MATHEMATICAL MODELING AND COMPUTING, Vol. 11, No. 1, pp. 154–165 (2024)

Simultaneous surrogate modeling and dimension reduction using unsupervised learning. Application to parametric wing shape optimization

Karafi Y.[1], Moussaoui Z.[1], Abou El Majd B.[1,2]

[1] *LMSA Laboratory, Faculty of Science, Mohammed V University in Rabat, Morocco*
[2] *University of Lille, 59655 Villeneuve–d'Ascq, France*

(Received 31 May 2023; Revised 16 February 2024; Accepted 18 February 2024)

This paper presents a machine-learning-based approach that enables simultaneous surrogate modeling and dimension reduction and applies it to aerodynamic parametric shape optimization. Aerodynamic shape optimization is a crucial process in various industries, including aerospace, automotive, and renewable energy. It involves iteratively improving the properties of a system by evaluating an objective function and driving its minimization or maximization using an optimization algorithm. However, the evaluation of aerodynamic objective functions requires computationally expensive operations, such as solving complex fluid dynamics equations and calculating performance metrics like lift and drag coefficients. This computational cost becomes particularly burdensome when derivative-free optimization algorithms need to evaluate numerous samples per iteration. Additionally, when the design space dimension is high, the efficiency and effectiveness of the optimization process decrease. To address these challenges, the paper proposes combining surrogate modeling and dimension reduction. Surrogate modeling constructs a reduced order model that approximates the coefficients of interest in a cost-effective manner, while dimension reduction identifies the most relevant design space dimensions using techniques like Proper Orthogonal Decomposition. The paper suggests an integrative approach that employs Artificial Neural Networks (ANN) and Unsupervised Learning, specifically AutoEncoder networks, to simultaneously build a surrogate model and reduce the problem dimension. This technique is applied to optimize the shape of an airplane wing aerofoil under transsonic flight conditions. The wing shape is parameterized using Free Form Deformation (FFD). The paper demonstrates that the suggested approach enables rapid and effective shape optimization.

Keywords: *machine learning; autoencoder; free-form deformation; artificial neural networks; shape optimization; aerodynamic analysis.*

2010 MSC: 65Kxx **DOI:** 10.23939/mmc2024.01.154

1. Introduction

Aerodynamic Shape Optimisation (ASO) is an essential and effective aspect of automating the design process, traditionally carried out using Computational Fluid Dynamics (CFD) simulations and manual design modifications. With powerful high-performance computing resources, CFD-based ASO has been applied to the design of wings [1], tails and other components, resulting in considerably reduced aircraft development cycle time and enhanced design performance. However, the efficiency of aerodynamic shape design optimization is mainly affected by three fundamental aspects: the dimensionality of the geometric design space, the cost of the aerodynamic analysis, and the convergence rate of the optimization algorithm. Recently, advanced Machine Learning (ML) methods have shown the potential to efficiently parameterize the aerodynamic shape, accurately predict aerodynamic performance with low computational cost, and innovate ASO workflows. With this new approach to aerodynamic shape optimization, promising results in terms of efficiency are expected.

Machine learning (ML) has emerged as an effective means of reducing the cost of aerodynamic evaluations in CFD-based aerodynamic shape optimization, which is largely dominated by the cost

of evaluating aerodynamic objective and constraint functions (and their derivatives with respect to design variables). Although CFD analysis provides rich information about the flow variables in the computational domain, ASO is typically based on a few aerodynamic performance metrics, such as lift and drag coefficients. As a result, modeling the aerodynamic coefficients is of great interest in aerodynamic design. A simple prediction function is fitted between the shape design variables and the aerodynamic coefficients using training data generated by high-fidelity aerodynamic analyses to accomplish this modeling. This prediction model is commonly referred to as the Surrogate Model (SM) or metamodel. Traditional ML approaches, such as kriging [2], have been successfully used in various engineering fields. Variations of kriging, such as co-kriging and hierarchical kriging [3], were developed to take advantage of multi-fidelity simulations. Boutemedjet and all [4] proposed the aerodynamic design procedure of a mini unmanned aerial vehicle involving genetic algorithms and artificial neural networks for wing preliminary computation. Liao and all [5] introduced multi-fidelity convolutional neural networks as a surrogate model for aerodynamic optimization. Jun TAO et al. [6] used a PCA-DBN substitution model based on Principal Component Analysis (PCA) and Deep Belief Network (DBN). PCA-DBN is applied to robust aerodynamic design optimizations of a natural laminar flow (NLF) airfoil and a transonic wing.

Compared to supervised learning, which relies on labeled training data, unsupervised learning operates with unlabeled training data. Unsupervised ML algorithms analyze the data patterns and automatically set up learning rules. Dimensionality reduction is a common unsupervised ML technique, allowing for the handling of problems with a large number of variables by coding the information into a reduced number of parameters or latent variables. Examples of such techniques are Principal Component Analysis (PCA) and Singular Value Decomposition (SVD). Artificial neural networks, such as Auto-Encoders (AEs), can represent nonlinear functions and efficiently encode information in a few latent variables while minimizing the reconstruction error during decoding. Variational Auto-Encoders (VAEs) improve on classic autoencoders by ensuring that the latent variables follow smooth statistical distributions, typically normal distributions. Kou et al. [7] used autoencoders to improve the optimization of airfoils where a multidisciplinary objective function combined aerodynamic and aeroacoustic targets. [8] used a convolutional-autoencoder to compress the CFD solution data into latent vectors (encodings) which had a much lower dimensionality than the original CFD solution space. An additional deep feedforward neural network (DNN) model was used to predict the latent vectors using the CFD model input boundary conditions. [9] used a convolutional neural network in conjunction with a generative adversarial network (GAN) to produce a one-to-one mapping from the parameters defining the aerofoil geometry to the resultant 2D pressure field.

In the present study, we propose an integrated deep learning approach that combines Auto-Encoders (AE) with the Surrogate model (SM) to predict the aerodynamic performance, specifically the drag and lift coefficients. The first step involves using the AE to establish a relationship between the reduced dimension variables. By incorporating the surrogate model, we can efficiently explore the parameter space to optimize aerodynamic shape. The surrogate model provides a quick estimation of aerodynamic performance for different shapes, guiding the search for optimal configurations. This model-assisted optimization approach accelerates the optimization process while maintaining reasonable prediction accuracy, which is particularly beneficial in domains where CFD simulations are computationally demanding in terms of time and resources.

The structure of this paper is as follows: in Section 2, we present the master points generation process for aerodynamic analysis. Section 3 introduces the proposed AE-SM model for reducing both the order and dimension. Section 4 showcases numerical results that demonstrate the application of the approach from Section 3 in the aerodynamic optimization of a transonic aircraft wing.

2. Master points generation

The data generation process for aerodynamic analysis is a critical step in designing efficient and high-performance aircraft. The process involves collecting representative data points, also called master

points, that will be used to build the aerodynamic models. These master points are the input and output values used to create the model, and they represent the design space of the aircraft.

To collect these master points, space-filling sampling techniques are employed, which enable the reduction of the number of collected points while maximizing the coverage of the design space. This technique ensures that the collected data points are as diverse and representative as possible. The collected points set comprises both the input parameters X and the output data Y. The steps of the aerodynamic data generation, are shown in Figure 1.

Fig. 1. Aerodynamic Data Generation.

— Parameterization and design of experiments (DoE): Specify the parameter space by defining the input variables and their range. Generate samples based on DoE theory.
— Mesh generation and CFD simulation: set up the flow field computational block and meshing. Obtain the aerodynamic data at the sample points by conducting CFD simulations.
— Establishment of aerodynamic database preparation: organize the design space and its corresponding aerodynamic characteristics into a database.

The process of choosing samples from the design space is called the Design of Experiment (DoE). Latin hypercube Sampling (LHS) is used as a DoE method to establish the distribution of input variables. LHS is a method of approximate random sampling from a multivariate parameter distribution belonging to hierarchical sampling technology and is often used in DoE. Samples x_{ij} obtained using the LHS method can be expressed as follow:

$$x_{ij} = \frac{\pi_j^i + U_j^i}{N}, \quad 1 \leqslant j \leqslant d, \quad 1 \leqslant i \leqslant N, \tag{1}$$

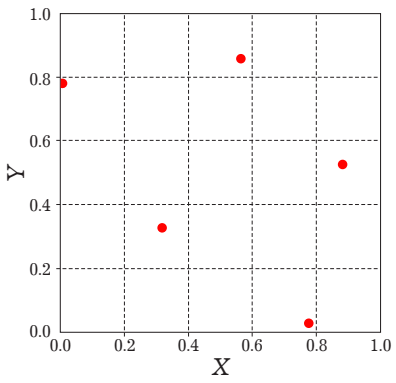

Fig. 2. Schematic of 5 sample points selected by Latin hypercube sampling.

where i denotes the i^{th} sample, j denotes the j^{th} design variable, U denotes a random number in $[0, 1]$, and j denotes a random permutation in $\{0, 1, \ldots, N-1\}$. An example in which the LHS method selects sample points in a DoE problem of two-dimensional input is shown in Figure 2.

The training data is composed of CFD analyses on samples chosen from the design space to compute the aerodynamic force coefficients, such as lift coefficient, and drag coefficient (with a given range for the free-stream Mach number and angle of attack). In the CFD analysis of each wing sample, we use the finite volume method to solve the compressible Euler's equations.

3. Surrogate modeling and dimension reduction

3.1. Surrogate modeling (SM)

In aerodynamic shape optimization, surrogate modeling plays a crucial role. The High Fidelity Model (HFM) based on Computational Fluid Dynamics (CFD) has related inputs and outputs:

$$Y = \mathcal{F}(X). \tag{2}$$

The objective of surrogate modeling is to substitute the computationally expensive high fidelity model, denoted as $\mathcal{F}$, with a Low Fidelity Model (LFM) f. This LFM is constructed based on observed parameters X and their corresponding outputs $Y = \mathcal{F}(X)$, allowing it to approximate the output $\hat{Y} = f(X)$ for new and unseen parameters X. By employing surrogate modeling techniques, such as meta-models, the time-consuming calculations involved in solving the Navier Stokes Equations

in Computational Fluid Dynamics (CFD) simulations can be mitigated. The surrogate model entails the creation of a mathematical approximation model, utilizing previously collected sample data, to make predictions of the objective function values at untested points. The framework of the surrogate model is depicted in Figure 3.

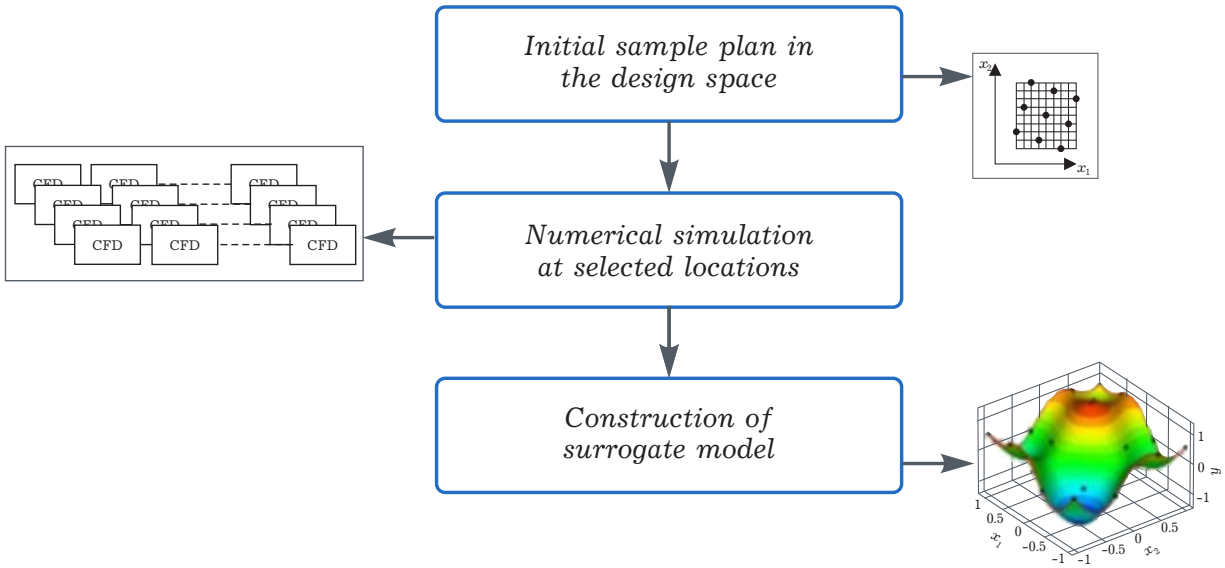

Fig. 3. Surrogate model framework.

In other words, surrogate modeling aims to build a simplified model that can predict the outputs of the HFM using only the input parameters. To achieve this, machine learning techniques such as regression are used to capture the relationship between the input parameters and the corresponding outputs of the HFM. The surrogate model can then predict outputs for new parameter combinations, saving time and resources by avoiding costly computational simulations each time. The most common surrogate modeling methods include Artificial Neural Networks (ANNs) [10], Multi-Layer Perceptron (MLP) [11], Recurrent neural networks (RNN) [11], Bayesian Neural Network (BNN) [12], and Gaussian Processes (GPs) [13].

3.2. Reducing both the order and dimension of the HFM

In the context of the High Fidelity Model (HFM), reducing both the order and the dimension of the Low Fidelity Model (LFM) can lead to significant computational savings. One way to achieve this is to introduce the third variable Z and additional model g to relate the variables X to Z,

$$\begin{cases} X = g(Z), \\ y = f(X). \end{cases} \tag{3}$$

The first approach involves introducing a model g that relates the reduced dimension variable X to Z, and then using a surrogate modeling technique to approximate the relation between Y and X. This approach is known as dimension reduction, and it involves the use of techniques such as Principal Component Analysis (PCA) or Proper Orthogonal Decomposition (POD) to obtain g, and a surrogate modeling technique f.

Alternatively, the second approach involves introducing two supplementary models g_1 and g_2 (4) that relate X and Y to Z. In this approach, the variable X is related to Z through g_1, while Y is related to Z through g_2,

$$\begin{cases} X = g_1(Z), \\ y = g_2(Z), \end{cases} \quad \text{with} \quad y = f(X). \tag{4}$$

g_1 can be thought of as a decoder that restitutes the parameters from the reduced space and g_2 as a mapper that maps lower dimension parameters to outputs. The relation between X and Y is then approximated using the HFM.

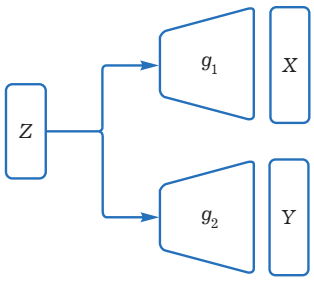

Fig. 4. Schematic of model with $y = f(X)$.

The proposed model is composed of two main parts, the first being an AE Auto-Encoder technique and the other being a substitution model prediction block. In this section, we present the theory behind this model with three different architectures. The surrogate model that aims at predicting the aerodynamic performance (drag and lift coefficients in our case) is also presented. Next, we describe in detail the custom training loop in which the AE and the SM are trained simultaneously.

3.2.1. Auto-Encoder (AE) model

An Auto-Encoder (AE) is a generative model used in unsupervised learning. The AE aims to reproduce its input while mapping the data to a latent space. It is commonly employed for dimensionality reduction by training the network to disregard irrelevant data.

An AE (Auto-Encoder) is a model that aims to represent an observation in the latent space without the explicit use of probability distribution. It consists of an **Encoder E**, which generates a compressed representation of each attribute of the latent state, and a **Decoder D** that attempts to reconstruct the input data from this latent representation. Unlike the VAE (Variational Autoencoder), the AE does not directly model the probabilistic distribution of the data in the latent space.

The main components of an AE involve taking input data X and using a neural encoding network to generate an output vector of dimension d, where d represents the dimension of the latent space. Specifically, the encoder produces a vector of Z values in $\mathbb{R}^{d\times 1}$, which represents the latent space associated with the input data:

$$Z = \mathbf{E}(X). \tag{5}$$

The sampled Z point is then used as the input of the decoder neural network, which aims to reconstruct the input data:

$$\hat{X} = \mathbf{D}(Z).$$

The loss function of an AE typically consists of a reconstruction term, which is calculated on the final layer of the network, and it serves to ensure efficient encoding and decoding of the input data.

The reconstruction term measures the discrepancy between the original input data and the reconstructed output produced by the decoder. Mathematically, it can be represented as

$$\mathcal{L}(X, \hat{X}) = \left\|X - \hat{X}\right\|_2^2. \tag{6}$$

The AE is trained by optimizing this reconstruction loss, aiming to minimize the discrepancy between the original data and the reconstructed data. By minimizing the reconstruction loss, the AE learns to extract meaningful and informative representations in the latent space that can effectively capture the salient features of the input data.

Regularization term. The representation of the AE in the latent space involves determining the bounds or ranges for the intervals of Z. For instance, if we observe that the latent points are widely scattered and lack organization, we can apply regularization techniques to impose structure on these latent points.

By regularizing the latent points, we aim to encourage certain desirable properties such as smoothness, sparsity, or clustering in the latent space. This can be achieved through the addition of regularization terms to the loss function of the AE during training. Common regularization techniques include L_1 or L_2 regularization, which introduce penalties based on the magnitude of the latent variables. In this paper, we employ the regularization constraint of equation (7) to ensure that the latent design space is the hyper-volume of radius 1,

$$R_1(Z) = \min\left(|1 - \|Z\|_\infty|, 0\right). \tag{7}$$

With the above regularization term, the loss function of AE becomes:

$$\mathcal{L}(X, \hat{X}) = \left\|X - \hat{X}\right\|_2^2 + \lambda_1 R_1(Z), \tag{8}$$

where λ_1 controls the weight of the regularization term.

3.2.2. Multi-model auto-encoder surrogate (AE-SM)

The proposed multi-model integrates two models, AE and SM, to capture both the geometry and the relationship between geometry and aerodynamic performance. The simplest and most straightforward approach is to merge both the input X and output Y into the same dataset and process them using an Auto-Encoder, as illustrated in Figure 5. Firstly, the encoder reduces the dimension, and then the decoder (D-SM) restores the original dimension and predicts the desired outputs based on the restored full-dimensional parameters.

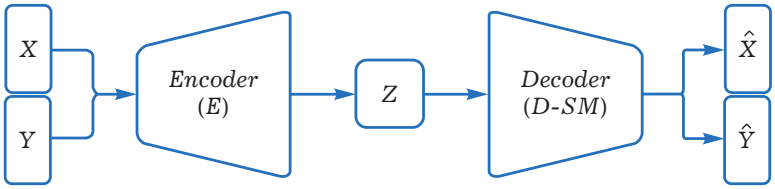

Fig. 5. Architecture of AE-SM.

The AE and SM models are trained concurrently, with the objective of minimizing a combined loss function that incorporates the loss function for the AE (8) and the Mean Square Error (MSE) for the SM,

$$\mathcal{L}(\hat{X},\hat{Y}) = \left\|X-\hat{X}\right\|_2^2 + \lambda_1 R_1(Z) + \left\|Y-\hat{Y}\right\|_2^2. \tag{9}$$

It is important to note that training the AE and SM simultaneously has a significant semantic impact compared to training the models separately. Specifically, the encoder influences all three losses (9). Consequently, the encoder is also trained to minimize the $\|Y-\hat{Y}\|_2^2$ loss, which would otherwise only depend on the surrogate model. On the other hand, the SM is trained using a latent space that gradually takes shape during the simultaneous AE training. This compels the encoder to adapt the latent representation to the desired features of the SM, providing regularization and preventing overfitting. While this approach may lead to a higher reconstruction loss $\|X-\hat{X}\|_2^2$, it ultimately results in improved overall performance.

Some critical aspects of this last model can be evidenced, for example, X and Y are decoupled. This raises equations about the soundness of this approach since the last part of equation (3) (i.e. $Y = f(X)$) is not enforced in the loss function. To drive the model training considering the correlation between X and Y one can think of a correlation metric such as Pearson correlation ρ given hereafter:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{\mathbb{E}\left[(X-\mu_X)(Y-\mu_Y)\right]}{\sigma_X \sigma_Y}. \tag{10}$$

A good model is expected to reproduce the same correlation as between $\hat{X}$ and $\hat{Y}$, thus minimizing the gap between $\rho_{X,Y}$ and $\rho_{\hat{X},\hat{Y}}$. We subtract two correlation matrices to obtain the correlation gap:

$$\rho = \rho_{X,Y} - \rho_{\hat{X},\hat{Y}}.$$

The cost function is defined as the sum of the absolute values of all elements in the "gap" matrix.

Alternatively, you can normalize the cost function by dividing the sum by the total number of elements in the "gap" matrix. This makes the cost function independent on the matrix size:

$$R_2(\rho) = \frac{\sum_{ij} |\mathrm{gap}_{ij}|}{n},$$

where n is the total number of elements in the "gap" matrix.

The complete expression of this loss function is

$$\mathcal{L}(\hat{X},\hat{Y}) = \left\|X-\hat{X}\right\|_2^2 + \lambda_1 R_1(Z) + \left\|Y-\hat{Y}\right\|_2^2 + \lambda_2 R_2(\rho). \tag{11}$$

This loss function consists of multiple components. The first term measures the squared Euclidean distance between the input data X and their corresponding predictions $\hat{X}$, quantifying the reconstruction error. The second term, $\lambda_1 R_1(Z)$, represents a regularization term applied to the latent space Z with a regularization parameter λ. The third term, $\|Y-\hat{Y}\|_2^2$, calculates the mean squared error between the

actual performance data Y and their predicted values $\hat{Y}$. This term captures the deviation between the predicted and true performance, allowing the model to learn and improve its predictions. Finally, the last term, represented as $\lambda_2 R_2(\rho)$, denotes a measure of the discrepancy or gap between the output correlation predictions.

By minimizing this loss function during the training process, the model aims to simultaneously reduce the reconstruction error, promote structured latent representations through regularization, improve performance predictions, and minimize the gap in correlation predictions.

4. Numerical results

4.1. Airfoil shape design optimization

The objective of a deterministic simulation-based shape optimization problem is to minimize a cost function $\mathcal{J}$, which relies on shape and state variables. In parametric approaches, the shape is described using a limited set of variables (x) that serve as optimization variables. This parametric approach effectively transforms the initial shape optimization problem, which has an infinite number of dimensions, into a problem with a finite number of unknowns. The state variables (i.e., the physical flow fields), governed by equations such as the Euler equations in this study, implicitly depend on the shape variables.

When optimizing the shape of an airfoil, the primary goal is typically to reduce drag while maintaining a minimum lift, subject to certain constraints. Drag and lift are commonly quantified using their respective coefficients, C_D and C_L. In practical scenarios, the airfoil configuration must be optimized to minimize the drag-to-lift ratio (C_D/C_L) at the design point. The optimization problem can be formulated as

$$\underset{\mathbf{x}\in\mathbf{R}^n}{\text{Minimize}} \left[\mathcal{J}(x,a) \equiv \frac{C_D/C_{D_0}}{C_L/C_{L_0}}(x,a)\right], \tag{12}$$

where C_{D_0} and C_{L_0} are respectively the drag and lift of the initial configuration, x represents the variable used to optimize the performance of the airfoil, and a represents parameters that define operating conditions such as the Mach number, the angle of attack, etc.

Test case description. This case study focuses on the shape optimization of a realistic 3D aircraft wing in a transonic regime, the state equations are the Euler equations. The nominal operating conditions are characterized by free-flow Mach number of $M_\infty = 0.83$ and an incidence angle of $\alpha = 2$. The initial shape of the wing is depicted in Figure 6.

Fig. 6. The initial shape of the wing.

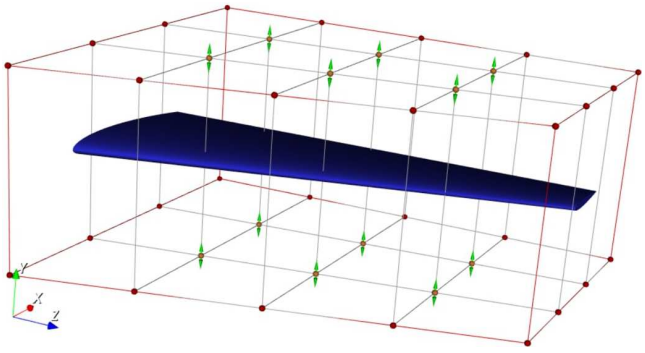

Fig. 7. Parametrization by FFD.

To perform the optimization, the FFD (Free-Form Deformation [14]) technique is employed [15,16], which involves enclosing the wing within a box to reduce its dimensions without altering the wing surface. This dimensional reduction is achieved by considering twelve control point variables (x) as shape variables for the optimization problem. These twelve parameters (x_i) correspond to the displacement of the control points in the FFD parametrization.

The lattice surrounding the initial wing shape is constructed with a margin of $(1500, 3000, 0)$ along the (X, Y, Z) axes, following the wing's shape along the Z axis. This margin allows for the inclusion of a portion of the CFD (Computational Fluid Dynamics) mesh in the deformation zone. The lattice is of order $(3, 1, 4)$, resulting in a total of forty control points. Only the inner points are movable, restricted to the Y direction, with the objective of optimizing the airfoil. This constraint reduces the degrees of freedom to twelve.

Consequently, the FFD parametrization process defines twelve shape parameters (see Figure 7). The displacement of the control points is limited to the range of $[-200, 200]^{12}$ to ensure proper mesh quality.

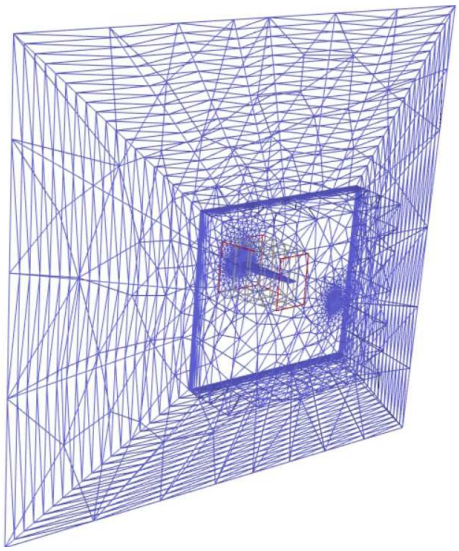

Fig. 8. Unstructured mesh.

An unstructured mesh consisting of 31 124 nodes and 173 445 tetrahedral elements is generated around the wing for spatial discretization. This mesh incorporates a refined area near the shock as illustrated in Figure 8. The flow is simulated by solving the compressible Euler equations using a finite volume method. The tetrahedral elements discretize the flow domain appropriately.

4.2. Reducing order and dimension of airfoil shape through AE-SM

The AE-SM network model, introduced in our study, is designed to address both the order and dimension problems by integrating two main components: an autoencoder (AE) technique and a surrogate model (SM). This combined approach offers an effective solution. To implement the AE-SM network, we utilized the architecture described in section 3.2.

To train these three models, we created a dataset comprising 1000 samples using the quasi-stochastic Latin Hypercube Sampling (LHS) method, as explained in section (2). Each sample consisted of 12 Free-Form Deformation (FFD) parameter variables, with values ranging from -200 to 200. For each data point in the dataset, we conducted an analysis to evaluate the aerodynamic coefficients, specifically the drag (C_D) and lift (C_L). This dataset served as the basis for training the AE-SM models, enabling us to accurately capture the relationships between the FFD parameters and the corresponding aerodynamic coefficients.

The architectures of the encoder and decoder (D-SM) utilized in the AE-SM model are shown in Table 1. We fix the AE-SM parameters, $\lambda_1 = 1$, $\lambda_2 = 1$, and the dimension of the latent variables $d = 6$. The encoder consists of an input layer that concatenates X and Y data, a dense layer with $(14 + 2) \times 10 = 140$ neurons using the swish activation function, and an output layer of size 6. The decoder includes an input layer that accepts patterns of size 6, and two dense layers that reduce the latent space to the size 12 and 2 of the input space. The design of the encoding and decoding layers was based on an empirical approach and prior knowledge of deep learning models.

The architectures of the encoder and decoder (D-SM) utilized in the AE-SM model are shown in Table 1. We fixed the AE-SM parameters, with λ set to 1, and the latent variable Z is limited to the range of $[-1, 1]^6$, where the dimension is set to 6.

Table 1. The AE-SM model architecture includes an Encoder and a Decoder (D-SM).

Layers	Encoder (E)	Decoder (D-SM)
L1	Concatenate [12, 2]	Dense (125 neurons), Swish.
L2	Dense (140 neurons), Swish.	Dense (200 neurons), Swish.
L3	Dense (6 neurons), Linear.	Dense (12 neurons), Linear.
L4		Dense (2 neurons), Linear.

The encoder consists of an input layer that concatenates the X and Y data, followed by a dense layer with $(14 + 2) \times 10 = 140$ neurons using the swish activation function. The output layer of the encoder has a size of 6. The decoder comprises an input layer that accepts patterns of size 6, and two dense layers that reduce the latent space to sizes 12 and 2, respectively, matching the input space. The design of the encoding and decoding layers was based on an empirical approach and prior knowledge of deep learning models.

To assess the accuracy of the trained model, we divided the dataset into 1000 samples for training and the remaining 500 samples for testing the model's generalization capability. Figure 9 shows the error evaluation as a function of the number of epochs (training curve).

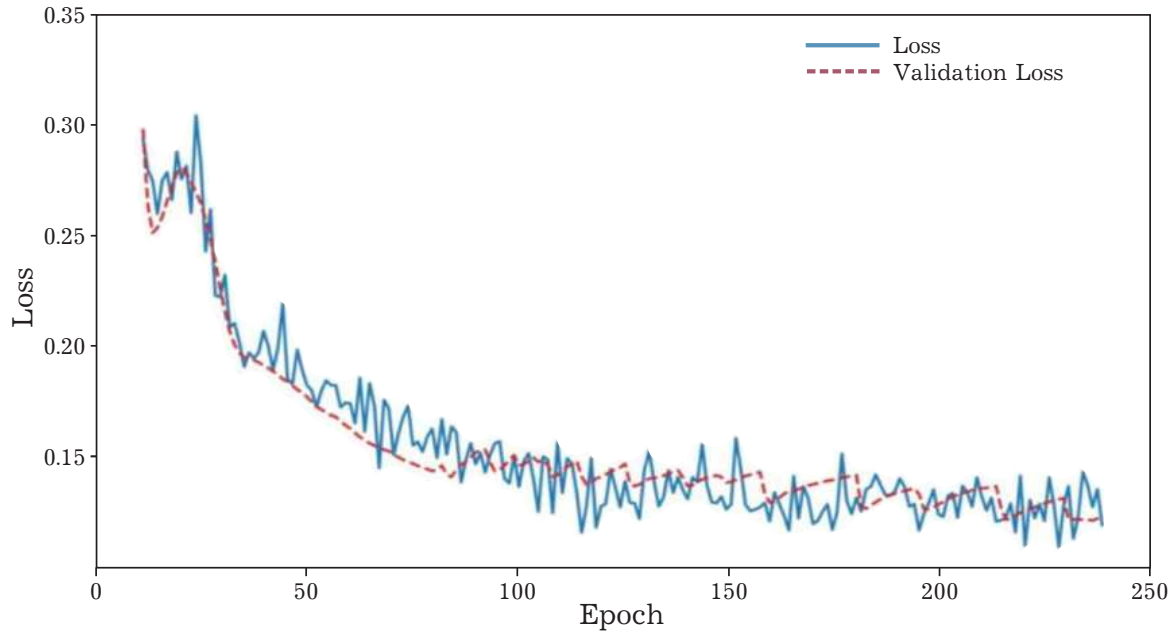

Fig. 9. Learning curve.

Given specific input data, we calculated the Normalized Root Mean Square Error (NRMSE) of the corresponding output pattern as follows:

$$\text{NRMSE} = \frac{\|X - \hat{X}\|}{\|\hat{X}\|} \times 100.$$

The average NRMSE over all the points in the test set was 2.96% for the SM and 10.2% for the AE. This result demonstrates that the model can accurately predict the aerodynamic performance (drag and lift coefficients) for a given profile with an accuracy of 97.04%.

4.3. Airfoil optimization

We present the proposed framework for optimizing airfoil shape in transonic flight regimes. The problem formulation for optimization is given by (13). The objective is to minimize the Drag-to-lift ratio C_D/C_L with respect to the latent variables Z. The flight conditions are defined by Mach number of $M_\infty = 0.83$ and an incidence angle of $\alpha = 2$,

$$\underset{\mathbf{Z}\in[-\mathbf{1},\mathbf{1}]^{\mathbf{6}}}{\text{Minimize}} \frac{C_D/C_{D_0}}{C_L/C_{L_0}}. \tag{13}$$

Once the mathematical formulation of the optimization problem is established (refer to (13)), an efficient optimization algorithm is employed to enhance the aerodynamic performance of a parameterized geometry under one or more crucial design conditions. There are primarily two categories of optimization methods. The first category consists of local optimization methods, such as the descent method, which iteratively optimizes the objective functions and constraints using gradients. This ap-

proach is widely utilized due to its rapid convergence. The second category encompasses gradient-free optimization methods that rely solely on the objective function values, without utilizing derivatives. Examples include the differential evolution (DE) algorithms, the particle swarm optimization (PSO) algorithms [17], and the NSGA-II algorithm.

In this particular case, the PSO algorithm is utilized to solve the problem defined in (13). The convergence curve of the PSO algorithm is shown in Figure 10.

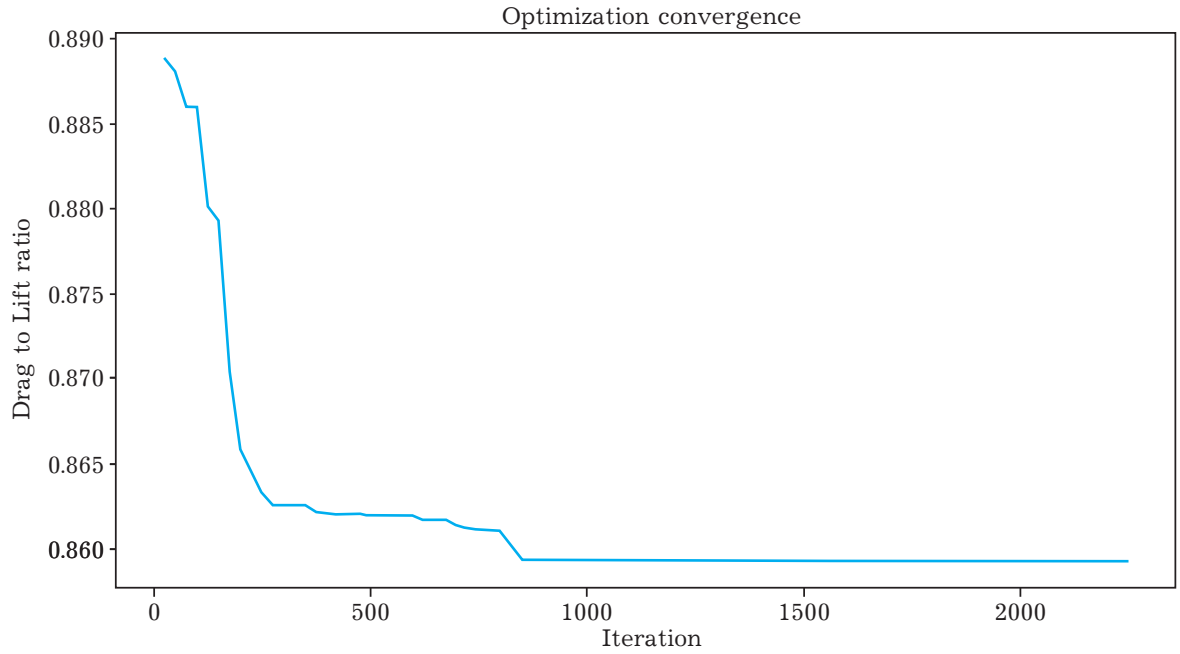

Fig. 10. Convergence curve of the PSO algorithm.

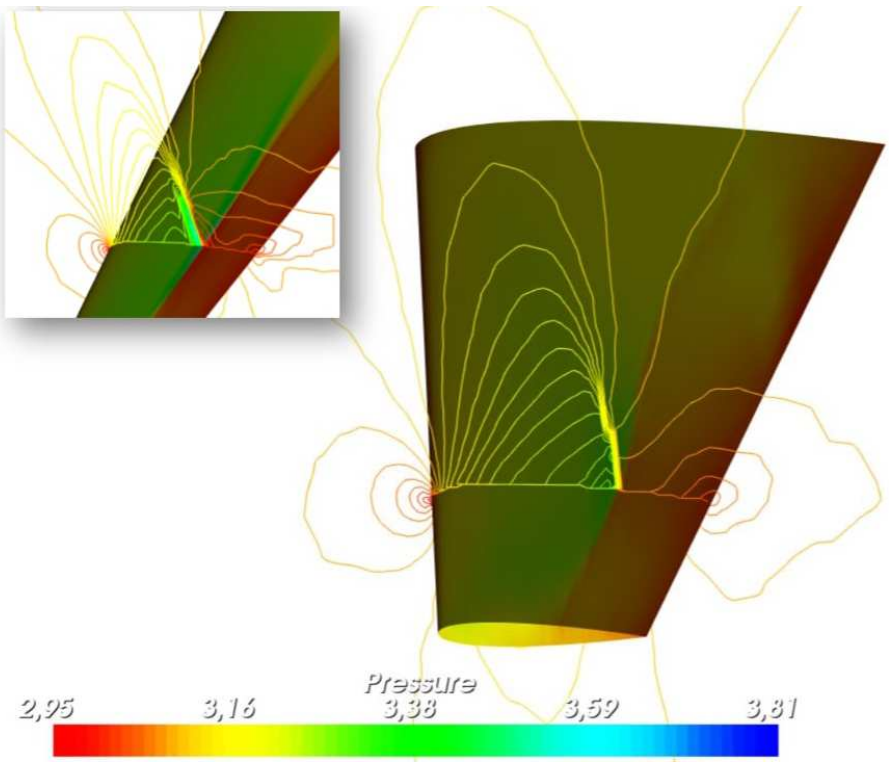

Fig. 11. Pressure field on the surface of the wing and isobars, of the optimal wing compared to the initial wing in the top-left corner.

In Figure 11, we present the isobaric pressure contours on the wing surfaces for both the initial shape and the optimized shape. It is evident that the flow exhibits greater smoothness in the left figure as opposed to the initial shape, which displays higher pressure gradients. This observation implies that the optimization process has successfully mitigated the intensity of the shock wave.

5. Conclusion

In this work, we demonstrated how unsupervised learning, specifically AutoEncoders, can be utilized to simultaneously construct a surrogate model and reduce the dimensionality of a design problem. The resulting model serves as a reliable predictor of both the design variables and their corresponding coefficients, as evidenced by our study on the shape FFD parameters, Lift and Drag coefficients. Moreover, we showcased the effectiveness of this model in facilitating rapid and efficient shape optimization. The AutoEncoder architecture employed in this study is the most commonly used and widely adopted. In future research, we plan to experiment with and compare alternative architectures, such as a sequential layout of the encoder, decoder, and surrogate model, or a parallel arrangement of the decoder and the surrogate network following an encoder.

Acknowledgments

The entire authors of this research are utterly grateful to the Arab Fund for Economic and Social Development (AFESD) for financial support for this work.

[1] Lyu Z., Kenway G. K. W., Martins J. R. R. A. Aerodynamic shape optimization investigations of the Common Research Model wing benchmark. AIAA Journal. **53** (4), 968–985 (2015).

[2] Wu X., Zhang W., Song S. Robust aerodynamic shape design based on an adaptive stochastic optimization framework. Structural and Multidisciplinary Optimization. **57** (3), 639–651 (2018).

[3] Han Z.-H., Görtz S. Hierarchical Kriging Model for Variable-Fidelity Surrogate Modeling. AIAA Journal. **50** (9), 1885–1896 (2012).

[4] Boutemedjet A., Samardžić M., Rebhi L., Rajić Z., Mouada T. UAV aerodynamic design involving genetic algorithm and artificial neural network for wing preliminary computation. Aerospace Science and Technology. **84**, 464–483 (2019).

[5] Liao P., Song W., Du P., Zhao H. Multi-fidelity convolutional neural network surrogate model for aerodynamic optimization based on transfer learning. Physics of Fluids. **33** (12), 127121 (2021).

[6] Tao J., Sun G., Guo L, Wang X. Application of a PCA-DBN-based surrogate model to robust aerodynamic design optimization. Chinese Journal of Aeronautics. **33** (6), 1573–1588 (2020).

[7] Kou J., Botero-Bolivar L., Ballano R., Marino O„ de Santana L., Valero E., Ferrer E. Aeroacoustic airfoil shape optimization enhanced by autoencoders. Expert Systems with Applications. **217**, 119513 (2023).

[8] Deng K., Chen H., Zhang Y. Flow structure oriented optimization aided by deep neural network. 10th International Conference on Computational Fluid Dynamics. ICCFD10-289 (2018).

[9] Wu H., Liu X., An W., Chen S., Lyu H. A deep learning approach for efficiently and accurately evaluating the flow field of supercritical airfoils. Computers & Fluids. **198**, 104393 (2020).

[10] Moussaoui Z., Karafi Y., Abou El Majd B. Robust shape optimization using artificial neural networks based surrogate modeling for an aircraft wing. Mathematical Modeling and Computing. **11** (1), 139–153 (2023).

[11] Du X., He P., Martins J. R. R. A. Rapid airfoil design optimization via neural networks-based parameterization and surrogate modeling. Aerospace Science and Technology. **113**, 106701 (2021).

[12] Moussaoui Z., El Bakkali H., Karafi Y., Abou El Majd B. Bayesian Approach for Aerodynamic Shape Robust Optimization. Proceedings of the IISE Annual Conference. Preprint (2023).

[13] Coppedè A., Gaggero S., Vernengo G., Villa D. Hydrodynamic shape optimization by high fidelity CFD solver and Gaussian process-based response surface method. Applied Ocean Research. **90**, 101841 (2019).

[14] Sederberg T., Parry S. Free-Form Deformation of Solid Geometric Models. ACM SIGGRAPH Computer Graphics. **20** (4), 151–160 (1986).

[15] Abou El Majd B., Désidéri J.-A., Janka A. Nested and selfadaptive Bézier parameterizations for shape optimization. International Conference on Control, Partial Differential Equations and Scientific Computing (dedicated to late Professor J. L. Lions), Beijing, China, 13–16, September 2004.

[16] Abou El Majd B. Parameterization adaption for 3D shape optimization in aerodynamics. International Journal of Science and Engineering. **6** (1), 61–69 (2014).

[17] Eberhart R., Kennedy J. A new optimizer using particle swarm theory. MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science. 39–43 (1995).

Одночасне сурогатне моделювання та зменшення вимірності за допомогою неконтрольованого навчання. Застосування до параметричної оптимізації форми крила

Карафі Ю.[1], Муссауї З.[1], Абу Ель Маджд Б.[1,2]

[1] *Лабораторія LMSA, факультет природничих наук, Університет Мохаммеда V у Рабаті, Марокко*
[2] *Університет Лілля, 59655 Вільньов-д'Аск, Франція*

У статті представлено підхід, який заснований на машинному навчанні, що забезпечує одночасне сурогатне моделювання та зменшення вимірності та застосовує його для оптимізації аеродинамічної параметричної форми. Оптимізація аеродинамічної форми є вирішальним процесом у різних галузях промисловості, включаючи аерокосмічну, автомобільну та відновлювану енергетику. Це передбачає ітераційне покращення властивостей системи шляхом оцінки цільової функції та її мінімізації або максимізації за допомогою алгоритму оптимізації. Однак оцінка аеродинамічних цільових функцій вимагає виконання обчислювальних затратних операцій, таких як розв'язування складних рівнянь гідродинаміки та обчислення показників ефективності, таких як коефіцієнти підйомної сили та лобового опору. Ця обчислювальна вартість стає особливо обтяжливою, коли алгоритми оптимізації без похідних повинні оцінювати численні вибірки за ітерацію. Крім того, коли вимірність простору проєктування велика, ефективність і результативність процесу оптимізації знижуються. Щоб вирішити ці проблеми, у статті пропонується поєднати сурогатне моделювання та зменшення вимірності. Сурогатне моделювання створює модель зменшеного порядку, яка наближає шукані коефіцієнти економічно ефективним способом, тоді як зменшення вимірності визначає найбільш важливі виміри проєктного простору за допомогою таких методів, як правильна ортогональна декомпозиція. У статті пропонується інтегративний підхід, який використовує штучні нейронні мережі (ANN) і неконтрольоване навчання, зокрема мережі AutoEncoder, щоб одночасно побудувати сурогатну модель і зменшити вимір задачі. Ця методика застосовується для оптимізації форми аерокрила літака в умовах трансзвукового польоту. Форма крила параметризована за допомогою деформації вільної форми (FFD). У статті показано, що запропонований підхід дозволяє швидко та ефективно оптимізувати форму.

Ключові слова: *машинне навчання; AutoEncoder; деформація вільної форми; штучні нейронні мережі; оптимізація форми; аеродинамічний аналіз.*