 (geometrical parameters) to 3(POD modes) would preserve 95 % of the dataset's information. We still need to verify this assumption.

In order to assess the efficiency of the POD reconstruction, we evaluate several mode combinations such as mode 1+2, mode 2+3 to reconstruct the dataset :

$$X_{POD, mode1+2} = \alpha_1 U_1 + \alpha_2 U_2, \quad X_{POD, mode2+3} = \alpha_2 U_2 + \alpha_3 U_3 \text{ and so on...}$$

The purpose is to determine if the reconstructed geometries remain accurate representations of the original data and to quantify its accuracy. To do this, we compute the reconstruction error for each sample by comparing the original geometry to the POD modes combinations as follows :

$$\text{geometrical error} = [\theta, c_1, c_2, c_3, l_2, l_1]_{\text{original}} - [\theta, c_1, c_2, c_3, l_2, l_1]_{POD}$$

Then we plot the error for every sample of the dataset :

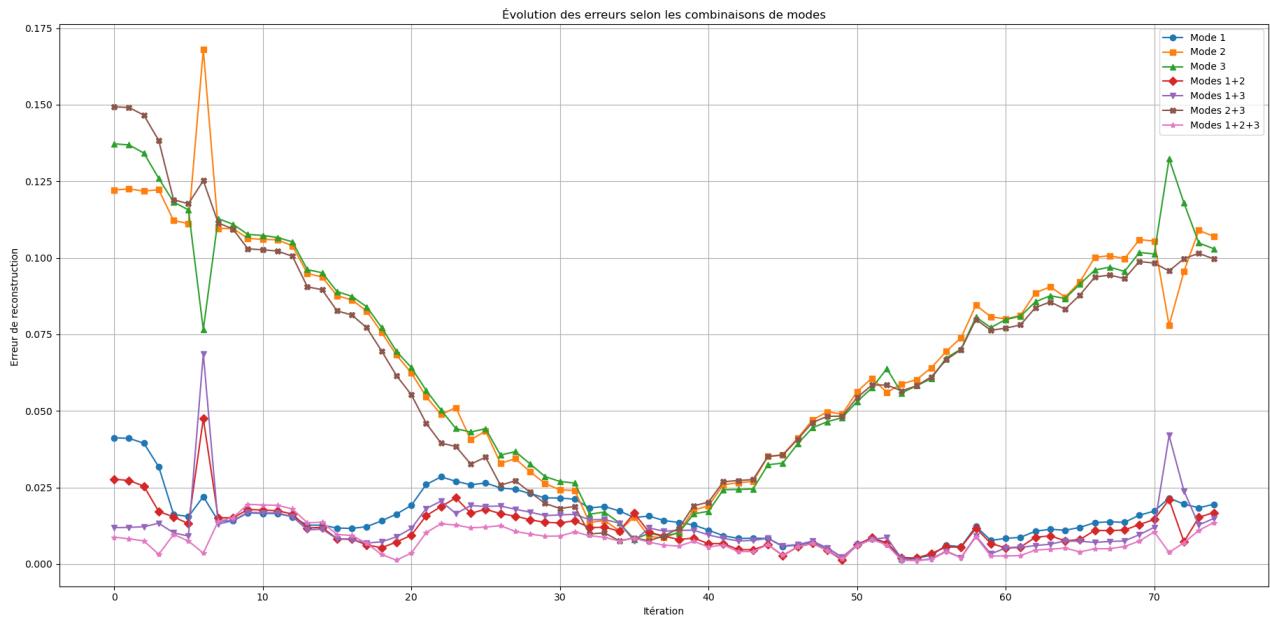


Figure 5: Geometrical error for POD mode configurations

On this plot (5), we can observe several key points. On the one hand, we see that every configuration excluding mode 1 or mode 2 tends to have a higher error at the edges of the dataset (for extreme AR wing cases), while the error approaches 0 for an average AR wing . The geometry accuracy decreases during the first 35 configurations, then increases during the next 35 configurations. We can conclude that every mode combination without mode 1 or 2 is unable to recover accurately the extreme AR cases.

On the other hand, other configurations exhibits a more linear trend. Specifically, for every combination, the average error remains around 2.5%, which is more than accurate.

To the light of this plot, we could consider the following configurations for dimensionality reduction model :

- Mode 1 : 65 % of energy captured and average geometrical error=1.5%
- Mode 1+ 2 : 83 % of energy captured and average geometrical error=1.10%
- Mode 1+ 2 + 3 : 96 % of energy captured and Mode 1+2+3 average error=0.77%

These combinations are good candidates to reconstruct the data because the geometrical errors are very low and the informations is reconstructed with acceptable values. But, we clearly see that adding more modes does not significantly reduce the geometrical error and significantly increase the computational cost (more modes so more data to process). Therefore, it would be valuable to determine the trade-off between computational cost and mode configuration but this is not the subject of the study.

Furthermore, since mode 1 captures 65% of the energy during the POD and recovers the original geometry with a average error of 1.5%, it suggests that most of the dataset is aligned in the direction (geometry) of Mode 1. As for the remaining 35% of the data, covered by the other modes, they are most likely to capture secondary variations in the dataset and maybe are not necessary to process.

From now on, we focus essentially on mode 1 reconstruction aiming to prove that it would be an accurate reconstruction of the dataset.

Let's re-plot the error between geometry reconstructed mode 1 and the original geometry and view some comparison of the reconstructed geometry (6) :

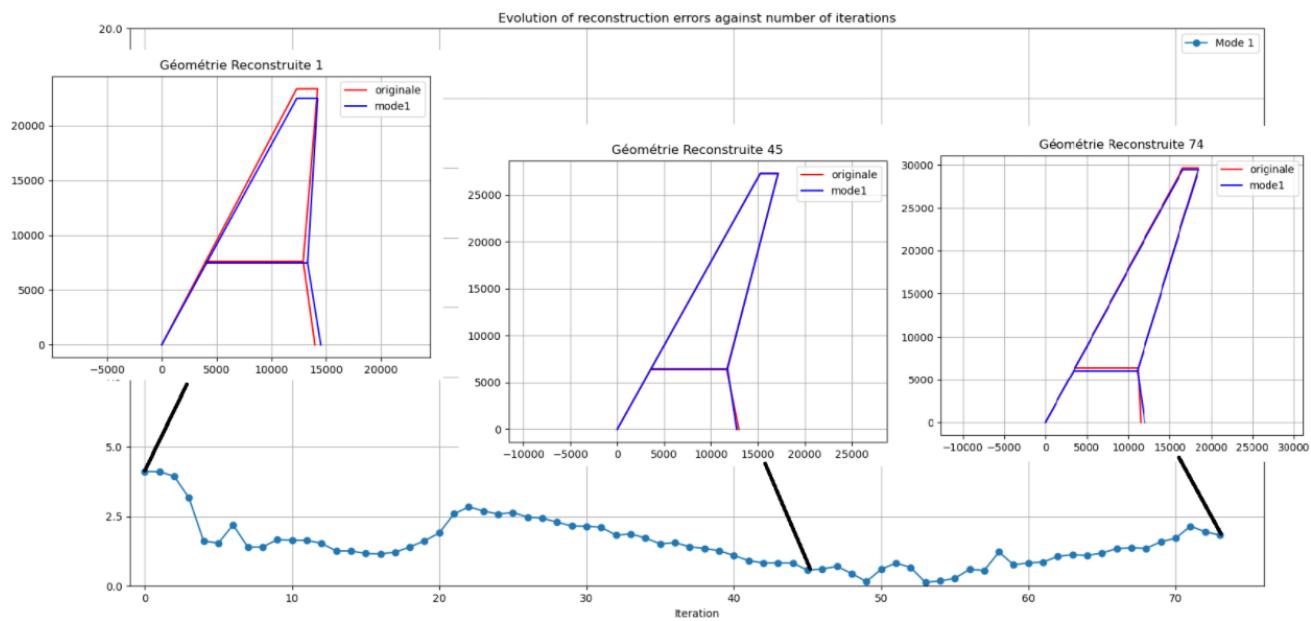


Figure 6: Evolution of reconstruction error with POD mode 1 reconstruction

On this plot 6, we see that the reconstructed geometry looks very similar if not the same which points us in the direction that mode 1 is a very good candidate to reconstruct the data.

After examining the global behavior (geometry), it becomes important to understand how modes contribute to individual parameters. Indeed, since each configuration is defined by six geometrical parameters ( $[\theta, c_1, c_2, c_3, l_2, l_1]$ ), it is reasonable to assume that some of them have a more significant impact than others or even that some mode are influent on specific geometrical features. For instance, Mode 3 could only influence the sweep angle  $\theta$ . This idea is discussed in [10] , where some approaches were used to identify dominant design variables in aerodynamic optimization.

To further investigate this approach, we can analyze the reconstruction error between different for each geometric feature.

We begin with the parameter  $l_1$ , as it exhibits the highest contribution in terms of energy captured across all modes accounting for approximately 70% of the total energy of each mode.

$$\text{l1 error} = l1_{\text{original}} - l1_{\text{POD}}$$

The next figure presents the reconstruction error between various mode combinations and the original geometry, specifically for the geometric feature  $l_1$ :

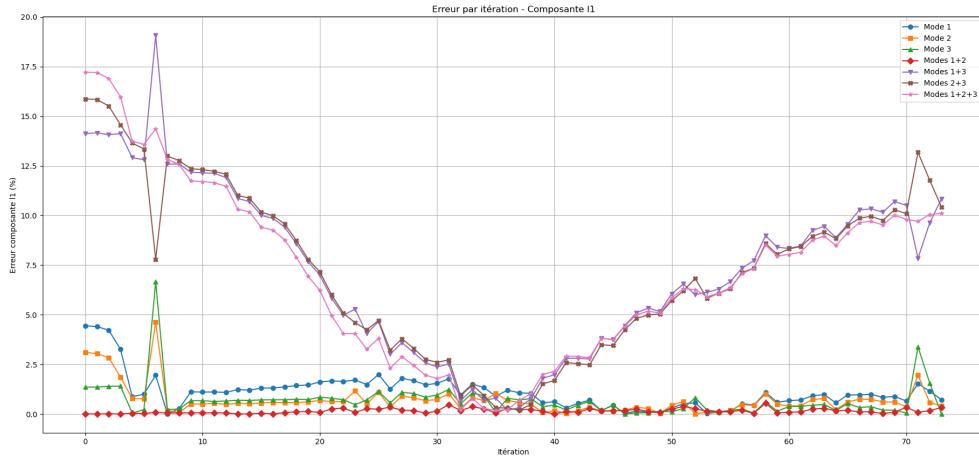


Figure 7: Reconstruction error on geometrical feature  $l_1$

On this plot (7), we observe that mode configurations follows the same trends as for the 'general geometrical error'. Every configuration without mode 1 or mode 2 is first decreasing to near 0 for the first 35 wing configurations, then increasing again. On the contrary, every configuration using mode 1 or mode 2 has an error almost linear for every iterations.

The same configurations stand out : Mode 1, Mode 1 + 2 and Mode 1 + 2 + 3 which is reinforcing our assumption of using only mode 1.

The plot is only for  $l_1$  so we have to see also for the other physical parameters before having conclusion.

In order to avoid 5 other plots, we will summarize every results in the following table and every other plot are available at the end in the appendices 26 :

Geometrical features	Acceptable Configurations	average error(respectively) in %
$\theta$	Mode 1, Mode 1+2, Mode1+2+3	0.85, 0.63, 0.31
c1	Mode 1, Mode 1+2, Mode1+2+3	1.67, 1.2, 0.82
c2	Mode 1, Mode 1+2, Mode1+2+3	1.52, 1.23, 0.76
c3	Mode 1, Mode 1+2, Mode1+2+3	0.15, 0.12, 0.01
l2	Mode 1, Mode 1+2, Mode1+2+3	3.46, 2.92, 2.53
l1	Mode 1, Mode 1+2, Mode1+2+3	1.11, 0.7, 0.15

Table 1: Error between mode configurations and geometrical parameters

This table demonstrates that all geometric features can be reconstructed using the same combinations of modes and among them is mode 1. Another conclusion is that the optimal compromise is to use only Mode 1, as it is sufficient to recover all features with a low average error.

Now that we have an understanding of the global influence each mode has on the geometric features, we can begin to explore whether more localized correlations exist between specific modes and particular parameters. For instance, during the first 30 samples, it is possible that a specific mode might influences a particular parameter more strongly than others.

To explore this further, we use a more local method called Self-Organizing Maps (SOM). Combined with dimensionality reduction techniques like POD, SOM could help us find patterns and correlation in the reduced dataset [11, 12]. This could allow us to better see how specific modes relate to the geometrical parameters.

## 2.2 Self Organizing Maps

### 2.2.1 Theoretical approach

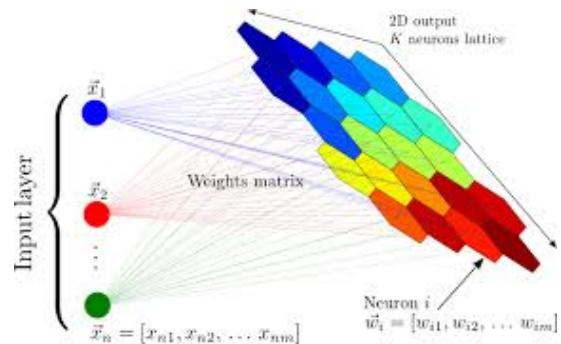
Self-Organizing Maps (SOMs) are an unsupervised learning algorithm used for dimensionality reduction and clustering of high-dimensional data. SOMs work by projecting complex, high-dimensional data onto a 2D grid to highlight the relationships between data.

This makes them an efficient way of organizing data based on similarity. According to Kohonen (2001) [13], SOMs are capable to reveal the hidden structure of the data. This is why they are qualified as local method because they discover correlation within small regions of the dataset.

In the context of our study, we use SOMs to organize and visualize local correlation between design parameters as done in Obayashi and al, 2012 [14]). This helps to identify clusters of wing configuration that have similar physical or performance properties.

This method requires multiple arguments as input such as:

- $\sigma$  : controls neighboring nodes' affection when updating the map
- learning rate : how quickly the map learns from the data
- neighborhood function : how the map adapts to the neighbors



Lets take a closer look into the root of the SOM :

1. Present a data point : One configuration (data point) is shown to the SOM.
2. Find the Best Matching Unit (BMU) : The algorithm looks for the node on the map whose weight vector is closest to the input. This node is called the Best Matching Unit (BMU).
3. Update the BMU and its neighbors : The SOM changes the weight of the BMU to make it more look like the input. It also adjusts the nearby nodes based on  $\sigma$ . The smaller the sigma the fewer neighbors are updated. Also based on the neighborhood function the influence decreases with distance from the BMU.
4. Repeat for all data points : The steps above are done for each data point, and repeated over many cycles (number of sample).
5. Final result : At the end, the SOM gives a 2D map where similar configurations are grouped by.

### 2.2.2 Experimental approach

In our case, SOM will serve two main purpose. Firstly, revealing the correlation between the physical parameters themselves as done in [15]. Secondly, we could use the SOM to discover if there are any local correlation between POD modes and physical parameters.

For our study, we have selected the following settings : the value of sigma is set to 2.0 and the learning rate is set to 0.5, and the neighborhood function follows a Gaussian distribution defined by the equation:

$$\text{neighbor}(x) = e^{-\frac{(x-x^T)^2}{2\sigma^2}}$$

It is important to note that each set of parameters is arbitrary. Indeed, changing SOM parameters will not affect the correlation, only the way it is displayed (color, position, order, size...). The SOM programm in available in 25. We will train the SOM not only to geometrical parameters but also to objective function because we want to discover correlation that could help reduce the dimension.

$$(\theta, c_1, c_2, c_3, l_2, l_1, \text{weight}, Cd)$$

After training the SOM, we obtain this type of plot 8 :

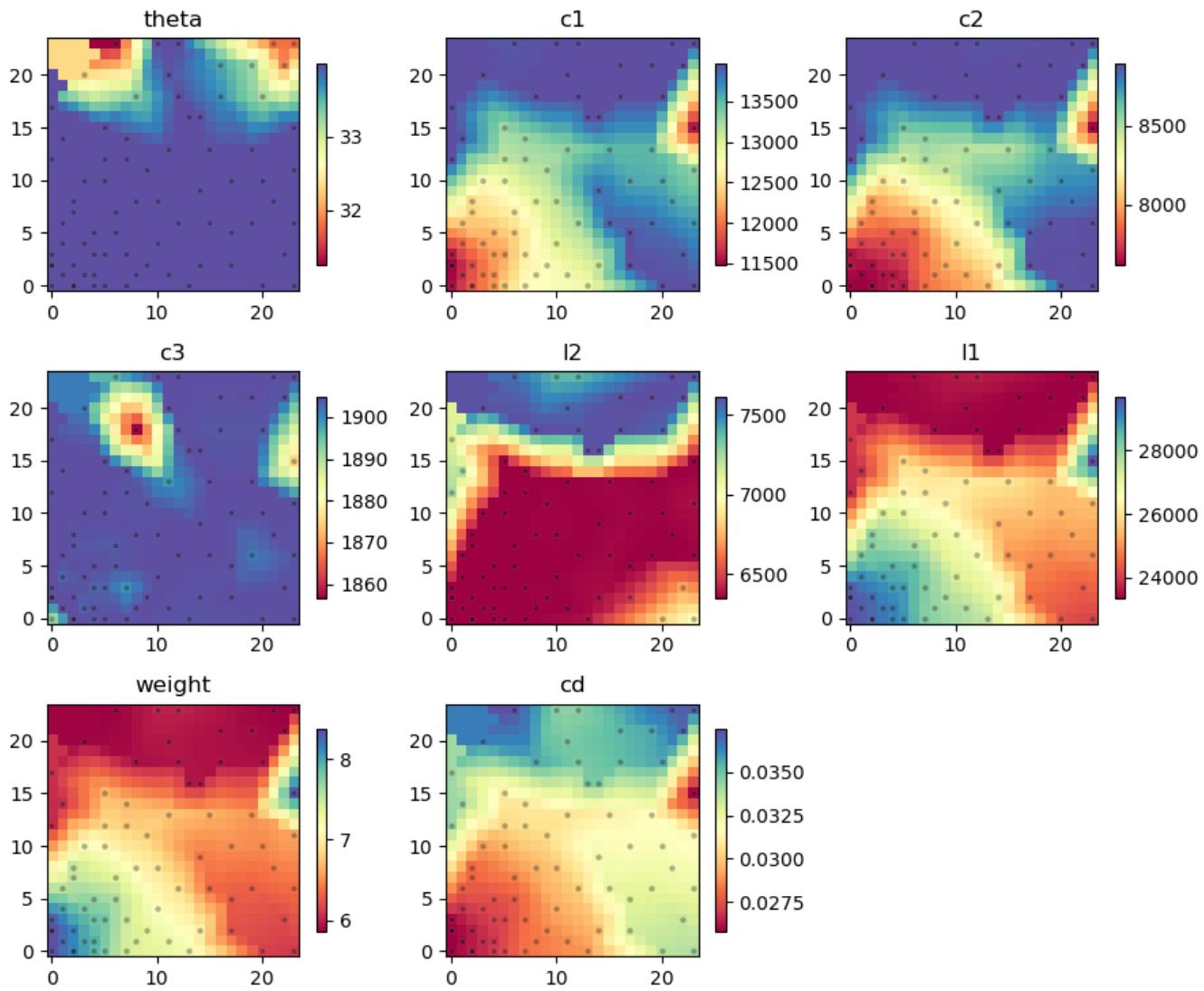


Figure 8: SOM heatmap of geometrical features and objective functions

We observe heatmap color-scaled according to the numerical values for each input. One expected but still interesting remarks is the negative correlation between objective function, which is clearly visible in the heatmap. Additionally, we notice that a longer span-wise wing ( $l_1$ ) tends to result in higher weight and lower drag coefficient, which is physically true and reinforces our confidence in the reliability of this method.

Concerning the shapes of the heatmaps, we can clearly observe strong similarities between the heatmaps of  $l_1$ ,  $c_1$ ,  $c_2$  as well as both objective function heatmap's. This suggests that theses 3 geometrical parameters directly influence the objective functions. This kind of heatmap is very useful because to estimate the dimensions of the wing based on a specific value of objective function. For instance, we want a wing with drag coefficient around 0.035, we know that  $l_1$  will be approximately 24000 cm while changing  $\theta$  and  $c_3$  will have little to no influence.

These observations lead us to an important conclusion : if we want to reduce the computational cost, we could only focus on the parameter  $l_1$ ,  $c_2$ , and  $c_1$  as they are the dominant one to influence the objectives functions.

Now, let's move on to the SOM of the POD coefficients to explore potential correlations between POD coefficients and objectives functions. We train the SOM map based on the POD coefficients as well as  $c_d$  and the structural weight.

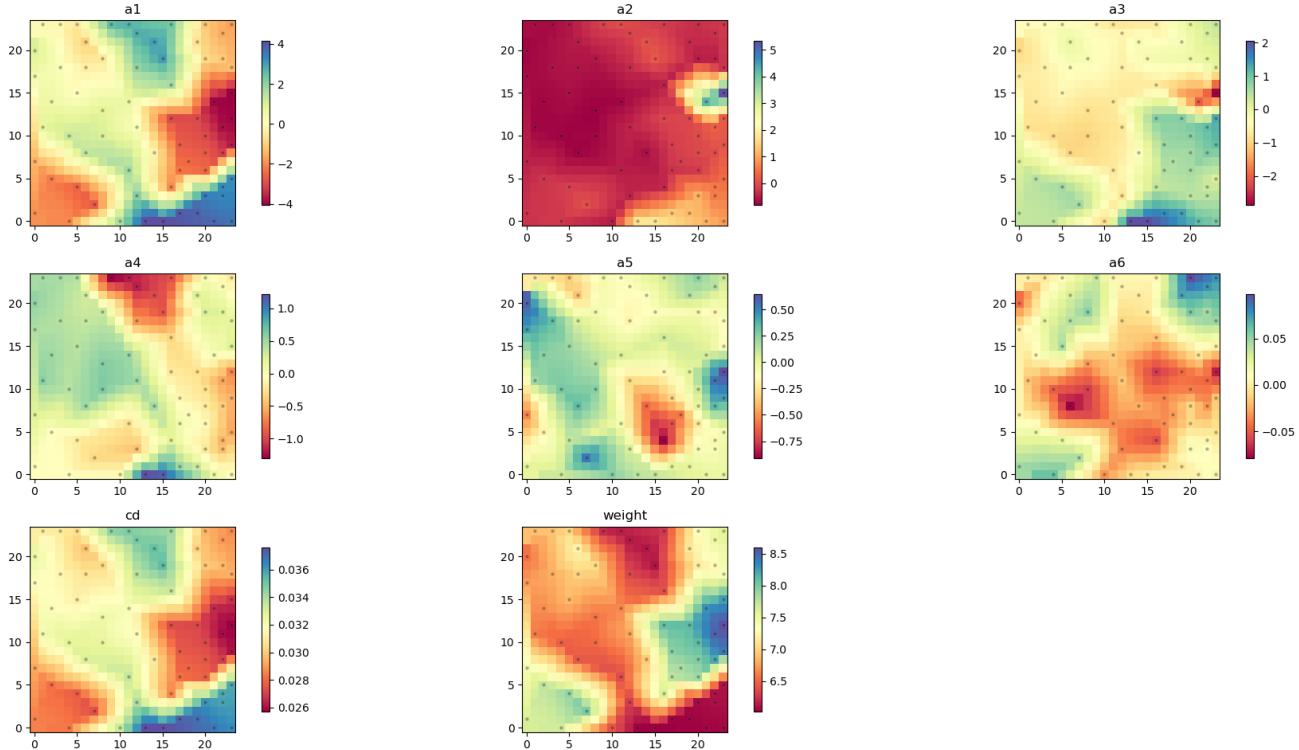


Figure 9: SOM of the POD coefficients and objective function

This heatmap 9 clearly shows the dominant influence of Mode 1 in the POD decomposition. The strongest correlation is observed between the heatmap of  $\alpha_1$  and the objective functions. In fact, the shapes are exactly the same and a positive value of  $\alpha_1$  corresponds to a high aspect ratio (high  $c_d$  and low weight) and conversely. Moreover, the visual similarities between Modes 2 and 3 suggest that these modes may be capturing similar geometric regions or features, indicating that using both of them might be repetitive. This heatmap strongly supports the idea of using only Mode 1 for reconstructing the data, due to its clear correlation with objective function performance.

After examining the global correlation between the objectives functions and the geometrical features as well as how each modes contributes to the objective functions. However, it's reasonable to think as in the previous part, when we try find correlation of some specific mode on individual geometrical features.that some local influence would appear. For instance, certain areas of the SOM heatmaps for a given parameter could be governed by a specific POD mode.

We focus first on the span-wise length  $l_1$ , which seems to be the most influencing parameters of the objective functions.

To better see the boundaries between regions on the heatmaps, we implemented a clustering method. We chose the K-Nearest Neighbors (K-NN) method, and applied it to the BMUs obtained from the SOM analysis. K-NN identifies the k closest data points to each sample based on distance. Then, points sharing similar neighbors ( $\epsilon$  distance) are grouped together forming clusters. The point is to make SOM clusters more obvious on the heatmap.

The following figure presents the SOM heatmaps linking the POD coefficients, the objective functions and  $l_1$  :

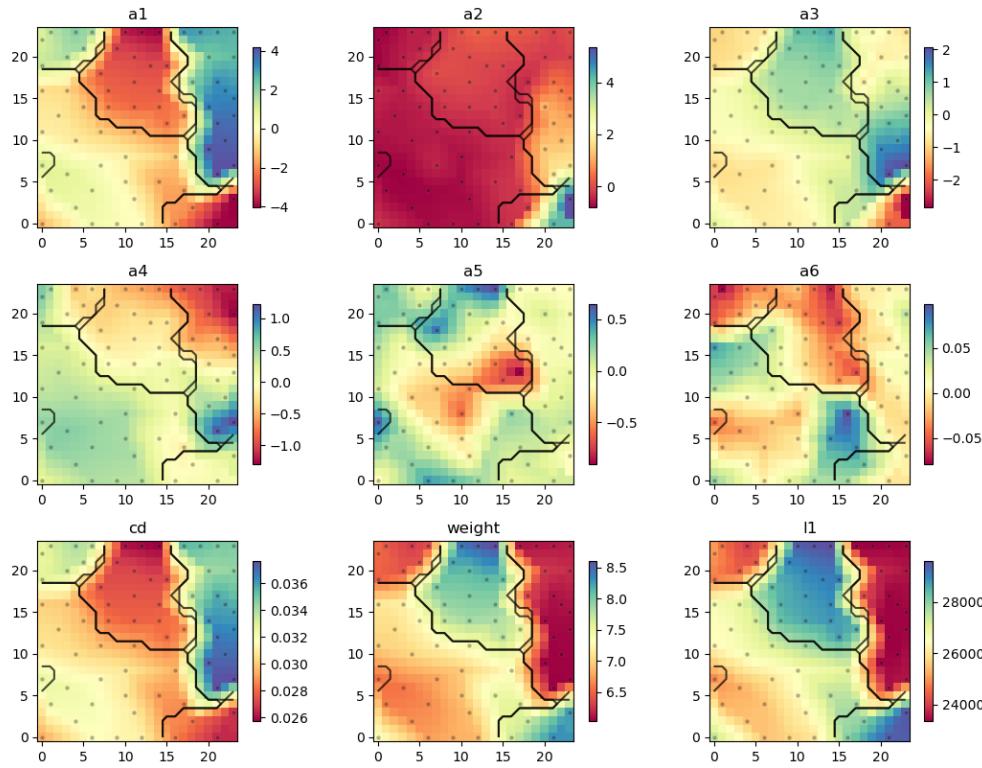


Figure 10: SOM heatmap on geometrical feature  $l_1$

As expected on figure 10, the heatmap of  $l_1$  shows a strong correlation with  $\alpha_1$ . We observe that a negative value of  $\alpha_1$  leads to a longer  $l_1$ , and conversely, a positive  $\alpha_1$  corresponds to a shorter  $l_1$ .

What is particularly interesting is that the heatmaps of the other POD coefficients show no visible correlation with the heatmap of  $l_1$ . This observation reinforces our hypothesis that Mode 1 alone is sufficient to capture the geometric variation and the global approach we made in the POD section.

However, before drawing a definitive conclusion, it is essential to verify the other geometric parameters are also primarily influenced by Mode 1.

To avoid presenting five additional figures, the behavior of each geometric parameter will be summarized in the following table and every plot will be at the end<sup>27</sup>.

Physical parameters	Modes influencing	Correlation
$\theta$	no visual influence	/
$c_1$	Mode 1	+
$c_2$	Mode 1	+
$c_3$	no visual influence	/
$l_2$	Mode 1	+
$l_1$	Mode 1	-
$C_d$	Mode 1	+
Weight	Mode 1	-

Table 2: Influence of the POD modes on the physical parameters

From this table, we can observe several key points. It is also important to note that for  $\theta$  and  $c_3$ , the range of the values in the dataset is very limited, so using a heatmap is less informative compared to the others, as most of the heatmap is uniformly colored, except for 2 or 3 BMUs.

Moreover, most physical parameters are influenced by mode 1 and mode 1 alone. This gives us further confidence in reconstrutring the dataset with mode 1 to reduce the dimension of the dataset.

Additionally, from the row correlations, we can observe how each physical parameter behaves relative to the mode influencing it. Most parameters increase as  $\alpha_1$  increases, except for  $l_1$  which decrease as  $\alpha_1$  increases.

### 2.3 Data-Driven ROM

To conclude this section, we applied two data-driven approaches to reduce the complexity of our Pareto-front dataset. The first method, Proper Orthogonal Decomposition (POD), allowed us to extract the dominant geometrical trends using orthogonal modes. In our case, the first mode and its coefficient  $\alpha_1$  were sufficient to reconstruct the wing geometries with good accuracy. This significantly reduced the dimensionality of the problem.

To go further, we used Self-Organizing Maps (SOMs) to explore local and nonlinear correlations. The SOM confirmed the dominance of  $\alpha_1$  and highlighted four key geometrical parameters:  $l_1$ ,  $c_1$  and  $c_2$ . This insight opens the door to additional simplifications by focusing only on these variables.

In summary, we reduced a 6-dimensional geometric problem to a 1-dimensional one, expressed through the following reduced-order model:

$$X_{\text{ROM}} = \bar{X} + \alpha_1(n)U_1$$

This reduction can also be interpreted as :

$$[\theta, c_1, c_2, c_3, l_2, l_1] \xrightarrow{\text{POD}} [\alpha_1]$$

In other words, given a set of geometrical parameters we can now estimate the POD coefficient  $\alpha_1$  but we could also consider the opposite path known as Inverse Optimisation. It is when we start from a,  $\alpha$  value to reconstruct a geometry.

$$[\alpha_1] \xrightarrow{\text{Inverse}} [\theta, c_1, c_2, c_3, l_2, l_1]$$

This inverse step will be introduced in the next section. The goal is to verify whether the simplified geometry still leads to consistent aerodynamic and structural performance.

### 2.4 Exploring Further: Towards Inverse Optimization

Building on the previous results, we explore the possibility of reconstructing the dataset using only the first POD mode. This approach leads to a simplified optimization strategy, which has been successfully applied in numerous aerodynamic and structural design studies [16, 17].

#### 2.4.1 Inverse Optimization Framework

In this context, inverse optimization becomes not only doable but highly relevant. Our dataset shows that most geometric variations can be captured by a single dominant POD mode,  $\alpha_1$ . This significantly reduces the dimensionality and complexity of the Bayesian optimization process. Instead of running costly simulations for each new geometry, inverse optimization allows us to directly target a desired physical behavior—such as a specific  $\alpha_1$ , drag coefficient  $C_d$ , or structural weight—and receive the corresponding geometry.

This methodology is aligned with the works of Willcox and Peraire [18], who demonstrated the ability of ROM to design optimization processes efficiently. It also resonates with research in reduced-order modeling (ROM), where inverse problems are solved in the reduced modal space to accelerate multi-objectives design workflows [19, 20].

This approach has two main advantages: it reduces computational costs and make automated design (Multi-Objective algorithm) easier. Even using only one or a few modes, the reconstruction is already accurate which makes the inverse approach useful and efficient.

The evolution of  $\alpha_1$  during the optimization iterations is illustrated in Figure 11:

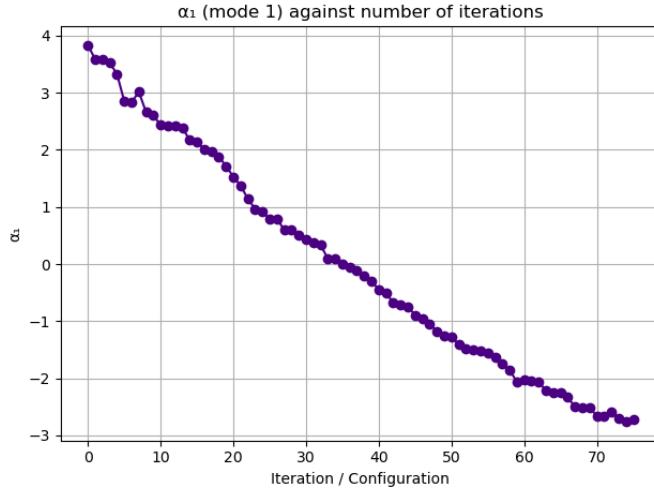


Figure 11: Evolution of  $\alpha_1$  over design iterations.

Each iteration corresponds to a specific wing configuration  $[\theta, c_1, c_2, c_3, l_2, l_1]$ , making a direct relation between  $\alpha_1$  and the wing.

Let  $f$  be the function mapping these parameters to the POD coefficient:

$$\alpha_1 = f(\theta, c_1, c_2, c_3, l_2, l_1)$$

Assuming this function is bijective, we can define its inverse:

$$f^{-1}(\alpha_1) = (\theta, c_1, c_2, c_3, l_2, l_1)$$

This formulation enables two operations: estimating  $\alpha_1$  from a known geometry, or determining a geometry that matches a target  $\alpha_1$ —which forms the basis of inverse optimization.

**Linear Model** Assuming a linear relationship, we model  $\alpha_1$  as:

$$\alpha_1 = a_1\theta + a_2c_1 + a_3c_2 + a_4c_3 + a_5l_2 + a_6l_1$$

The coefficients  $a_1$  through  $a_6$  are computed via least-squares regression. This results in a simple surrogate model capable of predicting  $\alpha_1$  from geometric inputs. Conversely, the same model can be used to solve an optimization problem: minimizing the discrepancy between a target  $\alpha_1$  and its predicted value, thus retrieving the associated geometry.

This linear model could serve as a foundation for more advanced techniques, such as Kriging models, Gaussian processes, or neural networks, which are increasingly used for inverse design tasks in computational mechanics [21, 22].

In summary, the combination of POD and inverse optimization gives an efficient and practical way to design. It allows transitions between geometric parameters and desired physical responses and the other way around, paving the way for faster design space optimisation.

The next step is to check the accuracy of this simplified model using CFD and CSM simulations. This will confirm if the reduced geometry can match aerodynamic and structural behaviors, or if we need to add additional modes for better accuracy.

### 3 POD based ROM

#### 3.1 Comparison Setup

Based on the previous section, we confirmed that mode 1 alone is enough to reconstruct the original dataset with an average geometrical error less than 5%. This aligns with findings from Tan et al. [23] and Bui et al. [24], who showed that POD-based geometry reconstructions with low modal error can describe accurately aerodynamic fields.

Since the geometric fidelity has been validated in the previous section, we now turn our attention to the CFD/CSM aspect. Specifically, the reconstructed POD geometry must reproduce the same CFD/CSM characteristics as the original geometry.

To assess this, we will compare CFD/CSM simulations, focusing on key aerodynamic features such as: Drag coefficient ( $C_d$ ), Mach number and Pressure distribution and key structural features as displacement field and thickness. The importance of conserving these flow features during POD-based reconstruction is also highlighted in the recent review by Manzanero et al. [25].

The main steps of the analysis are as follows :

1. Perform CFD/CSM simulations on both the original and the reconstructed geometries.
2. Visually compare the resulting flow fields extracted from the CFD/CSM simulations.
3. Quantify the error in key CFD/CSM features between the original and reconstructed geometries.

To ensure robustness, this analysis will be conducted across multiple samples, allowing us to sweep through the data range and assess performance across the dataset:

- Low AR configuration : Represents the first configuration in the dataset with a small span-wise length (big  $C_d$  and light weight), with a geometrical reconstruction error of 4.12%.
- High AR configuration : Represents the last configuration in the dataset with a long span-wise length (low  $C_d$  and heavy weight), with a geometrical reconstruction error of 1.94%.
- Average AR configuration : Represents the average configuration in the dataset, with a geometrical reconstruction error of 1.35%. These levels of reconstruction error are consistent with the thresholds under which physical fidelity is preserved in aerodynamic quantities [25, 24].

Before displaying the results, it is important to understand how the simulations are performed in the DASH-tool. The DASH-tool uses aeroelastic analysis to simulate fluid-structure interaction (FSI). This process involves two main models: a wing model for aerodynamic analysis using CFD, and a wing box model for structural analysis using FEM.

The simulation begins with a CFD analysis to compute the aerodynamic loads on the wing. These loads are then transferred to the structural model, where a load factor of 3.75 is applied to simulate a 2.5G maneuver. The FEM analysis calculates the resulting stresses, and a margin of safety (MS) is computed to assess structural integrity. If certain components do not meet the safety criteria, their thicknesses are updated accordingly. This sizing loop is repeated until all structural requirements are satisfied.

After convergence, a second FEM analysis is conducted with a load factor of 1.0 to simulate cruise conditions and determine structural displacements. These displacements are used to deform the aerodynamic model, which is then processed again through a new CFD simulation. This outer aerodynamic loop continues until convergence is achieved as explained in [26].

For further technical details on CFD, force and displacement interpolation, failure analysis, and structural sizing, refer to Date et al.[27] and following image.

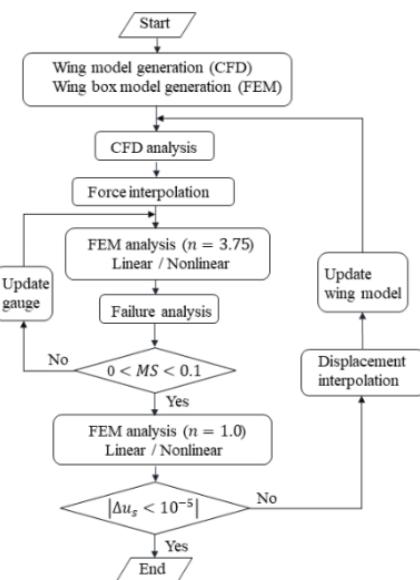


Figure 12: Fluid-Structure Interaction Process

### 3.2 CFD Trend

As a first step, we will perform a visual comparison of these CFD features to make sure that the aerodynamic behavior is preserved. We will display the pressure and Mach number distributions on both upper and lower surfaces of the wing, for both original and POD-reconstructed geometries.

These two features are crucial as they capture both the global physical characteristics of the flow around the wing (Mach number) and the detailed, localized spatial variations of the flow field (pressure).

#### 3.2.1 Qualitative analysis

##### Pressure distribution

Figure below illustrates the pressure fields for the three different samples, shown for both the original and the reconstructed geometries.

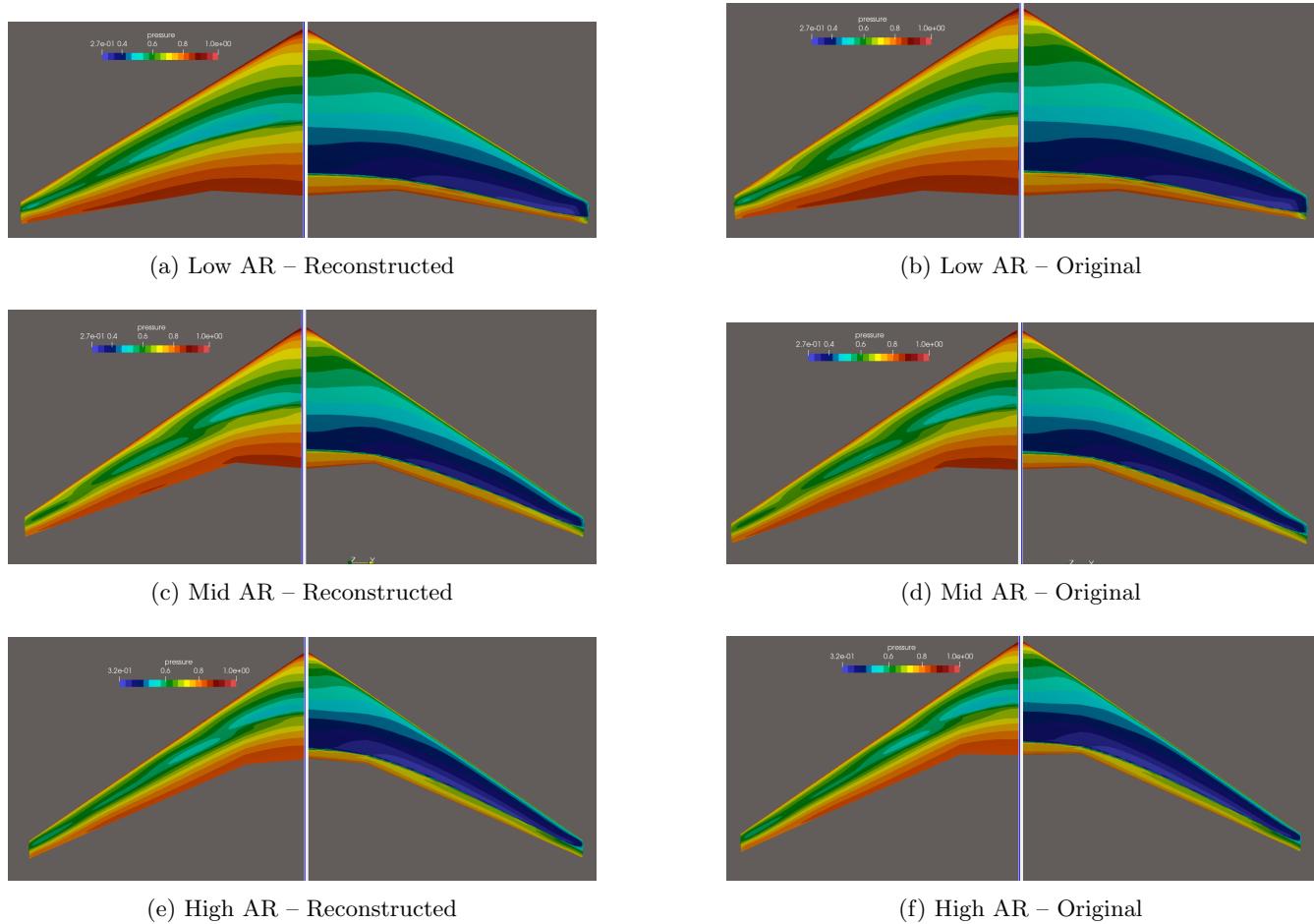


Figure 13: Visual comparison of reconstructed and original pressure fields for low, mid, and high AR configurations

##### Low AR sample

Figures 13a and 13b show the pressure distributions for the reconstructed (POD) and original geometries, respectively. Visually, the pressure field are very similar, indicating that the POD reconstruction accurately captures the aerodynamic behavior. However, compared to the average-span wing, some differences are noticeable. On the upper surface of the average-span wing, two distinct low-pressure regions (approximately 0.5 bar, shown in light blue) are clearly visible. In contrast, the low-span wing exhibits only one concentrated low-pressure area. This suggests that a shorter span may limit the development of multiple separated low-pressure zones, which can contribute to increased drag.

## Average AR sample

This sample corresponds to the average wingspan and had the lowest geometric error among the three (1.35%). Figures 13c and 13d present the pressure distributions for the reconstructed and original geometries. The CFD results are very similar, demonstrating that the POD method effectively captures the main aerodynamic features. Crucial regions such as the leading edge—where pressure varies fastly due to flow acceleration—are well represented. Furthermore, the pressure field near the trailing edge on the upper surface, which contributes to lift force, is also well represented.

## High AR sample

Figures 13e and 13f illustrate the pressure fields for the high aspect ratio configuration, reconstructed and original. As with the previous samples, the pressure distributions appear very similar. However, a comparison between samples reveals notable differences. In the high AR case, the two low-pressure zones (in light blue) on the upper surface are located closer to the wing root, whereas in the average-span sample they were more centered spanwise. This shift may be attributed to the increased span, which affects the distribution of aerodynamic loads along the wing.

After analyzing the pressure distributions, we now turn our attention to the Mach number fields to assess how well the POD method captures compressibility effects and shock-related phenomena.

## Mach number distribution

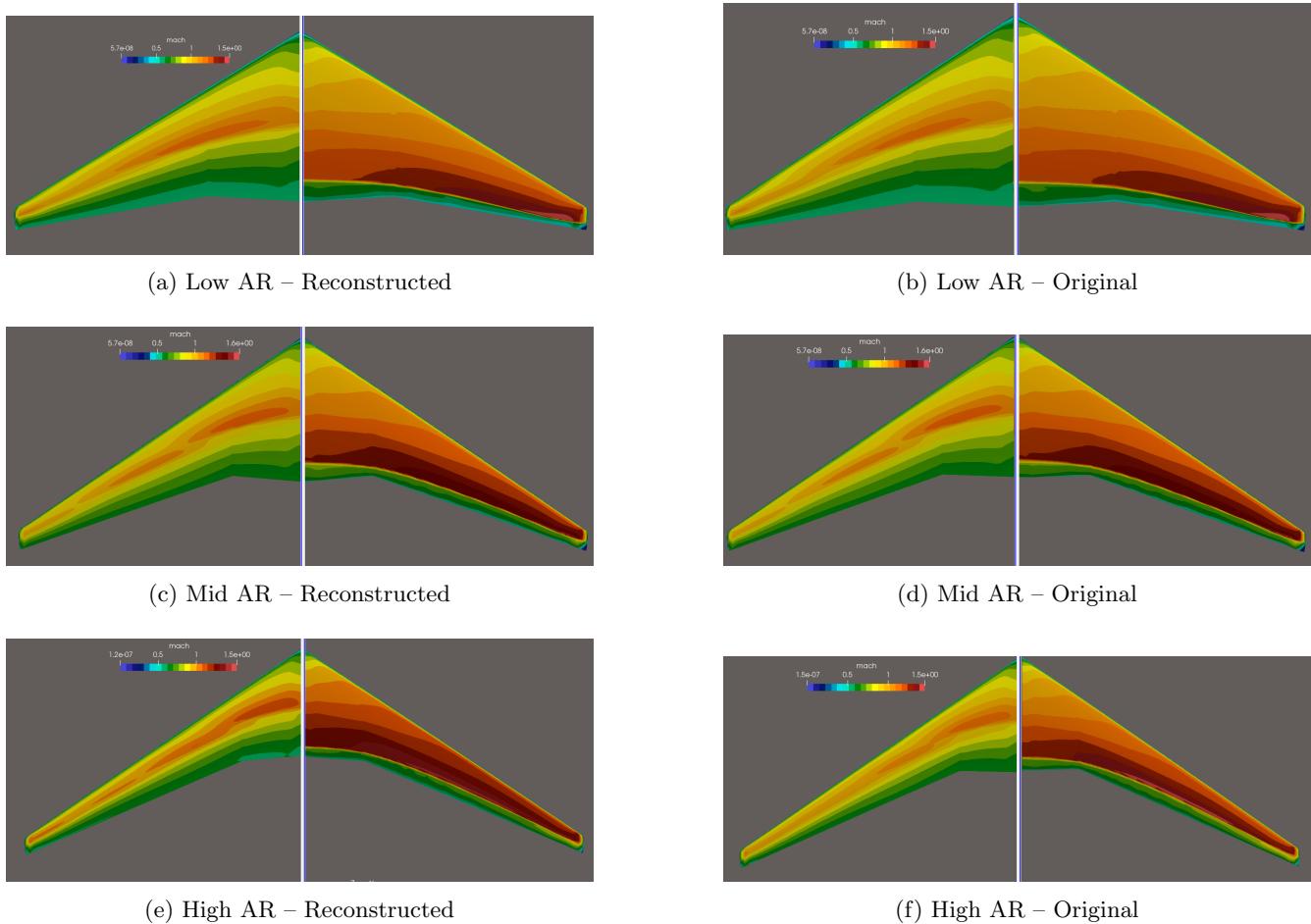


Figure 14: Visual comparison of reconstructed and original Mach number fields for low, mid, and high AR configurations

## Low AR sample

Figures 14a and 14b show the Mach number distribution on the POD-reconstructed and original geometries, respectively. The flow fields are in strong agreement, with the POD reconstruction successfully capturing the main aerodynamic features. In particular, the shock wave on the upper surface—characterized by a sharp increase in Mach number—is well reproduced. This demonstrates the ability of the POD method to retain critical flow phenomena such as shocks, which are essential for accurate aerodynamic analysis in transonic or supersonic regimes.

## Average AR sample

Figures 14c and 14d display the Mach number distributions for the reconstructed and original geometries. The visual similarity confirms the consistency of the POD approach, supporting conclusions similar to those drawn for the average-span pressure fields. Despite that, subtle differences can be found. In particular, the two local regions of higher Mach number identified in the average-span case appear to merge into a single, broader region in the low-span configuration, remembering the behavior previously observed in the pressure fields.

Also, in the low AR configuration, the shockwave moves in the direction of the wing tip, but in the average AR geometry, it stays closer to the wing's center. This shows that the spanwise length affects the shockwave as well as the flow fields.

## High AR sample

Figures 14e and 14f present the Mach number distributions for the POD-reconstructed and original geometries for high AR sample. As in previous cases, the reconstruction closely matches the original flow field, faithfully replicating key aerodynamic features such as shock waves and regions of rapid flow acceleration.

In summary, the analysis of the three samples demonstrates accurate reconstruction between the original geometries and the POD-reconstructed ones. Key aerodynamic features, such as the Mach number distribution and the location of critical zones, are well preserved. However, minor differences remain, reflecting the inherent variability between samples and highlighting the need for thorough validation in each specific case. More importantly, this visual comparison alone is insufficient to make definitive conclusions about this ROM. A quantitative analysis is necessary.

### 3.2.2 Quantitative analysis

Having established an understanding of both the global and local airflow behavior around the wing, we can now assess the aerodynamic performance more rigorously by plotting the drag coefficient on a specific wing cross-section and comparing the results for both cases. This approach will allow us to quantify the reconstruction error and provide a solid first quantitative evaluation of the method's accuracy.

We developed a Python script executed within the ParaView Python console to automate the application of each filter Annexe.

Let us break down the steps performed by this script:

- Load and extract the dataset.
- Determine the maximum and minimum values of the  $x$ -coordinate.
- Apply slice filters at three key cross-sections of the wing (root chord, intermediate chord, tip).
- Use the calculator filter to compute the normalized chordwise coordinate  $x/c$ .
- Separate the wing surfaces into extrados (upper surface) and intrados (lower surface).
- Visualize the results in ParaView.
- Plot the pressure coefficient  $C_p$  versus  $x/c$  for both lower and upper surfaces of the original and POD-reconstructed geometries.

Using this Python script, we obtain the following plots:

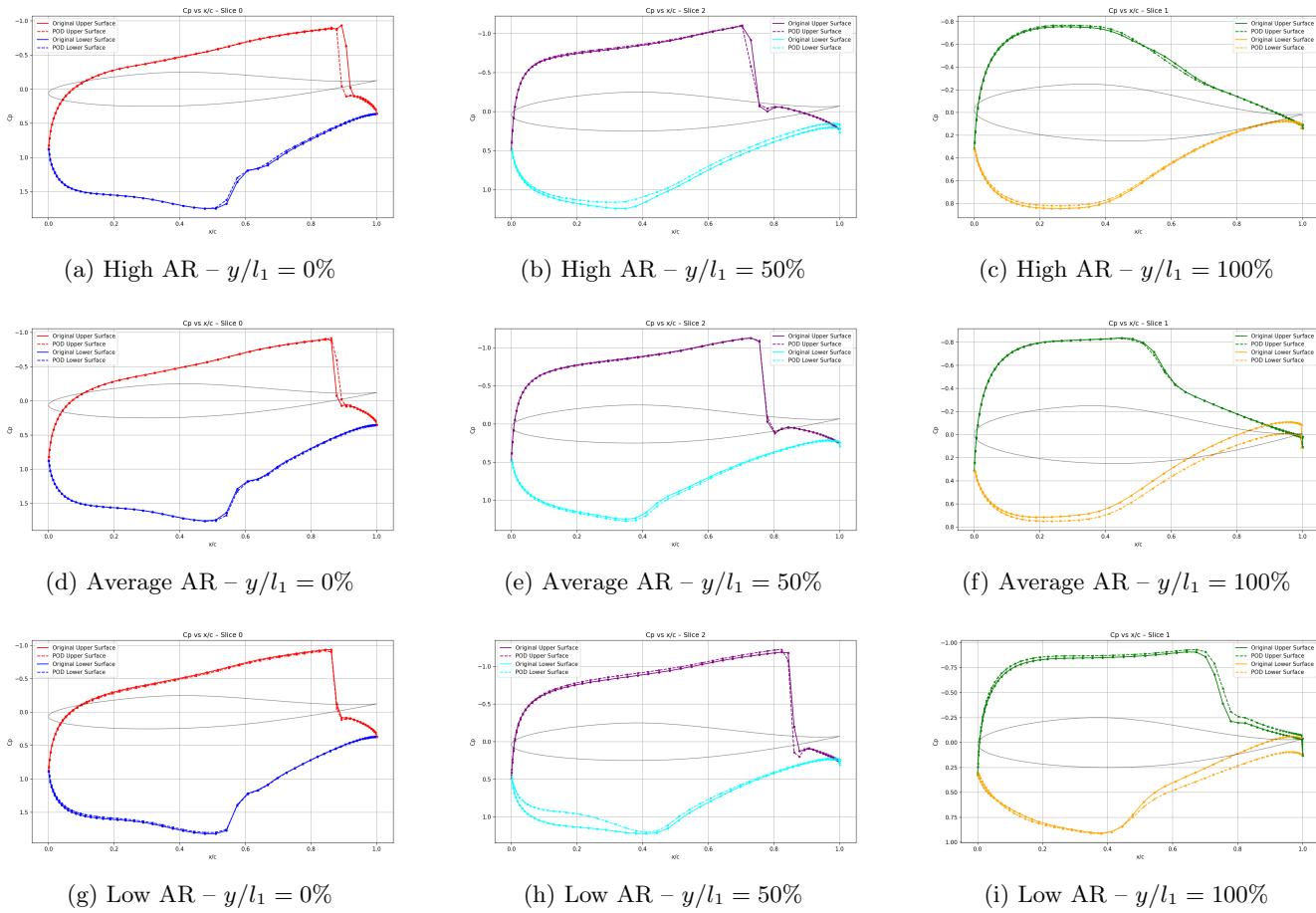


Figure 15: Comparison of  $C_p$  curves across three AR configurations and three spanwise positions ( $y/l_1 = 0\%, 50\%, 100\%$ )

Examining these graphs (15) allows us to quantitatively assess the POD reconstruction. It is clear that, for all three samples, the POD reconstruction captures the aerodynamic behavior very accurately.

Indeed, we obtain very similar  $C_p$  distributions, with only minor issues in some localized area. This discrepancy likely arises from our method of separating extrados and intrados values, which relies on a linear separation using splines that may not be precise enough. Aside from this, the global trends are well preserved by the POD.

Next step is to quantify the error between the POD reconstruction and the original data, even though the visual similarity is already very strong.

The error values are presented in the following table:

Table 3: Cp Error for High AR configuration

Slice Index	Y Orig	Y POD	Extrados Error (%)	Intrados Error (%)
0	0.00	0.00	1.52	1.44
1	2.55	2.42	2.68	2.9
2	1.30	1.30	6.25	6.05

Tables 3, 4, and 5 present the relative  $c_p$  errors between the original and reconstructed geometries using POD for high, low, and average aspect ratio (AR) configurations, respectively. For each configuration, the  $c_p$  error is evaluated at three spanwise slices on both the extrados and intrados.

Across all three cases, the reconstruction errors remain relatively low, generally below 10%. Notable exceptions include the intrados error on slice 1 for both the low AR (14.08%) and average AR (13.6%) cases, which represent localized discrepancies. Despite some high values, the results confirm the capacity of the POD reconstruction using

Table 4: Cp Error for Low AR configuration

Slice Index	Y Orig	Y POD	Extrados Error (%)	Intrados Error (%)
0	0.00	0.00	3.02	1.16
1	1.67	1.49	6.51	14.08
2	1.30	1.30	5.87	9.79

Table 5: Cp Error for Average AR configuration

Slice Index	Y Orig	Y POD	Extrados Error (%)	Intrados Error (%)
0	0.00	0.00	1.03	0.95
1	1.67	1.74	1.69	13.6
2	1.30	1.30	1.31	2.49

only one single mode. The reconstructed pressure and Mach number distributions matches well with the original data, suggesting that the method stays sufficient accurate enough for aerodynamic analysis while greatlyreducing the amount of data needed.

However, these results only come from the aerodynamic (CFD) behavior. To fully check if the reconstruction is accurate, we also need to consider the structural behavior which is as important because the CFD results are based on the structural analysis.

### 3.3 CSM behavior

We have access of a lot of different structural features such as the displacement or also the thickness which comes from structural design. As the dispacement is symetrical on both surface of the wing we will just show the upper surface.

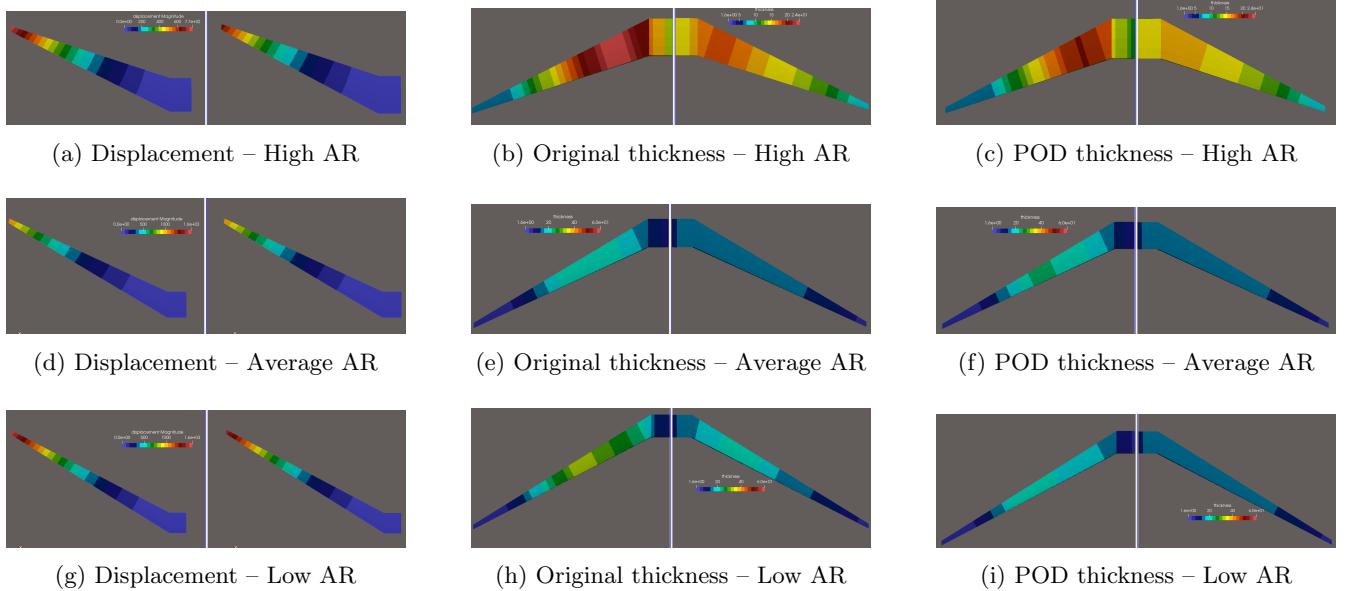


Figure 16: Comparison of displacement and thickness fields between original and POD-reconstructed configurations for different aspect ratios.

#### 3.3.1 Qualitative analysis

We now compare the structural behavior—specifically the displacement field and thickness distribution—between the original and POD-reconstructed configurations for different AR.

In Figure 16a, we observe a difference in displacement magnitude at the wingtip for high AR configuration, this is due to the length difference of approximately 20 cm between the original and the reconstruction. This might indicates a limitation of the POD reconstruction in preserving wingspan displacement.

Figures 16b and 16c show that the thickness distribution differs significantly in the midspan region, especially on the upper surface where a difference of up to 10 mm is visible (green vs. light blue).

For the average AR wing, the displacement fields shown in Figure 16d seems similar, except near the wingtip where the original wing shows more bending due to its longer span. However, as seen in Figures 16e and 16f, there are thickness differences of up to 8 mm, especially in the midspan region of both surfaces.

In contrast, Figure 16g for the low AR configuration shows an almost perfect match in displacement fields, indicating that the POD performs well when span lengths are similar. However, as seen in Figures 16h and 16i, while the lower surface thickness matches closely, the upper surface still displays a difference of about 5 mm in the mid-chord area.

These comparisons show that while the POD method does a good job capturing the overall shape and displacement of the wing, especially when the wings have similar spans (low AR sample), but it does not describe structural details very well. In particular, it tends to miss local variations in thickness near the middle of the wing, which are very important for structural performance.

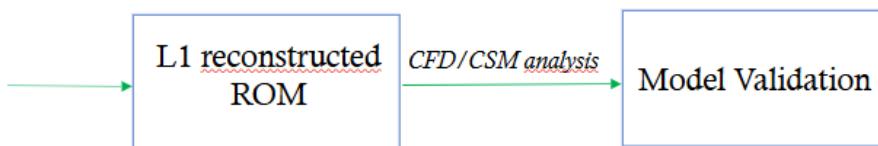
### 3.4 POD-reconstruction Conclusion

In this section, we compared the CFD and CSM results between the original and the POD-reconstructed geometries. On the CFD side, the POD reconstruction worked well: it was able to reproduce the pressure and Mach number distributions with good accuracy. Visual comparisons showed that the main aerodynamic features were preserved, and the drag coefficients were also very close, confirming that the POD method captures the global aerodynamic behavior.

However, for the structural part, the results were not accurate enough. We noticed multiple differences, due to variations of the geometry between the POD's and the original shapes. This shows that POD struggles with more locally effects, exactly like thickness variation or tip displacement, and is very sensitive to geometric differences.

To sum up, using just one POD mode gives good representation for CFD, since aerodynamic behavior dependance is global. However, as the structural behavior is non global (Saint-Venant's principle), this simplified reconstruction isn't accurate enough. It struggles to capture local variations and reacts strongly to small changes in geometry. A better solution would be to include 2 more POD modes for the CSM model to better reflect these local effects and get structural behavior closer to the original. Even if the reconstruction isn't perfect, it still offers a good trade-off for such a simple reduced-order model. Another option would be to explore a different reduced model based on the SOM analysis.

We also looked at how each geometric parameter could affects aerodynamic performance. The SOMs showed a strong correlation between the spanwise length  $l_1$  and objective functions. This suggests and was expected, that  $l_1$  plays a dominant role in the aeroelastic behavior. Based on this, we decided to implemented a other ROM based only on  $l_1$ . Indeed, while keeping the other parameters fixed only  $l_1$  will be modified from a configuration to another. The goal is to see if this simpler approach could still reconstruct the trends from the original Pareto front. This brings us to the next step : the  $l_1$ -based reconstruction method.



## 4 L1 based ROM

Based on the results of SOM analysis in section 2.2.2 , the parameter  $l_1$  was identified as the most influential geometric variable contributing to Mode 1 in the POD decomposition. This conclusion is reinforced by the strong spatial correlation observed between the heatmaps of  $l_1$  and the corresponding POD coefficient  $\alpha_1$ , consistent with how SOM can reveal trade-offs between design variables and aerodynamic objectives [28].

To further explore this relationship, we propose a simplified reconstruction approach in which all geometric parameters except  $l_1$  are held constant. The reference point will be the average AR configuration. Specifically, the values of  $\theta$ ,  $c_1$ ,  $c_2$ ,  $c_3$ , and  $l_2$  are fixed according to this geometry, while  $l_1$  is going to be changing based on the sample.

The goal is to assess if modifying  $l_1$  is enough to reproduce aeroelastic behavior original dataset. If accurate enough, this reduced approach would allow for a significant simplification of the optimization approach (more than the POD), as it would reduce the dimensionality of the design space to only 1 parameter by configuration.

As in previous analyses, the evaluation will be conducted on representative configurations: the first (low AR) and the last (high AR) from the Pareto front dataset.

### 4.1 Geometrical Comparison

We remember that our first objective is to reduce the dimensionality of the Pareto front dataset without losing major information. To assess this, we examine the geometric reconstruction error across all samples, similarly to what was done in 5 but this time by fixing all parameters except  $l_1$ .

In the next figure , we show the error between the original geometries from the Pareto dataset and the reconstructed geometries obtained by varying only  $l_1$  while keeping the other geometric parameters constant for every sample of the pareto front dataset :

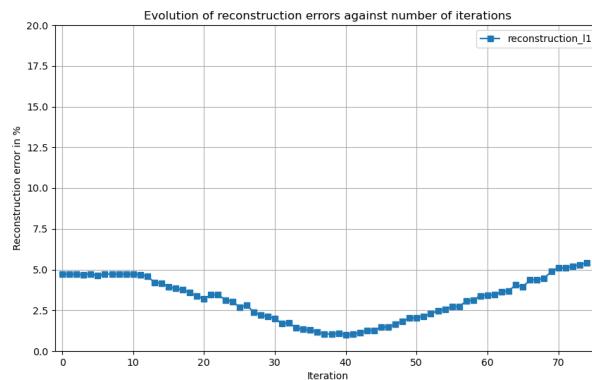


Figure 17: Evolution of the reconstruction error against iterations

Thanks to this plot 17, we can confirm that the l1 reconstructed successfully reconstructs the geometry across the entire dataset. The reconstruction error remains below 5% for all samples, with an average error of only 3.15%, which is well within an acceptable range.

Having validated the geometric fidelity of the reconstruction, the next step is to verify whether the aerodynamic (CFD) and structural (CSM) behaviors are also accurately preserved. To this end, we will perform both CFD and CSM simulations on the original geometries and those reconstructed by varying only  $l_1$ . The resulting data will then be compared, following the same methodology previously applied in the POD-based analysis.

In order to better grasp the geometry difference between the  $l_1$  reconstructed geometry and the original geometry we show both low AR configuration and 75 original geometrical features and the corresponding :

geometrical feature	original High AR	reconstructed High AR	original low AR	reconstructed low AR
$\theta$	34	33.99	31.2412	33.99
$c1$	11447.1	13310.2	13928.8	13310.2
$c2$	7684.35	8474.59	8889.67	8474.59
$c3$	1904.9	1898.74	1896.13	1898.74
$l2$	6353.16	6351.14	7552.31	6351.14
$l1$	29538	29538	23456.4	23456.4

Table 6: geometrical configuration of studied samples

This table informs us that both reconstructed geometries have the same geometrical features, except for the span-wise length. All other features match the average AR sample (sample 38 in the dataset).

We will proceed in the same manner as previously, starting with the CFD trend. This will include a visual comparison, followed by a quantitative analysis using the Cp plot. Finally, we will explore the CSM behavior.

## 4.2 CFD behavior

### 4.2.1 Qualitative comparison

#### Pressure distribution

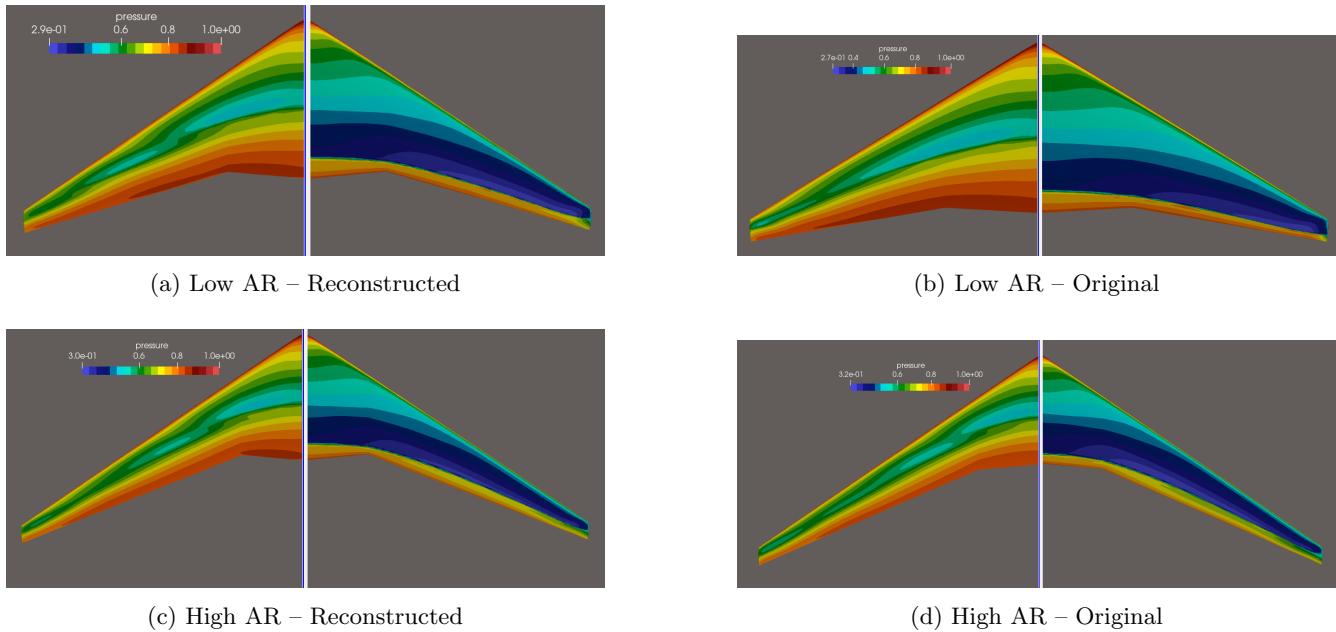


Figure 18: Visual comparison of reconstructed and original pressure fields for low and high AR configurations

Figures 18 (a) and (b) show that for the low aspect ratio (AR) configuration, the pressure distribution is generally similar between the original and reconstructed geometries. However, differences are noticeable on the upper surface, particularly near the trailing edge and in the central part, where the original geometry features a large low-pressure zone, while the reconstructed geometry has two smaller ones.

Similarly, in Figures 18 (c) and (d) for the high AR configuration, the pressure fields remain visually similar, with only a slight difference in the middle of the upper surface.

## Mach number distribution

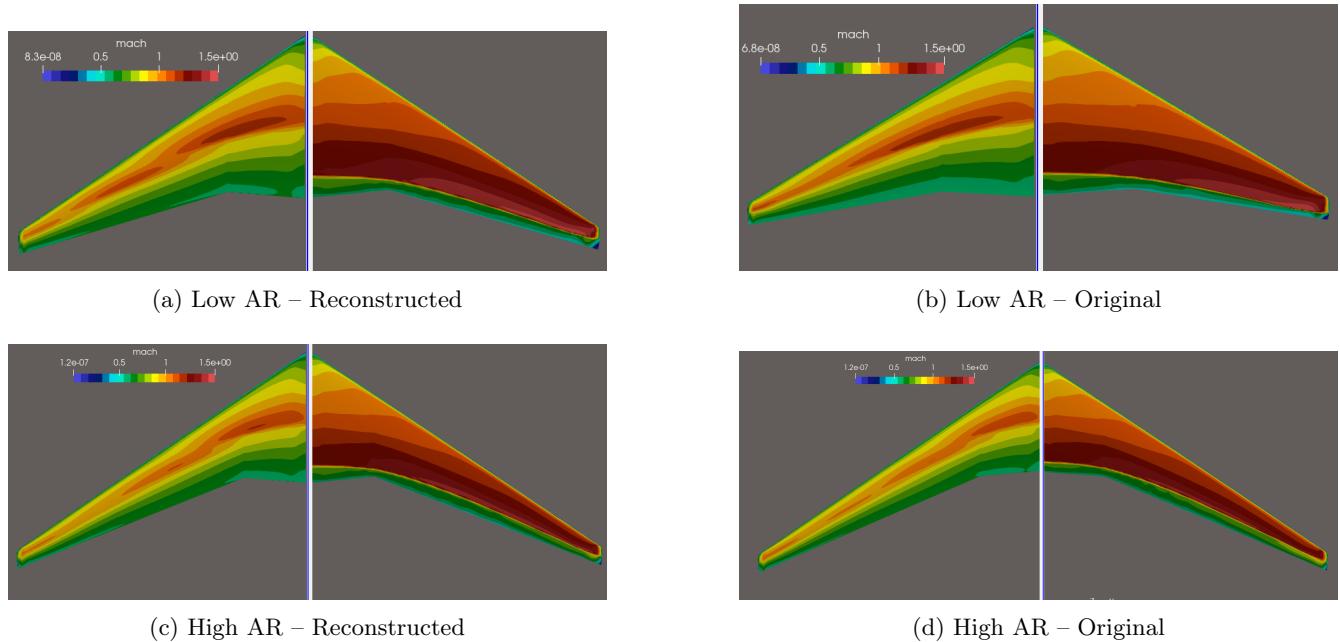


Figure 19: Visual comparison of reconstructed and original Mach number fields for low and high AR configurations

The plots presented in Figures 19 (a) and (b) illustrate the Mach number distribution for the low aspect ratio (AR) configuration. The overall distribution is quite similar, with notable discrepancies on the upper surface—especially near the trailing edge and in the central region. These discrepancies correspond closely to the differences observed in the pressure distribution.

In Figures 19 (c) and (d), which show the high AR case, the Mach number looks similar to the low AR case. There are small differences in the middle of the upper surface, like in the POD, but overall the Mach field is almost the same between the original and reconstructed shapes.

We clearly observe that both the POD-based reconstruction and the simpler  $l_1$ -based approach provide similar results regarding the aerodynamic behavior, including the Mach number distribution and the pressure field. Both methods successfully capture the main flow features, indicating that the spanwise length  $l_1$  alone plays a key role in driving the aerodynamic response. This confirms the strong correlation previously identified between  $l_1$ ,  $\alpha_1$ , and the global flow behavior.

After visually comparing the main aerodynamic features from the CFD results, we now move on to a more quantitative analysis. Specifically, we compare the pressure coefficient ( $C_p$ ) profiles at three key sections of the wing. This comparison is carried out for both the original and the reconstructed geometries, in order to assess how well each method preserves local flow characteristics.

#### 4.2.2 Quantitative analysis

##### Cp plot

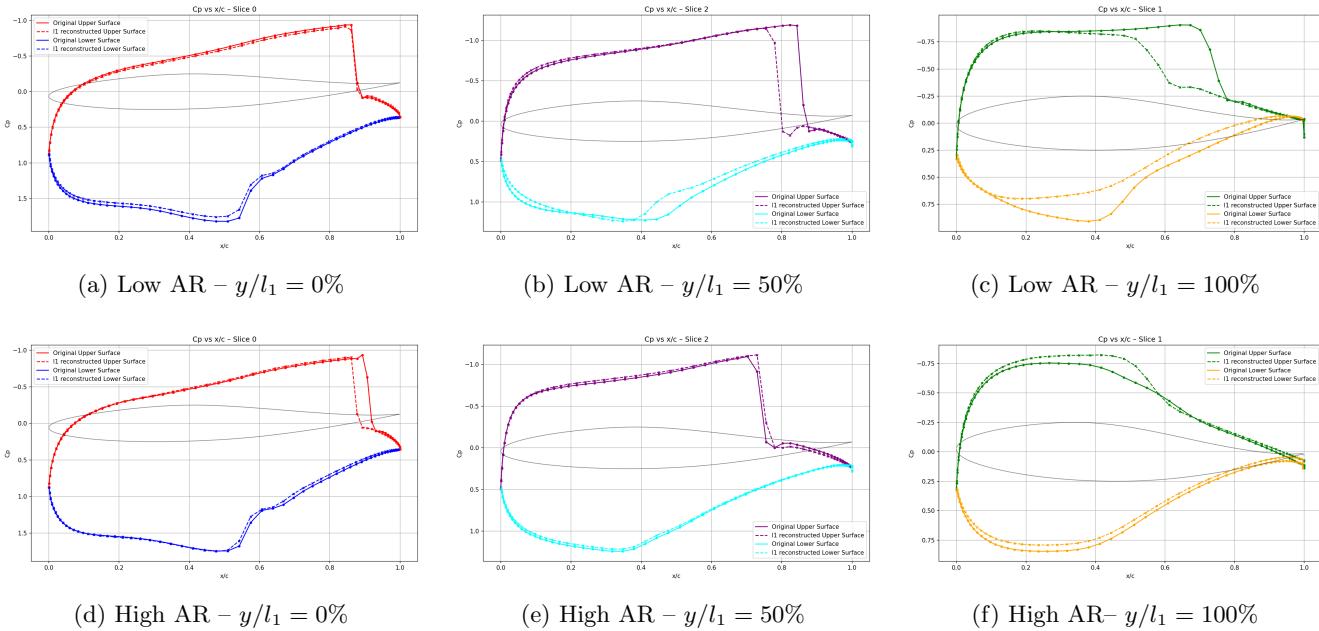


Figure 20: Comparison of  $C_p$  curves for different slices normalized by  $x/c$  for low/high AR configurations

The reconstruction results show some noticeable discrepancies, particularly in the mid-span region and near the trailing edge of the wing. These differences are likely caused by the simplicity of the reconstruction approach, which only varies the spanwise length  $l_1$  while keeping all other geometric parameters unchanged. Despite this simplification, the observed errors remain relatively small. This suggests that even with such a minimal input variation, the method is still capable of capturing the main aerodynamic trends of the original configurations.

A more detailed quantification of these discrepancies would provide valuable insights into the influence of each geometric parameter on the aerodynamic response.

Slice	$Y_{\text{orig}}$	$Y_{\text{reconstructed}}$	Error Upper surface(%)	Error Lower surface (%)
0	0	0	4.75	3.38
1	1.67	1.75	30.6	23.3
2	1.30	1.30	37.3	9.37

Table 7: Cp error on 3 slices for low AR configuration

Slice	$Y_{\text{orig}}$	$Y_{\text{reconstructed}}$	Error Upper Surface (%)	Error Lower Surface (%)
0	0.0	0.0	29.3	2.18
1	2.56	2.20	11.9	7.09
2	1.30	1.30	6.92	2.38

Table 8: Cp error on 3 slices for high AR configuration

Tables 1 and 2 present the relative errors in the pressure coefficient ( $C_p$ ) for three spanwise slices of the wing, comparing the original and reconstructed geometries. The first table corresponds to the low AR configuration and the second table summarizes for high AR configuration.

The data highlights that the most significant differences are located on the upper surface of the wing, particularly in the central region (slice 1) and near the trailing edge (slice 2). In the low AR configuration, the upper surface error exceeds 30% in slice 1 and reaches up to 37% in slice 2. In contrast, the lower surface exhibits much smaller discrepancies, generally remaining below 10%.

These results confirm the visual observations from the  $C_p$  plots: while the simplified reconstruction method manages to preserve the global aerodynamic shape, it struggles to reproduce local details in the pressure distribution, especially in regions where geometric changes have a stronger aerodynamic impact. This highlights the limitations of using only one geometric parameter ( $l_1$ ) for reconstruction and suggests the potential benefit of including additional variables to enhance accuracy.

### 4.3 CSM behavior

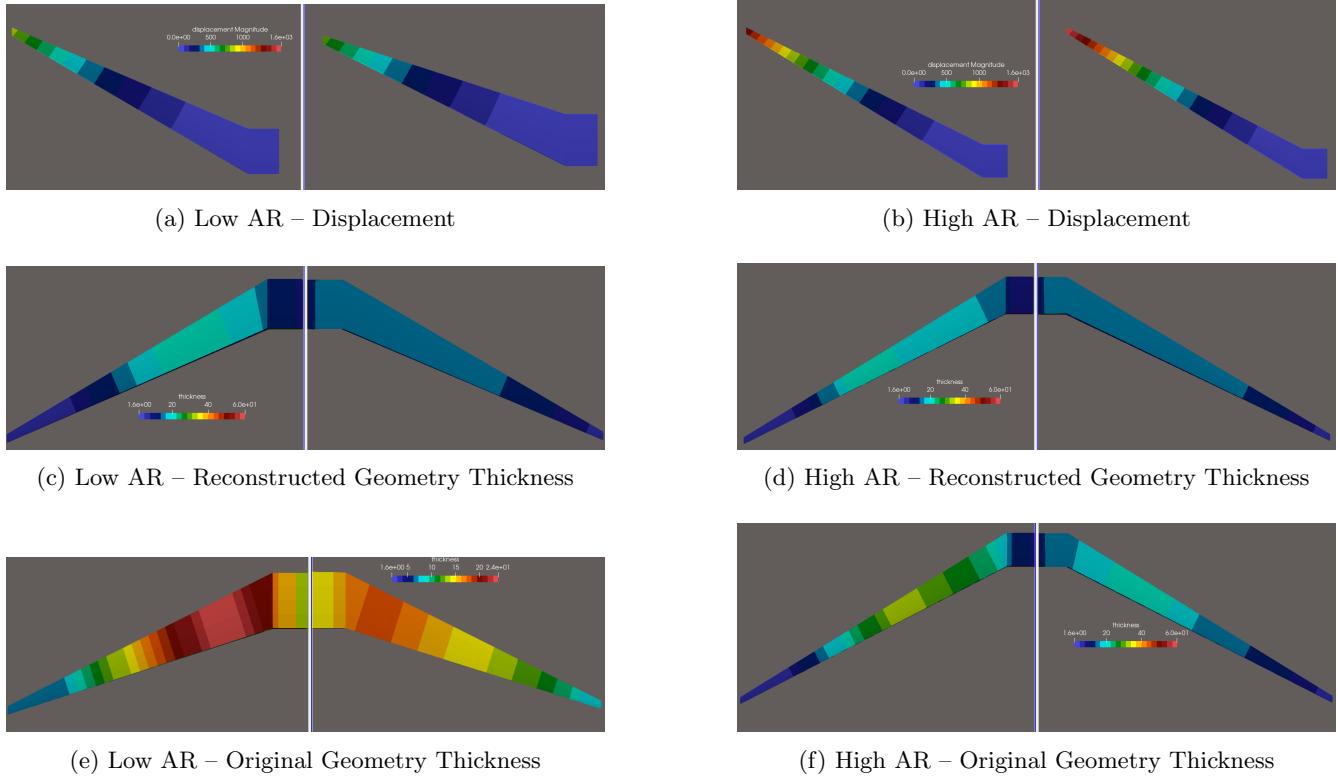


Figure 21: Comparison of displacement and thickness distributions for different aspect ratio configurations

The displacement distributions for both the low and high aspect ratio (AR) configurations, as shown in Figures 21 and 21, are quite similar. However, a slight difference can be observed near the tip of the wing due to geometrical variations between the reconstructed and original geometries. This difference is relatively minor but still worth noting.

In contrast, the thickness distribution shown in Figures 21 and 21 shows more noticeable differences, especially for the low AR configuration. A difference of up to 20 mm is observed, which is quite significant. Aside from this, the general trends for the reconstructed geometry are similar to the earlier POD reconstruction results, where the model doesn't fully capture the wing's structural behavior in some areas especially the tip of the wing.

Figures 21 and 21 show the same pattern, with the low AR configuration having larger thickness differences between the reconstructed and original geometries. This reinforces the earlier conclusion that the simple reconstruction has limitations in capturing the wing's true structural behavior.

### 4.4 L1-reconstruction conclusion

The  $l_1$ -based reconstruction gave good overall results for the geometry, with an average error of around 3.15%. That's small enough to be acceptable for early-stage design and shows that  $l_1$  captures the main trends quite well.

However, when we look closely at the aerodynamic and structural performance using CFD and CSM results, some limits appear. For example, the pressure coefficient ( $C_p$ ) can show differences from 1% to 30%, especially where the airflow is more complex. On the structural side, we also see noticeable differences in displacement and

thickness compared to the original shapes. This means that using only  $l_1$  is not accurate enough to fully capture how the structure behaves.

In short, while this method works well on a global level, it's less accurate as the POD reconstruction, especially for detailed or local effects. To get better results, it would make sense to also include other parameters like  $c_1$  or  $l_2$  to build a reduced model that's both simple and more precise.

Also, as for the POD approach, we have to highlight that these types of ROMs are very simple in terms of complexity. They are displaying very good results for such straightforward models.

## 5 General conclusion

This report presented a data-driven methodology for reducing the complexity of aerodynamic design datasets by combining Proper Orthogonal Decomposition (POD), Self-Organizing Maps (SOM), and inverse optimization techniques. The starting point was a Bayesian Pareto front of 76 optimized wing configurations.

Using Proper Orthogonal Decomposition (POD), we significantly reduced the dimensionality of the Pareto-Front dataset by keeping only the first mode. This simplification worked well for the aerodynamic behavior, both globally (pressure and Mach number fields) and locally ( $C_p$  plots). However, for the structural part, one mode was not enough to capture detailed responses like displacement and thickness, which are highly sensitive to local geometry changes. To improve structural accuracy and get more reliable results, we would need to include at least two more POD modes to capture 95% of the dataset's energy — even if this increases the model's complexity. In short, POD works very well to reconstruct global aerodynamic behavior, even with a simple model, but it struggles with local variations and structural effects. Looking ahead, this is an important result: in future work using multi-objective algorithms, we could keep using just one mode for CFD, while including more modes for CSM. This approach would still reduce the size of the design space without losing informations.

Parallel to that, the SOM analysis revealed strong correlations between the dominant POD coefficient  $\alpha_1$  and certain geometric parameters, particularly the spanwise length  $l_1$ . This observation led us to develop an alternative reconstruction approach based only on  $l_1$ , fixing all other parameters. Although this method reproduced global geometric trends well, it was not accurate enough for CFD and CSM performance, especially in complex flow or load regions. These limitations suggest that, if we aim to use a reduced model based on geometric parameters only, at least two or three variables—such as  $l_1$ ,  $c_1$ , and possibly  $l_2$ —would be required to better capture the relevant physical effects while maintaining a compact model.

In conclusion, this work confirms the value of reduction techniques like POD and SOM for guiding aerodynamic design and optimization. Nonetheless, it also shows the need for careful trade-offs between simplicity and accuracy: either by enriching the POD model with more modes to recover structural detail, or by including a few more key geometrical parameters in simple reconstruction methods.

## 6 Futur perspective of the internship

### 6.1 Context

#### 6.1.1 Partitioned Method – Block Gauss-Seidel (BGS)

In the context of modern aircraft design, composite materials are increasingly used for wings because of their high strength-to-weight ratio. However, their increased flexibility also makes them more sensitive to aerodynamic loads, leading to significant deformations. This makes aeroelastic simulations crucial to ensure both safety and performance.

In this project, aeroelastic simulations were performed using the DASH-tool, which uses the Block Gauss-Seidel (BGS) method. BGS is a partitioned and iterative coupling approach, meaning that separate CFD (fluid) and CSM (structure) solvers are used and are exchanging data at each iteration as explained in Date et al [26].

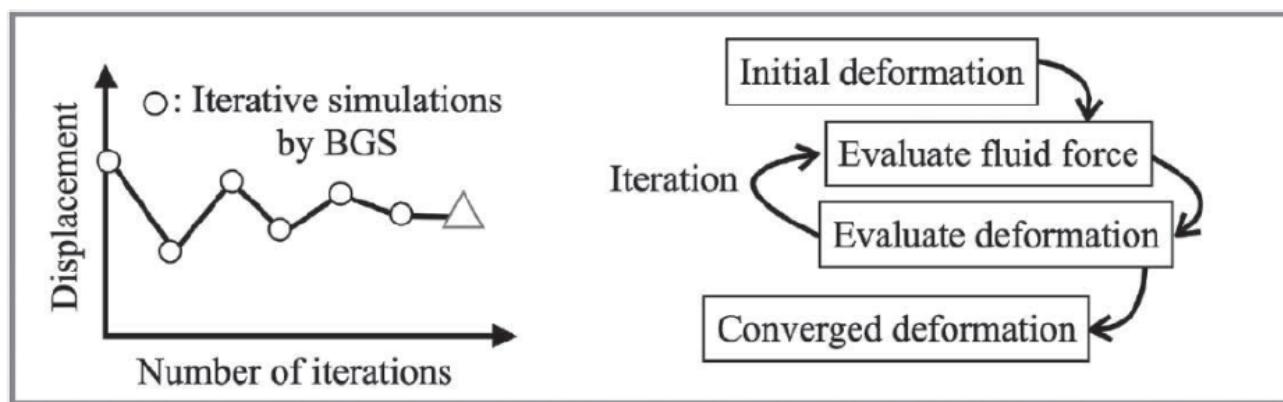


Figure 22: BGS method

As shown in Figure 22, this iterative scheme relies heavily on previous results to update the next solution. The process continues until the displacement converges. While BGS is simple to implement and benefits from solver modularity, it can become very time-consuming for strongly coupled problems — a single simulation may take several hours, and optimization loops may extend over several days. This is mainly because BGS performs one fluid-structure interaction per iteration in a sequential way, which slows convergence in highly nonlinear scenarios.

#### 6.1.2 Fully Partitioned Method

The fully partitioned method builds on the partitioned concept but introduces a more parallel and optimization-based approach. It still relies on independent CFD and CSM solvers — just like BGS — but instead of solving sequentially until convergence, it explores multiple candidate deformation shapes in parallel and evaluates them independently.

The key difference lies in the coupling strategy: while BGS iteratively updates the solution with one CFD-CSM interaction per step, the fully partitioned method assumes several possible displacements from the start, runs all corresponding CFD and CSM simulations, and evaluates a residual (typically force or displacement mismatch). Then, an optimization algorithm (such as Bayesian optimization) is used to find the best deformation that minimizes this residual.

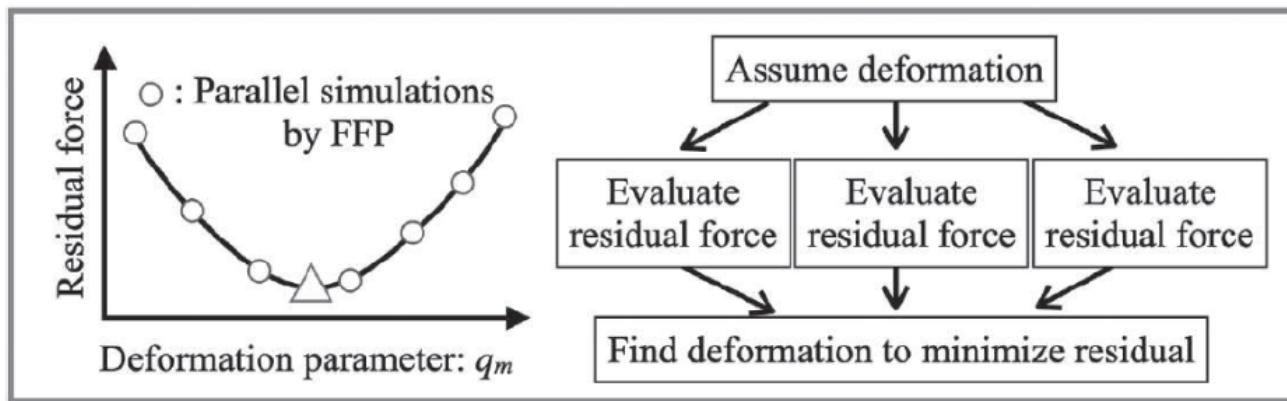


Figure 23: Fully Partitioned Method

As illustrated in Figure 23, although there are still iterations (from initial guess to final solution), the use of optimization drastically reduces the number of CFD/CSM runs needed. This approach enables more flexibility in solver use and opens the door to high-performance parallel computing. It also improves convergence in complex aeroelastic problems, while maintaining a strong physical separation between the fluid and structural models — a key advantage for modular simulation environments [29].

## 6.2 Using POD to Estimate Structural Displacement

In the fully partitioned approach, the optimization process requires three main steps:

- Defining the residual: This residual represents the difference between the fluid and structural solvers at the interface. It can be defined in terms of the mismatch in aerodynamic forces or structural displacements.
- Choosing how to represent the structure deformation: The deformation of the structure needs to be parametrized efficiently to reduce the number of optimization variables.
- Selecting an optimization algorithm: The choice of algorithm directly affects the computational cost and the robustness of the method so the choice is crucial.

While classical mode superposition accurately captures linear structural behavior, it becomes less effective when nonlinearities become significant — for instance, in cases of large bending or twisting, as observed in flexible composite wings. In such situations, Proper Orthogonal Decomposition (POD) proves particularly valuable. The approach consists in extracting a reduced basis directly from a set of realistic deformed configurations, inherently accounting for nonlinear effects.

In this project, I will use the POD-based approach to estimate complex structural displacements under aerodynamic loads. By projecting the deformation onto a POD basis, the geometry can be reconstructed using a small number of coefficients. These coefficients will then be treated as optimization variables, allowing the fully partitioned method to explore deformation states more efficiently, while accounting for nonlinearity. This strategy not only reduces the computational effort compared to full-order modeling but also increases the robustness of the optimization by working in a reduced and physically relevant deformation space.

The main objectives of this project are as follows :

- Build a POD approach
- Apply it to a non-linear geometry displacement obtained by Bayesian optimization algorithm
- Find an accurate representation of the displacement based on the POD coefficients: The final objective is to find a reduced-order representation of the displacement field using the POD coefficients.

While doing this work, I will focus on adapting all the Python scripts I have developed into the DASH tool. The goal is to fully automate the POD/SOM process and make it accessible and easy to use for others. To achieve this, I will employ Bash scripts and .sh files to create an efficient approach.

Additionally, I plan to write a short article summarizing the main findings and methodology of this report. This article will serve as the basis for a presentation I am scheduled to give at a conference in Sendai in November, aiming to showcase the work done during my internship and share it with a broader scientific and technical community.

## References

- [1] Naoya Okamoto, Kazuya Saito, Takahiro Mizuno, and Koji Fukagata. Data-driven reduced-order modeling for transonic buffet flow control. *Journal of Fluid Science and Technology*, 16(1):1–13, 2021.
- [2] S. Ashwin Renganathan, Romit Maulik, and Jai Ahuja. Enhanced data efficiency using deep neural networks and gaussian processes for aerodynamic design optimization. *science direct*, 2020. Use of data driven analysis to reduce dimensionnality of calculation.
- [3] J. Zoltak and W. Stalewski. Multi-objective optimization of a composite wing subject to strength and aeroelastic constraints. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 2011. Use of multi-objective optimisation algorithm to generate wing goemetry.
- [4] Ali Tariq, Khan, Muhammad Ali, Ochoa, Luisa, Fernando, Victor, and Youssef Marzouk. Two-way coupled aero-structural optimization of stable flying wings. *Aerospace*, 2020. Use of multi-objective optimisation algorithm to simultaneously reduce structural weight and drag coefficient.
- [5] Peter I. Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018. Tutorial on Bayesian optimization methods.
- [6] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002. Introduces NSGA-II with fast sorting, elitism, crowding distance.
- [7] David Van Veldhuizen and Gary Lamont. Evolutionary computation and convergence to a pareto front. Technical report, CiteSeerX, 1999.
- [8] Berengère Podvin. An introduction to the proper orthogonal decomposition. Technical report, Centre National de la Recherche Scientifique – LIMSI, Université Paris-Sud, 2001. Tutorial covering theoretical foundation of POD, including eigenvalue formulation and snapshot method.
- [9] Weigang Yao, Min Xu, and Xiaojuan Wang. Aerodynamic system modeling based on proper orthogonal decomposition. *International Journal of Information Technology and Computer Science*, 2011. Creation of Reduce Order Model using POD.
- [10] Patricia Wollstadt and Sebastian Schmitt. Interaction-aware sensitivity analysis for aerodynamic optimization results using information theory. *arXiv preprint arXiv:2112.05609*, 2021. Uses information-theoretic feature selection to identify dominant parameters.
- [11] Steven L. Brunton, Bernd R. Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 2020. introduce SOM in the context of POD as a tool to represent and analyze the latent space from dimensionality reduction.
- [12] Y. Kato, A. Nguyen, T. Lee, and M. Kawahara. Parametric model order reduction by machine learning for fluid flows. *Computational Mechanics*, 2023. show how to combine POD with machine learning methods, opening the door to applying nonlinear techniques like SOM to classify and interpret the modes.
- [13] Teuvo Kohonen. *Self-Organizing Maps*. Springer, 3rd edition, 2001. Provides comprehensive introduction to SOMs and their applications.
- [14] Shinkyu Jeong, Kazuhisa Chiba, and Shigeru Obayashi. Data mining for aerodynamic design space. In *Journal of Aerospace Computing*, 2012. Introduces SOMs for exploratory data aerodynamic analysis.
- [15] SungKi Jung, Won Choi, Luiz S. Martins-Filho, and Fernando Madeira. An implementation of self-organizing maps for airfoil design exploration via multi-objective optimization technique. *ResearchGate*, 2021.
- [16] Berkooz, G.Holmes, and P.Lumley. Proper orthogonal decomposition in analysis and simulation. *Annual Review of Fluid Mechanics*, 1993.
- [17] J.Willcox, K., and Peraire. Airfoil shape optimization via reduced-order modeling and inverse design. *AIAA Journal*, 2002.

- [18] Bui-Thanh, Tan, Damodaran, Murali, Willcox, and Karen. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA Journal*, 2004.
- [19] Tan, Binh, Bui-Thanh, Tan, Willcox, and Karen E. Proper orthogonal decomposition extensions for parametric applications in design. In *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2002.
- [20] Luo, Jiaqi, Zhang, Wei, Zhao, Liang, Zhang, and Zhongliang. Inverse design method based on proper orthogonal decomposition for centrifugal impellers. *Chinese Journal of Aeronautics*, 2017.
- [21] Forrester, Alexander, and Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2009.
- [22] Raissi, Maziar, Perdikaris, Paris, Karniadakis, and George E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 2019.
- [23] B.T. Tan, L. Sirovich, and M. Kirby. Gappy proper orthogonal decomposition for flow field estimation. *Journal of Fluid Mechanics*, 2002.
- [24] Bui-Thanh, T., Willcox, K., and J. Peraire. Aerodynamic data reconstruction using the proper orthogonal decomposition and gappy pod. *AIAA Paper*, 2002. Presented at the 40th AIAA Aerospace Sciences Meeting and Exhibit.
- [25] F. Manzanero, J., García-Hernández, D., Vega, J.M., and Terragni. Review of proper orthogonal decomposition in aerospace applications. *Mathematics*, 2020.
- [26] Yajun Liu, Shugo Date, Toshio Nagashima, Tomonaga Okabe, and Yoshiki Abe. Effects of aeroelastic coupling accuracy and geometrical nonlinearity on performances of optimized composite wings. *Aerospace Science and Technology*, 2024. in press.
- [27] Shugo Date, Yoshiaki Abe, and Tomonaga Okabe. Effects of fiber properties on aerodynamic performance and structural sizing of composite aircraft wings. *Aerospace science and Technology*, 2020. Explore the fluid-structure interaction using the software DASH.
- [28] SungKi Jung, Won Choi, Luiz S. Martins-Filho, and Fernando Madeira. An implementation of self-organizing maps for airfoil design exploration via multi-objective optimization technique. In *Journal of Aerospace Technology and Management*, 2016.
- [29] Yoshiaki Abe, Tomoki Yamazaki, Shugo Date, Minoru Takeuchi, Iori Shoji, Shigeru Obayashi, and Tomonaga Okabe. Fully partitioned method for the static aeroelastic analysis of composite aircraft wings. *Journal of the Japan Society for Composite Materials*, 2022.

# Appendices

## A Python Programs

### A.1 POD Program

```
# Perform POD (Proper Orthogonal Decomposition) on input data

def POD_planform(data):
    data_copy = np.copy(data) # Make a copy to avoid modifying the original data
    ave = np.mean(data_copy, axis=0) # Compute the mean of each column
    data_centered = data_copy - ave # Center the data by subtracting the mean
    std = np.std(data_copy, axis=0) # Compute the standard deviation for each column
    data_std = data_centered / std # Normalize the data

    # Compute the covariance matrix of the standardized data
    cov = np.cov(data_std, rowvar=False)

    # Singular Value Decomposition (SVD) of the covariance matrix
    U, w, ST = np.linalg.svd(cov)
    modes = []

    # Construct each mode using the eigenvectors and append it to the list
    for i in range(np.shape(data)[1]): # Loop over each mode
        mode = U[:, i] * std + ave # Rescale and shift the mode to original scale
        modes.append(pd.DataFrame(mode, columns=[str(i + 1)]))

    # Concatenate all modes side-by-side into a single DataFrame
    Modes = pd.concat(modes, axis=1)

    # Project the normalized data onto the principal directions (temporal coefficients)
    A = data_std @ U

    # Return eigenvectors (U), eigenvalues (w), standard deviation, mean, modes, and temporal coefficients
    return U, w, std, ave, Modes, A
```

Figure 24: Python script used for the POD computation

## A.2 SOM Program

```

def SOM(dataset, nd_x, nd_y, value, N_ITERATIONS, return_winners=False):
    dataset_df = pd.DataFrame(dataset)
    target = dataset_df[value].values

    dataset_norm = dataset_df.copy() # Copy for normalization
    for col in dataset_norm.columns:
        min_val = dataset_norm[col].min()
        max_val = dataset_norm[col].max()
        dataset_norm[col] = (dataset_norm[col] - min_val) / (max_val - min_val) # Min-max normalization

    # Initialize and train the SOM
    som = MiniSom(nd_x, nd_y, dataset_norm.shape[1],
                  sigma=2.0, learning_rate=0.5,
                  neighborhood_function='gaussian', random_seed=0)
    som.random_weights_init(dataset_norm.values) # Random weight initialization
    som.train(dataset_norm.values, N_ITERATIONS, verbose=False) # Train SOM

    # Create heatmap initialized to zeros
    heatmap = np.zeros((nd_x, nd_y))
    count = np.zeros((nd_x, nd_y))
    winners = []

    # Fill heatmap with target values at BMU locations
    for datapoint, val in zip(dataset_norm.values, target):
        x, y = som.winner(datapoint) # Get BMU
        winners.append((x, y)) # Store BMU
        heatmap[x, y] += val # Add value to corresponding cell
        count[x, y] += 1 # Count how many points fall into each cell

    # Compute mean in each cell (average target values)
    with np.errstate(divide='ignore', invalid='ignore'):
        heatmap = np.divide(heatmap, count, out=np.full_like(heatmap, np.nan), where=count != 0)

    # Interpolation to smooth the heatmap
    grid_x, grid_y = np.mgrid[0:nd_x, 0:nd_y]
    known_points = np.argwhere(~np.isnan(heatmap))
    known_values = heatmap[~np.isnan(heatmap)]
    heatmap_interp = griddata(known_points, known_values, (grid_x, grid_y), method='linear')

    # Fill remaining NaN values using nearest-neighbor interpolation
    nan_mask = np.isnan(heatmap_interp)
    if np.any(nan_mask):
        heatmap_interp[nan_mask] = griddata(known_points, known_values, (grid_x, grid_y), method='nearest')[nan_mask]

    if return_winners:
        return heatmap_interp.T, winners, som # Return transposed heatmap, BMUs, and SOM object
    return heatmap_interp.T, som # Return only heatmap and SOM object

```

Figure 25: Python script used for the SOM computation

## B Geometrical Parameter Error

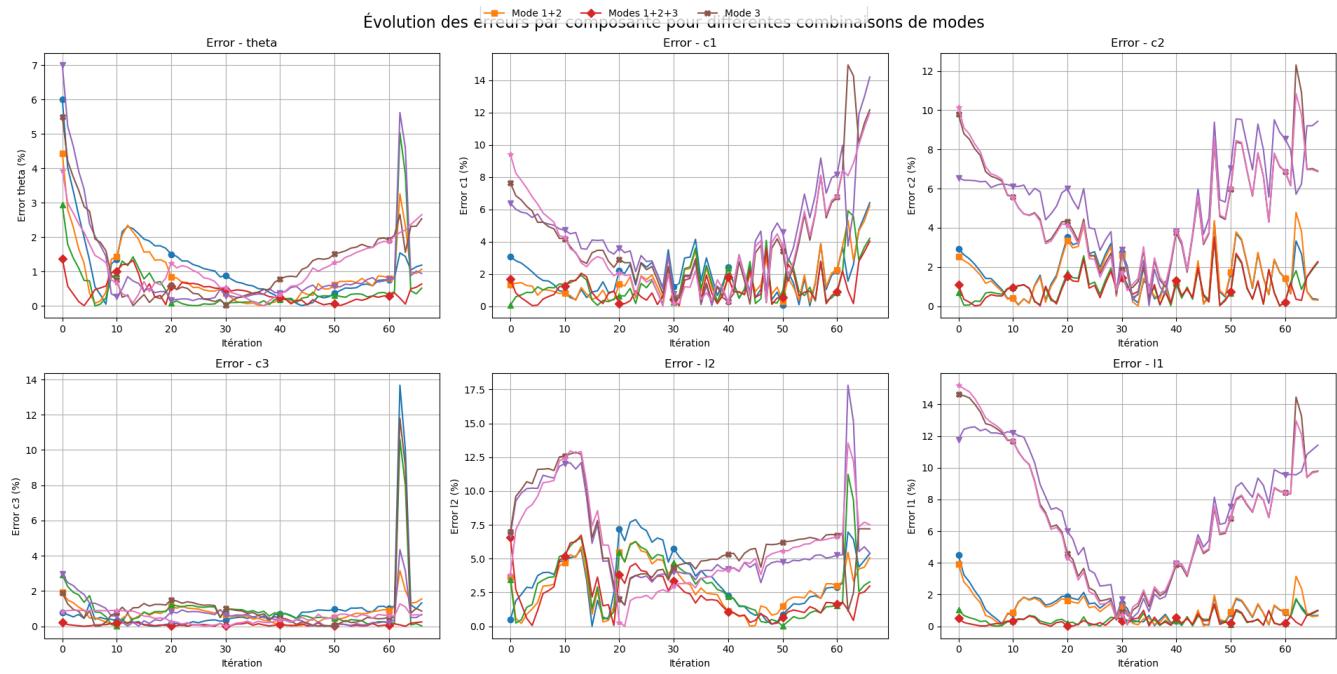


Figure 26: Reconstruction error on the geometrical parameters

## C SOM Heatmaps for Geometrical Parameters

### C.1 $c_1$

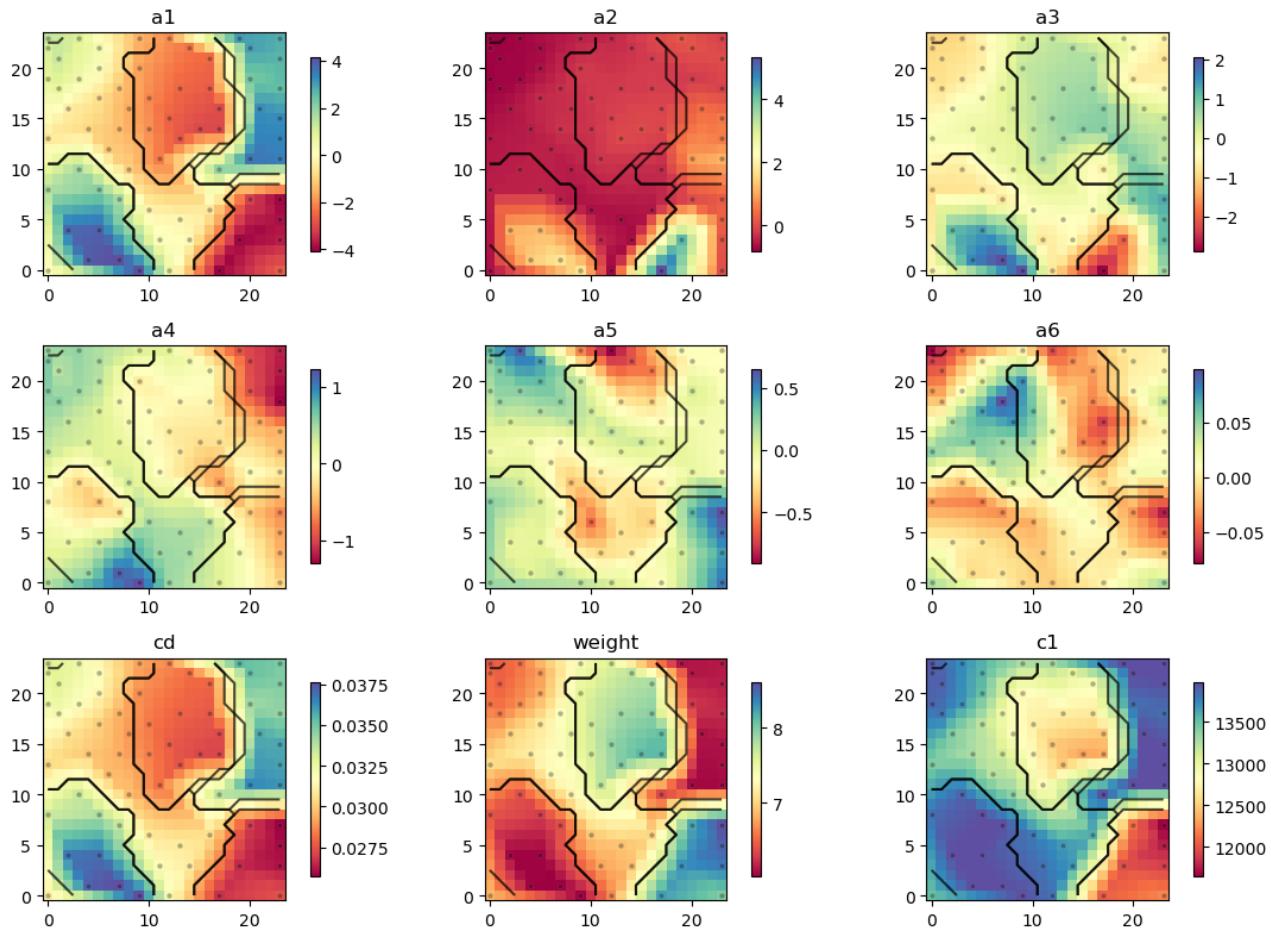


Figure 27: SOM heatmap for the parameter  $c_1$

## C.2 c2

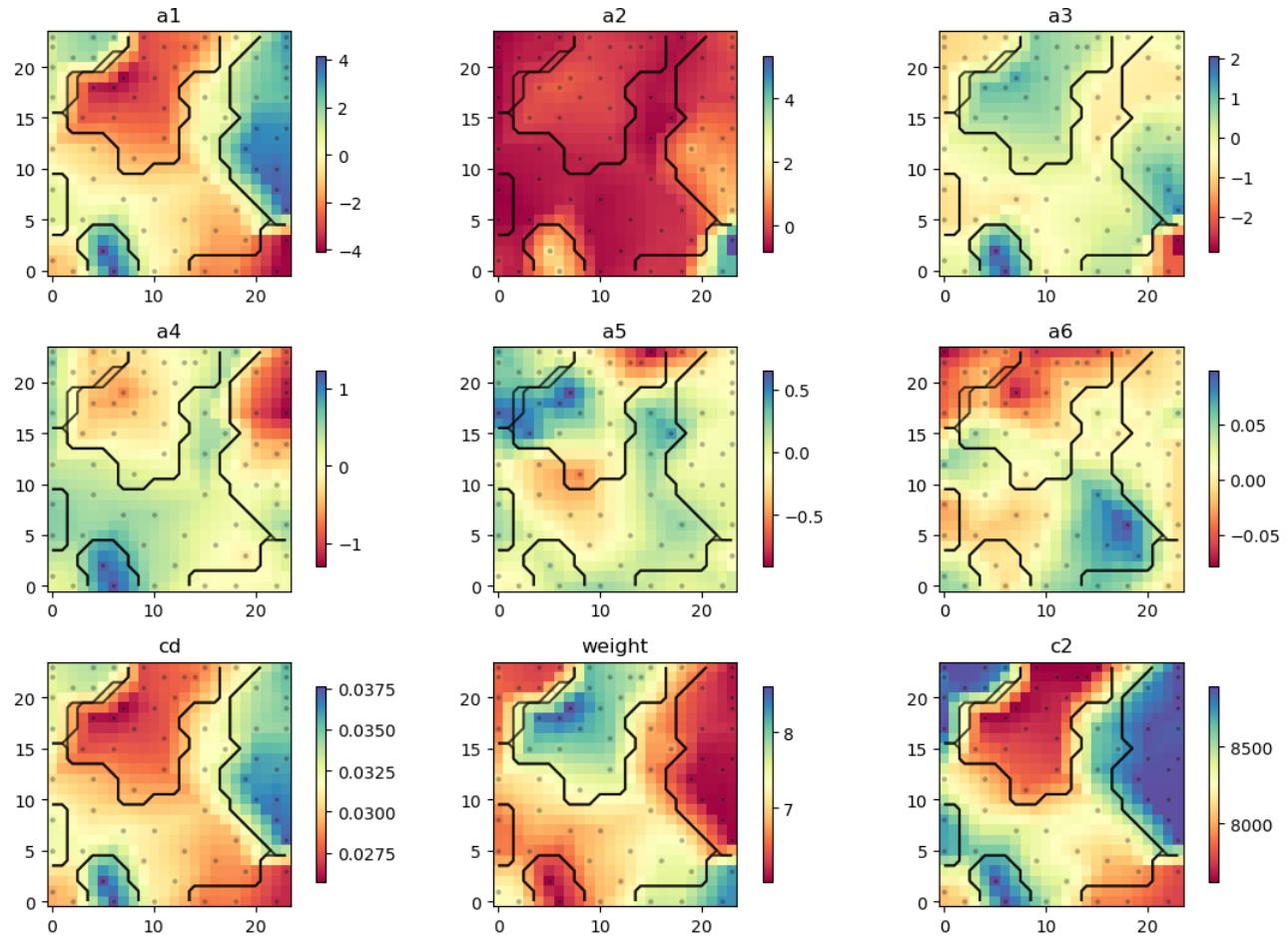


Figure 28: SOM heatmap for the parameter  $c_2$

### C.3 c3

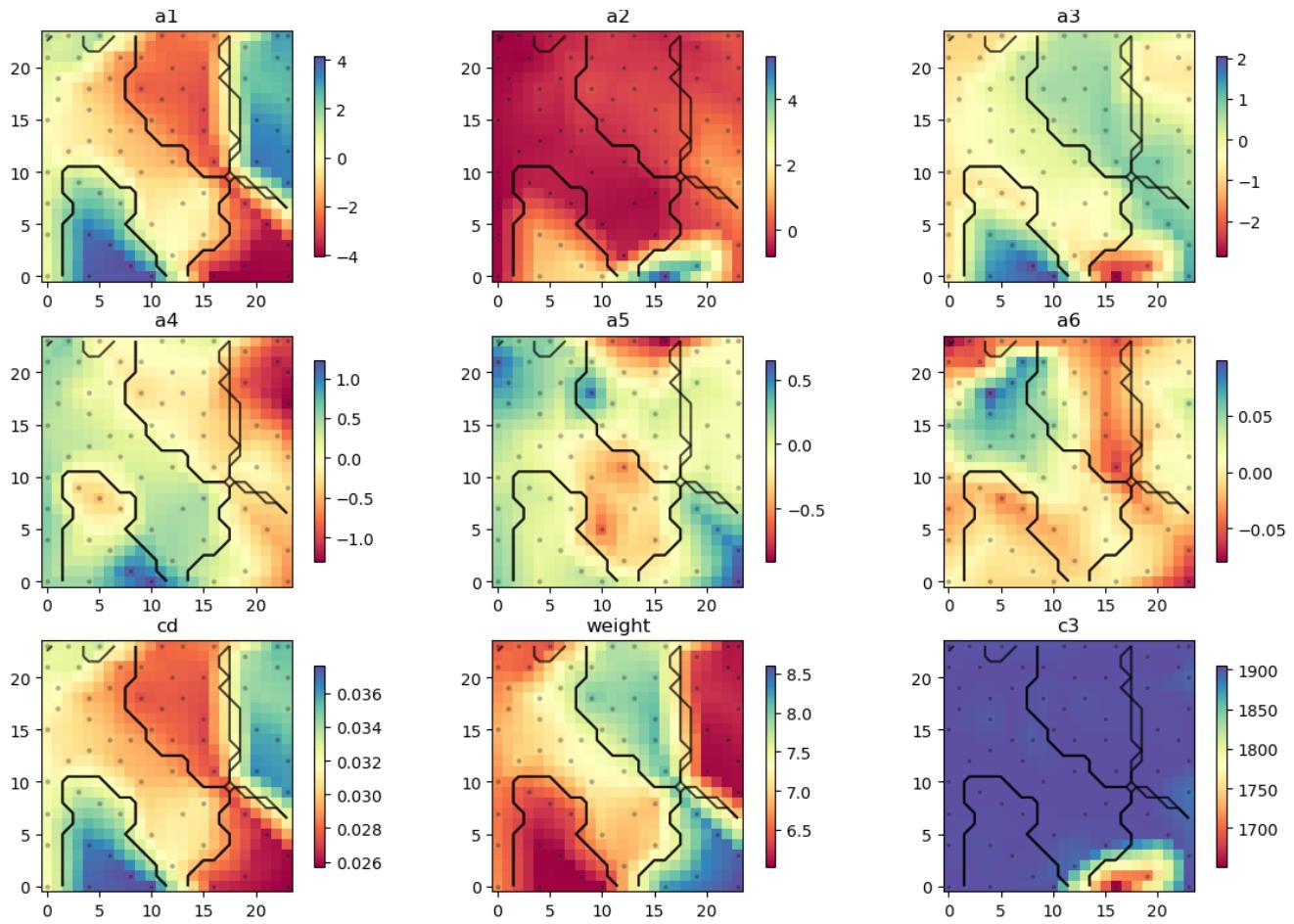
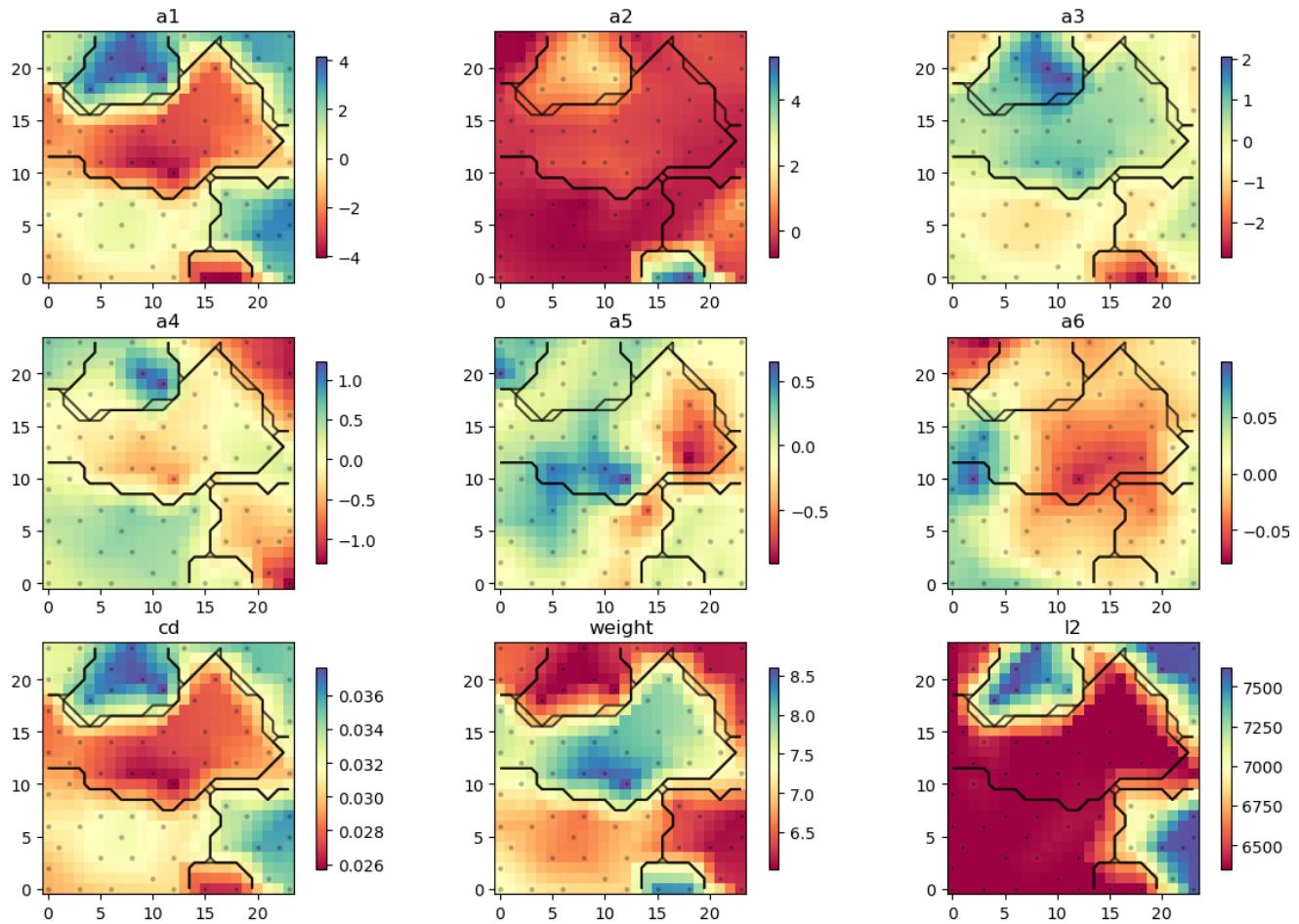


Figure 29: SOM heatmap for the parameter  $c_3$

## C.4 l2

Figure 30: SOM heatmap for the parameter  $l_2$

### C.5 theta

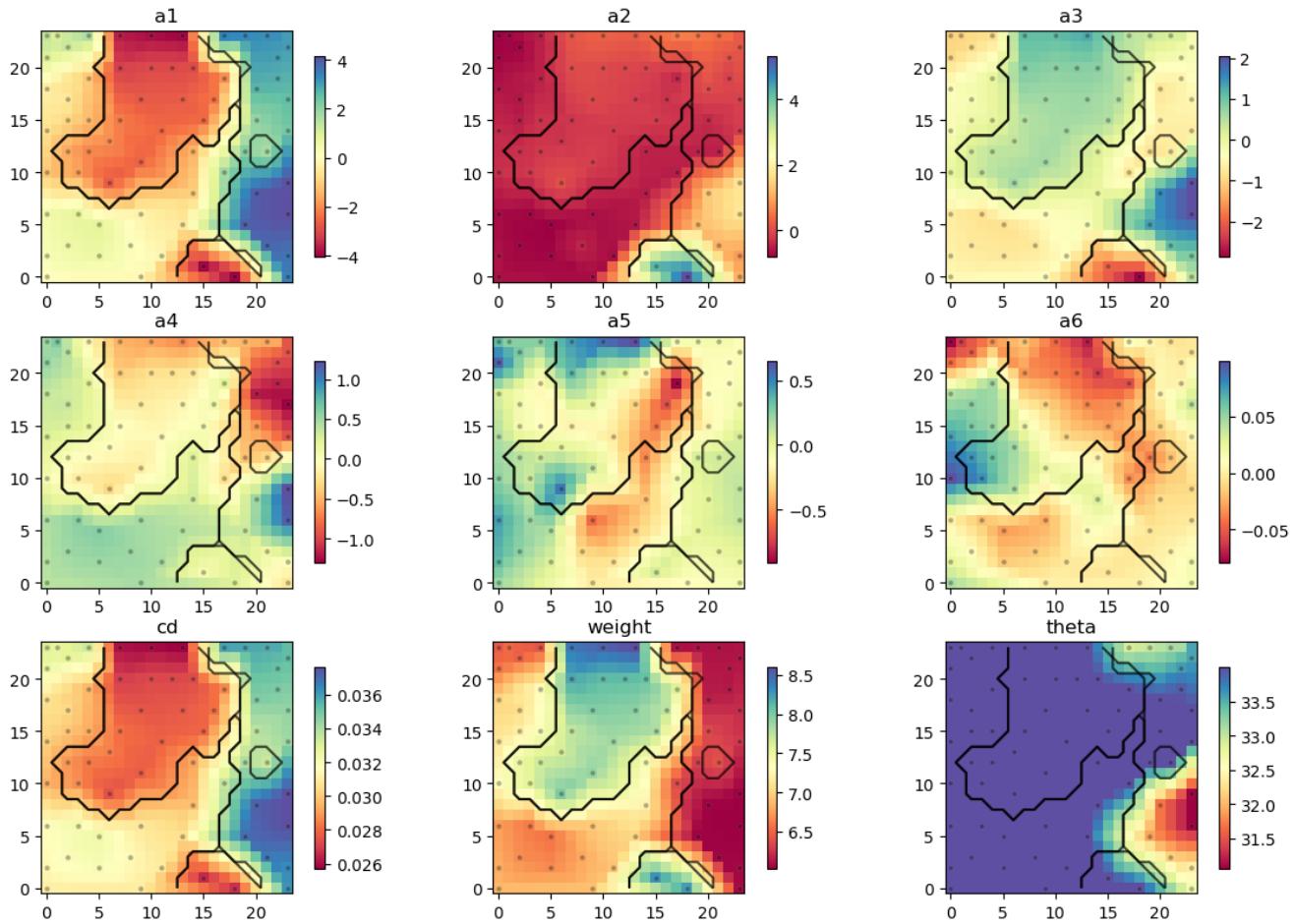


Figure 31: SOM heatmap for the parameter  $\theta$

## Résumé

Ce travail présente une approche de réduction d'ordre pour l'analyse de géométries d'ailes optimisées en matériaux composites, issues d'un front de Pareto généré par des algorithmes d'optimisation multi-objectifs (Bayesian Optimization et NSGA-II). La méthode repose sur la Décomposition en Modes Propres (POD) afin de capter les principales tendances géométriques tout en réduisant la complexité du jeu de données. Le premier mode POD s'avère suffisant pour reconstruire les géométries avec une erreur moyenne inférieure à 5 %, tout en préservant les caractéristiques aérodynamiques majeures validées par des simulations CFD. L'analyse par Cartes Auto-Organisatrices (SOM) permet d'identifier une forte corrélation entre la longueur en envergure  $l_1$  et le coefficient modal  $\alpha_1$ , motivant une reconstruction simplifiée ne faisant varier que ce paramètre. Si cette approche réduit efficacement l'erreur géométrique, elle ne permet pas de restituer avec précision les résultats CFD et CSM, en particulier dans les zones à forte variation locale. L'étude montre ainsi que l'usage combiné de POD et SOM constitue une base robuste pour la réduction de modèles en aérodynamique, sous réserve d'un choix judicieux des paramètres conservés.

## Mots clefs

- Réduction de modèle (ROM)
- Décomposition en modes propres (POD)
- Optimisation multi-objectifs
- Analyse aéroélastique (CFD/CSM)
- Représentation à faible dimension

## Abstract

This study presents a model order reduction approach for the analysis of optimized composite wing geometries derived from a Pareto front generated using multi-objective optimization algorithms (Bayesian Optimization and NSGA-II). The method is based on Proper Orthogonal Decomposition (POD), which extracts dominant geometrical trends while reducing dataset complexity. The first POD mode alone was sufficient to reconstruct the geometries with an average error below 5%, while preserving key aerodynamic features validated through CFD simulations. Self-Organizing Maps (SOM) revealed a strong correlation between the spanwise length  $l_1$  and the modal coefficient  $\alpha_1$ , motivating a simplified reconstruction strategy varying only  $l_1$ . While this approach proved effective for geometry recovery, it failed to accurately reproduce CFD and CSM results, particularly in regions with complex flow or structural behavior. This confirms that the combined use of POD and SOM offers a solid framework for aerodynamic model reduction, provided that the retained parameters are carefully selected.

## Keywords

- Reduced-order modeling (ROM)
- Proper Orthogonal Decomposition (POD)
- Multi-objective optimization
- Aeroelastic analysis (CFD/CSM)
- Low-dimensional representation