Comparison of multi-fidelity surrogate models for multi-objective aerodynamic optimization in turbomachinery under extreme cost imbalance

Marc Schouler, Anca Belme, Paola Cinnella

Schouler *et al.*
Advanced Modeling and Simulation in Engineering Sciences (2025) 12:35
https://doi.org/10.1186/s40323-025-00316-3

Advanced Modeling and Simulation in Engineering Sciences

RESEARCH Open Access

Comparison of multi-fidelity surrogate models for multi-objective aerodynamic optimization in turbomachinery under extreme cost imbalance

Marc Schouler[1*], Anca Belme[1†] and Paola Cinnella[1†]

†Anca Belme and Paola Cinnella contributed equally to this work.

*Correspondence:
Marc Schouler
marc.schouler@sorbonne-universite.fr
[1]Institut Jean Le Rond D'Alembert, Sorbonne University, CNRS, 4 Place Jussieu, Paris 75005, France

Abstract

Aerodynamic shape optimization of next-generation aeronautic components faces challenges related to robustness and scalability. Although switching to reliable, albeit expensive, high-fidelity flow models is essential for expanding the design space, the number of high-fidelity simulations that can be performed within the optimization loop is severely limited by computational budget constraints. Furthermore, optimization must handle multiple competing objectives and handle complex constraints in contexts where gradient information is difficult or impossible to retrieve. To address these challenges, we investigate gradient-free, multi-objective constrained optimization strategies based on multi-fidelity surrogate models. In particular, we focus on cases of extreme computational cost imbalance between high- and low-fidelity models, where optimization is driven by a very small number ($O(10)$) of high-fidelity simulations. In order to maximize the information extracted from the high-fidelity samples, we first generate a reduced representation of the design space. Next, we consider adaptive infill strategies for actively learning the surrogate using high-fidelity samples that best guide the optimization. Two strategies are proposed and compared: multi-objective constrained Bayesian optimization assisted by a co-kriging surrogate, and a genetic algorithm guided by a multi-fidelity neural network and active learning. The two approaches are evaluated using analytical benchmarks and a realistic use case involving a low-Reynolds linear outlet guide vane cascade, of interest for aeronautical engines. Coarse-mesh RANS simulations are used as the low-fidelity model while RANS simulations with a transition model and automatic mesh adaptation are selected as the high-fidelity ones. The efficacy of the two strategies is measured using various metrics. For the problem under consideration, it has been found that for a small number of high-fidelity samples and after dimension reduction, the Bayesian optimization strategy is more efficient. In the non-reduced design space however, both strategies yield similar performances.

Keywords Aerodynamic Shape Optimization, Multi-fidelity Surrogate Modeling, Adaptive Infill, Data Reduction, RANS Simulation, Feature-based Mesh Adaptation

Introduction

Aerodynamic shape optimization (ASO) plays a central role in the design of next-generation aircraft and turbomachinery components, with growing importance in addressing sustainability and performance objectives [1, 2]. As aerodynamic designs grow more complex and environmental constraints tighten, a shift toward high-fidelity, scale-resolving flow models –such as Large Eddy Simulation (LES)– becomes necessary to capture key physical mechanisms and unravel breakthrough designs. However, the high computational cost of these models, often orders of magnitude higher than traditional steady RANS, poses significant challenges for their integration into the optimization loop. In the context of this work, aerodynamic shape optimization (ASO) refers to techniques aimed at enhancing aerodynamic performance by deforming shape parameters that define external surfaces [3] or aeronautical engine components, such as turbomachinery [4]. Optimization methods in ASO are typically classified into two major families: gradient-based and gradient-free approaches. Gradient-based methods rely on the solution of an adjoint problem to efficiently compute sensitivities of objective and constraint functions with respect to the design variables [5]. These methods offer rapid convergence in high-dimensional design spaces and are widely used when gradient information is available and reliable. However, their practical deployment in industrial CFD contexts remains limited due to the challenges associated with differentiating complex legacy codes, particularly in the presence of turbulence models. While automatic differentiation (AD) offers a general solution in principle, its integration into large-scale CFD solvers is still limited due to memory overhead, performance loss, and the difficulty of handling discontinuous or non-smooth operators [6, 7]. Moreover, in turbulence-resolving simulations gradients tend to become highly irregular or even chaotic, rendering them unreliable for optimization. This issue arises due to the high sensitivity of the resolved turbulent structures to small perturbations in the design space. As a result, the meaningful use of gradient information in scale-resolving flows remains an open research question [8–10]. For this reason, gradient-free methods—such as evolutionary algorithms and particle swarm optimization—are often preferred when dealing with complex physical models, noisy objectives, multiple conflicting goals, or difficult constraints [11–13]. However, they typically require a far larger number of function evaluations to converge [13], which makes their use with expensive solvers inacceptably costly. In the following, we focus on approaches that significantly speed up gradient-free methods coupled with costly flow solvers.

While gradient-free ASO has benefited from the progress in computational fluid dynamics (CFD) and high-performance computing, several bottlenecks persist, specifically related to geometry parametrization, mesh quality under geometrical changes, and the solution robustness and accuracy under large mesh deformations [14]. One major strategy for reducing the cost of repeated CFD evaluations is the use of surrogate models (e.g., [15, 16]), which approximate the expensive cost functions with cheaper models learned from a set of training samples. However, surrogate models suffer from the "curse of dimensionality," requiring a large number of samples to achieve accuracy in high-dimensional design spaces. This can be prohibitive when costly models are used to drive the design. Multi-fidelity surrogates [17–19] offer a partial remedy by combining information from both low-cost, low-accuracy models and expensive high-fidelity simulations. Furthermore, adaptive sampling/active learning strategies can further increase

data efficiency by placing new samples where they will be most valuable for improving the surrogate model [20, 21]. In most ASO applications, the fidelity gap is usually moderate, often involving different mesh resolutions or simplified physical models, with a cost ratio between low- and high-fidelity evaluations of about 10 to 100 [22, 23]. In such cases, optimization frameworks can afford dozens or even hundreds of high-fidelity samples, and infill strategies can dynamically determine the appropriate fidelity level for each new sample [24, 25]. However, in regimes involving unsteady RANS [26] or large-eddy simulations (LES) [27, 28], the cost imbalance becomes extreme, with high-fidelity evaluations being 1000 to 10,000 times more expensive. In these settings, the total number of usable high-fidelity evaluations can be as low as 10–20. This regime demands careful pre-allocation of high-fidelity queries and aggressive strategies for maximizing the information extracted from each.

Moreover, the large number of function calls required by gradient-free methods may restrict the fidelity of the CFD solvers used, leading to significant numerical errors and poor convergence [29]. Even when starting from a well-converged baseline, shape deformations may introduce remeshing errors that degrade solution accuracy. In this context, automatic mesh adaptation [30] offers an attractive alternative by locally refining the mesh based on solution features. However, it remains underutilized in shape optimization workflows [31–33].

This study develops and compares two multi-fidelity, surrogate-assisted optimization strategies designed for scenarios in which only a limited number of high-fidelity samples can be computed. Although our ultimate objective is to apply the framework to very costly LES, in this study we deliberately restrict ourselves to a realistic yet tractable aerodynamic use case for which a reference high-fidelity solution can be obtained using a brute force approach. Specifically, we consider the design of a low-Reynolds number outlet guide vane (OGV) cascade, which is representative of small-scale turbomachinery. The low-fidelity model uses coarse-mesh RANS simulations, while the high-fidelity model employs a transition-sensitive turbulence model with feature-based anisotropic mesh adaptation [34] to ensure numerical accuracy. Specifically, we evaluate:

1. A Bayesian optimization strategy using co-kriging surrogate models and uncertainty-driven adaptive infill;
2. A multi-objective genetic algorithm assisted by a multi-fidelity deep neural network trained through active learning.

The main contributions of this paper are summarized below:

1) A comparison of multi-fidelity optimization strategies combining gradient-free methods with co-kriging and deep learning surrogates in a multi-objective turbomachinery design problem.
2) The integration of dimensionality reduction and active learning to enable surrogate training from very few high-fidelity samples.
3) The use of mesh adaptation to ensure robust and accurate high-fidelity simulations across deformed geometries.

The paper is organized as follows: the remaining of the introduction presents related work and background on multi-fidelity surrogate modeling. The optimization strategies and surrogate construction methods, including dimensionality reduction, are detailed in

the Methodology section. The application of the proposed approaches to the OGV cascade is presented in the Results and Discussion section. Final remarks and perspectives are discussed in Conclusion.

State of the art of ASO practices

Multi-fidelity surrogate assisted optimization

The use of surrogate models of the objective functions to replace expensive CFD solvers in gradient-free optimization has become a popular practice [13]. Furthermore, in response to the hierarchical fidelity and cost of CFD solvers in terms of model or mesh refinement levels, there has been an increasing focus on multi-fidelity surrogate models [35]. Among them, the co-kriging method (AR1), based on Kennedy and O'Hagan linear autoregressive information fusion scheme [36] is by far the most commonly used in engineering design [18]. This approach has been extended to non-linearly correlated low- and high-fidelity sources by Perdikaris et al. [37], who have proposed the non-linear auto-regressive multi-fidelity Gaussian processes (NARGP) model. More recently, deep learning based multi-fidelity methods have been introduced. Examples include multi-fidelity deep neural networks (MFDNN) [38], multi-fidelity deep Bayesian neural networks [39], multi-fidelity deep Gaussian processes [40], multi-fidelity deep neural operators [41] and multi-fidelity reinforcement learning [42]. Multi-fidelity surrogate models for optimization are reviewed in [35]. An examination of the number of publications for different multi-fidelity surrogate methods applied to engineering design and optimization shows that correction-based methods, which use additive and/or multiplicative correction terms to model the relationship between successive fidelity levels, are the most prevalent (39.1%), followed by AR1 methods (20.7%) and single models[1] (17.2%). Of course, newer methods such as non-linear auto-regressive Gaussian process (2.3%) or deep Gaussian processes (3.5%) have been used to a lesser extent. In addition, multi-fidelity surrogates under severe computational budget constraints for the high-fidelity model have received little attention in the literature. The question of whether newer models (theoretically capable of learning any kind of low-/high-fidelity correlation) should be prioritized over simpler models is still open.

Adaptive infill

Multi-fidelity surrogates are generally trained with a reduced number of samples, especially in the high-fidelity level, in order to reduce the computational cost of generating the training data. However, their accuracy may not be sufficient to guide the optimization towards the optimum of the actual objective function. To remedy this limitation, the model can be adaptively retrained several times during the optimization by carefully selecting additional samples that are likely to improve the model accuracy in the region of interest. This process is known as adaptive infill or adaptive sampling and is a topic of active research, especially in the context of multi-fidelity optimization [22–25, 43].

Table 1 summarizes the main characteristics of recent multi-fidelity optimization strategies with adaptive infill. When the surrogate models are of Bayesian nature, Bayesian acquisition functions can be used to select the next infill locations based on the current state of the surrogate model. This paradigm is known as Bayesian Optimization

[1] The single model approach consists of using independent surrogate models for each fidelity without assuming any kind of correlation between them.

Table 1 Main characteristics of multi-fidelity optimization methods with various adaptive infill strategies

Ref.	Problem	DOEs		Total infills		Infill	Cost
	dimension	LF	HF	LF	HF	strategy	ratio
Multi-fidelity single-objective optimization with MFDNN							
Zhang et al. [22]	30	120	60	120	60	PSO⋆& ED	7
Wu et al. [23]	12	160	20	1280	160	TLBO⋆& LHS	-
Multi-fidelity multi-objective BO with AR1							
He et al. [43]	144	-	720	-	150	GEIM+	10
Mourousias et al. [25]	30	2000	274	9		VFEIM+	30
Charayron et al. [24]	20	20	10	10		MPI+	116
Matar et al. [28]	4	41	5	15	2	PI & LCB	30 000

The "Total infills" column indicates the total number of infill samples for each level of fidelity. In the infill strategy column, the "★" symbol indicates that the high-fidelity sample is selected as the optimization algorithm best candidate. The "+" symbol indicates a variant of an existing strategy

(BO). The multi-fidelity multi-objective version is referred to as MF-MOBO [24, 25, 43]. Multi-objective acquisition functions mostly consist in generalizations of their single-fidelity equivalents. For instance the generalized expected improvement matrix (GEIM), also called variable fidelity expected improvement matrix (VFEIM), is an extension of the well-known expected improvement (EI) that can be used to derive multi-objective infill criteria [44, 45]. Similarly, the probability of improvement (PI) has been extended to multi-objective adaptive sampling in [20] and as the minimum probability of improvement (MPI) criterion in [46]. For non-Bayesian models, the alternative is to combine the surrogate model with an optimization algorithm from which one or multiple best-fit candidates are selected as infill samples. These are then computed with the high-fidelity solver while another strategy is used to sample low-fidelity infill samples. In [22], the particle swarm optimization (PSO) algorithm is used to sample high-fidelity infills and the max-min Euclidean distance (ED) criterion is used to sample additional low-fidelity infills. In [23], the teaching-learning-based optimization (TLBO) algorithm [47] and Latin hypercube sampling (LHS) are used to sample high- and low-fidelity infills, respectively. For both approaches, the surrogate is updated after each infill step and the procedure continues until it converges to an optimal solution that is in close agreement with the solver results.

Although [23] gives an ASO example where MFDNN outperformed single-fidelity kriging, both models were combined with a non-Bayesian infill strategy. Therefore, it is difficult to know whether this conclusion would hold if the kriging model was used in its multi-fidelity form and combined with a Bayesian infill strategy.

Given the promising results recently obtained with the MFDNN model [22, 23] and the enduring popularity of the AR1 model despite its seniority, we selected these two methods for further comparison in the context of multi-fidelity ASO with adaptive infill under a limited number of high-fidelity samples.

Shape parametrization and data reduction

In ASO, the shape under study is typically defined by a discretization of hundreds or thousands of surface points. These, in turn, are parametrized by tens (in 2D) or hundreds (in 3D) of design variables that form the problem dimension. Shape parametrization methods are reviewed in [3], which divides them into constructive and deformative models. Constructive models include various kinds of spline methods such as Bezier

spline, basis spline and non-uniform rational basis spline. On the other hand, deformative models include free-form deformation (FFD) [48], class shape transformation [49] and Hicks-Henne bump function [50]. As explained in [3], the simplest methods for CFD-based optimization are deformative methods, and FFD is particularly suitable for high-fidelity optimization where the baseline geometry is already sufficiently close to the optimal solution. It may be noted that in addition to these approaches, deep learning based methods such as DeepGeo [51], Bezier-GAN [52] or variational autoencoders [53] enable the exploration of wide design spaces using only a few parameters and have shown great promise in conditions without prior knowledge of the optimal shape.

Since the optimization convergence speed and the minimum amount of data required to train surrogate models increase with the problem dimensions, several data reduction methods have been proposed to reduce the parametrization size. Active subspace techniques use gradient evaluations to identify the most influential parameters of multivariate functions, where the principal directions of variability are not aligned with the coordinates of the input space [54]. Both linear (e.g. POD) and non-linear (e.g. autoencoders and generative adversarial networks) gradient-free reduction techniques applied to the shape parametrization have since become common practice [13], and this remains an active field of research [55–58]. In [57, 58], principal component analysis (also sometimes referred to as proper orthogonal decomposition, POD, or singular value decomposition, SVD) is shown to simultaneously reduce the number of necessary samples to train surrogate models and to improve their modeling accuracy. A similar approach has recently been considered in [27, 28]. For all these reasons, an approach combining FFD and POD is adopted herein.

Discretization error control

When it comes to CFD-based optimization, the challenge of mesh management is paramount, as the quality of the function objective evaluation depends on the quality of the discretisation used to compute it. Poor choices in mesh resolution and numerical solvers can lead to errors that propagate through the surrogate model [59], ultimately resulting in an incorrect optimal design. Indeed, Cinnella and Congedo [29] showed for a transonic airfoil single optimization problem that the numerical errors in the representation of the objective function makes the genetic algorithm convergence harder. While optimizations on fine grids or the use of Richardson extrapolation can mitigate the problems, none of them guarantees optimal mesh quality for all the designs explored during the optimization.

A more rigorous way of controlling numerical errors and ensuring mesh quality is to use automatic mesh adaptation techniques. There are two kinds of mesh adaptation method: feature-based mesh adaptation, which aims to provide the best mesh to compute the characteristics of a given sensor (e.g. Mach field), and the adjoint-based (or goal-oriented) mesh adaptation, which aims to provide the best mesh to observe a given scalar quantity of interest (e.g. drag coefficient) [30, 34, 60]. Despite their successful application to both external [34] and internal [61] flow computations, mesh adaptation tools are still not widely available in CFD codes, and they are rarely used in ASO. For instance, John et al. [33] showed the benefit of feature-based mesh adaptation for the aerodynamic design of turbomachinery configurations with fully structured block

meshes and single objective adjoint-based optimization. Chen et al. investigated single [62] and multi point [32] ASO with adjoint-based mesh adaptation. In both papers, particular attention is paid to avoiding over-refinement of undesirable designs according to the estimated error in the objective function, thus avoiding unnecessary computations. An attractive alternative to reduce the computational costs is to combine mesh adaptation with multi-fidelity optimization. This was recently done by Wackers et al. [63, 64], who showed the effectiveness of combining multi-fidelity radial basis functions with mesh adaptation and adaptive infills for the single-objective optimization of hydrodynamic shapes parametrized with only one or two design variables. However, to the authors' knowledge, such a strategy has not been applied to multi-objective ASO in high-dimensional spaces.

Methodology

In the following we consider two multi-fidelity methodologies for solving multi-objective optimization problems. In particular, we focus on two-objective problems. Given a design space $\Omega \subset \mathbb{R}^d$, with d the number of design variables, and n_{obj} objective functions J_k with $k \in [\![1, n_{\text{obj}}]\!]$, our goal is to find a set *PS* of Pareto-optimal individuals, i.e. designs x^* satisfying the Pareto dominance conditions [65]:

$$\forall x \in \Omega \begin{cases} \forall k \in [\![1, n_{\text{obj}}]\!],\ J_k(x^*) \leq J_k(x), \\ \exists j \in [\![1, n_{\text{obj}}]\!],\ J_j(x^*) < J_j(x). \end{cases} \tag{1}$$

In order to speed-up the optimization, in the following we seek for surrogate models that approximate the input/output relationships $J_k : x \mapsto J_k(x)$ using information from a set of low-fidelity and high-fidelity evaluations of the objectives. Specifically, our goal is to obtain the best possible approximation of *PS* using the fewest possible expensive high-fidelity function calls.

Multi-fidelity surrogate-based optimization

In the present section we introduce two multi-objective surrogate-based optimization strategies with adaptive infills. The first one relies on a AR1 multi-fidelity surrogate coupled with Bayesian optimization whereas the second one uses a multi-fidelity neural network combined with a genetic algorithm. The high level structure of the framework[2] is illustrated in Fig. 1 and each component (optimization algorithm, surrogate assistance, CFD fidelity, adaptive infill, data reduction) is configurable such that different optimization approaches can be compared.

Bayesian multi-fidelity optimization

The first approach under consideration combines a multi-fidelity Gaussian process, and a Bayesian optimization algorithm with adaptive infills.

In its bi-fidelity form, the co-kriging model (AR1) which relies on Kennedy and O'Hagan linear autoregressive fusion scheme [36] relates a high-fidelity model $f_{hf}(\cdot)$ to a low-fidelity model $f_{lf}(\cdot)$ through the expression:

$$f_{hf}(x) = \rho\, f_{lf}(x) + \delta(x). \tag{2}$$

[2] The framework is hosted on github (see \url{https://mschouler.github.io/aero-optim/})

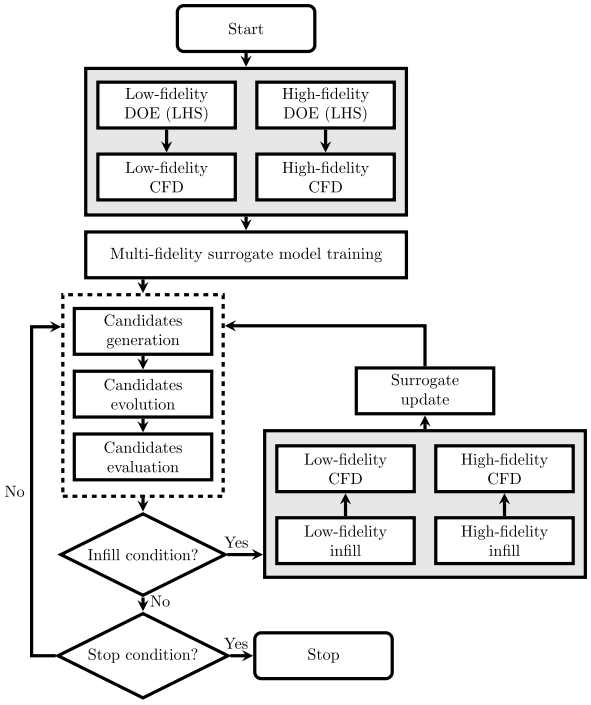

Fig. 1 Workflow of the multi-fidelity surrogate assisted optimization framework. The dashed box represents the optimization algorithm that may be skipped in case of Bayesian optimization. The gray boxes represent multi-step processes including candidate sampling, geometry deformation, mesh generation and low- or high-fidelity CFD simulations

The scaling factor ρ may depend on x but is usually assumed to be constant, while the additive bias $\delta(\cdot)$ is a Gaussian process that models the difference between the two fidelities. In this work, we use the AR1 implementation of the *smt* package[3] [66], which follows the recursive formulation of [17]. Such algorithm relies on a nested design of experiments (DOE) for the two levels of fidelity, i.e. high- and low-fidelity samples must be evaluated at the same locations in the parameter space. The nested structure enables the surrogate variance to be expressed in closed form. A benefit of this approach is that when dealing with L levels of fidelity, the predictive mean $\mu_l(\cdot)$ and variance $\sigma_l^2(\cdot)$ of each fidelity level $l \in [\![2; L]\!]$ can be expressed recursively as follows [24]:

$$\mu_l(x) = \rho_{l-1}\,\mu_{l-1}(x) + \mu_{\delta_l}(x),$$
$$\sigma_l^2 = \rho_{l-1}^2\,\sigma_{l-1}^2 + \sigma_{\delta_l}^2,$$

(with μ_{δ_l} and $\sigma_{\delta_l}^2$ the mean and variance of the bias term, respectively), so that each fidelity level can be trained separately.

With AR1 models, a separate surrogate model is trained for each objective function. Then, Bayesian optimization requires the definition of a suitable acquisition function that updates the best-performing designs using information from the surrogates. Hereafter, an optimization strategy is devised by mixing ideas from the last two approaches in Table 1 and described in Algorithm 1.

[3] *smt* is an open source Python library hosted on github (see https://smt.readthedocs.io/en/latest/).

Algorithm 1 Bayesian adaptive infill optimization

Compute low- and high-fidelity initial DOEs
Train AR1 models μ_1 and μ_2
for $i = 1$ to n_{iter} **do**
  $x_{hf} = \underset{x}{\text{argmax}}\left\{\alpha_{WB2S,MPI}(x)\right\}$
  $x_{lf} = x_{hf} \bigcup \underset{x}{\text{argmin}}\left\{LCB_1(x)\right\} \bigcup \underset{x}{\text{argmin}}\left\{LCB_2(x)\right\}$
  for $j = 1$ to n_{lf} **do**
    $X_{lf} = X_{lf} \bigcup x_{lf}$
    $x_{lf} = x_{lf} \bigcup \underset{x}{\text{argmax}}\left\{\min\, d(x, X_{lf})\right\}$
  end for
  Compute y_{hf} and y_{lf}
  Re-train AR1 models μ_1 and μ_2
end for
Run NSGA-II with μ_1 and μ_2

First, the low- and high-fidelity initial datasets are generated and AR1 models are trained. Then, at each iteration of the optimization, a high-fidelity infill sample is selected based on a regularized infill criterion. As explained in [67], regularization techniques helps to circumvent the ill-posedness of infill problems. Specifically, given any multi-objective acquisition function $\alpha_f(\cdot)$, the next infill sample is obtained by maximizing:

$$\alpha_{WB2S,f}(x) = \gamma\, \alpha_f(x) - \psi(\mu_f(x)). \tag{3}$$

where $\alpha_{WB2S,f}$ is a regularized acquisition function, γ is a real parameter, $\mu_f(\cdot)$ is the collection of the Gaussian processes for each objective and $\psi(\cdot)$ is a so-called scalarization operator (e.g. max or sum). Similar to what was done in [24], we chose ψ to be the sum function, while the initial acquisition function α_f is taken as the minimum probability of improvement (MPI) [46], defined as:

$$MPI(x) = \min_{u \in PS} \left(1 - \prod_{k=1}^{n_{obj}} \Phi\left(\frac{\mu_k(x) - \hat{y}_k(u)}{\sigma_k(x)}\right)\right). \tag{4}$$

In the preceding expression, $\Phi(\cdot)$ is the normal cumulative distribution function while $\mu_k(\cdot)$ and $\sigma_k(\cdot)$ denote the predictive mean and standard deviation of objective *k*. The true high-fidelity response for objective *k* is denoted $\hat{y}_k(\cdot)$ and $u \in PS$ form the current high-fidelity Pareto set.

In order to satisfy the nested dataset property, the infill point selected by the acquisition function is computed using both the high-fidelity and the low-fidelity model. In addition, two extra low-fidelity infill samples are selected by minimizing the lower confidence bound (LCB) of each objective as suggested in [27, 28]:

$$LCB_k(x) = \mu_k(x) - \sigma_k(x),\ k \in [\![1, n_{\text{obj}}]\!]. \tag{5}$$

To favor diversity, additional low-fidelity candidates are selected by maximizing the minimal Euclidean distance from [22]:

$$\max \min d(x, X_{lf}), \tag{6}$$

where $d(\cdot, \cdot)$ denotes the Euclidean distance between the new point *x* and the current low-fidelity DOE X_{lf}. At each infill step, all three optimization sub-problems resulting

from Equation 4, 5 and 6 are solved with particle swarm optimization. At that point, the surrogate models of each objective are updated and the procedure is repeated until the specified number of iterations based on the computational budget is reached. Lastly, an extra post-processing step is performed by running the NSGA-II genetic algorithm [68] with the final version of the surrogate models. This has for objective to provide a denser estimate of the Pareto front [24]. In the following, it will be referenced to as the predicted Pareto.

Neural-network assisted multi-fidelity optimization

The second considered strategy combines a multi-fidelity deep neural network (MFDNN) with a multi-objective genetic algorithm.

The MFDNN model was initially introduced by Meng et al. [38]. In contrast with the linear correlation used in AR1 models, it relies on the combination of a linear neural network $\mathcal{F}_l(\cdot)$ and a non-linear neural network $\mathcal{F}_{nl}(\cdot)$ according to the following equation:

$$f_{hf}(x) = \beta\mathcal{F}_l(x, y_L) + (1-\beta)\mathcal{F}_{nl}(x, y_L), \tag{7}$$

where the correlation coefficient $\beta \in [0; 1]$ can be learned [22] or fixed [23]. The network architecture is illustrated in Fig. 2, where the low-fidelity model is approximated with a neural network $NN_L(x_L, \theta_0)$, combined with two other networks $NN_{H_1}(x, y_L, \theta_1)$, $NN_{H_2}(x, y_L, \theta_2)$ that are used to approximate the correlation between the low- and high-fidelity data. The parameters θ_i for each network are trained by minimizing the loss function:

$$\begin{aligned}
\mathcal{L}oss &= \mathcal{L}oss_{y_L} + \mathcal{L}oss_{y_H} + \lambda \sum \theta_i^2, \\
\mathcal{L}oss_{y_L} &= \frac{1}{N_{y_L}} \sum_{i=1}^{N_{y_L}} \left(|\hat{y}_L - y_L|^2 \right) \\
\mathcal{L}oss_{y_H} &= \frac{1}{N_{y_H}} \sum_{i=1}^{N_{y_H}} \left(|\hat{y}_H - y_H|^2 \right),
\end{aligned} \tag{8}$$

where $\hat{y}_L$ and $\hat{y}_H$ are the true low- and high-fidelity values, while y_L and y_H are the predicted low- and high-fidelity values. Although not required by the formulation of MFDNN models, using nested DOEs to construct the training sets was suggested in

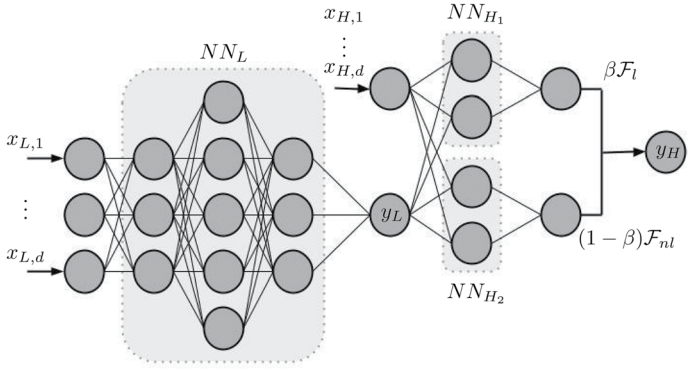

Fig. 2 MFDNN architecture reproduced from [23]

[23]. The effect of this choice is therefore investigated in the next section. The MFDNN was implemented using the *Pytorch*[4] library.

Since the MFDNN surrogate is deterministic, a non-Bayesian optimization strategy is selected in conjunction with this model, described in Algorithm 2. Unlike AR1 models, MFDNNs can easily be generalized to any number of objectives by extending the last layer of each of its sub-networks. As a drawback, no Bayesian consideration can be used to guide the selection of infill samples and update the optimization. For this reason, the MFDNN model is coupled with the NSGA-II genetic algorithm to obtain intermediary estimates of the Pareto front with increasing reliability. At each iteration, the best-performing candidates are extracted from the Pareto set. The candidate offering the best compromise is recomputed with both the low- and high-fidelity solvers, hence ensuring the nested property of the datasets. The best candidates with respect to each objective are recomputed with the low-fidelity solver only. Similarly to the Bayesian approach, additional low-fidelity samples can be selected by maximizing the minimal Euclidean distance. Once the specified number of iterations is reached, an extra run of NSGA-II is performed to densify the Pareto front.

Algorithm 2 Non-Bayesian adaptive infill optimization

Compute low- and high-fidelity initial DOEs
Train the MFDNN model
for $i = 1$ to n_{iter} **do**
  Run NSGA-II with the MFDNN model
  x_{hf} taken as the estimated Pareto center
  x_{lf} taken as the estimated Pareto ends and center
  for $j = 1$ to n_{lf} **do**
    $X_{lf} = X_{lf} \bigcup x_{lf}$
    $x_{lf} = x_{lf} \bigcup \underset{x}{\operatorname{argmax}} \left\{ \min\, d(x, X_{lf}) \right\}$
  end for
  Compute y_{hf} and y_{lf}
  Re-train the MFDNN model
end for
Run NSGA-II with the MFDNN model

Pareto front evaluation metrics

In the Results and Discussion section, the performances of different optimization strategies are compared in the light of three multi-objective metrics.

The inverted generational distance (IGD) is defined as [65]:

$$IGD(A) = \frac{1}{|Z|} \left(\sum_{j=1}^{|Z|} \hat{d}_j^p \right)^{1/p}, \tag{9}$$

where Z is the reference Pareto front, A is an experiment resulting Pareto set and $\hat{d}_j$ is the Euclidean distance from $z_j \in Z$ to its nearest objective vector A. The inverted generational distance plus (IGD+) is defined as [65]:

[4] *Pytorch* is an open source Python library hosted on github (see https://pytorch.org/).

$$IGD^{+}(A) = \frac{1}{|Z|}\left(\sum_{j=1}^{|Z|} d_j^{+2}\right)^{1/2}, \tag{10}$$

where $d_i^{+} = \max(a_i - z_i, 0)$ is the modified distance from $a_i \in A$ to its nearest reference point $z_i \in Z$. A major difference between these two metrics is that $IGD(\cdot)$ is not Pareto compliant whereas $IGD^{+}(\cdot)$ is weakly Pareto compliant [69]. Weak Pareto compliance means that if A_1 dominates A_2, then $IGD^{+}(A_1) < IGD^{+}(A_2)$ [69]. The last indicator is the hypervolume (HV), a quantity that measures the volume of objective space dominated by the Pareto front, bounded by a reference point (here, the baseline configuration).

Geometry parametrization

The geometry parametrization is a key component of the optimization strategy, since it determines the number of design variables, and hence the dimension of the search space. In this study, we focus on optimizing two-dimensional airfoil profiles for turbomachinery configurations, which may require a large number of design variables to control various engineering parameters. Several parametrization techniques have been considered in the literature to strike a compromise between flexibility and parsimony of the representation (see for example [70]). In the aerodynamic optimization use case considered later in this paper, we build on the work of the authors in [71, 72], who used Free-Form deformation (FFD) to describe a wide variety of turbine blade sections efficiently using a small number of control parameters.

First introduced by Sederberg in 1986 [48], FFD consists in embedding a solid geometry within a hull volume parametrized by control points. These points are then perturbed to deform the volume which results in geometric changes of the solid geometry. A 2D FFD of the DLR low Reynolds number (LRN) compressor cascade optimized in the Results and Discussion section is illustrated in Fig. 3, using 8 control points. Although this parametrization may seem overly simple and no convergence study was performed,

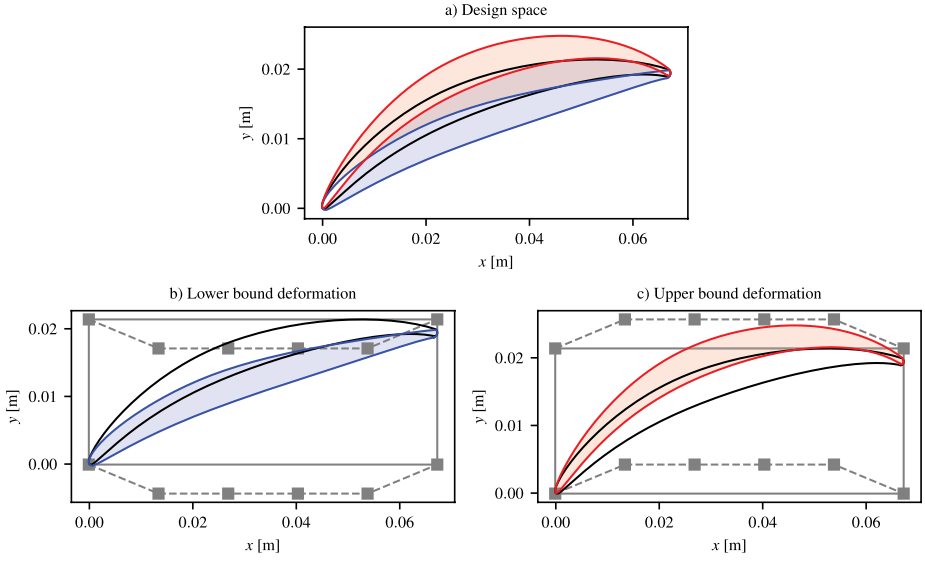

Fig. 3 LRN cascade design space (**a**) and extreme deformations (**b**-**c**) for 8 FFD control points

the variety of profiles covered by the Pareto front obtained with the brute-force approach (see CFD-based single-fidelity optimizations) supports the hypothesis of its sufficiency.

Problem dimension reduction

To reduce the number of design points even further, the geometrical space is also subjected to a Proper Orthogonal Decomposition (POD).

Starting off from a baseline geometry made of N_p coordinates $(x_b, y_b) \in \mathbb{R}^{N_p \times 2}$, N_s deformation samples are produced with a latin hypercube sampling (LHS) and used to build a database matrix:

$$S = [y_1, \ldots, y_{N_s}] \in \mathbb{R}^{N_p \times N_s}.$$

Then, the fluctuation matrix F is computed from S and $\bar{S}$ as:

$$F = S - \bar{S} \in \mathbb{R}^{N_p \times N_s}, \text{ with } \bar{S} = \frac{1}{N_p} \sum_{i=1}^{N_p} y_i.$$

Next, the correlation matrix $\mathbb{C}$ of shape $\mathbb{R}^{N_s \times N_s}$ is computed from:

$$\mathbb{C} = F^t F,$$

and yields the following eigenvalue problem:

$$\mathbb{C} V = \Lambda V.$$

From the solution V, the eigenfunctions matrix Φ is simply given by:

$$\Phi = F V \in \mathbb{R}^{N_p \times N_s}.$$

The reduced designs $S^\star$ are finally given by:

$$S^\star = \bar{S} + \Phi^\star V^{-1\star},$$

with $\Phi^\star \in \mathbb{R}^{N_p \times d^\star}$ and $V^{-1\star} \in \mathbb{R}^{d^\star \times N_s}$. At that point, $V^{-1\star}$ is the matrix:

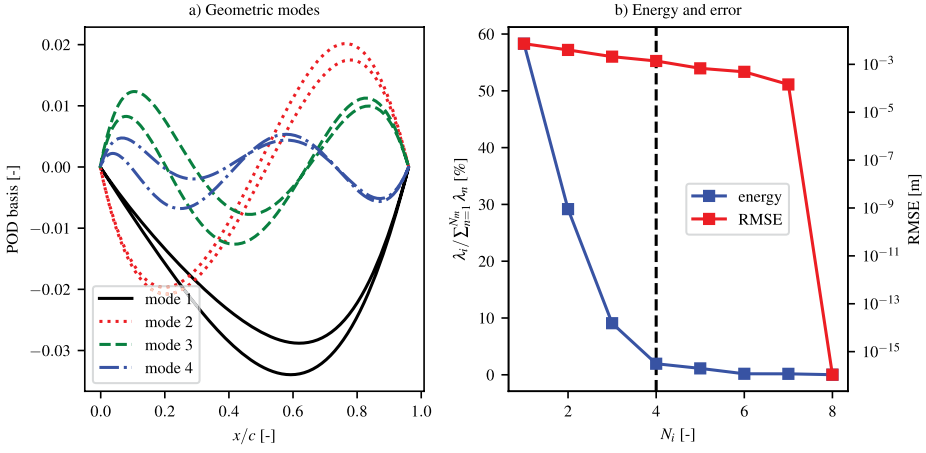

Fig. 4 LRN cascade POD 4 principal geometric modes (**a**), energy and reconstruction error as a function of the number of modes (**b**). The black dashed line indicates the selected mode number $N_m = 4$ which conserves 98.5% of the modal energy

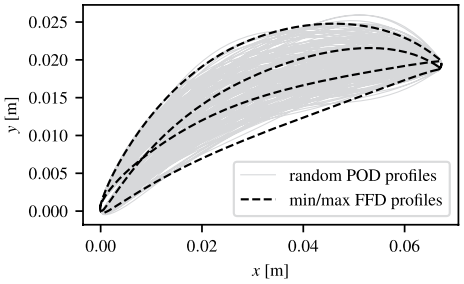

Fig. 5 Comparison between the FFD extreme profiles and 100 random reduced profiles sampled via LHS

$$V^{-1\star} = [\alpha_1, \ldots, \alpha_{N_s}] \in \mathbb{R}^{d^\star \times N_s},$$

where each modal coefficient column α_i is associated to a reduced profile:

$$y_i^\star = y_i + \epsilon = \bar{S} + \phi^\star \, \alpha_i, \tag{11}$$

and ϵ denotes reconstruction error. From this last equation, it is hence possible to compute the reduced parametrization α corresponding to any given profile y:

$$\alpha \simeq (\phi^{\star t}\phi^\star)^{-1}\phi^{\star t}\, y - (\phi^{\star t}\phi^\star)^{-1}\phi^{\star t}\, \bar{S}. \tag{12}$$

The POD decomposition maps the initial d-dimensional space, which is defined by the FFD control parameters, onto a reduced latent space of dimension $d^\star < d$. For the LRN cascade geometry in question, the geometric modes, the energy and the reconstruction error are represented in Fig. 4.

It is worth noting that although FFD control point displacements have a physical meaning, the modal coefficients in the projected space do not. An immediate consequence is that the reduced design variables are harder to interpret. Another consequence is that the extreme profiles are not conserved in the reduced space (see Fig. 5). This aspect can be particularly problematic if the meshing and simulation robustness are limited to strict deformation ranges but it is not the case in our context, but overall, the wide variety of profiles recovered with this simplification suggests that it is capable of capturing the complete behavior of the objective function.

CFD solver and feature-based anisotropic mesh adaptation

Throughout this work, both low- and high-fidelity solutions are computed with the CFD solver *wolf*, a vertex-centered mixed finite-volume/finite-element Navier-Stokes solver for both 2D or 3D unstructured meshes. *Wolf* is designed for anisotropic metric-based mesh adaptation. It can then be coupled with *feflo.a*, a generic purpose adaptive metric-based mesh generator, to generate adapted 2D or 3D meshes. A synthetic description of *wolf* and *feflo.a* is available in [73] and a detailed description is provided in [61]. Hereafter, we briefly recall the mesh adaptation strategy.

Starting off from a mesh $\mathcal{H}$, the goal of mesh adaptation is to compute an optimal mesh $\mathcal{H}_{opt}$ by minimizing a given error estimate E, for a fixed number of elements $\mathcal{N}_e$:

$$\mathcal{H}_{opt} = \underset{\mathcal{N}_e}{\operatorname{argmin}}\, E(\mathcal{H}). \tag{13}$$

The metric-based mesh adaptation approach recasts the discrete mesh $\mathcal{H}$ into a continuous Riemannian metric space where the continuous reformulation of Equation 13 can

be solved analytically for a fixed continuous mesh complexity $\mathcal{N}_c \propto \mathcal{N}_e$. In the specific case of feature-based mesh adaptation, the goal is to minimize the interpolation error of a given "sensor field". This problem can be solved analytically in the metric space as a function of the Hessian of the sensor field [73]. In the present computations we used the Mach number as a sensor because it guarantees the convergence of the solution and automatically captures complex flow features such as boundary layers, wakes and shock waves [61]. Details of the mesh adaptation algorithm can be found in the same reference.

The *wolf* solver uses the Spalart-Allmaras (SA) RANS model [74] to evaluate the turbulent stresses. However, RANS models are not well suited to capture laminar/turbulent transition phenomena that often occur in turbomachinery configurations. For this reason we consider the SA-BCM version, which takes into account the intermittency function γ, a local indicator of whether the flow should be considered as turbulent or not [75, 76]. As opposed to more sophisticated transition models, this formulation is algebraic and does not require the resolution of additional transport equation, while still providing reasonably accurate results, as shown in Results and Discussion.

Results and discussion

In the following we apply the two proposed multi-fidelity and multi-objective strategies to the ASO of a low Reynolds number (LRN) linear cascade of outlet guiding vane (OGV) blades. The system has been extensively investigated at the German aerospace center (DLR) [77, 78], and tested in the transonic cascade wind tunnel of DLR in Cologne. This use-case has been recently revived as part of the SciFi-Turbo[5] European research project.

Optimization problem description

The LRN-OGV use case aims at simultaneously optimizing the cascade efficiency for three operating points: the aerodynamic design point (ADP), whose flow conditions are given in Table 2, the operating point 1 (OP1) and the operating point 2 (OP2) with incidence angle variations of respectively +/−5° with respect to the ADP incidence angle. As such, these operating points are representative of near-stall and near-choking conditions. In addition, several geometric constraints are applied to make sure that the selected design can be manufactured. With these requirements, the optimization problem writes:

$$\text{minimize } \{w_{ADP}, w_{OP}\}, \qquad \text{with} \quad w_{OP} = \frac{1}{2}\left(w_{OP1} + w_{OP2}\right), \tag{14}$$

where w is the pressure loss coefficient defined as:

Table 2 LRN compressor cascade flow conditions at ADP

Inlet mach number	$\boldsymbol{M_1 = 0.60}$
Inlet Reynolds number	$Re = 1.5 \times 10^5$
Inlet flow angle	$\beta = 133°$
Inlet total pressure	$P_{01} = 18\,417$ Pa
Outlet static pressure	$P_2 = 16\,258$ Pa
Chord	$c = 70$ mm

[5] The project's website is available at the following link: https://scifiturbo.eu/.

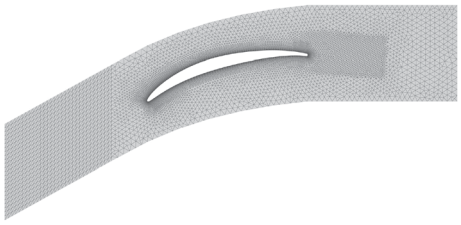

Fig. 6 LRN-OGV cascade: coarse mesh composed of $\mathcal{N}_e = 16\,000$ elements

Fig. 7 LRN-OGV cascade: adapted mesh (based on the SA-BCM model) of complexity $\mathcal{N}_c = 16\,000$ and composed of $\mathcal{N}_e = 38\,000$ cells

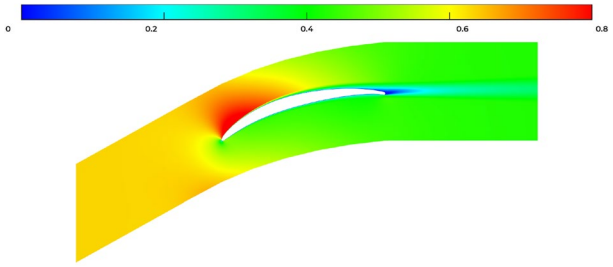

Fig. 8 LRN-OGV cascade: Mach field for the RANS SA model and coarse mesh

$$w = \frac{P_{01} - P_{02}}{P_{01} - P_1}, \tag{15}$$

with P_1 the inlet static pressure, P_{01} and P_{02} the inlet and outlet pitch-wise averaged total pressures. The baseline relative design constraints are:

$$\frac{\max(X_{th})}{c_{ax}} = \pm 20\%, \qquad \frac{\text{Area}}{c^2} = \pm 20\%, \qquad \frac{X_{cg}}{c_{ax}} = \pm 20\%,$$

where $\max(X_{th})$, c_{ax}, Area, c and X_{cg} respectively denote the axial position of maximal thickness, the axial chord, the blade area, the chord and the axial position of the center of gravity. The absolute design constraints are:

$$\frac{r_{le}}{c} > 0.5\%, \qquad \frac{r_{te}}{c} > 0.5\%,$$

where r_{le} and r_{te} correspond to the leading and trailing edge radii.

Due to the simultaneous optimization of the performance for three operating conditions, the computational cost of the objective function evaluation amounts to the execution of three simulations for each new design.

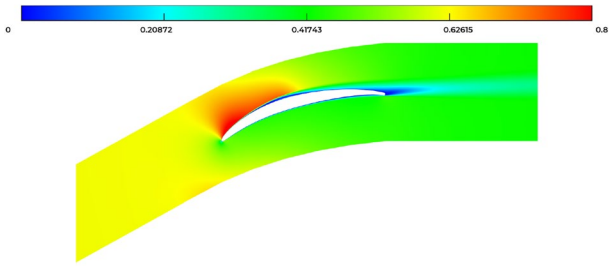

Fig. 9 LRN-OGV cascade: Mach field for the RANS SA-BCM model and adapted mesh

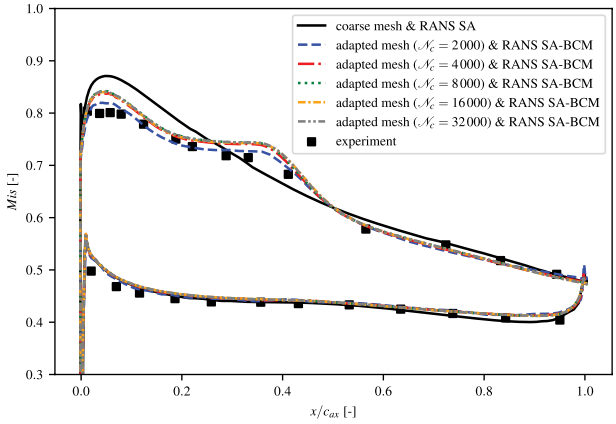

Fig. 10 Distributions of the isentropic Mach number along the blade wall for low- and high-fidelity solvers and experimental data from [77]. The high-fidelity model results are reported for various mesh complexities

Table 3 LRN compressor cascade baseline loss coefficients for the low- and high-fidelity solutions

Fidelity	Turbulence model	Mesh ($\mathcal{N}_e$)	w_{ADP} (std%)	w_{OP1} (std%)	w_{OP2} (std%)	w_{OP} (std%)
LF	RANS SA	coarse (16 000)	0.0360 (-)	0.0524 (-)	0.0411 (-)	0.0467 (-)
HF	RANS SA-BCM	adapted (38 000)	0.0403 (0.6)	0.0929 (1.8)	0.0430 (0.3)	0.0679 (1.2)

The high-fidelity results are averaged over 10 executions and std% indicates the standard deviation relative to the mean value in %

Baseline solutions

In the multi-fidelity approach, two levels of fidelity are considered as follows. The low-fidelity model corresponds to RANS simulations run on a coarse mesh automatically generated using the open source *gmsh* software[6]. The SA turbulence model available in the *wolf* code is used to account for the turbulent stresses. The high-fidelity solutions, in turn, are obtained by solving the Navier–Stokes equations supplemented with the SA-BCM model on an anisotropic adapted mesh.

For the baseline ADP conditions, the corresponding low- and high-fidelity meshes are shown in Figs.6 and 7. The corresponding Mach fields are presented in Figs. 8 and 9. The flow is characterized by a laminar separation bubble at the suction side, which is captured by the high-fidelity model but is missed by the low-fidelity one. The bubble strongly affects the distribution of the isentropic Mach number

[6] *gmsh* is an open source mesh generation software hosted on gitlab (see https://gmsh.info/).

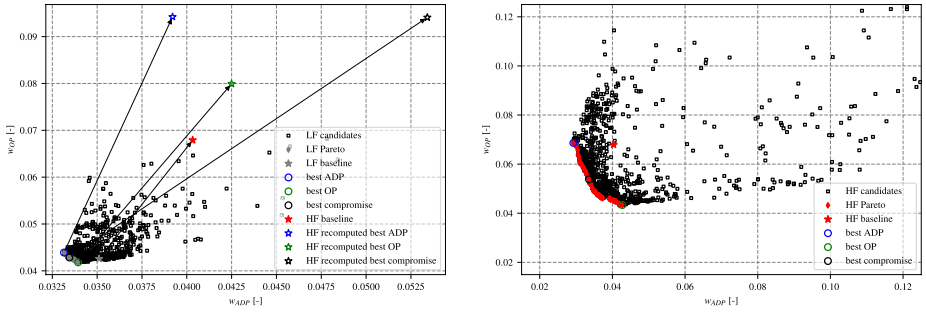

Fig. 11 Single-fidelity CFD-based Pareto fronts for the low- (left) and high-fidelity (right) models. The Pareto front designs are represented in plain diamonds; the best candidates with respect to w_{ADP} and w_{OP} are circled in blue and green; the best compromise at the center of the Pareto front is circled in black

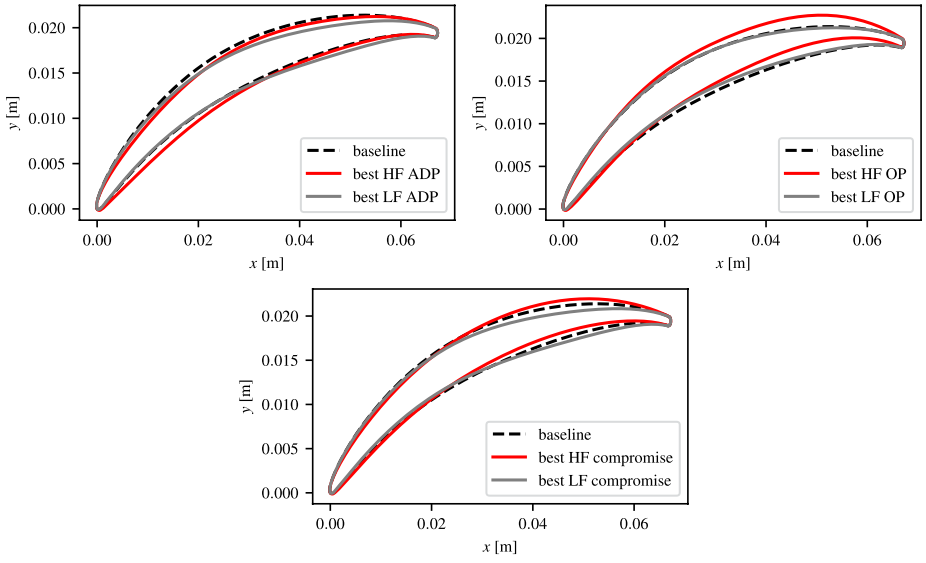

Fig. 12 Best-performing profiles with respect to w_{ADP} (top left), w_{OP} (top right) and best compromise (bottom)

$M_{is} = \left[\frac{2}{\gamma-1}\left(\left(\frac{P_{01}}{P}\right)^{\frac{\gamma-1}{\gamma}} - 1\right)\right]^{\frac{1}{2}}$ (with P the local static pressure and γ the gas isentropic exponent) along the wall, as reported in Fig. 10 where the results are compared with the measurements obtained in DLR testing facility. The complexity of the adapted grid is progressively increased until achieving grid convergence. A complexity $\mathcal{N}_c = 16\,000$ is finally deemed sufficient to obtain a mesh-converged solution.

In Table 3 we report the loss coefficients predicted by the low- and high-fidelity models. Due to the parallelism-induced non-deterministic nature of the adaptation procedure, 10 simulations are performed to estimate the uncertainty in the predicted loss coefficients. For ADP and OP2, the standard deviation is less than 1% of the mean value, while it is slightly below 2% for the near-stall condition OP1, for which the SA-BCM model tends to predict unsteady flow due to the appearance of a large separation bubble.

CFD-based single-fidelity optimizations

To investigate the effect of the CFD model on the optimization results, we first run a low- and a high-fidelity brute-force optimization using the NSGA-II optimizer with 50 generations of 20 candidates. The turn-around time of the high-fidelity optimization is

approximately 2 days on a 48 cores/96 threads workstation and about 3 hours for the low-fidelity optimization.

The results are illustrated in Fig. 11 where the final Pareto fronts after 50 generations are represented with plain diamonds, and the optimal solutions and the best compromise between the two objectives are circled. The low- and high-fidelity Pareto fronts show dramatic differences, arguing for the use of a high-fidelity model to drive the optimization. The recomputation of the optimal low-fidelity candidates with the high-fidelity model highlights the complexity of the relationship between both fidelities. Indeed, the correlation is opposite to that of the LES/RANS correlation of [27, 28] where the low-fidelity model systematically over-predicted the high-fidelity objective function values. This makes the low fidelity model particularly misleading to the optimization if not corrected with high-fidelity information.

The shape of the Pareto fronts hints that the Pareto-optimal high-fidelity candidates encompass very different geometries, whereas the tight low-fidelity Pareto front and small variations in w_{OP} suggest more similar shapes. This is confirmed by inspection of Fig. 12, which shows the optimal blade profiles for the two models. According to the high-fidelity model, the best-performing profile at nominal condition exhibits a flatter pressure side than the baseline, whereas the low-fidelity best individual mostly differ from the baseline at the rear of the suction side. Furthermore, the profiles ensuring the best off-design performance are highly cambered in the rear part and differ significantly from the baseline, which is not the case for the low-fidelity best-performing individual with respect to this criterion. Similarly, significant differences are also observed for the designs corresponding to the best compromise between the two objectives.

Surrogate-assisted multi-fidelity optimization

The LRN-OGV cascade optimization problem is then solved using the Bayesian and neural network based multi-fidelity strategies, with and without dimensionality reduction, for a total of six configurations detailed in Table 4. A preliminary study on analytical functions, reported in Appendix A, was used to select the MFDNN architecture. The latter, analogous to that of Zhang et al. [22], is given in Table 7 where the dimensions of the output layers y_L, y_H and the parameter β are extended to the number of objectives. The models are trained from initial nested DOEs including $n-1$ LHS samples and the baseline results. This minimizes the risks of having high-fidelity infill samples with poorer performance. In order to perform fair comparisons between each configuration, the same initial DOEs are used for the non-reduced optimizations (E1, E3, E5). In the same way, the initial DOEs of the two POD-based optimizations (E2, E4, E6) are

Table 4 Multi-fidelity multi-objective assisted optimization configurations

Experiments	Problem	DOEs		Total infills		Infill strategy
	dimension	LF	HF	LF	HF	(see Methodology)
E1: AR1	8	99 + 1	9 + 1	100	10	MPI, LCB & ED
E2: AR1 & POD	4	99 + 1	9 + 1	100	10	MPI, LCB & ED
E3: MFDNN	8	99 + 1	9 + 1	100	10	NSGA-II & ED
E4: MFDNN & POD	4	99 + 1	9 + 1	100	10	NSGA-II & ED
E5: AR1	8	99 + 1	9 + 1	100	10	$\alpha_{WB2S,MPI}$, LCB & ED
E6: AR1 & POD	4	99 + 1	9 + 1	100	10	$\alpha_{WB2S,MPI}$, LCB & ED

The "Total infills" column indicates the total number of infill samples of each fidelity

The number of infill steps is equal to the number of HF infills

Table 5 Comparison of the dataset Pareto mean IGD, IGD+ and HV for the different configurations after 5 and 10 infill steps over five runs

Exp.	IGD (↓)			IGD+ (↓)			HV (↑)		
	init.	final	ratio	init.	final	ratio	init.	final	ratio
Reference	-	0	-	-	0	-	-	$\mathbf{1.7 \times 10^{-4}}$	
Performance metrics after 5 infill steps									
E1	8.8×10^{-3}	$\mathbf{6.8 \times 10^{-3}}$	**1.3**	8.1×10^{-3}	$\mathbf{5.7 \times 10^{-3}}$	**1.5**	1.3×10^{-5}	2.1×10^{-5}	1.6
E2	9.0×10^{-3}	7.1×10^{-3}	1.3	8.9×10^{-3}	6.9×10^{-3}	1.4	3.9×10^{-6}	$\mathbf{3.8 \times 10^{-5}}$	**13.0**
E3	8.8×10^{-3}	8.0×10^{-3}	1.1	8.1×10^{-3}	7.5×10^{-3}	1.1	1.3×10^{-5}	1.8×10^{-5}	1.4
E4	9.0×10^{-3}	8.3×10^{-3}	1.1	8.9×10^{-3}	7.7×10^{-3}	1.3	3.9×10^{-6}	1.9×10^{-5}	5.0
E5	8.8×10^{-3}	8.6×10^{-3}	1.0	8.1×10^{-3}	7.4×10^{-3}	1.2	1.3×10^{-5}	1.6×10^{-5}	1.2
E6	9.0×10^{-3}	9.0×10^{-3}	1.0	8.9×10^{-3}	8.9×10^{-3}	1.0	3.9×10^{-6}	3.9×10^{-6}	1.0
Performance metrics after 10 infill steps									
E1	8.8×10^{-3}	5.2×10^{-3}	1.8	8.1×10^{-3}	4.0×10^{-3}	2.1	1.3×10^{-5}	5.6×10^{-5}	4.4
E2	9.0×10^{-3}	$\mathbf{4.9 \times 10^{-3}}$	**1.9**	8.9×10^{-3}	4.4×10^{-3}	2.1	3.9×10^{-6}	$\mathbf{8.4 \times 10^{-5}}$	**21.7**
E3	8.8×10^{-3}	7.2×10^{-3}	1.3	8.1×10^{-3}	5.7×10^{-3}	1.5	1.3×10^{-5}	2.6×10^{-5}	2.1
E4	9.0×10^{-3}	6.0×10^{-3}	1.6	8.9×10^{-3}	5.6×10^{-3}	1.7	3.9×10^{-6}	3.6×10^{-5}	10.3
E5	8.8×10^{-3}	7.5×10^{-3}	1.2	8.1×10^{-3}	6.7×10^{-3}	1.3	1.3×10^{-5}	1.6×10^{-5}	1.2
E6	$9. \times 10^{-3}$	5.0×10^{-3}	1.8	8.9×10^{-3}	$\mathbf{3.6 \times 10^{-3}}$	**2.5**	3.9×10^{-6}	7.0×10^{-5}	23.3

For each metric, "↓" indicates the lower the better, "↑" the opposite

The ratio is always to be maximized and in each block, the best performance is written in bold

the same and built by inverse reconstruction of the reduced parameters corresponding to the non-reduced configurations (see Problem dimension reduction). For each experience, five runs are performed.

In the end, each experiment yields two Pareto fronts. The dataset Pareto which corresponds to the one formed by the high-fidelity points, and the predicted Pareto which is given by the post-processing NSGA-II execution coupled with the final version of the surrogate model.

Dataset Pareto metrics evaluation

We begin with the analysis of the Pareto-front metrics. Table 5 summarizes the optimization results with respect to the performance metrics introduced at the end of the Methodology section. As evident from the first column of each block, it appears that in spite of our efforts to reduce the discrepancies between the initial DOEs, the cumulative effect of the POD reconstruction error and the model error in case of strong boundary layer detachment result in non-negligible objective value differences. For this reason, the POD-based configurations depart with a slightly poorer initial high-fidelity DOE in view of all performances metrics.

To get a grasp of the optimization speed, the metrics are evaluated after five and ten infill steps. After five infills, it seems that both POD-coupled models (E2, E4) offer a better exploration of the design space with a significant increase in the hypervolume below the baseline objective values. Regarding the inverse generational distance criteria, the AR1 model without data reduction (E1) yields the best IGD and IGD+ values and ratios but very close to that of the POD-coupled AR1 (E2). Given the problem sensitivity and in view of the three metrics, it appears that for a very low number of infills, the AR1-based configurations (E1, E2) are the most efficient.

At the end of the ten infills, the results indicate that the Bayesian-based strategies (E1, E2, E6) compare well and the POD-coupled versions (E2, E6) provide the best

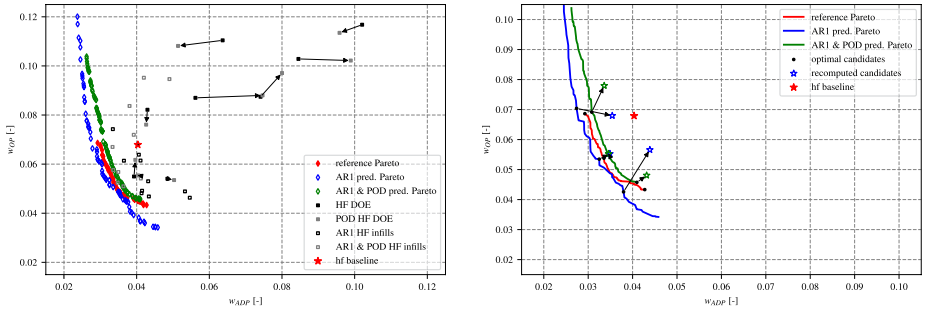

Fig. 13 Comparison of the AR1-based models optimization results (E1, E2) for one of the five runs. The predicted Pareto fronts are represented with blue (E1) and green (E2) empty diamonds. On the left side, the arrows between plain squares indicate the cumulative effect of the model and POD reconstruction errors between the initial high-fidelity DOEs. On the right side, the arrow between the empty diamonds and stars indicate the error between the actual and predicted values

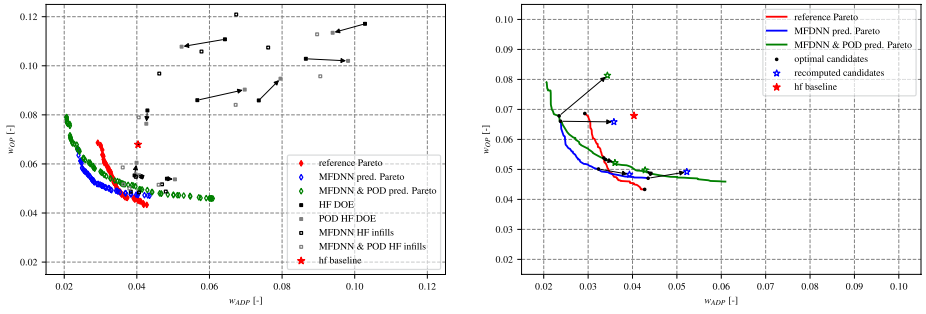

Fig. 14 Comparison of the MFDNN-based models optimization (E3, E4) results for one of the five runs. The predicted Pareto fronts are represented with blue (E3) and green (E4) empty diamonds. On the left side, the arrows between plain squares indicate the cumulative effect of the model and POD reconstruction errors between the initial high-fidelity DOEs. On the right side, the arrow between the empty diamonds and stars indicate the error between the actual and predicted values

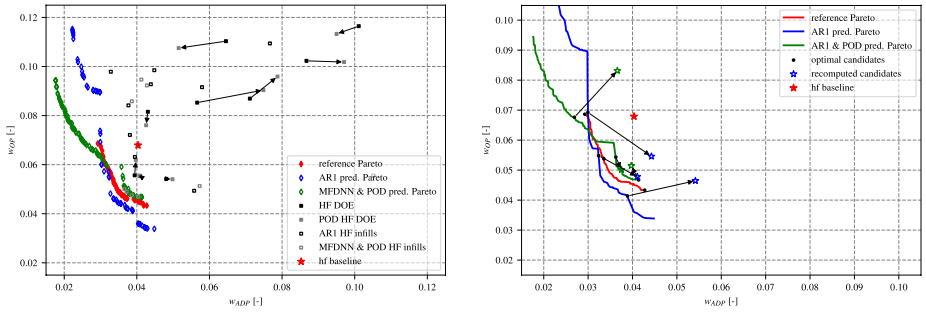

Fig. 15 Comparison of the AR1-based models and the regularized infill optimization (E5, E6) results for one of the five runs. The predicted Pareto fronts are represented with blue (E5) and green (E6) empty diamonds. On the left side, the arrows between plain squares indicate the cumulative effect of the model and POD reconstruction errors between the initial high-fidelity DOEs. On the right side, the arrow between the empty diamonds and stars indicate the error between the actual and predicted values

performances with respect to the three metrics. On average, the Bayesian strategies with dimensionality reduction (E2, E6) perform the best. However, when the optimization is performed in the full parametric space, the MFDNN approach (E3) provides performances somewhere between the two Bayesian variants (E1, E5).

Regarding the use of a regularised infill criterion, the results show no significant improvement. Regardless of the problem dimension (E5 or E6), regularisation does not

improve the optimisation speed, as all the metric ratios remain close to 1 after five infill steps. However, after ten infill steps, the POD-coupled version (E6) performs as well as the non-regularised Bayesian strategy, while the performance of the non-reduced version (E5) is slightly poorer than that obtained with the non-Bayesian strategy (E3). This suggests that the effectiveness of regularisation depends heavily on the dimensions of the problem, and that a non-Bayesian approach may be more effective in a high-dimensional design space.

Predicted Pareto evaluation

Finally for each strategy, examples of predicted Pareto obtained with a post-processing NSGA-II assisted execution are represented in Figs. 13, 14 and Fig. 15. As explained in [24], this last step can result in an unreliable predicted Pareto front if the model is not accurate enough. This word of caution is illustrated in this example where the predicted Pareto fronts extend in physically irrelevant areas of the design space. To assess the reliability of the predicted Pareto, the predicted candidates with the objective values the closest to those of the reference candidates (see Fig. 11) are extracted from the Pareto sets and recomputed with the high-fidelity model.

As expected, the results indicate for all strategies that the region of the predicted Pareto which are not supported by the nearby presence of high-fidelity samples are highly unreliable. On the contrary, in the vicinity of the reference Pareto center which is more densely populated for all POD-coupled strategies (E2, E4, E6), the discrepancy between the predicted and the recomputed values are much smaller. To get an idea of how this observation translate into the design space, the profiles of the recomputed candidates of Fig. 15 are given in Fig. 16. For the optimal candidates without POD-coupling (E5), the failure to compute high-fidelity samples near the reference Pareto results in highly inconsistent geometries with respect to the quantities of interest they are supposed to optimize. For the POD-coupled configuration (E6), the lack of high-fidelity samples in the low w_{ADP} area of the objective spaces results in a blade somewhere

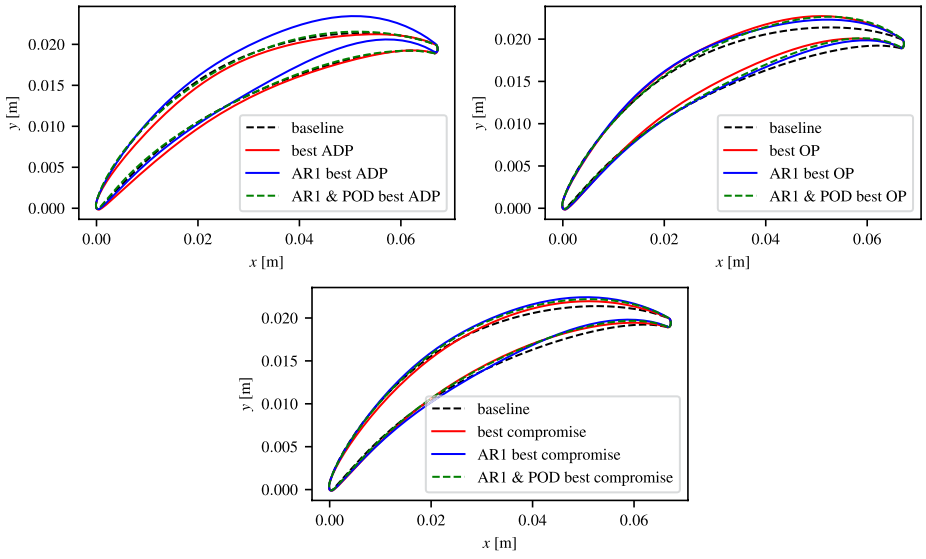

Fig. 16 Comparison of the optimal predicted profiles for one of the five runs of the AR1-based model and regularized infill optimization (E5, E6). The profiles with the lowest w_{ADP}, w_{OP} and at the center of the Pareto front are represented on the top left, top right and bottom

Table 6 Comparison of the average performances for the different configurations over five runs

Exp.	opt.	opt.	opt. trade-off
			w_{ADP}, w_{OP}
Reference	0.0293	0.0427	$0.0335, 0.0538$
	(27%)	(36.2%)	(16.8, 20.8)
E1	0.0357	**0.0459**	$0.0373, 0.0516$
	(11.6)	**(32.4)**	(7.4, 24.1)
E2	**0.0334**	0.0512	$\mathbf{0.0357}, \mathbf{0.0530}$
	(17.1)	(24.6)	**(11.6, 21.9)**
E3	0.0375	0.0484	$0.0390, 0.0521$
	(7.0)	(28.7)	(3.2, 23.2)
E4	0.0363	0.0485	$0.0384, 0.0526$
	(10.0)	(28.6)	(4.7, 22.5)
E5	0.0369	0.0472	$0.0392, 0.0547$
	(8.6)	(30.4)	(2.8, 19.5)
E6	0.0361	0.0476	$\mathbf{0.0363}, \mathbf{0.0526}$
	(10.6)	(29.9)	**(10.0, 22.5)**

The percentage of improvement with respect to the baseline values is given between parenthesis and the best performance is written in bold

between the reference optimal and the baseline geometries which leads to a smaller improvement of w_{ADP} and a higher degradation of w_{OP} than predicted. Regarding the other two optimal geometries, the agreement with the reference solution is much better.

Dataset Pareto evaluation

Going back to the initial objective which is to improve the performances of the baseline cascade, Table 6 summarizes the average improvements resulting from the high-fidelity samples of each configuration. In comparison to the reference results, all strategies prove to work remarkably well and bring noticeable improvements with respect to both quantities and especially for w_{OP}. The slighter difficulty to improve w_{ADP} comes from the fact that all five runs use the same initial DOEs which drive the optimization in the direction of better w_{OP} due to the presence of initial high-fidelity samples in that area. The quality of the candidates with the optimal objectives trade-off (i.e. located at the center of the dataset Pareto) is more variable as two (E3, E5) out of the six strategies fail to provide a compromise candidate concurrently improving both quantities of interest. Nevertheless, all POD-coupled strategies (E2, E4, E6) are highly satisfactory.

Summary

In regards of the optimization metrics (IGD, IGD+ and HV), the results indicate that, on average, the AR1 model associated to the considered Bayesian infill strategies and dimension reduction (E2, E6) performs better than its non-Bayesian counterparts (E4). Regarding the extra NSGA-II step, although it does provide a denser Pareto front estimate, its extrapolation capacities are highly dependent on the nearby presence of high-fidelity samples. Finally, for all configurations, the dimension reduction through POD-coupling proved to significantly improve the quality of the dataset Pareto in regards of both the optimization metrics and the optimized quantities of interest.

Conclusion

This study considers four approaches to addressing the main challenges of ASO under the assumptions of extreme evaluation cost imbalance and strict budget constraints. To this end, mesh adaptation, shape parametrization data reduction, multi-fidelity surrogates and adaptive infill are leveraged to solve the optimization under constraint of a low Reynolds number compressor cascade designed at DLR [77, 78]. The blade shape is parametrized with eight FFD control points and three optimization approaches are considered. Bayesian infill (with or without regularization) strategies are associated with AR1 models, while a non-Bayesian infill strategy (based on NSGA-II) is associated with MFDNN models. In addition, data reduction is performed using POD to halve the number of design variables hence yielding a total of six optimization configurations. The multi-fidelity models were trained for all configurations by combining low- and high-fidelity data, respectively obtained with the RANS SA model for coarse meshes and the transition RANS SA-BCM model for anisotropic adapted meshes. At the end of the adaptive optimization, a post-processing step which consists of running NSGA-II assisted with the final version of the surrogate model was performed as suggested in [24].

For the cascade optimization, the results show that, in regards of the considered optimization metrics (IGD, IGD+ and HV), the AR1 model associated with Bayesian approaches performs better than the MFDNN-assisted non-Bayesian optimizer, provided that the search space dimension is reduced using POD. In addition, the present conclusions hold for the very few high-fidelity sample regime targeted in the study. The NSGA-II post-processing step was found to provide a denser Pareto front; however, its effectiveness depends heavily on the presence of nearby high-fidelity samples. Therefore, it is recommended to recompute the high-fidelity solution for any geometry extracted from the predicted Pareto front. Finally, dimensionality reduction using Proper Orthogonal Decomposition (POD) proved highly effective for the Bayesian strategies, significantly enhancing results with respect to both the optimization metrics and the quantities of interest. Specifically, optimal candidates with w_{ADP} improved by 10 to 17% and optimal candidates with w_{OP} improved by 25 to 30% were obtained. Trade-off candidates that simultaneously improved both objectives were also identified, achieving up to 10% improvements in w_{ADP} and up to 22% in w_{OP}.

As RANS solvers are unable to capture the wealth of physical phenomena characteristic of turbomachinery flows, we plan as future work to assess the robustness of our proposed strategies in the context of RANS/LES multi-fidelity optimization.

Appendix A: preliminary studies with analytical test functions

Multi-fidelity surrogates evaluation: 1D test functions

Gaussian process based multi-fidelity models were compared by Brevault et al. in [18] with the aim of evaluating their strengths and weaknesses depending on the characteristics of several analytical and aerospace problems. Their study includes both simple correlation models such as AR1, and more advanced models such as the non-linear autoregressive Gaussian process and deep Gaussian process. Their results suggest that when the number of available high-fidelity samples is limited, simple models tend to perform better than more complex approaches. Therefore, to get a first idea of how MFDNN and

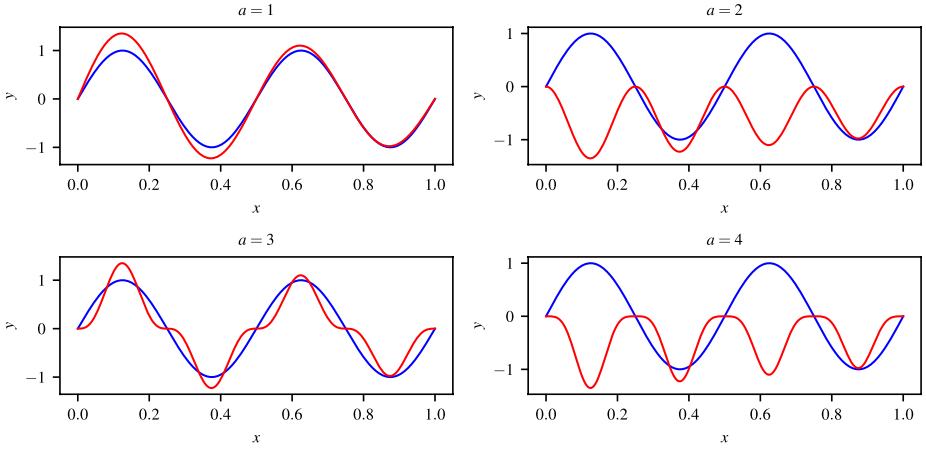

Fig. 17 1D analytical test functions from [18]. The red and blue curves respectively correspond to the high- and low-fidelity functions

Table 7 Parameters of the MFDNN architectures

Neural network	Layers	Neurons per layer	Activation	Weight decay
MFDNN Zhang et al. [22]				
NN_L	6	32	Tanh	10^{-4}
NN_{H1}	1	16	-	-
NN_{H2}	2	16	Tanh	10^{-4}
MFDNN Wu et al. [23]				
NN_L	6	20	Tanh	10^{-5}
NN_{H1}	3	16	-	10^{-5}
NN_{H2}	3	16	Tanh	10^{-5}

AR1 models compare, a preliminary comparison of these surrogates is made using the 1D test functions used in [18]:

$$
\begin{aligned}
&f_{lf}(x) = \sin(4\pi x),\\
&f_{hf}(x) = \left(\frac{x}{2} - \sqrt{2}\right)\sin(4\pi x + a\pi)^a,\\
&x \in [0;1] \text{ and } a \in [\![1;4]\!].
\end{aligned}
$$

The subscripts *hf* and *lf* denote the high-fidelity function and its low-fidelity approximation, respectively. The parameter a controls the correlation between the two fidelities by introducing phase and amplitude differences, as shown in Fig. 17.

For each value of a, DOEs of respectively 10 and 40 high- and low-fidelity samples are drawn through Latin hypercube sampling (LHS) and are used for training the surrogate. The accuracy of the models is then computed by means of the root-mean squared error (RMSE) between the models and the high-fidelity solution evaluated at 1000 random points. For each configuration, the measures are repeated ten times using different training and evaluation DOEs, and the mean, as well as the minimum and maximum errors across the repetitions are reported.

In the present series of numerical experiments, the AR1 scaling factor is learned. The considered MFDNN architectures are those of Zhang et al. [22] and Wu et al. [23] as given in Table 7. For both configurations β is learned as well, the learning rate is set to 10^{-4}, the loss function is reduced with Adam optimizer [79] and the network parameters

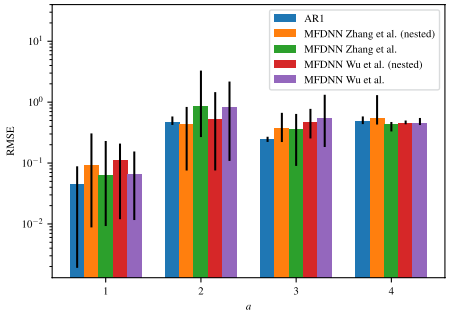

Fig. 18 RMSE of the multi-fidelity models for the 1D test functions with different values of the parameter a. Each model is trained ten times using different DOEs of respectively 40 and 10 low- and high-fidelity samples. The error bars denote the min/max RMSE registered across the repetitions

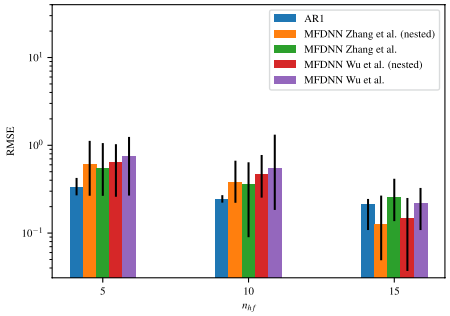

Fig. 19 RMSE of the multi-fidelity models for the 1D test function defined by $a = 3$ and different numbers of high-fidelity samples n_{hf}. Each model is trained ten times using different DOEs and the error bars denote the min/max RMSE registered across the repetitions

are initialized with Xavier's initialization method [80]. This means that for each repetition, the networks start from different initial states.

The histogram in Fig. 18 reports the mean errors obtained for the AR1 and various versions of the MFDNN, with error bars corresponding to the min/max RMSE values. For this simple 1D problem, all models perform reasonably well for any degree of correlation (related to the chosen value of a) between the low- and high-fidelity models, even if smaller errors are registered for the well-correlated case $a = 1$. Of note, the MFDNN variations around the mean RMSE are much higher than those found for the AR1, so that for all $a > 1$, there is at least one configuration for which the MFDNN outperforms AR1. However, the two models do not differ much on average.

Considering the non-linearly correlated functions given by $a = 3$, Fig. 19 shows that the MFDNN model tend to perform better than the AR1 when the number of high-fidelity point increases, but not in the low high-fidelity sample regime targeted in our study. The results also suggest that using nested DOEs improves the accuracy performances of the MFDNN.

Evaluation of surrogate-based optimization using Bayesian and non-Bayesian adaptive infills

In this section, we carry out a preliminary assessment of the proposed optimization strategies for two test problems extracted from the popular so-called ZDT test suite initially

Table 8 Averaged value and standard deviation (between parenthesis) of IGD and IGD+ obtained for ten repetitions.

Experiments	Infill strategy	ZDT1		ZDT2	
	(see Methodology)	IGD	IGD+	IGD	IGD+
E1: AR1	MPI, LCB & ED	1.20	1.19	1.89	1.86
		(0.53)	(0.53)	(0.72)	(0.74)
E2: MFDNN	NSGA-II & ED	0.85	0.73	0.69	0.55
		(0.38)	(0.42)	(0.31)	(0.29)
E3: AR1	$\alpha_{WB2S,MPI}$, LCB & ED	**0.12**	**0.083**	**0.25**	**0.14**
		(0.026)	(0.018)	(0.11)	(0.086)

The best performances are written in bold

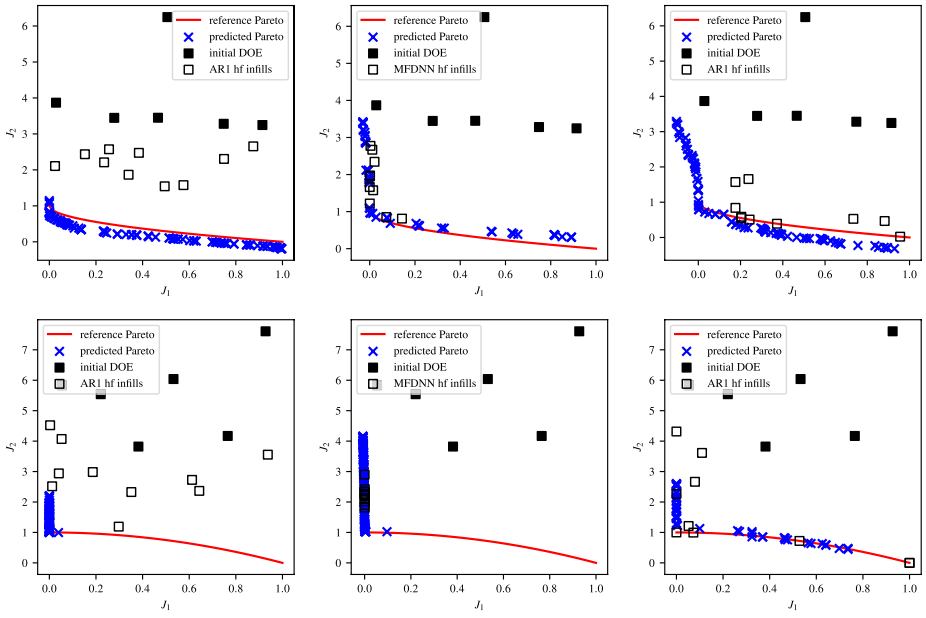

Fig. 20 One realization of the ten ZDT1 (top) and ZDT2 (bottom) optimizations for E1 (left), E2 (center) and E3 (right)

proposed in [81] for bi-objective optimization algorithms. Specifically, we focus on the ZDT1 and ZDT2 cases adapted to multi-fidelity optimization.

Given two objective functions J_1 and J_2, the ZDT1 problem is defined with the following low and high-fidelity functions [24]:

$$(J_1) \quad \begin{cases} f_{hf}(x) = x_1 \\ f_{lf}(x) = 0.9x_1 + 0.1 \end{cases} \qquad (J_2) \quad \begin{cases} f_{hf}(x) = u(x)v(x) \\ f_{lf}(x) = (0.8u(x) - 0.2)(1.2v(x) + 0.2) \\ u(x) = 1 + \frac{9}{d-1}\sum_{i=2}^{d} x_i \\ v(x) = 1 - \sqrt{\frac{x_1}{u(x)}} \end{cases}$$

where d is the problem dimension and $x \in \mathbb{R}^d$. This problem has a convex analytical Pareto front given by the following Pareto set:

$$0 \leq x_1 \leq 1 \text{ and } x_i = 0 \text{ for } i \in [\![2; d]\!]$$

The ZDT2 problem is given by the following functions [24]:

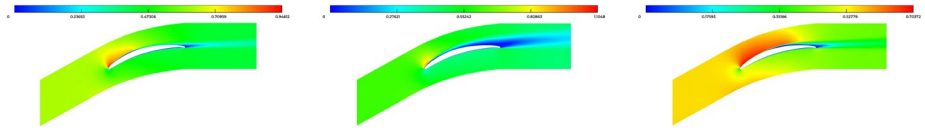

Fig. 21 ADP (left), OP1 (center), OP2 (right) Mach fields of the candidate with minimal w_{ADP}. Loss coefficients values are: $w_{ADP} = 0.02925$, $w_{OP1} = 0.08938$ and $w_{OP2} = 0.04789$

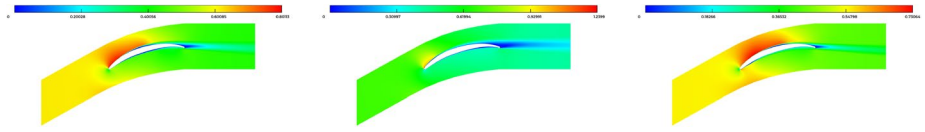

Fig. 22 ADP (left), OP1 (center), OP2 (right) Mach fields of the candidate with minimal w_{OP}. Loss coefficients values are: $w_{ADP} = 0.04274$, $w_{OP1} = 0.04617$ and $w_{OP2} = 0.04049$

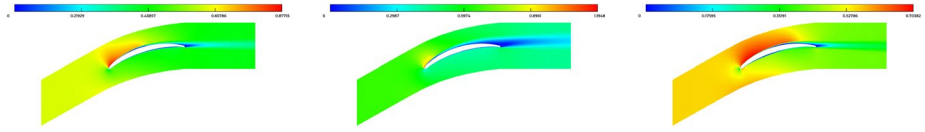

Fig. 23 ADP (left), OP1 (center), OP2 (right) Mach fields of the candidate with the best compromise. Loss coefficients values are: $w_{ADP} = 0.03354$, $w_{OP1} = 0.06473$ and $w_{OP2} = 0.04286$

$$(J_1) \quad \begin{cases} f_{hf}(x) = x_1 \\ f_{lf}(x) = 0.8x_1 + 0.2 \end{cases} \qquad (J_2) \quad \begin{cases} f_{hf}(x) = u(x)v(x) \\ f_{lf}(x) = (0.9u(x) + 0.2)(1.1v(x) - 0.2) \\ u(x) = 1 + \frac{9}{d-1}\sum_{i=2}^{d} x_i \\ v(x) = 1 - \left(\frac{x_1}{u(x)}\right)^2 \end{cases}$$

It has a concave analytic Pareto front given by the same Pareto set as that of problem ZDT1.

Both problems are solved with their dimension d set to 6 and the size of the initial low- and high-fidelity DOEs are respectively 12 and 6. The number of infill step n_{iter} is 10 and at each step, a single high-fidelity and $n_{lf} = 10$ low-fidelity samples are computed. Table 8 summarizes the results obtained with the optimization approaches introduced in the Methodology section. Only the IGD and IGD+ criteria are evaluated for the dataset Pareto and for each configuration, the results are averaged over ten realizations computed with distinct seeds. For each experiment, one realization is represented in Fig. 20.

Table 8 shows that for both problems, the Bayesian strategy with the regularized infill criterion performs the best. Indeed, as illustrated in Fig. 20, although the non-regularized Bayesian strategy provides relevant infill samples, they are still distant to the reference Pareto front. This is inherent to the MPI acquisition function which does not quantify the infill improvement itself but only its probability to be an improvement with respect to the other high-fidelity samples. With regularization, the Bayesian strategy systematically manages to give infill samples on the Pareto front. On average, this approach improves the Pareto-front metrics by an order of magnitude with respect to its non-regularized counterpart. On the other hand, the non-Bayesian strategy yields results somewhere

between the Bayesian strategies (see Table 8). But, as evident from Fig. 20, the NSGA-II algorithm tends to vertically extend the Pareto front on the left-side of the domain (i.e. for $J_1 \rightarrow 0$) in both cases. Since the non-Bayesian approach fully relies on NSGA-II runs, this results in the computation of high-fidelity infill samples in this particular area and a poor overall coverage of the Pareto front.

Based on these preliminary results, we retain the AR1 model with regularized Bayesian infill strategy and the MFDNN model with non-Bayesian infill strategy as the two most promising approaches to be further investigated for a more realistic ASO problem.

Appendix B: optimal candidates mach fields

This section presents the three operating points Mach fields for the best candidates selected on the reference Pareto front. Figure 21 for the candidate with the lowest w_{ADP}, Fig. 22 for the candidate with the lowest w_{OP} and Fig. 23 for the candidate at the center of the Pareto front which offers the best compromise between each objective.

Acknowledgements
The authors gratefully acknowledge Frédéric Alauzet from INRIA Saclay for giving us access to ***wolf*** and ***feflo.a***, and for his help in using them.

Author contributions
All authors conceived and designed the analysis. M. S. implemented the framework and performed the analysis. All authors contributed to the writing. All authors read and approved the final manuscript.

Funding
This work was supported by European Union funding under grant number 101138080 (Project Sci-Fi-Turbo). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Data availability
The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publicartion
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 1 September 2025 / Accepted: 4 November 2025
Published online: 18 December 2025

References

1. Le Clainche S, Ferrer E, Gibson S, Cross E, Parente A, Vinuesa R. Improving aircraft performance using machine learning: a review. Aerosp Sci Technol. 2023;138:108354. https://doi.org/10.1016/j.ast.2023.108354.
2. Afonso F, Sohst M, Diogo CMA, Rodrigues SS, Ferreira A, Ribeiro I, et al. Strategies towards a more sustainable aviation: a systematic review. Prog Aerosp Sci. 2023;137:100878. https://doi.org/10.1016/j.paerosci.2022.100878.
3. Skinner SN, Zare-Behtash H. State-of-the-art in aerodynamic shape optimisation methods. Appl Soft Comput. 2018;62:933–62. https://doi.org/10.1016/j.asoc.2017.09.030.
4. Li Z, Zheng X. Review of design optimization methods for turbomachinery aerodynamics. Prog Aerosp Sci. 2017;93:1–23. https://doi.org/10.1016/j.paerosci.2017.05.003.
5. Nadarajah SK, Jameson A. Optimum shape design for unsteady flows with time-accurate continuous and discrete adjoint method. AIAA J. 2007;45(7):1478–91.
6. Müller J-D, Cusdin P. On the performance of discrete adjoint cfd codes using automatic differentiation. Int J Numer Methods Fluids. 2005;47(8–9):939–45.
7. Margossian CC. A review of automatic differentiation and its efficient implementation. WIREs Data Min Knowl Discov. 2019;9(4):1305.
8. Chung SW, Freund JB. An optimization method for chaotic turbulent flow. J Comput Phys. 2022;457:111077.

9. Karbasian HR, Vermeire BC. Sensitivity analysis of chaotic dynamical systems using a physics-constrained data-driven approach. Phys Fluids. 2022. https://doi.org/10.1063/5.0076074.
10. Fidkowski, K.: A practical approach for computing sensitivities in chaotic turbulent flows. In: Cambridge Unsteady Flow Symposium, 2024; pp. 87–102. Springer
11. Ampellio E, Bertini F, Ferrero A, Larocca F, Vassio L. Turbomachinery design by a swarm-based optimization method coupled with a cfd solver. Adv Aircraft Spacecr Sci. 2016;3(2):149.
12. Karbasian HR, Vermeire BC. Gradient-free aerodynamic shape optimization using large eddy simulation. Comput Fluids. 2022;232:105185.
13. Li J, Du X, Martins JRRA. Machine learning in aerodynamic shape optimization. Prog Aerosp Sci. 2022;134:100849. https://doi.org/10.1016/j.paerosci.2022.100849.
14. Martins JRRA. Aerodynamic design optimization: challenges and perspectives. Comput Fluids. 2022;239:105391. https://doi.org/10.1016/j.compfluid.2022.105391.
15. Forrester A, Sobester A, Keane A. Engineering Design Via Surrogate Modelling: a Practical Guide. John Wiley & Sons; 2008.
16. Forrester AI, Keane AJ. Recent advances in surrogate-based optimization. Prog Aerosp Sci. 2009;45(1–3):50–79.
17. Le Gratiet L, Garnier J. RECURSIVE CO-KRIGING MODEL FOR DESIGN OF COMPUTER EXPERIMENTS WITH MULTIPLE LEVELS OF FIDELITY. Int J Uncertain Quantif. 2014;4(5):365–86. https://doi.org/10.1615/Int.J.UncertaintyQuantification.2014006914.
18. Brevault L, Balesdent M, Hebbal A. Overview of Gaussian process based multi-fidelity techniques with variable relationship between fidelities, application to aerospace systems. Aerosp Sci Technol. 2020;107:106339. https://doi.org/10.1016/j.ast.2020.106339.
19. Fernández-Godino MG. Review of multi-fidelity models. Adv Comput Sci Eng. 2023;1(4):351–400.
20. Keane AJ. Statistical improvement criteria for use in multiobjective design optimization. AIAA J. 2012;4:44.
21. Fuhg JN, Fau A, Nackenhorst U. State-of-the-art and comparative review of adaptive sampling methods for kriging. Arch Comput Methods Eng. 2021;28:2689–747.
22. Zhang X, Xie F, Ji T, Zhu Z, Zheng Y. Multi-fidelity deep neural network surrogate model for aerodynamic shape optimization. Comput Methods Appl Mech Eng. 2021;373:113485. https://doi.org/10.1016/j.cma.2020.113485.
23. Wu X, Zuo Z, Ma L, Zhang W. Multi-fidelity neural network-based aerodynamic optimization framework for propeller design in electric aircraft. Aerosp Sci Technol. 2024;146:108963. https://doi.org/10.1016/j.ast.2024.108963.
24. Charayron R, Lefebvre T, Bartoli N, Morlier J. Towards a multi-fidelity & multi-objective Bayesian optimization efficient algorithm. Aerosp Sci Technol. 2023;142:108673. https://doi.org/10.1016/j.ast.2023.108673.
25. Mourousias N, Marinus BG, Runacres MC. A novel multi-fidelity optimization framework for high-altitude propellers. Aerosp Sci Technol. 2024;153:109407. https://doi.org/10.1016/j.ast.2024.109407.
26. Leusink D, Alfano D, Cinnella P. Multi-fidelity optimization strategy for the industrial aerodynamic design of helicopter rotor blades. Aerosp Sci Technol. 2015;42:136–47.
27. Matar C. Analysis of loss mechanisms inside Organic Rankine Cycle turbines and multi-fidelity optimization with LES and RANS coupling. PhD thesis, Sorbonne Université (October 2024)
28. Matar C, Cinnella P, Gloerfelt X. Cost-effective multi-fidelity strategy for the optimization of high-reynolds number turbine flows guided by les. arXiv preprint (2025) https://doi.org/10.1016/j.ast.2025.110426
29. Cinnella P, Congedo PM. Convergence behaviours of genetic algorithms for aerodynamic optimisation problems. Int J Eng Syst Model Simul. 2013;5(4):197. https://doi.org/10.1504/IJESMS.2013.056782.
30. Alauzet F, Loseille A. A decade of progress on anisotropic mesh adaptation for computational fluid dynamics. Comput Aided Des. 2016;72:13–39. https://doi.org/10.1016/j.cad.2015.09.005.
31. Chen G, Fidkowski K. Output-Based Mesh Adaptation for Variable-Fidelity Multipoint Aerodynamic Optimization. In: AIAA Aviation 2019 Forum. American Institute of Aeronautics and Astronautics, Dallas, Texas (2019). https://doi.org/10.2514/6.2019-3057 . Accessed 2024-09-26
32. Chen G, Fidkowski KJ. Variable-fidelity multipoint aerodynamic shape optimization with output-based adapted meshes. Aerosp Sci Technol. 2020;105:106004. https://doi.org/10.1016/j.ast.2020.106004.
33. John A, Vivarelli G, Qin N, Shahpar S. Using Feature-Based Mesh Adaptation to Improve the Adjoint Optimisation of Transonic Compressor Blades. In: Volume 2C: Turbomachinery, pp. 02–35036. American Society of Mechanical Engineers, Virtual, Online (2020). https://doi.org/10.1115/GT2020-15351 . https://asmedigitalcollection.asme.org/GT/proceedings-abstract/GT2020/84089/V02CT35A036/1094562 Accessed 2024-10-15
34. Alauzet F, Frazza L. Feature-based and goal-oriented anisotropic mesh adaptation for RANS applications in aeronautics and aerospace. J Comput Phys. 2021;439:110340. https://doi.org/10.1016/j.jcp.2021.110340.
35. Li K, Li F. Multi-Fidelity Methods for Optimization: A Survey. arXiv. arXiv:2402.09638 [cs] (2024). http://arxiv.org/abs/2402.09638 Accessed 2024-09-24
36. Kennedy M. Predicting the output from a complex computer code when fast approximations are available. Biometrika. 2000;87(1):1–13. https://doi.org/10.1093/biomet/87.1.1.
37. Perdikaris P, Raissi M, Damianou A, Lawrence ND, Karniadakis GE. Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling. Proceedings of the Royal Society A Mathematical Physical and Engineering Sciences. 2017;473(2198):20160751. https://doi.org/10.1098/rspa.2016.0751.
38. Meng X, Karniadakis GE. A composite neural network that learns from multi-fidelity data: application to function approximation and inverse PDE problems. J Comput Phys. 2020;401:109020. https://doi.org/10.1016/j.jcp.2019.109020.
39. Meng X, Babaee H, Karniadakis GE. Multi-fidelity Bayesian neural networks: algorithms and applications. J Comput Phys. 2021;438:110361. https://doi.org/10.1016/j.jcp.2021.110361.
40. Cutajar, K., Pullin, M., Damianou, A., Lawrence, N., González, J.: Deep Gaussian Processes for Multi-fidelity Modeling. arXiv. arXiv:1903.07320 [cs, stat] (2019). http://arxiv.org/abs/1903.07320 Accessed 2024-09-26
41. Howard AA, Perego M, Karniadakis GE, Stinis P. Multifidelity deep operator networks for data-driven and physics-informed problems. J Comput Phys. 2023;493:112462. https://doi.org/10.1016/j.jcp.2023.112462.
42. Bhola S, Pawar S, Balaprakash P, Maulik R. Multi-fidelity reinforcement learning framework for shape optimization. J Comput Phys. 2023;482:112018. https://doi.org/10.1016/j.jcp.2023.112018.

43. He Y, Gui Q, Luo J. An efficient parallel multi-fidelity multi-objective Bayesian optimization method and application to 3-stage axial compressor with 144 variables. Aerosp Sci Technol. 2024;150:109235. https://doi.org/10.1016/j.ast.2024.109235.
44. Zhan D, Cheng Y, Liu J. Expected improvement matrix-based infill criteria for expensive multiobjective optimization. IEEE Trans Evol Comput. 2017;21(6):956–75. https://doi.org/10.1109/TEVC.2017.2697503.
45. He Y, Sun J, Song P, Wang X. Variable-fidelity hypervolume-based expected improvement criteria for multi-objective efficient global optimization of expensive functions. Eng Comput. 2022;38(4):3663–89. https://doi.org/10.1007/s00366-021-01404-9.
46. Rahat AAM, Everson RM, Fieldsend JE. Alternative infill strategies for expensive multi-objective optimisation. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 873–880. ACM, Berlin Germany (2017). https://doi.org/10.1145/3071178.3071276 . Accessed 2024-10-17
47. Rao RV, Savsani VJ, Vakharia DP. Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems. Inf Sci. 2012;183(1):1–15. https://doi.org/10.1016/j.ins.2011.08.006.
48. Sederberg TW. Free-form deformation of solid geometric models. SIGGRAPH Comput Graph. 1986;20(4):151–60.
49. Kulfan BM. Universal parametric geometry representation method. J Aircr. 2008;45(1):142–58. https://doi.org/10.2514/1.29958.
50. Hicks RM, Henne PA. Wing design by numerical optimization. J Aircr. 2012. https://doi.org/10.2514/3.58379.
51. Wei Z, Yang A, Li J, Bauerheim M, Liem RP, Fua P. DeepGeo: Deep Geometric Mapping for Automated and Effective Parameterization in Aerodynamic Shape Optimization. https://doi.org/10.2514/6.2024-3839 .
52. Chen W, Chiu K, Fuge M. Airfoil design parameterization and optimization using Bézier generative adversarial networks. arXiv. 2020:2006.12496. https://doi.org/10.2514/1.J059317
53. Kou J, Botero-Bolívar L, Ballano R, Marino O, De Santana L, Valero E, et al. Aeroacoustic airfoil shape optimization enhanced by autoencoders. Expert Syst Appl. 2023;217:119513. https://doi.org/10.1016/j.eswa.2023.119513.
54. Constantine PG, Dow E, Wang Q. Active subspace methods in theory and practice: applications to kriging surfaces. SIAM J Sci Comput. 2014;36(4):1500–24. https://doi.org/10.1137/130916138.
55. Cinquegrana D, Iuliano E. Efficient Global Optimization of a Transonic Wing with Geometric Data Reduction. In: 35th AIAA Applied Aerodynamics Conference. American Institute of Aeronautics and Astronautics, Denver, Colorado (2017). https://doi.org/10.2514/6.2017-3057 . Accessed 2024-09-26
56. Wu X, Zhang W, Peng X, Wang Z. Benchmark aerodynamic shape optimization with the POD-based CST airfoil parametric method. Aerosp Sci Technol. 2019;84:632–40. https://doi.org/10.1016/j.ast.2018.08.005.
57. Ullah S, Nguyen DA, Wang H, Menzel S, Sendhoff B, Back T. Exploring Dimensionality Reduction Techniques for Efficient Surrogate-Assisted optimization. In: 2020 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 2965–2974. IEEE, Canberra, ACT, Australia (2020). https://doi.org/10.1109/SSCI47803.2020.9308465 . Accessed 2024-09-26
58. Hou CKJ, Behdinan K. Dimensionality reduction in surrogate modeling: a review of combined methods. Data Sci Eng. 2022;7(4):402–27. https://doi.org/10.1007/s41019-022-00193-5.
59. Van Langenhove J, Lucor D, Alauzet F, Belme A. Goal-oriented error control of stochastic system approximations using metric-based anisotropic adaptations. J Comput Phys. 2018;374:384–412.
60. Belme A, Alauzet F, Dervieux A. An a priori anisotropic goal-oriented error estimate for viscous compressible flow and application to mesh adaptation. J Comput Phys. 2019;376:1051–88.
61. Alauzet F, Frazza L, Papadogiannis D. Periodic adjoints and anisotropic mesh adaptation in rotating frame for high-fidelity RANS turbomachinery applications. J Comput Phys. 2022;450:110814. https://doi.org/10.1016/j.jcp.2021.110814.
62. Chen G, Fidkowski KJ. Discretization error control for constrained aerodynamic shape optimization. J Comput Phys. 2019;387:163–85. https://doi.org/10.1016/j.jcp.2019.02.038.
63. Wackers J, Visonneau M, Serani A, Pellegrini R, Broglia R, Diez M. Multi-Fidelity Machine Learning from Adaptive-and Multi-Grid RANS Simulations. Japan: Osaka; 2020.
64. Wackers J, Pellegrini R, Diez M, Serani A, Visonneau M. Improving active learning in multi-fidelity hydrodynamic optimization, Washington DC, USA (2022)
65. Ishibuchi H, Masuda H, Tanigaki Y, Nojima Y. Modified Distance Calculation in Generational Distance and Inverted Generational Distance. In: Gaspar-Cunha, A., Henggeler Antunes, C., Coello, C.C. (eds.) Evolutionary Multi-Criterion Optimization, vol. 9019, pp. 110–125. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-15892-1_8 . Series Title: Lecture Notes in Computer Science. Accessed 2024-10-25
66. Saves P, Lafage R, Bartoli N, Diouane Y, Bussemaker J, Lefebvre T, et al. SMT 2.0: a surrogate modeling toolbox with a focus on hierarchical and mixed variables Gaussian processes. Adv Eng Softw. 2024;188:103571. https://doi.org/10.1016/j.advengsoft.2023.103571.
67. Grapin, R., Diouane, Y., Morlier, J., Bartoli, N., Lefebvre, T., Saves, P., Bussemaker, J.H.: Constrained Multi-Objective Bayesian Optimization with Application to Aircraft Design. In: AIAA AVIATION 2022 Forum. American Institute of Aeronautics and Astronautics, Chicago, IL & Virtual (2022). https://doi.org/10.2514/6.2022-4053 . Accessed 2024-11-18
68. Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Trans Evol Comput. 2002;6(2):182–97. https://doi.org/10.1109/4235.996017.
69. Falcón-Cardona JG, Emmerich MTM, Coello CAC. On the construction of Pareto-compliant combined indicators. Evol Comput. 2022;30(3):381–408. https://doi.org/10.1162/evco_a_00307.
70. Voß C, Siggel M, Backhaus J, Goinis G, Pahs A. A differentiated geometry blade parameterization methodology for gas turbines. Comput Fluids. 2025. https://doi.org/10.1016/j.compfluid.2025.106588.
71. Bufi EA, Cinnella P. Robust optimization of supersonic orc nozzle guide vanes. In: Journal of Physics: Conference Series, 2017;821,012014. IOP Publishing
72. Serafino A, Obert B, Cinnella P. Multi-fidelity robust design optimization of an orc turbine for high temperature waste heat recovery. Energy. 2023;269:126538.
73. Alauzet F, Spalart P. A new rotation correction for the Spalart-Allmaras model to improve off-body vortex prediction and vortex-vortex interaction effects. In: AIAA SCITECH 2024 Forum. American Institute of Aeronautics and Astronautics, Orlando, FL (2024)https://doi.org/10.2514/6.2024-1335 . Accessed 2024-10-18
74. Allmaras SR, Johnson FT, Spalart PR. Modi cations and Clari cations for the Implementation of the Spalart-Allmaras Turbulence Model. Computational Fluid Dynamics (2012)

75. Cakmakcioglu SC, Bas O, Mura R, Kaynak U. A Revised One-Equation Transitional Model for External Aerodynamics. In: AIAA AVIATION 2020 FORUM. American Institute of Aeronautics and Astronautics, VIRTUAL EVENT (2020). https://doi.org/10.2514/6.2020-2706 . Accessed 2024-10-15
76. Mura R, Cakmakcioglu SC. A Revised One-Equation Transitional Model for External Aerodynamics - Part I: Theory, Validation and Base Cases. In: AIAA AVIATION 2020 FORUM. American Institute of Aeronautics and Astronautics, VIRTUAL EVENT (2020). https://doi.org/10.2514/6.2020-2714 . Accessed 2024-10-15
77. Hergt A, Hage W, Grund S, Steinert W, Terhorst M, Schongen F, et al. Riblet application in compressors: toward efficient blade design. J Turbomach. 2015;137(11):111006. https://doi.org/10.1115/1.4031090.
78. Hergt A, Pesch M, Grund S, Flamm J. Low reynolds number effects in compressor blade design. International Journal of Gas Turbine, Propulsion and Power Systems. 2024;15(4):15–408.
79. Kingma DP. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
80. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In: Teh, Y.W., Titterington, M. (eds.) Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 9, pp. 249–256. PMLR, Chia Laguna Resort, Sardinia, Italy (2010). https://proceedings.mlr.press/v9/glorot10a.html
81. Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results. Evol Comput. 2000;8(2):173–95.

Publisher's Note